

# *K2000*

## *Musician's Guide*

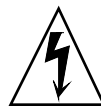
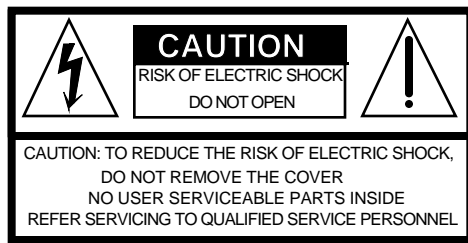
**KURZWEIL**

*Music Systems*

©1999 All rights reserved. Kurzweil is a product line of Young Chang Co.; V. A. S. T. is a registered trademark, and Kurzweil, K2vx, and K2000 are trademarks of Young Chang Co. All other products and brand names are trademarks/registered trademarks of their respective companies. Product features and specifications are subject to change without notice. Further information: <http://www.youngchang.com>

**Part Number: 910204-BIG**

---



The lightning flash with the arrowhead symbol, within an equilateral triangle, is intended to alert the user to the presence of uninsulated "dangerous voltage" within the product's enclosure that may be of sufficient magnitude to constitute a risk of electric shock to persons.



The exclamation point within an equilateral triangle is intended to alert the user to the presence of important operating and maintenance (servicing) instructions in the literature accompanying the product.

# IMPORTANT SAFETY & INSTALLATION INSTRUCTIONS

## INSTRUCTIONS PERTAINING TO THE RISK OF FIRE, ELECTRIC SHOCK, OR INJURY TO PERSONS

**WARNING** - When using electric products, basic precautions should always be followed, including the following:

1. Read all of the Safety and Installation Instructions and Explanation of Graphic Symbols before using the product.
2. This product must be grounded. If it should malfunction or breakdown, grounding provides a path of least resistance for electric current to reduce the risk of electric shock. This product is equipped with a power supply cord having an equipment-grounding conductor and a grounding plug. The plug must be plugged into an appropriate outlet which is properly installed and grounded in accordance with all local codes and ordinances.
- DANGER** - Improper connection of the equipment-grounding conductor can result in a risk of electric shock. Do not modify the plug provided with the the product - if it will not fit the outlet, have a proper outlet installed by a qualified electrician. Do not use an adaptor which defeats the function of the equipment-grounding conductor. If you are in doubt as to whether the product is properly grounded, check with a qualified serviceman or electrician.
3. **WARNING** - This product is equipped with an AC input voltage selector. The voltage selector has been factory set for the mains supply voltage in the country where this unit was sold. Changing the voltage selector may require the use of a different power supply cord or attachment plug, or both. To reduce the risk of fire or electric shock, refer servicing to qualified maintenance personnel.
4. Do not use this product near water - for example, near a bathtub, washbowl, kitchen sink, in a wet basement, or near a swimming pool, or the like.
5. This product should only be used with a stand or cart that is recommended by the manufacturer.
6. This product, either alone or in combination with an amplifier and speakers or headphones, may be capable of producing sound levels that could cause permanent hearing loss. Do not operate for a long period of time at a high volume level or at a level that is uncomfortable. If you experience any hearing loss or ringing in the ears, you should consult an audiologist.
7. The product should be located so that its location or position does not interfere with its proper ventilation.
8. The product should be located away from heat sources such as radiators, heat registers, or other products that produce heat.
9. The product should be connected to a power supply only of the type described in the operating instructions or as marked on the product.
10. This product may be equipped with a polarized line plug (one blade wider than the other). This is a safety feature. If you are unable to insert the plug into the outlet, contact an electrician to replace your obsolete outlet. Do not defeat the safety purpose of the plug.
11. The power supply cord of the product should be unplugged from the outlet when left unused for a long period of time. When unplugging the power supply cord, do not pull on the cord, but grasp it by the plug.
12. Care should be taken so that objects do not fall and liquids are not spilled into the enclosure through openings.
13. The product should be serviced by qualified service personnel when:
  - A. The power supply cord or the plug has been damaged; or
  - B. Objects have fallen, or liquid has been spilled into the product; or
  - C. The product has been exposed to rain; or
  - D. The product does not appear to be operating normally or exhibits a marked change in performance; or
  - E. The product has been dropped, or the enclosure damaged.
14. Do not attempt to to service the product beyond that described in the user maintenance instructions. All other servicing should be referred to qualified service personnel.
15. **WARNING** - Do not place objects on the product's power supply cord, or place the product in a position where anyone could trip over, walk on, or roll anything over cords of any type. Do not allow the product to rest on or be installed over cords of any type. Improper installations of this type create the possibility of a fire hazard and/or personal injury.

## RADIO AND TELEVISION INTERFERENCE

**Warning:** Changes or modifications to this instrument not expressly approved by Young Chang could void your authority to operate the instrument.

**Important:** When connecting this product to accessories and/or other equipment use only high quality shielded cables.

**Note:** This instrument has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This instrument generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this instrument does cause harmful interference to radio or television reception, which can be determined by turning the instrument off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the instrument and the receiver.
- Connect the instrument into an outlet on a circuit other than the one to which the receiver is connected.
- If necessary consult your dealer or an experienced radio/television technician for additional suggestions.

### NOTICE

This apparatus does not exceed the Class B limits for radio noise emissions from digital apparatus set out in the Radio Interference Regulations of the Canadian Department of Communications.

### AVIS

Le present appareil numerique n'emet pas de bruits radioelectriques depassant les limites applicables aux appareils numeriques de la class B prescrites dans le Reglement sur le brouillage radioelectrique edicte par le ministere des Communications du Canada.

## SAVE THESE INSTRUCTIONS

## Table of Contents

<b>Introduction</b> .....	<b>1-1</b>
Welcome! .....	1-1
Overview of the K2vx .....	1-1
What is VAST? .....	1-2
How the K2vx Works .....	1-2
How to use this manual .....	1-2
<b>Startup</b> .....	<b>2-1</b>
Basic Startup Checklist: .....	2-1
Startup—the Details .....	2-1
Before You Start... .....	2-1
The Rear Panel .....	2-2
Playing the Presets .....	2-4
Programs .....	2-4
Setups .....	2-4
Quick Access .....	2-5
Performance Controls .....	2-6
K2vx Boot Loader .....	2-6
Starting the Boot Loader .....	2-6
Boot-loader Main Menu: .....	2-6
Install System / Install Objects .....	2-7
Run System .....	2-8
Hard Reset .....	2-8
Run Diags .....	2-8
Fixed Diags .....	2-8
<b>User Interface Basics</b> .....	<b>3-1</b>
Mode Selection .....	3-1
Mode Buttons .....	3-1
Navigation .....	3-1
The Display .....	3-1
Pages .....	3-2
The Top Line .....	3-2
The Bottom Line .....	3-2
The Soft Buttons .....	3-2
The Cursor Buttons .....	3-2
Data Entry .....	3-4
The Alpha Wheel .....	3-4
The Plus/Minus Buttons .....	3-4
The Alphanumeric Pad .....	3-4
Double Button Presses .....	3-5
Intuitive Data Entry .....	3-6
Changing the Current Layer in Multi-Layer Programs .....	3-6
Search Function .....	3-6
The Panel Play Feature (K2vxR) .....	3-7

<b>The Operating Modes .....</b>	<b>4-1</b>
What the Modes Are .....	4-1
Selecting Modes .....	4-1
Using the Modes .....	4-3
<b>Editing Conventions .....</b>	<b>5-1</b>
Introduction to Editing .....	5-1
Object Type and ID .....	5-2
Saving and Naming .....	5-2
Deleting Objects .....	5-4
Memory Banks .....	5-5
Special Button Functions .....	5-6
<b>Program Mode and the Program Editor .....</b>	<b>6-1</b>
K2vx Program Structure .....	6-2
The Program Mode Page .....	6-2
What are these programs in parentheses? .....	6-2
The Soft Buttons in Program Mode .....	6-3
Using the Program Editor .....	6-4
The Soft Buttons in the Program Editor .....	6-4
Algorithm Basics .....	6-5
Common DSP Control Parameters .....	6-6
The Program Editor—Pages .....	6-9
The Algorithm (ALG) Page .....	6-9
The LAYER Page .....	6-10
The KEYMAP Page .....	6-14
The PITCH Page .....	6-16
F1–F3 Pages .....	6-17
The F4 AMP Page .....	6-17
The OUTPUT Page .....	6-18
The EFFECT Page .....	6-21
The COMMON Page .....	6-22
The Amplitude Envelope (AMPENV) Page .....	6-24
Envelopes 2 and 3 .....	6-27
The Envelope Control (ENVCTL) Page .....	6-27
The LFO Page .....	6-29
The ASR Page .....	6-30
The FUN Page .....	6-31
The VTRIG Page .....	6-32
The Function Soft Buttons .....	6-32
<b>Setup Mode and the Setup Editor .....</b>	<b>7-1</b>
Setup Mode .....	7-1
The Setup Editor .....	7-2
<b>Quick Access Mode and the</b>	
<b>Quick Access Editor .....</b>	<b>8-1</b>
<b>Effects Mode and the Effects Editor .....</b>	<b>9-1</b>
The Effects Mode Page .....	9-1

The Soft Buttons on the Effects mode Page .....	9-1
Effects Mode Parameters .....	9-2
Understanding FX Mode and FX Channel .....	9-2
Another Use for Effects Mode .....	9-5
The Effects Editor .....	9-5
Editing Effects .....	9-6
Configurations and Parameters .....	9-7
<b>MIDI Mode .....</b>	<b>10-1</b>
The Transmit (XMIT) Page .....	10-1
Physical Control Parameters .....	10-4
The Receive (RECV) Page .....	10-5
The Channels Page .....	10-8
Parameter Locks .....	10-10
Program Change Formats .....	10-10
Extended and Kurzweil Program Change Types .....	10-11
The Soft Buttons in MIDI Mode .....	10-18
<b>Master Mode .....</b>	<b>11-1</b>
The Master Mode Page .....	11-1
The Soft Buttons in Master Mode .....	11-4
Guitar/Wind Controller Mode .....	11-5
Object Utilities .....	11-6
Move .....	11-7
Copy .....	11-8
Name .....	11-9
Delete .....	11-9
Dump .....	11-10
Accessing the Object Utilities from the Editor .....	11-10
<b>Song Mode .....</b>	<b>12-1</b>
Getting Started with the Sequencer .....	12-1
A Word about the Local Keyboard Channel .....	12-1
Tutorial: Recording a song .....	12-2
.....	12-3
MAIN Page .....	12-11
Soft Buttons on the MAIN Page .....	12-15
Save this song? Dialog .....	12-16
Save New song? Dialog .....	12-17
Erase Track Dialog .....	12-17
MISC Page .....	12-18
Soft Buttons on the MISC Page .....	12-22
MIX Page .....	12-23
Soft Buttons on the MIX Page .....	12-24
The Edit Song Pages .....	12-25
Edit Song: COMMON Page .....	12-25
Parameters used with the Arrangement Feature .....	12-26
Soft Buttons on the Edit Song: COMMON Page .....	12-27

Edit Song: EVENT Page .....	12-28
Soft Buttons on the Edit Song: EVENT Page .....	12-29
Edit Song: TRACK Page .....	12-31
Common Parameters for Edit Song: Track Functions .....	12-32
Region / Criteria Window Parameters .....	12-32
Soft Buttons on the Edit Song: Track Page .....	12-33
Edit Song: Track Functions – Erase .....	12-34
Edit Song: Track Functions – Copy .....	12-34
Edit Song: Track Functions – Bounce .....	12-35
Edit Song: Track Functions – Insert .....	12-35
Edit Song: Track Functions – Delete .....	12-36
Edit Song: Track Functions – Quantize .....	12-36
Edit Song: Track Functions – Reference Quantize .....	12-37
Edit Song: Track Functions – Shift .....	12-38
Edit Song: Track Functions – Transpose .....	12-39
Edit Song: Track Functions – Change .....	12-39
Edit Song: Track Functions – Thin .....	12-40
Edit Song: Track Functions – Remap .....	12-41
Edit Song: Track Functions – Grab .....	12-41
Edit Song: STEP Page .....	12-43
Recording with the STEP editor .....	12-43
Soft Buttons on the Edit Song: STEP Page .....	12-45
Edit Song: ARRANGE Page .....	12-46
Triggering Steps from a Key .....	12-47
Soft Buttons on the Edit Song: ARRANGE Page .....	12-48
Selecting a Song for Playback .....	12-49
Effect Selection During Recording and Playback .....	12-49
Synchronizing Songs .....	12-49
Memory Limits .....	12-50
Loading Songs From Disk .....	12-50
Recording Multi-timbral Sequences via MIDI .....	12-50
<b>Disk Mode .....</b>	<b>13-1</b>
Disk Mode Page .....	13-1
Using your K2vx in a SCSI System .....	13-4
Directories .....	13-4
Path .....	13-4
Startup .....	13-5
Library .....	13-5
Disk Drive Information .....	13-5
Macro On Indicator .....	13-5
Disk Mode Soft Buttons .....	13-5
File List Dialog .....	13-6
The File Index .....	13-8
Soft Buttons in the File List Dialog .....	13-8
Total .....	13-10

---

Quick Scrolling to Subdirectories .....	13-10
Creating Directories .....	13-11
Creating a Directory with NewDir .....	13-11
Creating a Directory with Save -> NewDir .....	13-12
The Directory Selection Dialog .....	13-12
Disk Mode Functions .....	13-13
Loading Files .....	13-13
Loading Individual Objects .....	13-13
Shortcuts when Loading Objects .....	13-16
Loading Dependents of Selected Objects .....	13-17
Auditioning Samples from a Disk File .....	13-17
Loading Objects from Floppy Disk Files .....	13-18
Load Function Dialog .....	13-18
Bank Status Indicator .....	13-18
Loading Methods .....	13-18
Multiple Selection of Files to Load .....	13-21
Aborting a Multiple File Load .....	13-21
More Load Function Enhancements .....	13-22
Saving Files .....	13-22
Soft Buttons in the "Save Selection" Dialog .....	13-24
Saving Individual Objects .....	13-24
Shortcuts when Saving Objects .....	13-24
Auditioning Objects in RAM .....	13-25
Saving Dependent Objects .....	13-26
The Name Table .....	13-27
Working with Relink-by-Name .....	13-28
Not Loading the Name Table .....	13-31
Relink-by-Name Processing Time .....	13-31
The Multiple Object Selector Page .....	13-32
Multiple Object Selector Soft Buttons .....	13-33
Entering Selection Criteria in the Multiple Object Selector .....	13-34
More Features of the Save Dialog .....	13-37
Macros .....	13-38
The Macro Page .....	13-38
Macro Modes .....	13-39
The Macro Table .....	13-40
How to Make a Macro File .....	13-41
Macro Entries .....	13-45
Using the Bank and Mode Fields .....	13-46
Viewing the Object List for a Macro Entry .....	13-46
Unspecified Disk Drive ID .....	13-47
The Library Disk .....	13-47
Loading Selected Entries from a Macro File .....	13-49
Editing Macros .....	13-50
Macro Insert .....	13-53

---

Saving and Loading a Macro Table in a .KRZ file .....	13-54
Aborting a Macro Load .....	13-54
Disk Utilities .....	13-55
Find Files .....	13-56
List .....	13-57
Free .....	13-58
Moving Files Between Directories .....	13-59
Renaming Files .....	13-60
Deleting Files and Directories .....	13-61
Backup and Copy Functions .....	13-61
File Copy .....	13-63
Creating a Startup File .....	13-63
Deleting Banks in a Startup File .....	13-64
MS-DOS File System Compatibility .....	13-65
File Name Compatibility .....	13-65
Importing and Exporting Data using Standard File Formats .....	13-65
AIFF and AIFF-C Files .....	13-66
WAVE Files .....	13-66
MIDI Type 0 Files .....	13-66
<b>DSP Functions .....</b>	<b>14-1</b>
Introduction to Algorithm Programming .....	14-1
Additional Parameters .....	14-3
Filters .....	14-5
How to read the graphs .....	14-6
Equalization (EQ) .....	14-22
Pitch / Amplitude / Panner .....	14-31
Mixers .....	14-34
Waveforms .....	14-35
Added Waveforms .....	14-38
Non-linear Functions .....	14-38
Waveforms Combined with Non-linear Functions .....	14-47
Mixers with Non-linear Inputs .....	14-49
Hard Sync Functions .....	14-50
<b>Sampling and Sample Editing .....</b>	<b>15-1</b>
Setting Up For Sampling .....	15-1
Cables and Input Jacks .....	15-1
Entering The Sampler - Two Different Ways .....	15-1
Sampling Analog Signals .....	15-2
Recording Samples .....	15-5
Using the Digital Outputs .....	15-8
Editing Samples .....	15-8
The Function Soft Buttons in the Sample Editor .....	15-9
TRIM .....	15-13
The Soft Buttons on the DSP Page .....	15-17
DSP Functions .....	15-18



Crossfade and Volume Adjust Curves .....	15-30
Reading Samples .....	15-31
Akai .....	15-31
Roland .....	15-32
EPS .....	15-32
The Keymap Editor .....	15-33
The Soft Buttons in the Keymap Editor .....	15-34
The Keymap Editor Parameters .....	15-35
Building a Keymap .....	15-37
<b>FUNS .....</b>	<b>16-1</b>
The Mechanics of Control Sources .....	16-1
Programming the FUNs .....	16-2
The FUN Equations .....	16-3
The List of Equations .....	16-4
.....	16-10
Warp Equations .....	16-10
Sawtooth LFOs .....	16-13
Chaotic LFOs .....	16-13
Diode Equations .....	16-14
The Order of Evaluation for FUNs .....	16-16
<b>Other Editors .....</b>	<b>17-1</b>
The Intonation Table Editor .....	17-1
The Velocity Map Editor .....	17-3
Using the Velocity Map Editor .....	17-3
The Pressure Map Editor .....	17-6
<b>Audio Outputs .....</b>	<b>18-1</b>
Audio Configurations .....	18-1
Using the MIX Outputs .....	18-1
Using the Separate Outputs .....	18-2
Using Stereo Insert Cables .....	18-3
Output Groups and MIDI Channels .....	18-4
<b>Programming Examples .....</b>	<b>19-1</b>
Example 1 .....	19-1
Trumpet with Delayed Vibrato and Velocity-triggered Stabs .....	19-1
Example 2 .....	19-4
Lowpass Filter, Envelopes .....	19-4
Example 3 .....	19-6
Sample and Hold; Using a FUN .....	19-6
Example 4 .....	19-7
SHAPER and PANNER .....	19-7
Example 5 .....	19-9
Building a Drum Program; Using the Keymap Editor .....	19-9
<b>Front Panel .....</b>	<b>20-1</b>
Front Panel Quick Reference .....	20-1
Mode Button Functions .....	20-3

.....	20-4
<b>Programs, Setups, and Keymaps</b> .....	<b>21-1</b>
K2vx Program List .....	21-1
Setup List .....	21-9
..... Storing Objects in the Memory Banks	21-10
K2500 ROM Keymaps .....	21-11
<b>Effects</b> .....	<b>22-1</b>
List of Factory Preset Global Effects and Their Configurations .....	22-1
Effects Controller Numbers .....	22-2
<b>LFOs</b> .....	<b>23-1</b>
LFO Shapes .....	23-1
<b>Note Numbers and Intonation Tables</b> .....	<b>24-1</b>
K2vx Note Numbers and MIDI Note Numbers .....	24-1
Note Numbers for Percussion Keymaps .....	24-1
5-Octave Percussion Keymaps (C2 - C7) .....	24-1
2-Octave Percussion Keymaps (C3 - C5) .....	24-2
List and Description of Intonation Tables .....	24-3
<b>Control Sources</b> .....	<b>25-1</b>
Control Sources .....	25-1
Descriptions of Control sources .....	25-3
MIDI Control Source List .....	25-3
Main Control Source List .....	25-6
<b>DSP Algorithms</b> .....	<b>26-1</b>
<b>Memory Upgrades and Other Options</b> .....	<b>27-1</b>
Program RAM vs. Sample RAM .....	27-1
Viewing RAM Objects .....	27-1
Choosing SIMMs for Sample RAM .....	27-2
Using Headphones with the K2vx .....	27-2
<b>Maintenance and Troubleshooting</b> .....	<b>28-1</b>
Preventive Maintenance .....	28-1
Battery selection and Replacement .....	28-1
User-callable Diagnostics .....	28-2
Maximizing Music and Minimizing Noise .....	28-2
Power Problems and Solutions .....	28-4
Troubleshooting .....	28-4
Other Possible Problems .....	28-5
<b>MIDI and SCSI Sample Dumps</b> .....	<b>29-1</b>
SCSI Guidelines .....	29-1
K2vx and Macintosh Computers .....	29-2
Accessing a K2vx Internal Drive from the Mac .....	29-3
The MIDI Sample Dump Standard .....	29-3
SMDI Sample Transfers .....	29-7
<b>System Exclusive Protocol</b> .....	<b>30-1</b>
K2vx System Exclusive Implementation .....	30-1
Button Press Equivalence Table .....	30-7

---

<b>Glossary</b> .....	<b>31-1</b>
<b>Specifications</b> .....	<b>32-1</b>
K2vx FEATURES .....	32-1
Environmental Specifications .....	32-3
Physical Specifications .....	32-3
Electrical Specifications .....	32-3
MIDI Implementation Chart .....	32-4
<b>K2vx Program Farm</b> .....	<b>A-1</b>
Overview of Program Files .....	A-1
Programs Within the Files .....	A-2
ANACOMPS.K25 .....	A-3
ANALEADS.K25 .....	A-4
ANAPADS.K25 .....	A-5
BASS.K25 .....	A-6
BELLS.K25 .....	A-7
BRASS.K25 .....	A-8
DIGITAL.K25 .....	A-9
DKICKSNR.K25 .....	A-10
DRUMS.K25 .....	A-11
ENSEMBLE.K25 .....	A-12
ETHEREAL.K25 .....	A-13
FXSOUNDS.K25 .....	A-14
GUITARS.K25 .....	A-15
HYBPERC.K25 .....	A-16
HYBRIDS.K25 .....	A-17
ORGANS.K25 .....	A-18
PNOEPNO.K25 .....	A-19
STRINGS.K25 .....	A-20
VOX.K25 .....	A-21
<b>K2vx Compatibility</b> .....	<b>B-1</b>
K2vx Compatibility Files .....	B-1
Converting K2vx Files to K2vx Files .....	B-2
Converting programs from the K2vx to K2vx .....	B-3
Programs using Drum samples .....	B-3
Effects Programs .....	B-3
Keymaps .....	B-3
<b>Orchestral ROM</b> .....	<b>C-1</b>
<b>Contemporary ROM</b> .....	<b>D-1</b>

---



# Chapter 1

## Introduction

### Welcome!

Congratulations, and thanks for purchasing the Kurzweil/Young Chang K2vx! You've got your hands on an extremely capable musical instrument packed with great acoustic, electric, and synth sounds—combined with some of the most advanced synthesis features available, which you can use to create almost any sound imaginable.

### Overview of the K2vx

The K2vx has been designed to be a versatile instrument both for performance, and for multi-timbral sequencing and recording. Its Variable Architecture Synthesis Technology lets you build sounds from realistic instrumental samples and sampled synth waveforms—then modify the nature of those sounds through a dazzling array of digital signal processing (DSP) functions. The K2vx also generates its own synth waveforms, which can be combined with the samples or used on their own. The K2vx packs 8 Megabytes of on-board sound ROM, and you can load samples from disk into optional sample RAM.

Before we get into explaining Variable Architecture Synthesis, here are a few of the features that by themselves make the K2vx an impressive stage and studio machine. It's fully multi-timbral—different programs can be played on each MIDI channel. It's 48-note polyphonic, for a full sound no matter how many chords you play. There's an on-board digital effects processor, providing up to four simultaneous effects, including realtime effects control, internally or via MIDI.

In addition to the standard stereo audio output pair, there are eight separate outputs that can be configured as stereo pairs, or as individual mono outputs. You can also use the separate outputs as insert points for outboard gear.

The K2vx offers eight SIMM sockets (single, in-line memory modules) so you can add optional sample RAM, where you can store samples that you've loaded from disk. You can add up to 128 megabytes of sample RAM! (Sample RAM is not battery-backed, so RAM samples are erased from memory when you power down.)

For offline storage, there's also a floppy drive and two *SCSI* ports, so you can store files on floppies or on an external hard disk or load them from a CD-ROM drive. The two SCSI ports make it easy to chain multiple SCSI devices together. There's also provision for an internal SCSI hard disk. You'll find all this storage potential extremely useful for saving and loading samples, which can be transferred to and from the K2vx using the standard MIDI sample transfer format, or the new, faster, parallel SMDI sample transfer format (SCSI Musical Data Interchange). See the *Reference Guide* for information about MIDI and SMDI sample transfers.

The K2vx's battery-backed RAM will store about 400 of your own programs, or 30,000 notes recorded in the sequencer. This sequencer (Song mode) lets you play back MIDI type 0 sequences, record and play back your own songs, and record multi-timbral sequences received via MIDI. For more onboard storage you can add the P/RAM option, which will increase your battery-backed RAM to about 1280K, enough to store hundreds of additional programs, setups, songs, and other *objects*.

The Local Keyboard Channel feature enables you to use the K2vx's tri-zone setups even if your MIDI controller can transmit on only one channel. The K2vx will also rechannelize incoming

---

MIDI information and send it to its MIDI Out port, enabling you to control additional synths on three different channels.

An optional sampling feature is available, allowing you to make your own mono or stereo samples using analog or digital inputs.

And, of course, there's the incomparable Kurzweil sound. The K2vx comes to you with 200 preset factory programs (called patches, presets, voices, etc. on other synths), as well as 100 multi-zone performance setups. Play them straight from the box, tweak them in any number of ways, or develop your own programs from scratch—which brings us back to the powerful programming capabilities of the K2vx.

## What is VAST?

*Variable Architecture Synthesis* gives the K2vx its unprecedented flexibility. While many other synthesizers offer a fixed set of *DSP* tools (typically filtering, pitch, and amplitude modulation) the K2vx's Variable Architecture lets you arrange a combination of any five DSP functions from a long list of choices. The functions you choose define the type of synthesis you use.

Each layer of every program has its own DSP architecture, what we call an *algorithm*. Within each algorithm, you can select from a variety of DSP functions. Each function can be independently controlled by a variety of sources including LFOs, ASRs, envelopes, a set of unique programmable functions (FUNs), as well as any MIDI control message. The many different DSP functions and the wealth of independent control sources give you an extremely flexible, truly vast collection of tools for sound creation and modification.

## How the K2vx Works

The K2vx integrates two MIDI-driven components: a sound engine, and a global effects processor. The sound engine responds to the MIDI events generated by your MIDI controller and turns them into sounds that are processed within the variable architecture of the algorithms. The resulting sound can then be routed through the effects processor and to the MIX or separate audio outputs.

## How to use this manual

This manual includes the following:

- how to connect and power up your K2vx, getting around the front panel, and a brief description of the operating modes.
- basic editing, including the normal operations of each operating mode.
- the advanced programming features that make the K2vx so powerful—a number of familiar synthesis tools and quite a few new ones.

Even if you're a complete techie, you should read Chapter 3, User Interface Basics. Here you'll get a tour of the K2vx's front panel and learn how to navigate through the major functions.

Chapter 4 describes the concept and operation of the K2vx's eight operating modes, with a brief description of each. Chapters 6 through 13 describe each mode in detail, including the editors contained within them. Chapters 14 through 17 discuss the advanced editing features. Chapter 18 shows you how to use the multiple audio outputs to suit your needs. Chapter 19 is a programming tutorial, giving you specific examples of many of the K2vx's programming functions. Some are basic, some are advanced. By working through the tutorials, you'll become familiar with many synthesis techniques, and you'll get a first-hand look at how to unleash the power of the K2vx.

When manual text appears in boldface italic (*like this*), you'll find it described in the Glossary in the *Reference Guide*. Only the first one or two occurrences of these words are highlighted.

The best way to read this manual is with your K2vx in front of you. By trying the examples we give to illustrate various functions, you can get a quick understanding of the basics, then move on to the more advanced features.

If you like to figure out your equipment for yourself, and normally use manuals only as reference sources, you'll probably get the most use out of the *Reference Guide* that accompanies this manual, which contains brief descriptions of the K2vx's major operating features, and all sorts of useful lists—*programs*, *keymaps*, *algorithms*, effects, *control sources*, and more. A glossary and complete specifications for the K2vx are also included in the *Reference Guide*.

## Introduction

---

How to use this manual



## Chapter 2 Startup

If hooking up new gear is familiar to you, and you just want to get going, here's a quick description of all the basic things you need to cover to get started with your K2vx. If you need more information, thorough descriptions of each step follow. In either case, check out "Playing the Presets," later in this chapter.

### Basic Startup Checklist:

- Install the supplied batteries. See "Battery Installation" on page 2-2.
- Mount your K2vxR securely in a standard 19-inch MIDI rack, or set it on a hard flat surface. Make sure to leave plenty of room for ventilation.
- If you will be transporting the rack containing the K2vxR, be sure to support the back of the unit. It's fairly heavy, and you don't want your K2vxR to get damaged when you move the rack around.
- Connect the power cable.
- Make sure your sound system is at a safe volume level.
- Plug in a pair of stereo headphones or run standard (1/4-inch) audio cables from your amplifier or mixer to the MIX audio outputs on the K2vx (Use the MIX L out for mono). It's good practice to make the cable connection to the K2vx (or any instrument) last, since this will reduce the chance of creating static electricity that can cause an audible "pop" (and, in extreme cases, cause equipment damage).
- Connect a MIDI cable from the MIDI Out port of your MIDI controller to the MIDI In port of the K2vx.
- Power up and start jamming.
- If you hear distortion, reduce the gain on your mixing board, or use the pad if it has one.
- Scroll through the Program list with the Alpha Wheel (the large knob to the right of the display).
- If your MIDI controller can transmit on more than one MIDI channel, check out Setup mode by pressing the Setup mode button. Then use the Alpha Wheel to select tri-zone setups with independent programs, MIDI channels and controller assignments in each zone.
- Press the Quick Access mode button and use the numeric keys to select from programmable banks of ten programs or setups.
- If you don't hear anything, review these steps, or check the Troubleshooting section in the *Reference Guide*.

### Startup—the Details

This section will walk you through the hookup of your K2vx. We'll take a look at the rear panel, then describe the power, audio, and other cable connections.

#### Before You Start...

Don't connect anything until you make sure your K2vx is properly and safely situated. Also, if your K2vx has been out in the cold, give the unit a little time to warm up to room temperature before starting it, since condensation may have formed inside the K2vx.

You'll probably be mounting the K2vx in a standard 19-inch MIDI rack (it takes up three rack spaces). If you will be transporting the rack containing the K2vxR, be sure to support the back of the K2vx within the rack.

If you're not installing the K2vx in a rack, it should rest on a hard flat surface. In this case, it *must* rest on its rubber feet (supplied in the packing carton), and **NOT** on the bottom panel.

**NEVER** block the ventilation openings on the bottom or rear panels; doing so can cause overheating and serious damage. To provide adequate ventilation, the rear panel should be at least four inches from any vertical surface. **There are no user-serviceable parts in the K2vx.** Under no circumstances should you attempt to remove any panels (except for battery installation or replacement). If you attempt to open your K2vx, you'll risk electric shock, and you'll void your product warranty.

## The Rear Panel

### Battery Installation

We've included three AA batteries for your K2vx's battery-backed RAM. We didn't install the batteries at the factory because they would drain during shipping. You should install the batteries before you start up your K2vx.

Locate the battery compartment in the lower right-hand corner of the rear panel (refer the illustration on the following page). Remove the battery compartment cover by inserting a flat-head screwdriver or coin into its slot, then turning it counterclockwise until it pops out. Slide the three batteries, all positive side out, into the compartment. Install the battery compartment cover by lining up the tabs, pushing the cap inward, then turning it 90°. When you start up your unit for the first time after installing the batteries, it will take a few minutes to initialize all of its memory. This will not happen on every power up.

### Connecting the Power Cable (Line Cord)

The K2vx runs on 100-, 120-, 220-, or 240-Volt AC power at 50—60 Hz. Your dealer will set the voltage switch to match the voltage in your area. The voltage level is set with a selector on the rear panel of the keyboard models of the K2vx. On the K2vxR, however, the voltage setting can only be changed by an authorized Kurzweil service center.

As you face the rear panel of the K2vx, the power connector is toward the righthand side. When you've connected the cable at the K2vx end, plug it into a grounded outlet. If your power source does not have the standard three-hole outlet, you should take the time to install a proper grounding system. This will assure you of avoiding problems with audio hum, and will reduce the risk of a shock hazard.

### Connecting the Audio Cables.

*Did you turn down the level on your PA yet?!*

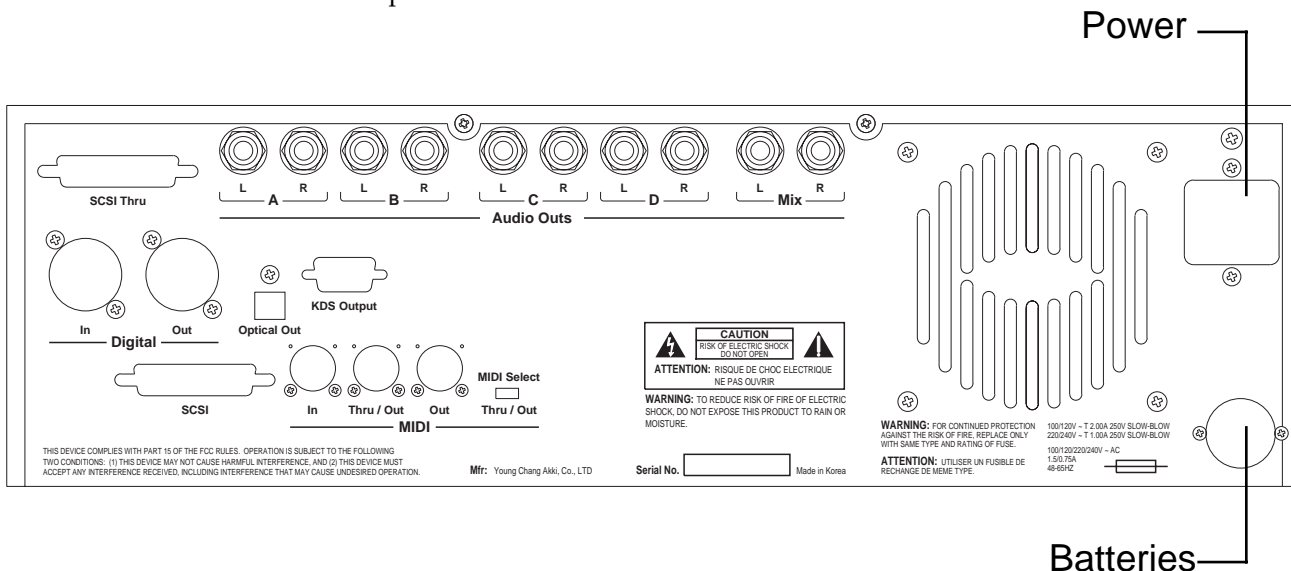
After you've turned down the level on your sound system, you can rig the K2vx's audio connections using a pair of mono audio cables. You'll find ten 1/4-inch jacks near the top of the rear panel. For now, connect one end of each audio cable to your mixing board or PA system inputs, and connect the other end to the jacks marked MIX L and R on the rear panel of the K2vx. If you have only one input available, use the K2vx's MIX L output to get the full signal in mono.

It's always a good idea to make the cable connection to the K2vx (or any instrument) after you've made your other audio connections, since this will reduce the chance of creating static electricity that can cause an audible "pop" (and, in extreme cases, cause equipment damage). Turn to Chapter 18 for more detailed information about the K2vx's audio configuration.

### Connecting MIDI

The simplest MIDI configuration uses a single MIDI cable, from the MIDI Out port of your MIDI controller to the MIDI In port of the K2vx. There are all sorts of possible configurations, including additional synths, personal computers, MIDI effects processors, and MIDI patch bays. Depending on your system, you may want to use the K2vx's MIDI Thru port to pass MIDI information from your MIDI controller to the K2vx and on to the next device in your system. You can also connect MIDI devices to the K2vx's MIDI Out port, which can send channelized MIDI information from your MIDI controller. See the discussion of the Local Keyboard Channel parameter in Chapter 10.

NOTE: You can perform a quick check of the K2vxR and your audio system without a MIDI controller connected to the K2vxR. Hold down the CANCEL button on the alphanumeric pad and press any other alphanumeric button, and you should hear notes. See "The Panel Play Feature" in Chapter 3 for more information about this feature.



## Connecting SCSI

You may not have a hard disk or other SCSI device to connect to your K2vx right away, but if you do, you can connect it to either of the SCSI ports. Please read the following information carefully; it's very important.

**IMPORTANT NOTE:** If you plan to connect more than one SCSI device to the K2vx (including an Apple Macintosh®), you *must* terminate your SCSI chain properly. Turn to Chapter 13 and read the section called "SCSI Termination." You can lose data, and possibly damage your K2vx and SCSI devices if they're not terminated properly.

## Switching On the Power

The power switch is located at the lower left of the front panel. When you power up, the display will briefly say "Welcome to the K2vx!" The Program Mode display will then appear. It looks like the diagram below (the programs shown in the diagram don't necessarily exist):

```

ProgramMode  XPose:091  <>Channel:1
198 Analog Jam
199 Default Program
  1 Righteous Piano
  2 Mondo Bass
  3 Killer Drums
  4 Weeping Guitar
Octav-  Octav+  Panic  View  Chan-  Chan+

```

Set the volume at a comfortable level. You'll get the best signal-to-noise ratio if you keep the K2vx at full volume (turn the volume knob fully clockwise), and adjust the level from your mixing board. You may also want to adjust the display contrast. This is done with the Contrast parameter in Master mode. See Chapter 11. As you trigger notes from your MIDI controller, you'll see the MIDI LED flash. If it doesn't flash, check your MIDI cables and connections.

## Playing the Presets

There are three things you'll want to check out right away: programs, setups and Quick Access banks. In performance situations, you'll be selecting your sounds using one of these three methods.

### Programs

The K2vx powers up in Program mode, where you can select and play programs stored in *ROM* or *RAM*. Programs are preset sounds composed of one to three layers of samples or waveforms—they're called patches, voices, and presets on other instruments. If you've left Program mode, just press the Program mode button to return.

Take a minute to familiarize yourself with the Program mode display. It gives you some helpful basic information, like the MIDI transposition, what MIDI channel you're on, and which program is currently selected. The box at the left of the display tells you which keymap is used by each layer of the current program (a keymap is a collection of samples). The line under each keymap name gives a rough indication of the keyboard range of the layer. In the figure below, the Grand Piano keymap covers the full range from C 0 to C 8. You can change MIDI channels with the Chan- and Chan+ buttons under the display. You can transpose by octaves by pressing the Octav buttons under the display (if the LocalKbdCh parameter on the RECV page in MIDI mode matches the transmitting channel of your MIDI controller—see Chapter 10). The bottom line of the display identifies the function of each of the buttons beneath the display.

```

ProgramMode   XPose:051   <>Channel:1
KeyMap Info   199 Default Program
Grand Piano   200*Crystal Horn
              1 Righteous Piano
              2 Mondo Bass
              3 Killer Drums
              4 Weeping Guitar
Octav- Octav+ Panic View Chan- Chan+

```

When you want to change programs, you have several options. The K2vx has six different settings for responding to MIDI program change commands. These are explained in Chapter 10, so we won't go into them here. You should be able to change programs by sending program change commands from your MIDI controller. Of course, you can always change programs from the K2vx's front panel using the Alpha Wheel. Turning it left or right will scroll through the program list. Press the View soft button to see the current program in large type. Press it again to return to the normal view. You can also change programs using the cursor buttons, or the Plus/Minus buttons under the Alpha Wheel. You can also use the CHAN/BANK buttons or the cursor buttons while in Program mode.

Be sure to try whatever performance controls your MIDI controller has: the Pitch Wheel, Mod Wheel, and switch or control pedals—different programs respond to them in different ways. Attack velocity and aftertouch also have varying effects. Check the tear-out sheets at the back of this manual for lists of factory programs and setups. There's also a list of programs in the *Reference Guide*, describing how each program responds to specific MIDI control messages.

If you don't hear anything, see the troubleshooting section in the *Reference Guide*.

### Setups

Setups are preset combinations of programs. Each setup has three zones, each of which can be assigned to any range of the keyboard (overlapping or split). Each zone can have its own program, MIDI channel, and MIDI control assignments. You can make use of setups even if your MIDI controller can transmit on only one MIDI channel at a time. See the *parameter* called Local Keyboard Channel in Chapter 10.

Press the Setup mode button to the left of the display. Its LED will light, telling you that you're in Setup mode. Notice that the Setup mode display is similar to the Program mode display. The box at the left shows you the programs assigned to each of the setup's three zones, and which MIDI channel is used for each program. Press the View button to *toggle* between large type and normal view.

```

SetupMode      Xpose:0ST
                201*Friday Gig
                202*Pop Rock Reggae
Chan/Program Info
1      9 Cool Traps
2     18 Sly Acoust
3     22*Izitt Jimmy
                1 Jazz Trio
                2 All Percussion
                3 Heavy Metal
                4 To Sequencer
Octav- Octav+ Panic View

```

## Quick Access

A really convenient way to select programs and setups is to use Quick Access mode, where you select a Quick Access *bank* from a list of factory preset or user-programmed banks. Each bank contains ten memory slots, or entries, where you can store any combination of programs or setups. While you're in Quick Access mode, any program or setup can be selected with the numeric buttons 0 through 9.

The K2vx comes with a few Quick Access banks already programmed so you can get an idea of how they work. You'll probably create your own Quick Access banks to help you select programs and setups with a minimum of searching. Press the Quick Access mode button to the left of the display. Its LED lights, to tell you you're in Quick Access mode. You'll see a display that looks like this:

```

QuickAccessMode <>Bank:1 For Show 1
Silk Rhodes    POLY TOUCH    Voice+String
NastyTrombone SINK MONSTA    Waterflute
Soon           Jazz Trio 2    Fretless Bas
                VELVETEEN
Xpose:0ST     SINK MONSTA    Chan:1
Octav- Octav+ Panic View    Chan- Chant+

```

The top line of the display tells you which Quick Access bank is selected. Use the CHAN/BANK buttons (to the left of the display) to scroll through the banks. The names of each of the ten entries in the bank are listed in the center of the display. Many of their names will be abbreviated. The currently selected entry's full name is shown near the bottom of the display. The amount of transposition is displayed to the left of the entry name. If the current entry is a program, you'll see the current keyboard (MIDI) channel displayed to the right of the entry's name. If it's a setup, you'll see the word "Setup."

The entries on the Quick Access page are arranged to correspond to the layout of the numeric buttons on the alphanumeric pad. On the page above, for example, the program "SINK MONSTA" is entry 5, and can be selected by pressing 5 on the alphanumeric pad.

The Panic button (the button below the word "Panic" in the bottom line of the display) will send an All Notes Off and an All Controllers Off message to the K2vx, and over all 16 MIDI channels. Press the View button to show the currently selected bank entry in large type. Press it again to return to the normal view.

When you're ready to create your own Quick Access banks, turn to Chapter 8 to learn about the Quick Access Editor.

## Performance Controls

### From your MIDI controller

The K2vx responds to attack velocity and release velocity, as well as mono and poly pressure (aftertouch) messages. It will also respond to most standard MIDI control parameters; see "MIDI and Setup Control Parameters" below.

### Volume knob

The Volume knob controls the volume at the MIX (stereo) audio outputs and at the headphone jack. The individual outputs and MIDI volume are not affected by this knob.

### MIDI and Setup Control Parameters

The K2vx responds to most standard MIDI control messages, including Pitch Wheel, Mod Wheel, switch pedals, control pedals, and control sliders. The K2vx's response to these controls is set on the XMIT (Transmit) page in MIDI mode, and works in conjunction with the Local Keyboard Channel parameter on the RECV (Receive) page in MIDI mode. This is described fully in Chapter 10. For now, you'll probably find that the K2vx responds predictably to most of these controls. If you want to change its response, turn to Chapter 10 and read the section covering the Local Keyboard Channel parameter.

You can also use the K2vx to relay control messages from your MIDI controller to other MIDI devices connected to the K2vx's MIDI Out port. This is also controlled by the Local Keyboard Channel parameter (MIDI RECV page) in combination with the settings for the control parameters on the MIDI XMIT (transmit) page or the Setup mode page.

## K2vx Boot Loader

The Boot Loader lets you install a new K2vx operating system and/or new K2vx ROM objects, from a SCSI device or the floppy drive into flash ROM. The Boot Loader also provides hard reset and diagnostics options.



*Note:* Your K2vx comes from the factory with the operating system and ROM objects already installed. You do not need to run the K2vx Boot Loader to start up a new K2vx.

### Starting the Boot Loader

When you start the K2vx, it displays a "Please wait..." message and waits for approximately 2 seconds. Press and release the **Exit** key while the "Please wait..." message is displayed to start the boot loader. Otherwise, the K2vx will start up normally.

The Boot Loader's Menu resemble K2vx dialog boxes. That is, they consist of a series of labels and a highlight bar that is used to select one of the labels. You can use the arrow keys to move the highlight bar up, down, right, and left. Press the **OK** soft button to invoke the highlighted menu option. Additionally, the data wheel and **Enter** key may be used to move the highlight bar and invoke the selected action.

### Boot-loader Main Menu:

When the boot loader is first invoked, its main menu is displayed, as shown below.

```

----- K2vx Boot Loader v1.01 -----
      Install System      Hard Reset
      Install Objects    Run Dia9s
      Run System         Fixed Dia9s
                               OK

```

Initially, the "Install System" option is highlighted. After invoking various actions, however, other options may be selected by default, as the context suggests, upon returning to this menu.

## Install System / Install Objects

Use "Install System" to install the K2vx operating system and diagnostics by loading them from a floppy or from a SCSI device and programming them into Flash ROM. Use "Install Objects" to install ROM objects into Flash ROM. After either option is invoked, a dialog box such as the following will be displayed.

```

Device to install from:  floppy
SCSI ID of this unit:   N/A
                        Cancel  OK

```

From this dialog box, you can select which device contains the system or ROM-objects file. This is done using the alpha wheel, the + and - keys, and the cursor control keys in the same way that the K2vx engine software does. If you choose to install from a SCSI device, the SCSI ID of this unit will change from "N/A" to "SCSI 6." If there is another device on your SCSI bus with an ID of 6, you will need to change the SCSI ID of this unit to a free SCSI ID. At this point you can continue by hitting the **OK** soft-button or abort the operation by pressing the one labeled **Cancel**. Canceling the operation will return you to the Boot Loader menu, and the "Run System" option will be highlighted.

When you press **OK**, the K2vx will try to access the drive in question and scan it for files. If it is successful, a screen such as the following will then appear:

```

                                FILE1.KOS
                                DIR1  <DIR>
                                FILE1.K25
File to Install:  FILE1.KR2
                                VLABEL <VOL>
NewDrv  Root  Parent  Cancel  OK

```

While this screen is displayed, you can use the up and down cursor keys, the alpha wheel, or the + and - keys to highlight different items in the file list. Names that have "<DIR>" displayed to their right are sub-directories. If the disk has a volume label, it will be displayed in the list, marked with "<VOL>." Operating system files typically have a ".KOS" extension, while ROM object files have a ".KRZ" extension. The **OK** soft-button selects the highlighted file for loading, or changes to the highlighted directory. The **Cancel** button aborts the installation and returns to the menu. **Parent** changes to the current directory's parent directory, while **Root** changes to the disk's root directory. The **NewDrv** button returns to the previous screen, allowing you to choose a different drive from which to install.

After you select a file by pressing the **OK** button, you will be asked to confirm the installation. The K2vx will display a message of the form "Ready to install operating system from *filename* on *device*" or "Ready to install ROM objects from *filename* on *device*." When it displays this message, it will fill in *filename* and *device* with the name of the file and device you have selected. To confirm that you want to proceed, press the **OK** soft-button. If you do so, you will not be able to interrupt the installation. Alternatively, the **Cancel** button will return you to the menu, or the **Choose** button will allow you to select a different device or file from which to install.

While the K2vx is installing the file into Flash ROM, the top line of the display will gradually fill to indicate the progress of the operation. Note that operating system files and ROM object files are of different formats. If you try to install a file using the incorrect menu option, the K2vx will think the file is corrupt and will refuse to install it into Flash ROM. If this happens, simply choose the correct menu option from the Boot Loader's main menu, and repeat the installation.

If there is a problem accessing the drive or installing the file, an error message will be displayed. At this point, you can (1) abort the operation with the **Abort** soft button and return to the menu, (2) press the **Retry** button to try the operation again, or (3) choose another device or file by pressing the **Choose** button.

## Run System

The "Run System" option executes the engine software. When this option is invoked, the operating system and ROM setups are checked for integrity. If either does not appear to be installed, a message will be displayed explaining this. If this happens, you can (1) double press the **Run Anyhow** soft buttons and be given a chance to try to run the system anyhow, knowing that the machine might crash, (2) install the uninstalled portion with the **Instal** button, or (3) return to the menu by using the **Menu** button. If you do the latter, the menu will be displayed with the "Load System" or "Load Setups" option highlighted as appropriate.

## Hard Reset

The "Hard Reset" option causes all user RAM to be erased. When you select this option, a strict warning message is displayed. You must press the up and down cursor keys simultaneously to actually erase the RAM. If you push any other button or turn the data wheel, the operation is aborted, and the menu is displayed with the "Run System" option highlighted. Otherwise, the RAM will be marked for re-initialization, and a message declaring this fact will appear prior to return to the menu. When the menu reappears, it will have the "Run System" option highlighted.

## Run Diags

The "Run Diags" option executes the diagnostics that are loaded with the system. When you choose "Run Diags", the K2vx checks its operating system for integrity. If it does not appear to be installed, a message will be displayed explaining this. If this happens, you can (1) double press the **Run Anyhow** soft buttons and be given a chance to try to run the diagnostics anyhow, knowing that the machine might crash, (2) install the operating system with the **Instal** button, or (3) return to the menu by using the **Menu** button. If you do the latter, the menu will be displayed with the "Load System" or "Load Setups" option highlighted as appropriate. Some diagnostic tests erase the non-volatile user RAM, and you will be warned of this fact before continuing.

Restart your K2vx to leave the diagnostics and return to regular operation.

## Fixed Diags

The "Fixed Diags" option executes the diagnostics that are contained in the boot block. Some diagnostic tests erase the non-volatile user RAM, and you will be warned of this fact before continuing.

Restart your K2vx to leave the diagnostics and return to regular operation.



## Chapter 3

# User Interface Basics

Chapter 3 will show you how to get around the front panel of your K2vx. Your interactions can be divided into three primary operations: mode selection, navigation, and data entry.

## Mode Selection

The K2vx is always in one of eight operating modes. The modes are selected by pressing one of the eight buttons beneath the display—the ones with LEDs to their right. Selecting a mode gives you access to a large set of related parameters. Only one mode can be selected at a time. The modes are:

- PROGRAM MODE: Select and play programs, and modify them with the *Program Editor*. Rearrange and modify samples in the Keymap and Sample Editors.
- SETUP MODE: Select and play setups (three keyboard zones with independent MIDI channel, program and control assignments), and modify them with the Setup Editor.
- QUICK ACCESS MODE: Select from a list of preset banks, each containing a list of ten programs and/or setups that can be viewed in the display for easy selection. Modify the preset banks and create your own with the Quick Access Editor.
- EFFECTS MODE: Define the behavior of the on-board global effects processor. Modify the preset effects and create your own with the Effects Editor.
- MIDI MODE: Define how your K2vx sends and receives MIDI information, and configure each channel to receive independent program, volume, and pan messages that override the normal Program mode settings.
- MASTER MODE: Define performance and control characteristics for the entire K2vx.
- SONG MODE: Use the K2vx's sequencer to record and play back your keyboard performance, play type 0 MIDI sequences, and record multi-timbral sequences received via MIDI.
- DISK MODE: Interface with the K2vx's floppy disk drive, an external SCSI device, or an optional internal SCSI disk to load and save programs, setups, samples, and more.

## Mode Buttons

The mode buttons are labeled in white. When you press a mode button, the LED to its right lights up to indicate that the mode has been selected. If pressing a mode button does not light its LED, press the EXIT button one or more times, then try again.

The green labeling under each mode button indicates special functions that relate to some of the K2vx's editors. These functions are described in Chapter 5.

## Navigation

The navigation section of the front panel consists of the display and the buttons surrounding it. These navigation buttons will take you to every one of the K2vx's programming parameters.

## The Display

Your primary interface with the K2vx is its backlit graphic display. As you press various buttons, this fluorescent display reflects the commands you enter and the editing changes you make. The ample size of the display (240-by-64 *pixels*) enables you to view lots of information at one time.

## Pages

Within each mode, the functions and parameters are organized into smaller, related groups that appear together in the display. Each one of these single-screen groups of parameters is called a *page*. Each mode has what we call an entry level page; it's the page that appears when you select that mode with one of the mode buttons. Within each mode and its editor(s), the various pages are selected with the navigation buttons. There are many pages, but there are a few features common to each page. The diagram below shows the entry level page for Program mode.

```
ProgramMode  Xpose:051  <>Channel:1
200*Analog Jam
KeyMap Info  202*Heart Strings
Grand Piano  1 Righteous Piano
2 Mondo Bass
3 Killer Drums
4 Weeping Guitar
Octav- Octav+ Panic View Chan- Chan+
```

## The Top Line

On the top line of most pages, there's a reminder of which mode you're in and which page you're on. Many pages display additional information in the top line, as well. The Program mode page above, for example, shows you the current amount of MIDI transposition and the currently selected MIDI channel. The top line is almost always "reversed"—that is, it has a white background with blue characters.

## The Bottom Line

The bottom line is divided into six (sometimes fewer) sets of reversed characters that serve as labels for the six buttons directly beneath the display. These labels—and the functions of the buttons—change depending on the currently selected page. Consequently the buttons that select these functions are called "soft" buttons.

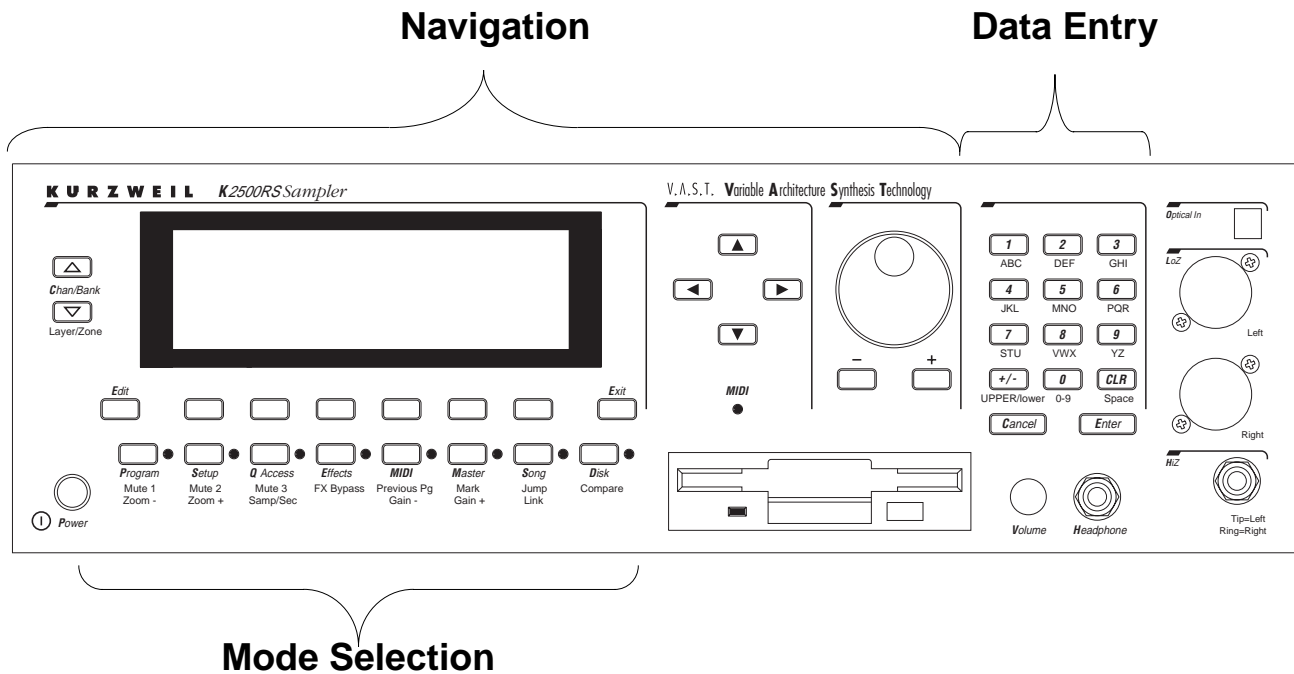
## The Soft Buttons

The soft buttons are called "soft" because their functions change depending on the currently selected mode. Sometimes they perform specific functions, like changing MIDI channels in Program mode. In the Program Editor, they're also used to move to different pages of programming parameters.

## The Cursor Buttons

To the right of the display are four buttons arranged in a diamond fashion. These are called the cursor buttons. They move the cursor around the currently selected page, in the direction indicated by their labels. The cursor is is a highlighted (reversed) rectangle (sometimes it's an underscore). It marks the value of the currently selected parameter.


Programming the K2vx involves selecting various parameters and changing their values. Parameters are selected by highlighting their values with the cursor. The highlighted value can be changed with any of the data entry methods described in the data entry section below.



## The MIDI LED

Below the cursor buttons is a red LED labeled MIDI. This LED will flash whenever the K2vx receives MIDI information from your MIDI controller.

## The CHAN/BANK Buttons

To the left of the display are two buttons labeled “CHAN/BANK.” Their function is related to the two small arrows——that appear in the top line of many different pages. When you see these arrows, you can use the CHAN/BANK buttons to scroll the values of the parameter that appears to the right of the arrows. The arrows don’t have to appear in the top line, however, for the CHAN/BANK buttons to have an effect. In Program mode, for example, they shift through the MIDI channels, showing the program assigned to each channel.

When you’re in the Program editor, the CHAN/BANK buttons let you view each layer in the program. You can see the corresponding parameters in each layer by scrolling through the layers with these buttons. In the Setup editor, the CHAN/BANK buttons scroll through the zones in the current setup. In Quick Access mode, they scroll through the Quick Access banks, and in Song mode they scroll through record tracks.

We’ll let you know, when applicable, what the CHAN/BANK buttons do.

## The EDIT Button

The EDIT button activates each of the K2vx’s editors, and acts as a shortcut to many pages within the Program Editor. Pressing the EDIT button tells the K2vx that you want to change some aspect of the object marked by the cursor. For example, when a program is selected and you press EDIT, you enter the Program Editor. If a setup is selected, you enter the Setup Editor.

There are editors accessible from every mode except Disk mode. To enter an editor, choose one of the modes (mode selection), and press EDIT. An editing page for that mode will appear. You can then select parameters (navigation) and change their values (data entry). If the value of the selected parameter has its own editing page, pressing the EDIT button will take you to that

page. For example, in the Program Editor, on the PITCH page, you might see LFO1 assigned as the value for Pitch Control Source 1. If you select this parameter (the cursor will highlight its value—LFO1 in this case), then press the EDIT button, you'll jump to the page where you can edit the parameters of LFO1. Naturally, you can find every page in the current editor by using the soft buttons, but often it's easier to use the EDIT button shortcut.

### The EXIT Button

Press EXIT to leave the current editor. If you've changed the value of any parameter while in that editor, the K2vx will ask you whether you want to save your changes before you can leave the editor. See Chapter 5 for information on saving and naming. The EXIT button also takes you to Program mode if you're on the entry level page of one of the other modes. If at some point you can't seem to get where you want to go, press EXIT one or more times to return to Program mode, then try again.

## Data Entry

The data entry section of the front panel includes the Alpha wheel, the Plus/Minus buttons, and the 14-button alphanumeric pad.

### The Alpha Wheel

The Alpha Wheel is especially useful because it can quickly enter large or small changes in value. If you turn the Alpha Wheel one click to the right, you'll increase the value of the currently selected parameter by one increment. One click to the left decreases the value by one increment. If you turn it rapidly, you'll jump by several increments.

### The Plus/Minus Buttons




These buttons are located just under the Alpha Wheel. The Plus button increases the value of the currently selected parameter by one, and the Minus button decreases it by one. These buttons are most useful when you're scrolling through a short list of values, or when you want to be sure you're changing the value by one increment at a time. One press of the Plus or Minus button corresponds to one click to the right or left with the Alpha Wheel. These buttons will repeat if pressed and held.

Pressing the Plus and Minus buttons simultaneously will move you through the current list of values in large chunks instead of one by one. Often this is in even increments (10, 100, etc.). Don't confuse these buttons with the +/- button on the alphanumeric pad. This button is used primarily for entering negative numeric values and switching from uppercase to lowercase letters (and vice versa).

### The Alphanumeric Pad

As its name implies, this set of 14 buttons lets you enter numeric values, and to enter names one character at a time. Depending on where you are, the K2vx automatically enters letters or numerals as appropriate (you don't have to select between alphabetic or numeric entry).

When you're entering numeric values, press the corresponding numeric buttons, ignoring decimal places if any (to enter 1.16, for example, press 1, 1, 6, ENTER). The display will reflect your entries, but the value won't actually change until you press ENTER. Before pressing ENTER, you can return to the original value by pressing CANCEL. Pressing CLEAR is the same as pressing 0 without pressing ENTER.

When entering names, you'll use the left/right cursor buttons or the  /  soft buttons to move the cursor to the character you want to change. Use the labels under the alphanumeric buttons as a guide to character entry. Press the corresponding button one or more times to insert the desired character above the cursor. The CANCEL button is equivalent to the  soft button, and ENTER is the same as OK. The CLEAR button replaces the currently selected character with a space. The "+/-" button *toggles* between uppercase and lowercase letters.

## Double Button Presses

Pressing two or more related buttons simultaneously executes a number of special functions depending on the currently selected mode. Make sure to press them at exactly the same time.

<b>IN THIS MODE:</b>	<b>THESE BUTTONS:</b> (Pressed simultaneously)	<b>WILL DO THIS:</b>
PROGRAM MODE	<b>Octav-, Octav+</b>	Reset MIDI transposition to 0 semitones. Double-press again to go to previous transposition.
	<b>Chan-, Chan+</b>	Set current MIDI channel to 1.
	<b>Plus/Minus</b>	Step to Program 100, 200. etc.
MASTER MODE	<b>CHAN/BANK</b>	Enables Guitar/Wind Controller Mode.
SONG MODE	<b>left/right cursor buttons</b>	Toggle between Play and Stop.
	<b>up/down cursor buttons</b>	Toggle between Play and Pause.
	<b>Plus/Minus</b>	Select Quantize Grid values on MISC page and Edit Song:TRACK Quantize page. Select duration for a step on Edit Song:STEP page. Increment GateTime by 20% intervals on Edit Song:STEP page.
	<b>CHAN/BANK</b>	Select all tracks on Edit Song:TRACK page.
DISK MODE	<b>2 leftmost soft buttons</b>	Issue SCSI Eject command to currently selected SCSI device.
	<b>CHAN/BANK</b>	Hard format SCSI device. List selected objects when saving objects.
	<b>left/right cursor buttons</b>	Select all items in a list. Move cursor to end of name in naming dialog.
	<b>up/down cursor buttons</b>	Clear all selections in a list. Move cursor to beginning of name in naming dialog.
PROGRAM EDITOR	<b>CHAN/BANK</b>	Select Layer 1.
SAMPLE EDITOR	<b>2 leftmost soft buttons</b>	Toggle between default zoom setting and current zoom setting.
	<b>Plus/Minus</b>	Set the value of the currently selected parameter at the next <i>zero crossing</i> .
ANY EDITOR	<b>Plus/Minus</b>	Scroll through the currently selected parameter's list of values in regular or logical increments (varies with each parameter).
	<b>2 leftmost soft buttons</b>	Reset MIDI transposition to 0 semitones. Double-press again to go to previous transposition.
	<b>Center soft buttons</b>	Select Utilities menu.
	<b>2 rightmost soft buttons</b>	Sends all notes/controllers off message on all 16 channels (same as Panic soft button).
SAVE DIALOG	<b>Plus/Minus</b>	Toggle between next free ID and original ID.

## Intuitive Data Entry

Many parameters have values that correspond to the standard physical controls present on most MIDI controllers (primarily keyboards). In many cases, you can select these values “intuitively,” rather than having to scroll through the *Control Source* list. This is done by selecting the desired parameter, then holding the ENTER button while moving the desired physical control.

For example, on the LAYER page in the Program Editor, you can set the range of the currently selected layer as follows: use the cursor buttons to move the cursor to the value for the “LoKey” parameter, press (and hold) the ENTER button, then use your MIDI controller to trigger the note you wish to be the lowest note for the currently displayed layer. The note you triggered (it has to be between C 0—C 8) will appear as the value for the LoKey parameter. Repeat the process for the HiKey parameter.

Another example: select Program 199 while in Program mode. Press EDIT to enter the Program Editor. Press the PITCH soft button to select the PITCH page. Move the cursor to the Src1 parameter. Hold the ENTER button, and move your controller’s Pitch Wheel. PWheel will be selected as the value for Src1.

You can also use the keyboard of your MIDI controller to choose control sources, since each key number corresponds to a value on the control source list. If you have a certain control source that you use over and over (for example, LFO1), this can be the quickest way to enter its value. To do this: highlight a parameter which uses a value from the control source list, hold down ENTER, then strike the key corresponding to the control source you want to choose. LFO1, for example, is assigned to B5.

Also, for almost every parameter, holding the ENTER button links the K2vx to your MIDI controller’s Data slider, if it has one. Moving the Data Slider will run through the range of values for the currently selected parameter. This is not as precise as the Alpha Wheel, but much faster.

## Changing the Current Layer in Multi-Layer Programs

When editing a multi-layer program (including drum programs), you can quickly switch between layers by holding the ENTER button, then striking a key. The K2vx will change the current layer to that key’s layer. If the key is part of more than one layer, subsequent key strikes will cycle through each layer that has that key in its range.

## Search Function

There’s a convenient way to find any alphabetic or numeric string of characters within the currently selected list, or range of values. Hold the ENTER button and press any of the numeric keys. A *dialog* like the Name Dialog will appear. Type in the string of characters you want to find. For example, if you’re looking at the Program list and you want to find all programs containing the word “Horn,” you would type h-o-r-n. This function is not case sensitive; it will find upper and lower case characters regardless of what you type.

When you’ve typed the string of characters you want to find, press ENTER. The K2vx searches through the current list of values and finds all values that match the string of characters you typed. Hold ENTER and press one of the Plus/Minus buttons to search for the next higher- or lower-numbered object that contains the string of characters.

The string you select will remain in memory. You can store and select a string of characters with each of the numeric buttons. Hold ENTER and press one of the numeric buttons at any time to select that string for a search. When the string appears, you can change it, or just press ENTER to find that string.

## The Panel Play Feature (K2vxR)

You can play notes on the K2vxR even if you don't have a MIDI controller connected to its MIDI In port. Using the alphanumeric pad, you can play the octave from C 4 to C 5 and send the notes to the MIX and Headphone outputs.

While in any mode, press and hold the CANCEL button on the alphanumeric pad. Press one of the other alphanumeric buttons to play various notes: 1 is C 4, 2 is C# 4, and so on up to ENTER, which plays C 5. Notes are sent to the K2vx's sound engine on the current MIDI channel.

Pressing the Up or Down cursor buttons while holding CANCEL will transpose up or down an octave while using this feature. This transposition will be remembered the next time you use this feature, but does not affect the K2vx's normal MIDI transposition.

Notes are played at approximately the mf velocity level. You can adjust the attack velocity of the notes using the Plus/Minus buttons while holding down the CANCEL button. The velocity values will continue to change if you hold the buttons down.

You can press the Right cursor button while holding CANCEL to sustain notes (if this doesn't work, set a value of Sustain for the FtSw1 parameter on the MIDI mode XMIT page). The Left cursor button will release notes.





# Chapter 4

## The Operating Modes

In this chapter we'll discuss the theory behind the mode system, and describe the basic operating features of each mode.

### What the Modes Are

The modes exist to make the K2vx logical to work with. With as many performance and programming features as the K2vx has, it's helpful to break them into groups. These groups are called modes. There are eight of them; they're described briefly in the section called "Using the Modes," below. Chapters 6 through 13 are dedicated to explaining each mode in turn.

Each mode is named for the kind of operations you perform while in that mode, and each mode's editor (if any) contains all of the parameters related to editing the type of *object* found in that mode. In Setup mode, for example, you select setups (and only setups) for performance or editing. All of the setup-editing parameters are grouped together on the Setup Editor page, which is accessible through Setup mode.

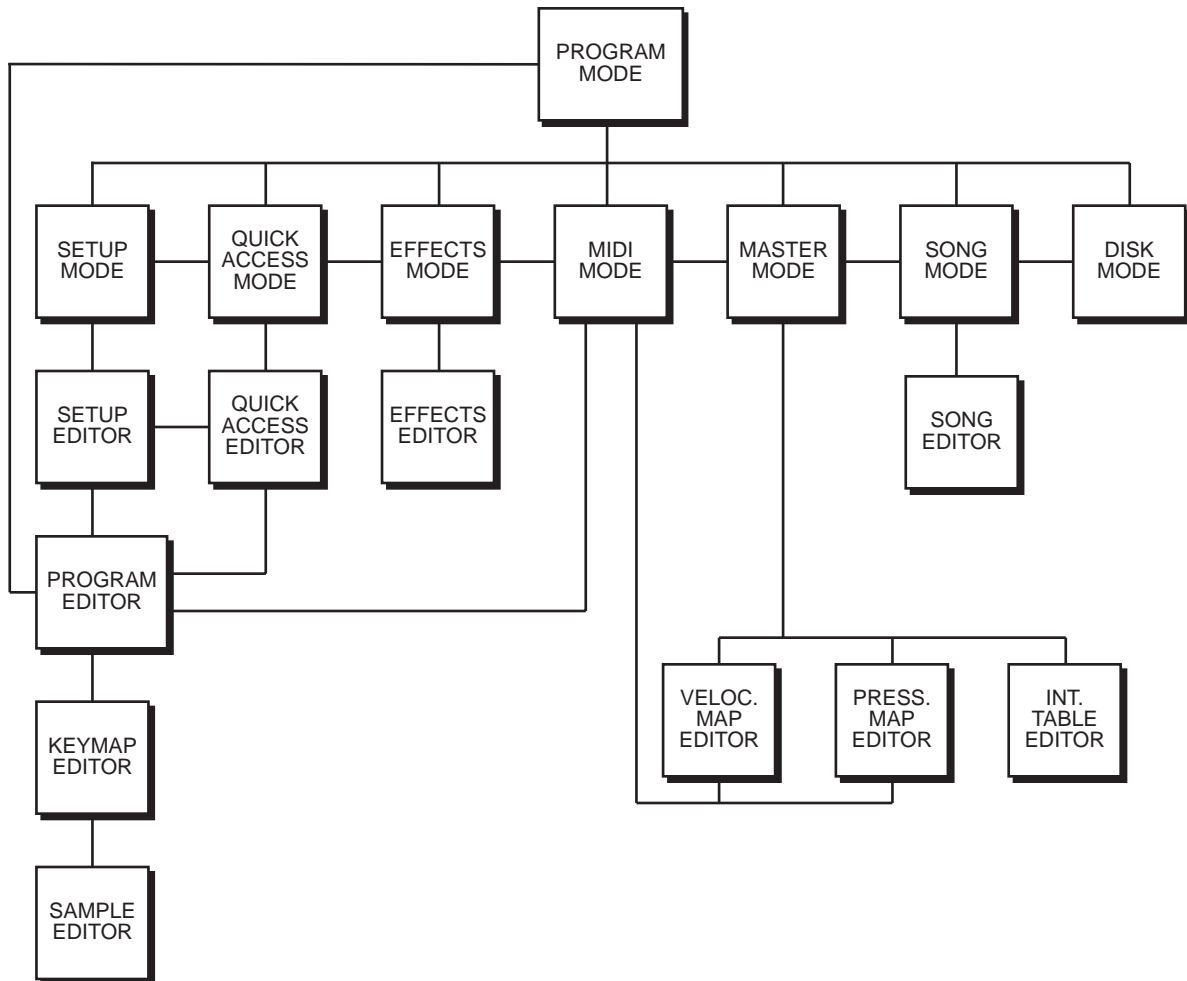
### Selecting Modes

When the K2vx is on, it's always operating in one of the eight modes represented by the LED-highlighted buttons beneath the display. Pressing one of the mode buttons selects that mode. This is the mode's entry level. At the entry level, the LED of the selected mode is lit. Only one mode can be selected at a time.

At the entry level, you can exit any mode simply by pressing one of the other mode buttons. If you enter the mode's editor, however, you must press EXIT to return to the mode's entry level before selecting another mode.

All of the modes except Disk mode give you access to one or more editors for changing the values of the parameters within that mode. Press the EDIT button to enter the editor of the currently selected mode. When you do this, the mode LED goes out.

It's possible to enter another mode's editor without leaving the currently selected mode. For example, if you press EDIT while in Setup mode, you'll enter the Setup Editor. The Setup editor page will appear, and one of the programs in the setup will be highlighted by the cursor. If you press EDIT again, you'll enter the Program Editor, where you can edit the currently selected program. While you can edit and save programs as you normally would, you're still in Setup mode, and you can't select another mode at this point. When you exit the Program Editor, you'll return to the Setup Editor page. Press EXIT again, and you'll leave the Setup Editor, returning to the Setup mode page.



### Nested Editors

Starting at the Program mode level, there are three “nested” editors, each related to the parameters that make up different components of a program. The first is the Program Editor, which you enter when you press EDIT while in Program mode. Programs consist, among other things, of *keymaps*; they determine which samples play on which keys. Keymaps can be edited as well. The Keymap Editor is entered from within the Program Editor, by selecting the KEYMAP page with the soft buttons, then pressing EDIT.

Similarly, keymaps consist of *samples*, which also can be edited. The Sample Editor is entered from the Keymap Editor, by selecting the Sample parameter and pressing EDIT. When you enter the Sample Editor, you’ve worked through three nested levels of editors, all related to the components that make up a program. And in fact, you’re still in Program mode (if that’s where you started from). Pressing EXIT while in the Sample Editor will return you to the Keymap Editor. Pressing EXIT again will return you to the KEYMAP page of the Program Editor. Once more, and you’re back to Program mode’s entry level.

## Finding Square One

If, at any time, you don't know where you are, and the mode LEDs are all unlit, press EXIT one or more times. This will return you to the entry level of whatever mode you were in, and if you press EXIT enough times, you will always return to Program Mode, the startup mode. If you've made any changes, you'll be asked whether you want to save before leaving any editor. Press the No soft button or the EXIT button if you don't want to save. If you want to save, press the Rename or Yes soft button, and you'll see the Save dialog, which is described in Chapter 5, in the section called "Saving and Naming."

## Using the Modes

You can play your K2vx regardless of the mode you're in. In fact, the only times you can't play it are when you're in the middle of a disk operation (loading, saving, formatting) or a SMDI sample transfer. With these two exceptions, the K2vx's MIDI response is almost always active. Even so there are three modes that are more performance-oriented than the others. These are Program, Setup, and Quick Access modes. We'll describe each of the eight modes briefly in this section.

### Program Mode

The K2vx starts up in Program mode, where you can select, play and edit programs. The Program mode entry level page shows the currently selected program, as well as a small segment of the program list.

The Program, Keymap, and Sample Editors are nested within Program mode. They take you to the core of the K2vx's sound editing parameters. We'll discuss them in Chapters 6 and 15.

### Setup Mode

Setup mode is used to select, play, and edit setups, which consist of three separate zones, split or overlapping. Each zone has its own program, MIDI channel and control parameters. Setups are great for performance situations, whether you're playing multiple K2vx programs or controlling additional synths connected to the K2vx's MIDI Out port. Chapter 7 describes Setup mode and the Setup Editor in detail. You can make use of Setup mode even if your MIDI controller can transmit on only one MIDI channel at a time. To do this, go to the RECV page in MIDI mode (by pressing the RECV soft button while in MIDI mode), and set the Local Keyboard Channel parameter to a value that matches the transmit channel of your MIDI controller. When you select Setup mode, the K2vx will interpret incoming MIDI information according to the settings for the currently selected setup. See the discussion of the Local Keyboard Channel parameter in Chapter 10 for details.

### Quick Access Mode

Another feature for live performance, Quick Access mode enables you to combine programs and setups into banks of ten entries. Each of these programs or setups can be selected with a single alphanumeric button. Different banks are selected with the CHAN/BANK buttons. There's a selection of factory preset banks, and you can use the Quick Access Editor to create your own banks and store them in RAM. There's a full description in Chapter 8.

### Effects Mode

Effects mode is used to set the behavior of the K2vx's global effects processor. The Effects mode page lets you tell the K2vx how to select preset effects when you change programs or setups, and lets you choose a preset effect and mix level that's applied to every K2vx program. (There's also an EFFECT page in the Program Editor, where you can select a preset effect and mix level

for each program individually.) The Effects Editor allows you to tweak the preset effects, and create your own. Chapter 9 shows you how. You can also listen to the sounds of various effects while in Effects mode, without selecting different programs.

### **MIDI Mode**

You'll use MIDI mode to configure the K2vx's interaction with other MIDI instruments, by setting parameters for transmitting and receiving MIDI. You'll also use it to configure your K2vx for multi-timbral sequencing. On the CHANLS page, you can assign a program to each channel, and enable or disable each channel's response to three types of MIDI control messages: program change, volume and pan. See Chapter 10.

### **Master Mode**

Master mode, described in Chapter 11, contains the parameters that control the entire K2vx. Global settings for tuning, transposition, velocity and aftertouch sensitivity, and audio mix are adjusted here, as well as the contrast of the backlit display. You can also access the Sample page from here.

### **Song Mode**

Song mode enables you to play MIDI type 0 sequences stored in the K2vx's RAM. It provides a fully featured sequencer that you can use to record from the keyboard. You can also record multi-timbrally via MIDI. See Chapter 12.

### **Disk Mode**

Finally, Disk mode is used to load and save programs and other objects using the K2vx's internal floppy disk drive, an optional internal SCSI disk, or an external SCSI disk (or CD-ROM drive) connected to either of the K2vx's SCSI ports. Chapter 13 has the details.

# Chapter 5

## Editing Conventions

### Introduction to Editing

Programming (editing) the K2vx always involves three basic operations: mode selection, navigation, and data entry.

First, select the mode that relates to the object you want to edit—a program, a setup, etc. Then select the object you want to edit, and press the EDIT button to enter the editor within that mode. An editor contains all the parameters that define the object you're programming.

Next, you navigate around the editor's page(s) with the soft buttons, and select parameters with the cursor buttons. When you've selected a parameter (its value is highlighted by the cursor), you can change its value with one of the data entry methods. When you change a value, you'll normally hear its effect on the object you're editing. The K2vx doesn't actually write your editing changes to memory until you save the object you're working on. It then allows you to choose between writing over the original object, or storing the newly edited version in a new memory location.

#### What's an Object?

If you've been wondering what we mean by the term "object," it's an expression we use for anything that can be named, saved, deleted, or edited. Here's a list of all the types of objects:

- **Samples**—digital recordings of instrumental sounds or waveforms. Samples actually have two separate parts: the actual sample data and the sample header information, which contains start, alternative start, loop, and end points, as well as the information on the MISC page.
- **Keymaps**—collections of samples assigned to specific velocity ranges and/or keys.
- **Programs**—factory preset or user-programmed sounds stored in ROM or RAM. A program is one or more layers of sound, with programmable DSP functions applied to the keymaps within each layer.
- **Setups**—factory preset or user-programmed MIDI performance presets consisting of three zones, each with its own program, MIDI channel, and controller assignments.
- **Songs**—Type 0 MIDI sequence files loaded into RAM, or MIDI data recorded in Song mode.
- **Effects**—factory preset or user-programmed configurations of the K2vx's onboard digital audio effects processor.
- **Quick access banks**—factory preset or user-programmed banks of ten entries each, that store programs and setups for single-button access in Quick Access mode.
- **Velocity maps**—factory preset or user-programmed curves that affect the K2vx's response to, and MIDI transmission of, attack velocity values.
- **Pressure maps**—factory preset or user-programmed curves that affect the K2vx's response to, and MIDI transmission of, pressure (aftertouch) values.
- **Intonation tables**—factory preset or user-programmed tables that affect the intervals between the twelve notes of each octave.
- **Master table**—the values that are set for the global control parameters on the Master mode page, as well as the settings for the parameters on the CHANLS page in MIDI mode, and the programs currently assigned to each MIDI channel.
- **Name table**—contains a list of dependent objects needed by the other objects in a file at the time the file was saved.
- **Macro**—list of disk files to be loaded into the K2vx's memory at start-up time.

## Object Type and ID

The K2vx stores its objects in RAM using a system of ID numbers that are generally organized into banks of 100. Each object is identified by its object type and object ID; these make it unique. An object's type is simply the kind of object it is, whether it's a program, setup, song, or whatever. The object ID is a number from 1 to 999 that distinguishes each object from other objects of the same type. For example, within the 200s bank, you can have a setup, a program, and a preset effect, all with ID 201; their object types distinguish them. You can't, however, have two programs with ID 201.

OBJECT TYPE	OBJECT ID	OBJECT NAME
Program	201	Hot Keys
Setup	404	Silicon Bebop
Velocity Map	1	Linear
Sample	3	Hey Moe

ROM (factory preset) objects have ID numbers from 1 to 199. When you save objects that you've edited, the K2vx will ask you to assign an ID. If the original object was a ROM object, the K2vx will suggest the first available ID, starting with 200. If the original object was a RAM object, you'll have the option of saving to an unused ID, or replacing the original object.

Objects of different types can have the same ID, but objects of the same type must have different IDs to be kept separate. When you're saving an object that you've edited, you *can* assign the same ID to an existing object of the same type, but if you do, the new one will be written over the old one. For example, if you assign an ID of 1 to a program you've edited, the K2vx will ask you if you want to "replace" the ROM program currently stored with that ID. We'll discuss this further in "Saving and Naming," later in this chapter.

Many parameters have objects as their values—the VelTouch parameter on the Master mode page, for example. In this case, the object's ID appears in the value field along with the object's name. You can enter objects as values by entering their IDs with the alphanumeric pad. This is especially convenient for programs, since their ID numbers are the same as their MIDI program change numbers (when you're using the Extended or Kurzweil Program Change format—see "Program Change Formats" in Chapter 10).

The object type and ID enable you to store hundreds of objects without losing track of them, and also to load files from disk without having to replace files you've already loaded. See "Memory Banks" later in this chapter for more information on object type and ID.

## Saving and Naming

When you've edited an object to your satisfaction, you'll want to store it in RAM. There's a standard procedure for saving and naming, which applies to all objects.

You can press the Save soft button, of course, but it's easier to press the EXIT button, which means "I want to leave the current editor." If you haven't actually changed anything while in the editor, you'll simply exit to the mode you started from. If you *have* made changes, however, the K2vx will ask you if you want to save those changes. This is the first Save dialog. A dialog is any display that asks a question that you need to answer before the K2vx can proceed.

Save Righteous Piano before exiting?

**Rename** **Cancel** **Yes** **No**

The best way through this process is to press the **Rename** soft button. This takes you immediately to the naming dialog, where you assign a name to the object you're saving. You haven't saved yet, but you'll be able to after you've named the program.

Program Name: Righteous Piano

**Delete** **Insert** **<<** **>>** **OK** **Cancel**

The cursor underlines the currently selected character. Press the **<<<** or **>>>** soft buttons to move the cursor without changing characters. Press an alphanumeric button one or more times to enter a character above the cursor. The characters that correspond to the alphanumeric buttons are labeled under each button. If the character that appears is not the one you want, press the button again. Press the +/- button on the alphanumeric pad to switch between upper and lower case characters.

Press 0 one or more times to enter the numerals 0 through 9. Press CLEAR (on the alphanumeric pad) to erase the selected character without moving any other characters. Press the Delete soft button to erase the selected character. All characters to the right of the cursor will move one space left. Press the "Insert" soft button to insert a space above the cursor, moving all characters to the right of the cursor one space to the right.

Press the **Cancel** soft button if you decide not to name the object. Press **OK** when the name is set the way you want to save it.

In addition to the letters and numerals, there are three sets of punctuation characters. The easiest way to get to them is to press one of the alphanumeric buttons to select a character close to the one you want, then scroll to it with the Alpha Wheel. Here's the whole list:

! " # \$ % & ' ( ) \* + , - . / 0 1 2 3 4 5 6 7 8 9

: ; < = > ? @ A through Z

[ \ ] ^ \_ ` a through z. (space).

Pressing the Plus/Minus buttons simultaneously will select 0, A, a and (space).

If you're wondering how we came up with this sequence of characters, it's composed of ASCII characters 33 through 122.

When you press **OK**, the final Save dialog appears, where you assign an ID to the edited object. If you change your mind about the name, press the **Rename** soft button for another try.

Save Righteous Piano as: ID#200

Object

Rename Save Cancel

### ROM Objects

If the object you started from was a ROM (factory preset) object, the K2vx will automatically suggest the next available (unused) ID as the ID for the edited object. If that's the ID you want, press the **Save** soft button, and the object will be stored in RAM with that ID. Otherwise, you can select any ID from 1 to 999. This screen also gives you the opportunity to return to the naming dialog (as described in the previous section) or, by pressing the **Object** soft button, to access the Object Utilities (described in Chapter 13 of this manual).

If you select an ID that's already in use, the K2vx will tell you that you're going to replace the ROM object that's already been assigned that ID. If you don't want to do that, you can select a different ID. Or you can press the Plus/Minus buttons simultaneously to toggle between the ID that the K2vx suggested and the original ID. Or press the **Cancel** soft button to cancel the operation.

If you decide not to cancel or change the ID, and you press the **Replace** soft button, the K2vx will write your newly edited object over the existing ROM object. Actually, it only appears that way, since you can't truly write to ROM. The ROM object will reappear if you delete the newly edited object (there are soft buttons in each editor for deleting objects).

### RAM Objects

If the original object was a RAM object, the K2vx will assume you want to replace it, and will suggest the same ID as the original object (if it has an asterisk—\*—between its ID and its name, it's a RAM object. As with ROM objects, you can cancel, replace, or change the ID and save to an unused ID. If you replace a RAM object, however, it's definitely gone!

## Deleting Objects

Within most editors, there are soft buttons for deleting objects. When you want to delete an object, press the **Delete** soft button, and the K2vx will ask you if you want to delete the object. Press **OK** if you want to delete it, or press **Cancel** if you don't. Although it seems that you can delete ROM objects, you can't actually do it. The K2vx will behave as if it's deleting the ROM object, but it will still be there the next time you select it. (What actually happens is that the ROM object is copied to RAM as soon as you press EDIT, and when you "delete" the ROM object, you're actually deleting the RAM copy. The original ROM object remains in memory.)

RAM objects, on the other hand, are *gone* when you delete them! If you've "replaced" a ROM object by saving a RAM object with the same ID, the ROM object is invisible, but still there. Deleting the RAM object stored at the same ID will restore the ROM object.

You can use the Delete function to delete any object from the current list of objects. After you press the **Delete** soft button, use one of the data entry methods to select any other object in the current list of objects. Then press **Delete** again. You'll often delete objects to gain RAM space, or to organize the *memory banks* before saving objects to disk. To delete multiple objects, use the Delete Objects utility available in Master mode. It's described on page 11--9.



## Memory Banks

To help organize the storage of your edited objects, the K2vx's memory is divided into ten banks, each of which stores objects having IDs within a certain range. Objects within the same range of IDs are stored in the same memory bank, regardless of their types. The banks are in increments of 100, that is, objects with IDs from 1 through 99 are stored in the first bank, IDs from 100 to 199 in the second bank, and so on. We refer to them simply as the "Zeros bank," "100s bank," "200s bank," etc. If you save an object as ID 203, for example, it's in the 200s bank.

You can store up to 100 objects of each type in each memory bank. The number of objects of a given type that can be saved in a memory bank depends on its type. For example, you can store 20 Quick Access banks in each memory bank. As you begin to save objects that you've edited, you'll notice that the IDs suggested by the K2vx sometimes increase in large chunks—from 219 to 300, for example. This is due to the limit to the number of objects of a given type that can be stored in a single memory bank. This limit can be important in terms of organizing your objects for storing to disk. Check out the section called "Storing Objects in the Memory Banks" in Chapter 2 of the *Reference Guide* for lists of how many objects of each type can fit into each memory bank.

You'll want to think about organizing your edited objects in the memory banks when you start using the floppy disk drive or a SCSI device to store your programs, samples, and other objects. Objects that are stored in the same memory banks are automatically stored in the same *file* on disk. You can also store all the banks to one file by selecting "Everything" in the Disk mode Bank *dialog*.

The memory banks work automatically, that is, you don't have to select the different banks to gain access to the objects stored in them. The K2vx selects the appropriate bank when you enter the object ID you want to work with. To select Program 201 while in Program Mode, for example, just press 2, 0, 1, ENTER on the alphanumeric pad. The 200s bank is automatically selected, and the program list will show programs numbered in the 200s. If your MIDI controller can send program change commands from 0 to 127 or 1 to 128 only, you'll probably want to adjust the way the K2vx responds to program change commands. See the discussion of the Program Change Type parameter (ProgChgType) in Chapter 10.

When you do a save operation in Disk mode, you're creating a file to be saved to disk. This will save either individually selected objects or an entire bank of objects from the K2vx's RAM. If you choose to save a complete bank, then all objects with IDs in the range of the selected bank, regardless of type, are saved as part of the file. For example, if you save the 200s bank (objects with IDs from 200 - 299), then every object with an ID from 200 to 299 will be saved to the file.

This system makes it easy for you to keep track of everything you save. The first program you save, for example, will have an ID of 200 (unless you specify another ID). The first setup you create will also have an ID of 200 (since they're different *types* of objects, the IDs can be the same). If you were to save the 200s bank, both your program and your setup would be saved to the same file.

### Saving and Loading Files—Disk Mode

Saving to disk simply involves selecting objects or a complete bank of objects to be stored as a single file. All objects with IDs within that range will be saved to the file. When you *load* a file, the K2vx asks you which bank will receive the file. You can load a file into any of the ten banks, regardless of the bank it was saved from. The K2vx will automatically reassign the object IDs. A file saved from the 200s bank, for example will be stored on disk with its objects numbered from 200–299. If you load it back into the 300s bank, its objects will be renumbered from 300–399.

See Chapter 13 for more information on loading and saving files.

## Special Button Functions

The mode buttons to the left of the display have additional functions, as described below:

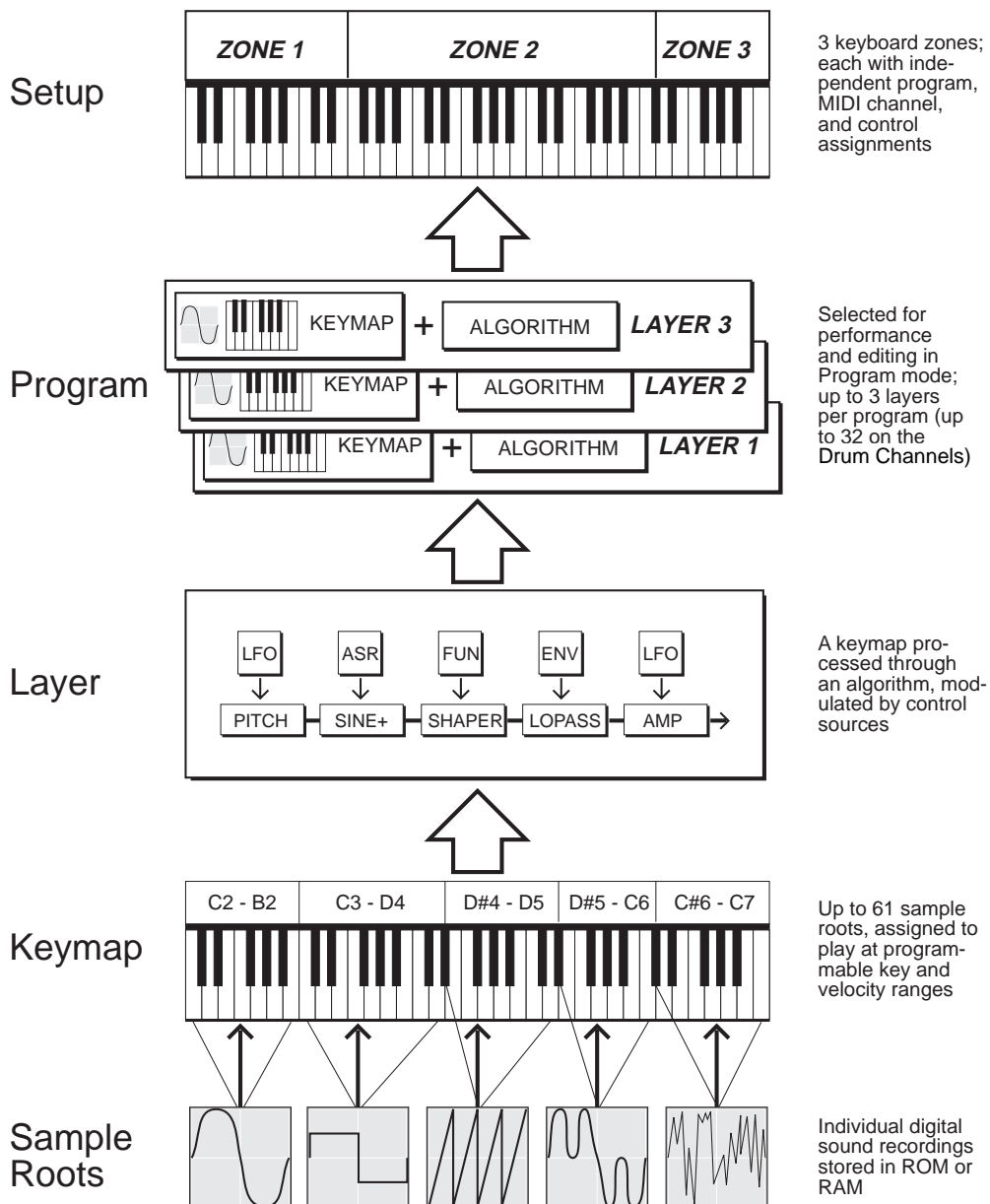
PROGRAM / Mute 1	When you're in the Program Editor, this button will mute Layer 1 of the current program or the currently displayed layer for drum programs. While in the Setup Editor, it will mute Zone 1 of the current setup. On MIXER page of Song mode, mutes either track 1 or 9.
SETUP / Mute 2	When you're in the Program Editor, this button will mute Layer 2 of the current program, if any. For drum programs, solos currently displayed layer. While in the Setup Editor, it will mute Zone 2 of the current setup. On MIXER page of Song mode, mutes either track 2 or 10.
QUICK ACCESS / Mute 3	When you're in the Program Editor, this button will mute Layer 3 of the current program, if any. For drum programs, solos currently displayed layer. While in the Setup Editor, it will mute Zone 3 of the current setup. On MIXER page of Song mode, mutes either track 3 or 11.
EFFECTS / FX Bypass	When you're in the Program Editor, pressing this button will bypass (mute) the preset effect assigned to the current program, letting you hear just the sound of the layer(s) you want to hear. On MIXER page of Song mode, mutes either track 4 or 12.
MIDI / Prev pg	In the Program Editor, pressing this button will take you to the previously selected editing page. The K2vx remembers the four most recently selected pages, so you can press this button up to four times to backtrack through the pages you've viewed. Pressing it a fifth time will take you back to the ALG page. On MIXER page of Song mode, mutes either track 5 or 13.
MASTER / Mark	This is handy for marking Program Editor pages that you use frequently. Pressing this button will mark the currently selected page. You can mark as many pages as you like. Then you can use the Jump button to select the marked pages in the order you marked them. Marked pages will show an asterisk in the top line of the display, just before the name of the page. A marked page can be unmarked by pressing the Mark button while the page is visible. On MIXER page of Song mode, mutes either track 6 or 14.
SONG / Jump	Use this button to jump to pages in the Program Editor that you've marked with the Mark button. This will cycle through all the currently marked pages in the order they were marked. On MIXER page of Song mode, mutes either track 7 or 15.
DISK / Compare	This button works in most editors, and lets you compare your edits with the original version of the object you're editing. When you press the Compare button, the display changes to remind you that you're listening to the original version. Press any button to return to the currently selected page of whatever editor you're in. On MIXER page of Song mode, mutes either track 8 or 16.
CHAN/BANK / Layer/Zone	In the Program Editor, these buttons let you scroll through the layers in the currently selected program. In the Setup Editor, you can scroll through the zones. In the Effects Editor, you can scroll through the effect configurations. In the Quick Access Editor, they scroll through the entries in the currently selected Quick Access bank. In the Keymap Editor, they scroll through the velocity levels of multi-velocity keymaps. In Song mode, switches record track.
EDIT	Whenever the selected parameter's value is an editable object or a programmable parameter, pressing the EDIT button will take you to that object's editor, or to the parameter's programming page.

# Chapter 6

## Program Mode and the Program Editor

Program mode is the heart of the K2vx, where you select programs for performance and editing. The K2vx is packed with great sounds, but it's also a synthesizer of truly amazing depth and flexibility. When you're ready to start tweaking sounds, the Program Editor is the place to start. But first there's a bit more general information about Program mode that wasn't covered in Chapter 2. Refer to the illustration below as you read the sections that follow.

### K2vx Program Structure



## K2vx Program Structure

Programs are the K2vx’s performance-level sound objects. They’re preset sounds that can be played on any of the 16 MIDI channels. Other synths refer to them as patches, presets, voices, multis, etc. Each program consists of from one to three *layers* (or up to 32 layers for a drum program - see below), each of which in turn consists of a keymap and an algorithm for processing the samples that make up the keymap. Samples are stored in the K2vx’s ROM, or loaded into optional RAM via Disk mode, MIDI standard sample transfer, SMDI sample transfer, or by your own sampling efforts. Each sample is a separate digital recording of an instrumental note, a drum, a waveform or a sound effect. The individual samples are assigned to specific key ranges and are assigned to be triggered at specific attack velocities. These assignments constitute the keymap.

When you trigger a note, the K2vx looks to the keymap of each *layer* of the currently active program(s) to determine which samples to play. The sound engine then fetches the requested samples and generates a digital signal that represents the timbres of the samples. This signal first passes through the five DSP functions that make up the algorithm, then through the global effects processor (if assigned to an audio output pair that uses effects), then finally appears at one or more of the audio outputs.

The layer is the K2vx’s basic unit of polyphony, that is, each layer constitutes one of the 48 voice channels the K2vx can activate at any time. If you have a program that consists of two layers covering the note range from C 0 to C 8, each note you trigger will trigger two voice channels.

## The Program Mode Page

```

ProgramMode XPose:051 <>Channel:1
199 Default
KeyMap Info 209*(Dig it al
  Grand Piano 1 Righteous Piano
                2 Mondo Bass
                3 Killer Drums
                4 Weeping Guitar
Octav- Octav+ Panic View Chan- Chant
    
```

The top line of the Program mode entry level page shows your location, the present MIDI transposition, and the current MIDI channel.

The box at the left of the Program mode page tells you which keymap is assigned to each layer. Layer 1 is at the top. Layers 2 and 3, if any, are listed below layer 1. If a *drum program* is selected, the box tells you how many layers it has. If a drum program is selected, and the current MIDI channel is not a drum channel, the program’s name will appear in parentheses in the display, and the program will not play (more on this in the next section). The box at the left of the display will tell you which channels are drum channels. (See the Drum Channel parameter in Chapter 11).

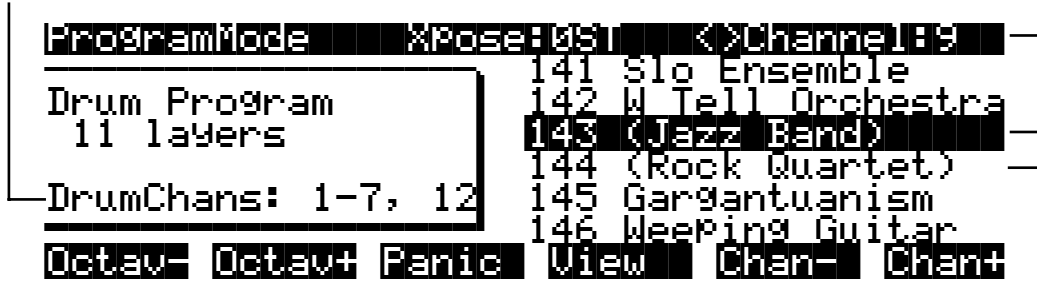
The line beneath the name of the keymap indicates the keyboard range of the layer. On the Program mode page above, for example, the layer extends from C 0 to C 8—the default range. The representation of these layer ranges is approximate; they’re intended to let you know if you have a layered keyboard (lines overlapping) or a split keyboard (lines not overlapping).

## What are these programs in parentheses?

While you are scrolling through different programs on various MIDI channels, you may occasionally see a program that is in parentheses and doesn’t make any sound. The parentheses

tell you that you have selected a drum program without being on one of the drum channels. Unlike normal programs on the K2vx, which can contain up to 3 layers, drum programs can have as many as 32 layers.

Current drum channels are 1-7 and 12.



Current MIDI channel is 9. Parentheses around drum program names show that they won't sound on this channel.

Even though drum programs don't have to have drum sounds assigned to them, we use the name drum program because that is the most likely purpose for a program with 32 layers. The only limitation for a drum program is that it must be played on a MIDI channel that has been designated as a drum channel. A drum channel can play any program - it does not have to play a drum program, but a drum program must be played on a drum channel.

You can have up to 8 drum channels on the 2000. Channels 1 through 8 are drum channels by default, but you can change this to be channels 1 through 7 plus the channel of your choice. To set that channel, go to the Master page and set the Drum Channel parameter to any value between 9 and 16 to select that channel plus 1 through 7. In the example above, DrumChan on the Master page has been set to 12.

## The Soft Buttons in Program Mode

If the value of the Local Keyboard Channel parameter (on the RECV page in MIDI mode) matches the transmission channel of your MIDI controller, you can use the Octav- or Octav+ soft buttons for quick transposition. The top line will reflect the change. This will transpose the K2vx as well as any MIDI devices connected to the K2vx's MIDI Out port. Press both Octave soft buttons simultaneously to reset the transposition to zero. Changing the transposition with the soft buttons will also change the corresponding setting on the MIDI XMIT page. If the Local Keyboard Channel parameter is not set to match the transmission channel of your MIDI controller, the Octave soft buttons will have no effect.

Pressing the Panic soft button will send an All Notes Off message and an All Controllers Off message on all 16 MIDI channels. If you press the View soft button, the current program's name will be displayed in large letters. Press it again to return to the normal display.

Use the Chan- and Chan+ soft buttons to change the current MIDI channel. This changes the MIDI channel the K2vx uses internally, as well as the channel you're using to send information to other synths connected to the K2vx's MIDI Out port (MIDI slaves). Changing the current MIDI channel with the soft buttons will also change the corresponding setting on the MIDI XMIT page.

## Using the Program Editor

The Program Editor is where you begin to modify the K2vx's resident sounds, and to build your own sounds around samples (ROM or RAM) and/or waveforms. There's virtually no limit to the sounds you can create using the tools in the Program Editor.

To enter the Program Editor, start in Program mode and press EDIT. The Program mode LED will go out, and the ALG (Algorithm) page will appear.

```
EDITPROG:ALG <>Layer:1/1
Algorithm: 1
  PITCH → NONE → AMP →
<more> ALG LAYER KEYMAP PITCH <more>
```

The top line of the display gives you the usual reminder of your location. It also tells you which layer you're viewing, and how many layers there are in the program. You can use the CHAN/BANK buttons to scroll through the layers, if the program has more than one.

Here's a method for jumping quickly to a specific layer in a program that is especially useful in multi-layered drum programs. Hold the ENTER button and strike a key on your controller. The display will change to the layer assigned to that key. If more than one layer is assigned to the same key, repeatedly striking the key (while continuing to hold the ENTER button) will cycle through all layers assigned to that key. This method will work in most places within the Program Editor, but there is an exception: if the parameter you have highlighted has a note number or control source as its value, then holding ENTER and striking a note will call up that note or control source (as described in the "Intuitive Entry" section on page 3--6). For all other parameters, however, this method will switch between layers.

## The Soft Buttons in the Program Editor

The Program Editor's soft buttons are labeled by the words that appear in the bottom line of the display. These buttons have two important jobs in the Program Editor: selecting pages, and selecting specific functions. If a soft button is labeled in all uppercase letters, pressing it will take you to the page it describes. If the button is labeled in mixed uppercase and lowercase letters, pressing it will execute the software function described by the label. Pressing the PITCH soft button, for example, will select the PITCH page, while pressing the Save soft button will initiate the process for saving the currently selected program.

There are more pages and functions in the Program Editor than there are soft buttons. Therefore, two of the soft buttons are dedicated to scrolling through the list of pages and functions. If you don't see the button for the page or function you want to select, press one of the soft buttons labeled "<more>," and the labels will change. This doesn't change the currently selected page, it merely changes the selection of available soft buttons.

Five of the soft buttons in the Program Editor are special cases. They're the soft buttons that select the editing pages for the five control input pages for the *DSP functions*. One of these soft buttons is always labeled "PITCH," since the first DSP function in every algorithm is the pitch control. The remaining four vary somewhat depending on the DSP functions you choose for the currently selected algorithm, but they always have the prefixes F1, F2, F3, and F4, and they always take you to the pages for the four DSP functions that follow the pitch control function.

## Algorithm Basics

The basic definition: an algorithm is the “wiring” (signal path) of a sample to the audio outputs, through a series of digital signal processing (DSP) functions that you select. The K2vx’s algorithms are the core of Variable Architecture Synthesis Technology. The DSP functions are synthesis tools (filters, oscillators, etc.) that you assign to the various stages of the algorithm. The DSP functions you choose determine the type of synthesis you use.

Each of the 31 available algorithms represents a preset signal path. You can’t change the path of the algorithms, but you can select different algorithms, and assign a wide variety of DSP functions to the individual stages of each algorithm’s signal path. Take a look at Algorithm 1 in the diagram below. It’s one of the simplest algorithms.

### Algorithm 1



The DSP functions are represented by the rectangular blocks. The horizontal arrows indicate the flow of the digital signal from left to right. They represent what we call the “wire” of the algorithm, the actual physical path that the signal follows through the algorithm. Selecting different algorithms can be compared to connecting different DSP functions with different wiring diagrams.

Think of the left side of each block as its input, and the right side as its output. Depending on the algorithm, the signal may split into two wires, enabling part of the signal to bypass certain portions of the algorithm. Split wires may rejoin within the algorithm, or they may pass all the way through as split signals. If the last block has two wires at its output, we call it a double-output algorithm. If it has one wire, it’s a single-output algorithm, even if there are two wires in earlier portions of the algorithm.

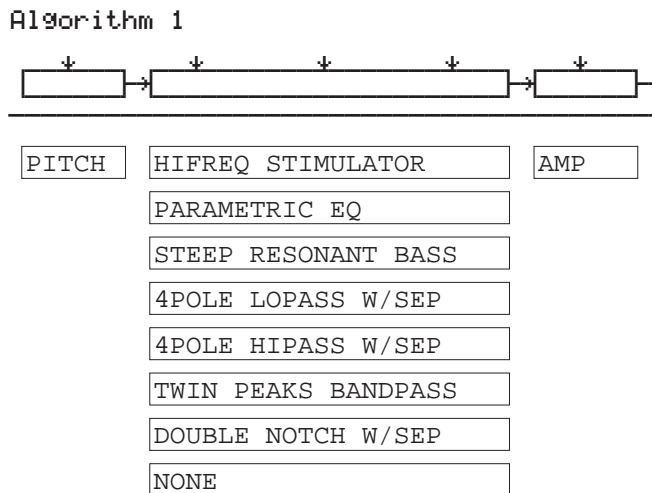
The five downward-pointing arrows indicate the five realtime control inputs to the DSP functions. There are usually five inputs, each of which has its own page within the Program Editor. (Algorithms 26-31, which use hard sync oscillation, have only four inputs; you can read about hard sync functions on page 14--50.) Each of these pages has several parameters that can modulate its related DSP function. Often a single DSP function will have more than one input. That’s why some blocks are larger, and have more than one arrow pointing to them. Each function can be independently controlled by a variety of sources (the Control Source list), including LFOs, ASRs, envelopes, programmable functions, and external MIDI.

In Algorithm 1, the signal flows first through a one-stage DSP function that controls the pitch of the samples in the keymap. In fact, the first DSP function in each algorithm *always* controls pitch, even though it doesn’t apply in every instance. Similarly, the last DSP function always controls the final amplitude of the signal. It can be a one-stage or two-stage function. In Algorithm 1, it’s a one-stage function.

The second, larger block indicates a single three-stage DSP function, meaning that it has three control inputs that can be adjusted to modulate the signal.

Once again, you can’t change the wiring path of an algorithm—you can simply select a different algorithm to get a different path. And within each algorithm, you can assign a large number of different DSP functions to each of the five control inputs. The diagram below, for

example, shows Algorithm 1 with all the possible values for each DSP function lined up under the blocks that represent the DSP functions.



Notice that PITCH is the only value available for the first block, and AMP is the only value available for the last block. The center, three-stage block, however, allows you to choose from seven DSP functions. An eighth value, NONE, deactivates the block.

## Common DSP Control Parameters

The type of DSP function available for any function block depends on the algorithm. Some of the specialized functions like the PANNER are always located just before the final AMP function. Others, like the three-input functions, appear only in algorithms that are structured for three-input functions.

You can change the nature of each layer of a program simply by assigning different DSP functions to the layer's algorithm. Your level of control goes much deeper than that, however. Each DSP function has one or more inputs to which you can patch a variety of control sources to modify the behavior of the DSP functions themselves. These control inputs are represented by the arrows pointing down at the blocks that make up the algorithm. For each input arrow, there's a corresponding control input page that you can select with the five special soft buttons we mentioned above (PITCH, and F1-F4). All of the DSP functions have at least one control input, but many of them have two or even three inputs.

The parameters on the various control input pages are very similar; in fact, there are six parameters that appear on almost every page. Consequently we refer to them as the common DSP control parameters. Although the parameters on the control input pages differ slightly from function to function, you can expect to see some or all of the common DSP control parameters whenever you select the control input page for any of the DSP functions. They are:

### Initial Setting Parameters

These have no input, but set the overall level for the function, the starting point from which the other parameters modulate the function.

Coarse adjust (abbreviated "Coarse")

Fine adjust (Fine)

### Hard-wired Parameters

These always take their input from your MIDI controller; the note number and the attack velocity values of each Note On event.



Key tracking (KeyTrk)  
 Velocity tracking (VelTrk)

**Programmable Parameters**

These can accept any control source as their input, and have related parameters for further control.

Source 1 (Src1)  
 Source 2 (Src2)

Take a look at the PITCH page, as an example—we’ll look at how these six control parameters are used in the pitch control function. If you’re not already on the PITCH page, you can get there by pressing the soft button labeled “PITCH.” If you don’t see “PITCH” on the bottom line of the display, press one of the “<more>” buttons until it appears.

```

editProg: PITCH Layer: 1/1
Coarse: 0ST Src1 : OFF
Fine : 0ct Depth : 0ct
FineHz: 0.00Hz Src2 : OFF
KeyTrk: 0ct/key DptCtl: MWheel
VelTrk: 0ct MinDpt: 0ct
MaxDpt: 0ct
<more> ALG LAYER SAMPLE PITCH >more>
    
```

You’ll recognize the common DSP control parameters, along with several other parameters. Keep in mind that there’s a set of common control parameters for each of the DSP functions; in this case we’re describing them only as they apply to the pitch control function.

**Coarse Adjust**

The Adjust parameter (sometimes coarse and fine adjust) is the fixed amount of adjustment you add to any DSP function. On the PITCH page, the Coarse Adjust parameter will change the pitch in semitone increments. Use this as a starting point to set the pitch where you want it to be normally. This will shift the pitch of the currently selected layer, and will affect the playback rate of sampled sounds. Sampled sounds have an upper limit on pitch adjustment. It’s normal for the pitches of sampled sounds to “pin” (stop getting higher) when you adjust the pitch upward in large amounts. The oscillator waveforms can be pitched higher. Any sound can be pitched downward without limit.

The primary use of the Adjust parameters (Coarse *and* Fine) is to offset the cumulative effects of the other parameters on the control input pages. For example, you might set a high value for key tracking (defined below) for a dramatic change in effect across the keyboard. The effect might be too much at one end of the keyboard, however, so you could use one of the Adjust parameters to reduce the initial amount of that effect.

The K2vx always uses real values of measurement, rather than just arbitrary numbers, for adjustable parameters. This means that you specify pitch in semi-tones and cents, amplitude in dB, and filter cutoff frequency in hertz.

Remember that the parameters on the control input pages are cumulative—they can add to or subtract from the effects of the other parameters on the page, depending on their values. For example, even if you’ve adjusted the pitch of a sample so high that it pins, the effects of the other parameters may bring the pitch back down to a workable range.

### Fine Adjust

You can add slight detuning to the pitch by changing the fine adjust parameter. Notice that there are actually two fine adjust parameters on the PITCH page: one that changes the pitch in cents (100ths of a semitone), and one that changes it according to its frequency (in increments of Hertz—cycles per second). Since we're discussing the universal control sources here, and not specifically pitch, we'll move on for now, as the "Fine Hz" parameter applies only to pitch-related functions. See "The PITCH Page," later in this chapter, which describes Fine Hz more thoroughly.

### Key Tracking

This is a quick way to get additional control based on the MIDI note number of each note you trigger. Key tracking applies a different control signal value for each note number. In the case of pitch, key tracking enables you to change the tuning of each note relative to its normal pitch.

Middle C is the zero point. Regardless of the key tracking value, there is no effect on Middle C. If you set a non-zero value for key tracking, the effect increases for each note above or below Middle C. In the case of pitch, for example, say you assign a value of 5 cents per key for the key tracking parameter. Triggering Middle C (C 4 on the K2vx) will play a normal C 4. Triggering C# 4 will play a note 5 cents higher than C# 4. Triggering D 4 will play a note 10 cents higher than D 4, and so on. Notes below Middle C will be tuned lower than their normal pitches. If you set a negative value for key tracking, notes above Middle C will be tuned lower than their normal pitches.

Keep in mind that key tracking on the PITCH page works in conjunction with the key tracking parameter on the KEYMAP page. This is why you can set the KeyTrk parameter on the PITCH page to 0ct/key, and the K2vx will still increase in pitch by 100 cents/key as you go up the keyboard. It's because the KeyTrk parameter on the KEYMAP page is already set at 100 cents/key.

### Velocity Tracking

A positive value for velocity tracking will raise the pitch as you trigger notes with higher attack velocities. This is great for getting a trace of detuning based on your attack velocity, especially in drum programs, where you can make the pitch of the drum samples rise slightly with higher-velocity Note Ons, just as drums do when you strike them harder. Negative values will lower the pitch as you increase the attack velocity.

### Source 1

This parameter takes its value from a long list of control sources (you can find it in the *Reference Guide*—it's called the Control Source list) including every MIDI control number, a host of LFOs, ASRs, envelopes and other programmable sources.

Src1 works in tandem with the parameter beneath it on the page: Depth. Choose a control source from the list for Src1, then set a value for Depth. When the control source assigned to Src1 is at its maximum, the pitch will be altered to the full depth you set. For example, if you set Src1 to "MWheel," and set Depth to 1200 ct, the pitch will rise as you push the Mod Wheel up on your MIDI controller, reaching a maximum of 1200 ct (12 semitones, or one octave).

### Source 2

This one's even cooler. Like Src1, you choose a control source from the list. But instead of setting a fixed depth, you can set a minimum and maximum depth, then assign another control source to determine how much depth you get. Try this example. (Make sure Src1 is set to OFF first, so the two sources don't interact). Start with Program 199, and press EDIT. Press the PITCH soft button to select the PITCH page. Set the Src2 parameter to a value of LFO1, then set the Minimum Depth parameter to 100 ct, and Maximum Depth to 1200 ct. Then set the Depth

Control parameter to MWheel. This lets you use your MIDI controller's Mod Wheel to vary the depth of the oscillation in pitch generated by the LFO.

Now, when the Mod Wheel is down, the pitch will oscillate between a semitone (100 ct) up and a semitone down (the default waveform for LFO1 is a sine wave, which goes positive and negative—if this perplexes you, see the *Reference Guide*, where there's an explanation of how the K2vx generates and interprets control source signals). With the Mod Wheel up, the pitch will oscillate between an octave up and an octave down.

Since the Mod Wheel is a continuous control, you can achieve any amount of depth control between the minimum and maximum. If you had set the Depth Control to Sustain, for example, then you'd get only two levels of depth control: the maximum (1200 cents) with your MIDI controller's sustain pedal down, or the minimum (100 cents) with the sustain pedal up.

### Summary of Common DSP Control Parameters

These six control source parameters are just a few of the control sources available throughout the Program Editor. We've given them special attention because they appear on *all* the pages relating to the DSP functions, not just on the PITCH page.

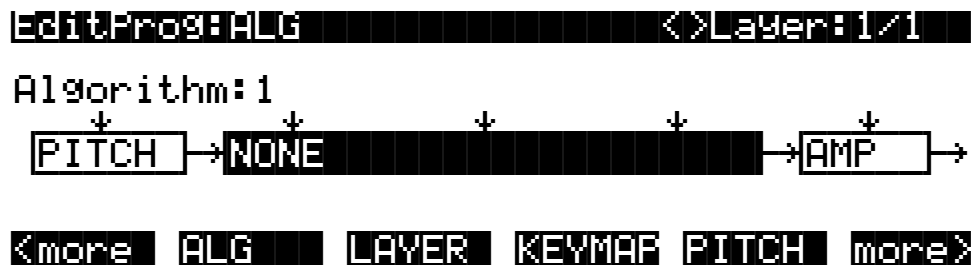
As with the PITCH parameters, you can go to each of the DSP functions' control input pages, and set a similar set of parameters to control each of those functions as well. The units of measurement may differ, but you'll almost always find one or two adjustment parameters, key and velocity tracking, and two programmable control sources. And remember, we've been talking about one layer in one program here. You can add one or two more layers to your program, and start all over with another identical set of control sources for each layer, each of which can be programmed independently.

On any given page, the settings for the control parameters are added to each other before the signal leaves the DSP function. Depending on the values you set, they may cancel each other out, or they may add up to huge amounts of modulation. If things get out of control, the easiest way to get a handle on the situation is to set some of the parameters to values of 0 or OFF. Adjust the value for one parameter at a time to hear the effect of that one parameter.

## The Program Editor—Pages

### The Algorithm (ALG) Page

The ALG page is the first page you see when you enter the Program Editor. It enables you to select from among the 31 possible algorithms, and assign the DSP functions within the current algorithm.



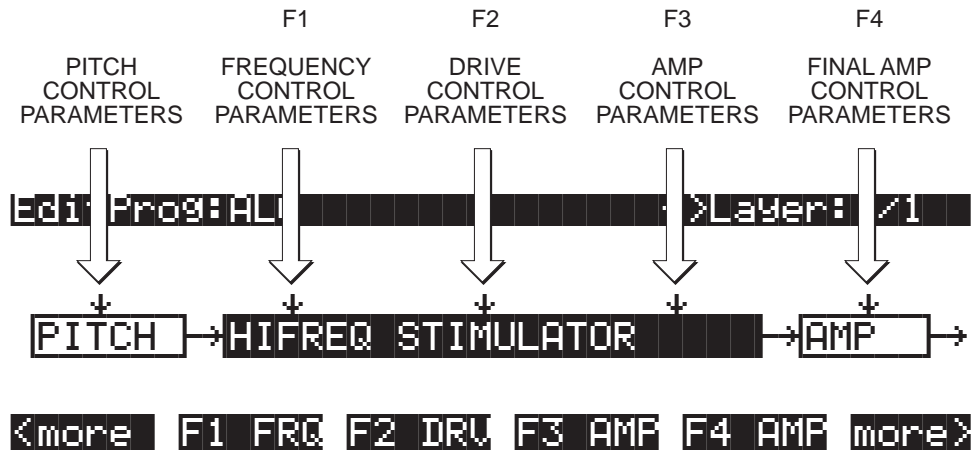
The top line of the display gives you the usual mode reminder, and tells you which layer you're looking at, as well as how many layers are in the current program (in the diagram above, it's the first layer of a one-layer program). You can view the ALG pages of any other layers in the program by using the CHAN/BANK buttons.

The central portion of the page shows the algorithm for the currently selected layer. You see the number of the algorithm (from 1 to 31) and a graphic representation of the signal path, as well as the currently selected DSP functions within the signal path.

To use a different algorithm, select the Algorithm parameter and use any data entry method to select a different one. To change the DSP function within an algorithm, move the cursor to the block you want to change, then use the Alpha Wheel or Plus/Minus buttons. There's a staggering number of combinations of algorithms and DSP functions alone, not to mention the numerous controls that can be used to modify the DSP functions. The *Reference Guide* contains a list of all 31 algorithms and the DSP functions available for each one.

NOTE: Changing a layer's algorithm can affect the layer's sound drastically. It's a good idea to bring down the volume of your K2vx or your sound system before changing algorithms.

The five downward-pointing arrows represent inputs to the DSP functions that are available for the current algorithm. Each input arrow has its corresponding page. The first arrow points to the PITCH function. The soft button for the PITCH page is already visible. Press it to view the parameters affecting pitch for the currently selected layer. The buttons for the other four DSP functions are not visible when you first enter the Program Editor. To see them, press the more> soft button (on the right side of the page). You're still on the same page, but the soft buttons' labels change to let you select a different set of pages, as the diagram on the next page shows.



The pages (F1–F4) that control the DSP functions are described later in this chapter. There's some general information on algorithms earlier in this chapter; see "Algorithm Basics." Finally, Chapter 14 will give you a thorough description of each the DSP functions and the parameters found on their respective editing pages.

### The LAYER Page

Press the LAYER soft button to call up the LAYER page. Here you'll set a number of parameters that affect the current layer's keyboard range, attack and release characteristics, and response to various controls.

```

EditProg: LAYER <>Layer: 1/1
LoKey :C 0  DlyCtl:OFF      SusPdl:On
HiKey :C 8  MinDly:0.000s  SosPdl:On
LoVel :ppp  MaxDly:0.000s  FrzPdl:On
HiVel :fff  Enable:ON     IgnRel:Off
PBMode:All  S:Norm 64 127 ThrAtt:Off
Trig  :Norm Opaque:Off   TilDec:Off
<more>  ALG  LAYER  KEYMAP  PITCH  >more>
    
```

PARAMETER	RANGE OF VALUES	DEFAULT
LOW KEY	C -1 to G 9	C 0
HIGH KEY	C -1 to G 9	C 8
LOW VELOCITY	ppp to fff	ppp
HIGH VELOCITY	ppp to fff	fff
PITCH BEND MODE	Off, Key, All	All
TRIG	Normal, Reversed	Normal
DELAY CONTROL	Control Source list	OFF
MINIMUM DELAY	0 to 25 seconds	0
MAXIMUM DELAY	0 to 25 seconds	0
LAYER ENABLE	Control Source list	ON
ENABLE SENSE	Normal, Reversed / Min. / Max.	Normal / 64 / 127
OPAQUE LAYER	Off, On	Off
SUSTAIN PEDAL	Off, On	On
SOSTENUTO PEDAL	Off, On	On
FREEZE PEDAL	Off, On	On
IGNORE RELEASE	Off, On	Off
HOLD THROUGH ATTACK	Off, On	Off
HOLD UNTIL SUSTAIN	Off, On	Off

### Low Key (LoKey)

This sets the lowest active note for the current layer. This parameter’s value cannot be set higher than the value for HiKey. The standard MIDI key range is C -1—G 9 (0-127). Middle C is C 4 (ISP).

### High Key (HiKey)

Here you set the highest active note for the current layer. This parameter’s value cannot be set lower than the value for LoKey.

### Low Velocity (LoVel)

With this parameter you define the lowest attack velocity at which the layer will be enabled (generate a sound). The values for this parameter and the next are expressed in the standard musical dynamics markings, similar to the values available for the velocity maps. Attack velocities that are below this velocity threshold will not trigger notes. If you set this parameter’s value higher than the HiVel value, the layer will not play at all.

### High Velocity (HiVel)

Similarly, this will set the highest attack velocity at which the layer will be enabled. Attack velocities above this velocity will not trigger notes in this layer.

### Pitch Bend Mode (PBMode)

This determines how Pitch bend control messages will affect the current layer. A value of “All” bends all notes that are on when the Pitch bend message is generated. A value of “Key” bends

only those notes whose triggers are *physically* on when the Pitch bend message is generated (notes held with the sustain pedal, for example, won't bend). This is great for playing guitar solos on top of chords—play a chord, hold it with the Sustain pedal, then play your licks and bend them all you want; the chord won't bend with it. A value of "Off" disables Pitch bend for the current layer.

### Trigger (Trig)

Set Trig to "Rvrs" to have notes triggered on key-up. The initial velocities of notes triggered this way are determined by the release velocities of the keys that trigger them. The default setting is "Norm"

### Delay Control (DlyCtl)

Here you select, from the Control Source list, a control source that will delay the start of all notes in the current layer. The length of the delay is determined by the following two parameters. You'll assign a continuous control like MWheel for the DlyCtl parameter when you want to vary the delay time, and a switch control if you want the delay to either be its minimum value (switch off), or its maximum (switch on).

The delay control will affect only those notes triggered *after* the delay control source is moved; the delay time is calculated at each note start, based on the status of the delay control source at that time.

### Minimum Delay (MinDly), Maximum Delay (MaxDly)

The length of the delay is determined by these two parameters. When the control source assigned to DlyCtl is at its minimum, the delay will be equal to the value of MinDly. The delay will be equal to the value of MaxDly when the control source is at its maximum. If DlyCtl is set to OFF, you get the minimum delay. If it's set to ON, you get the maximum delay. This doesn't change the note's attack time, just the time interval between the Note On message and the *start* of the attack. The delay is measured in seconds.

### Enable

This selects a control source to activate or deactivate the layer. When the assigned control source is on (or above the midpoint of 64 for continuous controls) the layer will be active. The layer will not sound when the control source is off or below its midpoint.

For example, if you wanted to create an octave doubler that would kick in on demand, you could create a program with a second layer, transpose it up or down an octave, and set its Enable parameter to any control source, like MWheel. Then whenever your MIDI controller's Mod Wheel is above its midpoint, you'll hear the second layer.

### Enable Sense (S)

This lets you reverse the orientation of the MIDI control you've assigned to enable the layer. A value of Normal has no effect on the Enable parameter, while Reversed will activate the layer when the Enable control source is off or below its midpoint, and deactivate the layer when the Enable control source is on or above its midpoint.

You could use this parameter to set up a two-layer program that would let you use a MIDI control to switch between layers, say a guitar sound and a distorted guitar. Both layers would have their Enable parameters set to the same control source, say MWheel. One would have its Enable Sense parameter set to Normal, and the other would have it set to Reverse. Then the first layer would play when your MIDI controller's Mod Wheel was above its midpoint, and the second layer would play when the Mod Wheel was below its midpoint.

This is also the place to set the minimum and maximum values at which the layer will be enabled. The range for each is -128-127.

### **Opaque**

An opaque layer blocks all higher-numbered layers in its range, allowing only the opaque layer to play. This is an easy way to change a small range of notes in a program, leaving the original sound playing above and below the new sound. Just create a new layer (Layer 2), set its range (say, C 3 to D 3), assign it the keymap you want, and set its Opaque parameter to On. Then duplicate the original layer, so its number is higher than that of the new layer (the duplicate layer is Layer 3). You'll now have a three-layer program. Delete Layer 1 (the original layer), and the new Layer will become Layer 1, while the duplicated original layer becomes Layer 2. Now the new layer will block out the original layer at the notes C 3–D 3.

### **Sustain Pedal (SusPdl)**

When this parameter is on, the layer will respond to all sustain messages (MIDI 64) received by the K2vx. When off, the current layer will ignore sustain messages.

### **Sostenuto Pedal (SosPdl)**

When Sostenuto is on, the layer will respond to all sostenuto messages (MIDI 66) received by the K2vx. When off, sostenuto messages are ignored by the layer.

### **Freeze Pedal (FrzPdl)**

Activate or deactivate the layer's response to Freeze pedal messages (MIDI 69). The Freeze pedal control causes all notes that are on to sustain without decay until the Freeze pedal control goes off. If a note is already decaying, it will freeze at that level.

### **Ignore Release (IgnRel)**

When on, the layer will ignore all Note Off messages received by the K2vx. This should be used only with sounds that decay naturally, otherwise the sounds will sustain forever. When IgnRel is off, the layer responds normally to Note Off messages.

This parameter can come in handy when your K2vx is slaved to a drum machine or sequencer, which sometimes generates Note Ons and Note Offs so close together that the envelope doesn't have time to play before the note is released. You'll also want to use this parameter when you're playing staccato, and the sound you're playing has a long amplitude envelope. This parameter should be used only with notes that eventually decay to silence. Sustaining sounds will sustain forever.

### **Hold Through Attack (ThrAtt)**

When on, this parameter causes all notes in the layer to sustain through the entire first attack segment of their amplitude envelopes, even if the notes have been released.

If you have a sound with a slow attack, or an attack that's delayed with the delay control, setting this parameter to On will make sure your notes reach full amplitude even if you're playing fast. When set to Off, notes will release as soon as you release the note (generate a Note Off). If the first attack segment of the layer's amplitude envelope is very short, you probably won't notice a difference between values of On and Off.

### **Hold Until Decay (TilDec)**

When on, this parameter causes all notes in the layer to sustain through all three attack segments in their amplitude envelopes even if the notes have been released. Looped amplitude envelopes will not loop, however, if the notes are released before reaching the end of the final attack segment. Notes will go into their normal releases if they are released after the envelope has looped. When set to Off, notes will release as soon as a Note Off message is generated.

## The KEYMAP Page

Press the KEYMAP soft button to call up the KEYMAP page. The parameters on this page affect sample root selection—which samples are played on which keys.

```

EditProg:KEYMAP          <>Layer:1/1
KeyMap:1 Grand Piano    Stereo:Off
XPose :0ST              TimbreShift :0ST
KeyTrk:100ct/key       AltSwitch  :OFF
VelTrk:0ct              PlaybackMode:Normal

<more  ALG  LAYER  KEYMAP  PITCH  more>
    
```

PARAMETER	RANGE OF VALUES	DEFAULT
KEYMAP	Keymap list	1 Grand Piano
TRANPOSE	± 60 semitones	0
KEY TRACKING	± 2400 cents per key	100
VELOCITY TRACKING	± 7200 cents per key	0
STEREO	Off, On	Off
TIMBRE SHIFT	± 60 semitones	0
ALT SWITCH	Control Source list	OFF
PLAYBACK MODE	Norm, Rvrs, Bidirectional, Noise	Normal

### Keymap

Assign a ROM or RAM keymap to the current layer. Keymaps are collections of samples assigned to note and velocity ranges. There are nearly 200 ROM keymaps to choose from. You'll find a list of them in the *Reference Guide*.

### Transpose (Xpose)

Transpose the current keymap up or down as much as 60 semitones (5 octaves).

### Key Tracking (KeyTrk)

This is one of the six common DSP control parameters. On the KEYMAP page, key tracking affects the interval between notes. The default value of 100 cents (hundredths of a semitone) gives you the normal semitone interval between each note. Higher values increase the interval; lower values decrease it. Negative values will cause the pitch to decrease as you play higher notes. You can create a mirror-image piano by setting the key tracking to -100 and transposing the layer up 4 semitones. When you make changes to this parameter, you'll need to keep in mind that KeyTrk on the KEYMAP page works in conjunction with KeyTrk on the PITCH page. Therefore, you'll need to check the KeyTrk value on both pages to see how key tracking works within a program.

### Velocity Tracking (VelTrk)

This is another common DSP control parameter. As with the other parameters on the KEYMAP page, this shifts the position of the keymap. Different attack velocities will play different pitch shifts of the sample root assigned to that note range. If the shift is great enough, the next higher or lower sample root will be played, which in some cases (like the drum programs) will play an entirely different sound. Positive values will play higher pitches of the sample root when you use hard attack velocities (they shift the keymap downward), while negative values will play lower pitches.



## Stereo

You'll use this parameter when you're working with stereo samples. When you use the optional stereo piano program or load stereo samples from disk, the K2vx views both sides of the sample as a single sample object. When you select a stereo sample as the value for the Sample parameter, you'll see the letter "S" as part of the sample name (for example, "204\*StratoBlaster E3 S").

When you set this parameter to On, the KEYMAP page changes slightly:

```

EditProg: KEYMAP <>Layer: 1/1
KeyMap1: 1 Grand Piano
KeyMap2: None Stereo: On
Xpose : 0ST TimbreShift : 0ST
KeyTrk: 100ct/key AltAttackCtl: OFF
VelTrk: 0ct PlaybackMode: Normal

<more> ALG LAYER KEYMAP PITCH >more>

```

An additional Keymap parameter appears. The two keymap parameters are distinguished as Keymap 1 and Keymap 2. The KEYMAP page parameters will affect both keymaps. When the Stereo parameter is on, the OUTPUT page for the current layer will show an additional pair of Pan parameters.

To get the samples to play together, set the Stereo parameter to On, and select the keymap as the value for *both* the Keymap 1 and Keymap 2 parameters. The K2vx automatically uses the left side for Keymap 1, and the right side for Keymap 2. Then go to the OUTPUT page and set the panning for each sample as desired. Keep in mind that using stereo keymaps reduces the polyphony of the program. For example, if you had a two-layer program with stereo keymaps in each layer, each note you play would use 4 of your 48 voices, allowing a total of 12 notes before all the voices have been used.

This parameter is unnecessary if you're not using stereo samples, and should be set to Off in that case.

## Timbre Shift

This parameter works only on multi-sample keymaps, and changes the root selection for each key you play. With this parameter you can radically alter the current layer's timbre (basic sound characteristics). The nature of the change depends on the timbre itself, so this parameter calls for experimentation. Basically, timbre shifting changes a note's timbre by imposing different harmonic qualities onto the note. A timbre-shifted note retains its original pitch, but its harmonics are those of the same timbre at a higher or lower pitch. Positive values for this parameter tend to brighten a sound, while negative values darken.

Here's an example. If you shift the timbre up 4 semitones, then playing C 4 will result in the *pitch* C 4, but will actually play the sample normally assigned to G# 3, and shift its pitch up 4 semitones. This will increase the playback rate of the sample, so although the pitch remains normal, the timbre is brighter. You'd get the same effect by setting the Xpose parameter on the Keymap page to -4 semitones, then setting the Adjust on the PITCH page to + 4 semitones. For multi-sample layers with narrow key ranges, large amounts of timbre shifting will cause different sample roots to be played back.

## Alternative Switch (AltSwitch)

You can assign a control source to change the sound by using an alternate start point or alternate end point for the current keymap. Whether or not it is an alternate start or alternate end depends on the position of the Alt parameter for the sample (set in the Sample Editor).

When set before the end point, it is used as an alternate start (the Alt point can be before or after the normal Start point). When set after the end it is used as an alt end.

If you place the Alt point after the initial attack transients of the sample, then you can use the Alt Switch to emulate legato playing in an acoustic instrument. As an example, set the Keymap to "#14 Flute". Now set the Alt Switch parameter to Chan St (Channel State). Now if you play notes separately, the initial breathy chuff will be heard. But if you play the notes legato (connecting them smoothly), the Alt point is used and you do not hear the chuff. This is because the Chan St is turned on as long as any note is being held. Most of the K2vX's ROM samples have their Alt points set for purposes of legato play. In most cases the difference in attacks is subtle, but for some sounds, like drums, the difference can be more noticeable.

For more information, refer to the discussion of the TRIM page's Alt point on page 15--14.

### Playback Mode

This gives you four options for manipulating the samples in the current layer as you trigger them. Normal leaves the samples unaffected, while Reverse plays them in reverse. At a value of Reverse, the samples will continue to loop as long as notes are sustained. To play them just once in reverse, you would adjust the length of the layer's amplitude envelope (explained later in this chapter). BiDirect (bidirectional) causes the samples to loop infinitely, alternating between normal and reversed playback. Finally, Noise replaces the samples with a white noise generator.

## The PITCH Page

Press the PITCH soft button, and the PITCH page will appear. These parameters adjust the pitch (playback rate) of the samples after the root has been selected by the keymap.

```

EditProg PITCH                               Layer: 1/1
Coarse: 0ST                                  Src1 : OFF
Fine : 0ct                                    Depth : 0ct
FineHz: 0.00Hz                               Src2 : OFF
KeyTrk: 0ct/key                              DptCtl: MWheel
VelTrk: 0ct                                   MinDpt: 0ct
                                                MaxDpt: 0ct
<more>  ALG  LAYER  KEYMAP  PITCH  >more>
    
```

PARAMETER	RANGE OF VALUES	DEFAULT
COARSE ADJUST	-120 to 60 ST (semitones)	0
FINE ADJUST (cts)	± 100 cents	0
FINE ADJUST (Hz)	± 6 Hz	0
KEY TRACKING	± 2400 cents per key	0
VELOCITY TRACKING	± 7200 cents per key	0
SOURCE 1	Control Source list	OFF
DEPTH	± 7200 cents	0
SOURCE 2	Control Source list	OFF
DEPTH CONTROL	Control Source list	MWheel
MINIMUM DEPTH	± 7200 cents	0
MAXIMUM DEPTH	± 7200 cents	0

These parameters were described in the section on common DSP control parameters earlier in this chapter, so we won't repeat them here, but a word about the Fine Hz parameter is in order.

### Fine Hz

This measures pitch adjustment by the relative frequency (in Hertz) of each note. This is useful for controlling the beat frequency between layers in a multi-layered program. Using this parameter to detune chorusing layers will keep the beat frequency constant across much of the keyboard. Although the ratio of frequencies between each layer remains constant, the detuning will increase at lower pitches, and can become extreme. The K2vx automatically limits the amount of detuning when it becomes extreme, so you'll notice the beat frequencies moving out of sync when you play low pitches.

### F1–F3 Pages

These pages are reached by pressing the F1, F2, and F3 soft buttons, respectively. They contain the parameters governing the three variable DSP functions in each algorithm. The pages vary depending on the DSP functions selected for the three middle DSP control inputs, represented by the downward-pointing arrows on the ALG page. See the *Reference Guide* for a complete list of the algorithms and their available DSP functions.

### The F4 AMP Page

Press the F4 AMP soft button to call up this page, which features five of the six common DSP control parameters, in this case controlling the final amplitude of the current layer before it reaches the audio outputs. There's also a parameter that enables you to pad (attenuate) the current layer's signal before its final amplification.

```

┌──────────┐Prog: F4 AMP(FINAL AMP) <> Layer: 1/1
Adjust: 6dB          Src1 : OFF
                    Depth : 0dB
                    Src2 : OFF
KeyTrk: 0.00dB/key  DptCtl: MWheel
VelTrk: 20dB        MinDpt: 0dB
Pad : 0dB           MaxDpt: 0dB
└──────────┘
<more  F1      F2      F3      F4 AMP  more>
    
```

PARAMETER	RANGE OF VALUES	DEFAULT
ADJUST	−96 to 48 dB	6
KEY TRACKING	± 2 dB	0
VELOCITY TRACKING	± 96 dB	20
PAD	0, 6, 12, 18 dB	0
SOURCE 1	Control Source list	OFF
DEPTH	± 96 dB	0
SOURCE 2	Control Source list	OFF
DEPTH CONTROL	Control Source list	MWheel
MINIMUM DEPTH	± 96 dB	0
MAXIMUM DEPTH	± 96 dB	0

### Adjust

Adjust the overall amplitude (gain) of the currently selected layer. In multi-layer programs, this parameter adjusts the amplitude of the layers relative to each other. This is the final output control for the layer (Post-amp pad). Be careful not to set this too high! If one of your layers is too loud, it's generally better to cut its level than to boost the others. This will keep distortion to a minimum.

### Key Tracking

This uses the MIDI note numbers of the notes you play as a control source affecting the individual amplitudes of each note in the current layer. Positive values increase the amplitude as you play higher-pitched notes. For example, if the key tracking is .20 dB/key, then C#4 will be .20 dB louder than C 4 if triggered with the same attack velocity. If the value for this parameter were negative, C#4 would have less amplitude than C 4. A word of caution here: values above 0.30 dB/key (or below -0.30) can generate extremely high amplitude levels. If you set this parameter that high without lowering the value of the Adjust parameter to -12 dB or lower, your sound may clip, which can be useful, but it isn't necessarily what you want.

### Velocity Tracking

This essentially adjusts the steepness of the curve applied by the VelTouch setting (Master mode). At a value of 0, every note in the current layer would have the same amplitude, regardless of its attack velocity. When the value is positive, note amplitude increases as attack velocity increases. When the value is negative, note amplitude *decreases* as attack velocity increases. Larger values increase the range between minimum and maximum amplitude, so with a large positive value, the amplitude will be low when you play softly. Small values decrease the range between min and max, so with a small positive value, you'll get nearly full amplitude even with light attack velocities.

### Pad

Select one of four attenuation levels for cutting the amplitude of the current layer *before* the final amp stage (Pre-amp pad). Use the pad if the layer's sound distorts when played. Note: clipping can occur in earlier algorithm blocks as well. If this is the case, you'll probably want to try to remove the clipping in the earlier block, if possible.

### Sources 1 and 2, Depth Controls

These are common DSP control parameters, which in this case let you assign control sources to affect the amplitude of the current layer. The functions of common DSP control parameters are explained in their own section earlier in this chapter.

## The OUTPUT Page

This page is reached by pressing the OUTPUT soft button. This is where you route the signal to the eight separate outs, and to the MIX outputs, with or without passing through the effects processor on the way. If you're using the MIX outputs, select Output Group A if you want the layer to appear at the MIX outputs with effects, or Output Group B, C, or D if you want the layer to appear dry at the MIX outputs.

The OUTPUT page gives you broad control over the audio signal. You can adjust the output routings of every layer in every program, enabling you to take maximum advantage of the K2vx's flexible audio output capabilities.

There are actually four different configurations of the OUTPUT page. The one you see depends on whether the current layer uses a stereo keymap, and whether it uses a double-output algorithm. A double-output algorithm is one whose signal path is split into two parts before final amplification.

Regardless of the page's configuration, there are parameters for adjusting the Output Group, the Pan position, the Output Mode, the Gain, the Crossfade control, and the Crossfade sense. Layers that use stereo keymaps, or that use double-output algorithms, have additional sets of Output Group and Pan parameters on their OUTPUT pages.

The following page is for a layer with one keymap and a single-path algorithm.

```

EditProg: OUTPUT <>Layer: 1/1
Pair: A(FX)
Pan : L * R
Mode: +MIDI
Gain: 6dB
CrossFade : OFF XFadeSense: Norm
<more OUTPUT EFFECT COMMON SetRng more>
    
```

PARAMETER	RANGE OF VALUES	DEFAULT
PAIR	A(FX), B(DRY), C(DRY), D(DRY)	A(FX)
PAN	Left to Right (15 positions)	Center
MODE	Fixed, +MIDI, Auto, Reverse	+MIDI
GAIN	-12 to 30 dB (6 dB increments)	6 dB
CROSSFADE CONTROL	Control Source list	OFF
CROSSFADE SENSE	Normal, Reversed	Normal

**Pair**

This parameter defines the Output Group of the current layer—that is, which group of audio outputs the layer uses. If the layer is assigned to Pair A, for example, its audio signal will appear at the Group A outputs. The signal will also appear at the mixed outputs, with effects applied, if there are no cables inserted into the Group A outputs.

**Pan**

Use this parameter to position the current layer’s audio signal between the left and right outputs of whichever Output Group they’re assigned to.

**Mode**

When the mode is Fixed the pan position remains as defined with the Pan parameter, ignoring MIDI pan messages. When the mode is +MIDI, MIDI pan messages (MIDI 10) will shift the sound to the left or right of the Pan parameter setting. Message values below 64 shift it left, while those above 64 shift it right. A setting of Auto assigns the pan setting of each note based on its MIDI note number. In this case, Middle C (MIDI note number 60) is equivalent to the Pan parameter’s setting. Lower notes shift increasingly left, while higher notes shift increasingly right. A setting of Reverse shifts low notes right, and high notes left. MIDI pan messages will also affect the pan position when values of Auto and Reverse are selected.

NOTE: If you’re using the PANNER DSP function in the algorithm for any of the layers in a program, that layer will respond to MIDI pan messages even if the Mode parameter is set to a value of Fixed. That’s just the way it is.

**Gain**

Boost (or cut) the amplitude of the current layer. For layers using double-output algorithms, the gain is divided evenly between the two signal paths. Since this gain is not affected by the layer’s amplitude envelope, you can use it to add a constant amount of gain to a layer.

**Crossfade, Crossfade Sense (XFadeSense)**

The Crossfade parameter lets you select a control source to fade the current layer’s amplitude from zero to maximum. When Crossfade Sense is Normal, the layer is at full amplitude when the Crossfade control is at minimum. With Crossfade Sense set to Reverse, the layer is at zero amplitude when the Crossfade control is at minimum.

This parameter is similar to the Src1 and Depth parameters on the F4 AMP page, but the attenuation curve for the Crossfade parameter is optimized specifically for crossfades.

To crossfade two layers in the same program, assign the same control source for the CrossFade parameters in both layers, then set one of their XFadeSense parameters to a value of Norm, and the other's to Rvrs.

**Other OUTPUT Page Configurations**

The following page is for a layer with one keymap and a double-output algorithm. The U and L stand for the upper and lower wires (signal paths). You have independent control of the output parameters for each wire.

```

EditProg:OUTPUT <>Layer:1/1
  Pair:  Pan:  Mode:  Gain:
U:A(FX) L      *R +MIDI 6dB
L:A(FX) L*    R  +MIDI 6dB
CrossFade :OFF XFadeSense:Norm
<more OUTPUT EFFECT COMMON SetRng more>
    
```

Next are the two page configurations for layers with stereo keymaps: the first one uses a single-output algorithm, and the second uses a double-output algorithm.

With a single-output algorithm, stereo keymap layers let you adjust the pan position of each keymap, but all other parameters are identical for both keymaps.

```

EditProg:OUTPUT <>Layer:1/1
Pair:A(FX)
Pan1:L*      R
Pan2:L      *R
Mode:+MIDI
Gain:6dB
CrossFade :OFF XFadeSense:Norm
<more OUTPUT EFFECT COMMON SetRng more>
    
```

When a stereo keymap layer uses a double-output algorithm, both keymaps are split between the upper and lower wires. In other words, both wires carry the signal from each of the keymaps. The Output Group (Pair), Output mode, and Gain level of Keymap 1 are mimicked by Keymap 2 (that's why these parameters aren't displayed for Keymap 2 on the OUTPUT page). You can, however, set the pan positions independently for the upper and lower wires of both keymaps.

```

EditProg:OUTPUT <>Layer:1/1
  Pair:  Pan:  Mode:  Gain:
U1:A(FX) L*    R  +MIDI 6dB
L1:A(FX) L    *R +MIDI 6dB
U2:      L    *R
L2:      L*   R
CrossFade :OFF XFadeSense:Norm
<more OUTPUT EFFECT COMMON SetRng more>
    
```

## The EFFECT Page

On the EFFECT page you'll adjust the parameters that control the preset effect that the global effects processor applies to the currently selected program. Press the EFFECT soft button and the EFFECT page appears:

```

[30] [Prog] [Effect] All Layers
EffectPreset: 1 Sweet Hall 2

Wet/Dry Mix : Adjust: Source: Depth:
              45%Wet  OFF      0
(Realtime 1): 0      OFF      0
(Realtime 2): 0      OFF      0
<more> [OUTPUT] [EFFECT] [COMMON] [SetRng] [more]>

```

The top line of the page gives you the usual mode reminder, and shows that the preset effect you select will apply to all layers of the program. Keep in mind that these parameters are global—that is, they affect every part of the program's sound that goes through the global effects processor. The display reminds you of this by showing "All Layers" in the top line.

The Effect Preset parameter determines which preset effect will be selected when you select the program in Program mode.

The remaining three parameters—Adjust, Source, and Depth—apply to three inputs that can be used to modify the preset effect in real time, using any of the control sources. The inputs are the Wet/Dry Mix, and two other realtime inputs that vary with the preset effect—chorus delay, reverb time, and other similar multi-effects DSP functions.

The Adjust parameter sets the initial level for the three inputs. This defines the status of the effect when no control source signal is applied to the inputs. For the Wet/Dry Mix, the range of levels is 0 to 100%. For the two variable inputs, the ranges are -128 to 127.

This Source parameter lets you define which control source is used to modify each of the inputs. The values are taken from the Control Source list.

The Depth parameter defines how much you can modify each input with its assigned control source. The range for each input is -128 to 127.

### Wet/Dry Mix

The Wet/Dry Mix parameter determines how much of the preset effect is applied to the program. A value of 0% leaves the program dry; no effect is heard. A value of 100% removes all of the dry signal, and you hear a full effect level on the sound. For best results, start with values of near 50%, and tweak them from there.

### Realtime 1 and 2

These two inputs differ with each preset effect. When you select different preset effects, the names of these two inputs will change to reflect the effect parameter that can be modified via the control source you've assigned with the Source parameter.

When you're recording sequences and including realtime effects changes in the sequence, be sure that the realtime effects messages occur at least 200 milliseconds after program or setup changes. It takes the K2vx approximately 200 milliseconds to switch preset effects, and realtime effects events can interfere with the selection of the new effect. The most likely result is that the realtime effects messages will be ignored.

## The COMMON Page

Here's where you find six frequently-used parameters that affect the entire current program, not just the current layer. The COMMON page is reached by pressing the COMMON soft button in the Program Editor.

```

EditProg COMMON All Layers
Pitch Bend Range: 200ct Globals: Off
MonoPhonic : Off
    
```

```

<more OUTPUT EFFECT COMMON SetRng more>
    
```

Notice that when the Monophonic parameter is set to its default value of Off, the four monophonic parameters do not appear on the page.

PARAMETER	RANGE OF VALUES	DEFAULT
PITCH BEND RANGE	± 7200 cents	200 cents
MONOPHONIC	Off, On	Off
(LEGATO PLAY)	Off, On	Off
(PORTAMENTO)	Off, On	Off
(PORTAMENTO RATE)	1 to 3000 keys per second	70
(ATTACK PORTAMENTO)	Off, On	On
GLOBALS	Off, On	Off

### Pitch Bend Range

Use this parameter to define how much the pitch will change when you move your MIDI controller's Pitch Wheel. Positive values will cause the pitch to bend up when the Pitch Wheel is pushed up, while negative values will cause the inverse. Large positive values can cause samples to bend to their maximum upward pitch shift before the Pitch Wheel is fully up. This will not happen when bending pitch down.

### Monophonic

When off, the program is polyphonic—it can play up to 48 notes at a time. Notice that when the Mono Mode parameter is off, the three Portamento parameters do not appear on the COMMON page. This is because only monophonic programs can use portamento.

When On, the program will play only one note at a time. This makes it possible to use portamento, so the remaining parameters appear.

```

EditProg COMMON All Layers
Pitch Bend Range: 200ct Globals: Off
MonoPhonic : On
Legato Play : On
Portamento : Off
Portamento Rate : 70.0key/s
AttackPortamento: Off
<more OUTPUT EFFECT COMMON SetRng more>
    
```



## Legato Play

When Legato Play is on, a note will play its attack only when all other notes have been released. This is useful for realistic instrumental sounds.

## Portamento

This parameter is either on or off. The default value of Off means that portamento is disabled for the current program.

Portamento is a glide between pitches. On actual acoustic instruments like violin and bass, it's achieved by sliding a finger along a vibrating string. On most keyboards that offer portamento, it's achieved by holding down a key that triggers the starting note, then striking and releasing other keys. The pitch glides toward the most recently triggered note, and remains at that pitch as long as the note remains on. The K2vx gives you two ways to get portamento. See the Attack Portamento parameter below.

When you're applying large amounts of portamento to multi-sampled sounds (Acoustic Guitar, for example), the K2vx will play more than one sample root as the pitch glides from the starting pitch to the ending pitch. This may cause a small click at each sample root transition. You can reduce the number of clicks you'll hear by entering the Program Editor and adjusting the KeyTrk parameter on both the KEYMAP and PITCH pages. The quickest way is to set the KeyTrk value on the KEYMAP page to 0, and to 100 on the PITCH page. This will stretch the sample root that plays at C 4 across the entire keyboard. Now any amount of portamento will play only one sample root, and the clicks will disappear.

There's a tradeoff here, since many sounds will change in timbre as these single sample roots are pitch-shifted during the portamento. This will be most noticeable for acoustic instrument sounds, and may not be noticeable at all for single-cycle waveforms like sawtooth waves. Furthermore, some samples will not glide all the way up to the highest notes—there's a limit to the amount of upward pitch-shifting that can be applied to samples. If this doesn't work for you, you can compromise between the number of clicks and the amount of timbre change by further adjusting the KeyTrk parameters on the KEYMAP and PITCH pages.

As long as the combined values of the KeyTrk parameters on both pages add up to 100, you'll have normal semitone intervals between keys. If you set both parameters to values of 50, for example, the sound will still play normally, and you'll have several sample roots (about half the number of the original sound) stretched evenly across the keyboard, instead of just one. This will give you fewer clicks than in the original sound, but not as much change in timbre as setting the KEYMAP KeyTrk value all the way to 0. Set the KEYMAP KeyTrk parameter higher to reduce the change in timbre, or set the PITCH KeyTrk value higher to reduce the number of clicks. Just make sure the combined values add up to 100, to preserve the normal intervals between notes.

## Portamento Rate

The setting for Portamento rate determines how fast the current note glides from starting pitch to ending pitch. The value of this parameter tells you how many seconds the note takes to glide one semitone toward the ending pitch. At a setting of 12 keys/second, for example, the pitch would glide an octave every second. The list of values is non-linear; that is, the increments get larger as you scroll to higher values.

## Attack Portamento

This parameter toggles between two types of portamento. When set to On, the K2vx remembers the starting pitch so you don't have to hold a note on to achieve portamento. The pitch always glides to each new note from the previously triggered note. When set to Off, the pitch will glide to the most recently triggered note only when the previous note is still on (in other words, you must use legato fingering).

### Globals

This is another toggle, which affects LFO2, ASR2, and FUNs 2 and 4. When off, these four control sources are local; they affect each individual note in the layers that use them as a control source. They begin operating each time a note in that layer is triggered.

When the Globals parameter is set to On, these control sources become global, that is they affect every note in every layer of the current program, not just the one to which they're applied. When these control sources are global, they begin operating as soon as the program is selected. When Globals are on, LFO2, ASR2, and FUNs 2 and 4 will appear on their respective pages preceded by the letter "G" to indicate that they're global.

You'll use global control sources when you want to affect each note in a given layer uniformly, and local control sources when you want to affect each layer's note independently. For example, you'd use a global LFO controlling pitch to create a *Leslie effect* on an organ sound, since you want the affect applied to all the notes you play. You'd use a local LFO controlling pitch to create a vibrato for a solo violin, since you want to be able to vary the rate and depth of the vibrato for each note.

### The Amplitude Envelope (AMPENV) Page

Amplitude envelopes have three sections: attack, decay, and release. The attack section determines how long each note takes to reach its assigned amplitude level after you trigger a Note On event. The decay section determines how quickly and how much a sustained sound fades before a Note Off is triggered. The release section determines how quickly a sound fades to silence *after* a Note Off is triggered.

Press the AMPENV soft button to reach the Amplitude Envelope page. For many programs, it will look like the diagram below, which tells you that the amplitude for the current layer is the default, "natural" ROM amplitude envelope that's applied to each sample and waveform during its original development process. You'll leave the amplitude envelope in Natural mode when you don't want to change the way the current layer's loudness develops.



If you want to build your own amplitude envelope, just turn the Alpha Wheel a click. The word "Natural" will change to "User," and a set of AMPENV parameters will appear. The sound will change when you do this, because the default settings for the User envelope, as shown in the diagram below, take effect as soon as you leave Natural mode. Returning to Natural mode applies the original amplitude envelope once again.

```

EditProg:AMPENV [1/1] (<>)Layer:1/1
Att1:Att2:Att3:Dec1:Rel1:Rel2:Rel3:Loop:
0s 0s 0s 0s 0s 0s 0s Off
100% 0% 0% 100% 0% 0% User Inf
┌───┐
│   │
│   │
│   │
│   │
│   │
│   │
│   │
│   │
│   │
│   │
└───┘
<more> AMPENV ENV2 ENV3 ENVCTL >more>
    
```

You'll tweak the parameters on the AMPENV page when you want to shape the amplitude characteristics of your sounds. A graphic view of the amplitude envelope will appear on the display to give you a visual sense of the envelope's characteristics. The dots along the envelope graphic indicate the breakpoints between the envelope's various segments. The small horizontal arrow represents the end of the decay section. The small downward-pointing arrow represents the beginning of the release section.

Because the K2vx's ROM samples are stored in a compressed format, applying an altered amplitude envelope can change more than just the amplitude of your sound, since it also changes the rate at which the samples are decompressed for playback. When the samples are made to play back with altered envelopes, the timbres can evolve in new and interesting ways.

The AMPENV page's top line gives you the usual location reminder, points out the currently selected layer, and tells you the relative scale of the envelope's graphic view. The envelope graphic shrinks in scale as the segment times get longer. This auto-zoom feature maximizes the available display space. Try lengthening one of the segment times. The envelope graphic will stretch to fill the display from left to right. When it fills the display, it will shrink to half its size, and the top line will indicate that the scale has changed (from [1/1] to [1/2], for example).

Each parameter on this page has two values, as listed below. For the envelope segments, the first (upper) value is the duration of the segment, and the second is the amplitude level at the completion of the segment. For the Loop parameter, the values define how the envelope loops, and how many times the loop cycles.

PARAMETER		RANGE OF VALUES
ATTACK SEGMENT 1, 2, 3	TIME	0 to 60 seconds
	LEVEL	0 to 100 %
DECAY SEGMENT	TIME	0 to 60 seconds
	LEVEL	0 to 100 %
RELEASE SEGMENT 1, 2, 3	TIME	0 to 60 seconds
	LEVEL	0 to 100 %
LOOP	TYPE	Off, Forward, Bidirectional
	# OF LOOPS	Infinite, 1 to 31 times

### Attack Segment Times

These indicate how long it takes for the current layer's amplitude to reach its final level from its starting level.

### Attack Segment Levels

These are the final levels that each segment achieves at completion. The levels are expressed as percentages of the maximum possible amplitude for the current layer. Attack segment 1 always

starts at zero amplitude, and moves to its assigned level in the time specified by its time value. So the default settings of 0 seconds and 100% mean that the first segment of the attack section moves instantly from zero amplitude to 100% amplitude. Increase the time of Attack segment 1 if you want the sound to ramp up more slowly.

Attack segments 2 and 3 affect the sound only when you set a non-zero value for time. They will then move to their assigned levels in the time specified. Their starting levels are equal to the final levels of the preceding segment.

### Decay Segment

The decay section has only one segment. It has values for time and level, just as for the attack section. The decay section begins as soon as the attack section has been completed. It starts at the same amplitude level as the attack segment preceding it, and moves to its assigned level in the time specified. You'll hear a note's decay section only when the attack section is completed before a Note Off message is generated for that note.

To create a sustaining envelope, simply set the Decay segment to a non-zero value.

### Release Segments

Like the attack and decay sections, each of the three segments in the release section has values for time and level. Each segment reaches its assigned level in the time specified for that segment. Release segment 1 starts at the Note Off event for each note, at the current amplitude level of that note—whether it's in the attack section or the decay section. It then moves to its assigned level in the time specified. Release segments 2 and 3 start at the final levels of the segments before them. Release segments 1 and 2 can be set to any level from 0 to 100%. Release segment 3 always has a level of 0%, so you can't adjust its level. In place of its Level parameter you see a parameter that lets you toggle between User envelopes and the sound's preprogrammed "natural" envelope.

### Loop Type

There are seven different values for Loop type.

A value of Off disables looping for the current layer's amplitude envelope.

Values of seg1F, seg2F, and seg3F are forward loops. In each case, the amplitude envelope plays through the attack and decay sections, then loops back to the beginning of the first, second, or third attack segments, respectively.

Values of seg1B, seg2B, and seg3B, are bidirectional loops. The amplitude envelope plays through the attack and decay sections, then reverses and plays backward to the beginning of the first, second, or third attack segment, respectively. When it reaches the beginning of the assigned attack segment, it reverses again, playing forward to the end of the decay section, and so on.

### Number of Loops

A value of Inf makes the amplitude envelope loop until a Note Off is generated. Values of 1 through 31 indicate how many times the loop will repeat after the amplitude envelope has played once through its normal cycle.

Regardless of the loop type and the number of loops, each note goes into its release section as soon as its *Note State* goes off (that is, when a Note Off is generated). The envelope will continue to loop as long as Note State remains on, whether it's held on by a pedal, by the IgnRel parameter (described in the section entitled "The LAYER Page"), or whatever.

## Envelopes 2 and 3

The K2vx offers two envelopes in addition to the amplitude envelope. Like the amplitude envelope, Envelopes 2 and 3 can be assigned like any other control source. The only difference between these two envelopes and the amplitude envelope is that Envelopes 2 and 3 can be bipolar. This means that you can set negative values for them. (Obviously, you can't have an amplitude less than zero, so the amplitude envelope is unipolar—the values range from 0 to 100%). A bipolar envelope controlling pitch, for example, could modulate the pitch both above and below its original level.

```

editProg:ENV2 [1/1] <>Layer:1/1
Att1:Att2:Att3:Dec1:Rel1:Rel2:Rel3:Loop:
2.00 2.00 2.00 0s 0s 0s 0s Off
100% -100 100% 100% 0% 0% 0% Inf

```

```

<more AMPENV ENV2 ENV3 ENVCTL more>

```

The pages for Envelopes 2 and 3 are reached with the soft buttons ENV2 and ENV3. When you select these pages, you'll find a display that looks very much like the AMPENV page. The only differences are that you can program an amount for Rel3, and in the envelope graphic, which has a dotted line running horizontally across the display. This is the zero level line; negative level values for the various envelope segments will cause the envelope graphic to dip below this line.

## The Envelope Control (ENVCTL) Page

Envelopes are control sources with outputs that evolve over time without repeating (unless you want them to). You can make the envelopes even more powerful by using envelope control. This gives you realtime control over the rates of each section of the envelopes. Press the ENVCTL soft button to reach the ENVCTL page.

```

editProg:ENVCTL <>Layer:1/1

```

Att:	Adjust:	KeyTrk:	VelTrk:	Source:	Depth:
1.000x	1.000x	1.000x	1.000x	MIDI73	1.000x
Dec:	1.000x	1.000x		MIDI72	1.000x
Rel:	1.000x	1.000x		MIDI72	1.000x
Imp:	0.0dB	0.0dB	0.0dB	OFF	0.0dB

```

<more AMPENV ENV2 ENV3 ENVCTL more>

```

The display's top line reminds you of the current layer. The first line of text in the center of the display shows five of the common DSP control parameters: Adjust, Key tracking, Velocity tracking, and Source/Depth.

This page is a table showing the five envelope control parameters, and their values for each of the three sections of the envelopes. Additionally, the line above the soft buttons lets you make use of the Impact feature, which adds an amplitude overshoot to the first 20 milliseconds of a note's attack. It's important to keep in mind that if you set up an envelope control source, it affects Envelopes 2 and 3, as well as the amplitude envelope (Natural or User). Furthermore, the values for the various parameters are cumulative. With the exception of Impact, though, ENVCTL does not affect the attack sections of natural envelopes.

The parameters and values in the following list apply to *each* of the three envelope sections—attack, decay, and release. We'll describe them only once, since their functions are largely the same for each envelope section. The only difference is with velocity tracking, which is hard-wired to control only the attack sections of the envelopes (you can assign attack velocity as the value for the Source parameter in each of the sections, however).

The values of each of these parameters multiply the *rates* of the envelope sections they control. Values greater than 1.000x make the envelope sections run *faster* (they *increase* the rate), while values less than 1.000x make the envelope sections run *slower*. Say for example that on the current layer's AMPENV page you had set the Decay section's time at 2.00 seconds, and its level at 0%. This sets the layer's amplitude to fade to silence two seconds after the completion of the last attack segment. The decay *time* is two seconds; the decay *rate* is 50% per second. Now if you select the ENVCTL page and set the Decay Adjust parameter to a value of 2.000x, you've increased the decay *rate* by a factor of two. The rate increases to 100% per second, and the decay time is now one second instead of two.

PARAMETER	RANGE OF VALUES (Att, Dec, Rel)
ADJUST	0.018 to 50.000x
KEY TRACKING	0.018 to 50.000x
VELOCITY TRACKING	0.018 to 50.000x
SOURCE	Control Source list
DEPTH	0.018 to 50.000x

### Adjust

This is the familiar Coarse adjust found on many other pages. Use it here to change the rate of one of the envelope sections without reprogramming the envelope itself. This parameter doesn't give you realtime control over the envelope. It is, however, a good way to adjust the natural envelopes without switching to a User envelope and trying to approximate the Natural envelope.

### Key Tracking

This uses the MIDI note number of each key as the control input for the current layer's corresponding envelope section. When the value of this parameter is greater than 1.000x, notes above C 4 will make the envelope section run faster, while notes below C 4 will make it run slower. When the value of this parameter is less than 1.000x, notes above C 4 will make the envelope section run slower, and notes below C 4 will make it run faster. This gives you realtime envelope control right from your MIDI controller. You might use it, for example, to cause an acoustic guitar sound to decay quicker at the high end (set the key tracking to a positive value).

### Velocity Tracking

Use your attack velocity as the control input for the current layer's attack section (this parameter doesn't apply to decay or release). When the value of this parameter is greater than 1.000x, attack velocities greater than 64 make the attack section run faster, and attack velocities below 64 make it run slower. This gives you realtime attack control over the envelope.

### Source, Depth

These two parameters work together to let you assign a control like the Mod Wheel to affect the current layer's envelopes in realtime. The value of the Source parameter defines which control affects the envelope section, and the value of the Depth parameter defines how much the rate is multiplied when the control is at its maximum.

### Impact

Impact punches the volume during the first 20 milliseconds of the attack of an envelope. Use this feature to get maximum "thump" from your bass and drum sounds.

Programs you create that use Impact will not work on a K2vx.

## The LFO Page

These are low-frequency oscillators. You'll use the LFO page to define the behavior of the two LFOs available to each layer. LFOs are periodic (repeating) control sources. The basic elements are the rate and shape, which define how frequently the LFO repeats, and the waveform of the modulation signal it generates.

With the K2vx, you can set upper and lower limits on each LFO's rate, and assign a control source to change the LFO's rate in realtime, if you wish.

Because of its periodic nature, the LFO is perfect for creating effects like vibrato (cyclic variation in pitch) and tremolo (cyclic variation in amplitude). When you're editing LFOs, or any control source, remember that it must be assigned to control some parameter before you'll hear the effects of your edits.

LFO1 is always local, meaning that it's triggered with each Note On event, and runs independently for each note in the layer. LFO2 is local by default, but can be made global. This is done on the COMMON page, by setting the Globals parameter to On, which causes LFO2, ASR2, FUN2 and FUN4 all to become global. Global controls uniformly affect every note in each layer.

```

EditProg: LFO <> Layer: 1/1
LFO1: MnRate: MxRate: RateCt: Shape: Phase:
      2.00H  0.00H  OFF   Sine  0deg
LFO2:  OFF   0.00H  OFF   Sine  0deg

<more LFO ASR FUN UTRIG more>
    
```

The top line of this page gives the usual mode reminder and tells you which layer you're looking at. There are five parameters for each of the LFOs.

PARAMETER	RANGE OF VALUES	DEFAULT
MINIMUM RATE	0 to 24 Hz	2.00 (OFF for LFO2)
MAXIMUM RATE	0 to 24 Hz	0.00
RATE CONTROL	Control Source list	OFF
LFO SHAPE	LFO Shape list (Ref. Guide)	Sine
LFO START PHASE	0, 90, 180, 270 degrees	0

### Minimum Rate

This is the slowest rate at which the LFO will run. When its Rate control is set to OFF, or when the control source assigned to it is at its minimum, the LFO runs at its minimum rate.

### Maximum Rate

This is the fastest possible rate for the LFO. When its Rate control is set to ON, or when the control source assigned to it is at its maximum, the LFO runs at its maximum rate.

### Rate Control

Assign any control source in the list to modulate the LFO's rate between its minimum and maximum. A continuous control like the Mod Wheel is a natural choice, enabling you to get just about any rate between min and max. But you can use a switch control too, to get just the min or max with nothing in between. Assigning MPress as the rate control for an LFO vibrato gives you an easy way to increase the vibrato rate in realtime, as you can on many acoustic instruments.

### LFO Shape

The shape of the LFO waveform determines the nature of its effect on the signal its modulating. There are diagrams of each LFO shape in the *Reference Guide*; these will give you an idea of how each LFO shape affects the signal. An easy way to check the effects of the different LFO shapes is to set LFO1 as the value for the Src1 parameter on the PITCH page, and set the Depth for Src1 to 400 cents or so. Then go to the LFO page, set the Min and Max rates for LFO1 at 0.00 Hz and 4.00 Hz or so, and set the Rate control to MWheel. Now play your MIDI controller and you'll hear the LFO's rate change when you move its Mod Wheel. Select different LFO Shapes and check out the effect on the pitch.

### LFO Phase

Use this parameter to determine the starting point of the LFO's cycle. One complete cycle of the LFO is 360 degrees. 0 degrees phase corresponds to a control signal value of 0, becoming positive. Each 90-degree increment in the phase represents a quarter-cycle of the LFO.

When an LFO is local, the phase parameter gives you control over the starting point of the LFO for each note (for example, you could make sure every vibrato started below the pitch you played instead of at the pitch you played). The LFO's phase also affects global LFOs, although it's often indistinguishable, since global LFOs start running as soon as the program containing them is selected, even if you don't play any notes.

## The ASR Page

ASRs are three-section unipolar envelopes—attack, sustain, and release. The K2vx's ASRs can be triggered by a programmable control source, and can be delayed. ASR1 is always a local control. ASR2 is local by default, but becomes global if the Globals parameter on the COMMON page is set to On.

ASRs are frequently used to ramp the depth of pitch or amplitude in a vibrato or tremolo, enabling delays in those effects. Chapter 19 gives an example of creating a delayed vibrato.

The ASR page consists of two rows of five parameters, one row for each of the ASRs.

```

EditProg:ASR                               <>Layer:1/1
ASR1:  Trig:  Mode:  Delay:  Attack:Releas:
      ON      Hold  1.00s  1.00s  1.00s
ASR2:  UN      Rept  1.00s  1.00s  1.00s

<more  LFO  ASR  FUN  UTRIG  more>
    
```

PARAMETER	RANGE OF VALUES
TRIGGER MODE	Control Source list Normal, Hold, Repeat



**DELAY**                    **0 to 60 seconds**  
**ATTACK**                   **0 to 60 seconds**  
**RELEASE**                   **0 to 60 seconds**

**Trigger**

This defines the control source that starts the current layer’s ASRs. The ASR starts when the trigger switches from off to on. If the Trigger parameter is set to ON, global ASRs will run as soon as the program containing it is selected. Switch controls are better suited for ASR triggers because of their binary (on/off) nature. A continuous control will trigger the ASRs when its signal value is above its midpoint.

**Mode**

This parameter sets the sustain section of the ASR. The ASR’s mode determines what the ASR does when it finishes its attack section.

If the Mode parameter is set to Normal, the ASR will run directly from its attack section to its release section (no sustain). At a setting of Repeat, the ASR will cycle through the attack and release sections, then loop forward and cycle through again until the ASR’s trigger switches off. If the mode is set to Hold, the ASR maintains its position at the end of the attack section until the ASR’s trigger switches off. The ASR then goes into its release section.

If the ASR’s trigger switches off before the attack section is complete, the ASR goes directly to its release section.

**Delay**

When the ASR’s trigger switches on, the ASR will start immediately if this parameter is set to zero. Non-zero values will cause a corresponding delay between the ASR trigger and the start of the ASR.

**Attack**

This defines how long the ASR takes to ramp up from minimum to maximum effect on whatever it’s patched to.

**Release**

This defines how long the ASR takes to fade to minimum from its maximum. If the ASR’s trigger switches off before the ASR has reached maximum, the ASR releases from that level.

**The FUN Page**

“FUN” is short for “function.” The K2vx’s four FUNs greatly extend the flexibility of the control sources. Each FUN accepts input from any two control sources, performs a selectable function on the two input signals, and sends the result as its output, which can be assigned like any other control source. Using the FUNs involves defining them on the FUN page, then assigning one or more of them as control sources. The FUN page looks like this:

```

edit:PROG:FUN <>Layer:1/1
FUN1:  OFF      OFF      a+b
FUN2:  OFF      OFF      a-b
FUN3:  OFF      OFF      (a+b)/2
FUN4:  OFF      OFF      a/2+b
<more LFO ASR FUN UTRIG more>
    
```

There are three parameters for each FUN. Inputs **a** and **b** can be any control source from the Control Source list. The control sources you want to combine are the ones you'll assign as the values for these parameters.

The Function parameter determines what mathematical function is applied to the two inputs. When a FUN has been assigned as a control source, the K2vx reads the values of the two control sources defined as Inputs **a** and **b**. It then processes them according to the setting for the Function parameter, and the resulting value is the FUN's output.

Chapter 16 describes each of these functions, and provides a few diagrams to give you a hint of the immense control (as well as some chaos) that these functions make possible.

## The VTRIG Page

The velocity triggers base their operation on the attack velocity of each note you play. To use a VTRIG, you simply set its velocity level (threshold), then set it to switch on or off when your attack velocities exceed that threshold. Then assign it as a control source for some other parameter. They're handy for triggering ASRs, for example.

```

EditProg:VTRIG <>Layer:1/1
VTrig1 Level:ppp
VTrig1 Sense:Norm
VTrig2 Level:ppp
VTrig2 Sense:Norm
<more LFO ASR FUN VTRIG more>
    
```

PARAMETER	RANGE OF VALUES	DEFAULT
VEL. TRIGGER LEVEL	ppp to fff	ppp
VEL. TRIGGER SENSE	Normal, Reversed	Normal

The velocity trigger's level is expressed in terms of the standard dynamic markings of western music—ppp, pp, p, mp, mf, f, ff, and fff. The K2vx converts each attack velocity value it receives into one of these eight levels. When a velocity trigger has been assigned as a control source, the K2vx compares the velocity trigger's level and sense with the attack velocity values it receives. If the sense is Normal and the attack velocity value is greater than the velocity trigger's level, the trigger switches on. When the velocity trigger's sense is reversed, the trigger switches on when the attack velocities it receives are lower than the velocity trigger's level. Keep in mind that you won't hear the effect of editing the VTRIG page until you've assigned a VTRIG as a control source for some other parameter.

## The Function Soft Buttons

The remainder of this chapter describes the soft buttons that perform specific functions, as opposed to selecting programming pages. The descriptions below are arranged in the order in which you would see the soft buttons if you pressed the more> button repeatedly. You can always get to these buttons, regardless of which page is currently selected.

### Set Range (SetRng)

The Set Range (SetRng) soft button gives you a quick way to set the lowest and highest notes in the currently selected layer. Press this button, and the K2vx will prompt you to trigger the note you want to set as the low note for the layer. Press the Cancel soft button if you change your

mind. Otherwise, trigger the desired note on your MIDI controller. When you trigger a note, the K2vx prompts you to trigger the note you want to be the highest in the layer. When you trigger another note, the previously selected page returns, and the notes you triggered will be recorded as the new values for the LoKey and HiKey parameters on the LAYER page. You'll notice that the higher of the two notes you triggered is entered as the HiKey value, regardless of the order in which you triggered the two notes.

### **Name**

Call up the page that enables you to change the name of the current program.

### **Save**

Start the process of saving the current program.

### **Delete**

Delete the current program from RAM. You can also delete any other RAM program by scrolling through the list that appears when you press the Delete soft button, then pressing Delete again when the desired program is selected. If you attempt to delete a ROM program, the K2vx will say it's deleting the program, but it doesn't actually do it.

### **Dump**

Send a MIDI System Exclusive dump of the current program's settings. See the *Reference Guide* for more information about System Exclusive messages.

### **New Layer (NewLyr)**

Create a new layer, numbered one above the highest existing layer. The new layer's parameters are those of the single layer in Program 199, called "Default Program." When you press this button, the K2vx will tell you that it is creating a new layer, then will return to the page you were on. The new layer becomes the current layer. If the current program already has its maximum number of layers, the K2vx will tell you that you can't add any more. If the current keyboard channel is a drum channel, you can add up to 31 additional layers, for a total of 32. Otherwise, the maximum is a total of three.

Program 199 makes a good template for programs that you build from the algorithm up. You might want to edit Program 199 to adjust one or more parameters to values you want to use in your template program. If you like the settings of the default layer as they are, however, remember not to make any permanent changes to Program 199.

### **Duplicate Layer (DupLyr)**

Create a copy of the current layer, duplicating the settings of all its parameters. The copy becomes the current layer, and is numbered one higher than the original.

### **Import Layer (ImpLyr)**

Copy a specific layer from another program into the current program. This button brings up a dialog that prompts you to select a layer number and a program number. The dialog tells you the currently selected layer, and the total number of layers in the program. Use one of the two leftmost soft buttons (or the up/down cursor buttons) to change the layer number. If the current program has only one layer, pressing these buttons will have no effect. Use one of the two center soft buttons (or the left/right cursor buttons) to change the program number.

When you have selected the desired layer from the desired program, press the **Import** soft button, and the selected layer will be copied from the selected program, becoming the current layer. Importing layers is a convenient alternative to creating layers from scratch. If you have a favorite string sound, for example, and you want to use it in other programs, just import its

layer(s) into the program you're building. This will preserve the envelopes and all the control settings so you don't have to reprogram them.

### **Delete Layer (DelLyr)**

Delete the current layer. When you press this button, the K2vx asks you if you want to delete the layer; press the Yes soft button to start the deletion process, or the No soft button to cancel it. This prompt prevents you from accidentally deleting a layer.

## Chapter 7

# Setup Mode and the Setup Editor

### Setup Mode

Even if your MIDI controller can transmit on only one MIDI channel at a time, you can use the K2vx's setups by setting the Local Keyboard Channel parameter on the MIDI mode RECV page to match the channel you're using to transmit from your MIDI controller. This will enable you to play the setup's three zones, and send the MIDI information from your MIDI controller to the K2vx's MIDI Out port, on the channels used by the current setup. See the discussion of the Local Keyboard Parameter in Chapter 10.

Selecting performance setups in Setup mode is much like selecting programs in Program mode—just use one of the normal data entry methods to scroll through the list of setups. Standard program changes will select the correspondingly numbered setups while the K2vx is in Setup mode (if the Local Keyboard Channel matches the transmit channel of your MIDI controller). Setups enable you to use three keyboard (MIDI controller) zones, each of which can have its own program, MIDI channel, and control assignments. You can store 100 setups in each memory bank.

```

SetupMode      XPose:051
Chan/Program Info
1   9 Cool Traps
2  18 Sly Acoust
3  22 IZIT_JIMMY
Octav- Octav+ Panic View
21 Friday Gig
22 BoP Rock Reggae
1 Jazz Trio
2 All Percussion
3 Heavy Metal
4 To Sequencer

```

Enter Setup mode by pressing the Setup mode button. You'll see a list of setups that you can scroll through with any data entry method. Press the "View" soft button to see the currently selected setup in large-type display. Press it again to return to the normal view. You can transpose the entire setup up or down with the two Octav soft buttons (as long as the Local Keyboard Channel matches the transmission channel of your MIDI controller). Press them simultaneously to set the transposition back to zero. When you transpose a setup, the split points between zones remain in place; each program is transposed within its respective zone.

The box at the left side of the display shows you the programs assigned to the three zones in the currently selected setup, and the MIDI channels on which they're being transmitted. The lines beneath the program names represent the approximate note range of each zone, and let you know whether the three zones overlap.

When you select a setup in Setup mode, the K2vx sends Program Change commands to its MIDI Out port, on each of the MIDI channels used by the setup. The values of these Program Change commands depend on the setting you have for the PChgType parameters in MIDI mode. See "Program Change Formats" in Chapter 10.

## The Setup Editor

Press the EDIT button to enter the Setup Editor, and you can make changes to the currently selected setup. The Setup Editor page looks like this:

```

EditSetup                               <>zone:1
Program: 1 Grand Piano                   ModWhl: MWheel
Channel: 1                               Mode: Both      FtSw1: Sustai
XPose: 0ST                               PBend: On       FtSw2: SostPd
LoKey: C 0                               PChng: On      CPedal: Foot
HiKey: G 9                               FXMix: 48%Wet  Slider: Data
Effect: 1 Sweet Hall 2                  Press: MPress
Name   Save   Delete  Dump   SetRng  U:127

```

The upper line gives you the usual mode reminder, and shows you which zone is currently selected. Use the CHAN/BANK buttons to select different zones. Each of the three zones in a setup has its own page identical to the one shown above. Zones can overlap if you want, but they don't have to. If you have other MIDI devices connected to the K2vx's MIDI Out port, you should always use different MIDI channels for each zone. You can get to the Program Editor from the Setup editor by pressing EDIT when the Program parameter is selected. This gives you access to the other editors nested within the Program Editor, as well.

### Setup Editor Parameters

The parameters on the Setup Editor page define what each of a setup's zones sends to the MIDI Out port. They also determine how the K2vx responds to MIDI signals received from a MIDI controller connected to the K2vx's MIDI In port (when the Local Keyboard Channel matches the transmit channel of your MIDI controller). They're in effect *only* when you're in Setup mode, however, so although they override the corresponding parameters on the MIDI mode XMIT page, they don't actually change those settings. The MIDI mode settings resume control when you exit Setup mode.

Note: Only the Program, Effect, and Effects Mix parameters will function as programmed if the Local Keyboard Channel does not match the transmit channel of your MIDI controller. When they are set to the same channel, the remaining parameters will take effect as well.

PARAMETER	RANGE OF VALUES
PROGRAM	Program list
CHANNEL	Off, 1 to 16
TRANPOSE	± 60 semitones
LOW KEY	C -1 to G 9
HIGH KEY	C -1 to G 9
EFFECT	Preset Effects list
EFFECTS MIX	0 to 100% wet
MODE	Off, Both, MIDI, Local
PITCH BEND	Off, On
PROGRAM CHANGE	Off, On
MOD WHEEL	MIDI Control Source list
FOOT SWITCH 1	MIDI Control Source list
FOOT SWITCH 2	MIDI Control Source list
CONTROL PEDAL	MIDI Control Source list
CONTROL SLIDER	MIDI Control Source list
MONO PRESSURE	MIDI Control Source list
V	Off, 1-127

## Program

The Program parameter selects the program for the currently displayed zone. This determines the program change number that will be sent on that zone's MIDI channel when the setup is selected (if the Program Change parameter is set to On). See "Program Change Formats" in Chapter 10 for a full explanation of the K2vx's program change methods.

## Channel

The Channel parameter defines the MIDI transmit channel for the currently selected zone. It can be set to a value of Off, to disable the zone, or it can be set to any of the 16 MIDI channels. The most important thing to remember about setups is that each zone must be on a separate channel. This is because you can only have one program on a channel at a time. The only exception to this would be if you wanted the same program in more than one zone. For example, you could take a program and assign it to two zones, then transpose one zone up an octave, allowing you to play two octaves of the same sound. In general, though, you'll be using different programs for each zone.

## Transpose (Xpose)

This defines the amount of transposition sent by the currently selected zone. This will affect both the K2vx and MIDI devices receiving from the K2vx.

## Low Key (LoKey), High Key (HiKey)

The LoKey and HiKey parameters define the note range of the currently selected zone. The easiest way to change these values is to press the SetRng soft button. You'll be prompted to trigger the notes (low note first) you want to be the lowest and highest notes for the zone. When you do, you'll return to the Setup Editor page, and the notes you triggered will be reflected in the values for LoKey and HiKey. They'll also be represented by the lines beneath the program names in the box at the left of the Setup mode page. You can set these values with normal data entry methods as well.

## Effect

This tells the global effects processor which effect to use for the current setup. The values available for this parameter are the 47 factory preset effects, and any that you program yourself. It's not possible to use more than one preset effect per setup (the effects processor can apply only one preset effect at a time, even though that preset can combine up to four digital effects). Changing the effect for one zone will change it for all zones. It also may change the value of the FX Mode parameter in Effects Mode to Auto (if it isn't set to Auto already). This enables the effect to be selected automatically when a program or setup is selected, regardless of the operating mode at the time.

## Effects Mix (FXMix)

The FXMix parameter lets you set the level of effect from 0% (no effect) to 100% (maximum). This also affects all three zones. Remember that the programs in the setup must have their audio output groups set to A (either on the OUTPUT page in the Program Editor or on the CHANLS page in MIDI mode), and you must use the MIX audio outputs if you want effects applied to the programs.

## Mode

This determines whether the currently selected zone affects only the K2vx (Local), gets sent only via MIDI (MIDI), controls both the K2vx and connected MIDI instruments (Both), or is disabled (Off). If the values you set for the other control parameters on this page don't seem to be doing what you expect, check the Mode parameter to make sure the information is being sent where you think it is.

#### **Pitch Bend (PBend)**

The Pitch Bend parameter determines whether the currently selected zone of the currently selected setup will receive the Pitch Bend message when your MIDI controller's Pitch Wheel is moved. This lets you use Pitch bend with some zones and not from others. Choose On to enable the Pitch Wheel message in the current zone, or Off to disable it. This parameter is relevant only if the Local Keyboard Channel parameter on the RECV page in MIDI mode is set to match the MIDI channel you're using to transmit from your MIDI controller.

#### **Program Change (PChng)**

This enables or disables program change commands sent to the MIDI Out port when you select setups from the K2vx's front panel or from your MIDI controller when the K2vx is in Quick Access mode. If it's set to On, then the program numbers for the programs in the three zones will be sent via MIDI when a setup is selected. When this parameter is set to Off, you can select a setup to play on the K2vx, without changing the programs on MIDI devices receiving from the K2vx.

#### **Volume (V)**

This parameter enables you to control the initial MIDI volume setting for each zone of the current setup. When you call up a setup in Setup mode, the K2vx sends a MIDI volume control signal on each of the setup's MIDI channels. This sets the starting volume level for each zone. Subsequent MIDI volume control signals sent to the setup's MIDI channels will affect the volume normally. The Setting of the "V" parameter on the MIDI mode XMIT page (see page 10-4) determines whether or not this parameter will have any effect.

#### **The Remaining Parameters**

A long list of values (the Control Source list) is available for each of the remaining six parameters, which determine two things: how the K2vx responds to Mod Wheel, switch pedal, continuous control pedal, data slider and mono pressure messages from your MIDI controller, and how the K2vx remaps these messages to send to its own MIDI Out port. (See the discussion of the Local Keyboard Channel parameter in Chapter 10.) Each zone's assignments are independent; you can assign a given control in one zone and not the others. While you remain in Setup mode, the values you set for these parameters will override the configuration of the XMIT page in MIDI mode.

#### **The Setup Editor's Soft Buttons**

There are three soft buttons—Name, Save, and Delete—for basic library functions, and one for dumping the whole setup as a MIDI System Exclusive message. The Set Range (SetRng) soft button gives you a quick way to set the values for the LoKey and HiKey parameters for the currently selected zone. Press this button, and the K2vx will prompt you to trigger the low note. Press the Cancel soft button if you change your mind. Otherwise, trigger the note you want to be the lowest in the current zone. When you trigger a note, the K2vx prompts you to trigger the note to be the highest in the zone. When you trigger another note, the Setup Editor page returns, and the notes you triggered will be recorded as the new values for the LoKey and HiKey parameters. You'll notice that the higher of the two notes you triggered is entered as the HiKey value, regardless of the order in which you triggered the two notes.

#### **It's Not Just for Editing Anymore**

You might think that the Setup Editor is intended strictly for programming setups between gigs or sessions. But it's convenient in performance situations, as well. All of the K2vx's performance features function normally while the Setup Editor is active. You can make quick changes to the current setup—choose a different lead sound, for example, or switch one zone to a different MIDI channel to control one of the rack-mount synths at the studio. You can save your changes if you want, or you can exit the Setup Editor without saving to restore your original setups.



## Chapter 8

# Quick Access Mode and the Quick Access Editor

In Quick Access mode, you can select programs or setups with a single press of an alphanumeric button (or with the other data entry methods). For example, in the illustration below, you would simply press "5" on the alphanumeric pad to choose "FM Harmonica". Notice that your selection becomes highlighted in the list, as well as appearing on the line just above the soft button labels.

```
QuickAccessMode  <>Bank:1 For Show 2
Stage Piano      Marimba          Voice+String9
Honky-Tonk      FM Harmonica      ElvisTheDog9
Memorymoos 4    Jazz Trio 2      Fretless Bas
Velveteen

Xpose:0ST      FM Harmonica      Chan:1
Octav- Octav+  Panic View      Chan-  Chan+
```

Using Quick Access mode involves selecting Quick Access banks from the list of factory preset or user-programmed banks. This can be done by using the bank selection shortcut: press the +/ - or CLEAR button on the alphanumeric pad, and you'll be prompted to enter a bank number. Type the desired number on the alphanumeric pad, then press ENTER. The bank will be selected, and you'll return to the Quick Access page. Or use the CHAN/BANK buttons to scroll through the QA banks.

Each bank contains ten memory slots, or entries, where you can store programs or setups in any combination. Any program or setup in the currently selected bank can be selected with the numeric buttons 0 through 9.

The ROM (factory preset) QA banks are organized into useful groupings of sounds that we think you'll find convenient.

You can store 20 Quick Access banks in each memory bank (except the Zeros bank, which can store 75). See "Storing Objects in Memory Banks" in Chapter 2 of the *Reference Guide* for a breakdown of the Quick Access bank IDs that belong in each memory bank. Press both CHAN/BANK buttons simultaneously to quickly move between memory banks.

The MIDI Program Change commands that the K2vx sends when in Quick Access mode can differ from those in Program or Setup mode. This depends on the setting you have for the PChgType parameter in MIDI mode. If the setting is "Extended" or "Kurzweil," the Program Change commands sent are the same as in Program or Setup mode. If the setting is "QA Extended" or "QA Kurzweil," the K2vx will send Program Change commands that correspond to the current Quick Access bank and the entry you select, not the actual program number of the entry. See "Program Change Formats" in Chapter 10.

Everything you need to know about using Quick Access mode for performance is covered in Chapter 2, in the section called “Playing the Presets,” so we’ll move on to the Quick Access Editor, which you’ll use to create your own Quick Access banks.

The first step in editing Quick Access banks is to select Quick Access mode. Use the CHAN/BANK buttons to select the bank you wish to edit. The currently selected bank is shown in the top line of the Quick Access mode page. Press the EDIT button, and you’ll enter the editor, where you can examine each entry in the bank you selected. The Quick Access Editor page looks like this:

```
editQuickAccess          <>Entry:9
198 Blues in CMOS
Entry:  Type:          199 Default Program
9       Program       1 Righteous Piano
                          2 Mondo Bass
                          3 Killer Drums
                          4 Elvis Again
Name  Save  Delete  Dump
```

The top line gives you the usual mode reminder, and shows you which of the ten entries you’re looking at. The cursor is highlighting the object (program or setup) that’s stored in that entry.

The easiest way to edit the bank is to use the CHAN/BANK buttons to scroll through the ten entries. The entry number will change both at the top of the page, and at the left of the page. As the entry number changes, the highlighted objects at the center of the page will change as well, showing you what’s stored in each entry. On the page above, for example, entry 9 is the current entry. The Type parameter tells you that the object stored at entry 9 is a program. The cursor highlights the program’s ID and name.

In this example, you could select a different program with your favorite data entry method. If you wanted to store a setup in that entry instead of a program, you would move the cursor to the Type parameter and change its value to Setup. The list of objects would change from the program list to the setup list, and you could move the cursor back to the setup list and select another setup. When you select the Entry or Type parameter, the list of objects at the right disappears, leaving only the currently selected object. This makes it easier to see when it’s not highlighted by the cursor.

When you’ve filled each entry with the object you want, press the Name soft button if you want to rename the bank, or press the Save soft button to begin the save procedure. Press the Dump soft button to dump the bank via MIDI System Exclusive.

## Chapter 9

# Effects Mode and the Effects Editor

The K2vx's global effects processor operates much like an outboard MIDI effects box. You can route your sounds through the processor or bypass it, and you can edit its preset effects from the K2vx's front panel. It responds to MIDI programming commands, and you can program control sources to get realtime control over the Wet/Dry mix and two parameters that vary from preset to preset (this is done on the EFFECT page in the Program Editor). You can store ten preset effects in each memory bank (except the Zeros bank, which holds 37). See "Storing Objects in Memory Banks" in the *Reference Guide* for a breakdown of the preset effects ID that belong in each memory bank.

Note: the K2vx's effects processor is mono, not stereo, so keep in mind that if you pan any layers, only the dry portion of the sound (the part without effects) will be panned; the wet portion will remain unchanged.

## The Effects Mode Page

Every K2vx program and setup can be assigned a different preset effect. There are 47 factory preset effects stored in ROM. You can use the Effects Editor to create your own effects, selecting from 27 configurations of effects types—like chorus, flange, delay, etc. Preset effects are applied to the MIX outputs (and to the headphone jack) when the currently selected program (or the program on the FX Channel) is routed to Output Group A.

There are three purposes for Effects mode. The primary purpose is to define the behavior of the global effects processor, instructing the K2vx how to select preset effects when you select programs or setups. The other two purposes are explained below. Press the Effects mode button to enter Effects mode. You'll see the Effects mode entry level page, as shown below. The top line of the page, as usual, reminds you of your current location, and indicates the amount of transposition and the current MIDI channel.

```
EffectsMode   XPose:051  <>Channel:1
Effect :1 Sweet Hall 2
Wet/Dry:50% Wet
FX Mode:Auto
FX Chan:Current
Octav-  Octav+  Panic           Chan-  Chan+
```

### The Soft Buttons on the Effects mode Page

The Octav- and Octav+ soft buttons let you change the MIDI transposition in increments of one octave. The Panic soft button sends All Notes Off and All Controllers Off to the K2vx and on all 16 MIDI channels. The Chan- and Chan+ soft buttons change the current MIDI channel.

## Effects Mode Parameters

PARAMETER	RANGE OF VALUES	DEFAULT
<b>EFFECT</b>	<b>Preset Effect list</b>	<b>1 Sweet Hall</b>
<b>WET/DRY</b>	<b>0 to 100% Wet</b>	<b>50%</b>
<b>FX MODE</b>	<b>Auto, Master, Program, Setup</b>	<b>Auto</b>
<b>FX CHANNEL</b>	<b>None, Current, 1 to 16</b>	<b>Current</b>

### Effect

This indicates the preset effect that's selected while you're in Effects mode. By changing the value of this parameter, you can view and listen to the various preset effects without changing the preset effect for any of the programs or setups. If the FX Mode parameter (explained below) is set to a value of Auto (its default value), then when you enter Effects mode, the value of the Effect parameter automatically changes to match the effect assigned to the most recently selected program or setup. If FX Mode is set to Master, Program, or Setup, the Effect parameter is not automatically updated, so the effect you hear when you enter Effects mode may not be the same as the one assigned to the program or setup you were listening to before you entered Effects mode. There's a list of the factory preset effects in the *Reference Guide*.

### Wet/Dry

This gives you the mix level of the currently selected preset effect. Use one of the data entry methods to adjust the effects mix. 0% removes the effect entirely, while 100% removes all of the dry (unaffected) sound, so that you hear only the wet sound (sound with effect). Your best results will usually come from settings near 50%.

If you're using a stereo insert cable in one or both of the Group A outputs to create a loop to an outboard effects processor, you can bring the effects from the outboard unit into the K2vx and out the MIX outputs. If you want to use only the outboard effects, the easiest way to defeat the K2vx's effects is to set the Wet/Dry mix to 0%, and the FX Mode to Master. You could also go to the CHANLS page in MIDI mode, and set the OutPair parameter to a value of B(DRY), C(DRY), or D(DRY) for one or more channels. Any program assigned to these MIDI channels will then be routed to Output Group B, C, or D, and will appear at the MIX outputs without going through the effects processor.

## Understanding FX Mode and FX Channel

FX Mode and FX Channel are important to understand because they determine how the effects processor is controlled.

The K2vx's effects work as a single effects processor. You can combine different effects together, but you can't split them up between programs or channels. In other words, whichever effect you pick is going to affect all of the programs - you can't have reverb for one program and delay on another.

The FX Mode parameter lets you choose what will control the effects processor. The default value is Auto. This means that the current mode is controlling the effects. Each program and setup has an effect assigned to it. So if you go to Program mode and call up a program, the effect assigned to the program on the effects page in the Program editor will be called up. For example, if you go to Program mode, choose program 1, Acoustic Piano and then go back to the Effects mode, you will see that the effect is Small Hall. If you go back to Program Mode, call up program 2, Stage Piano, and go back to Effects, you will see Medium. Setups function similarly.

The FX Mode parameter works in conjunction with the FX Channel parameter. The default value for this is Current. This means that the program assigned to the MIDI channel currently in the display is in control of the effects. But you can also set this to a specific channel. For example, you could set it so that the Program on Channel 1 is in control of the effects.

As we mentioned, in Auto, either a Program or a Setup can be in control, depending on which mode you are in. You can also set this param to Setup or Program only. The final value is Master. When set to Master, the effect that you call up on this page will be the one you hear, no matter what program you call up. In this situation, the FX channel becomes the channel you would use to send program changes on to call up effects. You can also use this channel to send controllers to alter parameters in real time.

### **Using Effects in Song Mode**

If you are using the K2vx's sequencer, you have a couple of additional options. If you set FX Mode to Master, you can choose your effect on the Effects page and it will remain at that setting. If you put it in Auto, then you will have use of the Effect Channel parameter which is found on the Edit Common page of the song. In this case, the program on the Effect Channel will be controlling the effects. See page 12-26 for information on the EffectChan parameter.

### **FX Mode**

The FX Mode parameter determines how preset effects will be selected when you change programs or setups. Refer to "Understanding FX Mode and FX Channel", above, for more information.

The four possible values for FX Mode are explained below.

**Master:** The preset effect displayed on the Effects mode page will be applied to every program or setup you select, and won't change unless you change it in Effects mode. Use this value when you want to use the same preset effect for every program or setup you select, for example, when recording multi-timbrally. This is also the setting you'll use when you want to be able to use external MIDI to control the K2vx's effects in realtime. See the discussion of FX Channel below.

**Program:** When you select a program on the FX channel (explained below), the K2vx will select the preset effect that's assigned to that program. The program's preset effect is assigned with the Effects Preset parameter on the EFFECT page in the Program Editor. Set FX Mode to Program when you want the preset effect to change every time you select a program on the FX channel. If the value is set to Program, then while you're in Effects mode, the preset effect you hear will be the effect assigned to the program on the FX channel, *not* the one you see displayed. If you select a program on a channel other than the FX channel, the preset effect will not change. If the value of the FX Channel parameter is "Current," then the FX channel always matches the current MIDI channel, and the preset effect will change according to any program you select.

If the value is set to Program and you enter Setup mode, the K2vx uses the preset effect that's assigned to the program on the FX Channel (explained below). Therefore, the effect will change only if you select a setup containing a program assigned to the FX Channel.

**Setup:** When you select a setup, the K2vx will select the preset effect that's assigned to that setup. The setup preset effect is assigned with the Effect parameter in the Setup Editor. Set FX Mode to Setup when you want the preset effect to change every time you select a setup. The preset effect you hear while in Effects mode will be the effect assigned to the current setup, not the one you see displayed. If you enter Program mode, the effect will not change when you select programs.

**Auto:** At this setting, the FX mode automatically matches the K2vx's operating mode. Selecting a program will select the effect assigned to that program. Selecting a setup will select the effect assigned to that setup. When you enter Effects mode, the value of the Effect parameter is automatically updated to match the effect assigned to the most recently selected program or setup, so when you enter Effects mode, the effect you hear will be the one you were just listening to.

This setting is especially useful if you're using Quick Access mode, and you have programs and setups loaded into your Quick Access banks. A setting of Auto will ensure that the desired effects are selected when you select a program or setup. In Song mode, a setting of Auto lets you use the Effect Channel parameter on the Edit Common page of the song. In this case, the program on the Effect Channel will be controlling the effects. See page 12-26 for information on the EffectChan parameter.

You'll normally leave the FX mode set to Auto, and use the other settings when you want to override the automatic selection of preset effects.

If you've set the FX Mode parameter to a value of other than Auto, and you change the preset effect in the Program or Setup Editor, the value of the FX mode may revert to Auto. This will happen a) when FX Mode is set to Setup and you change the preset effect in the Program Editor; b) when FX mode is set to Program and you change the preset effect in the Setup Editor, and c) when FX Mode is set to Master and you change the preset effect in *either* the Program or Setup Editor. This ensures that if you change the preset effect in either Program or Setup mode, you'll be able to hear the change.

### FX Channel

This parameter is closely tied to the FX Mode parameter, and exists to help the K2vx decide which preset effect to select. Refer to "Understanding FX Mode and FX Channel", above, for more information.

When you select a program or setup, the K2vx checks the value of the FX Channel parameter. It then selects the preset effect that's assigned to the program on that channel. The value of the FX Mode parameter determines the range of values for the FX Channel parameter. We'll discuss these values in terms of each of the values for the FX Mode parameter.

**FX Mode = Auto:** In this case, the preset effect is automatically selected for each program or setup you select. In Setup mode, the FX channel is irrelevant. In Program mode, the FX channel defines the one MIDI channel on which program changes will change the preset effect. This can be set to any of the 16 MIDI channels, or to a value of Current, which means that the FX Channel always matches the current MIDI channel as set in Program mode. When you're in Program mode and you want the preset effect to change whenever you change programs from the K2vx's front panel, use a value of Current.

If the value for FX Mode is Auto, and the value for FX Channel is Current, then when you're in Song mode—or you move to another mode during song playback—the channel indicated by the EffectChan parameter on the Edit Song:COMMON page controls the selection of effects. This prevents the effects from changing if you change MIDI channels during song playback. When you exit Song mode—and you're not recording or playing back a song—the currently selected program or setup will regain control over the selection of effects. See page 12-26 for information on the EffectChan parameter.

**FX Mode = Master:** At this setting the preset effect doesn't change when you select programs or setups, so the FX Channel isn't needed for preset effect selection. When the FX mode is set to Master, the FX channel parameter selects the MIDI channel for realtime effects control via MIDI. Choose a value of 1–16 to specify the channel on which the K2vx will accept incoming MIDI control signals and apply them to the realtime effects parameters (explained later in this chapter). Choose a value of None to disable external MIDI effects control.

FX Mode = Program: In this case, the available values for the FX Channel parameter are Current, and 1–16. Here, the default value of Current is your most likely choice, since preset effect selection and realtime control messages will be tied to the current MIDI channel. Set a value of 1–16 when you want to be able to change programs on the current MIDI channel without changing the preset effect.

FX Mode = Setup: At this setting, the preset effect is determined by the value for the Effect parameter on the Setup Editor page, so the FX Channel parameter is irrelevant, and the only available value is None. Realtime effects are disabled in Setup mode, since there's no way to determine which program should generate the control source signals assigned to control the realtime effects.

## Another Use for Effects Mode

As we mentioned, the main purpose of Effects mode is to tell the K2vx how to select preset effects when you select programs or setups—but there's another convenient use as well.

### Global Preset Effect

If you set the FX Mode to Master, the effects assignments for programs and setups are overridden. The Effect parameter becomes global. Even when you exit Effects mode and select programs and setups, the preset effect remains unchanged. This is an easy way to hear a particular effect applied to several different programs—without editing the programs themselves. You'll use this when you're playing or recording a multi-timbral sequence and you want to apply a certain effect to every sound.

Setting the FX Mode to Master tells the K2vx to ignore the preset effects settings at the program and setup level. If at some later point you go to Program or Setup mode and change one of the preset effect settings there, the K2vx will assume that the newly selected preset effect is the one you want to use. The value of FX Mode will automatically revert to Auto, so that the newly selected effect can be applied to the program or setup you're editing. This defeats the override, and from that point on, selecting a program or setup will select the preset effect assigned to that program or setup. If you still want the override to apply, return to Effects mode and reset the FX Mode parameter to Master.

## The Effects Editor

There are three ways to reach the Effects Editor. One is to select Effects Mode, select the effect you wish to edit, then press EDIT. You also can get there from the EFFECT page in the Program Editor, simply by pressing EDIT. If you're in Setup mode, enter the Setup Editor by pressing EDIT, then select the Program parameter and press EDIT again. You're now in the Program Editor, where you can select the EFFECT page, and press EDIT to enter the Effects Editor. In any case, you'll see a page that looks something like this:

```

EditEffect      <>Config:Ultimate Reverb
DecayTime:1.0s  EarlyDly:1ms  Dry Level:0
Room Vol :0.8   EarlyDff:5     Early Lvl:8
HFDamping  :8    LaterDly:9ms  Later Lvl:0
Envelopmnt:0    LaterDff:0
Name  Save  Delete  Dump

```

The top line tells you that you're in the Effects Editor, and identifies the configuration (explained below) of the currently selected preset effect. The bottom line labels the active soft buttons. The parameters you see depend on the currently selected configuration.

The Name, Save, and Delete soft buttons handle the basic library functions for user-defined effects, and the Dump soft button initiates a System Exclusive dump of the current effect's settings.

## Editing Effects

The K2vx interacts with the global effects processor as if it were a separate multi-effects unit that's been mounted internally. There are 47 factory preset effects, and room for you to store 80 of your own preset effects. Like any other K2vx object, the preset effects are edited by selecting parameters and changing their values.

Each factory preset effect—as well as any preset effects you create yourself—uses one of 27 available configurations. The configurations are simply different sets of familiar programmable effects generators, the kind you'd find on any outboard effects unit—reverb, chorus, parametric and graphic EQ, delay, and mixers. Various combinations of these effects generators make up the configurations.

When you enter the Effects Editor, the configuration for the currently selected preset effect is shown on the top line of the display. If you want to tweak a preset effect slightly, simply adjust the values for one or more of the parameters on the page. You can then name and save the new effect, either “replacing” the ROM preset or saving to a RAM location by giving it an unused ID. If you “replace” a ROM effect with one of your own, deleting your effect will restore the ROM effect.

Use the CHAN/BANK buttons to change the configuration to be used by the currently selected preset effect. You'll want to do this only when you're creating new effects that differ in type from the current effect, since changing the configuration can drastically change the nature of the effect.



## Configurations and Parameters

The configurations define various types and combinations of effects—reverb, delay, EQ, etc. This section shows you the pages corresponding to each of the configurations. Here's the entire list:

DRY  
 STEREO CHORUS  
 STEREO FLANGE  
 STEREO DELAY  
 4-TAP DELAY  
 ULTIMATE REVERB  
 ROOM SIMULATOR  
 GATED REVERB  
 REVERSE REVERB  
 PARAMETRIC EQ  
 GRAPHIC EQ  
 PARAMETRIC EQ+DELAY+MIXER  
 PARAMETRIC EQ+CHORUS+MIXER  
 CHORUS+ROOM+MIXER  
 DELAY+ROOM+MIXER  
 CHORUS+HALL+MIXER  
 DELAY+HALL+MIXER  
 EQ+GATED REVERB+MIXER  
 EQ+REVERSE REVERB+MIXER  
 PARAMETRIC EQ+CHORUS+DELAY+MIXER  
 PARAMETRIC EQ+FLANGE+DELAY+MIXER  
 CHORUS +DELAY+ROOM+MIXER  
 FLANGE+DELAY+ROOM+MIXER  
 CHORUS+DELAY+HALL+MIXER  
 FLANGE+DELAY+HALL+MIXER  
 EQ+CHORUS+4-TAP-DELAY+MIXER  
 EQ+FLANGE+4-TAP DELAY+MIXER

*Dry*

```
editEffect <>Config: Dry
```

```
Name Save Delete Dump
```

Use this configuration if you want a particular program or setup to play through the MIX outputs without effects, even though it's routed to Output Group A.

**Stereo Chorus**

editEffect <>Config: Stereo Chorus

Chr Delay: 0ms      Dry Level: 0  
LFO Speed: 8        Right Lvl: 10  
LFO Depth: 45       Left Lvl : 10

Name   Save   Delete   Dump

PARAMETER	RANGE OF VALUES
CHORUS DELAY	0 to 60 milliseconds
LFO SPEED	0 to 65
LFO DEPTH	0 to 99
DRY LEVEL	0 to 10
RIGHT—LEFT LEVEL	0 to 10

**Stereo Flange**

editEffect <>Config: Stereo Flange

FlngDelay: 1ms      Dry Level: 0  
LFO Speed: 3        Right Lvl: 10  
LFO Depth: 63       Left Lvl : 10  
Feedback : 80%

Name   Save   Delete   Dump

PARAMETER	RANGE OF VALUES
FLANGE DELAY	10 milliseconds
LFO SPEED	0 to 65
LFO DEPTH	0 to 99
FEEDBACK	0 to 99%
DRY LEVEL	0 to 10
RIGHT—LEFT LEVEL	0 to 10

*Stereo Delay*

```

editEffect <>Config: Stereo Delay
DelayTime: 135ms Dry Level: 0
Feedback : 30% Right Lvl: 10
Left Lvl : 10

```

```

Name Save Delete Dump

```

PARAMETER	RANGE OF VALUES
DELAY TIME	0 to 750 milliseconds
FEEDBACK	0 to 99%
DRY LEVEL	0 to 10
RIGHT—LEFT LEVEL	0 to 10

*4-Tap Delay*

```

editEffect <>Config: 4-Tap Delay
Tap1Delay: 345ms Dry Level: 0
Tap2Delay: 660ms Tap1Lvl R: 9 L: 0
Tap3Delay: 355ms Tap2Lvl R: 0 L: 0
Tap4Delay: 680ms Tap3Lvl R: 6 L: 6
FeedDelay: 680ms Tap4Lvl R: 3 L: 6
FeedBack : 20%
Name Save Delete Dump

```

PARAMETER	RANGE OF VALUES
TAP 1—4 DELAY	0 to 1500 milliseconds
FEEDBACK DELAY	0 to 1500 milliseconds
FEEDBACK	0 to 99%
DRY LEVEL	0 to 10
TAP LEVEL 1—4 RIGHT	0 to 10
TAP LEVEL 1—4 LEFT	0 to 10

*Ultimate Reverb*

```
editEffect <>config:Ultimate Reverb
DecayTime:1.0s EarlyDly:0ms Dry Level:0
Room Vol :0.9 EarlyDff:3 Early Lvl:7
HFDamping:2 LaterDly:0ms Later Lvl:7
Envelopmnt:9 LaterDff:6
Name Save Delete Dump
```

PARAMETER	RANGE OF VALUES
DECAY TIME	1 to 99 seconds
ROOM VOLUME	0.0 to 0.9
HIGH-FREQUENCY DAMPING	0 to 9
ENVELOPMENT	0 to 9
EARLY DELAY	0 to 70 milliseconds
EARLY DIFFUSION	0 to 9
LATER DELAY	0 to 70 milliseconds
LATER DIFFUSION	0 to 9
DRY LEVEL	0 to 10
EARLY LEVEL	0 to 10
LATER LEVEL	0 to 10

*Room Simulator*

```
editEffect <>config:Room Simulator
GrossSize:Studio Dry Level:9
DecayTime:2.80s Rev Level:6
ListenPos:Front
HFDamping:5
Name Save Delete Dump
```

PARAMETER	RANGE OF VALUES
GROSS SIZE	Studio, Chamber, Club, Hall, Arena
DECAY TIME	0.7 to 23.8 seconds
LISTENING POSITION	Front, Middle, Back
HIGH-FREQUENCY DAMPING	0 to 9
DRY LEVEL	0 to 10
REVERB LEVEL	0 to 10

*Gated Reverb*

```

EditEffect <>Config:Gated Reverb

```

```

Pre-Delay:0ms Dry Level:10
Envelope :Flat AccentLvl:0
DecayTime:250ms Right Lvl:10
AccentDly:0ms Left Lvl :10

```

```

Name Save Delete Dump

```

PARAMETER	RANGE OF VALUES
PRE-DELAY	0 to 80 milliseconds
ENVELOPE	Flat, Decaying
DECAY TIME	50 to 600 milliseconds
ACCENT DELAY	±50 milliseconds
DRY LEVEL	0 to 10
ACCENT LEVEL	0 to 10
RIGHT—LEFT LEVEL	0 to 10

*Reverse Reverb*

```

EditEffect <>Config:Reverse Reverb

```

```

Pre-Delay:45ms Dry Level:0
Rev Time :550ms AccentLvl:3
AccentDly:50ms Right Lvl:10
Left Lvl :10

```

```

Name Save Delete Dump

```

PARAMETER	RANGE OF VALUES
PRE-DELAY	0 to 80 milliseconds
REVERB TIME	50 to 600 milliseconds
ACCENT DELAY	±50 milliseconds
DRY LEVEL	0 to 10
ACCENT LEVEL	0 to 10
RIGHT—LEFT LEVEL	0 to 10

*Parametric EQ*

```

EditEffect <> Config:Parametric EQ
Band1 Frq:0.10KHz
Band1 Lvl:-12dB      EQ Level :9
Band2 Frq:0.80KHz
Band2 Lvl:6dB
Band3 Frq:0.48KHz
Band3 Lvl:6dB
Name Save Delete Dump
    
```

PARAMETER	RANGE OF VALUES
BAND 1 FREQUENCY	0.1 to 12.60 KHz (cycles per second)
BAND 1 LEVEL	±12 DB
BAND 2—3 FREQUENCY	0.1 to 12.80 KHz
BAND 2—3 LEVEL	±12 DB
EQ LEVEL	0 to 10

*Graphic EQ*

```

EditEffect <> Config:Graphic EQ
63: 125:250:500:1k: 2k: 4k: 8k: 16k:
4dB 4dB 2dB 0dB -2d 2dB 4dB 10d 8dB
Name Save Delete Dump
    
```

PARAMETER	RANGE OF VALUES
ALL (Frequency in Hz)	±12 dB in 2dB increments

*Parametric EQ, Delay, and Mixer*

```

EditEffect <> Config:ParamDelay+Mixer
Band1Frq:0.10K
Band1Lvl:-12dB DelaySrc:EQ Dry Lvl :6
Band2Frq:0.10K DlyInLvl:9 EQ Lvl :0
Band2Lvl:-12dB DlyTime :0ms DlyR Lvl:0
Band3Frq:0.10K Feedback:60% DlyL Lvl:0
Band3Lvl:0dB
Name Save Delete Dump

```

PARAMETER	RANGE OF VALUES
BAND 1 FREQUENCY	0.10 to 12.60 KHz
BAND 1 LEVEL	±12 dB
BAND 2—3 FREQUENCY	0.10 to 12.80 KHz
BAND 2—3 LEVEL	±12 dB
DELAY SOURCE	EQ, Dry
DELAY IN LEVEL	0 to 10
DELAY TIME	0 to 1500 milliseconds
FEEDBACK	0 to 99%
DRY LEVEL	0 to 10
EQ LEVEL	0 to 10
DELAY LEVEL RIGHT—LEFT	0 to 10

*Parametric EQ, Chorus, and Mixer*

```

EditEffect <> Config:ParamChorus+Mix
Band1Frq:9.05K
Band1Lvl:10dB ChorSrc :Dry Dry Lvl :9
Band2Frq:10.00 ChorDly :10ms EQ Lvl :9
Band2Lvl:10dB LFOSpeed:0 ChrR Lvl:9
Band3Frq:0.10K LFODepth:10 ChrL Lvl:9
Band3Lvl:10dB
Name Save Delete Dump

```

PARAMETER	RANGE OF VALUES
BAND 1 FREQUENCY	0.10 to 12.60 KHz
BAND 1 LEVEL	±12 dB
BAND 2—3 FREQUENCY	0.10—12.80 KHz
BAND 2—3 LEVEL	±12 dB
CHORUS SOURCE	EQ, Dry
CHORUS DELAY	0 to 60 milliseconds
LFO SPEED	0 to 65
LFO DEPTH	0 to 99
DRY LEVEL	0 to 10
EQ LEVEL	0 to 10
CHORUS RIGHT—LEFT LEVEL	0 to 10

*Chorus, Room Reverb, and Mixer*

```

EditEffect <>Config:Chorus+Room+Mix
ChorDly :0ms RevIn Dry:0 Dry Lvl :0
LFOSpeed:0 RevIn Chr:0 ChrR Lvl:0
LFODepth:0 RevPreDly:0ms ChrL Lvl:0
HiFrqDamp:Wrm RevR Lvl:0
Rev Decay:0.1s RevL Lvl:0
Name Save Delete Dump
    
```

PARAMETER	RANGE OF VALUES
CHORUS DELAY	0 to 60 milliseconds
LFO SPEED	0 to 65
LFO DEPTH	0 to 99
REVERB IN DRY	0 to 10
REVERB IN CHORUS	0 to 10
REVERB PRE-DELAY	0 to 60 milliseconds
HIGH-FREQUENCY DAMPING	Warm, Soft, Bright
REVERB DECAY	0.1 to 1.2 seconds
DRY LEVEL	0 to 10
CHORUS LEVEL RIGHT—LEFT	0 to 10
REVERB LEVEL RIGHT—LEFT	0 to 10

*Delay, Room Reverb, and Mixer*

```

EditEffect <>Config:Delay+Room+Mixer
DlyTime :140m RevIn Dry:0 Dry Lvl :0
Feedback:0% RevIn Dly:9 DlyR Lvl:0
RevPreDly:10ms DlyL Lvl:0
HiFrqDamp:Wrm RevR Lvl:0
Rev Decay:0.1s RevL Lvl:0
Name Save Delete Dump
    
```

PARAMETER	RANGE OF VALUES
DELAY TIME	0 to 750 milliseconds
FEEDBACK	0 to 99%
REVERB IN DRY	0 to 10
REVERB IN DELAY	0 to 10
REVERB PRE-DELAY	0 to 60 milliseconds
HIGH FREQUENCY DAMPING	Warm, Soft, Bright
REVERB DECAY	0.1 to 1.2 seconds
DRY LEVEL	0 to 10
DELAY LEVEL RIGHT—LEFT	0 to 10
REVERB LEVEL RIGHT—LEFT	0 to 10



*Chorus, Hall Reverb, and Mixer*

```

EditEffect <>Config:Chorus+Hall+Mix
ChorDly :0ms RevIn Dry:9 Dry Lvl :0
LFOSpeed:9 RevIn Chr:0 ChrR Lvl:0
LFODepth:9 RevPreDly:0ms ChrL Lvl:0
HiFrgDamp:Wrm RevR Lvl:0
Rev Decay:1.00 RevL Lvl:0
Name Save Delete Dump

```

PARAMETER	RANGE OF VALUES
CHORUS DELAY	0 to 60 milliseconds
LFO SPEED	0 to 65
LFO DEPTH	0 to 99
REVERB IN DRY	0 to 10
REVERB IN CHORUS	0 to 10
REVERB PRE-DELAY	0 to 60 milliseconds
HIGH-FREQUENCY DAMPING	Warm, Soft, Bright
REVERB DECAY	1.00 to 20.0 seconds
DRY LEVEL	0 to 10
CHORUS LEVEL RIGHT—LEFT	0 to 10
REVERB LEVEL RIGHT—LEFT	0 to 10

*Delay, Hall Reverb, and Mixer*

```

EditEffect <>Config:Delay+Hall+Mixer
DlyTime :0ms RevIn Dry:9 Dry Lvl :0
Feedback:90% RevIn Dly:9 DlyR Lvl:0
RevPreDly:0ms DlyL Lvl:0
HiFrgDamp:Brt RevR Lvl:0
Rev Decay:1.00 RevL Lvl:0
Name Save Delete Dump

```

PARAMETER	RANGE OF VALUES
DELAY TIME	0 to 750 milliseconds
FEEDBACK	0 to 99%
REVERB IN DRY	0 to 10
REVERB IN DELAY	0 to 10
REVERB PRE-DELAY	0 to 60 milliseconds
HIGH FREQUENCY DAMPING	Warm, Soft, Bright
REVERB DECAY	1.00 to 20.0 seconds
DRY LEVEL	0 to 10
DELAY LEVEL RIGHT—LEFT	0 to 10
REVERB LEVEL RIGHT—LEFT	0 to 10

*EQ, Gated Reverb, and Mixer*

```

editEffect <> Config:EQ+Gated+Mixer
LoPassFrq:0.10KHz EQ R Lvl :2
EQ L Lvl :0
RevPreDly:9ms Acc R Lvl:0
GateEnv :Flat Acc L Lvl:0
Rev Decay:500ms Rev R Lvl:0
AccentDly:-50ms Rev L Lvl:0
Name Save Delete Dump
    
```

PARAMETER	RANGE OF VALUES
LOWPASS FILTER CUTOFF FREQ.	0.10 to 18.00KHz
REVERB PRE-DELAY	0 to 80 milliseconds
GATE ENVELOPE	Flat, Decaying
GATE DECAY TIME	50 to 600 milliseconds
ACCENT DELAY	±50 milliseconds
EQ LEVEL RIGHT—LEFT	0 to 10
ACCENT LEVEL RIGHT—LEFT	0 to 10
REVERB LEVEL RIGHT—LEFT	0 to 10

*EQ, Reverse Reverb, and Mixer*

```

editEffect <> Config:EQ+Reverse+Mixer
LoPassFrq:15.00KHz EQ R Lvl :9
EQ L Lvl :2
RevPreDly:0ms Acc R Lvl:0
Rev Time :50ms Acc L Lvl:0
AccentDly:-50ms Rev R Lvl:0
Rev L Lvl:0
Name Save Delete Dump
    
```

PARAMETER	RANGE OF VALUES
LOWPASS FILTER CUTOFF FREQ.	100 Hz to 18.00KHz
REVERB PRE-DELAY	0 to 80 milliseconds
REVERSE TIME	50 to 600 milliseconds
ACCENT DELAY	±50 milliseconds
EQ LEVEL RIGHT—LEFT	0 to 10
ACCENT LEVEL RIGHT—LEFT	0 to 10
REVERSE LEVEL RIGHT—LEFT	0 to 10

*Parametric EQ, Chorus, Delay, and Mixer*

```

EditEffect <> Config: ParamEQof013011x
Frq1:0.10 ChrSrc:EQ DlyEQSrc:EQ Dry :1
Lv11:-12d ChrDly:0m DlyEQIn :0 EQ :1
Frq2:0.10 LFOSpd:0 DlyChrIn:0 DlyR:1
Lv12:-12d LFODep:0 DlyTime :9ms DlyL:1
Frq3:0.48 Feedback:0% ChrR:1
Lv13:-8dB ChrL:1
Name Save Delete Dump

```

PARAMETER	RANGE OF VALUES
BAND 1 FREQUENCY	0.1 to 12.60 KHz
BAND 1 LEVEL	±12 dB
BAND 2—3 FREQUENCY	0.1 to 12.80 KHz
BAND 2—3 LEVEL	±12 dB
CHORUS EQ SOURCE	EQ, Dry
CHORUS DELAY	0 to 60 milliseconds
CHORUS LFO SPEED	0 to 65
CHORUS LFO DEPTH	0 to 99
DELAY EQ SOURCE	EQ, Dry
DELAY EQ IN	0 to 10
DELAY CHORUS IN	0 to 10
DELAY TIME	0 to 1500 milliseconds
DELAY FEEDBACK	0 to 99%
DRY LEVEL	0 to 10
EQ LEVEL	0 to 10
DELAY LEVEL RIGHT—LEFT	0 to 10
CHORUS LEVEL RIGHT—LEFT	0 to 10

*Parametric EQ, Flange, Delay, and Mixer*

```

EditEffect <> Config:ParamEQ+0.9+0.1
Frq1:0.10 FlaSrc:EQ DlyEQSrc:EQ
Lvl1:10dB FlaDly:0 DlyEQIn :9 EQ :9
Frq2:10.0 LFOSpd:64 DlyFlaIn:0 DlyR:9
Lvl2:10dB LFODep:98 DlyTime :9ms DlyL:9
Frq3:10.8 FeedBk:10 Feedback:90% FlaR:9
Lvl3:10dB FlaL:9
Name Save Delete Dump
    
```

PARAMETER	RANGE OF VALUES
BAND 1 FREQUENCY	100 Hz to 12.60 KHz
BAND 1 LEVEL	±12 dB
BAND 2—3 FREQUENCY	0.1 to 12.80 KHz
BAND 2—3 LEVEL	±12 dB
FLANGE SOURCE	EQ, Dry
FLANGE DELAY	0 to 10 milliseconds
LFO SPEED	0 to 65
LFO DEPTH	0 to 99
FLANGE FEEDBACK	0 to 99%
DELAY EQ SOURCE	EQ, Dry
DELAY EQ IN	0 to 10
DELAY FLANGE IN	0 to 10
DELAY TIME	0 to 1500 milliseconds
DELAY FEEDBACK	0 to 99%
DRY LEVEL	0 to 10
EQ LEVEL	0 to 10
DELAY LEVEL RIGHT—LEFT	0 to 10
FLANGE LEVEL RIGHT—LEFT	0 to 10

*Chorus, Delay, Room Reverb, and Mixer*

```

edit effect <> Config: Chorus+Delay+Room+Mix
ChrDly: 0m DlyDryIn: 0 RevDryIn: 9 CR: 0
LFOSpd: 0 DlyChrIn: 0 RevChrIn: 0 CL: 1
LFODep: 0 DlyTime: 0ms RevDlyIn: 6 DR: 1
          Feedback: 0% RvPreDly: 0 DL: 1
          HFrqDamp: WrmRR: 1
          Rev Time: 0.1RL: 1
DryLvl: 0
Name Save Delete Dump

```

PARAMETER	RANGE OF VALUES
CHORUS DELAY	0 to 60 milliseconds
CHORUS LFO SPEED	0 to 65
CHORUS LFO DEPTH	0 to 99
DRY LEVEL	0 to 10
DELAY DRY IN	0 to 10
DELAY CHORUS IN	0 to 10
DELAY TIME	0 to 750 milliseconds
DELAY FEEDBACK	0 to 99%
REVERB DRY IN	0 to 10
REVERB CHORUS IN	0 to 10
REVERB DELAY IN	0 to 10
REVERB PRE-DELAY	0 to 60 milliseconds
HIGH FREQUENCY DAMPING	Warm, Soft, Bright
REVERB TIME (DECAY)	100 milliseconds to 1.2 seconds
CHORUS LEVEL RIGHT—LEFT	0 to 10
DELAY LEVEL RIGHT—LEFT	0 to 10
REVERB LEVEL RIGHT—LEFT	0 to 10

*Flange, Delay, Room Reverb, and Mixer*

```

EditEffect <> ConfigFile+Uly+Room+Mx
FlaDly:0m DlyDryIn:9 RevDryIn:9 FR:1
LFOSpd:64 DlyFlaIn:0 RevFlaIn:9 FL:1
LFODep:98 DlyTime :9ms RevDlyIn:0 DR:1
FeedBk:90 Feedback:90% RvPreDly:0 DL:1
DryLvl:0 HFrqDamp:WrmRR:1
Rev Time:1.0RL:1
Name Save Delete Dump
    
```

PARAMETER	RANGE OF VALUES
FLANGE DELAY	0 to 60 milliseconds
FLANGE LFO SPEED	0 to 65
FLANGE LFO DEPTH	0 to 99
FLANGE FEEDBACK	0 to 99%
DRY LEVEL	0 to 10
DELAY DRY IN	0 to 10
DELAY FLANGE IN	0 to 10
DELAY TIME	0 to 750 milliseconds
DELAY FEEDBACK	0 to 99%
REVERB DRY IN	0 to 10
REVERB FLANGE IN	0 to 10
REVERB DELAY IN	0 to 10
REVERB PRE-DELAY	0 to 60 milliseconds
HIGH FREQUENCY DAMPING	Warm, Soft, Bright
REVERB TIME (DECAY)	100 milliseconds to 1.2 seconds
FLANGE LEVEL RIGHT—LEFT	0 to 10
DELAY LEVEL RIGHT—LEFT	0 to 10
REVERB LEVEL RIGHT—LEFT	0 to 10

*Chorus, Delay, Hall Reverb, and Mixer*

```

edit effect <> Config: Chorus+Delay+Hall+Mixer
ChrDly: 0m DlyDryIn: 0 RevDryIn: 9 CR: 0
LFOSpd: 0 DlyChrIn: 0 RevChrIn: 0 CL: 1
LFODep: 0 DlyTime: 9ms RevDlyIn: 6 DR: 1
          Feedback: 90% RvPreDly: 0 DL: 1
          HFrqDamp: WrmRR: 1
          Rev Time: 0.1 RL: 1
DryLvl: 0
Name Save Delete Dump

```

PARAMETER	RANGE OF VALUES
CHORUS DELAY	0 to 60 milliseconds
CHORUS LFO SPEED	0 to 65
CHORUS LFO DEPTH	0 to 99
DRY LEVEL	0 to 10
DELAY DRY IN	0 to 10
DELAY CHORUS IN	0 to 10
DELAY TIME	0 to 750 milliseconds
DELAY FEEDBACK	0 to 99%
REVERB DRY IN	0 to 10
REVERB CHORUS IN	0 to 10
REVERB DELAY IN	0 to 10
REVERB PRE-DELAY	0 to 60 milliseconds
HIGH FREQUENCY DAMPING	Warm, Soft, Bright
REVERB TIME (DECAY)	1.0 to 20.0 seconds
CHORUS LEVEL RIGHT—LEFT	0 to 10
DELAY LEVEL RIGHT—LEFT	0 to 10
REVERB LEVEL RIGHT—LEFT	0 to 10

*Flange, Delay, Hall Reverb, and Mixer*

```

EditEffect <> Config Flange+Delay+Hall+Mixer
FlaDly:0m DlyDryIn:9 RevDryIn:9 FR:1
LFOSpd:64 DlyFlaIn:0 RevFlaIn:9 FL:1
LFODep:98 DlyTime :9ms RevDlyIn:0 DR:1
FeedBk:90 Feedback:90% RvPreDly:0 DL:1
DryLvl:0 HFrqDamp:WrmRR:1
Rev Time:1.0RL:1
Name Save Delete Dump
    
```

PARAMETER	RANGE OF VALUES
FLANGE DELAY	0 to 60 milliseconds
FLANGE LFO SPEED	0 to 65
FLANGE LFO DEPTH	0 to 99
FLANGE FEEDBACK	0 to 99%
DRY LEVEL	0 to 10
DELAY DRY IN	0 to 10
DELAY FLANGE IN	0 to 10
DELAY TIME	0 to 750 milliseconds
DELAY FEEDBACK	0 to 99%
REVERB DRY IN	0 to 10
REVERB FLANGE IN	0 to 10
REVERB DELAY IN	0 to 10
REVERB PRE-DELAY	0 to 60 milliseconds
HIGH FREQUENCY DAMPING	Warm, Soft, Bright
REVERB TIME (DECAY)	100 milliseconds to 1.2 seconds
FLANGE LEVEL RIGHT—LEFT	0 to 10
DELAY LEVEL RIGHT—LEFT	0 to 10
REVERB LEVEL RIGHT—LEFT	0 to 10



*EQ, Chorus, 4-Tap Delay, and Mixer*

```

editEffect <>Config:EQ+Chor+4Tap+Mx
LoPass:0 DlyEQIn:8 FdbkDly:0ms TP1R:1
0.67kHz DlyChIn:0 FeedBck:0% TP1L:1
ChrDly:0m Tap1Dly:0ms Tap2:1
LFOSpd:0 Tap2Dly:0ms EQLvl:1 Tap3:1
LFODep:8 Tap3Dly:7ms ChrLv:1 Tp4R:1
Tap4Dly:0ms Tp4L:1
Name Save Delete Dump

```

PARAMETER	RANGE OF VALUES
LOWPASS FILTER CUTOFF FREQ.	100 Hz to 18 KHz
CHORUS DELAY	0 to 60 milliseconds
CHORUS LFO SPEED	0 to 65
CHORUS LFO DEPTH	0 to 99
DELAY EQ IN	0 to 10
DELAY CHORUS IN	0 to 10
TAP 1—4 DELAY	0 to 1500 milliseconds
FEEDBACK DELAY	0 to 1500 milliseconds
4-TAP FEEDBACK AMOUNT	0 to 99%
EQ LEVEL	0 to 10
CHORUS LEVEL	0 to 10
TAP 1—4 LEVEL RIGHT—LEFT	0 to 10

*EQ, Flange, 4-Tap Delay, and Mixer*

```

editEffect <>Config:EQ+Flan+4Tap+Mx
LoPass:0 DlyEQIn:8 FdbkDly:0ms TP1R:1
0.67kHz DlyFlIn:0 FeedBck:0% TP1L:1
FlaDly:0m Tap1Dly:0ms Tap2:1
LFOSpd:0 Tap2Dly:0ms EQLvl:1 Tap3:1
LFODep:8 Tap3Dly:7ms FlLvl:1 Tp4R:1
FeedBk:90 Tap4Dly:0ms Tp4L:1
Name Save Delete Dump

```

PARAMETER	RANGE OF VALUES
LOWPASS FILTER CUTOFF FREQ.	100 Hz to 18 KHz
FLANGE DELAY	0 to 60 milliseconds
FLANGE LFO SPEED	0 to 65
FLANGE LFO DEPTH	0 to 99
FLANGE FEEDBACK	0 to 99%
DELAY EQ IN	0 to 10
DELAY FLANGE IN	0 to 10
TAP 1—4 DELAY	0 to 1500 milliseconds
FEEDBACK DELAY	0 to 1500 milliseconds
4-TAP FEEDBACK AMOUNT	0 to 99%
EQ LEVEL	0 to 10
FLANGE LEVEL	0 to 10
TAP 1—4 LEVEL RIGHT—LEFT	0 to 10



## Chapter 10

# MIDI Mode

Press the MIDI mode button to enter MIDI mode. Here you'll configure the K2vx's response to incoming MIDI messages, as well as customize those that the K2vx sends to other MIDI devices in your system.

When you enter MIDI mode, you'll see one of the three available MIDI mode pages. When you exit MIDI mode, the K2vx remembers which page you were on. The next time you select MIDI mode, that page appears.

## The Transmit (XMIT) Page

Press the XMIT soft button, and the MIDI XMIT page appears. Use these parameters to control how the K2vx sends MIDI information to its MIDI Out port. These settings will affect the K2vx's response to your MIDI controller (like Mod Wheel messages, etc.). It also affects the responses of other MIDI devices receiving MIDI from the K2vx, when the Local Keyboard Channel parameter is set to match the transmission channel of your MIDI controller.

It's important to remember that many of the settings of the XMIT page are in effect only when a *program* is selected, either in Program mode or in Quick Access mode. If a *setup* is selected, in Setup mode or in Quick Access mode, the setup's MIDI settings override the corresponding settings on the XMIT page. The XMIT page looks like this:

```

MIDI Mode: TRANSMIT
Channel : 1      PBend: On   ModWhl: MWheel
TransPos: 0S1   PChng: On   FtSw1 : Sustain
Control : Both  Btns: On   FtSw2 : SostPd
PChgType: Extended  U: On   CPedal: Foot
VelocMap: 1 Linear      Slider: Data
PressMap: 1 Linear      Press : MPress
XMIT  RECU  CHANLS  ProgChg  RsetChg  Panic

```

PARAMETER	RANGE OF VALUES	DEFAULT
CHANNEL	1 to 16	1
TRANSPOSITION	±60 semitones	0
CONTROL	Both, MIDI, Local	Both
PROGRAM CHANGE TYPE	(See below)	Extended
VELOCITY MAP	Velocity Map list	1 Linear
PRESSURE MAP	Pressure Map list	1 Linear
PITCH BEND	Off, On	On
PROGRAM CHANGE	Off, On	On
BUTTONS	Off, On	On
VOLUME	Off, On	On
MOD WHEEL	MIDI Control Source list	MWheel (MIDI 01)
FOOT SWITCH 1	MIDI Control Source list	Sustain (MIDI 64)
FOOT SWITCH 2	MIDI Control Source list	Sostenuto (MIDI 66)
CONTINUOUS CONTROL PEDAL	MIDI Control Source list	Foot (MIDI 04)
CONTROLLER SLIDER	MIDI Control Source list	Data (MIDI 06)
MONO PRESSURE	MIDI Control Source list	MPress

Keep in mind that only the Program Change Type, Program Change, and Buttons parameters will operate as programmed if the LocalKbdCh parameter does not match the transmit channel of your MIDI controller. If you set them to match, the remaining parameters will take effect as well.

### **Channel**

This defines which MIDI channel the K2vx uses to transmit MIDI messages. The value for this parameter matches the current MIDI channel displayed on the top line of the Program mode page. If you change the current MIDI channel while in Program mode, the setting of this parameter will change accordingly, and vice versa.

### **Transpose**

This parameter affects the transposition that's applied to the MIDI data stream. Adjusting this parameter will transpose the K2vx's notes, as well as notes on slaves receiving from the K2vx. This transposition setting is not overridden when you use Setup mode, but is *added* to the transposition settings for the currently selected setup.

### **Control**

Here you determine where the K2vx sends MIDI information. A value of MIDI sends the MIDI signal to the K2vx's MIDI Out port, but not to the K2vx itself. This is also known as Local Control Off.

If you're using your K2vx with a MIDI sequencer and have the K2vx's MIDI Out connected to the sequencer's MIDI In, and the sequencer's MIDI Out connected to the K2vx's MIDI In a MIDI loop, you'll need to select a value of MIDI when your sequencer's Patch Thru feature (also known as Play Thru, Soft Thru, etc.) is on. This will prevent the K2vx's MIDI signal from looping back on itself, which can cause problems. If you deactivate your sequencer's Patch Thru feature, set the Control parameter's value to Both, and the K2vx will play normally. Also, you may want to set the value of the LocalKbdCh parameter to None when you have a MIDI loop, because you can have problems with doubled notes and MIDI overload. You won't have problems, however, as long as the channels transmitted by the K2vx are all different from the incoming MIDI channel.

A value of Local disables the MIDI Out port. Use this setting when you want to play the K2vx, but not to send any MIDI information to other MIDI instruments (local control only). A value of Both (the default) enables you to play the K2vx and send MIDI information from its MIDI Out port.

### **Program Change Type (PChgType)**

With this parameter you determine how the K2vx sends program change commands to its MIDI Out port. This does not affect the selection of K2vx programs from its own front panel. Whatever value you select for the PChgType parameter on the XMIT page will automatically be selected for the ProgChgType parameter on the RECV page. This will ensure that program change commands remain consistent.

The K2vx has special rules for interpreting program change commands, since it numbers its programs from 1 to 999, and normal MIDI program change commands range from 0 to 127 only. See "Program Change Formats" at the end of this chapter for an in-depth explanation of the Program Change Type parameter.

### **What value do I choose?**

The value you select for the PChgType parameter depends on your MIDI system. If you always change programs from the K2vx's front panel, you'll never need to change the value of this parameter. The K2vx knows how to respond to program change commands from its own front panel, regardless of the setting for this parameter.

If you're connected to another MIDI device and you want to exchange program change commands between them, the value you select depends on what the other MIDI device is. If it's an old MIDI device, select a value of 0–127 or QA 0–127. If it's a newer MIDI machine, and can handle the MIDI standard Controller 0 Controller 32 method of program changes, select a value of Extended or QA Bank E. Also select Extended or QA Bank E if it's another K2vx, (make sure the other K2vx is set the same way). If it's a Kurzweil K1200 or a 1000 Series with Version 5 software, select a value of Kurzweil. See "Program Change Formats" later in this chapter for explanations of how these settings work.

### **Velocity Map—Transmit (VelocMap)**

The transmit velocity map affects the way the K2vx sends velocity information to its MIDI Out port. Different maps generate different velocity values for the same attack velocity—that is, they apply different curves to the attack velocities the K2vx receives and remap them to new velocities before transmitting them to the MIDI Out port.

**Important:** The MIDI velocity maps affect only those MIDI velocity values transmitted via the K2vx's MIDI Out port, and are used exclusively to adjust the response of MIDI devices connected to the Out port. If you have a DX7 connected to your K2vx, for example, and the DX is distorting, selecting a transmit Velocity Map like "Hard2" should handle the problem. Changing the velocity map on this page will not affect the response of the K2vx's sound engine to your MIDI controller. That's done on the RECV page. See Chapter 17 if you're interested in editing velocity maps.

**Also important:** Both the transmit and receive velocity maps should be left at values of Linear unless you really need to change them. The linear maps will give you the most consistent results.

### **Pressure Map—Transmit (PressMap)**

This is like the VelocMap, but it controls the aftertouch values sent by the K2vx to its MIDI Out port. Use this exclusively to adjust the response of MIDI devices connected to the K2vx's MIDI Out port. Changing the pressure map on this page will not affect the response of the K2vx's sound engine to your MIDI controller. That's done on the RECV page. See Chapter 17 for information about editing pressure maps.

### **Pitch Bend (PBend)**

Set this parameter's value to On when you want the K2vx to send Pitch Bend messages to its MIDI Out port when you move your MIDI controller's Pitch Wheel. A value of Off disables Pitch Bend transmission.

### **Program Change (PChng)**

When On, the K2vx will send program change commands to its MIDI Out port when you select programs or setups from the front panel or from your MIDI controller. Select a value of Off when you want to change programs on the K2vx but don't want to send program change commands to the MIDI Out port. This parameter doesn't affect the type of program change command that's sent; it just determines whether any command is sent at all.

### **Buttons (Bttns)**

If you set the value of the Buttons parameter to On, the System Exclusive (SysEx) messages generated by your button presses are sent to the MIDI Out port. This enables you to do two things: control a remote K2vx (or K2500), and record sequences of programming button presses to a sequencer or SysEx software package.

If you have the MIDI In port of another K2vx (or K2500) connected to the first one's MIDI Out port, the second K2vx (or K2500) will respond to every button press on the first K2vx (or K2500), just as if you were pressing the buttons of the second one. Keep in mind that both devices must be in exactly the same state (the same page in the same mode, with identical lists

of RAM objects) when you start. Otherwise the button presses you make on the first device may execute other functions on the second device.

Much more useful is to send streams of button presses to your sequencer. When you dump them from your sequencer back to the K2vx, the K2vx responds as if the buttons were actually pressed. This enables you to set up a variety of "macros," which are strings of commands that can be executed all at once by a single initial command. For example, you can record a sequence of button presses that enters Disk mode, selects a specific SCSI device, and loads one or more banks of samples while you do something more entertaining. Again, it's important to keep in mind that the state of your K2vx must be identical to its state when you recorded the sequence of button presses. If you've added or deleted any objects stored in RAM, for example, the sequence of button presses will select different objects when you play back the button press sequence.

Note: Make sure this parameter is set to Off before you initiate a SysEx dump of any kind. If this parameter is On when you start a dump, the buttons you press to begin the dump will also generate SysEx messages, which will probably corrupt the SysEx dump.

### **Volume (V)**

This parameter acts as a global override to the "V" parameter on the Setup Editor page (described on page 7-4). When the "V" parameter is off, the K2vx will not transmit initial volumes when Setups are selected.

## **Physical Control Parameters**

### **Mod Wheel, Foot Switches 1 and 2, Continuous Control Pedal (CPedal), Controller Slider (Slider), and Mono Pressure (Press)**

The remaining parameters on the MIDI XMIT page are relevant only when the value for the Local Keyboard Channel parameter on the MIDI mode RECV page matches the transmitting channel of your MIDI controller. In that case, these MIDI control messages, when received by the K2vx, are remapped to the controller numbers defined by the values for these parameters, and sent to the MIDI Out port. See the discussion of the Local Keyboard Channel parameter in the next section for a full explanation.

The settings for Mod Whl, FtSw1, FtSw2, CPedal, Slider, and Pressure are overridden when you enter Setup mode or select a setup in Quick Access mode. The values on the XMIT page don't change, but the K2vx uses the currently selected setup's MIDI transmit settings instead. The MIDI mode settings take over again when you return to Program mode, or select a program in Quick Access mode.

You may never need to change the default settings for these parameters, but doing so may enable you to increase the flexibility of your MIDI system if you have MIDI devices connected to the K2vx's MIDI Out port.

## The Receive (RECV) Page

Press RECV to select the Receive page, where you define the K2vx's response to incoming MIDI signals.

```

MIDI MODE RECEIVE
BasicChannel: 1 SysExID: 0
MIDI Mode : Multi SCSI ID: 6
AllNotesOff : Normal BendSmooth: On
ProgChgType : Extended LocalKbdCh: None
VelocityMap: 1 Linear BankSelect: Ctl 32
PressureMap: 1 Linear PowerMode : User
XMIT RECV CHANLS ProgChg RsetOr Panic

```

PARAMETER	RANGE OF VALUES	DEFAULT
BASIC CHANNEL	1 to 16	1
MIDI MODE	Omni, Poly, Multi	Multi
ALL NOTES OFF	Normal, Ignore	Normal
PROGRAM CHANGE TYPE	(See below)	Extended
VELOCITY MAP	Velocity Map list	1 Linear
PRESSURE MAP	Pressure Map list	1 Linear
SYSTEM EXCLUSIVE ID	0 to 126	0
SCSI ID	0 to 7	6
BEND SMOOTH	On, Off	On
LOCAL KEYBOARD CHANNEL	None, 1 to 16	None
BANK SELECT	0 only, 32 only, Ctl 0, Ctl 32	Ctl 32
POWER MODE	User, Demo	User

### Basic Channel

The basic channel determines which channel will always be available to receive MIDI information. Depending on the MIDI receive mode (below), the Basic channel may be the only receiving channel, or one of several.

### MIDI Receive Mode (MIDI Mode)

The MIDI Mode parameter determines the MIDI receiving capabilities of the K2vx. When set to Omni, the K2vx responds to incoming MIDI events on all MIDI channels, and plays them on the current channel. This is normally used for diagnostic purposes only.

At a setting of Poly, the K2vx will respond only to events that are sent on the same channel as the K2vx's current MIDI channel (the one displayed on the top line of the Program mode page). In Poly mode, the currently selected channel is always the basic channel, so if you change channels, the basic channel changes accordingly.

With a value of Multi (the default), the K2vx will respond to events on all active channels. This is the mode you'll use when you're driving the K2vx with a sequencer, since you can play a different program on each channel. At this setting, you can turn individual channels on and off (on the CHANLS page, described later in this chapter).

### All Notes Off

If this parameter's value is set to Normal, the K2vx will respond to All Notes Off messages received over MIDI. Set to Ignore, these messages will be ignored. If you're using a Roland product as a MIDI controller for your K2vx, you'll want to set the value of this parameter to Ignore. This is because Roland products occasionally send an All Notes Off message even when notes should normally be sustained. You might find all your sustains missing from your

sequence, for example, if you're driving your K2vx from one of Roland's hardware sequencers. Setting this parameter to Ignore will take care of this problem.

Regardless of the setting for this parameter, the K2vx will always respond to its own Panic button by shutting off all active notes and controllers.

### **Program Change Type (ProgChgType)**

This determines how the K2vx will respond to program change commands received via MIDI. The value of this parameter automatically matches the value of PChgType on the XMIT page; changing it on one page will change it on the other. See "Program Change Formats" at the end of this chapter for an explanation of the various values available for this parameter.

### **Velocity Map—Receive**

The velocity map applies a preset curve to incoming velocity messages. It maps incoming velocity levels to new levels that correspond to the eight dynamic levels used by the VTRIGs and keymaps for velocity level selection. See Chapter 17. Normally you'll leave this set to 1 Linear. Adjust this parameter's value only when you need to alter the K2vx's response to the velocity messages from your MIDI controller, for example, if you're getting too much or too little volume when you play, or when you're using a sequencer to drive the K2vx.

### **Pressure Map—Receive**

Like the velocity map, this determines how the K2vx responds to incoming pressure (aftertouch) messages.

### **System Exclusive ID (SysExID)**

The SysExID parameter differentiates between more than one MIDI device of the same model. You won't need to change the default setting of 0 unless you have multiple K2vxs receiving SysEx messages from a single source. In that case, make sure each K2vx has a different SysExID. Then you can direct SysEx messages to the appropriate K2vx with the SysExID byte that's included with every SysEx message. A value of 127 specifies "Omni Receive." That is, at this value, a K2vx will respond to a SysEx message regardless of the SysEx ID of the message (as long as the manufacturer and device IDs match—see the *Reference Guide* for more information on System Exclusive messages).

### **SCSI ID**

Use this parameter to change the SCSI ID of your K2vx. You can ignore this parameter unless you've connected a SCSI device (external SCSI disk or CD-ROM drive) to the K2vx's SCSI port. You can use either or both SCSI ports to chain up to seven SCSI devices to the K2vx (a total of eight devices can be chained together); just be sure to set each one to a different SCSI ID. Most SCSI devices available today make it easy to change their SCSI IDs, so you may not have to adjust this parameter even if you have several SCSI devices connected. See Chapter 13 for more information about using SCSI devices.

### **Bend Smooth**

This parameter can improve your K2vx's performance when you're driving it from a MIDI guitar controller. Its default value is On.

You may find that pitch bending seems to carry over from the previous note to the next note, causing it to start on the wrong pitch. This is probably due to the automatic pitch smoothing provided by the K2vx. If this is happening, try setting the BendSmooth parameter to a value of Off.

### **Local Keyboard Channel (LocalKbdCh)**

Note: You won't generally want to change the setting of the Local Keyboard Channel parameter unless you are using the rack version of the K2vx.



The available values for this parameter are None, and 1–16. The default is None, which disables the local keyboard feature, since you may not want to send your MIDI controller's MIDI information to devices connected to the K2vx's MIDI Out port. If you want to use Setup mode on the K2vx, however, you will have to change the setting of this parameter. This is because LocalKbdCh enables you to take advantage of the K2vx's three setup zones, even if your MIDI controller transmits on only one MIDI channel at a time.

It is important to understand that a Setup is a control oriented function. On the keyboard version of the 2000, the keyboard itself will transmit on up to three channels when in Setup mode. But with the rack, if your keyboard only sends information on one MIDI channel, you need a way to turn that information on one channel into three channels. This is what the Local Keyboard Channel parameter does. It takes the signal coming in one channel and turns it into different information, depending on where you are in the 2000.

The Local Keyboard Channel will change the way the 2000 performs in other modes as well. It changes the incoming information depending on what you have displayed in the 2000. For instance, if you are in Program Mode with Channel 5 in the display, then the information coming in on channel 1 will be turned into channel 5 and you will hear the program assigned to channel 5. But if you turn Local Keyboard Channel off by setting it to None, then if you send on channel 1, you will hear the program that is assigned to channel 1, even if you are looking at channel 5.

Local Keyboard does more than just change the MIDI channel. You will notice that the Octave+ and - soft Buttons now will transpose the information. And you can even use it to change one type of MIDI controller to another.

Here's how it works. The K2vx will receive MIDI information on the channel that corresponds to the value you set for this parameter, and will relay it to its MIDI Out port, using the MIDI channels currently shown in the display. If you're in Program mode (or in Quick Access mode with a program selected), the K2vx will relay the LocalKbdCh MIDI information to the channel to which the program is assigned. If you're in Setup mode (or in Quick Access mode with a setup selected), the K2vx will relay the LocalKbdCh MIDI information to all the channels currently shown in the display.

The K2vx will also remap the six controller messages found on the MIDI XMIT page (and in the Setup Editor). This enables you to receive Modulation (01), Foot (04), Data (06), Mono Pressure, Sustain (64), and Sostenuuto (66) messages from your controller and remap them to any available value on the K2vx's Control Source List. The chart below will clarify.

<b>The control message received from your controller...</b>	<b>...gets sent to the MIDI control number assigned as the value for this parameter on the MIDI XMIT page</b>
<b>Modulation (01)</b>	<b>ModWhl</b>
<b>Sustain (64)</b>	<b>FtSw1</b>
<b>Sostenuuto (66)</b>	<b>FtSw2</b>
<b>Foot (04)</b>	<b>CPedal</b>
<b>Data (06)</b>	<b>Slider</b>
<b>Mono Pressure</b>	<b>Press</b>

The controllers listed in the second column correspond to the control source parameters on the MIDI XMIT page and in the Setup Editor. When you're in Program mode, the values that you've set for these six parameters on the MIDI mode XMIT page determine what control message gets sent to the K2vx and to its MIDI Out port when the K2vx receives the control messages in the first column. When you're in Setup mode, the values for these same six parameters on the Setup Editor page determine which control signal is sent.

Here's a more specific example. Suppose your MIDI controller transmits on MIDI channel 1, and you've set the LocalKbdCh parameter to a value of 1. You've also set the value of the ModWhl parameter on the Setup mode page to a value of Volume (MIDI 07) for each setup zone. Then you've selected a setup that uses MIDI channels 1, 2, and 3. When you send a Modulation message (MIDI 01) from the MIDI controller, you'll affect the K2vx's volume (unless the VolLock parameter for Channel 1 is on), and the K2vx will send a Volume message to its MIDI Out port, on channels 1, 2, and 3.

### **Bank Select**

"BankSelect" allows you to choose between having the K2vx respond to Controller 0 or Controller 32 or both. The reason for this is that various manufacturers have chosen one method or the other. The four possible values for this parameter are:

0 only - transmits and responds to controller 0 only.

32 only - transmits and responds to controller 32 only.

Ctl 0 - transmits 0 and responds to 0 or 32.

Ctl 32 - transmits 32 and responds to 0 or 32.

### **Power Mode**

"Power Mode" has two possible values: User and Demo. When set to User, the user's parameter settings are retained. When set to Demo the parameters are returned to default values when the unit is powered up. The default value for the Power Mode parameter is User.

The following parameters are reset when Power Mode is set to Demo; default values for the parameters are shown in parentheses.

Master Mode - Drum Channel (1)

MIDI Transmit - Control (Both)

Effects Mode - FX Mode (Auto)  
FX Chan (Current)

Disk Mode - Current Disk (Floppy)

The unit also remembers what channel you were on and any octave transpose value (it used to return to channel 1 and 0 transpose).

## **The Channels Page**

Press the CHANLS soft button to select the CHANLS page, where you can define numerous parameters for each MIDI channel independently. Use the CHAN/BANK buttons to select the MIDI channel you wish to work on.

The CHANLS page is very useful when you're doing multi-timbral sequencing, with programs assigned to numerous MIDI channels. The CHANLS page lets you set several control characteristics for each MIDI channel. This makes it easy to adjust the playback of the sequence without editing the sequence itself. For example, you might turn off the Enable parameter for one or more channels to mute the tracks on those channels. You could also set the VolLock parameter to On, to ignore any MIDI volume messages the K2vx receives on a given MIDI channel.

```

MIDI Mode: CHANNELS <> Channel: 1
Enable : On
Program : 1 Grand Piano PrgLock: Off
Pan : 64 PanLock: Off
Volume : 127 VolLock: Off
OutPair : Prog
OutGain : Prog
XMIT RECV CHANLS ProgChg RsetOf Panic

```

PARAMETER	RANGE OF VALUES	DEFAULT
ENABLE	Off, On	On
PROGRAM	Program list	Program ID 1
PAN	0 to 127	64 (centered)
VOLUME	0 to 127	127 (maximum)
OUTPUT PAIR	Prog, A (FX), B (DRY), C(DRY), D(DRY)	Prog
OUTPUT GAIN	Prog, -12 to 30 dB in 6 dB increments	Prog
PROGRAM LOCK	Off, On	Off
PAN LOCK	Off, On	Off
VOLUME LOCK	Off, On	Off

### Enable

Use this parameter to turn the currently selected channel on or off. When on, the channel will receive MIDI information, and the settings of the parameters on the MIDI CHANLS page will be in effect. When off, the channel will ignore all MIDI information.

### Program

Use this parameter to assign a program to the currently selected channel. The channel will still respond to program change commands received via MIDI, unless the PrgLock parameter (described below) is set to On.

### Pan

This will offset the pan position of the current program as set on the OUTPUT page in the Program Editor. A value of 0 is maximum offset to the left, 64 is no offset, and 127 is maximum offset to the right. Changing the value of this parameter is like inserting a MIDI pan message. MIDI Pan (MIDI 10) messages will change the value of this parameter, unless the PanLock Parameter (described below) is set to On.

### Volume

This sets the volume for any program assigned to the currently selected channel. A value of 0 is silence, and a value of 127 is full volume. The value of this parameter will change in response to MIDI Volume (MIDI 07) messages, unless the VolLock parameter (described below) is set to On.

### Output Pair (OutPair)

This parameter sets the audio output group for the program assigned to the currently selected channel. The default value of Prog means that the output group is determined by the program's value for the Pair parameter on the OUTPUT page in the Program Editor. In this case, the channel's output group will change depending on the program assigned to it, with the output group being routed on a per layer basis within the program. Values of A, B, C, or D fix the output group regardless of the program that's assigned to the channel.

**Output Gain (OutGain)**

OutGain will boost or cut the level at the audio outputs for any program assigned to the currently selected channel. This allows you to make a program louder or softer without having to edit the program.

**Parameter Locks****Program (PrgLock), Pan (PanLock), Volume (VolLock)**

When the parameter locks are set to On, the three parameters they control will not respond to their respective MIDI controller messages. In that case, you could change the Program, Pan, and Volume settings from the front panel, but not via MIDI.

**Program Change Formats**

Straight out of the box, the K2vx has more programs than the MIDI program change specification can handle (MIDI lets you send program change numbers from 0 to 127 or 1 to 128 only). So we've designed a system that makes program selection more flexible. This is true whether you're selecting programs from the K2vx's front panel, or via MIDI.

<b>Program Change Type</b>	<b>For use with:</b>
0–127	Older MIDI devices that accept program change commands in the range from 0–127 only.
Extended	Other K2vxs (or K2500s) similarly set, plus all other instruments that use the Bank Change controller
Kurzweil	K1200s, and 1000s with version 5 software
QA Bank E	Other K2vxs (or K2500s) similarly set, when in Quick Access mode
QA Bank K	K1200s and v5 1000s, when in Quick Access mode
QA 0–127	With the K2vx in Quick Access mode, when using it with older MIDI devices

First of all, the K2vx's programs (and all of its objects) are numbered and grouped according to a decimal system, that is, in multiples of ten. This is much easier to keep track of than the binary-oriented groupings of many synths, which feature banks of 8, 16, or 64 programs.

Next, the K2vx gives you 999 program change numbers to work with. These are organized into ten banks of 100 each (the memory banks). A program's object ID is its program change number, as discussed in Chapter 5. This makes it easy to keep track of your programs. The K2vx can use several different formats for interpreting Program change commands. The values for the program change Type parameters on the XMIT and RECV page determine which format is used, and the one you should select depends on your MIDI system.

If you expect you'll always change programs from your K2vx's front panel, you can finish this paragraph and skip the next few sections. In this case, selecting programs is as simple as entering the program change number (the program's object ID) on the alphanumeric pad, and pressing ENTER. Even program numbers above the usual MIDI limit of 127 can be selected this way.

### 0-127 Program Change Type

The next simplest system involves connecting your K2vx to an “old” MIDI device—one that was built before the MIDI Controller 0 program change format was developed. If your MIDI controller is one of these (if its manual doesn’t mention MIDI Controller 0 program changes, it’s an “old style” machine), you might want to set the program change Type parameter to a value of 0–127. This will enable you to select programs 0–127 from the controller. This limits your range of program selection, but it configures the K2vx to respond predictably to the controller. (You’ll have to select higher-numbered programs from the K2vx’s front panel) Of course, you can still use the Extended format explained below, but in many cases you’ll have to send two program change commands to get the program you want.

Note: you can define how the K2vx will respond to and transmit MIDI Controller commands using the Bank Select parameter, described on page 10-8.

## Extended and Kurzweil Program Change Types

In the early days of MIDI, most instruments had small numbers of memory locations, usually 32, 64, or 128. As instruments began to have more memory locations, however, users ran against the limitation of only 128 values for program changes in the MIDI spec. Because of this, Bank Change Controller was added, allowing users to switch between banks of up to 128 programs per bank.

Previous to the addition of the Bank Change Controller, Kurzweil had developed their own method of switching banks by using two program changes, one to switch the bank, the second to call up the program within the bank (as described below). The K2vx can respond to either the Bank Change controller or the double program change method. In a nutshell, the difference between the Extended setting and the Kurzweil setting is this: In Extended, the 2000 will receive and respond to both the Bank Change controller and double program change method, but it will always transmit the Bank Change controller. When set to Kurzweil, the 2000 will receive and transmit only the double program change method.

### Extended Program Changes

If you’re connected to a MIDI device that can handle the MIDI Controller 0 program change format, your flexibility increases considerably. We’ll discuss first how the K2vx receives program change commands in this format, then how it transmits. In a system of this sort, you’ll set the program change Type parameter to a value of Extended (or QA Bank E, but that explanation comes later).

First, the receiving end. When you’re using the extended program change format, the K2vx will respond to either MIDI controller 0 program change commands, or standard program change commands, but only within certain ranges, as follows:

Program change command type	Range of values	Result
MIDI controller 32 (MC 32)	0 to 9	Selects memory bank 0s–900s
	10 to 127	Ignored
Standard (PCH)	0 to 99	Selects correspondingly numbered program in current memory bank
	100 to 109	Selects memory bank 0s–900s
	110 to 127	Ignored

If your K2vx is already in the memory bank you want to use, you can send it single PCHs from 0 to 99, to select programs within that memory bank. If you want to change the memory bank, the K2vx must receive either an MC 32 message with value 0–9, or a PCH with value 100–109. This will select the new memory bank, but will not change the current program. The next PCH in the range 0–99 will select the correspondingly numbered program in the newly selected bank. The following table of examples should help make it clear.

<b>1st program change command received:</b>	<b>2nd program change command received:</b>	<b>Result:</b>
PCH: value 64	None	64th program in current bank selected (e.g. program 264 if in 200s bank)
PCH: value 99	PCH: value 27	27th program in current bank selected (99 is selected, then overridden by 27)
PCH: value 102	PCH: value 16	Program 216 (200s bank, 16th program)
PCH: value 127	PCH: value 99	99th program in current bank (1st PCH is ignored, since it's above 109)
PCH: value 127	PCH: value 104	No change in current program; 400s bank selected pending next PCH
MC 32: value 0	PCH: value 99	Program 99 (0s bank, 99th program)
MC 32: value 1	PCH: value 42	Program 142 (100s bank, 42nd program)
MC 32: value 9	PCH: value 0	Program 900 (900s bank, 0th program)
MC 32: value 9	None	900s bank selected, no change in current program (bank selection is pending for next PCH)
MC 32: value 10	PCH: value 99	MC 32 message ignored; 99th program in current bank selected (e.g. program 199 if in 100s bank)

That's the receiving end of extended program changes. On the transmitting side, the rules are similar; if you select a program in the current or another memory bank, two commands are sent. The first selects the memory bank, and is always an MC 32 type. The second is always a PCH. Some examples follow. They assume you use the alphanumeric pad, but you could use the other data entry methods as well.

<b>Front panel selection:</b>	<b>Program change commands sent:</b>
9 9 ENTER	MC 32: value 0; PCH: value 99
2 1 6 ENTER	MC 32: value 2; PCH: value 16
9 1 1 ENTER	MC 32: value 9; PCH: value 11

## Kurzweil Program Changes

When you use the Kurzweil program change format, the rules are similar to the extended format, but when two Program change commands are sent, the first is necessarily of the PCH type. The receiving end works as follows:

1st program change command received:	2nd program change command received:	Result:
PCH: value 39	None	39th program in current bank selected
PCH: value 99	PCH: value 27	27th program in current bank selected (99 is selected, then overridden by 27)
PCH: value 102	PCH: value 16	Program 216 (200s bank, 16th program)
PCH: value 105	PCH: value 44	Program 544
PCH: value 109	PCH: value 0	Program 900
PCH: value 127	PCH: value 99	99th program in current bank (1st PCH is ignored, since it's above 109)
PCH: value 127	PCH: value 104	No change in current program; 400s bank is selected pending next PCH

When you *send* Kurzweil program changes via MIDI, the K2vx always sends two PCHs. A few examples:

### Front panel selection: program change commands sent:

2 7 ENTER	PCH: value 100, PCH: value 27
9 9 ENTER	PCH: value 100, PCH: value 99
2 1 6 ENTER	PCH: value 102, PCH: value 16
9 1 1 ENTER	PCH: value 109, PCH: value 11

## Quick Access Banks—Extended (QA BANK E)

Using this setting is similar to using the Extended program change format, but it goes one step further. When receiving, the incoming program change commands are interpreted just as they are in the normal Extended format. But the resulting program change number, instead of selecting a program, selects a Quick Access bank entry (you must be in Quick Access mode for this to work). There are two advantages to using this format. First, it allows you to select both programs and setups using program changes, without having to switch modes. Second, you can remap incoming program change commands to select programs or setups with different IDs. This is handy if the sending unit can't send program change commands higher than 127.

First, a brief review of Quick Access bank structure. Each Quick Access bank can store ten entries, each of which can be a program or a setup. Each of the K2vx's 10 memory banks can store 20 Quick Access banks (except the Zeros bank, which can store 75). Therefore when you're in Quick Access mode, you have access to 200 (or 750 in the Zeros bank) programs or setups without leaving the currently selected bank. The QA Bank E program change format lets you select any one of those programs or setups via MIDI. If you select another memory bank, you have a different set of 200 programs and setups at your disposal. The programs and setups aren't selected by their IDs, as they normally would be. They're selected in terms of their location within the Quick Access banks.

We'll discuss the receiving side first. When you're using this format, the K2vx will respond to either MC 32s or PCHs. The acceptable ranges of values are different, however, as is the interpretation the K2vx makes. Instead of responding by selecting programs, the K2vx responds by selecting entries within the currently selected Quick Access bank. These selections are made according to their "chronological" listing within the QA bank without regard for the IDs of the objects in the QA bank. In most of the *memory* banks, there are 20 QA banks of 10 entries each, for a total of 200 entries. The program change values the K2vx receives correspond to the sequential numbering of the entries in the QA banks.

Program change command type	Range of values	Result
MIDI controller 32 (MC 32)	0 to 7	Selects QA bank 0n, 1n, 2n, 3n, 4n, 5n, 6n, 7n in current memory bank
	10–127	Ignored
Standard (PCH)	0–99	Selects last digit (n above) of QA bank, and entry within that bank
	100–107	Selects QA bank 0n, 1n, 2n, 3n, 4n, 5n, 6n, 7n in current memory bank
	110–127	Ignored

Depending on the QA bank entry you want to select, you'll send the K2vx one or two program change commands. Sending a single command will let you select from a range of 10 QA banks and select an entry within that bank (see the table below). To select a different range of QA banks, send two program change commands.

If you send a single command, it should be a PCH with a value of 0 to 99. If you send two commands, the first can be either an MC 32 with value 0 to 7, or a PCH with value 100 to 107. The second command should be a PCH with a value of 0 to 99.

Within the QA Bank E format, the first program change command specifies the range of Quick Access banks that will be selected. It does **not** select a different memory bank. In fact, you can't change the memory bank via MIDI when using this format. All program and setup selections are made within the currently selected memory bank. You'll know which memory bank is selected by looking at the ID of the currently selected Quick Access bank in the top line of the Quick Access mode page. Several examples follow.



**If the Zeros Memory Bank is Currently Selected**

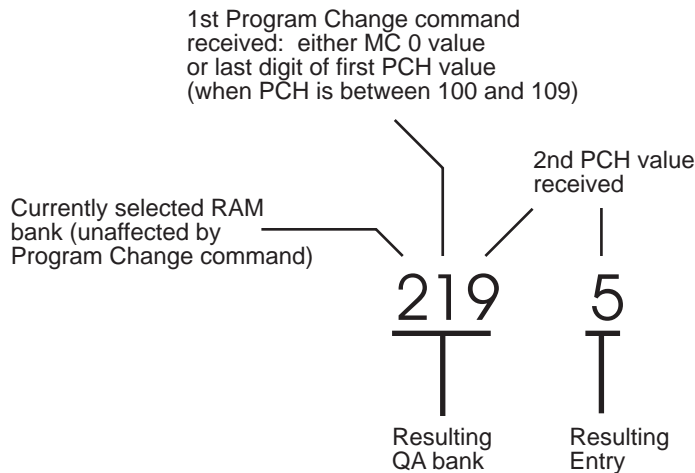
<b>1st program change command received:</b>	<b>2nd program change command received:</b>	<b>Resulting selection:</b>
MC 32: value 0	PCH: value 6	No change (the K2vx interprets this as QA bank 0, entry 6. There is no QA bank 0. The lowest valid PCH value in this particular case is 10, which would select QA bank 1, entry 0)
PCH: value 9	None	Entry 9 in current QA bank
MC 32: value 0	PCH: value 32	QA bank 3, entry 2
MC 32: value 1	PCH: value 4	QA bank 10, entry 4
MC 32: value 1	PCH: value 28	QA bank 12, entry 8
MC 32: value 2	PCH: value 44	QA bank 24, entry 4
PCH: value 100	PCH: value 9	No change (QA bank 0 doesn't exist)
PCH: value 100	PCH: value 99	QA bank 9, entry 9
PCH: value 102	PCH: value 27	QA bank 22, entry 7

Remember that in the Zeros memory bank, the Quick Access bank IDs go through 75. So if the Zeros memory bank is the current memory bank, you can enter MC 32 values as high as 7, and PCH values as high as 107 for the first program Change command. And you can enter PCH values as high as 59 for the second program change command.

**If the 200s Memory Bank is Currently Selected**

<b>1st program change command received:</b>	<b>2nd program change command received:</b>	<b>Resulting selection:</b>
MC 32: value 0	PCH: value 6	QA bank 200, entry 6
MC 32: value 0	PCH: value 32	QA bank 203, entry 2
MC 32: value 0	PCH: value 99	QA bank 209, entry 9
MC 32: value 1	PCH: value 4	QA bank 210, entry 4
MC 32: value 1	PCH: value 28	QA bank 212, entry 8
MC 32: value 2	PCH: value 44	No change; MC 32 value 2 is invalid in 200s bank.
PCH: value 44	None	QA bank 204, 214, 224, 234, 244, 254, 264, or 274 (tens digit doesn't change); entry 4
PCH: value 100	PCH: value 0	QA bank 200, entry 0
PCH: value 100	PCH: value 99	QA bank 209, entry 9
PCH: value 100	PCH: value 127	No change; PCH value 127 is invalid for QA Bank E format
PCH: value 101	PCH: value 8	QA bank 210, entry 8
PCH: value 101	PCH: value 36	QA bank 213, entry 6
PCH: value 102	PCH: value 27	No change; PCH value 102 is invalid in 200s bank

There are two more ways to describe the QA Bank E format: one verbal, one visual. The first (hundreds) digit of the QA bank that gets selected is always the same as the currently selected memory bank. The QA bank's second (tens) digit is equal to the value of the first program change command: either the value of the MC 32 message, or the third (ones) digit of the PCH. The QA bank's third (ones) digit is equal to the tens digit of the second PCH. The QA bank entry that gets selected is equal to the ones digit of the second PCH. The following diagram will clarify:

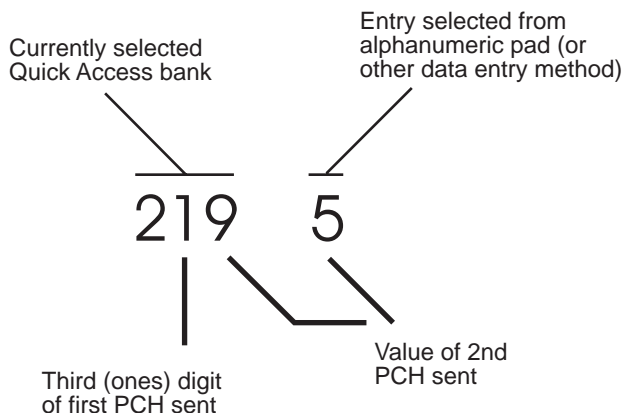


That takes care of the receiving end of the QA Bank E format. The diagram above also explains the transmitting side. If, for example, you're currently in Quick Access bank 219, and you press 5 on the alphanumeric pad, the K2vx selects for itself whatever program or setup is programmed for that bank and entry. The K2vx then sends the following two program change commands to its MIDI Out port—MC 32: value 1, followed by PCH: value 95. The first program change command is always of the MC 32 type. The currently selected memory bank is not included in the program change command.

### Quick Access Banks—Kurzweil (QA BANK K)

In terms of receiving program change commands, this works almost exactly like the QA Bank E format. The only exception is that within the QA Bank K format, the K2vx expects the first program change command to be of the standard program change command type. MIDI Controller 32 messages will not be recognized.

The transmitting side of this format is similar to the QA Bank E format as well, as the following diagram shows. The banks and entries you select are translated into two standard program change commands sent in rapid succession.



Current QA Bank	Entry from alphanumeric pad	Program change commands sent	
1	0	100	10
1	9	100	19
2	0	100	20
2	9	100	29
9	9	100	99
10	0	101	0
19	9	101	99
20	0	102	0
29	9	102	99
75	9	107	59
100	0	100	0
105	9	100	59
110	9	101	99
119	9	101	99
48	4	104	84
117	7	101	77

### QA 0–127

Finally, there's the QA Bank format for use with older MIDI devices (Program change commands 0–127 only). It works similarly to the other QA formats, but the allowable range of values is limited to 0–107. The K2vx expects to receive PCHs of value 0–99 to select a bank and entry, or a pair of PCHs, the first having a value of 100–107 to select a different 10-bank range.

When you select an entry with the alphanumeric pad while in QA mode, the K2vx sends two PCHs, the first with a value of 100–107, the second with a value of 0–99.

## The Soft Buttons in MIDI Mode

The first three soft buttons select the three MIDI mode pages. The PrgChg soft button lets you send a program change command on any MIDI channel. The RsetCh soft button lets you return all channel parameters to their default values. The Panic soft button sends an All Notes Off and an All Controllers Off messages to the K2vx and on all 16 MIDI channels.

### Program Change (PrgChg)

When you press this soft button, a dialog appears:

```
Send Program Change:
On Channel  1, Send Program  0

Chan-  Chan+  Prog-  Prog+  Send  Cancel
```

This dialog lets you send program changes out the MIDI Out port, but does not change internal programs.

The CHAN/BANK buttons, the Up/Down cursor buttons, and the Chan- and Chan+ soft buttons can all be used to change the channel on which the program change command will be sent. The Left/Right cursor buttons, the Plus/Minus buttons, the Alpha Wheel and the Prog- and Prog+ soft buttons can all be used to change the program change number that will be sent. When you've set the channel and the program change number, press the Send soft button to send the program change command. Or press the Cancel soft button if you don't want to send it.

You can change the channel and the program number as many times as you want before you press **Send**. You also can use the alphanumeric pad to select a program number directly.

### Reset Channels (RsetCh)

When you press this soft button, the K2vx will ask you if you want to reset all channels, and a pair of Yes/No soft buttons will appear. If you press the Yes soft button, all settings on the CHANLS page will return to their default values. For example, you may have set several MIDI channels to route their audio to Output Group B for a special project. When the project's over, you can reset the Channels to restore the audio routing to each individual program (a value of Prog), rather than selecting each channel's page and setting the Pair parameter back to a value of Prog. Press the No soft button if you decide not to reset the channels.

### Panic

This soft button sends an All Notes Off and All Controllers Off message both to the K2vx and over all MIDI channels.

# Chapter 11

## Master Mode

Press the Master mode button to enter Master mode, which contains parameters affecting the K2vx's overall performance.

### The Master Mode Page

On the Master mode page you'll find parameters for setting the overall tuning and transposition of the K2vx, the MIDI channel to be used for drum programs (explained below), and for several keyboard, audio, display, and programming adjustments. You can also enter the sampler from the Master mode page

```

MasterMode: Samples:8192k Memory:121k
Tune       :0ct          OutA->Mix: Stereo
Transpose  :0ST         OutB->Mix: Stereo
DrumChan   :1          OutA->FX : L+R
VelTouch   :1 Linear    Contrast  :0
PressTouch :1 Linear    Confirm   :Off
Intonation :1 Equal     IntonaKey:C
Object     Delete      Util      Sample Panic Reset

```

PARAMETER	RANGE OF VALUES	DEFAULT
TUNE	± 100 cents	0
TRANSPOSE	± 60 semitones	0
DRUM CHANNEL	1 to 16	1
VELOCITY TOUCH	Velocity Map list	1 Linear
PRESSURE TOUCH	Pressure Map list	1 Linear
INTONATION	Intonation Table list	1 Equal
OUT A PAN MODE	Mono, Stereo	Stereo
OUT B PAN MODE	Mono, Stereo	Stereo
OUT A EFFECTS POSITION	L Only, L + R	L + R

#### Tune

Adjusting the value of this parameter will tune every program in the K2vx by the amount you specify. Tuning can be adjusted up or down 100 cents (one semitone) in one-cent increments. This parameter is useful for tuning up with recordings and acoustic instruments. Adjusting the tuning in Master mode does not change the settings on the PITCH page of individual programs, but will be added to any adjustments you make there. Master mode tuning adjustments affect only the K2vx's notes.

#### Transpose

Like the Tune parameter above, Transpose will affect every K2vx program, but not those notes sent to the MIDI Out port. You can adjust the MIDI transposition on the XMIT page in MIDI mode.

#### Drum Channel

Most K2vx programs have a maximum of three layers, which is enough to create just about any sound you can think of, especially when that sound covers most or all of the keyboard. But what do you do if you want to create a program with a dozen or more different sounds across the range from C 0 to C 8, and you want to be able to run each sound independently through a set of DSP functions?

Drum programs allow you to do exactly that. They can have up to 32 layers, and each layer has all the programming features of regular programs—its own keymap, its own algorithm, etc. We call

them drum programs because a likely application for this type of program is to create drum kits with many different percussion timbres. You can assign any keymap to any layer, however, to create as complex a program as you want. All those extra layers create a lot of information for the K2vx's sound engine to process. Every bit of information that's sent to the sound engine is associated with one of the MIDI channels. The K2vx's sound engine has enough processing power to handle three layers' worth of information from 8 channels, and 32 layers' worth from the other 8. By default, MIDI channels 1-8 are defined as drum channels.

If you'd like, you can use the DrumChan parameter to define a channel numbered higher than 8 to be a drum channel. In this case, the channel you choose, along with channels 1-7 will be your drum channels. For example, if you set DrumChan to 13, then your eight drum channels will be 1, 2, 3, 4, 5, 6, 7, and 13. If you set DrumChan to a value of 8 or less, however, your drum channels will always be 1, 2, 3, 4, 5, 6, 7, and 8.

Drum channels are the only channels that can have drum programs (any program with more than three layers) assigned to them. So when you're in Program mode, the current MIDI channel must be a drum channel if you want to assign a drum program to it. If you're in Program mode (or in Quick Access mode with a program selected) and the current MIDI channel isn't a drum channel, the drum program's name will be in parentheses and the box at the left of the Program mode display will tell you which channels are the drum channels. If you're in Setup mode (or in Quick Access mode with a setup selected) and the current MIDI channel isn't a drum channel, the drum program's name will be enclosed in parentheses. In either case, the program will not play. Setting the DrumChan parameter to match the current MIDI channel (or vice versa) will restore the drum program to normal operation.

### **Velocity and Pressure Touch**

These two parameters are only used by the keyboard version of the K2vx, and have no function in the K2vxR. If you are using a K2vxR, you should leave them set to their default values.

### **Intonation**

Most modern western music uses what is known as equal temperament. This means that the interval between each semitone of the 12-tone octave is precisely the same as every other interval. Many different intonation intervals have evolved over the centuries, however, and the K2vx supplies you with 17 different intonation "tables" to choose from. (There's also an eighteenth "table" listed, which we'll describe in a moment.) By changing the value for this parameter, you select from among the intonation tables stored in the K2vx's memory. Each of these tables defines different intervals between each of the semitones in a single octave.

Scroll through the list of Intonation tables, and listen for the differences between semitones. Some of the intervals between semitones may be quite different from equal intonation, but you'll notice that all notes are precisely tuned with notes that are an octave apart. This is because the intonation tables set the intervals within a single octave, and apply those intervals to each octave. If this doesn't make sense, the explanation of the Intonation Table Editor, in Chapter 17, will help clarify things. If you're hoping to create fully microtonal tunings by editing intonation tables—sorry, that's not possible. But you *can* create microtonal tunings using the Keymap Editor; see Chapter 15. There's a list of the intonation tables in Chapter 17; descriptions of each can be found in Chapter 5 of the *Reference Guide*.

### **Determining the version number of your ROM Objects (Intonation Table 18)**

As you're scrolling through the list of intonation tables, you may notice a listing for an eighteenth intonation table with a name such as "18 Obj *vn.n*". This isn't really another intonation table. Rather, this is where the K2vx stores the version number of your ROM objects. If you ever need to find out what version of ROM objects you've got loaded, this is where you look. Simply go to the Master page, then scroll the Intonation parameter until 18 is displayed. And don't forget to return to your correct intonation table when you've checked the version number of your ROM objects.

### **Outs A and B—Pan Mode**

The parameters OutA->Mix and OutB->Mix determine the panning of the audio signal for output groups A and B at the MIX outputs. If they're set to Stereo, then whatever panning is applied to each program (as set on the OUTPUT page in the Program Editor) will show up at the MIX outputs. If

either of these parameters is set to Mono, then programs routed to the corresponding audio output group will send all of their sound equally to both MIX outputs, regardless of any panning that might be applied to them on the OUTPUT page. This is a quick and convenient way to convert the MIX outputs into two mono outputs. Output groups C and D are permanently set to Stereo, and cannot be changed.

### **Out A Effects Position (Out A->FX)**

This parameter is intended for special cases when you want an additional separate output that doesn't go through the K2vx's effects processor. A value of L Only bypasses the effects processor for all sounds assigned to the right side of Output Group A.

There's more than one way to set this up; we'll give you one possible example that should get you started. You're going to play a multi-timbral sequence. You want to send your percussion sounds through an external effects box. You want your lead sound to go through an external fuzz box. You want the rhythm section to use the K2vx's internal effects. Finally, you want to send all the sounds through the K2vx's MIX outputs so you'll use only two inputs on your mixing board. This is done by using stereo insert cables in the separate outputs, which enables you to loop sounds through external devices, back to the K2vx, and through the MIX outputs. See Chapter 18 for more information about using stereo insert cables.

The percussion sounds are routed to the B outputs (this is done on the OUTPUT page in the Program Editor). You connect the stereo end of an insert cable in each of the K2vx's B outputs. The send sides (tips carry signal) of the insert cables are connected to the inputs of your effects box. The return sides (ring receives signal) are connected to the return jacks on the effects box. The percussion sounds will now appear at the K2vx's MIX outputs without going through the internal effects processor (sounds routed to Output Group B never go through the internal effects).

All the other sounds are routed to Output Group A. The lead sound is panned fully right, and the rhythm sounds are panned fully left. You connect an insert cable to the A Right output. The send side of the insert cable goes to the input of your fuzz box, and the return side goes to the fuzz box's return or output jack. The lead sound will now appear at the right side of the MIX outputs. The rhythm sounds will appear at the left side of the MIX outputs. At this point, the K2vx's internal effects will be applied to the lead sound as well as to the rhythm sounds. You don't want the internal effects applied to the lead sound, so you set the value of the Out A->FX parameter to L Only. Now the lead sound will be dry and the rhythm sounds will be wet.

You want the lead and rhythm sounds to be centered in the mix, so you set the Out A->Mix parameter to Mono. This sends both sounds at equal levels to both the MIX outputs, but the lead sound is still dry.

### **Contrast**

Use this parameter to adjust the contrast of the display, to adapt to different lighting conditions. The K2vx will remember your setting even when you power down.

### **Confirm**

Confirmations are special displays that the K2vx shows you when you are about to alter memory permanently. The confirmations ask if you really want to do what you're about to do, and give you another chance to cancel the operation you're about to execute. With the Confirm parameter set to Off, these prompts do not appear. You'll still be alerted before doing something that might cause you to lose your work.

### **Intonation Key (Intona Key)**

This sets the tonic, or base note from which the currently selected intonation table calculates its intervals. If you select G as the intonation key, for example, and the intonation table you select tunes the minor 2nd down by 50 cents, then G# will be a quartertone flat relative to equal intonation. If you change the intonation key to D, then D# will be a quartertone flat. If you use non-standard intonations, you'll want to change the intonation key as you change the key you're playing in.

You can also set the intonation key from an external MIDI device. Note On events at C -1 through B -1 (MIDI note numbers 0 through 11) will set the intonation key at C through B, respectively.

To trigger notes in the range required to set the Intonation key, you can transpose the K2vx temporarily from its front panel, or from your MIDI controller if it has the ability. Alternatively, you could create a setup with just the lowest octave transposed down two octaves, then select it when you want to change the Intonation key. If you're driving your K2vx from a sequencer, you could simply insert the appropriate note events anywhere in the sequence to change the intonation key.

## The Soft Buttons in Master Mode

### Object

This soft button brings up the object utilities. They're described beginning on page 11-6.

### Delete

This soft button brings up the file dialog, enabling you to erase sets of objects, either entire banks or all objects, from RAM. If the Confirm parameter on the Master mode page is set to a value of On, you'll be given an extra chance to cancel before the set of objects is actually deleted. Once the deletion is complete, the objects are irretrievable, so you may want to save objects to disk before deleting them from RAM.

To delete individual objects, use the functions that are available when you press the **Object** soft button.

### Util

With this button you call up the Utility page, which gives you access to four analytic and diagnostic tools. Double pressing the two center soft buttons from any editor is another way to get to the Utility page. The Utility page looks like this:

Select what to display:

**MIDI** **Objects** **Voices** **Stealer** **Done**

The **MIDI** soft button launches MIDIScope™, a useful subprogram that lets you monitor the MIDI messages from the K2vx's keyboard and those received via MIDI. This is a good way to make sure you're receiving MIDI from MIDI masters. It's also good for making sure your controls are assigned where you want them, checking your attack velocities, etc.

The **Objects** soft button displays the entire list of objects stored in RAM. This is an easy way to check the object ID of any object you've created. You cannot manipulate objects, though, as you can with the Objects Utility (see page 11-6).

When you press the **Voices** soft button, the display shows the K2vx's active voice channels as you play. Blocks of capital Xs in six columns of eight represent the 48 notes that the K2vx can play simultaneously. The Xs change to lower case xs, then to commas and periods, then finally drop out as each voice releases or decays to silence. This feature gives you an indication of the envelope level of each voice, though not necessarily the volume level. Nonetheless, this can give you a valuable indication of how your voices are being used. For example, if all or most of the voices are being represented by capital Xs on this screen, then there's a good chance that when voice stealing takes place an audible voice will be reallocated.

Use the **Stealer** soft button to select a display that will show how the K2vx is allocating its 48 voice channels. When you trigger a note, the note number will appear in one of the display's three columns, and will remain visible while the note is sustained. The four-digit numeral you see is an internal value that has no direct significance. As long as fewer than 48 voice channels are being used, new note numbers will appear as you play additional notes, and the note numbers for notes



that have decayed or have been released will disappear. When all 48 voices have been activated, the display will show which voice channels are shut off ("stolen") to enable new notes to play.

Press the **Done** soft button when you are finished with the Utility page. This is the same as pressing the Exit button.

### Sample

Press the **Sample** soft button to enter the K2vx's sampler. Refer to Chapter 15 for complete information on the sampler.

### Panic

This soft button sends an All Notes Off and All Controllers Off message to both the K2vx and over all MIDI channels.

### Reset

Press the **Reset** soft button if you want to return your K2vx's memory to the state it was in when you bought it.

**Caution!** The K2vx will ask you if you want to delete everything (meaning all RAM objects), and a pair of Yes/No soft buttons will appear. Press No if you want to keep any objects you may not have saved. Press Yes, and everything stored in RAM will be erased. All parameters will be restored to default values. After a few seconds, the K2vx will return to the Program mode page.

## Guitar/Wind Controller Mode

If you are using a wind controller or guitar controller with your K2vx, you may not always get the sound you expect. Since these controllers will sometimes send a MIDI Note On command before sending Breath or Volume data, the attack transients that characterize each instrument may be lost.

Therefore, the K2vx provides a special mode that may improve its response to your guitar or wind controller. To enter Guitar/Wind Controller mode, press both Chan/Bank buttons while in Master mode, then confirm with the **Yes** soft button.

Enable Guitar/Wind controller mode?

**Yes** **No**

Guitar/Wind Controller mode slightly delays MIDI Note On and Note Off commands, so that response to pitch bend and other expressive components of a note will be more accurate. If you are hearing a "glitch" in the attack of notes from your guitar or wind controller, you should try setting your K2vx to this mode. Keep in mind, however, that since this mode slightly changes the order in which MIDI commands are sent, it may affect the performance of the K2vx under some circumstances.

When you reboot the K2vx, it will reset Guitar/Wind Controller Mode to off.

## Object Utilities

Object Utility functions are useful for moving or copying objects into various banks, naming objects, deleting objects, and dumping objects over MIDI. To access these functions, press the **Object** soft button while in Master mode. You will see the following screen:

```
Select database function:
```

```
Move Copy Name Delete DUMP Done
```

The soft buttons are used to choose the various object utility functions.

**Move** Move selected objects to a new bank or a specific starting ID.

**Copy** Copy selected objects to a new bank or a specific starting ID.

**Name** Name selected objects.

**Delete** Delete selected objects.

**Dump** Dump selected objects over MIDI.

**Done** Exit from the object utilities.

Each function's multiple object selection interface is identical to the one used in the Save Objects dialog. For a complete description of this interface, see the "Saving Individual Objects" section on page 13-24. Here is what the **Move** object screen would look like (with several objects selected):

```
Func: MOVE Sel: 14/211 Index: 1
Sample 200*Zild 20 Ride Hrd C 4S 260K
Sample 201*Zild 20 Ride Sft C 4S 224K
Sample 202*Zild 16 Crash C 4S 341K
Sample 203*Zild NewBeatOpen C 4S 198K
Select Next TYPE Multi OK Cancel
```

The name of the function is displayed on the top line:

```
Func: MOVE
```

If you press **Cancel** while in one of the object utilities, you are returned back to the Object Utilities page (the "Select database function:" screen pictured above.) Any objects that were selected when you pressed Cancel will still be selected if you subsequently enter a different object utility (by pressing a different soft button such as **Name**, for instance.) The selections are reset when you exit the Object Utilities page (by pressing the **Done** button.)

All of the features of the Save Object dialog are accessible here:

**You can use the Multiple Object Selector** (described on page 13-32) to select ranges of objects according to object types, IDs, strings in the object names, or dependent relationships.

**You can quickly select or deselect all objects** using the Left/Right cursor and Up/Down cursor double-presses.

**You can audition any of the Program, Keymap, Sample, and Song objects** by pressing either the Left or Right cursor arrow button, when the desired object is highlighted. Songs will play until either cursor is pressed again. To audition a Program, Keymap, or Sample object, play a note on your MIDI controller (after pressing the Left or Right cursor button). (K2vxR users also have the shortcut of holding the CANCEL button, then pressing the numeric keys to play notes.)

## Move

Pressing **Move** from the Object Utility page takes you to the Move utility. The Move utility allows you to select any group of objects and move them to a different bank. If you select several objects of a single type, then you are allowed to set a specific starting ID for the objects, of any number (0-999). For instance, you could move a group of samples from scattered IDs to a continuous range of IDs starting from ID# 354.

If you move objects that are dependent objects of other objects (such as samples that are dependents of a particular keymaps,) the parent objects are automatically relinked to their dependents that have moved. What this means is that you can move any objects to any new ID numbers, without having to worry if your programs, keymaps or songs will still play correctly.

When you have selected the objects that you want to move, press **OK**. If you have selected more than one object, you will see the following dialog:

```

Move to bank: 200...299
              300...399
              400...499
              500...599
              APPend Fill Cancel
  
```

This is similar to the Load function, where you are asked to choose a bank and mode for loading. If you have selected more than one object, and all of the selected objects have the same type, then there will be an additional mode available, "ID":

```

ID APPend Fill Cancel
  
```

Here is a description of the above soft buttons:

**Append** Try to use each object's ID offset within its current bank as the ID offset within the specified destination bank. If there is already an object at this offset in the new bank, increment the ID until a free ID slot is found. For example, if you were moving Programs 202, 209, 217, and 230 to the 400's bank, which already has a program at 409, the resulting IDs for the moved programs would be 402, 410, 417, and 430.

**Fill** Use consecutive numbering for each object that is moved, starting from the beginning of the specified destination bank. Any object IDs that are already being used in the specified bank will be skipped over. For instance, if you were moving

Songs 300, 315, 489, and 841 to the 200's bank, which already contains Songs 200 and 203, then the moved songs' ID numbers would be 201, 202, 204, and 205.

**ID** Use consecutive numbering for each object that is moved, skipping over IDs that are in use (like Fill mode), starting from the ID that you specify. You will see the following dialog when you press the **ID** soft button:

Select starting ID: **201**

**OK** **Cancel**

**Cancel** Return to the Move object dialog.

The objects are moved as soon as Append or Fill is pressed, or when a starting ID is selected. After the Move function completes, you will still be in the Move object dialog, and you will still be scrolled to the previously highlighted object, even if it has moved to a new ID.

If you only select a single object to move, you will see a different dialog, similar to the Save/Replace dialog from the editor:

Move Awesome Click to: **ID#198**  
(rePlace Click)

**rePlace** **Cancel**

Note that when you only select one object to move, you can replace another object. However, when you select multiple objects for moving you cannot overwrite any objects.

## Copy

Pressing **Copy** from the Object Utility page takes you to the Copy utility. The Copy utility allows you to select any group of objects and copy them to a different bank. Only object data is copied, and not sample data. If you copy a sample object, you will end up with a "copy sample" that points to the same region of sample RAM as the original.

The operation of the Copy utility is identical to the Move utility just described.

If the objects to be copied in a single operation include any objects grouped together with any of their dependents, the new copies of the parent objects will reference the new copies of the dependent objects. As an example, suppose you select Song 400 and its three dependent programs, Program 200, 210, and 303. If you copy all of these objects at once into the 700's bank, using **Fill** mode, you will see the copies at Song 700 and Program 700, 701, and 702. Song 700 will reference the copies of the programs (at 700, 701, and 702.) In contrast, if you had only made a copy of Song 400 as Song 700, the song would reference the old programs (at 200, 210, and 303.)

## Name

The **Name** utility allows you to name one or more objects with the same name. This is much faster than naming each object individually. A lot of times you might want several objects to have the same name except for a unique identifier at the end of the name. Using this utility function, you could assign a common name to multiple objects at once, and then quickly name each one a little differently.

When you press **OK** after selecting objects, you will see the following screen prompting you for an object name, with a suggested default:

```
Object Name:   Zithep
```

```
Delete Insert <<< >>> OK Cancel
```

The default name that you are presented with is taken from the highlighted object in the object list, whether the highlighted object is selected or not. This makes it easy to copy the name of one object on to another.

As in all naming dialogs on the K2vx, you can do a double-press of the Left/Right cursor buttons to put the naming cursor on the last character of the string. This is helpful when putting unique characters at the ends of names.

### Left/Right cursor button double-press -> Move cursor to the end of the name

The Relink-by-Name feature (described earlier) relies on there being unique names for dependent objects of the same type, so it is a good practice to make object names unique, particularly samples.

## Delete

The Delete Objects utility is very useful for reclaiming unused object and sample RAM in your K2vx. This utility allows you to select any arbitrary group of objects for deleting, and audition them if necessary before getting rid of them.

If any of the selected objects have dependents that were not selected, you will see the question:

```
Delete dependent objects?
```

```
Yes No
```

If you answer **Yes** to this question, all dependent objects of the selected objects are deleted, unless they are being used as dependents of other objects that are to remain in memory.

Answering **No** will delete only those objects that were selected.

## Dump

This utility is for dumping selected objects over MIDI. If any of the selected objects have dependents that were not selected, you will be asked the question "**Dump dependent objects?**".

Press **OK** to initiate a MIDI System Exclusive dump of the selected set of objects, one by one out the MIDI Out port of the K2vx. Dumping everything can generate massive dumps, so you should know the limits of the device you're dumping to. You can cancel the dump at any time with the **Cancel** soft button.

Note that only sample objects (which contain the Start, Alt, Loop, and End points, as well as the values of all parameters found on the MISC page in the Sample Editor) are dumped by this utility, and not RAM sample data. (RAM sample data can be dumped via the MIDI Sample Dump Standard from within the EditSample page. See Chapter 10 of the *Reference Guide* for more information on the MIDI Sample Dump Standard.) Dumping the sample object of a RAM sample is not very useful because the sample memory address ranges are fixed in the object. This means that if you load the sample object back into the K2vx via MIDI, there is virtually no way it will point to and play back the same area of sample memory as when it was dumped, let alone the same sample data. However, this can lead to some interesting results.

Sample objects that reference the K2vx's ROM sample area will reference the same area when you load them back in via MIDI.

## Accessing the Object Utilities from the Editor

The object utilities can also be accessed while editing any object. This is provided as a convenience, for example to be able to do certain housekeeping work such as deleting samples to free up room in your sample RAM, or making copies of objects. Access to the utilities can be done by pressing the **Object** soft button from any Save/Replace dialog in the editor:

```
Save Train Wreck as: 10#412  
(rePlace Train Wreck)
```

```
Object                      Rename RePlace Cancel
```

You can get to this Save/Replace dialog when editing an object by either exiting after you have modified the object, or pressing the **Save** soft button.

If you try to use the Copy utility to copy the exact object you are in the process of editing, you will make a copy of the edited version. With sample objects this would be one way to save off a copy sample that references a small part of a much larger sample. You could remain in the sample editor, and continue to edit the larger sample, by pressing **Done** followed by **Cancel** after making the copy. This may be a faster way to save many "snippets" out of a sample than continually re-entering the sample editor after saving copy samples to different IDs.

When using the Object Utilities from within the editor, you must be careful not to delete any of the objects you are currently editing. This could have unpredictable results.

## Chapter 12

# Song Mode

### Getting Started with the Sequencer

The K2vx's sequencer is a powerful and versatile tool for songwriters, composers, and anyone else who needs to record and play back songs. As with any tool, however, it's best to start with the basics. This section begins with a tutorial where you will record a song, then shows some of the mixing capabilities of the sequencer. If you are familiar with other sequencers, you will have no problem using Song mode in the K2vx. Read through this section, however, to learn about the features that make the K2vx's sequencer unique.

#### What is a Sequencer?

A sequencer is similar in some ways to a multi-track tape recorder: you can record and play back all sorts of music and sounds, layer sounds on top of other sounds, and change or manipulate things that you've previously recorded. Unlike a tape recorder, however, you do not actually record sounds with a sequencer. Rather, you are recording commands that cause sounds to be played. Nonetheless, we will sometimes explain sequencer features by drawing analogies to familiar tape recording techniques such as splicing and overdubbing.

There are several advantages to recording a song by sequencing. For one thing, sequencer commands take up much less disk space than digitally recorded music would, so you can get a lot of information (i.e., music) on a single floppy disk. Furthermore, you can easily make changes to your sequences. For example, you can change individual notes, transpose parts, or change instrumentation. Lastly, you can share the sequences you create with other musicians.

#### A Word about the Local Keyboard Channel

Before you begin sequencing, we'd like to remind you about the Local Keyboard Channel parameter on the MIDI Receive page (described on page 10-6). Local Keyboard Channel is especially important for sequencing with the rack mountable K2vxR, since it enables you to record on different tracks without constantly switching transmit channels on your controller. Therefore, you should do the following before you begin sequencing with the K2vxR (or any K2vx model, when using an external controller):

- Set the Local Keyboard Channel on the MIDI Receive page to a specific channel (1-16).
- Set your controller (e.g., keyboard) to transmit on the same channel.

Performing the above two steps means that you'll be able to hear the individual channels (each of which is assigned by default to a separate record track) as you scroll through the different record tracks in the K2vx's Song mode. Local Keyboard Channel performs a "re-channelizing" function that makes this happen.

#### Patch Through

There's one more use for Local Keyboard Channel. With any model of the K2vx, the Local Keyboard Channel parameter lets you "patch through" (also known as "soft through") to external sound modules. When Local Keyboard Channel is enabled, the K2vx takes the rechannelized information and sends it out the MIDI port. This lets you hear an external module while you are recording a track assigned to that module.

### Tutorial: Recording a song

In this tutorial, we'll record a song by using the steps described below. Bear in mind, though, that this is just one approach to sequencing a song. This example includes:

- Assigning programs to channels
- Recording a drum loop
- "Unlooping" the drum track and adding the remaining instruments
- Mixing the song

#### Assign Programs to Channels

Start by deciding what instruments you want to use in the song. Suppose you want to record a bass / drums / organ rhythm track with a lead instrument on top. You've decided to use:

- Jazz Kit (Program No. 52)
- Mix Bass (Program No. 80)
- Gospel Organ (Program No. 97)
- Brt Saxy Lead (Program No. 46)

Set up your K2vx so that each of these instruments is on a separate MIDI channel. Since Song mode automatically assigns each channel to a separate sequencer track (1-16, consecutively), you'll then be all set when you start laying down tracks, and won't have to go scrolling through the program list. Don't worry about changing your mind later, though, since you can always make changes after you've recorded your initial tracks.

It will be most logical to assign Jazz Kit to one of the drum channels, such as channel 1. If you want to put the drums on a channel that isn't a drum channel, though, you'll have to change the drum channel on the Master page (see page 11-1). For the sake of simplicity, however, we'll put the drums on channel 1 and assign Mix Bass, Gospel Organ, and Brt Saxy Lead to channels 2, 3, and 4.

Follow these steps to assign the programs to separate channels:

1. Press the Song mode button to enter Song mode. The screen will look something like this:

Record track is set to track 1

```

SongMode: MAIN  Events: 188K  STOPPED
CurSong: 1  NewSong  Tempo: 120
RecTrk : 1  Vol: 127  Pan: 64  Mode : Merge
Program: 1  Acoustic Piano  Locat: 1:1

Track  :R  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
Channel: 1  2  3  4  5  6  7  8  9  10 11 12 13 14 15 16
Record  Play  Stop  Erase  MISC  MIXER

```

Notice that the sequencer is ready for you to record a new song, and the record track (RecTrk) is set to track 1. If "1 NewSong" doesn't appear in the CurSong field, press the MISC soft button, then press the New soft button on the MISC page. You will be returned to the MAIN page, and CurSong will say "1 NewSong."



2. Use the down arrow button to move the cursor to the Program field.

When this field is highlighted, type "52" then press the ENTER button. You've now assigned "Jazz Kit" to channel 1.

```

SongMode:MAIN  Events:188K  STOPPED
CurSong:1  NewSong          Tempo:120
RecTrk :1    Vol:127 Pan:64    Mode :Merge
Program:52 Jazz Kit          Locat: 1:1

Track  :R - - - - -
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record  Play  Stop  Erase  MISC  MIXER

```



**NOTE:** If "52 Jazz Kit" appears in parentheses, it means that the current channel is not a drum channel. You can either go to the Master page to change the drum channel, or assign the drums to a current drum channel in Song mode.

3. Now use the up arrow button to highlight the RecTrk field.

Change RecTrk to "2". You can use either the alpha wheel or the keypad.

4. Return to the Program field, and set it to "80 Mix Bass".

```

SongMode:MAIN  Events:188K  STOPPED
CurSong:1  NewSong          Tempo:120
RecTrk :2    Vol:127 Pan:64    Mode :Merge
Program:80 Mix Bass          Locat: 1:1

Track  :- R - - - - -
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record  Play  Stop  Erase  MISC  MIXER

```

Notice that the "R" on the Track line, which stands for "Record", has moved to track 2. Also notice how each of the sixteen tracks has a default channel associated with it. You could change this if you wanted to, but most people find it easiest to associate track 1 with channel 1, track 2 with channel 2, etc.

5. Repeat the above two steps to assign "97 Gospel Organ" to channel 3 and "46 Brt Saxy Lead" to channel 4.

You've now chosen the programs for your first sequence. It's important to realize, though, that you have not recorded them yet. They will be there when you need them, but they have not yet been included in a song. Also, don't forget that you can change the program assignments any time before or after you record the song.

### Record a Drum Loop

Our song will be based around a four measure drum loop that we'll record now. Then, a little later on, we'll "unloop" the drum track and really start jamming.

The length of the drum loop is determined by the current *endpoint*, so we'll start by recording 4 measures of silence to set the endpoint.

1. Set RecTrk to "1" then press the Record soft button. The Song Status indicator (top right hand corner of the display) flashes "REC READY".

```

                                song status indicator
                                ┌───┴───┐
                                │           │
SongMode:MAIN Events:188K REC READY
CurSong:1 NewSong          Tempo:120
RecTrk :1   Vol:127 Pan:64  Mode :Merge
Program:52 Jazz Kit         Locat: 1:1

Track  :R - - - - - - - - - - - - - - -
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record  Play  Stop  Erase  MISC  MIXER

                                └───┬───┘
                                │           │
                                Locat parameter
                                shows current measure
                                and beat number
    
```

2. Press the Play soft button. The Song Status indicator now reads "RECORDING".  
 The K2vx's built-in metronome will begin clicking, and the Song mode LED will also flash at the current tempo.  
 Notice the "Locat" parameter on the right side of the display, which shows the current measure and beat number. When you begin recording, the K2vx will provide you with a four beat count-off, during which time Locat's measure and beat number will be preceded by a minus sign.  
 Since we are recording four measures of silence, press the Stop soft button as soon as Locat reads "4:4".



*NOTE: The sequencer will truncate to the nearest downbeat, so as long as you press Stop before Locat reads "5:2" (but after it reads 4:4) you'll be alright. Don't worry about this too much, though, since in the next step we'll show you how to check (and change, if necessary) the endpoint.*

When you press Stop, you'll be asked to confirm and name this song. Even though we've just recorded four measures of silence, when you save it and give it a name it's officially a song.

For the purposes of this example, we'll assume that you pressed Stop a few beats too late. You'll see how easy it is to correct this sort of thing in the Event Editor.

3. On the Song mode MAIN page, make sure that the cursor is highlighting any field other than the Program field, then press the EDIT button.  
 If the Program field is highlighted when you press EDIT, you'll enter the Program Editor, which is not what you want to do right now. The EditSong: COMMON page appears:

```

editSong:COMMON <> track:1
Tempo      :120      StartStep   :1
TimeSig    :4/4      TempoControl:Song
EffectChan :1        TriggerChan :1
ChainTo    :0 None   TriggerCtl  :ON
TrackDest  :-----
DrumTrack  :-----
<more>  EVENT TRACK STEP ARRANG <more>
    
```

- Now press the EVENT soft button to bring up the Event Editor, which will look something like this:

```

EditSongEvent (On 1) <> track:1
1:1 1:1.000 CTRL BANK 0
1:1 1:1.000 PCHG 52
1:1 1:1.000 CTRL VOL 127
1:1 1:1.000 CTRL PAN 64
Cut Copy Paste View AllOn Done

```

As you can probably decipher, the Event Editor gives you access to an editable list of all note, controller, and other MIDI events that Song mode uses to describe your sequence. While you're looking at the Event Editor, notice the data that the sequencer keeps track of, even when no notes are played.

- Use the Alpha wheel to scroll to the bottom of the event display.

The last event listed is your endpoint, which should be the first beat of the measure following the last measure you want to record. For our four measures of silence, then, the endpoint should be 5:1. The display below, however, shows that five measures have been recorded:

```

EditSongEvent (On 1) <> track:1
1:1 1:1.000 CTRL VOL 127
1:1 1:1.000 CTRL PAN 64
6:1 6:1.000 END
Cut Copy Paste View AllOn Done

```

Fortunately, it's quite easy to change the endpoint from "6:1" to "5:1" to remove the extra measure that's been accidentally recorded. (If your endpoint is at 5:1, you won't need to change it now. If, however, your endpoint is less than or greater than 5:1, you should proceed with the next step.)

- Check the endpoint, and change it if necessary.

To change the endpoint from 6:1 (or any other incorrect value) to 5:1, press the right arrow button to position the cursor in the Bar:Beat:Tick column ("6:1.000" in the above example). Type "51000" then press the ENTER button. The endpoint is changed, and the song is now four measures long.

- Press EXIT to return to the SongMode:MAIN page.

If you've made changes in the Event Editor, confirm them now.

- Make sure that record mode and play mode are set to "Loop."

From the SongMode:MAIN page, press the MISC button. Set the RecMode parameter on the SongMode:MISC page to "Loop".

The PlayMode parameter should already be set to "Loop". If it isn't, turn the alpha wheel until "Loop" is highlighted.

```

RecMode & PlayMode parameters          Quantize parameters
-----
SongMode:MISC      Events:188K      STOPPED
RecMode : Loop     Quant:Off      CountOff:1
PlayMode: Loop     Grid :1/8      Click :Rec
KeyWait : Off      Swing:0%      ClickCh :16
Locate  : 1:1      Sync :Both    ClickPrg:198
AutoIn  : 1:1      Clock:Int     ClickKey:C 4
AutoOut : 1:1      Tempo:Auto    ClickVel:90
Record  Play  Stop  New  In/Out  MAIN
    
```

**9. Turn on Input Quantization.**

While you're on the MISC page, take a look at the quantize parameters (Quant, Grid, and Swing) in the middle column of the page. Quantization is a very useful feature, especially for the rhythmically challenged. When you quantize a track, the sequencer moves the elements of that track closer to a grid based on the time signature of the tune. You can use quantization to subtly tighten up a rhythm track or to create a precise, unwavering mechanical rhythm.

For our drum loop, we'll try the total quantize experience, so position the cursor over the Quant parameter, then turn the alpha wheel until the value is set to "100%". Move the cursor down to the Grid parameter. The default value "1/8" indicates that quantization will move the notes you play to the closest eighth note. Try double-pressing the increment/decrement buttons below the alpha wheel to move through a range of useful grid values. Note that some of the values have "tr" or "t" appended to them. These are grid settings that allow you to maintain a triplet feel. We'll use a setting of "1/16", however, so set this value, then press EXIT to return to the SongMode:MAIN page.



*NOTE: The K2vx's sequencer also provides a full range of advanced quantization features that you can apply to previously recorded tracks. To learn about these, check out the Quantize and Reference Quantize functions on the EditSong:TRACK page.*

**10. Make sure you are in Merge Mode.**

On the SongMode:MAIN page, check the Mode parameter, and make sure it is set to "Merge" rather than "Erase". Being in Merge Mode means that while you are recording this track you will be able to overdub more sounds with each successive loop.

**11. Begin recording drums.**

Press the Record soft button, observe the "REC READY" indicator on the top line, then press the Play soft button when you're ready to begin. Remember to wait for the four beat count off before you start to play.

Since you are in Merge Mode, you don't need to do everything at once. A common approach to making drum loops is to record a different instrument each time the loop comes around. For example, on the first loop you could record snare hits on the back beats (1:2, 1:4, 2:2, 2:4, etc.). Then you could add kick drum to the snare when the loop comes around again; you'll be able to hear the previously recorded part, as well as the new part. On the third pass you might record ride cymbal, followed by hi-hat or other percussive accents. Keep it simple at first, because you can always save the part while it's

basic (but correct), then make additions later. To keep track of where you are, watch the flashing Song mode LED or the Locat parameter on the SongMode:MAIN page.

**12. Press the Stop soft button when you've finished recording the drums.**

Confirm that you want to keep the changes to your song by pressing the Yes soft button followed by the Replace soft button.

### Record a Bass Line

When you are satisfied with your drum loop, you can begin using it as the foundation for a song. What we'll do here is set Record Mode to "unloop" while leaving Play mode set to loop. This means that the drum loop will keep playing while we record new material of any length. The endpoint of the song will change to reflect the length of the newly recorded material.

**1. Press the MISC soft button to bring up the SongMode:MISC page.**

**2. Set the RecMode parameter to "Unloop". Leave the PlayMode parameter set to "Loop".**

Depending on the type of song you are recording, you may also want to turn quantization off before you record your bass part.

```

SongMode:MISC  Events:186K  STOPPED
RecMode :UnLoop  Quant:100%  CountOff:1
PlayMode:Loop   Grid :1/16   Click  :Rec
KeyWait  :Off    Swing:0%   ClickCh :16
Locate   : 1:1   Sync :Both  ClickPrg:198
AutoIn   : 1:1   Clock:Int  ClickKey:C 4
AutoOut  : 1:1   Tempo:Auto ClickVel:90
Record   Play   Stop   New   In/Out MAIN

```

**3. Press the MAIN soft button to return to the SongMode:MAIN page.**

**4. Set the record track (RecTrk) to track 2.**

This track already has material recorded on it.

```

SongMode:MAIN  Events:186K  STOPPED
CurSong:200*NewSong  Tempo:120
RecTrk :2    Vol:127 Pan:64  Mode :Merge
Program:80 Mix Bass  Locat: 1:1
□
Track  :P R - - - - -
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record Play Stop Erase MISC MIXER

```

Track status indicators: track 1 is set to "Play", track 2 is set to "Record". Tracks 3 through 16 are empty.

Since you previously assigned "Mix Bass" to channel 2, it should appear in the Program field when you set track 2 as the record track. Note, too, that the track status indicator for

track 1 changes to "P" (for "Play") when you select track 2 for recording. The small square above the track status indicator tells us that material is contained on that track.

- 5. **Press the Record soft button to enter "REC READY" mode.**
- 6. **Press the Play soft button, then begin laying down a bass track.**

Remember that by default there is a four beat count off, during which time the Locat value will be preceded by a minus sign (-). No material is recorded during the count off, though anything you play during the countoff will be quantized to the first note of the song. As you are recording the bass track, your drum loop will keep playing. Play for as long as you want; the sequencer will lengthen the song as needed.

- 7. **Press the Stop soft button when you are done recording the bass.**

You will be given the usual save options. To keep what you've just recorded, press the Yes soft button followed by the Replace soft button.

Since you unlooped the drum track when you recorded the bass, you've changed the endpoint of the song to be wherever you stopped the bass track. You can check the endpoint (and change it, too, if you want) using the Event Editor, as described earlier.

### Record the Remaining Instruments in Your Song

Now that you've defined your song with the bass and drum tracks, you can put the organ and Brt Saxy Lead (or whatever instruments you've chosen) into your song.

- 1. **Set the record track (RecTrk) on the SongMode:MAIN page to track 3.**

Notice the small squares above the track status indicators for tracks 1 and 2, reminding you that you've now got material on two tracks.

```

SongMode:MAIN  Events:186K  STOPPED
CurSong:200*NewSong  Tempo:120
RecTrk :3  Vol:127  Pan:64  Mode :Merge
Program:97  Gospel Organ  Locat: 1:1
  □ □
Track  :P P R - - - - -
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record  Play  Stop  Erase  MISC  MIXER

```

- 2. **Press the MISC soft button to bring up the SongMode:MISC page.**
- 3. **Set RecMode to "FixLen".**  
 Since you've defined the length of your song with the bass track, setting RecMode to "FixLen" means that the song will just play through once each time you record a new part.
- 4. **Record the organ in the same way that you recorded the bass track in the previous section.**  
 Notice that you can do this from the MISC page, without returning to the MAIN page.
- 5. **Continue recording instruments until you have played all the parts of your song.**

## Mix Your Song

The SongMode:MIX page lets you change the panning and volume levels for the tracks in your song. Needless to say, this is one of the most important steps in the completion of your song production, and potentially one of the most creative. This example will keep things simple by showing you how to change the volume level of one of your instruments. We'll also take a quick look at the track mute feature.

1. **Press the MIXER soft button to bring up the SongMode:MIX page.**

The icons that represent pan position knobs and volume level faders resemble the controls on a traditional mixing board. Manipulating them should be quite intuitive. Simply position the cursor over a pan position knob or volume level fader, then turn the alpha wheel to set the level you want.



Track 3 Volume Level Fader

For example, suppose you want to turn down the organ on track 3:

2. **Use the arrow keys to position the cursor over track 3's volume level fader on the SongMode:MIX page.**
3. **Use the alpha wheel to turn down the volume of the track by changing the position of the volume slider.**
4. **Press the Keep soft button and confirm the change.**

Now when you play back the song, track 3's volume will start playing at the newly set level.

You can also record real-time volume and pan changes from the MIX page or enter numeric values for these parameters on the MAIN page.

## Using the Mode Buttons to Mute a Track

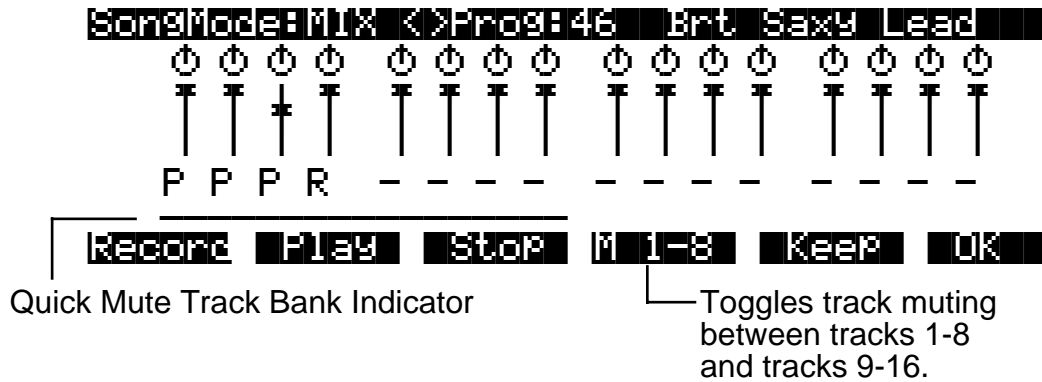
Finally, we'll take a quick look at the track mute feature, which lets you use the K2vx's mode buttons to mute individual tracks. This can be invaluable during mixdown.

You may have noticed a horizontal line underneath the sliders for tracks 1 through 8 on the SongMode:MIX page. As the illustration below shows, this is the "Quick Mute Track Bank Indicator", showing which bank of eight tracks will respond to the eight track mute buttons on the front panel of the K2vx. If you're looking for the track mute buttons on the front panel of your K2vx, they're the eight mode buttons (Program, Setup, etc.); press one of these buttons

## Song Mode

### Getting Started with the Sequencer

while you are on the SongMode:MIX page and playing a song, and the associated track will be muted.



Use the M 1-8 / M 9-16 soft button to toggle between banks of eight tracks, either 1 through 8 or 9 through 16. When you press this soft button, the horizontal bar will reposition itself below the affected tracks. The table below shows the K2vx's mode buttons, and which tracks they will mute when you are playing a song:

Mode Button	M 1-8	M 9-16
Program	1	9
Setup	2	10
Quick Access	3	11
Effects	4	12
MIDI	5	13
Master	6	14
Song	7	15
Disk	8	16

Let's listen to our song with and without drums:

1. **From the SongMode:MIX page, press the Play soft button.**  
The song, with all of its instruments, begins to play.
2. **Press the Program mode button on the front panel of the K2vx.**  
The Program mode LED will light, and track 1 (the drum track) will be muted.
3. **Press the Program mode button again to unmute the drum track.**  
Using your K2vx's front panel buttons, you can mute one or more tracks at once, or even mute eight tracks at a time.

With any luck, you've now completed your first song. You should now be ready to take on the more advanced features of K2vx Song Mode that are described in the remainder of this chapter.



## MAIN Page

The Song Mode MAIN Page allows real time recording and playback, song and track selection. From this page you can view and edit the tracks' channel, program, volume and pan settings, as well as other useful items.

```

SongMode:MAIN  Events:317K  STOPPED
CurSong:1  NewSong  Tempo:120
RecTrk :1  Vol:127  Pan:64  Mode :Merge
Program:2  Stereo Grand  Locat: 1:1

Track  :R - - - - -
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record  Play  Stop  Erase  MISC  MIMER

```

PARAMETER	RANGE OF VALUES	DEFAULT
CURRENT SONG	SONG ID & NAME	1 NEWSONG
RECORD TRACK	1 – 16, NONE, MULT	1
PROGRAM	PROGRAM ID & NAME	CURRENT PROGRAM
SETUP	SETUP ID & NAME	CURRENT SETUP
TRACK STATUS	-(EMPTY), R, M, P	-(EMPTY)
CHANNEL	1 – 16	1 – 16
VOLUME	0 – 127	127
PAN	0 – 127	64
TEMPO	1 – 255 BPM	120 BPM
MODE	MERGE, ERASE	MERGE
LOCATION	1:1 – 9999:9	1:1

The **Events** field on the top line displays the amount of free RAM space. 317 K in the above example represents maximum available memory with P/RAM installed. The maximum is 60 K without P/RAM. This field is replaced by the **Used** field, as shown below, when the Song Status is REC READY or RECORDING, to display the percentage of the record buffer (not the RAM memory) being used.

```

SongMode:MAIN  Used: 0%  REC READY

```

**Song Status**, also on the top line of the display, will be one of the following:

- **STOPPED** is displayed whenever the sequencer is not in another status, or when the **Stop** or **Pause** button is pressed.
- **PLAYING** is displayed when the **Play** button is pressed (assuming **Record** has not been previously pressed; and Key Wait is Off and the Clock set to Internal)
- **REC READY** is displayed when the **Record** button is pressed while the sequencer is STOPPED. It flashes, indicating that it is waiting to start recording.
- **RECORDING** is displayed when the **Play** button is pressed while REC READY is flashing (unless Key Wait is On or the Clock is set to Ext). RECORDING will also be displayed if the **Record** button is pressed while the status is PLAYING.
- **KEY WAIT** is displayed when the **Play** button is pressed, if the KeyWait parameter on the MISC page is set to On. It will go this status from both the STOPPED and REC READY status. It flashes, indicating it is waiting for a key to be struck to start recording or playing.

- **EXT. CLOCK** is displayed when the **Play** button is pressed, if the Clock parameter on the MISC page is set to Ext. It will go this status from both the STOPPED and REC READY status. It flashes, indicating it is waiting for an external MIDI clock message to start recording or playing.

### **Current Song (CurSong)**

Current Song ID# and 16 character name selected for recording, playback or editing. When a Song is selected, Program Change, Volume, and Pan information is sent to all MIDI channels assigned to tracks that have data on them, and the internal clock is set to the Tempo.

### **Tempo**

Controls tempo for the selected song. You can make temporary changes, record real time tempo changes, or set an initial tempo for the current Song.

Whatever the tempo is set to when you record your first track will be the Song's initial tempo. Temporary changes may be made during playback, but the tempo will reset to the initial tempo when the sequencer is STOPPED.

To change a Song's initial tempo, press **Record** (the Song Status will change to REC READY), set the tempo desired, then press **Stop**. The initial tempo can also be changed with the Tempo parameter in the EditSong : COMMON page. The Song will always start playback at the initial tempo, even though this tempo marker does not get recorded as a tempo event on any track.

If the sequencer is RECORDING, any tempo value changes will be recorded in real time. Unlike the special case of setting the initial tempo, any tempo changes recorded in real time are recorded as tempo events.

### **Fractional Tempos**

You can use fractional tempos (120.5, etc.) in your sequence. However, the initial tempo can not be fractional, and you cannot enter a fractional number in the tempo parameter on the MAIN or COMMON pages. You must first record a real time tempo event, then go to the EVENT editor and change it to a fractional amount.

To do this, press **Record**, then **Play**. The sequencer starts recording. Use any data entry method to choose a tempo. The value is unimportant since you will be changing it in the EVENT editor. Next press **Stop** and save the Song. Now when you go to the EVENT editor (see page 12-28), you will see a tempo event. You can now edit the value to a fractional amount. To have the song start immediately with the fractional tempo, edit its location to 1:1:000.

### **Record Track (RecTrk)**

Determines which track is record enabled. Set the record enabled track to Multi to record more than one channel simultaneously or to use a Setup in your Song.

When RecTrk is set to a single track ( 1 - 16 ), Record (R) is displayed for that track in the Track Status Indicator field (above the Track Channels). Conversely, with one exception, when any Track's Status Indicator is changed to Record (R), that Track is shown as the value for the RecTrk parameter.

The exception is when RecTrk is already set to Multi, you can select the record enabled tracks by toggling the Track Status Indicator to Record (R), and the RecTrk will remain set to Multi.

When Multi is initially selected, all of the empty tracks will be record enabled. Tracks containing data will remain set to play (P), but you can manually set them to record (R).

The parameter(s) below RecTrk will change according to the value of RecTrk and whether or not you select a setup. If RecTrk is set to a single track (1-16), Program is displayed and you can select the program to be assigned to that track. If you change RecTrk to Mult or None, the display changes to show the Channel parameter followed by the Program parameter (although the the word "Program" no longer appears). An example of this is shown in the screen below. If you switch through the channels, the program will also change, showing the program currently on that channel. Finally, if RecTrk is set to Mult, then if you go directly to Setup Mode and then back to Song mode, the parameter changes to Setup, allowing you to choose a setup with which to record.

### Program

Scroll through the Program objects in memory to select the Program before initially recording each track of your Song. Any MIDI program changes on the current RecTrk or Chan will cause the ID# and name of the track's program to change during playback.

This parameter's name, Program, is not present on the display when the RecTrk is set to None or Multi (to make room for the Chan parameter), but the value is still displayed.

The parameter is replaced with Setup when you select a Setup for playback or recording. The Setup parameter functions similarly to Program.

Programs selected in Program Mode or from a Quick Access bank will be selected as the program on the current RecTrk when you return to Song Mode.

```

SongMode:MAIN  Events:188K  STOPPED
CurSong:1 NewSong          Tempo:120
RecTrk  :MULT Vol:127 Pan:64  Mode :Merge
Chan:4   2 Stereo Grand      Locat: 1:1

Track   :R R R R R R R R R R R R R R R R
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record  Play  Stop  Erase  MISC  MIXER

```

### Channel (Chan)

This parameter determines the control channel and is only available when the RecTrk is set to None or Mult. The Program parameter value remains on the display and can be edited, even though the parameter's name, "Program", no longer appears.

### Setup

Displays the ID# and name of Setup to be recorded. This parameter is available by setting RecTrk to Mult, then entering Setup mode momentarily by pressing the SETUP button followed by the SONG button to return to Song mode. The screen below shows an example of the Song mode page with the Setup parameter displayed.

```

SongMode:MAIN  Events:188K  STOPPED
CurSong:1 NewSong          Tempo:120
RecTrk  :Mult Vol:127 Pan:64  Mode :Merge
Setup   :4 Total Orchestra    Locat: 1:1

Track   :R R R R R R R R R R R R R R R R
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record  Play  Stop  Erase  MISC  MIXER

```

The Chan parameter will be replaced with Setup. It is important to know how many channels, and consequently how many tracks, are needed for recording a particular Setup. Each Setup can have up to three Zones that can respond to your playing differently, depending on what range of the keyboard is being played, or if certain velocity and/or controller values determines when a particular Zone will respond. Be aware of the behavior of each Setup you intend to record so that you can allocate the proper tracks and channels needed in your Song.

**Volume (Vol)**

You can set an initial volume level for the playback and recording of each track as a value between 0 and 127. If the channel of the RecTrk (or the control channel, if RecTrk is set to Multi or None) contains any recorded volume change (controller code 7), the change will be reflected as the Vol parameter's value in real time.

**Pan**

You can set an initial pan position (the balance between the Left and Right audio channels) for the playback and recording of each track as a value between 0 and 127. A value of 64 is center. If the channel of the RecTrk or the control channel contains any panning data (controller code 10), the Pan parameter's value is modified in real time.

**Mode**

If Mode is set to Merge you will be able to overdub when recording on a track containing previously recorded data. You'll usually want to set Mode to Merge when RecMode (on the MISC page) is set to Loop. Otherwise, each time through the loop, the previously recorded information will be erased.

If you set Mode to Erase, the previously recorded data on the record enabled track will be replaced with the new data only during the Bars and Beats you are actually recording, and the previously recorded data before and after the newly recorded Bars and Beats will be preserved.

**Location (Locat)**

The Bar and Beat displayed as the Locate value changes relative to current location of the Song during playback and recording. You can set this to a negative Bar and Beat location to start playback a set length of time before the beginning of the Song.

Whenever you set the Locate point, that location will be used as the return point when **Stop** is pressed. Simply press **Stop** again to reset the Song to the top (1 : 1).

		□	□	□	×													
Track	:	P	P	M	R	-	-	-	-	-	-	-	-	-	-	-	-	-
Channel:		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	

**Mode Indicators (+ and x):**

Mode Indicators only appear when there is pre-existing data on a track(s).

A plus sign (+) appears above the Track Status Indicator of a track set to record (R) when the Mode parameter is set to Merge.

An (x) appears above the Track Status Indicator of a track set to Record (R) when the Mode parameter is set to Erase.

**Activity Indicators ( □ )**

A small square (□) above the Track Status Indicator of a track set to Play (P) or Mute (M) means there is data contained on that track.

During playback and recording, the indicators above tracks containing any MIDI data will flash a small, filled-in square when any MIDI activity is detected. The filled-in square also flashes over a track any time that there is incoming MIDI data on that track's channel, even while the sequencer is STOPPED.

### Track Status Indicators

Using the Up, Down, Left, and Right cursor arrows to position the cursor onto a Track Status Indicator, you can toggle an empty track (-) into Record (R) with the Alpha Wheel or Increment/Decrement buttons.

Once a track contains data, it will have a (P) as a Track Status Indicator, and it will be played during playback. You now will be able to toggle between Play (P), Mute (M), and Record (R).

The track selected as the RecTrk will have an (R) in this field, designating it as the record enabled track. If the RecTrk is set to Mult, initially all empty tracks will have Record (R) as a Track Status Indicator, any of which can be switched back to empty (-) if at any time recording on specific tracks is not desired.

If there isn't a track with an (R) in this field, the RecTrk parameter's value will be None. (The exception is when the RecTrk is set to Mult and you have switched all of the tracks out of record enable.)

### Track Channels

Each track has a MIDI Channel that it uses to receive and transmit data. By default, tracks 1 through 16 of a new Song are assigned to channels 1 through 16 respectively, although a track can play or record on any channel and the same channel can be assigned to more than one track. However, keep in mind that only one program can be assigned to a channel at a time, so if you have more than one track assigned to the same channel, they will be playing the same program.

## Soft Buttons on the MAIN Page

Although these soft buttons may look similar to the transport controls on a tape deck that might require that you hold down play and record simultaneously to begin recording, they do not operate in the same fashion. It is important that you only press one of these transport style soft buttons at a time to insure proper record start points, and to always be sure of the current sequencer status.

**Record** **Pause** **Stop** **Erase** **MISC** **MIXER**

**Record** changes the Song Status to REC READY if the current Song Status is STOPPED. If the current Song Status is PLAYING, it will be switched to RECORDING when you press **Record**.

If the RecTrk is set to None, pressing **Record** will select the first available empty track for recording, thus setting the RecTrk to the newly record-enabled track number and placing an (R) in that track's status display. Song Status will change to REC READY or RECORDING, depending on the previous Song Status as described above.

**Play** plays back any recorded data when pressed while the song status is STOPPED. Playback will begin from the bar and beat specified in the Locate parameter.

When the Song Status is REC READY, pressing the **Play** soft button will begin recording.

**Pause** and **Play** share the same soft button. **Pause** appears only when the Song Status is PLAYING or RECORDING. Pressing **Pause** while the song is playing will stop the playback (soft button switches to

*Play*), and the location remains at the current bar and beat, allowing you to continue from that location by pressing **Play** again.

Pressing **Pause** while recording will stop the recording process as if you had pressed **Stop**.

**Stop** halts the playback or recording, and resets the Song's location to either the default "bar 1, beat 1" value, or to whatever location you defined with the Locate parameter. If the location is defined as something other than "bar 1, beat 1", press **Stop** twice to return to 1:1.

Pressing **Stop** when the Song Status is RECORDING will always prompt the "Save this song?" dialog (shown on page 12-16), and provides you with the opportunity to listen to the "New" Song and compare it with the "Old", previously saved, Song before answering Yes or No.

Here are two useful alternatives to using these button presses:

- Footswitches can be assigned to **Play/Stop** and **Record/Stop**. On the MIDIMode : TRANSMIT page, assign either footswitch to controller number 102 or 103. Using controller 103, you can even record from within the Song editor.
- Double button presses allow Song **Play**, **Pause** and **Stop** soft button functions from almost anywhere where these buttons are not available. Use the cursor **Left+Right** double-press for **Play/Stop**, and use the cursor **Up+Down** double-press for **Play/Pause**.

**Erase** removes all channelized data from the track on which the cursor is currently positioned (on either the Track or Channel fields) or from the record enabled track if the cursor is positioned elsewhere. As shown on page 12-17, a dialog appears that allows you to verify your intentions before permanently erasing any data. The Erase function will not prompt the dialog or erase any data if an empty track is selected.

**MISC** accesses more sequencer control parameters found on the SongMode : MISC page. These miscellaneous controls include record and play mode settings, auto punch-in points, quantization on input settings, tempo, and click attributes, as documented later in this chapter, starting on page 12-18.

**MIXER** lets you view a graphic representation of panpots and level faders for each track in the EditSong : MIX mode. You can modify program changes, volume and pan position in this Mode. Record status can also be selected here. The section on the Mixer begins on page 12-23.

## Save this song? Dialog

The following dialog page appears after you have recorded a track and pressed **Stop**, or if you have entered the Song editor and made changes, then pressed Exit, or if you press **Save** in the Song editor.

Save this song?

PlayOld PlayNew Stop

Yes No

**PlayOld** appears along with the **PlayNew** soft button in the "Save this song?" dialog after the recording process has been stopped. Pressing **PlayOld** will play the current song, minus the last, but not yet saved, recorded data. The exception is when there is no previously recorded data (i.e., recording the first track of a NewSong), then you will only have a **Play** soft button to audition the recorded data just entered.

**PlayNew / Play** soft buttons allow you to play all of the recorded data, including data on the track(s) you have just recorded.

**Stop** halts the playback of either the Old or the New version of the Song you are currently auditioning. Press the **Stop** soft button to stop the playback or recording, and reset the Song's location to either the default "bar 1, beat 1" value, or to whatever location you defined in the Locat parameter

**Yes** saves the data on the track(s) you just recorded. Whatever was played back when you pressed **PlayNew** will be the version of the Song to be saved when you press Yes. The "Save New Song?" Dialog (shown below) will be displayed.

**No** returns you to the Song mode page in which you were last recording, without saving any unsaved changes to the current Song.

## Save New song? Dialog

Save NewSong as: ID#200

**Object**

**Rename**

**Save**

**Cancel**

Saving a Song requires that you assign it an ID# and this is where you do so. If you select an ID# of an existing Song Object in RAM, the **Save** soft button will become **Replace** and "Replace existing Song" will appear on the display as a warning and a chance to change your mind.

There are only 20 ID#s available for Song Objects per bank in banks 100 - 900. In the zeros bank, there are 75, ID#s 1 - 75. In the 200's bank there are ID#s 200 - 219, in the 300's bank, ID#s 300 - 319, and so on.

**Object** accesses some useful database functions while still in Song mode, before you actually save the current Song. Pressing **Object** jumps you directly to the Object Utility, described on page 11-6. When you press **Done** on the Utility page, you will be returned to the "Save New Song" dialog.

The **Rename**, **Save**, and **Cancel** soft buttons function the same as they do in all other editors.

## Erase Track Dialog

This dialog asks you if you are sure you want to erase a specific track. The track number will correspond to the track currently selected by the cursor position. When RecTrk is set to Multi or None, the Erase track function looks to the cursor position to determine the track to erase. If the cursor is positioned somewhere other than the Track or Channel fields, Erase will have no effect. Erasing a track will not alter the Song's End point, nor will it remove any tempo events, since these elements are common to all tracks in the Song.

**Yes** erases all data on the selected track and returns to the MAIN Song Mode. The Song will keep the erased track record enabled, but it will be an empty track.

**No** aborts the erasing of the selected track and returns to the MAIN Song Mode with all previously recorded tracks intact.

## MISC Page

This page contains parameters that determine how the sequencer behaves during playback and recording, and when the K2vx is connected to another MIDI sequencer. Most of these settings are stored in the Master Object, (all except the Auto punch-in points) and none of these are saved with the Song.

```

SongMode:MISC Events:188K STOPPED
RecMode :Linear Quant:Off CountOff:1
PlayMode:Loop Grid :1/8 Click :Rec
KeyWait :Off Swing:0% ClickCh :16
Locate : 1:1 Sync :Both ClickPrg:198
AutoIn : 1:1 Clock:Int ClickKey:C 4
AutoOut : 1:1 Tempo:Auto ClickVel:90
Record Play Stop New In/Out MAIN
    
```

PARAMETER	RANGE OF VALUES	DEFAULT
RECORD MODE	LINEAR, FIXLEN, LOOP, UNLOOP, AUTO	LINEAR
PLAY MODE	LINEAR, LOOP, LIST, CHAIN	LOOP
KEY WAIT	ON, OFF	OFF
LOCATE	-25:2 – 999:4	1:1
AUTO IN	1:1 – 999:4	1:1
AUTO OUT	1:1 – 2500:	1:1
INPUT QUANTIZE	OFF, 1% – 100%	OFF
GRID	1/1 – 1/384	1/8
SWING	-99% – 125%	0%
SYNC	BOTH, XMIT, RECV, OFF	OFF
CLOCK	INT, EXT	INT
TEMPO	AUTO, FIXED	AUTO
COUNTOFF	OFF, 1, 2, 3	1
CLICK	OFF, REC, ON, CNT	REC
CLICK CHANNEL	1 – 16	16
CLICK PROGRAM	1 – 999	198
CLICK KEY	C-1 – G9	C4
CLICK VELOCITY	1 – 127	90

The top line displays the amount of free RAM space and the current sequencer state, which is one of the following: STOPPED, PLAYING, REC READY, RECORDING, KEY WAIT, or EXT. CLOCK.

### RecMode

When recording the first track of a new Song, you will be able to record as if you had an endless length of “tape” no matter what the Record Mode is set to.

When you record the first track, all Record Modes operate the same way. This is because until you define the length of a new Song, its End point is the default setting of Bar 8001, the maximum amount of Bars in a Song. The End point is referenced and modified in different ways depending on the RecMode.

The End point of a song is defined when:

- 1) The **Stop** button is pressed to end the recording of the first track. The new End point is aligned to the nearest downbeat of the (empty) Bar immediately following the last Bar you were recording when **Stop** was pressed.



- 2) **Stop** is pressed while recording any track past the previously set End point in Linear or UnLoop Record Mode. Again, the new End point is aligned to the downbeat of the (empty) Bar immediately following the last Bar you were recording when **Stop** was pressed.
- 3) The AutoOut Bar and Beat is set past current End point, after recording in Auto RecMode, and when the first track is recorded in Auto RecMode, the AutoOut Bar and Beat becomes the End point.
- 4) A new End point is entered in the EVENTS edit mode.
- 5) Using the Track edit functions Copy, Insert, and Delete to alter the Song's length.

The End point of a song is used as a loop point in Loop and UnLoop modes, and it defines the Fixed Length of a Song when you record in FixLen mode.

**Linear** - Record as if you had a nearly endless length of "tape".

**FixLen** - The song will not continue recording past the End point of the song when the RecMode is set to Fixed Length. Recording will automatically stop at the End point.

**Loop** - While RECORDING, the song will play the data between 1 : 1 and the End point over and over, allowing you to overdub in each pass if the Mode parameter on the MAIN SongMode page is set to Merge. Make sure the Mode parameter is set to Merge if you intend to overdub in Loop Record, or else each consecutive pass in Loop RecMode will erase the data recorded on the previous pass. While you are still recording in Loop mode, you may selectively erase individual note events by pressing and holding the **Enter** button and depressing the desired notes on the keyboard during the times you would like them erased. Once the Song is saved, you can not use this feature to erase individual notes.

**UnLoop** - When recording in UnLoop Mode, any existing tracks will be played back as if they were looping from Bar 1 : Beat 1 to the End point, but they are actually being re-recorded linearly over absolute Bars and Beats until you press **Stop**. UnLoop allows you to record a linear track over a short looping section without first having to copy the section over and over again to achieve a new desired Song length. The End point of the Song is extended to the downbeat of the (empty) Bar immediately following the last Bar you were recording when **Stop** was pressed.

For example, let's say you have a recorded a four bar drum loop and now want to record an eight bar bass line. This would be a situation where UnLoop would come in handy. While the drum track keeps looping, the bass track will record in linear fashion, and the end point will be moved to the point at which you press **Stop**. Actually, the drum track will also change. It will play through its loop twice, but while the information is repeating in the loop, it will be recorded to the track. So now if you look at the drum track, you will see information in bars 5-8 (a duplicate of the information in bars 1-4).

**Auto** - Set RecMode to Auto to punch-in record, (either in merge or erase mode), on a track between the Bars and Beats Defined in AutoIn and AutoOut. To punch in and continue recording until you press **Stop**, set the AutoOut point before the AutoIn point.

### PlayMode

The PlayMode setting, along with the End point, determine how the Song or Songs are played back.

**Linear** - Set the PlayMode to Linear to hear the current Song played only once, from the song position set in the Locate parameter to the End point. The Song will be returned to the Locate Bar and Beat when it reaches the End point.

**Loop** - The Loop PlayMode will loop the current Song from the End point back to Bar 1, Beat 1 continually during playback until **Stop** is pressed.

**List** - Song Objects in memory can be played back to back in numerical ID# order starting from the current Song followed by the Song with the next highest ID#. Once the current Song reaches its End point, immediately the CurSong parameter is updated to the next highest Song Object ID# and it will play from Bar 1, Beat 1 to its End point. When the Song with the highest ID# in memory has played through to its End point, the sequencer will stop and that Song will be the new "CurSong".

**Chain** - For every Song, there is a parameter called ChainTo found on the COMMON page that determines what other Song, if any, will immediately follow this Song's playback when PlayMode is set to Chain. When the current song ends, the ChainTo Song will replace the previous CurSong, and if its ChainTo parameter is set to any value other than "0 None", then the Chain PlayMode will continue playback with the next Song being chained. When the last Song in the chain has played through to its End point, the sequencer will stop and that Song will be the new CurSong.

The most common use for the Chain PlayMode is to construct a Set List of different Songs. Since the Songs chained together are played immediately after one another, it is recommended that you include a few measures of silence either at the end or the beginning of each Song in the Chain.

### **KeyWait**

KeyWait specifies whether the sequencer will wait for a Note event before going into PLAYING or RECORDING status. With the KeyWait On, press the **Play** soft button while the sequencer is STOPPED or REC READY and the new status, KEY WAIT, will flash in the Song Status Field until a key is played. You can override the KEY WAIT status by pressing the **Play** soft button twice.

### **Locate**

The Locate Bar and Beat will change in real time during play back and recording to reflect the Song's current position. It can be set to a Bar and Beat before (negative values) or during a Song. Once a Song's length is defined, the End point is the maximum value for the Locate parameter. If Locate is not set to 1 : 1, the count off, if any, is disabled during play back or recording.

This parameter is identical to the Locate parameter on the MAIN page.

### **AutoIn**

When you are in Auto record mode, AutoIn is the Bar and Beat when the recording will begin. If AutoIn is not set to 1 : 1, the count off, if any, is disabled. The AutoIn setting will not have an effect on recording unless the RecMode is set to Auto.

### **AutoOut**

When you are in Auto record mode, AutoOut is the Bar and Beat when the recording will stop. Set the AutoOut location earlier than the Bar and Beat defined as the AutoIn point in order to record to the very end of a Song. The AutoOut setting will not have an effect on recording unless the RecMode is set to Auto.

### **Input Quantize (Quant)**

This parameter determines how much Note events are moved towards grid locations upon the initial input of the events. If set to Off, no Quantizing will occur while you record, and the exact timing of your performance will be preserved during play back. If set to 100%, every recorded Note event will be aligned to the closest grid location, defined by the Grid setting. Input

Quantize is used to quantize your performance as you record it in. However, you may wish to record without quantization and go back and quantize at a later point. To do this, use the Quantize function in the Track editor, as described on page 12-36.

### **Grid**

This setting determines the size of the Input Quantize grid expressed as a fraction of a Bar with a 4/4 meter. Set Grid to 1/1 for whole note grid, 1/16 for sixteenth notes. All of the standard note durations and every fractional Bar divisions in-between (including triplets, e.g., 1/12=1/8tr, 1/24=1/16tr, etc.) are available as the size of the Input Quantize grid. You can select commonly used Grid values by double-pressing the Increment/Decrement (+/-) buttons.

### **Swing**

The Swing percentage is applied to the quantize grid. Zero percent swing is straight time, 100% produces a swing (triplet) feel. A positive Swing value determines how close every other grid location is moved to a point 1/3 of the way towards the next grid point. Negative Swing moves every other grid location closer to a point 1/3 of the way towards the previous grid point.

### **Sync**

The Sync parameter is used in conjunction with Clock. It controls transmission and reception of MIDI sync messages except actual clock. These are the messages it controls: Song Start, Song Stop, Song Continue, Song Select, and Song Position Pointer.

### **Clock**

Specifies the source clock as being internal or external. When the Clock is set to external, the K2vx will wait to receive MIDI clock data, via its MIDI In Port, from another device capable of generating MIDI clock data before playback and real time recording can begin.

### **Tempo**

The Tempo parameter, when set to Fixed, provides a tempo lock feature to override any real time tempo changes recorded into a sequence. If set to Auto, tempo changes will be respected.

### **CountOff**

Selects the number of bars of countdown, if any, before playback or recording starts. This works in conjunction with the click, so if the click is Off, the CountOff setting will have no effect. If the click is set to record only, then the CountOff will only happen when RECORDING.

### **Click**

The Click parameter controls the click behavior. Set to Off, there is no click, and consequently no CountOff. When it is set to On, a click is present during playback and recording. To have a click only while RECORDING, set the Click to Rec. The Cnt value means that there will only be a click during CountOff, if any.

### **ClickCh**

Specifies which MIDI channel will be used for the metronome click.

### **ClickPrg**

If click is in use, ClickPrg specifies which Program will be used as the metronome click's sound. The Click Channel will be locked on to this Program internally, and this program number will be transmitted via MIDI on the Click Channel to external any device(s) when playback or record is started.

### **ClickKey**

The ClickKey is the note to be used for the metronome click.

### **ClickVel**

ClickVel determines the attack velocity to be used by the metronome click. The first beat of each measure will be the click played at exactly this velocity level while the other clicks will be scaled to about 90% of this value as a way to provide an accent.

## **Soft Buttons on the MISC Page**

**Record**, **Play(Pause)**, and **Stop** work the same as they do on the MAIN Page, described above.

**New** selects "1 NewSong" as the current Song and jumps back to the SongMode : MAIN page. The tracks in the new Song will be empty, but all initial program, volume and pan settings, and all parameters in the MISC page remain set the same way they were in the previous Song.

**In/Out** provides a quick way to enter the Auto punch-in points in real time. This button has no effect when the sequencer is stopped, but pressing it once during playback (or while recording) will set the AutoIn point to the Bar, Beat, and Tick nearest to the time you pressed this soft button. Press InOut a second time to set the AutoOut point.

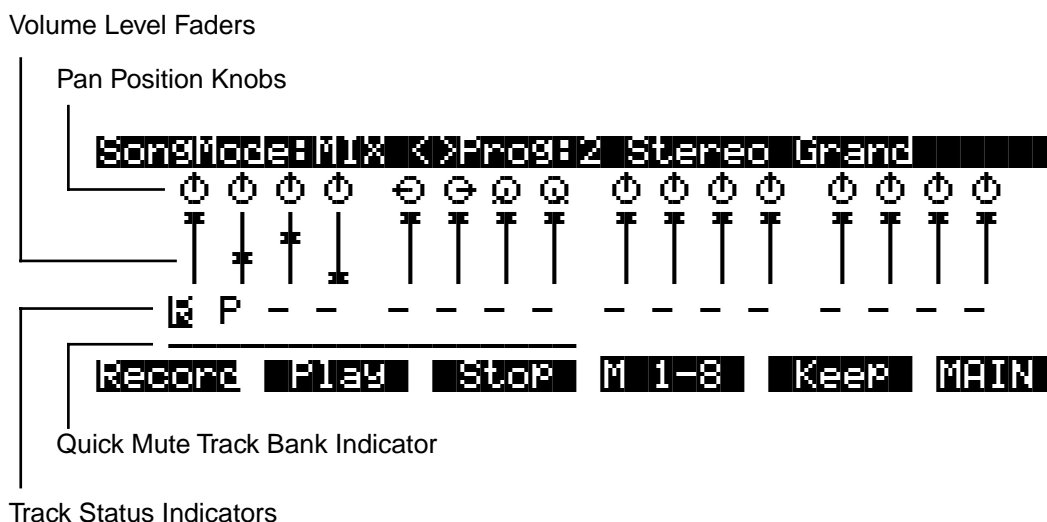
The AutoIn and AutoOut parameters only display bars and beats. However, the actual In and Out points will be precise to the Tick occurring at the time the In/Out button was pressed.

**MAIN** returns you to the MAIN Page.

## MIX Page

The MIX page allows you to set and keep new initial settings for all sixteen tracks' program changes, volumes, and pan positions. You can set these three parameters for each track and then press the **Keep** button, prompting a dialog that asks if you really want to update these settings. Press **Yes** to make the changes. Another way to set these initial settings is to press **Record** to put the sequencer into REC READY status, make any desired changes, then press the **Stop** soft button.

You can also record real-time changes by changing the value of the highlighted fader or knob while recording on that track. If you do make recording changes in real-time, though, make sure that the Record Mode is set to Merge, or else the data previously recorded on the track will be erased.



### Changing Programs on the MIX page

The top line of this page displays the Program number and name for the currently highlighted track. The different tracks are selected by moving the cursor with the Left and Right arrows (< / >). The Program for each track can be changed in the MIX page with the CHAN/BANK buttons. Pressing the CHAN/BANK buttons at the same time will jump to the next Bank of 100 Programs.

### Pan Position

Position the cursor over any one of the sixteen Pan Position "Knobs" on the display and turn the Alpha Wheel to change the panning for the selected track. The graphic display will move smoothly between the left and right settings and these changes can be recorded in real time. Use the numeric keypad to enter in a value between 0 (hard left) and 127 (hard right) if you want to have a track jump immediately to a new pan position. The default pan position is 64 (straight up).

### Volume Level

Position the cursor over any one of the sixteen Volume Level "Faders" on the display and turn the Alpha Wheel to change the MIDI Volume for the selected track. The graphic display will move smoothly, setting volume changes that can be recorded in real time. Use the numeric keypad to enter in a value between 0 (no volume) and 127 (maximum volume) if you want to have a track jump immediately to a new volume level. The default value is 127 (maximum volume).

### Track Status Indicators:

Using the Up, Down, Left, and Right cursor arrows to position the cursor onto a Track Status Indicator, you can toggle an empty track (-) into Record (R) with the Alpha Wheel or Increment/Decrement buttons.

Once a track contains data, it will have a (P) as a Track Status Indicator, and it will be played during playback. You now will be able to toggle between Play (P), Mute (M), and Record (R).

The track selected as the RecTrk will have an (R) in this field, designating it as the record enabled track. If the RecTrk is set to Mult, then all of the empty tracks will have Record (R) as their Track Status Indicator. If RecMode is not set to Mult and there isn't a track with an (R) in this field, the RecTrk parameter's value will be None. If you are in Multi record mode, and have turned all record enabled tracks (R) back to empty (-) so that there isn't a track with an (R) in this field, the RecTrk will remain set to Mult.

### Quick Mute Track Bank Indicator

This parameter appears as an underscore directly below the graphics for either the bank of tracks 1 -8 or tracks 9 - 16. It indicates which bank of eight tracks will respond to the Quick Mute feature, described with the **M 1-8/9-16** soft buttons, below.

## Soft Buttons on the MIX Page

**Record**, **Play(Pause)**, and **Stop** work the same as they do on the MAIN page, described earlier.

**M 1-8 / (M 9-16)**: The eight mode select buttons to the left of the screen on the front panel of the K2vx (PROGRAM - DISK) are used as Track Mutes when on the MIX page. For example, press the SETUP button to mute track 2; notice that its track status indicator changes to "M." Each of these buttons has a LED to indicate that the corresponding track is being muted, but since there are sixteen tracks and only eight buttons, this soft button will select, and display, the bank of eight tracks that can be muted in this fashion.

Press the **M 1-8** soft button to toggle the Quick Mute Track Bank Indicator under tracks 1 -8 or tracks 9 - 16, selecting which bank of eight tracks will respond to the Quick Mute feature.

**Keep**: If you have made any changes to the initial program, volume or panning of a track on the MIX page, press **Keep** to prompt this dialog:

Update initial Prog/vol/Pan?

Yes No

Press **Yes** if you are sure you want to update the initial program change, MIDI volume, and pan settings for tracks already containing data in the current Song, to the new values you just made on the MIX page. The settings will be modified and you are returned to the MIX page. Press **No** to abort the updating of the initial program change, MIDI volume and pan settings for the current Song.

**MAIN** returns you to the MAIN page.

## The Edit Song Pages

There are a few conventions shared by all of the EditSong pages (except the Arrange page). Displayed at the top of each EditSong page is the name of the page and the currently selected track(s). All of the values for the parameters found in any of the EditSong pages are saved in the Song Object.

### Soloing the Current Track

In all of these Edit pages you can **solo** the current track by pressing either the **Setup** or **Quick Access** mode buttons. This in effect mutes all other tracks in the current Song. Using the CHAN/BANK buttons allows you to select the current track for editing and/or soloing. You can select the track to solo even if you are already in the solo mode.

## Edit Song: COMMON Page

Press the Edit button on the front panel of the K2vx to display the Edit Song: COMMON page and begin editing a Song. This is where you will find parameters common to all tracks, such as tempo and time signature, control parameters for effects and arrangements, and soft buttons for switching to other Song edit modes.

```

editSong:COMMON          <>Track:1
Tempo      :120          StartStep  :1
TimeSig    :4/4          TempoControl:Song
EffectChan:1            TriggerChan:1
ChainTo    :0 None      TriggerCtl  :ON
TrackDest  :-----
DrumTrack  :-----
<more>  EVENT  TRACK  STEP  ARRANG  more>

```

PARAMETER	RANGE OF VALUES	DEFAULT
TEMPO	1 – 255 BPM	120
TIME SIGNATURE	1 – 99 / 1, 2, 4, 8, 16	4/4
EFFECTS CHANNEL	1 – 16, NONE	1
CHAIN TO	0 NONE, SONG ID	0 NONE
TRACK DESTINATION	-, L, M, X	-
DRUM TRACK	-, D	-
START STEP	NONE, 1 – 255	1
TEMPO CONTROL	SONG, ARRANGEMENT	SONG
TRIGGER CHANNEL	1 – 16, NONE	1
TRIGGER CONTROL	CONTROL SOURCE LIST	ON

The top line of this page displays the selected track or tracks. Select the available current track(s) for editing by using the CHAN/BANK buttons. Press both of the CHAN/BANK buttons together to select All tracks.

Although the current track information is displayed on the top line, the COMMON page's parameters are global settings for the song and do not directly affect individual tracks.

### Tempo

This is another place where the Song's initial tempo can be set or modified.

**TimeSig**

Affects the click, playback looping, and locate function as well as some editing operations. Does not change the recorded data.

**EffectChan**

If the FX Mode and FX Chan parameters in Effects mode are set to Auto and Current respectively, this parameter specifies which channel will control effects while in SongMode and during playback of a Song. In this situation, the effect assigned to the program (in the Program Editor) currently on the channel designated as the EffectChan will be heard.

The EffectsChan is selectable per Song. See "Understanding FX Mode and FX Channel" on page 9-2 for more information.

**ChainTo**

The ChainTo setting is only used when the PlayMode parameter on the MISC page is set to Chain. Another Song object is selected as the ChainTo value and will playback immediately after the current Song has played for its entire length.

## Parameters used with the Arrangement Feature

The next four parameters, as well as the Drum Track parameter, are used in conjunction with the Arrangement feature, which you access by pressing the ARRANG soft button. This feature allows you to create a song by arranging it in a series of sections, called Steps. You can save the various sections of your sequence as separate Songs, then assemble them using the Arrangement editor. This method allows you to save memory, since you can repeat steps without having to duplicate the actual song data.

Another great feature within the arrangement editor gives you the ability to trigger Steps by striking a specific note or group of notes on your controller. See page 12-46 for more information on arrangement features.

**StartStep**

The value for this parameter determines which Step in this Song's Arrangement is played first. If set to None, the playback of the Arrangement data, using the transport soft buttons, will be disabled, but Steps can still be triggered from Note events.

**TempoControl**

When the current Song is comprised of other Songs entered as Steps in the Arrangement editor, the TempoControl parameter determines whether the current Song's tempo setting or the arranged Songs' tempo settings will be the tempo(s) used for the playback and recording of the Arrangement.

Set TempoControl to Song if you want to use the current Song's tempo as the master tempo. All of the Steps' Songs will playback at the same tempo, ignoring the tempos originally set for each Song when you recorded or edited them.

Use the Arrangement setting for TempoControl if you would like the current Song (containing arrangement data) to playback and record at the tempos originally defined in each Arrangement Step's Song, ignoring its own tempo setting. The master tempo will change to the tempo of the current Step's Song

**TriggerChan:**

This parameter controls the MIDI channel used to trigger arrangement Steps via key presses as defined in the Arrangement editor.



### TriggerCtl

This parameter specifies the global control source used to enable the triggering of Arrangement Steps via key presses as defined in the Arrangement editor.

### TrackDest

```
TrackDest :LLL- M--- -M-- ---x
DrumTrack :D--- ---- -D-- ----
```

The MIDI data on each track has a destination assignment selectable in the TrackDest field. There are four possible indicators:

- (-) This is the default setting for all tracks in a new Song. A hyphen (-) means that MIDI data on the track is transmitted both locally to the K2vx, and out the MIDI Out Port (on the track's channel) to any external devices.
- (L) An (L) means that the track's MIDI data will only be transmitted locally to the K2vx's internal sound generator. None of the data on that track will be sent to the MIDI Out Port.
- (M) An (M) means that the track's MIDI data will only be transmitted out to external devices via the MIDI Out Port on the back of the K2vx.
- (x) Select the (x) to disable the transmission of the track's MIDI data to either of the destination assignments, (L)ocal or (M)IDI.

### DrumTrack

Any of the Song's tracks can be defined as "Drum Tracks" so that their Note events do not get transposed when a transposition is applied in the Arrangement editor.

This feature is particularly useful when a drum kit Program (or any other non-pitched Program) is used in a Song, assigned to a Step of an Arrangement, which is being triggered over a range of keys, and you want the sounds produced by each note number in that Program to be preserved in each transposition. If in the Song being used as a Step in an Arrangement (NOT the Arrangement Song itself), there is a (D) designating the track playing the drum Program as a "Drum Track", the originally recorded Note events on that track will remain unchanged.

The Drum Tracks settings do not have any effect on edits made in the EditSong : TRACK mode. Any tracks defined as "Drum Tracks" will be transposed when a transposition is applied to these tracks from the EditSong : TRACK mode.

## Soft Buttons on the Edit Song: COMMON Page

**EVENT** accesses an Event List style editor. From the EditSong : EVENT page, you can scroll through, modify, add, or delete any or all of the tracks' MIDI events.

**TRACK** brings you to the EditSong : TRACK page. This page accesses useful track based edit functions. There is a selectable edit function that can be applied to the selected track or all tracks in your Song.

**STEP** brings up the EditSong : STEP page, from which non-real time note / rest entry is performed. The term STEP is used here to refer to Step recording, and does not refer to the Steps on the Edit Song:ARRANGE page.

ARRANG enters the EditSong : ARRANG page, gaining access to a group of arrangement specific parameters.

## Edit Song: EVENT Page

Every type of recorded MIDI event is visible from this page. You can view and change these events if necessary.

Location	Bar:Beat:Tick	Event Type and Value
1:1	1:1.000	CTRL BANK 0
1:1	1:1.000	PCHG 64
1:1	1:1.000	CTRL VOL 127
1:1	1:1.000	CTRL PAN 64
1:1	1:1.000	C 3 v 98 ^ 93 0.356
2:1	2:1.000	END

Cut Copy Paste View AllOn Done

To scroll through the events, make sure the first column bar and beat is highlighted. Use the wheel, the up and down cursor buttons, or the + and - buttons. As you scroll through the events, each event is executed by the sequencer. In the case of note events, you will hear the note played, although the duration will be short. If you have scrolled through a Sustain (controller 64) message with an On value then you will hear the note sustain as if the sustain pedal was depressed. The note will continue to sustain until you scroll through a Sustain message with a value of Off.

You can also jump directly to a specific bar and beat by typing the bar number and beat number, then pressing ENTER. Keep in mind if you have controller or program data previous to the point that you jump to, those events may not have been executed and you may hear unexpected results. For instance, if you have program changes at bar 1 and bar 8, then if you jump from bar 1 to bar 9 any notes you scroll through will be played with the program change from bar 1.

The channel of the selected Event is displayed on the top line of the page. When all of the Tracks are being viewed in the Event List, the Track of the selected Event is displayed along with its channel. Use the CHAN/BANK buttons to select an active track to view and edit the MIDI events recorded on it. You can view the data on all tracks by pressing both CHAN/BANK buttons at the same time. As you scroll through each Event, the track, in addition to the channel, for the selected Event will also be displayed.

### Location

The first column represents the Bar and Beat Locations of the different events in a Song. Scroll through the events on the selected track(s) with the Alpha Wheel or enter in a specific Bar and Beat on the numeric keypad to jump to events occurring on that Beat. A quick way to jump to the End point in a track is to press 9999 and then Enter on the numeric keypad.

### Bar, Beat, and Tick

Bar, Beat, and Tick are editable parameters for each Event. They determine when an Event happens relative to the other Events within the Song. To enter values with the numeric keypad, first position the cursor in this column and then simply enter up to three digits if you only want

to adjust the Tick value, four or five digits in order to change the location to a new Beat and Tick within the current Bar, or enter in the complete Bar, Beat, and Tick value to relocate the event to another Bar. No punctuation is necessary when entering any of these values.

### Event Type and Value

The Event Type field displays the MIDI event type at each Event List location in the Song. With the exception of a Note event, which is displayed as the actual note name (e.g., C#4), an event type cannot be changed to another event type. You can also change controller numbers and values on the EVENT page.

The PCHG, BEND, MPRS, SYSX, TMPO, and END events will show their values in the display field directly to the right of the Event Type field. All of their values can be edited. Here are the ranges of values for each event type:

<b>Program Change (PCHG)</b>	0 - 127
<b>Pitch Bend (BEND)</b>	(-8192) - 8191
<b>Mono Pressure (MPRS)</b>	0 - 127
<b>System Exclusive (SYSX)</b>	The message in hexadecimal form
<b>Tempo (TMPO)</b>	1.00 - 255.00 bpm
<b>End Point (END)</b>	Any Bar : Beat : Tick setting

Note events have four editable values: Note Number, Attack Velocity (indicated by a "v"), Release Velocity (indicated by a "^"), and Note Duration.

<b>Note Number</b>	C-1 – G9
<b>Attack Velocity</b>	v1 - v127
<b>Release Velocity</b>	^1 - ^127
<b>Note Duration</b>	Beats : Ticks

Controller events have two editable values: Controller Type and Controller Value. Defined controllers are referred to by their names.

<b>Controller Type</b>	0 - 127
<b>Controller Value</b>	0 - 127

## Soft Buttons on the Edit Song: EVENT Page

**Cut** removes the currently selected event from the Event List and temporarily stores it in a memory buffer so that you can immediately Paste it into a new location.

**Copy** makes a duplicate of the currently selected event and temporarily stores it in a memory buffer so that you can immediately Paste it into a new location.

**Paste** inserts the most recent cut or copied event into the Event List at the currently selected Bar : Beat : Tick location. The pasted event will share the same location with the event that already existed at that location in the Event List, but it will appear before the pre-existing event.

**View** brings you to the EVENT : View page where you can set the view filter parameters for the Event List.

## Song Mode

Edit Song: EVENT Page

```
EVENT:View
Notes      :On      Volume      :On
PitchBend  :On      Pan         :On
ModWheel   :On      Bank/Prog   :On
Data       :On      Tempo       :On
Sustain    :On      SysEx      :On
Pressure   :On      Other      :On
Toggle AllOff AllOn Done
```

The EVENT edit page can be set for you to view all, some, or just one type of MIDI event on a track by adjusting which events are on or off in the EVENT : View page. If you turn an event Off for viewing, then when you return to the Edit Song: EVENT page, the events will not be seen and they will not be executed as you scroll through the list. For example, if you turn sustain Off, then as you scroll through the list none of the notes normally affected by sustain will be sustained. However, if you exit the editor and play the Song, all events will be executed regardless of the View filter settings.

When all of the parameters in the EVENT : View page are set to On, the Event list in the EditSong : EVENT page will display all of the recorded MIDI data on the selected track(s). If any of these view parameters are set to Off, the **View** soft button in the EditSong : EVENT page will be in brackets (as shown below) indicating that you might not be seeing all of the recorded MIDI data on the selected track(s).

```
Cut Copy Paste [View] AllOn Done
```

Press the **Toggle** soft button to switch all View filter parameters to the opposite of each one's currently set value. **AllOff** sets all View filter parameters to Off. The only item viewable in the EditSong : EVENT page will be an END point. **AllOn** sets all View filter parameters to On. This enables you to view all of the MIDI events that are on any or all tracks. The **Done** soft button in the EVENT : View page will return you to the EditSong : EVENT page. The view filter settings will be applied to the Event List.

**AllOn** on the EditSong : EVENT page sets all View filter parameters to On. This enables you to view all of the MIDI events that are on any or all tracks. The **View** soft button will no longer be in brackets, indicating that all recorded event types are being listed.

**Done** on the EditSong : EVENT page returns to the EditSong : COMMON page.

## Edit Song: TRACK Page

This page allows you access to useful track-based edit functions. These functions are:

- Erase
- Copy
- Bounce
- Insert
- Delete
- Quantize
- RefQuant
- Shift
- Transpose
- Change
- Thin
- Remap
- Grab

For each function, there is a set of parameters to control how the function operates, and on what region of the selected track(s). As usual, the top line of this page displays the selected track or tracks. Select the available current track(s) for editing by using the CHAN/BANK buttons. Press both of the CHAN/BANK buttons together to select All tracks.

Below is an example of the Edit Song: Track page for the Bounce function.

```

                                     Region/Criteria Window
editSong:TRACK          <>Track:1
Function:Bounce        From : 1:1 To: 2:1
DstTrack:1            Events:Notes
Mode :Erase           LoKey :C -1 Hi:G 9
                       LoVel :1 Hi:127

Locate : 1:1
FromTo Play Stop Go Done
  
```

You will notice that the screen is divided into two halves, with the right half being its own "window". This is called the Region/Criteria window. The parameters in this window are used to select the range of bars as well as which types of events will be edited with using the chosen function. The parameters in this window will generally be the same for most functions. However, for certain functions, certain parameters may not apply. For example, Quantize and Transpose apply only to notes, while Remap applies only to Controllers. In addition to the Region/Criteria window parameters, the Locate parameter is also found on each function. Since these parameters are common to the various functions, we will define them first. Then we'll describe the individual functions along with the parameters specific to each, which are normally found on the left side of the screen. Two functions, (Quantize and Reference Quantize) have a unique parameters in their Region/Criteria windows. We'll describe those parameters along with the functions.

Once you've chosen a function and set the parameters to your liking, press **Go**. This executes the editing function. You can then play the sequence to hear the results of your edit. If you don't like your edit, simply exit the editor and press **No** when you are asked if you want to Save. If you do like your edit, you can press **Done** and use the more buttons to get to **Save**, or just exit the editor and save the changes. Or, you can go to another edit function. Keep in mind though, that if you choose to perform more than one edit without saving, and you are not satisfied with one of the changes you make, you will have to exit the editor without saving and then redo each of the changes you made. That's why it's usually best to save after each successful edit.

## Common Parameters for Edit Song: Track Functions

### ***Locate***

This parameter is available in every TRACK edit function.

The Locate Bar and Beat will change in real time during play back and recording to reflect the Song's current position. It can be set to any Bar and Beat, including negative values. Play back begins at, and Stop resets the Song to the Locate Bar and Beat.

## Region / Criteria Window Parameters

### ***From and To***

From and To are available in most TRACK edit functions to define a range of time on the selected track(s).

The From value defines the first Bar and Beat in a range of time selected for editing. Although From is always expressed in Bars and Beats, and will be aligned exactly on the Beat when adjust manually, it is possible for From to represent a value with finer resolution when a range of time is defined in real time using the **FromTo** soft button.

The To value defines the Final Bar and Beat in a range of time selected for editing. Although To is always express in Bars and Beats, and will be aligned exactly on the Beat when adjust manually, it is possible for the To value to represent a value with finer resolution when a range of time is defined in real time using the **FromTo** soft button.

### ***Events***

Any and all types of MIDI events are available for editing, selectable in this parameter. Some events will provide you with settings for a range of values, or other MIDI event specific criteria. Available Values are: All, Notes, Controllers, MonoPress, PitchBend, ProgChange, PolyPress, SysEx, and Tempo

#### **When Events is set to: All**

All MIDI events on the track(s) you are editing, that occur in the region of time between the From and To settings, will be affected by the edit function.

#### **When Events is set to: Notes**

Note Number and Key Velocity ranges can be set for Note events.

### ***LoKey***

Determines the lowest note in a range of notes to be affected. This can be set to any MIDI note value; the default is C-1.

### ***Hi(Key)***

Determines the highest note in a range of notes to be affected. This can be set to any MIDI note value; the default is G9.

### ***LoVel***

An attack velocity range can be specified as criteria for selecting Note events for editing. The LoVel parameter sets the lowest velocity a Note needs to have in order to be edited. Notes on the selected track(s) with a attack velocities lower than the LoVel will not be affected by the edit. The available values are 1 – 127; the default is 1.

**Hi(Vel)**

The Hi(Vel) parameter sets the highest attack velocity a Note needs to have in order to be edited. Notes on the selected track(s) with attack velocities higher than the Hi(Vel) will not be affected by the edit. The available values are 1 – 127; the default is 127.

**When Events is set to: Controller**

If your Song contains any MIDI Controller data, the Ctl parameter is used to select particular controller data to be edited. Refer to the Control Source list in Chapter 6 of the *Reference Guide*.

**LoVal**

You may further specify a particular range of values to edit by setting a high and low value. LoVal will define the lowest modifiable value in the selected controller's recorded data. Value ranges are not definable when Ctl is set to All. Available Values are 0 - 127.

**Hi(Val)**

Hi(Val) will define the highest modifiable value in the selected controller's recorded data. Value ranges are not definable when Ctl is set to All. Available Values are 0 - 127.

**Soft Buttons on the Edit Song: Track Page**

**FromTo** is a quick way to define the region of time you intend to edit. There are a couple of ways to use this feature when the sequence is playing back in real time, and both ways will set the temporal boundaries of the region to a finer resolution than Bars and Beats.

One way is to first position the cursor over the From field in the *Region / Criteria Window* and then press the **Play** soft button. At this point, every time you press **FromTo**, the From Bar and Beat will reflect a location in the Song at each time the button press was made. Position the cursor over the To field and set the To location in a similar fashion.

When the cursor is positioned somewhere other than over the From or To fields, the function of the **FromTo** soft button is slightly different. Now, once you start the sequence playing back, and you press **FromTo** before the current To Bar and Beat location, you will set a new From location. Without stopping the sequence and starting it again, press **FromTo** a second time, and if it was pressed at a point after the From location, that point in time will represent a new To location. If the second **FromTo** press happens before the current From Bar and Beat, that point in time becomes the new From location, and the old From location becomes the new To.

To sum up this functionality, when this button is pressed while the Song is playing, the K2vx will know which field, (the From or the To), to update based on the relative position, (before or after the current settings), when the button was pressed.

**Play** will start the playback of the Song from the Bar and Beat set in the Locate parameter. When the Song is playing, this soft button becomes **Pause**.

**Stop** stops the playback of the Song and return to the Bar and Beat set as the Locate value.

**Go** performs any of the Track-based edit functions described above.

**Done** will return you to the EditSong : COMMON page.

## Edit Song: Track Functions – Erase

This function erases specified events from a region of time, but it doesn't delete the region of time. The result is like erasing a section of recording tape. If you want to completely remove a segment and shorten the length of the track, you can do it with the Delete function.

```
edit song: TRACK <> track: 1
Function: Erase
From : 1:1 To: 2:1
Events: SysEx

Locate : 1:1
FromTo Play Stop Go Done
```

## Edit Song: Track Functions – Copy

Use the Copy function to duplicate the selected events from the current track and place them in the same track or on another track, either merging with or overwriting existing data.

```
edit song: TRACK <> track: 1
Function: Copy
DstTrack: 1
Location: 1:1
Mode : Merge
Times : 1
Locate : 1:1
FromTo Play Stop Go Done
Events: Controllers
Ctl : MWheel
LoVal : 0 Hi: 127
```

If you do not want to copy all of the MIDI events in the defined range of time on the current track, use the Events parameter in the Region/Criteria window to select a specific MIDI event type you would like the edit function to affect. Some event types provide you more criteria selection parameters. It is often a good idea to set Events to Notes when copying, and then add any necessary controller or other data to the track at a later time.

**DstTrack:** 1 - 16 / All

Select a destination track for the copied events with the DstTrack parameter. All selected events described in the Region / Criteria Window will be placed in the destination track(s) at any Bar and Beat you specify.

If the currently selected track is All tracks then the destination track will be All tracks as well.

No matter what channel the current track (source track) is set to when you use the copy function, the Events will be played on the destination track's channel.

**Location:** 1 : 1

Specify a Bar and Beat location in the destination track where the copied data will be placed with the Location parameter. If the length of the copied region extends from the Location point beyond the Song's existing End point, a new End point is defined.

**Mode:** Merge / Erase



The Mode setting determines whether the copied events merge with, or erase existing events on the destination track from the location point to the end of the copied region.

**Times:** 1 - 127

The value selected for the Times parameter determines how many copies of the selected region are placed, one after another, in the destination track.

## Edit Song: Track Functions – Bounce

Use the Bounce function to move the selected events from the current track to another track, either merging with or overwriting existing data on the destination track. The Bounce function differs from the Copy function in that the original data is not preserved in the original track. As on a multi-track tape recorder, Bounce will always put the data in the same timeline on the new track that it was on the old track.

```

editSong: RACK                                <> track:1
Function: Bounce                               From : 1:1 To: 2:1
DstTrack: 1                                   Events: Notes
Mode : Erase                                  LoKey : C -1 Hi: G 9
                                              LoVel : 1 Hi: 127

Locate : 1:1
FromTo Play Stop                               Go Done
  
```

**DstTrack:** 1 - 16

Select a destination track for the events to be moved to with the DstTrack parameter. All selected events described in the Region / Criteria Window will be placed in the destination track at the data's original location.

No matter what channel the current track (source track) is set to when you use the bounce function, the events will be played on the destination track's channel.

**Mode:** Merge / Erase

The Mode setting determines whether the bounced events merge with, or erase existing events on the destination track from the location point to the end of the copied region.

## Edit Song: Track Functions – Insert

The Insert function is used to add blank time to the current Song, modifying the Song's End point appropriately. The Insert function will affect all tracks. This is similar to splicing a piece of blank tape to an existing segment of recording tape.

```

editSong: RACK                                (Effects all tracks)
Function: Insert
Location: 1:1
Amount : 1:0

Locate : 1:1
FromTo Play Stop                               Go Done
  
```

## Song Mode

Edit Song: Track Functions – Delete

**Location:** 1 : 1

The insertion point for the blank time being added is selected as a Bar and Beat Location value. Events that occurred at or after this Bar and Beat, before you insert time, are not erased when you perform this function, rather they are offset by the length of the blank time being added to a Bar and Beat later in the Song.

**Amount:** 1 : 0

The length of the blank time being added is defined as a number of Bars and Beats in the Amount parameter.

There are no Region / Criteria Parameters available for the Insert function.

## Edit Song: Track Functions – Delete

The Delete function is used to remove a region of time from the current Song. This function is different from the erase function because not only does it remove the events from the selected time, it will delete the entire selected range of time from the Song, modifying the Song's End point appropriately (on all tracks). This is similar to cutting a section out of a tape and splicing the ends.

```
edit song: [RHCK] (Affects all tracks)
Function: Delete From : 1:1 To: 2:1

Locate : 1:1
FromTo Play Stop Go Done
```

## Edit Song: Track Functions – Quantize

Use the Quantize function to adjust the timing of Note events. Keep in mind that only Note events are quantized; other types of events, such as controllers, are not quantized.

```
edit song: [RHCK] (<> pack: 1
Function: Quantize From : 1:1 To: 2:1
Quant : 100%
Grid : 1/8 LoKey : C -1 Hi: G 9
Swing : 0% LoVel : 1 Hi: 127
Shift : 0.000 Release: No
Locate : 1:1
FromTo Play Stop Go Done
```

**Quant:** Off / 1% – 100%

The Quantize parameter determines how much the selected Note events are moved towards grid locations. If set to Off, no aligning of previously recorded Notes to grid locations will occur. If set to 100%, every recorded Note event will be aligned to the closest grid location, defined by the Grid setting. Notes will be moved to a position half way between the grid location and the original Note event location if Quant is set to 50%.

**Grid:** 1/1 – 1/384

This setting determines the size of the Quantize grid, expressed as a fraction of a Bar with a 4/4 meter. Set Grid to 1/1 for whole note grid, 1/16 for sixteenth notes. All of the standard note durations and every fractional Bar divisions in-between are available as the size of the Input Quantize grid. The double button press for quickly selecting Grid values is pressing the Increment/Decrement at the same time.

**Swing:** -99% – 125%(defaults as 0%)

The Swing percentage is applied to the quantize grid. Zero percent swing is straight time, 100% produces a swing feel (triplet feel). A positive Swing value determines how close every other grid location is moved to a point 1/3 of the way towards the next grid point. Negative Swing moves every other grid location closer to a point 1/3 of the way towards the previous grid point.

**Shift:** -26.020 – 26.020 (Beats and Ticks, 480 Ticks = 1 Beat)

In addition to quantizing the selected note events to specified grid locations in varying amounts, the Quantizing function allows you to offset the original note locations forward and backward in time any number of ticks (1/480th of a Beat) up to @ 26 Beats, before aligning them to Grid locations. Shift is used to compensate for any notes played too early or late.

### Region / Criteria Window

**Release:** Yes / No

Set the Release parameter to Yes if you would like each quantized note event's Note Off message to be aligned to the Grid location nearest to the time the key was originally released.

## Edit Song: Track Functions – Reference Quantize

The Reference Quantize function is similar to the Quantize function in that it aligns Note events to a grid. The difference is that the grid locations are not mathematically perfect divisions of a Bar. Instead, Reference Quantize defines the grid based on the timing of note events from a previously recorded reference track.

```

editSong: [RHJK] <> track:1
Function:RefQuant From : 1:1 To: 2:1
RefSong :1 NewSon LoKey :C -1 Hi:G 9
RefTrack:1 LoVel :0 Hi:127
Timing :100% Width :1/8
Velocity:Off
Locate : 1:1
FromTo Play Stop Go Done
  
```

**RefSong:** 1 NewSong

The previously recorded reference track can be selected from any Song object currently residing in the RAM memory. Select the Song containing the desired reference track as the RefSong value.

**RefTrack:** 1 – 16

The RefTrack value is set to the track number of the desired Reference Track in the Reference Song. The timing of this track's note events is used as the Grid that other tracks will reference when being Referenced Quantized.

## Song Mode

### Edit Song: Track Functions – Shift

**Timing:** Off / 1% – 100%

The Timing parameter determines how much the selected note events are moved towards grid locations. If set to Off, no aligning of previously recorded Notes to grid locations will occur. If set to 100%, every recorded note event will be aligned to the closest grid location, defined by the timing of note events from the reference track. Notes will be moved to a position half way between the grid location and the original Note event location if Quant is set to 50%.

**Velocity:** Off / 1% – 100%

In addition to referencing the timing of note events on the reference track, you may also scale the attack velocities of the note events being quantized to velocity values closer or identical to the velocities played on the reference track.

Leave the Velocity parameter set to Off if you just want to reference the note event timing when reference quantizing. Set Velocity to 100% in order to have the note events being quantized to have velocity values matching those of the note events on the reference track. A setting of 50% will change the velocities to values half way between the values originally played and the values of the velocities on the reference track.

### Region / Criteria Window

**Width:** 1/1 – 1/384

The Width setting determines the duration of a window of time centered around each of the referenced grid locations. If a note event on the track being quantized happens during this window of time, it will be moved closer to the referenced grid location according to the Timing percentage. Note events occurring outside this window of time remain unquantized.

## Edit Song: Track Functions – Shift

The Shift function allows you to offset the existing MIDI events forward or backward in time any number of ticks (1/480th of a Beat) up to approximately 26 Beats. This function does not affect the End point.

Events can not be shifted beyond the End point or before Bar 1 : Beat 1. The events can be shifted only as far as these temporal boundaries. All events that can't be shifted the full Ticks amount will be placed at the boundary location.



**Ticks:** -26.020 – 26.020 (Beats and Ticks, 480 Ticks = 1 Beat)

The Ticks parameter specifies the number of Beats and Ticks that the MIDI events, from within the selected region, are moved forward or backward in time relative to their original locations.

## Edit Song: Track Functions – Transpose

Use the Transpose function to change the MIDI Note numbers of the selected Note events.

```

EditSong: TRACK                                <> track:1
Function: TransPos                             From : 1:1 To: 2:1
Semitone: 12ST                               LoKey :C -1 Hi:G 9
                                              LoVel :1 Hi:127

Locate : 1:1
FromTo Play Stop Go Done

```

Semitone: -128ST – 127ST

An increment of one semitone represents a change of one MIDI Note number. Note events are only transposable within the range of MIDI Note numbers from zero to 127.

## Edit Song: Track Functions – Change

The Change function is used to modify attack velocities, release velocities, or the values of any existing controller data on the current track. A static change of values can be made as well as having the change take place over a region of time.

Change can not modify or add data that doesn't exist on the current track. If you hear note events played back on a track, then you know there is an attack and release velocity value for each one, and the effect of the Change function can usually be easily detected. Controller values are sometimes more difficult to change since there can be inconsistent gaps of time in-between each controller event.

```

EditSong: TRACK                                <> track:1
Function: Change                               From : 1:1 To: 2:1
Scale : 100%                                  Events: Velocity
Offset : 0                                     LoKey :C -1 Hi:G 9
Mode : Constant                               LoVel :1 Hi:127

Locate : 1:1
FromTo Play Stop Go Done

```

Scale: 0% – 20000%

The selected velocity or controller events' values can be changed to a percentage of the original values determined by the Scale parameter. A setting of 100% has no affect. Values are scaled lower with a Scale percentage set from 0% to 99%. Low values can be set higher using a Scale percentage above 100% on up to 20,000%, although the maximum value of 127 can not be exceeded for any velocity or controller type.

Offset: -128 – 127

Offset can be used alone or in conjunction with Scale to add or subtract a set amount to or from the original (or scaled) values. Values for velocities can not be less than 1 or greater than 127. Values for controllers can not be less than 0 or greater than 127.

As an example, to set all Velocities to a value of 55, you would set Scale to 0% (multiplies all original values by zero) and set Offset to 55 (adds 55 to the product of the Scale parameter).

**Mode:** Constant / PosRamp / NegRamp

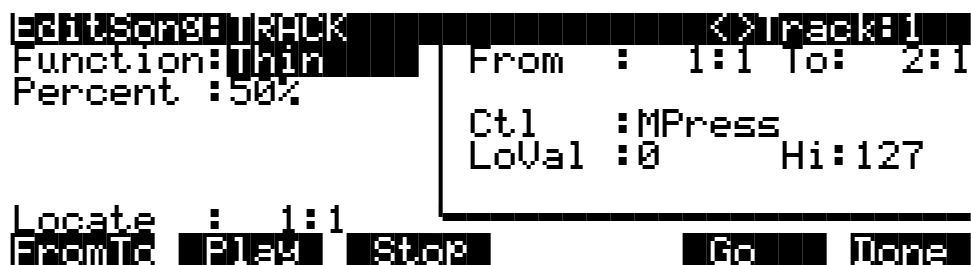
Set Mode to Constant to have values modified in a uniform fashion, as determined by the Scale and Offset settings, for the entire selected region of time and range of values.

When the Change function is applied with Mode set to Positive Ramp, the selected velocity or controller values will gradually change over the region of time, defined by the locations set for the From and To parameters, from the original value to the new value determined by the Scale and Offset settings. The first events being modified within the region will have little or no change from their original values. The amount of Scale and Offset applied will increase as the Song approaches the Bar and Beat defined in the To parameter, where the full amount of described change will occur.

You can set Mode to Negative Ramp to achieve the opposite dynamic effect of Positive Ramp. Negative Ramp works in the same way, but the amount of Scale and Offset applied will decrease from the full amount of change described by Scale and Offset to little or no change as the Song approaches the Bar and Beat defined in the To parameter.

## Edit Song: Track Functions – Thin

Use the Thin function to reduce the number of actual controller events used for any controller type on the current track. Sometimes a controller can produce the same noticeable effect by using fewer events to describe its change, which in turn will save memory and possibly prevent timing delays caused by an over abundance of MIDI data.



**Percent:** 0% – 100%(defaults as 50%)

The Percent value determines how much effect the Thin function will have on a stream of controller data. With this value set to 0%, there will not be any reduction of data when Thin is applied. Set Percent to 100% to eliminate most of the specified controller's events from the current track. Even at 100%, certain controller messages will not be eliminated; these include all pitch bend messages with a value of 0 and the initial settings of some controllers, such as volume, pan, etc.

## Edit Song: Track Functions – Remap

Use the Remap function to apply the values of any one type of controller data, already recorded on a track, to another controller type. The effect the real time changes of the “Old” controller had will be replaced by the effect the “New” controller has by using the exact same controller values.

```

EditSong: TRACK <>Track:1
Function: Remap      From : 1:1 To: 2:1
Old      : MWheel
New      : Volume

Locate   : 1:1
FromTo   Play Stop  Go Done

```

**Old:** Controller Codes (0 – 120)

Defined controllers are referred to by their names.

The “Old” controller is set to the controller type that you wish to remap. This controller data must already exist on the current track in order to apply it to the “New” controller type.

**New:** Controller Codes (0 – 120)

Defined controllers are referred to by their names.

The “New” parameter is set to the controller code you wish to have use the existing values, once used by the “Old” controller, to produce a different effect.

## Edit Song: Track Functions – Grab

Grab is similar to the Copy function, except that the Grab function allows you to copy selected data from tracks that exist in other Songs in memory.

```

EditSong: TRACK <>Track:1
Function: Grab      From : 1:1 To: 2:1
SrcSong  : 1 NewSong Events: Controllers
DstTrack : 1       Ctl  : MWheel
Location : 1:1     LoVal : 0      Hi: 127
Times    : 1
Locate   : 1:1
FromTo   Play Stop  Go Done

```

**SrcSong:** 1 NewSong

The Source Song parameter is set to the ID# and name of the Song in RAM memory that contains the desired track data you wish to grab in order to use it in the current Song. The source track is determined by the <>Track parameter displayed on upper right hand side of the page, selectable with the CHAN/BANK buttons.

**DstTrack:** 1 – 16 / All

Select a destination track for the grabbed events with the DstTrack parameter. All selected events from the source Song and track described in the Region / Criteria Window will be placed in the destination track(s) at any Bar and Beat you specify.

## Song Mode

---

### Edit Song: Track Functions – Grab

If the currently selected track is All tracks then the destination track will be All tracks as well.

No matter what channel the current track (source track in the source Song) is set to when you use the grab function, the Events will be played on the destination track's channel.

**Location:** 1 : 1

Specify a Bar and Beat location in the destination track where the grabbed data will be placed with the Location parameter. If the length of the grabbed region extends from the Location point beyond the Song's existing End point, a new End point is defined.

**Times:** 1 – 127

The value selected for the Times parameter determines how many copies of the selected region are placed, one after another, in the destination track.



## Edit Song: STEP Page

```

Edit Song: STEP <> track:1
DubMode : Erase
Duration: 1/16
Modifier: None
GateTime: 75%
Velocity: Played
Locate : 1:1.000
Play Stop <Back Step> Done

```

PARAMETER	RANGE OF VALUES	DEFAULT
DUBMODE	MERGE, ERASE	MERGE
DURATION	1/1 – 1/64	1/8
MODIFIER	NONE, TRIPLET, DOTTED, DOTDOT	NONE
GATETIME	0 – 250%	75%
VELOCITY	PLAYED, 1 – 127	PLAYED
LOCATE	BAR:BEAT.TICK	1:1.000

The Step Editor allows for non-real time entry of note events and rests of varying durations and attack velocities at any location within a Song. The Step Editor can be used to enter the first note events into a new Song, but it will not set the Song's End point. Initial program changes, volume and pan settings are remembered as if you recorded the first track using one of the real time recording modes. The End point will be modified if tracks already containing data are Step edited beyond the current End point.

The top line displays the currently selected RecTrk and can then be switched to any other track. When All tracks are selected, the Step edits are performed on the currently selected RecTrk.

### Recording with the STEP editor

Recording using the step editor is easy. Set the parameters to your desired settings and make sure the Locate parameter is set to the bar and beat at which you wish to start recording. Then just strike a key. The note you play will be displayed in the window on the right along with its velocity or duration. If the Velocity parameter is set to Played, then it will reflect the actual velocity you played. Otherwise, it will be specific amount determined by that parameter. The duration of the note is determined by a combination of the Duration and Gate Time parameters.

Once you release the key, the Locate parameter will advance in time, by an amount determined by the Duration parameter. If you want to enter a rest, press the **Step>** soft button without striking a key, and once again the Locate parameter will advance the selected amount.

You can enter note events longer than the selected duration by holding a key and pressing the **Step>** button. You will see the duration of the current note change in the window on the right. This can be a quicker method of entry than changing the duration parameter if you need a note that is twice or three times as long as the duration. For instance, if the duration is set to 1/8, you can quickly enter 1/4 note by holding the note and pressing **Step>** once, or enter a whole note by pressing the **Step>** button three times. Note that if you are using this method and the Gate Time parameter is set to less than 100%, it will only be applied to one of the steps. For example, if you have a Duration of 1/8 and a Gate Time of 75%, then striking a note will produce a duration of 180 (75% of 240). But if you hold the note and press **Step>**, then the duration will be 420 (180 + 240), instead of 360 (75% of 480).

You can strike more than one note at a time to enter chords. You can also strike several notes one at a time (holding on to the previous notes) to create a chord. As long as you do not release all of the notes, the Locate parameter will stay at the same point and the notes will be entered as a chord. If you want to create a chord where certain notes have longer durations than others, strike the notes of the chord, then release the notes you wish to be shorter and press the **Step>** button. The window on the right will change to show only the notes you are still holding with their longer durations.

If you make a mistake while you are entering notes, you have two ways to fix it. With Dub Mode set to Merge, if you are still holding the note, you can press the **<Back** soft button. If you set the Dub Mode to Erase, you can just press the **<Back** button to erase the previous event and then strike the correct note. Or you can set the Locate parameter to the proper spot and strike the correct note.

### DubMode

DubMode determines if the note events entered in the EditSong : STEP page will merge with, or erase all other existing MIDI events located on the selected track, and in the Bars and Beats being Step edited.

You would set DubMode to Erase to replace all existing track data just in the Bars and Beats you edit. Data before and after the edited locations will be preserved. Set to Merge to add note events to existing track data.

The DubMode value will reflect the last value given to the Mode parameter on the SongMode : MAIN page, and if the DubMode in the Step editor is changed, the MAIN page Mode will be changed to the same value.

### Duration

The actual duration of a note event entered in Step time is determined by the values given for the Duration, Modifier, and GateTime parameter.

The Duration parameter sets the base note duration where 1/1 is a whole note, 1/2 is a half note, 1/32 is a thirtysecond note, and so on. The setting for this parameter, along with its Modifier, determine the size of the jump from the current Song location to the next Step entry location. The double button press of the Increment/Decrement buttons is one method of selecting a Step's duration.

Use the numeric keypad to quickly select a note duration and modifier whenever the cursor is positioned over the Duration field. These are the keypad duration settings:

- 1 whole note
- 2 half note
- 3 thirtysecond note
- 4 quarter note
- 5 sixtyfourth note
- 6 sixteenth note
- 7 applies a dotted modifier to current duration
- 8 eighth note
- 9 applies a triplet modifier to current duration
- 0 removes any modifier

### Modifier

The Modifier parameter allows you to change the Duration value with three standard modifiers. Set to None if you do not chose to use a modifier.

Use the triplet modifier for a resulting duration 2/3 of the value set in the Duration parameter. (e.g., an eighth note = 240 ticks, an eighth note triplet = 160 ticks).

A Dotted modifier adds half of the current Duration value to that duration, and a DotDot modifier will add 3/4 of the current Duration value to itself. (e.g., an eighth note = 240 ticks, a dotted eighth note = 360 ticks, and a double dotted eighth note = 420 ticks).

### GateTime

GateTime determines the percentage of the note duration, (set with the Duration and Modifier), that is actually played. The value given for the GateTime parameter does not affect the size of the jump from one Step to another, but only the length of time that the note(s) entered for each Step are sustained. You can use a value greater than 100 for a legato feel, but keep in mind that when two identical notes overlap you may get unexpected results. This is because the Note Off command from the first (overlapping) note will also cut off the second note (when it is identical).

Double-press the - and + buttons to increment this value in 20% intervals.

### Velocity

Each note event entered in the Step edit mode is assigned an attack velocity either by setting the Velocity parameter to a value between 1 and 127, or by setting the value to Played, where the velocity at which you actually enter notes is used for each Step.

### Locate

The Bar, Beat and Tick displayed as the Locate value is the location where the next Step entry will be placed in the Song, and once the Step is entered, Locate will advance an amount of time determined by the Duration and Modifier settings. The Bar and Beat, but not the Tick, will update during playback. It can be set to any Bar and Beat, including negative values, so that playback begins at, and Stop resets the Song to the Locate Bar and Beat. If the Bar and Beat value is set beyond the current End point, and no notes are entered before you press Play, then that location will be ignored when you begin playback and the Song will start from Bar 1 : Beat 1. If set to a negative Bar and Beat, no Step entry is possible.

## Soft Buttons on the Edit Song: STEP Page

**Play** starts the playback of the Song from the Bar and Beat set in the Locate parameter. When the Song is playing, this soft button becomes **Pause**.

**Stop** halts the playback of the Song and return to the Bar and Beat set as the Locate value.

**<Back / Step>** move the current location of the Song backward and forward in time, as reflected in the Locate Bar, Beat, and Tick. How far the location is changed from its current setting is determined by the Duration and Modifier settings.

**Done** returns to the EditSong : COMMON page when you are finished Step editing.

## Edit Song: ARRANGE Page

This page allows you to create a song by arranging other songs together in the order you specify. The other songs become sections (called Steps) of the current song, which can be repeated, transposed, etc. You can even trigger Steps by striking specific keys on the controller.

You can also create an arrangement of other songs to play back simultaneously with the current song. This gives you up to 32 tracks: 16 tracks in the current song plus 16 in the song or songs called by the arrangement (which are specified by the ARRANGE page's Song parameter). The arrangement you create can be as simple as including another song to add more tracks to your composition, or as complex as you can imagine.

More often than not, you would want to start using the Arrange feature from a "New Song" that doesn't have any track data recorded yet. Press the **ARRANG** soft button on the **COMMON** page to enter the EditSong : ARRANG page, gaining access to a group of arrangement specific parameters. All Songs have these parameters for creating complex arrangements of existing Song Objects, using ranges of the keyboard and realtime performance controls for triggering steps, but not every Song will utilize them.

When a Song containing an Arrangement is selected as the "CurSong" from the MAIN page, three dashes (---) appear above the Track parameter name to indicate that the current Song has been saved with programmed arrangement data.

When an Arrangement is PLAYING, the three dashes on the MAIN page are replaced with the ID# of the Song currently being played (or triggered) in the Arrangement, and the number of the current Step.

The square Track Mode Indicators will flash when MIDI data is present on the tracks during play back of an arrangement. The actual track data playing back is supplied by the Songs entered into the steps of an arrangement.

```

EditSong: ARRANGE          <>Step: 1/1
Step : 1
Song : 0 None
Mutes: ---  ---  ---  ---
XPose: 0st Times: 1x  Mode : Next

LoKey: Off HiKey: Off Latch: Off VelTrk: On
Add  Delete SetRng Play  Stop  Done
    
```

PARAMETER	RANGE OF VALUES	DEFAULT
STEP	1 – 99	1
SONG	ANY SONG IN MEMORY	0 NONE
MUTES	- , M	- (NOT MUTED)
XPOSE	-128ST – 127ST	0ST
TIMES	1X – 120X, INFINITE	1X
MODE	NEXT, STOP	NEXT
LOKEY	C-1 – G9, OFF	OFF
HIKEY	C-1 – G9, OFF	OFF
LATCH	ON, OFF	OFF
VELTRK	ON, OFF	ON

The first number displayed in the **Step** field on the top line is the current step. The second number is the total amount of steps in the arrangement. Use the **CHAN/BANK** buttons to select the current step for editing.

## Step

The number of steps in an arrangement defaults to 1. All other steps are then added when needed. The step of the Arrangement selected for editing is displayed as the Step. Use the CHAN/BANK arrows to select the different steps in your Arrangement. StartStep on the EditSong: COMMON page, as well as the key ranges of the steps, control what steps are played.

The behavior and control of each step is defined with the parameters described below.

## Song

The Song is the ID# and name of a Song Object in memory that will play back in proper numerical order with the other steps in an arrangement or will respond to any triggering from a defined keyboard range in that step. This song's (up to) 16 tracks will play in addition to any tracks in the current song.

## Mutes

The possible sixteen tracks of the current step's Song can be individually muted. When the selected step's display has tracks with an (M) in this field, those tracks will be muted during play back of the Song in the Arrangement.

## Xpose

Each Step in an Arrangement can impose a transposition on the Song selected in its Song parameter during playback. Xpose determines how many semitones a Step's Song is transposed above or below the Song's original key during playback of that step in an Arrangement.

All tracks of a Song being transposed as a Step will be transposed, except for those tracks defined as a Drum Track in the EditSong : COMMON page of the Song called up by the Arrangement, NOT the Arrangement Song itself.

## Times

The number of times that a Song is looped before an Arrangement moves on to its next Step can be determined with Times. You can set this value to repeat the Song up to 120 times before the next Step is played. An infinite loop can be set for any Step, but the Arrangement will play that Step until **Stop** is pressed.

## Mode

The Mode setting found in the Arrangement editor determines whether an Arrangement plays the next Step or if the sequencer will stop after playing the current Step.

## Triggering Steps from a Key

There are four parameters along the bottom of the Arrangement edit page that are settings for real time control of the Steps in an Arrangement. Triggering and transposing Steps from a MIDI keyboard, transmitting on the TriggerChan (see page 12-26), is possible when the TriggerCtl parameter, found on the COMMON page, is set to ON, or when the selected controller input activates arrangement control.

## LowKey

The first parameter, LowKey, is the setting for the lowest note in a keyboard range, that when played by the TriggerChan, will trigger the start of the current Step. Triggering a Step from the LowKey will play back its Song in its original key transposed the number of semitones set in the Xpose setting. As you play up the keyboard chromatically, the Step will transpose its Song

in semitone increments. All tracks of a Song being transposed by keyboard triggering will be transposed accordingly, except for tracks defined as a Drum Track on the COMMON page.

### **HiKey**

The highest note of the keyboard range designed to trigger the current Step is set in the HiKey parameter. HiKey defines the largest transposition of a Step from the key of the Song triggered by the LowKey. The HiKey must be a higher note value than the LowKey in order to trigger any Step from the keyboard.

### **Latch**

If Latch is set to Off, the playback of the Step, triggering it from the defined keyboard range, will be gated for as long as the triggering Key is depressed. Set Latch to On if you want the use the keyboard to start the Step, and have it continue playing after you stop holding the key down. The Step will play for its entire length unless you press **Stop** or until it is retriggered.

### **VelTrk**

Turn the VelTrk parameter On to make your real time Arrangements more dynamically controllable. The original attack velocity of every Note event in the triggered Step's Song will be scaled to values determined by the attack velocity of the note you play when triggering.

## **Soft Buttons on the Edit Song: ARRANGE Page**

**Add** adds another Step to the Arrangement. The new Step will be inserted as the Step number directly after the current Step. You would add a Step to place a new section into a Song Arrangement or to define another keyboard range for triggering.

**Delete** removes the current Step.

**SetRng** lets you quickly define the keyboard range for triggering the current Step from the MIDI keyboard. This dialog will appear:

Strike low key...

**Cancel**

"Strike low key..."

Play the lowest note of the range you wish to define, then:

"Strike high key..."

Play the highest note of the range you wish to define.

At any time you may abort the range setting procedure by pressing the **Cancel** soft button. The display will verify that no range has been set by replying to this button push with Canceled., and then returns to the ARRANGE page.

**Play** lets you audition your Arrangement from the ARRANGE page, starting playback from the currently selected Step.

**Stop** halts the playback of an Arrangement from the ARRANGE page.

**Done** exits the ARRANGE page and returns to the COMMON page.

## Selecting a Song for Playback

Select the Song parameter with the cursor buttons, then use any data entry method to scroll through the list of songs. Press the **Play** soft button, and the song will begin playing. Press the **Stop** soft button, and the song will stop and “rewind” to the beginning. If you press **Play** while the song is playing, the song will stop and the play pointer will revert to the Locate parameter, and will show your current location in the song. At this point you have two options. If you press the **Play** button again, the song will continue from its current location. If you press the **Stop** soft button, the song will return to 0:0.

The K2vx automatically selects programs for playback based on the MIDI channel(s) and the programs assigned to them at the time the song was recorded. When you start playback, the K2vx sends program changes, on all relevant channels, to its sound engine and to the MIDI Out port if the PChng parameter on the MIDI mode XMIT page is turned on.

If you want to use a different program for playback than the one originally recorded, you’ll need to defeat this automatic program selection. To do so, set the ProgLock parameter to a value of On. The ProgLock parameter is found on the MIDI mode CHANLS page; you can set it independently for each of the 16 MIDI channels. When a channel’s ProgLock parameter is set to a value of On, the K2vx’s sound engine will ignore all program changes it receives on that channel, either via MIDI or from the K2vx’s front panel.

When you’ve set the ProgLock for each of the channels used in the song, you can go to Program mode, select the MIDI channels used by the song, and assign the programs you want to use. When you return to Song mode and play the song, the automatic program changes are defeated, and the song plays the programs you assigned.

## Effect Selection During Recording and Playback

When you’re recording or playing back a song, the preset effect applied to the song is determined by the program assigned to the FX Channel. Only one effect can be applied at a time, even for multi-part songs using more than one program. Depending on the combination of values for the FX Mode and FX Channel parameters in Effects mode, the FX channel may automatically track the current MIDI channel. In this case, if you change the current MIDI channel during playback, or during a MIDI recording, the current FX Channel, and consequently the current effect, will change also, which might not suit your needs. Consequently, there’s a way to force the FX channel to remain constant during playback or MIDI recording, even if you move to another mode during the playback or MIDI recording and change the current MIDI channel.

This is done by setting the FX Mode parameter to a value of Auto, and the FX Channel parameter to a value of Current—which is the most generally useful combination of settings for these parameters. In this case, when you’re in Song mode—or when you move to another mode during playback or MIDI recording—the FX Channel parameter is temporarily forced to a value of 1. Therefore the effect is determined by the program assigned to MIDI channel 1, and will not change, even if you change the current MIDI channel during playback or MIDI recording.

Of course, changing MIDI channels during playback or MIDI recording will not change the FX channel if the FX Channel parameter is set to a value from 1 to 16, or if the FX Mode parameter is set to a value of Master.

## Synchronizing Songs

The K2vx has an internal MIDI clock, which is always running at a speed set by the Tempo parameter. When you’re in Song mode and the Clock parameter is set to a value of Internal, songs will sync to the K2vx’s internal clock. At this setting, the clock signal is sent to the K2vx’s

MIDI Out port. This is standard MIDI Sync, and any device that accepts MIDI Sync will play in sync with the K2vx.

If you set the Clock parameter to External, the K2vx expects to receive MIDI Sync at its MIDI In port. When you play back a song, pressing the **Play** soft button will cue the K2vx to wait for the sync signal. It will begin playing when it receives the signal, or when you press the **Record** button again. If no sync signal is received, the K2vx uses its internal clock at the tempo of the current song.

If you're using Song mode to capture a sequence that you've recorded on an external sequencer, you'll want to consider the Clock parameter's setting before you record via MIDI. If you have the Clock parameter set to External, the K2vx will follow the clock of your external sequencer. As a result, the notes in the song you create will fall regularly on the beats (unless your externally recorded sequence uses a time signature other than 4/4). This will make it easier for you to find the notes in the Song Editor. If you set the Clock parameter to Internal, the notes in the song will not necessarily align with the beats of the measures in the song, but the song will play back exactly as you recorded it on the external sequencer—including tempo changes you may have incorporated into the externally recorded sequence. If the Clock parameter is set to External, tempo changes will not carry over to the K2vx, and will not be heard when you play back the song.

Finally, keep in mind that when the Clock parameter is set to External, programs that use one or more of the Clock control sources (see *Reference Guide*) will sync to the external MIDI signals. If no external sync signal is received, the Clock control sources are disabled.

## Memory Limits

While there's no actual time limit to the length of the songs you record, their size is limited to 64K (or to the maximum amount of available free RAM space you have, if it's less than 64K). However, you can create longer songs by recording each section as a separate song, then putting it together with the Arrange feature. If you run out of RAM space while recording a song, the recorder stops and prompts you to save the song. It's a good idea to check your free RAM space before you begin recording a song, and to check the "Used" readout as you record. If you've used all the available RAM for recording, you may find that when you go to the Song Editor to delete a song or edit its tempo, the K2vx tells you that there's not enough memory to edit. In this case you won't be able to edit any object greater than 4K in size. Objects smaller than 4K can still be edited, because the K2vx always reserves a minimum of 4K of RAM.

If you want to delete a song and the K2vx won't let you enter the Song Editor, select one of the Metronome songs. Since the metronome songs are smaller than 4K, you'll be able to enter the Song Editor. Press the **Delete** soft button, then use the Alpha Wheel to select the program you want to delete. Press the **Delete** button again, and the song will be erased, freeing up enough RAM to edit other songs.

## Loading Songs From Disk

If you have a Type 0 (single-track, single- or multi-channel) MIDI sequence file stored on an MS-DOS disk (720K or 1.4 M), you can load it into one of the RAM banks, and the K2vx will be able to play it from Song mode.

## Recording Multi-timbral Sequences via MIDI

You can record sequences from an external MIDI device using Song mode. Program numbers and MIDI channel assignments of multi-timbral sequences are recorded with the notes. To record via MIDI, connect the MIDI Out port of your sequencer to the K2vx's MIDI In port.



Select Song mode, and set the Clock parameter to External. This will sync the K2vx with the MIDI clock of the external sequencer.

You will probably want to set the Local Keyboard Channel parameter to None when recording from an external sequencer, since the rechannelizing effect of that parameter could have unintended results.

To record all your tracks in one pass, set the RecTrk parameter to Mult and make sure that for each channel of information on your source sequence, you have a track enabled to record and a Channel a unique channel assigned to that track. (The default setting of all tracks enabled to record on channels 1-16 will always work.)

You can also record individual tracks from your source sequence by setting the RecTrk parameter to a specific track. The K2vx will record only information coming in on the channel that the RecTrk parameter is set to.

Press the **Record** button, and the K2vx will wait for the first clock start from the sequencer. Make sure that your sequencer is set to send MIDI clock signals, and start the sequence. The K2vx will begin recording when it receives the first MIDI clock start from the sequencer. When the sequencer has finished its playback, press the **Stop** soft button, and the K2vx will stop recording and ask if you want to save the song.

## **Song Mode**

---

Recording Multi-timbral Sequences via MIDI

## Chapter 13

# Disk Mode

Disk mode lets you load and save files of objects between the K2vx and the outside world, through the K2vx's floppy drive or SCSI ports. The floppy drive accepts MS-DOS format double-sided double-density (DSDD—720K) and high-density (HD—1.4 M) floppy disks. It will also accept single-sided double-density disks (SSDD—360K) if they're formatted as double-sided. To save data from the K2vx to a floppy disk, the disk must be formatted (initialized) by the K2vx or by an IBM®-compatible computer running MS-DOS® software.

Most SCSI (Small Computer System Interface) devices will operate with the K2vx. The K2vx's SCSI ports require 25-pin SCSI connectors. The most common use for the SCSI ports is to connect a hard disk or chain of hard disks for storing samples and other objects. You can also connect a CD-ROM drive for reading files to the K2vx. The K2vx will treat a CD-ROM drive like any other SCSI device (except that you can't save files to it).

Disk Mode in the K2vx allows flexibility to organize disk files and their contents. Many powerful operations are included that can save a lot of time by allowing you to easily specify exactly what you want to load or save. Examples of this range from organizing related files into directories, to loading macro lists of files or selected objects from multiple SCSI drives, to setting up programs to automatically link with samples off of a CD-ROM.

Lastly, the K2vx offers MS-DOS file system compatibility, sample transfer using the standard audio file formats "AIFF" and "WAVE", and saving of song files as MIDI Type 0 format.

## Disk Mode Page

To enter Disk mode, press the Disk button, and the Disk mode page will appear:

```

DiskMode      Samples:65535K  Memory:747K
Path = \DRUMS\
(Macro on)
CurrentDisk:SCSI 4      Startup:Off
                          Library:Off

Direct Access, 121MB
TAXMOR XL3-1001      1.07
<more> Load Save Macro Delete more>

```

As usual, the current mode is displayed on the top line. At the middle of this line, the amount of available sample RAM is shown (if you have added optional sample RAM; if you haven't, this part of the top line will be blank). To the right of the top line you see the amount of memory available for storing all other RAM objects. Straight from the box, this number should be around 240K. If you have added the optional P/RAM extension kit, the number will be about 1260K. The P/RAM kit enables you to store hundreds of additional programs, songs, and any other objects. If you do a lot of programming or sequencing, the P/RAM option is a good idea.

In the center of the page is a line indicating the currently selected device. Select different devices using any data entry method. You can select the floppy drive, or SCSI IDs 0–7. (If you connect an Apple Macintosh® personal computer, don't select SCSI ID 7, since that's the SCSI ID of the Mac, which can't be changed. All SCSI devices connected in a network must have different SCSI IDs in order for the network to function). When you want to communicate with any of the SCSI storage devices in your network, set the Current disk parameter to the value

that matches the SCSI ID of the device you want to address. That is, the K2vx will interact with the SCSI storage device whose SCSI ID matches the value of the Current Disk parameter. If you choose a value of Floppy, the K2vx will communicate with its floppy disk drive.

The manual for your SCSI disk should tell you its SCSI ID. Most newer SCSI disks show their SCSI IDs on their rear panels, and many have adjustable SCSI IDs.

The currently selected device will be read from or written to when you load, save, rename, or delete files. Use the soft buttons to start any of these operations. Refer to "Soft Buttons on the Disk Mode Page" on page 13-5 for complete information.

### **Formatting a Floppy Disk**

Floppy disks must be formatted (initialized) to the Kurzweil format before they can be used with the K2vx. The K2vx uses the MS-DOS format, so floppies formatted on DOS computers should work with the K2vx. To format a disk on the K2vx, insert a floppy disk in the K2vx's floppy drive. Make sure the disk is unlocked (the sliding switch on the back of the disk is set so you can't see through the hole on that side of the disk).

Press the DISK button to enter Disk mode. Make sure the Current disk parameter says "Floppy," so you don't accidentally format any SCSI devices you might have connected! Press the soft button labeled Format. The K2vx will ask you if you want to format, and a pair of Yes/No soft buttons will appear. Press the **Yes** soft button, and three soft buttons will appear, letting you choose between formatting the floppy as a 720K (double-density) or 1.4M (high-density) disk. Press the appropriate soft button, or press the Cancel soft button.

If the floppy you inserted is double-density and you press the 1.4M soft button, the format procedure will fail. This is also true if the disk is high-density and you press the 720K soft button. If the floppy is single-sided (SSDD), you can press the 720K soft button, and the floppy will be formatted as a double-sided disk. (The only difference between single- and double-sided floppies is that both sides of a double-sided floppy have been tested by the manufacturer.)

Once you select between the 720K or 1.4M format, the K2vx will remind you that formatting will erase the floppy, and will give you two more chances to cancel the formatting procedure—we want to make sure you don't accidentally erase any disks. Press the **Yes** soft button to continue formatting. When formatting begins, the display will tell you that the disk is being formatted. You'll hear the disk drive turning, and the disk drive LED will light.

Formatting a 1.4M floppy disk takes just under three minutes, including the automatic verification. When formatting is finished, the display returns to the Disk mode page.

### **Connecting a SCSI Device**

It's easy to connect SCSI devices to the K2vx's SCSI ports. Using a SCSI device will give you off-line storage, and can speed up your loading and saving operations considerably.

You'll need a SCSI cable with a 25-pin SCSI connector on the end to be connected to the K2vx. If your SCSI device does not have a 25-pin connector at one end, you can find SCSI cables like these at any personal computer store. Connect the 25-pin end of the cable to either of the K2vx's SCSI ports, and the other end to your SCSI device. Two SCSI ports are provided to enable you to chain SCSI devices together. Before you start connecting cables, however, please read the next section carefully. We've also included important information about SCSI in Chapter 8 of the *Reference Guide*.

## SCSI Termination

Simply put, SCSI termination prevents the electrical signals used by SCSI devices from being reflected from unconnected SCSI ports, and possibly disrupting the data stream. The rule for SCSI termination is that the two SCSI devices on the ends of a chain of SCSI devices must be terminated, and all devices in between, however many, must be unterminated. Newer SCSI devices usually make it easy to enable or disable their termination settings. Older SCSI devices may require an external terminator to be installed. These are available at all personal computer stores. Make sure you get the right size for your device (25-pin or 50-pin).

It's impossible to describe all the possible configurations of SCSI devices, so we'll provide you with a few general guidelines that will cover the requirements for most SCSI systems. If you're chaining large numbers of SCSI devices together, you may have to do a little juggling, but chances are you'll already have some experience with SCSI termination.

First of all, it's *very* important that you terminate your SCSI system properly. Improper termination can result in lost data, can interfere with the operation of your SCSI devices, and over the long term, can damage them.

If your SCSI system includes an Apple Macintosh® personal computer, you'll need to be sure that it is internally terminated. This is because the Mac has only one SCSI port, and therefore is necessarily going to be at one end of your SCSI chain. If your Mac has an internal hard disk, you can go ahead and hook it up, since it's sure to be internally terminated. Some Macs without internal hard disks are *not* terminated, so this may cause a problem. If your Mac has no internal hard disk, and you're not sure whether it's internally terminated, you should call your local Apple dealer for confirmation.

If your SCSI system includes only the K2vx and an internally terminated Mac, you're all set. Just connect the Mac's SCSI port to either of the K2vx's SCSI ports, and you're ready to roll. The K2vx is internally terminated, so in this case, you've satisfied the requirement of terminating both ends of the chain.

If you have an internally terminated Mac, a K2vx and an external hard disk with *two* SCSI ports, setting up is also painless. Connect the Mac's SCSI port to one of the hard disk's SCSI ports, and the K2vx's SCSI port to the hard disk's other SCSI port. Make sure the hard disk is not terminated, since it's in the middle of the chain. In this configuration (with a terminated Mac at one end and the K2vx at the other), you can chain up to six hard disks between them. Make sure they're all unterminated, and don't forget to set each disk's SCSI ID to a different value.

If you have a Mac, a K2vx and an external hard disk with only *one* SCSI port, you'll need to have your K2vx serviced. The only way to connect a system of this sort is to have the K2vx in the middle (with one SCSI port connected to the Mac, and the other to the hard disk). In this case, you'll need to have your Kurzweil/Young Chang dealer or service center remove the K2vx's internal termination. This is very easy for a qualified technician to do, and shouldn't take more than a half hour or so.

If you install an internal hard disk in your K2vx, the technician who does the installation should remove the K2vx's termination as part of the installation. This will not necessarily enable you to connect the K2vx in the middle of a SCSI chain, however, since the internal hard disk has its own termination. With an internal hard disk, the K2vx should be installed at the end of a SCSI chain. The internal hard disk's terminator will provide the termination for itself and the K2vx. You can remove the termination from the internal hard disk as well, if you want to connect your K2vx in the middle of a SCSI chain. In this case, you must either connect it in the middle of a chain, or install an external terminator on the unused SCSI port if you connect it at the end of a chain.

In this case, the K2vx can be connected in the middle of a SCSI chain without causing problems. If you connect it at the end of a SCSI chain, however, you should install an external terminator on the unused SCSI port.

If you're planning to buy a SCSI hard disk to use with your K2vx, it's a good idea to buy one with two SCSI ports. Most new hard disks have two ports, and can be terminated or unterminated relatively easily. This gives you added flexibility, since you can install it at the end of a chain, leaving its termination in place, or in the middle of a chain, using both its SCSI ports, and removing its termination.

When your SCSI device is connected, you can select it with the Current disk parameter on the Disk mode page. Use any data entry method to select the SCSI ID that matches the SCSI ID of your SCSI device. If you're using the numeric keypad to select the device, enter "9" to select the floppy. Newer SCSI devices usually have an external switch for setting their IDs. Older units may not have these; check your device's owner's manual for its SCSI ID.

## Using your K2vx in a SCSI System

### SCSI IDs

All devices in a chain of SCSI devices must have different SCSI IDs, including the K2vx. The K2vx's SCSI ID is set at 6 by default, and can be changed on the RECV page in MIDI mode. If your SCSI system includes a Mac, be sure not to use SCSI ID 7 for any of your other devices, since the Mac's SCSI ID is 7, and can't be changed. Once you've made sure that all connected devices are set to different SCSI IDs, you should be able to select the devices, format them, and start loading and saving files.

### Formatting a SCSI Device

The procedure for formatting hard disks is essentially the same as with floppy disks, once the SCSI device is selected with the CurrentDisk parameter. The K2vx will recognize the disk as a SCSI disk, and will warn you that formatting will erase the contents of the disk. Compared with personal computers, the K2vx's formatting time for SCSI disks is surprisingly short.

## Directories

A directory is a file on the disk that lets you group other files together as you might separate documents using folders in a file cabinet. You can create directories on K2vx Format SCSI and floppy disks. You can even create directories within directories, which are also called *subdirectories*.

Directories are very useful for organizing your sample, song, and program files. The K2vx provides many operations for setting up and managing the directories on your disks and the files within them.

## Path

The Path field shows the current directory on the current disk if it is a K2vx format disk. This field is displayed upon returning to the Disk Mode page after you have pressed one of the disk function soft buttons and viewed the file contents of a specific disk. The current directory is always set to the top-level or root directory when the K2vx is powered on, or when the CurrentDisk parameter is changed. After this, the current directory is remembered by the K2vx as the most recent directory accessed by the disk functions.

The root directory is displayed as

```
Path = \
```

If you press the Load button and load in a file from a subdirectory called SOUNDS, the Path field will appear as

```
Path = \SOUNDS\
```

The slash character is also used as a directory separator, as in the following Path:

```
Path = \NEWTUNE\SAMPLES\DOGS\
```

This represents the subdirectory DOGS, that is located in the SAMPLES subdirectory of the NEWTUNE directory in the root directory. If the path is too long to fit on the top line of the screen, its display may be abbreviated. The maximum length of a path in the K2vx is 64 characters (including '\' characters).

## Startup

The Startup parameter determines what disk drive will be used for loading the power-up macro file BOOT.MAC (See "Creating a Startup File," page 13-63). If this is set to None, then the K2vx will power-up in a normal fashion. If this is set to a SCSI device or Floppy, when the K2vx is next powered on it will look for the BOOT.MAC file in the root directory of the specified disk, and load in each of the entries in the macro specified within.

This feature provides a very flexible way to automatically configure your K2vx's memory contents whenever you turn the power on.

## Library

This feature works in conjunction with the Macro feature to provide a way to distribute Macro files that load data from removable media without having to know in advance the SCSI ID of the removable drive. A Macro file stores its references to drives by drive ID (SCSI ID or Floppy), or by either a "Library" or "Unspecified" designation (see the section on Macro files, page 13-38.) Typically, you would set the Library parameter to be the same as the SCSI ID of your CD-ROM drive, if you were loading Macro files from a Floppy disk or another SCSI drive that referenced CD-ROM files containing samples or keymaps.

Use of the Library feature is described in more detail later on in this guide.

## Disk Drive Information

Below the media type and disk capacity is displayed specific information about the current disk's manufacturer, model number and internal mechanism. The K2vx requests this information from a SCSI drive when the drive is selected by setting the Current Disk parameter. This information may be needed when determining if a given drive is compatible for SCSI operation with the K2vx.

## Macro On Indicator

When the "**(Macro on)**" indication is displayed, it means that any load operation performed will be recorded in the K2vx's macro table. The Macro feature is described starting on page 13-38 in this guide.

## Disk Mode Soft Buttons

Here is a brief description of each of disk mode's soft button:

**Load** - Loading files or loading files from the current disk into K2vx memory.

**Save** - Saving banks of objects, selected objects, or a macro as a K2vx file on the current disk.

**Macro** - Controls macro recording. Allows modifying entries from the current macro list.

**Delete** - Delete files from the current disk if it is a K2vx disk. See "Deleting files", below.

**Rename**- Change the file name of a file on a K2vx disk. See "Renaming Files", below.

**Move** - Change the location of a file from within one directory to another (on the same disk).

**Util** - Check the free space, find files, and view directory organization and sizes on the current disk.

**NewDir**- Create a new directory on K2vx disks.

**Backup**- Hierarchical file backup between drives.

**Copy** - Single file copy between drives.

**Sleep** - Send SCSI sleep command to the current disk. See "The Sleep Soft Button", below.

**Format** - Format the current disk as a K2vx disk.

### The Sleep Soft Button

Many SCSI devices will "sleep" when they've been idle for a few minutes. In other words, the disk will stop spinning, in order to save power and reduce wear. The K2vx lets you tell your SCSI devices to sleep. Just press the Sleep soft button, and if your devices have this feature, they will sleep. This is particularly useful in a quiet studio situation.

Any Disk mode operation will "wake" the device again. The K2vx will ask you to wait while the device's disk starts spinning. As soon as the disk is spinning at full speed, the K2vx will execute the operation you selected. Some SCSI devices automatically sleep when they power up. (A device of this type usually provides a way to override this feature; check its manual). Any Disk mode operation will wake a disk in this case, as well.

## File List Dialog

The file list dialog is displayed anytime you select a disk function to operate on one or more files on a drive (such as Load or Rename.) Here is a typical file list dialog, for the Load function:

```

Dir:\                               Sel:0/3                               Index: 1

File to load: BASSOON .KRZ 3456K
              MHY25      (dir)
              PERC      .KRZ 101K

Total: 3557K
Select Root Parent Open OK Cancel

```

When you enter this dialog, the K2vx selects the current directory to display, and displays its files in an alphabetized scrolling list. If the current directory cannot be located (for instance, if a removable media disk has been changed), the default is to display the disk's root directory. The root directory will also be selected if the disk drive was just chosen by the CurrentDisk parameter on the Disk Mode page (remember that the current directory is always set to the top level when the CurrentDisk parameter is changed, or if the K2vx has just been powered on.)

The display for all drives (including floppy) shows the 3-character extension of all files in the directory (except directories themselves). Extensions are created when the file is saved by the K2vx. You cannot modify the extensions on the K2vx. This is because the K2vx uses the extensions to tell it what kind of data is stored within the file.

Directories created by the K2vx have up to 8-character names, with no extension. A directory can have an extension if it is created on an external computer (more on this later on).



Here is a list of extensions used by or accepted by the K2vx:

- .AIF - Audio Interchange File Format (AIFF)
- .KOS - Kurzweil K2vx operating system file
- .KRZ - Kurzweil K2vx format file
- .K25 - Kurzweil K2vx format file containing objects and/or sample data
- .MAC - Kurzweil K2vx disk macro file
- .MID - MIDI type 0 sequence file
- .WAV - Microsoft RIFF WAVE format

When loading files, the K2vx will try to find out the type of file if the extension is not the same as is suggested above (with one exception, .MAC files.) The K2vx can create files with any of the above extensions.

The top line of the file list contains several items of information pertaining to the currently displayed directory contents. A typical information line looks like this:

```
Dir: ..\HHS\ Sel: 0/54 Index: 24
```

In the center of this line is an indicator of the number of files in the currently displayed directory. This number is grouped together with the number of selected files:

```
Sel: 0/54
```

The first number means "No files selected" and the number after the slash means "Fifty-four files are in the directory". File selection is possible in several of the disk functions, more on this below. The total number of files also includes any subdirectories of the current directory, but not the files within the subdirectories.

On the left end of the top line of the file list page is the current directory, sometimes in an abbreviated form. If you are in the root directory, the display will read:

```
Dir: \ Sel: 0/54 Index: 24
```

If you are in the directory \MONDAY, the display will read:

```
Dir: \MONDAY\ Sel: 0/54 Index: 24
```

If you are in a directory that is more than one level down from the root directory, such as \FX2\GLASS\BREAKING, the display will read:

```
Dir: ..\BREAKING\ Sel: 0/54 Index: 24
```

The "..\" indicator tells you that you are more than one level down from the root directory.

### The File Index

On the right end of the top line is the Index field. This tells the position of the highlighted file relative from the beginning of the file list. The first entry in a file list is index 1.

```

Dir:\                               Sel:2/23   Index:    3
      HXM      .KRZ*    122K
      CHIME    .KRZ     42K
File to load: DOORS .KRZ* 3456K
      JUNE27   (dir)
      LONGSMPS (dir)
Total: 21034K
      FLUTE   .KRZ     .5K
Select Root Parent Open OK Cancel

```

Typing a number on the K2vx numeric keypad will automatically scroll the display to the corresponding entry in the file list. Typing an out of range value such as 999 is a quick shortcut to get to the end of the file list.

In addition to remembering the current directory on the most recently used drive, the K2vx now also remembers the index within the file list for the current directory. For instance, if you were to hit cancel on the above screen, and then either choose a different disk function (such as Rename), or even if you exited Disk Mode, went to Program Mode, and then later returned to Disk Mode to load a file off of the same drive, the file index would still show up on 3 (DOORS.KRZ). This index is remembered until a new drive is selected by scrolling the Current Disk parameter on the Disk Mode page.

There are exceptions to this however. For example, when a file is written to the drive using the Save function, the index will subsequently be set to the file that was just saved (see Save function). The index can also be explicitly set using the List and Find utilities (see below).

If there are no files in the current directory, then the index is 0:

```

Dir:\                               Sel:0/0   Index:    0
File to load:
Total: 0K
Select Root Parent Open OK Cancel

```

You will also see that there is a blank cursor bar, but no file names are displayed.

The maximum number of files that can be accessed within a single directory is 360. If you have more files than this amount in a single directory, then you will not be able to view the entries past index 360.

While in this dialog, pressing the CHAN/BANK buttons (to the left of the LCD display) will scroll the file list in either the forward or backward direction by "pages" of 5 entries. It is often easier to scroll the list this way when looking to see if a particular file is present in a directory.

### Soft Buttons in the File List Dialog

Use the **Select** soft button for multiple file selection in the Load, Delete, and Move functions. In the display below, there are two files selected, as indicated by the asterisk ("\*") after the .KRZ file extension. If OK were pressed for the following screen,

```

Dir:\                               Sel:2/23       Index:      3
AXM                                 .KRZ         122K
CHIME                               .KRZ         42K
File to load:DOORS                  .KRZ*       3456K
JUNE27                              (dir)
LONGSMPS                            (dir)
Total:21034K                        FLUTE        KRZ*        5K
Select Root Parent Open OK Cancel

```

the files DOORS.KRZ and FLUTE.KRZ would be loaded.

The Select button will toggle the selection, meaning that if you press Select on a given file, the asterisk will go on if it is currently off, and vice-versa. Selecting can only be done for files, not for directories. You can select as many files as you wish using the Select button. There is also a way to select all files at once, or clear all file selections at once, using a double-press of the cursor buttons:

**Left/Right cursor double-press -> Select All Files**

**Up/Down cursor double-press -> Clear All Selections**

Pressing either the Left or Right cursor individually performs a separate function for finding directories, described below.

You can only select multiple files from within a single directory. Changing directories clears any selections.

Once you have selected one or more files, press **OK** to perform the disk function (in this example, Load) on all files marked with an asterisk, whether or not they are visible on the screen. If there are no files marked with an asterisk, the function will operate only on the highlighted entry, as in earlier versions of the K2vx system software.

The **Root** soft button returns the listing to the top-level or root directory. If the display is already at the root directory (as indicated by the Dir: \field on the top line of the display) the only effect of pressing Root will be reset the file index to 1 if there are files in the directory.

The **Parent** soft button is used to go up one level in the directory hierarchy. If the display is already at the root directory, this button will have no effect.

The **Open** soft button performs a different operation depending on the disk function and the type of file that is currently indexed in the display. In all disk functions, pressing **Open** on a subdirectory (indicated by a "(dir)" after the file name) will open that directory, read in and display the file list.

When a directory is first opened for viewing, the index will be set to one (the first file in the list). You might notice that the K2vx remembers the index the previous directory you were in before you pressed **Open**, so that if you return to that directory by pressing Parent, you are put back where you were in the list. This index is only remembered for one level down, and therefore is useful when stepping through a list of subdirectories from a single directory level.

In the Load function, pressing **Open** for a standard .KRZ file will start the Load Object feature. This allows selected individual objects from the file to be loaded in to the K2vx. If **Open** is pressed on a Macro file (.MAC extension), then individual file entries within a macro file can be selected for loading. Both of these features are explained in great detail below.

For all other functions, if **Open** is pressed when cursor to a .KRZ or a .MAC file, the object file or the macro file will be opened for viewing. For instance, pressing Open on a .KRZ file

while in the Delete function will display the objects within the file in a scrollable list, however no delete action will be possible on the individual objects.

Pressing the **OK** soft button will cause the K2vx to proceed with the selected function. After pressing OK, there may be further dialogs such as bank specification (for the Load function), confirmation (for Delete), or name entry (for Rename). One exception to this is in the Load function, if the file list is scrolled to a directory (dir) entry, then the OK button will enter the directory, the same as if the Open button is pressed on a directory entry.

The **Cancel** soft button exits the file list dialog, completing the disk function with or without any operation taking place. The K2vx returns to the Disk Mode page. Pressing the Exit button will do the same thing as Cancel.

## Total

The total size of all the files in the directory is indicated at the bottom left of the file display above the soft buttons. This total represents only the disk space used by the files in the directory being viewed. The K2vx includes a free space utility that can be used find out how much space is being used on the current drive all of the files on the disk. Also, there is a List utility that can be used to calculate the size of all files within a selected directory sub-tree. These functions are described under "Disk Utilities" on page 13-55.

## Quick Scrolling to Subdirectories

It is sometimes difficult to locate a subdirectory entry in the file list for the current directory, if there are many files in the current directory. To make this easier, individually pressing either the Left or Right cursor buttons will set the file index to the next or previous directory (respectively) in the current directory list. The index will wrap around the beginning or end of the list, so that repeated presses of either cursor button will cycle through all of the subdirectories. If you have many subdirectories, you can scroll through them all very quickly using this method.

For example, given the following file list display,

```
CYMBALS      (dir)
DOGS         .KRZ      122K
DOORS        .KRZ*    3456K
E4PROG       .KRZ      10K
LONGSMPS     (dir)
LUTE         .KRZ*     .5K
```

Pressing the right cursor takes you two entries further to the next directory,

```
DOORS        .KRZ*    3456K
E4PROG       .KRZ      10K
LONGSMPS     (dir)
LUTE         .KRZ*     .5K
MOON         .KRZ*    3456K
TRIANGLE     .KRZ      10K
```

or, pressing the left cursor takes you two entries back to the previous directory

```

ALTO                (dir)
ROOBAMS             .KRZ*   .5K
CYMBALS             (dir)
DOGS                .KRZ    122K
DOORS               .KRZ*   3456K
E4PROG              .KRZ    10K

```

## Creating Directories

As stated above, you can create directories for organizing your K2vx files, whether you are using SCSI or Floppy disks. No reformatting is necessary to add directories to disks that were formatted with a K2vx.

Directories appear in the normal file list with the indicator "**(dir)**" to the right of the directory name.

There are two ways to create new directories.

First, the **NewDir** button can be pressed directly from the Disk Mode page, or second, a new directory can be created and entered in one step during the Save operation, by pressing **Save** then **NewDir**.

### Creating a Directory with NewDir

When you press **NewDir**, you will first name the directory that you are about to make, for instance:

```
Directory name: THINGS_
```

```
Delete Insert >>End Choose OK Cancel
```

Pressing **>>End** will take the cursor to the last character in the name. The **Choose** button allows you to grab a file name from the current disk (this is discussed later, in the section titled "Saving Files".) Otherwise, the name will default to either "NEWFILE" after a power-up, or the name will be that of the most recent file saved or loaded. Once you choose a name to start with (or the default), you can edit the name using the left and right cursor buttons, the **Delete** and **Insert** soft buttons, and the **>>End** soft button.

After you have chosen the directory name and pressed **OK**, you have the choice of where (in what directory) to put the new directory you are creating.

```
Use current directory for THINGS?
(Path = \)
```

```
Change OK Cancel
```

Pressing OK will select the default path, which is the current directory. Pressing Change will allow you to view the disk, traversing its directories, until you find the one in which you want to create the new subdirectory, in this case "THINGS". (see Change Directory Dialog, below)

### Creating a Directory with Save -> NewDir

Save->NewDir is used as a convenience when saving files to a directory. This is accomplished by pressing **Save** from the Disk Mode page and then pressing **NewDir** at the "Save what:" dialog. After you have created the directory in this way, the current directory is automatically switched to while you are still in the save dialog. See the section on saving files for more information.

When you have confirmed your choice of what directory to create a new sub-directory in, the K2vx will create the named directory on your drive, and print a brief message such as:

```
Created directory /THINGS
```

If the new directory function was created by pressing the NewDir button on the DiskMode page, the K2vx returns to the Disk Mode page. Otherwise, if NewDir was pressed from the Save dialog, the K2vx returns to the "Save what:" dialog, having set the current directory to be the one just created.

## The Directory Selection Dialog

When making a new directory, as well as in many of the disk functions, you will be presented with the opportunity to change the current directory, or the default directory for a disk operation. A good example is the "Use current directory for THINGS? (Path = \)" screen above. If you press "Change" you will see a familiar file list dialog, with a few changes, through which you can select any directory on the disk. The display looks like this:

```

Dir:\                               Sel:0/23           Index: 7
                                MELLOTRN .KRZ    122K
                                N123AB   .KRZ     42K
Select directory: THINGS .KRZ 3456K
                                STRINGS   (dir)
                                T1         (dir)
Total:21034K  UNDUATE .KRZ .5K
Root Parent Open           Current Exit

```

When you enter this dialog, you will be in whatever directory was displayed as the default. From here you can go into other directories by using the soft buttons Root, Parent, and Open. Notice that there is no "Select" button. This is because the purpose of this dialog is to choose a single directory as opposed to selecting multiple files. However, the Root, Parent, and Open buttons function exactly as described above (for the file list dialog). The Sel: parameter (on the top line) is useful as an indication of the number of entries in the directory you are looking at.

If you are scrolled to a directory entry, there is one additional soft button displayed, "SetDir". Notice the "Current" button moves over one button to the left:

```

Dir:\                               Sel:0/23       Index: 8
                                N123AB      .KRZ      42K
                                OCEANS      .KRZ     3456K
Select directory:STRINGS          (dir)
                                T1           (dir)
                                UNDULATE .KRZ      .5K
                                ZORK      .KRZ      .5K
Total:21034K
Root Parent Open Current SetDir Exit

```

There are two ways that you can select a directory in this dialog.

**Pressing "Current"** - This selects the directory you are currently in (whose file list you are viewing), as specified in the Dir: parameter on the top line of the display. For instance if you wished to select the directory STRINGS using the "Current" button, you would first press "Open" to display the contents of that directory, and then press "Current". If you instead wanted to choose the root directory, you would simply press "Current", since that is the directory you are viewing (notice the Dir: \ " at the top).

**Pressing "SetDir"** - This selects the directory you are scrolled to, such as STRINGS in the display above. This method is often quicker and more convenient than pressing "Open" followed by "Current," which does the same thing. The "SetDir" soft button is only present in the display when the scrollbar highlights a directory entry.

## Disk Mode Functions

Now that you are familiar with the basics of creating directories and moving around in the K2vx file system, it is time to discuss some of the features provided in the disk functions themselves.

### Loading Files

The **Load** button instructs the K2vx to copy a file, from the currently selected device, to the K2vx's RAM. Press the **Load** button, and a list of files stored in the currently selected device will appear. Scroll through the list of files with the Alpha Wheel or Plus/Minus buttons, then press **OK**—or press **Cancel** to return to the Disk mode page.

When you press **OK**, the Bank dialog will appear, as described in "Load Function Dialog" on page 13-18, and you'll be asked to select the memory bank to load the file into. Scroll through the list of banks with the Alpha Wheel or Plus/Minus buttons until the desired memory bank is highlighted, then press **OK**. Or press **Cancel** to back up a page and select another file to load. Once you have selected a bank to which to load, you will be asked to choose a method for loading. The method you choose determines how the objects in the file will be ordered when loaded into the bank.

### Loading Individual Objects

Since files can contain over 3000 objects, it is often useful to load only a subset of the information contained in a K2vx file. Sometimes, this capability is necessary even to be able to load certain files, if the size of the file's samples or data is greater than the K2vx's internal RAM size (in case you haven't gotten those extra SIMMs or SIPP's yet).

You can select individual objects or groups of objects (i.e. Samples, Programs, Keymaps, Effects, Songs) for loading from within a single K2vx file.

The Load Object feature is accessible from within the Load File dialog. To activate it, scroll the file list until you have highlighted the file that you wish to load objects from:

## Disk Mode

### Disk Mode Functions

```
Dir:\                               Sel:0/6                               Index: 3
BASSDRMS .KRZ                       426K
HIHATS   .KRZ                       788K
File to load: SAXES .KRZ             2510K
TOMS     .KRZ                       301K
TOMS1    .KRZ                       1400K
XCYMB    .KRZ                       1012K
Total: 5037K
Select  Root  Parent  Open  OK  Cancel
```

Press **Open** to begin the Load Object dialog. (Note: The file must have a .KRZ extension in order to load individual objects from it.) The K2vx then scans the file contents in order to present a list of all of the objects in the file. Sometimes this procedure can take a few moments, depending on how many objects are in the file. During this time, you will see the following display:

```
Reading file SAXES.KRZ [...]
```

```
Select  Next  Type  Multi  OK  Cancel
```

The soft buttons in the above display do not become active until the process of scanning is finished. When this happens, the K2vx will display a list of the file's objects, in the exact order that they are stored in the file:

```
Func:LOAD                               Sel:0/5                               Index: 1
Sample 200 Tenor Sax1                   G 2S 250K
Sample 201 Tenor Sax2                   C 3S 179K
Sample 203 Tenor Sax3                   F#3S 198K
Keymap 200 Tenor Sax                    176
Select  Next  Type  Multi  OK  Cancel
```

The objects in the list are usually grouped by type (Sample, Program, Keymap, etc.) The list can be scrolled using the Alpha wheel or the up/down cursors. The CHAN/BANK buttons on the front panel can be used for fast scrolling. The list will jump by five entries at a time, moving the entry on the bottom line to the top line.

NOTE: When scrolling through large numbers of objects (more than 100), the K2vx may sometimes pause for a few seconds if it needs to get more information from the disk file. When this happens, some gyrating dots will briefly appear in place of the Index value on the top line of the screen.

Each line in the scrollable list represents one object, and displays the object's type, ID, name, and size. Samples have additional information, the sample's root key and a stereo sample indicator:



Object Type	ID	Name	Root Key and Stereo Indicator (samples only)	Size
Sample	203	Tenor Sax3	F#3S	198K

The ID numbers are the same numbers that were used to reference the objects when the file was last saved by the K2vx. These numbers will usually be different after the objects are loaded in, depending upon the bank (i.e. 200...299) and mode that is specified for loading. There is more information on these modes below, in the section "Load Function Dialog."

The Size field is interpreted differently for samples and non-samples. For non-samples (i.e. Songs, Programs, etc.), it shows the number of bytes used by the object in the file, and hence the amount of program memory that the object will occupy in the K2vx. For samples, the size field shows the size of the all sample data associated with the object, and is displayed in kilobytes (K).

For samples, the letter 'S' after the root key indicates a stereo sample.

Due to display space constraints, if the sample's root key happens to be in the lowest MIDI octave range (i.e. C -1 through B -1), it will be displayed in a truncated form. For instance, if a sample's root key was set to G#-1, the display would read:

Sample	293	Tenor Sax	G#-	198K
--------	-----	-----------	-----	------

The status line at the top of the screen specifies the object function being performed,

```
Func: LUW
```

the number of selected objects in the list followed by the total number of objects in the file,

```
Sel: 0/5
```

and the current list index:

```
Index: 1
```

As with the file list, entering in a number from the numeric keypad will jump to the indexed entry, and typing in a large number like 9999 will go to the end of the list.

The soft buttons on this page are used for multiple selection of the objects in the list as well as for moving around the list when there are many items selected or listed. This same dialog is also used for many other functions in the K2vx, namely for saving selected objects to disk and for several object utility functions that are described later.

Here is a brief description of each button's function, followed by a detailed explanation of its operation.

- Select** - Select or deselect an object.
- Next** - Jump to the next selected object.
- Type** - Jump to the next object of a different type.
- Multi** - Go to the Multiple Object Selector page.
- OK** - Tell K2vx to proceed to load the selected objects.
- Cancel** - Exit back to the File List Dialog.

Press the **Select** button to choose the highlighted object for loading. An asterisk (\*) is placed in between the object name and the object ID for any items that are selected. Deselect a selected

object by pressing **Select** again. The asterisk will disappear. The easiest way to choose objects for loading is to scroll the list and individually press **Select** on each object you want to load.

If you only want to select one object for loading, you need not select it with the **Select** button. Instead, pressing **OK** implicitly selects the highlighted object if there are no other objects selected. If there are objects selected, however, then the highlighted object will not be loaded unless it is selected.

This screen shows 3 samples selected for loading:

Func:	LOAD	Sel: 3/20	Index:	4
Sample	201*	Machine Snare	C 4S	100K
Sample	203*	Big Bass Loop	C 4S	218K
Sample	304*	Distorted Riff	C 4S	401K
Sample	305	EPiano1	D 2	24K
Sample	306	EPiano2	D 3	25K
Sample	313	EPiano3	D 4	15K
<b>Select</b> <b>Next</b> <b>Type</b> <b>Multi</b> <b>OK</b> <b>Cancel</b>				

The **Next** button will cause the index into the list to jump to the next selected object, forward in the list. When the end of the list is reached, the search will wrap around from the beginning. If you have more than one object selected, then if you repeatedly press Next you can easily cycle through all selected items. If there are no items selected, then this button doesn't do anything.

The **Type** button jumps to the next object of a different type from the one that is currently highlighted. This is a convenient way to find a particular type of object in the list. If you want to skip over the samples and the keymaps in an object list and jump right to the Programs, press **Type** about two or three times, stopping when you notice that a Program is highlighted.

Press the **Multi** button to enter the Multiple Object Selector (described on page 13-32.) This powerful utility can be used to control the selection or deselection of many objects, cross-referenced by object types and ranges of ID numbers.

When you are all done selecting objects to load, press **OK**. As stated above, if only one object is to be loaded, it is implicitly selected if it is the currently highlighted object and there are no other selected objects in the list.

**Cancel** returns to the file list dialog, highlighting the file you just opened. You can load the entire file after pressing Cancel by pressing OK when you return to the file list.

## Shortcuts when Loading Objects

### Select All/Deselect All

Selecting or deselecting all of the objects at once can be done with the same double-presses as described for the file list dialog, namely:

**Left/Right cursor double-press -> Select All Objects**

**Up/Down cursor double-press -> Clear All Selections**

If you want to load most but not all of the items from a file (for instance, if there happens to be a Master Table in the file that you don't want to load), it may be fastest to first select all objects using the Left/Right double-press, and then manually deselect any unwanted items.

## Viewing the Name Table

The Name Table is an object that appears in files that were created using the "Save dependents by name" feature (described below, in the section on the "Save" function). This object contains a list of dependent objects needed by the other objects in the file at the time the file was saved. There is more information about this later on, however it is worth mentioning here that a highlighted Name Table object's contents can be viewed by pressing either one of the Left or Right cursor buttons.

**Left cursor OR Right cursor -> View Name Table Contents**

## Loading Dependents of Selected Objects

When you press OK after selecting one or more objects, the K2vx may ask the question:

Load dependent objects?

Yes No

This question will be asked if any of the selected objects could possibly have dependents associated with them in the file. (Remember, dependents are those objects needed by other objects; Samples are dependents of Keymaps, Effects and Keymaps are dependents of Programs, and so on.) If this question is asked it does not necessarily mean that there really are dependents of the selected objects. The K2vx will not know whether there are dependent objects in the file until it begins to read in the selected objects, and determines what their dependents are.

Answering "Yes" to the question tells the K2vx to also load the dependents. You may wish to answer "No" if, for instance, you are simply loading in a Program or a Keymap as a template for use with other objects. You can also manually select only some of an object's dependents, and then answer "No" to "Load dependent objects?" to prevent other unwanted dependents from being loaded.

To summarize, it is not necessary to select any of the dependents of an object if you plan on loading all of the dependents in. As an example, for a file containing dozens of Programs, Keymaps, and Samples, you may choose to highlight a certain Program and press OK, and answer "Yes" to the "Load dependent objects?" question. The K2vx will do the rest, by only loading in the samples and keymaps that are needed by the selected program.

Similarly, if you were only to select certain keymaps from a file, and then answered "Yes" to "Load dependent objects", the K2vx would figure out exactly what samples need to be loaded in as dependents of the selected keymaps.

## Auditioning Samples from a Disk File

Often when working with files that contain samples it is helpful to be able to hear what the samples sound like before loading in all or part of the file. It is possible to audition samples in the file, from within the Load Object dialog.

To audition a sample, first scroll to the sample that you wish to hear. Then, press either the Left or Right cursor. The K2vx will load in up to 1 second of the sample, or less if the total sample time is shorter than one second. The audition starts from the very beginning of the sample data

(note that if the first second of data is silence then you won't hear very much when the sample is auditioned.) If the loop points fall within the first second of the stored sample data, they will be loaded in as well. The K2vx display will "blink" after the completion of loading the sample audition data. When the sample segment has been loaded, it can then be played back at its root key as well as transposed up and down the keyboard.

Once a sample has been auditioned, it remains active across the keyboard until another sample is auditioned. The audition function ends when either OK or Cancel are pressed.

There must be sufficient sample RAM in the K2vx to load in one second of the sound for auditioning. This amount varies according to the sample rate of the sample, but for most samples this will be less than 100K bytes. If the auditioned sample does not play, check that there is enough free sample memory in the K2vx. It is also possible to see the following error if the K2vx object RAM is full or very near full:

**Not enough memory to audition**

### Loading Objects from Floppy Disk Files

Individual objects can be loaded from K2vx floppy disk files, with the caveat that some samples in a multiple-floppy disk file cannot be auditioned because they reside on more than one disk. These samples will still appear in the object list, but their size will be in parentheses meaning they cannot be loaded (either explicitly or as dependent objects) and cannot be auditioned.

## Load Function Dialog

### Bank Status Indicator

After you have chosen what you wish to load, you are presented with a dialog allowing you to determine what bank will be used to load in the file's data.

This indicator takes the form of an asterisk after the bank. If an asterisk is present after the bank (i.e. 400...499\*), it means that there are objects in the bank, whether they are RAM or ROM objects. ROM objects are in the 0's and 100's bank, unless you have an optional ROM block installed. In that case ROM objects are also stored in the 800's and/or 900's bank. If there is no asterisk on the line for a bank, it means the bank is empty.

In the following screen, there are user objects in the 200's and 400's bank, and possibly also in other banks that become visible when the selection is scrolled.

```
Load this file as: 200...299*
                  300...399
                  400...499*
                  500...599
                  OK Cancel
```

This indicator makes it easier to find an empty bank to use for loading, if needed.

### Loading Methods

Once you have pressed **OK** to decide on what bank to use, you will see this screen if the bank is empty:

```

Load this file as: 200...299*
                  300...399
                  400...499*
                  500...599
                  600...699
                  Append Fill Cancel

```

You will see the following screen if the bank contains any objects (in RAM or ROM):

```

Load this file as: 200...299*
                  300...399
                  400...499*
                  500...599
                  600...699
                  700...799
OvFill Overwrt Merge Append Fill Cancel

```

The soft buttons control the mode for loading and renumbering of objects from the file. The meaning of these soft buttons is:

- OvFill** First deletes all RAM objects in the selected bank, and then loads in objects using consecutive numbering.
- Overwrt** First deletes all RAM objects in the selected bank, and then loads in objects using the object ID numbers stored in the file.
- Merge** Preserve the object ID numbers stored in the file for the objects to be loaded, overwrite objects already in memory if necessary.
- Append** Try to use the object ID numbers stored in the file for the objects to be loaded. If an ID number is already in use, increment the ID number until a free slot is found.
- Fill** Ignore the object ID numbers stored in the file. Try to use consecutive numbering from the beginning of the selected bank. If an ID number is already in use, increment the ID number until a free slot is found.
- Cancel** Cancel the mode selection, and go back to choosing a bank. Scrolling to a different bank value will have the same effect as Cancel.

Typically, you will just want to use the **Fill** method. **Append**, **Merge**, and **Overwrt** try to preserve the numbers stored with the objects in the file, but this should only really be necessary if you depend on program numbers or effect numbers to be at a certain MIDI program change number. **OvFill** is like **Fill** except the selected bank (or Everything) is cleared out before loading.

**Overwrt** and **OvFill** operate in different ways after a selected bank has been filled up for a given object type (for example, after you have loaded in more than 100 programs into a bank). **Overwrt** will begin continue to preserve the objectIDs stored in the file, and will individually overwrite objects in the bank following the just filled bank. **OvFill** will not overwrite any further objects past the end of the selected bank; it instead skips over object IDs that are in use. Because of this difference, it can sometimes be faster to load in a file using **OvFill** rather than **Overwrt**. However, this only applies if the objects to be loaded would extend past the end of a selected bank.

## Disk Mode

### Load Function Dialog

---

Note that when loading into a specific bank (as opposed to loading as "Everything"), the object ID's in the file are used as follows: The "bank" digit is ignored, and the remainder of the number is used when the K2vx re-banks the object ID into the bank that you specify. For instance, if you save Program 453 into a file, and load it back into the 300's bank, the K2vx will use the number "53" when deciding upon a new object ID. If the 300's bank was previously empty, and the load mode is Append, then the Program will end up with ID 353.

For loading as "Everything", the ID number for an object stored in a file is taken literally, and not re-banked (except if Fill or OvFill mode is chosen, in which case the K2vx will use ID numbers starting from 200.)

Here is an example that compares all of the above load modes.

EXAMPLE: Starting with the following objects already stored in the K2vx internal RAM:

```
Program 200 Existing Pro91
Program 204 Existing Pro92
Program 205 Existing Pro93
Program 210 Existing Pro94
Program 211 Existing Pro95
```

Suppose you were to load file containing the following objects into the 200's bank:

```
Program 405 Electric Piano
Program 406 Stereo EPiano
Program 409 Weird Piano
Program 410 Wild Funk Piano
```

Here is what these programs would end up with as ID numbers when loaded in, depending on the load mode:

<u>Name</u>	<u>Append</u>	<u>Fill</u>	<u>Merge</u>	<u>Overwrt</u>	<u>OvFill</u>
Electric Piano	206	201	205	205	200
Stereo EPiano	207	202	206	206	201
Weird Piano	209	203	209	209	202
Wild Funk Piano	212	206	210	210	203

In the Merge case, the existing programs 205 and 210 would have been deleted. In the Overwrite case, all of the existing programs in the 200's would have been deleted.

## Multiple Selection of Files to Load

As stated previously, you can select multiple files for loading into the K2vx from within a single directory, in one operation. This is done from the file list dialog with the **Select** button.

After you have selected one or more files in this way, you will still choose a bank and mode to be used for the load process, just as with loading a single file. However, the dialog prompt will say "Load selected as":

```
Load selected as: 200...299*
                  300...399
                  400...499*
                  500...599
                  OK  Cancel
```

If you selected any macro files (.MAC extension) from the directory, then once you have select the mode for loading, you will see the question:

```
Load macros as specified?
```

```
Yes  No
```

The answer to this question instructs the K2vx that any macro files will have their macro entries loaded according to the bank and mode:

**Yes** specified in the macro entry.

**No** currently specified for this multiple file load. In other words, whatever you select for Bank and Mode will override the instructions for each entry in the macro.

At this point the files will begin to load. When all the files have been successfully loaded or the load process has been aborted, the K2vx returns to the Disk Mode page.

If there are any errors encountered during a multiple file load, such as running out of object RAM, you will be asked once if you wish to abort the load. In some cases, you may wish to continue loading. If you continue (and don't abort), the only way to abort will be to use a special procedure described in the next paragraph.

## Aborting a Multiple File Load

There is a way to abort the process of loading multiple files. Aborting can only be done "in between" files that are being loaded, and not during the load of any one file (short of powering off or soft-resetting the machine by pressing +/-, 0, and CLR simultaneously, but this is not recommended!).

Aborting a multiple file load is done by pressing and holding down either of the Plus (+) or the Minus (-) buttons that are located just below the Alpha wheel. This should be done at least one-half second before you anticipate the current file to finish loading, or else the K2vx will not sense that you wish to abort the load.

You will see the following question after the current file being loaded is completed:

Abort the load?

Yes No

It may be a good idea to practice using this method of aborting a multiple file load, so that when the time comes that you accidentally select 100 files, you will remember how to abort the process. This same method (of holding the Plus or Minus buttons down) is also used to abort the Backup feature and the Macro file load feature.

If you run out of object or sample RAM, you will have one opportunity to abort the load as explained above. However, if you continue from that point you may end up seeing the same error message "Memory is full" for each file that you had selected. This can be a rather tedious process, however it is still possible to abort out of this by holding down the Plus or Minus button simultaneously while pressing "Yes" when you see the following question:

Abort this Partial load?

Yes No

### More Load Function Enhancements

There are more features having to do with the Load function that are described later on in this guide (i.e. loading Macro files, loading AIFF files, etc.)

## Saving Files

The Save button starts the process of saving from the K2vx to the currently selected device. When you press the **Save** soft button on the Disk Mode page you will see the bank dialog:

```
Save selection: 200...299
                300...399
                400...499
                500...599
Export Macro Object NewDir Ok Cancel
```

The **Macro** soft button will only be present if Macro recording is turned on. (See page 13-38 for more information on Macros.)

You can save an entire bank of objects, or by pressing the **Object** soft button, select individual objects to be saved. If you choose to save using the bank method, all RAM objects within that



bank will be saved. (You cannot save ROM objects. If you wish to save a ROM object, such as a program, you must first save it internally as a RAM program.) If any objects within the selected bank have dependent RAM objects that exist in a different bank, you will be asked if you want to save dependent objects. See page 13-26 for more on saving dependent objects.

Use one of the data entry methods to select a bank to be saved. If you press the **Cancel** soft button, you'll return to the Disk mode page. After you've selected the bank, press **OK**. The following page will appear:

Save as:                    **NEWFILE**

**Delete** **Insert** **>>End** **Choose** **OK** **Cancel**

You can now name the file according to the naming procedures outlined in Chapter 5. You can enter up to eight characters. When you've entered a name, press "OK" to save the file as shown in the display, or press **Cancel** to return to the file dialog. When the file is saved, the K2vx adds an extension (.KRZ) to the filename. This enables the K2vx to recognize it as a Kurzweil file when it examines the disk's directory.

### **Saving Master and Everything Files**

Among your choices in the Bank dialog are Master files and Everything files. Master files consist primarily of the items on the Master mode page and the three MIDI mode pages. They also include information like marked pages, view settings, and MIDI channel and program assignment. In fact, saving Master files (or dumping them via SysEx) is a good way to configure your K2vx (or another K2vx) to your performance or sequencing needs. For example, you might save different Master files with every sequence you create using an external sequencer. Then, when you load the Master file, you would have all the correct programs assigned to the appropriate MIDI channels.

Everything files consist of the Master file parameters and every other RAM object. Saving an Everything file will literally save everything in RAM, including samples, into a single file. These can be quite large, so if you don't have a hard disk, be sure to have a few preformatted floppy disks handy.

### **Split Files**

When you're saving memory banks, it's not uncommon to create files larger than 720K or even 1.4M—especially when you're saving RAM samples. If you're saving to floppy disks, the K2vx will create split files stored on multiple disks. A little advance preparation is necessary for this.

If the K2vx fills a floppy disk before it finishes saving a file, it will prompt you to insert a second disk in the floppy drive. This disk must be preformatted; the K2vx will not format disks in the middle of a save operation. When the second disk is inserted, the K2vx will continue to save. This process will be repeated until the entire file is saved. The K2vx marks these disks internally with a number that indicates the sequence in which they were saved.

When you're loading these split files, they must be inserted in the same sequence as they were saved. When the K2vx has loaded the contents of the first disk, it will prompt you to insert the second disk, and so on. Make sure to label your disks as soon as you save to them, so you'll keep them in the correct order.

## Soft Buttons in the "Save Selection" Dialog

The meaning of the soft buttons in the "Save selection:" dialog is as follows:

- Export** Save a sample or a song in an exported file format (i.e. AIFF, WAVE, MIDI Type 0). This feature is described in the section "Using Standard File Formats."
- Macro** Save entries from the current macro table as a Macro file (.MAC). This soft button is only displayed if macro file recording is on. This is described later in the section "Saving Macro Files."
- Object** Save selected objects from the K2vx's RAM.
- NewDir** Create a new directory on the current disk, and return to this dialog afterwards. This is described previously in the section "Creating Directories."
- OK** Save all the objects from the highlighted bank (i.e. 200...299), and optionally also save dependent objects.
- Cancel** Exit from the Save function.

**Export**, **Macro**, and **NewDir** are all explained elsewhere in this guide. This section will describe the process of saving K2vx objects into K2vx format disk files.

## Saving Individual Objects

You can select any group of objects in the K2vx's RAM for saving into a single file. To save individual objects, from the above dialog, press **Object**. The K2vx will display a scrollable list of all the objects in RAM, very similar to the display for the Load Object feature (described previously):

```

Func: SHVE      Sel: 0/8      Index: 1

Sample 500 Lo Vocal      A 3S      250K
Sample 501 Hi Vocal      G 4S      179K
Keymap 500 VocalsMap      176
Program 500 Dry Vocals      270
Select  Next  Type  Multi  OK  Cancel

```

The procedures for saving objects are essentially the same as the procedures described on page 13-13 for loading objects.

## Shortcuts when Saving Objects

### Select All/Deselect All

Selecting or deselecting all of the objects at once can be done with the following double-presses (two front-panel buttons simultaneously pressed):

- Left/Right cursor double-press -> Select All Objects**
- Up/Down cursor double-press -> Clear All Selections**

If you want to save most but not all of the items from a file (for instance, if there are some songs in RAM that you don't want to be saved in the file), it may be fastest to first select all objects using the Left/Right double-press, and then manually deselect any unwanted items.

## Viewing Selected Objects

When there are lots of objects selected, but they are scattered in the objects list, it can be helpful to be able to view a list of only the currently selected objects. Such a list is displayed by double-pressing the CHAN/BANK buttons to the left of the K2vx's LCD display.

### Double-press of CHAN/BANK buttons -> View Selected Objects

For instance, if there were 10 objects selected, and you pressed both CHAN/BANK buttons simultaneously, the K2vx would show a list similar to this:

```
View Selected Objects 10/134
Program 300 Biggest Kit          7124
Program 301 RePercussions        7124
Song 300 Drum Groove 7         12092
Song 301 Drum Groove 8         24700
Song 421 Nasty Funk            3122
Song 500 Beethoven             1024
OK
```

The top line shows 10 objects selected out of 134 that are in RAM. If the number of selected objects is larger than the 6 objects that fit on one screen (as in this example), the list can be scrolled to view all of the information.

Note that this feature is not available in the Load Object dialog.

## Auditioning Objects in RAM

When deciding which individual objects to save, it can be difficult to know if you are selecting the correct ones. This is especially true if many objects have similar or identical names, or if the names of the objects are not descriptive enough to know what they are. The K2vx has a feature that allows auditioning of Samples, Keymaps, Programs, and Songs right from the Save Object dialog (as well as all of the other object utility dialogs that are discussed later.) To activate this feature, scroll to an object of an appropriate type to be auditioned, and press either the Left or Right cursor button. The screen will "blink", and the objects can now be heard as follows:

**Samples** play at their root key, as well as transposed across the keyboard. Stereo samples will play in stereo. Auditioning samples in this way is similar to listening to samples from the MASTER->Sample page. The samples are auditioned using a "hidden" program set up according to the parameters in **Program 199 Default Program**. This default program can be customized if needed by editing and saving a new program 199.

If you audition any sample objects, the last one that you audition will become the "preview" sample the next time you go to the MASTER->Sample page. This can be a quick way to edit the sample without having to edit a program and a keymap.

**Keymaps** are reproduced accurately, and are played according to the parameters in Program 199 Default Program. This default program in ROM is set up to have a 0% effects level (dry). Therefore, auditioning keymaps can be a very convenient way to hear them isolated from the effects.

**Programs** play exactly as they would if they were selected from the Program Mode page.

**Songs** start playing when either the Left or Right cursor button is pressed, and stop playing when either cursor is subsequently pressed. The most recent song that is

auditioned from this page become the Current Song (as seen on the **Song Mode** page).

Once auditioned, the above object types remain active on the keyboard until another object is auditioned, or **Cancel** is pressed. If a song is being auditioned, no other objects are auditioned until the song audition is stopped (by pressing one of the Left or Right cursor buttons).

## Saving Dependent Objects

When you save a file, you may see a prompt as part of the Save dialog that asks you whether you want to save dependent objects. A dependent object is simply an object that's associated with another object. The dependent object can be stored in a different memory bank, for example, a RAM sample with ID 301 that's used in a program with ID 402, or in the same bank as the file being saved. Rather than forcing you to save dependent objects separately and to keep track of them yourself, the K2vx gives you the option of automatically saving the dependent objects as part of the file you save. When you load the file again, the dependent objects will be loaded along with the objects to which they're attached.

There are a few things to keep in mind regarding dependent objects. First, it's not uncommon for RAM samples to be dependent objects, and they can take up quite a bit of memory. If you save a RAM sample as a dependent object when you save a bank of programs to disk, you may create files which won't fit onto a 720K or 1.4M floppy disk. Since the K2vx can't format floppy disks in the middle of a save operation, you should have spare formatted disks ready to go before you start saving. See the section called "Split Files" on page 13-23.

While the K2vx makes it easy for you to keep track of your dependent objects, you need to keep aware of what happens with dependent objects when saving to disk and reloading. Consider this example. Suppose you create 30 new programs, each of which uses a keymap containing four different RAM samples. If you save these programs to a disk file, and save dependent objects with them, you've created a file containing 30 programs and 120 dependent RAM samples. So far, so good. Suppose you then load that file into the 300s bank. The K2vx will load the 30 programs into the 300s bank just fine, but it will be able to load (at most) only the first 100 dependent objects to the 300s bank (each memory bank can hold a maximum of 100 objects of a given type). The remaining 20 dependent objects will be loaded into the 400s bank. If there are no objects of the same type in the 400s bank, there's no problem. But if there are objects of the same type in the 400s bank, some or all of them will be replaced by the newly loaded dependent objects.

The easiest way to prevent this is to make sure that you don't create more than 100 dependent objects attached to the other objects in a given memory bank. The easiest way to do *this* is to avoid creating dependent objects when possible, by saving objects with IDs in the same memory bank as the objects to which they're related. For example, if you create a program that uses RAM samples, and you save the program with ID 201, resaving the RAM samples used by that program with IDs in the 200s will prevent dependent objects from being created for that program. If you do this, you'll minimize the number of dependent objects you create, and you'll be unlikely to force dependent objects to be loaded into a higher-numbered memory bank when you load files.

Once you have selected objects for saving (either individually as just described or by bank selection), the K2vx will determine if any of the items chosen to save have any dependent objects in RAM that were not chosen. For instance, if you select a program to be saved and nothing else (using the Save Object feature), the program may have dependent effects, keymaps, and samples that are in RAM. Dependent objects that are in ROM (i.e., ROM samples or keymaps) do not get saved to disk.

You will see the following dialog displayed if there are any dependent objects in RAM of any objects that were selected for saving:

## Save dependent objects?

**Names** **Yes** **No**

Choosing **Yes** will cause any dependent objects to be saved in the file together with the selected objects. Choosing **No** means that unselected dependents will not be saved. The **Names** button creates a new kind of object to be stored in the file, called the Name Table.

### The Name Table

The Name Table is a list of any dependent objects that were not explicitly selected for saving in the file. Each entry in the Name Table contains the object type, object ID, and the name of a dependent object.

A file's Name Table is used by the K2vx at only one time, when the file is loaded in. At that time, the K2vx will search for dependent objects that were not saved in the file originally. The search matches dependent objects by name with objects that are already in RAM, and links them to the "parent" object. The Name Table data is then discarded when the file load is finished. This search feature is referred to as **Relink-by-Name**.

Relink-by-name offers new and very efficient ways of working with K2vx objects and disk files. Careful use of this feature can save you many megabytes of disk storage, as well as a lot of time spent working on music and production instead of waiting for sample data to be resaved.

Relink-by-Name allows you to save objects and their dependent objects separately (in multiple files) and be able to link them up later on by loading the files in the correct order. This can be a very efficient way of working with the K2vx's many levels of dependent objects. The most common way in which Relink-by-Name speeds up development of sounds is when making small adjustments to a program that has as its dependents a large amount of sample data. you can separate the program and sample data, so that after changing a program parameter, only a file containing the program and a Name Table need be re-saved.

When loading in a file that contains a Name Table, the following rules should be observed in order for correct relinking to occur.

1. **Use unique names for dependent objects at every level.** As an example, this would mean that if you were going to be re-linking several samples from one file with a program and a keymap from another file, that each sample should have a different name. Otherwise, the dependent objects (the samples) will not get re-linked properly. This will create problems such as keymap ranges that don't play as they are supposed to.
2. **The dependents to be re-linked must already be loaded.** Otherwise they will not be found and re-linked when the file containing the parent objects is loaded in. This constraint on the order of file loading can be made easier to work with by using the Macro file feature (described later). You can construct a Macro file to automatically load in the dependents files and the parent files in the correct order, making sure that any files containing dependents are loaded first. An alternative to loading the files with a Macro would be to save the dependent and parent files in the same disk directory with similar file names such that they will appear consecutively in the alphabetized file list. Once you have done this, it is easy to select both files for loading in the correct order.

These rules may appear complicated at first, but they will seem natural once you have worked out a few examples with your own files.

The **Search Algorithm** used for relinking dependent objects to their parent objects during loading is as follows:

**The search for a dependent object (whose name matches that of an entry in the name table) begins at the beginning of the bank that is specified for loading the parent file.** All possible IDs are then consecutively searched. When the last ID of the 900's bank has been searched (typically 999), the search will wrap around to ID 1 up until the end of previous bank to the specified bank. The search stops once a dependent with a matching name has been found and relinked.

For example, if a file containing a one-layer program is loaded into the "400's" bank, and the file includes a Name Table that lists the layer's keymap by name, then the K2vx will begin to look through all possible keymap IDs starting at 400, until ID 999. The search then continues from ID 1, stopping at ID 399. If the search does not successfully find a match, the dependent will be unresolved, and in this example the program would have an "*object id* Not found" indication in its Keymap parameter, where the *object id* is the value that was stored in the file.

The search is done in this "circular" manner so as to allow you to direct which dependent objects get re-linked. This may be necessary if you end up with multiple copies of dependent objects with the same name; you can differentiate between them by loading the parent file into a specific bank that is the same bank or "before" the bank containing the objects you wish to re-link to. Note that this can only be taken so far, since it would be impossible for the K2vx to differentiate between objects with the same name within the **same** bank.

The re-linking process happens in the background, without any notification or error messages if items cannot be relinked.

## Working with Relink-by-Name

Here are a couple of more in-depth examples that can show how Relink-by-Name works in a practical situation.

Consider that your K2vx's RAM contains the following one-layer program and also its dependent keymap and samples (the techniques used in this example could well apply no matter how many programs with any number of layers you want to save):

Program:           **Program 317 Steinwave Piano**  
Keymap:           **Keymap 300 Steinwave Piano**  
Samples:           **Sample 300 StwaveG1 ..... Sample 310 StwaveC7**

In this case you might wish to save the samples and the keymap in one file, and the Program in another file. So, from the Save Object dialog you could first select all the Samples 300-310, and Keymap 300, for saving into a file, let's say **STWAVE1.KRZ**.

You would then return to the Save Object dialog and save just Program 317 in a separate file in the same directory, let's say **STWAVE2.KRZ**....only this time, you will be asked the "Save dependent objects" question pictured above. Answer this by pressing **Names**.

After saving, the file STWAVE2.KRZ will contain two objects in it, Program 317 and a Name Table. You can easily verify this by going to the Load function (or any other disk function) and

pressing "Open" on the file just saved (which should come up already highlighted). The display of objects for the file will look like this:

```
Func: LOAD      Sel: 0/2      Index: 1
```

```
Table 36 Names      334
Program 317 Steinwave Piano      274
```

```
Select  Next  Type  Multi  OK  Cancel
```

The Name Table will always be the first object in the list. You can verify the exact contents of the name table by using the "View Name Table" shortcut (previously described in the Load Object section); make sure the Name Table object is highlighted, and **press either the Left or Right cursor button** on the front-panel (as if you were "auditioning" the Name Table object.) You would then see the following informational display:

```
Name Table Contents
Keymap 300 Steinwave Piano
Sample 300 StwaveG1
Sample 301 StwaveD2
Sample 302 StwaveB2
Sample 303 StwaveE3
Sample 304 StwaveB3
Sample 305 StwaveG4
```

OK

The **Name Table Contents** display shows what would have been saved in the file had you answered Yes to "Save dependent objects?" instead of answering by pressing Names. More importantly, it allows you to see what objects will need to be already loaded in to the K2vx before loading this file.

The object IDs shown in the table are the same numbers that those dependent objects used at the time this file was saved. (The ID numbers are necessary in order for Relink-by-Name to function, since they are the "link" between the higher level objects and the names of the dependents.)

An important thing to notice about this particular Name Table is that the Sample names are not needed by the K2vx for relinking purposes. In fact, the only information necessary for relinking the dependent objects of this file is the Keymap object. The reason for this is that when this file containing the program is loaded, all of these dependent objects should already have been loaded, and the Keymap should already be correctly linked to the Samples. Although the Samples' names are redundant from the K2vx's point of view, they are include for free, so to speak, and you may find them very helpful if you ever need to know exactly what the dependents of this file were intended to be.

The Name Table Contents List is scrollable if there is object data past the 7 lines that fit on the screen.

Now that the two files STWAVE1.KRZ and STWAVE2.KRZ have been created using the Name Table, they can be loaded back in and correctly relinked. The files can be loaded into any bank - they do not need to go back into the bank they were originally in - since the STWAVE2.KRZ file will search through all the banks to find the objects by name in order to relink them. In fact, if you were to immediately reload just the file containing the Program (STWAVE2.KRZ), into any bank, you would find that it was automatically relinked to the correct Keymap, since the Keymaps and samples are currently in memory.

Furthermore, you could edit the program and create more variations of it that reference the "Steinwave Piano" keymap, add ROM layers, and / or effects programs if desired, and resave all of the programs (and any effects) to the same or a new file (remember to press **Names** when you are asked "Save dependent objects?".) You never have to resave the file STWAVE1.KRZ that contains the keymap and samples, if all you have done is edited the programs or added more of them. This can be a tremendous time-saver.

If the Keymap and Sample files are found on a CD-ROM disk, then using Relink-by-Name is not only a time-saver, but a disk space saver as well. If you like the samples and keymaps from a CD-ROM file, there is no need to duplicate the sample data on your own writable hard drive. Instead, all you have to do is save a program file in the above manner, and then make sure the CD-ROM file is loaded first before you load the program file.

If you needed to add some sample data to the file (for instance, you want to add a root to the keymap or process and reloop a sample from the CD-ROM), you can do this by explicitly selecting the new sample data and the keymap for saving along with the program and the Name Table. Then, the new sample would not be listed in the Name Table (it would be in the same file as the Name Table), and the Keymap would be relinked to all of the samples by name instead of the program being relinked to the keymap (as before). What you put in the different files is up to you, and there is no limit to where you can break up the objects in one file or another. The main thing to be aware of are the two rules for Relink-by-name mentioned above:

1. **Always use unique names for like objects types.** (NOTE: In cases where duplicate names do exist in different banks, load the file that contains the name table into a bank equal to or preceding the bank in which the file with dependent objects was loaded.)
2. **Files containing dependent objects must be loaded first.**

As you will see later, you can create a **Macro file** that will automatically load both of the files in the correct order, no matter what drives they are on or what disk directories they are in. By using Macro files in this way, you can avoid having to explicitly load 2 files and remember the correct order each time.

You can also use the **Multiple Object Selector** (described in the next section) to help in the process of identifying dependent objects and parent objects that you want to place into separate files. For instance you could easily select all dependent Keymaps and Samples of any group of programs, to create a "dependents" file. Then, you could quickly select the programs and any other objects that you wanted to be relinked later on, and save them in another file.

Here is another practical example using Songs (sequences). Suppose you have loaded several files into your machine, such that you now have all your favorite instruments on-board in the K2vx's RAM. Then, you make a bunch of songs using a combination of ROM programs and the RAM programs you loaded in.

The dependent object structure of the songs would look something like this:

Songs:	<b>Song 400 Wild Jam ,Song 401 Memphis Groove</b>
Programs:	<b>Program 600 Drawbarz   Program 245 FendJazzBass Program 231 Funky GTR   Program 400 ObieWarble Pad Program 103 3 Layer Kit (from ROM)</b>
Effects:	ROM effects
Keymaps, Samples:	Lots of them...

In this case you might want to save all of the songs in one file, and be able to automatically relink the dependent programs used by the song tracks. All of the programs are presumably already saved in separate files. The only file that needs to be created is one that contains all of



the Song objects, plus a Name Table. Once again, this is done by selecting the Songs from the Save Object dialog, and answering Names to "Save dependent objects?". The contents of this file can then be displayed by pressing Open (as was done for the previous example).

```

Func:LOAD  Sel:0/2  Index: 1
-----
Table 36 Names 700
Song 400 Wild Jam 12114
Song 400 Memphis Groove 34002
-----
Select Next Type Multi OK Cancel

```

Also as shown in the previous example, you can display the contents of the Name Table:

```

Name Table Contents
-----
Program 231 Funky GTR
Program 245 FendJazzBass
Program 400 Obie Warble Pad
Program 600 Drawbarz
Keymap 220 Funk Guitar
Keymap 229 Jazz Bass
-----
OK

```

Notice that the ROM program, "Program 103 3 Layer Kit", will not be listed in the Name Table. Any dependent objects that are in ROM do not need to be relinked by name. ROM objects are always directly referenced by their object ID number, since they don't get saved in any files.

Once the Song file has been saved, it can be loaded in at any time and correctly relinked, as long as the other files containing the necessary programs have already been loaded.

For this type of situation, where you may be working on songs always using a consistent set of programs, it is beneficial to make a Macro file that can be loaded in one step to direct all of the various program files to be loaded in. After that, any time you load in a song file containing a Name Table referencing these programs, the songs should get relinked to the correct programs.

If you happen to have multiple copies of the necessary programs already loaded into different banks, you can control which bank of programs will be linked to the songs by choosing a certain bank to load the song file into. The relinked programs will be the first set encountered according to the **Relink-by-Name search algorithm** defined above.

## Not Loading the Name Table

There may be a time that you wish to load in objects from a file containing a Name Table, but you don't want the K2vx to relink any dependent objects according to the Name Table. This can be accomplished by "Opening" the file from the Load function, and selecting any desired objects from within the file, **except** the Name Table object. The selected objects will be loaded in to the bank you specify, however the Relink-by-Name mechanism will not function.

## Relink-by-Name Processing Time

Normally, the time taken to relink several dependent objects using the name search will be insignificant, relative to the time it takes to load in the data from the file. However, if you are attempting to relink a very large amount of dependents by loading one file (i.e. say, 200 samples or so), there may be a noticeable wait while the K2vx searches its object database for the dependents. If this happens, it's best to be patient.

## The Multiple Object Selector Page

The Multiple Object Selector is used for making multiple selections of objects from an object list according to combinations of selection modes such as object type, object ID, and dependencies. This page is available for use in the **Load Object** dialog, the **Save Object** dialog, and all of the object utilities that are described in the Master mode chapter (**Move, Copy, Name, Delete, and Dump Objects.**)

The Multiple Object Selector page is entered by pressing the **Multi** soft button when in the dialog pages for the above functions. This page contains parameters to define a group of objects cross-referenced by object type and object ID. Objects can be grouped according to the dependencies of one or more objects, for all of the object functions *except Load Object*. (As described previously, the Load Object function can not determine exact object dependency relationships until the objects are actually being loaded into the K2vx. The other object functions operate on lists of the objects currently stored in the K2vx's RAM, and because of this, dependent relationships are available to be used as selection criteria.)

This is what is displayed by the K2vx when the Multi button is first pressed:

```
Multiple Object Selector
Select :Type/Range
Type   :Sample
Bank   :200's
StartId: 200      EndId: 299
```

```
All  Type  Toggle  Clear  Set  Cancel
```

This is the screen from which you can specify a range of object ID numbers cross-referenced with object type. If you increment the **Select** parameter to the **Dependents** setting, you will see the following additional parameter layout (not available for Load Object). This is used for selecting dependent objects of either the object currently scrolled to (from the object list that you were just viewing before pressing Multi) or dependent objects of the selected objects (indicated with an asterisk in the object list):

```
Multiple Object Selector
Select :Dependents
Of     :Current Item
Specify:All
```

```
Current = Program 205 Viola Section
```

```
All  Type  Toggle  Clear  Set  Cancel
```

The "**Current =**" parameter is for information only and represents the currently highlighted item in the object list that you are working with.

Two other selection modes do not have any additional parameter associated with them. These are **Everything** and **Search String**.

```
Multiple Object Selector
Select :Everything
```

```
Multiple Object Selector
Select :SearchStrg
```

"Select: Everything" is one way to choose all objects in the list. The All button performs a similar function, as described below.

"Select:SearchStrg" allows you to enter a string to be searched for in all of the object names in the list. Any objects with names containing the string can be selected. (Like "Dependents", this mode is available for all functions except Load Object.)

## Multiple Object Selector Soft Buttons

Here is a functional description of the soft buttons displayed on the Multiple Object Selector page. This is followed by an explanation of the parameter operation for all of the above object selection modes.

**All** Returns the Select parameter to **Type/Range**, if it was not already set that way. Sets the Type to **All Types** and the Bank to **All Banks**, and also the StartId to **0** and the EndId to **999**. This soft button is provided as a quick way to define the selection range as "Everything". Once the Type and Bank parameters have been set up in this manner, it is easy to change either parameter to further define the selection range (i.e. All Types of objects in the 400's bank, or Programs in All Banks)

**Type** Returns the Select parameter to **Type/Range**, if it was not already set that way. Sets the Bank to **All Banks**, and also the StartId to **0** and the EndId to **999**. The Type parameter is set so that it **matches the type of the currently indexed object from the object list**. For instance, if you were scrolled to a Setup object when you pressed Multi, pressing the Type soft button would set up the Type parameter to Setup. This is usually used to quickly select or deselect all objects of a particular type by scrolling to the first object of that type, and then pressing **Multi->Type->Set** or **Multi>Type->Clear**. If you don't want to include all banks in the selection range, it is easy to adjust the Bank or ID parameters to narrow the range.

The following three soft buttons will trigger the selection and deselection of items from the current object listing, according to the function of the particular button. Only objects that are included in the specified range will be selected or deselected. When this process is completed, the display returns to the object function dialog.

**Toggle** For each of the objects in the specified range, "**toggle**" the selection status of the object. In other words, if an object is already selected, then the object will be deselected (the asterisk will go away). If an object is not already selected, it will be selected (as indicated by an asterisk in the display between the object ID and object name.)

Toggle is very useful for situations where you want to select all objects in the list except those that meet certain qualifications. For instance, in the Delete Object utility (from the Master page, press **Object->Delete**), you may want to delete all objects that are not being used by a song that you are working on, in order to free up sample or object RAM in the K2vx. You could do this by first selecting the song and its dependents (with no other objects selected, highlight the song, press **Multi**, then select "Dependents" of "Current Item" and press **Set**) and then returning to the

Multiple Object Selector page, set the selection range to Everything (or press the **All** soft button), and press Toggle. This will return you to the object list display, with everything that is **not** in use by the song now selected for deletion. All you would have to do then is press OK and confirm the delete operation.

- Clear**    **Clears the selection status of all objects within the specified range.** A basic example of this would be if you wanted to save a file containing everything in RAM except Programs. You could first select all the objects with the Left/Right cursor button double-press, and then go to the Multiple Object Selector. From there you could set the selection range to be Programs in All Banks, and press Clear. The K2vx would return to the object list display, with the only the desired objects selected for saving.
- Set**      **For each of the objects in the specified range, select the object.** For Example, to select all Keymaps and Samples in the 300's bank for saving, you would first set the selection criteria to Type/Range, set the Type to Samples, set the Bank to 300's, and press Set. Then, you would press Multi again, and just change the Type parameter to Keymap, and press Set again. The desired range of objects would then be selected and ready to proceed with saving.
- Cancel**    Exits from the Multiple Object Selector page, with no action, and returns to the object function, with the previously displayed object list.

## Entering Selection Criteria in the Multiple Object Selector

This section describes the operation of the selection modes provided on the Multiple Object Selector page. These are accessed by scrolling the **Select:** parameter to different values, as pictured above.

### Selection by Type/Range

This mode sets a specific range of object types cross referenced with a specific ID range.

Parameters	Possible Values	Function
<b>Type</b>	Sample, Keymap, Effect, Program, Setup, QABank, Vel-Map, PrsMap, IntTbl, Song, Table, All Types	Sets the desired object type. The value <b>All Types</b> will select all of the other possible types.
<b>Bank</b>	000's, 100's, 200's, 300's, 400's, 500's, 600's, 700's, 800's, 900's, All Banks	<p>Sets the desired bank. Changing this parameter causes the StartId and the EndId to be set to the limits of the chosen bank (i.e. setting this to 300's will set the StartId to 300 and the EndId to 399). Setting this parameter to the value All Banks will set the StartId to 0 and the EndId to 999.</p> <p><b>The actual range used for selections when Toggle, Set, or Clear is pressed is taken from the setting of the StartId and EndId parameters.</b> For example, if you set the Bank to 200s and then change the StartID to 300 and the EndID to 399, the 300s bank will be selected, not the 200s.</p> <p>The Bank parameter is used as a quick way to set up the Id range for an entire bank, or all banks.</p>
<b>StartID</b>	0 - 999	Sets the specific starting ID of the selection range.
<b>EndId</b>	0 - 999	Sets the specific ending ID of the selection range.

It is possible to set the EndId before the StartId. If this is the case, no selection range will be enabled.

### Selection by Dependents

This mode is used to select a group of objects that are dependents of other objects. This is not available in Load Object

Parameters	Possible Values	Function
<b>Of</b>	Current Item, Selected Items	<p>If set to <b>Current Item</b>, selection range is confined to those objects in the object list that are dependents of the currently indexed item (<b>Current =</b>), including the currently indexed item itself.</p> <p>If set to <b>Selected Items</b>, then the selection range includes any objects in the object list that are dependents of any currently selected objects. The currently indexed item that is displayed is ignored unless it is already explicitly selected (has an asterisk).</p>
<b>Specify</b>	All, All->Keymap, All->Program, Keymap->Sample, Samples Only	<p>This parameter is used to limit which dependent objects are included in the selection range for the appropriate objects included via the <b>Of</b> parameter. The normal setting is <b>All</b>, which means all dependents are included. The other settings are primarily useful when separating objects into different files for reloading later using Macros and Relink-by-Name.</p> <p>If set to <b>All-&gt;Keymap</b>, then the selection range will include any dependent objects down to the level of key-maps. That is, samples will be excluded from the selection range.</p> <p>If set to <b>All-&gt;Program</b>, then the selection range will include any dependent objects down to the level of programs and effects. Another way of saying this is that any Keymaps and Samples will be excluded from the selection range.</p> <p><b>Keymap-&gt;Sample</b> cause the selection range to only include dependent objects that are keymaps or samples.</p> <p><b>Samples Only</b> will just include sample dependents.</p>
<b>Current =</b>	Type, ID, and name of the currently indexed object	Displays the object that will be used if Current Items is the value of the <b>Of</b> parameter.

### Everything

"Everything" will include all objects in the list. You may prefer to use the All button for this purpose.

### Search String

This selection mode will ask for a search string to be entered, as soon as either the Toggle, Clear, or Set button is pressed. The range for the selection/deselection will be any objects whose names contain the search string, ignoring upper/lower case. Search String mode is not available in Load Object.

## Working with the Multiple Object Selector

The Multiple Object Selector can be helpful to avoid a lot of button presses and quickly allow you select whatever group of items in the K2vx 's RAM that you want. It is available in all of the related object management functions.

You may notice that the cursor positions and parameter settings are remembered whenever you exit the Multi Selector dialog, even if you exit the dialog and choose a different different function. For instance, if you end up doing a lot of selecting of samples, or of dependents at various levels, the parameters will stay set up the way you left them as you move from function to function (i.e. from Copy to Delete to Save, etc.)

The "Select Dependents" mode is very useful not just for saving dependents, but also for splitting up groups of objects for placing in different files. By using the optional settings for the Specify parameter (**All->Keymap**, **All->Program**, **Samples Only** etc.), you can separate the group of objects that you want to save at any level of the object tree that is necessary.

### Examples of possible operations using Multiple Object Selector:

- Select all the keymaps that are dependents of a block of programs.
- Select all the samples starting from ID number 398.
- Select all the objects that have "piano" in their object name.
- Select the programs, setups, and effects that are dependents of song 200.
- Select all of the keymaps and samples that are dependents of songs 400-410.

## More Features of the Save Dialog

### The Choose File Name Function

When entering in a file name for saving, there is a **Choose** soft button. This was briefly mentioned earlier in the section "Creating Directories." When Choose is pressed from the file naming dialog, the K2vx will access the current disk directory and display the following:

```

Dir:\                               Sel:0/10   Index:    1

Choose file name: BOTTLE   .KRZ   48K
                  CLAU    .KRZ   1207K
                  REGGAE   (dir)
Total:664K        STICK    .KRZ   550K
                  Root   Parent   OK   Cancel

```

The function of this dialog is to grab the text of any file name on the current disk, and either use it as a starting point in the file naming dialog, or else use the chosen filename exactly. This helps when replacing files on the disk (where the name must exactly match the file being replaced), or adding files to the disk that have similar names or appended revision numbers. You can save time by not having to type the entire file name in from the K2vx's keypad.

The **Open** soft button is only visible in "Choose file name:" dialog when a subdirectory is highlighted.

Traversing directories from the Choose function does not change the current default directory.

### Selecting the Directory to use for Saving a File

After you specify the file name when saving any file, you can then choose either the default current directory for saving the file in, or any other directory on the current disk:

```
Use current directory for BOTTLE.KRZ?  
(Path = \)
```

```
Change  OK  Cancel
```

Pressing OK will select the default path (the current directory), which in this display is the root directory, "\". Pressing Change will allow you to view the disk, traversing its directories, until you find the one in which you want to save the file. If you choose a different directory from the default, it will become the new default directory. For more information on selecting a directory, see "The Directory Selection Dialog", near the beginning of this chapter.

### Saving Any File sets the File Index

After saving a file, you can go to any disk function (such as Load), and the just saved file will be automatically highlighted. This makes it easy to find a file that you have just saved, in case you want to delete it, add it to a Macro, move it to a different directory, open it (if it is a.KRZ or a .MAC file), etc.

## Macros

The K2vx lets you create lists of disk files called Macros. The files can be located on any drive, whether SCSI based or the internal Floppy drive. Files from SCSI drives in Roland and Akai format can also appear in Macros. Ensoniq files are not currently supported by K2vx macros.

Macros are stored in a data object called a Macro Table, and these can exist in two forms:

1. A Macro Table object in the K2vx's non-volatile RAM.
2. A disk file, containing one Macro Table object. This disk file is called a Macro file, and it has a .MAC extension (visible in the directory listing.)

Macros are primarily used to configure a K2vx with sound and sequence data from several files, or selected objects within files. When a Macro file is loaded, every file that is listed in its Macro Table will be loaded in, according to the order of the entries in the Macro Table.

### The Macro Page

There can only be one Macro Table in the K2vx's memory at any time. This object is created for the first time by turning on **Macro Record** mode, from the Macro page, which you reach by pressing the **Macro** soft button on the Disk mode page, as shown below.



```

DiskMode   Samples:12313K  Memory:102K
Path = \CYBER\

CurrentDisk:SCSI 2           Startup:Off
                             Library:Off
Direct Access, 84MB
Psyquest PS-427             XMC1.7
<more>  Load  Save  Macro  Delete  more>

```

The following screen is what you will see if Macro recording is Off:

```

Func:MACRO   [ Off ]   Index: 0

Select  Modify  Load  Record  On  Exit

```

The top line displays the disk function ,

```
Func:MACRO
```

the current Macro mode,

```
[ Off ]
```

and an index value into the Macro Table.

```
Index: 0
```

## Macro Modes

The K2vx has 3 Macro modes -- Record, Pause, and Off. The meaning of each of these modes is:

**Off** There is no Macro object in the K2vx memory.

**Record** A Macro object exists, and the K2vx records any file load operations that occur.

**Pause** A Macro object exists, but the K2vx *does not* record any file load operations that occur.

Note that whenever the Macro mode is Off, there are two soft buttons labeled **Record** and **On**. Pressing On will enable Macro Record mode, and then will return to the Disk Mode page. As an alternative, pressing Record will also enable Macro Record mode, but the display will remain on the Macro page. Once you press Record, the soft buttons and the top line of the display will change. The screen will look like this:

```
Func:MACRO [Record] Index: 0
```

```
[Blank line]
```

```
Select Modify Load Pause Off Exit
```

The new Macro mode is displayed ([Record]), and the soft button that used to say Record now says **Pause**. The soft button that used to say On now says **Off**. Pressing Pause will cause the Macro mode to read [Pause ] and the Record soft button will reappear. You can switch between Record and Pause by pressing this button repeatedly.

Whenever Macro Record mode is enabled, you will see the indicator "(Macro on)" near the top left of the display on the Disk Mode page:

```
DiskMode Samples:10022K Memory: 42K
Path = \
(Macro on)
CurrentDisk:Floppy Startup:Off
Library:Off
```

```
<more Load Save Macro Delete more>
```

### The Macro Table

When Macro Record mode is enabled after being in the Macro Off state, a new object called a Macro Table gets created in the K2vx's memory. (The Macro Table is sometimes also referred to as a *Macro list*.) In the Save Object function display list, the Macro Table would appear as:

```
Table 35 Macro 14
```

A Macro Table can only be deleted from memory by pressing the **Off** soft button, or by performing a hard-reset of the K2vx. Pressing the Off button will display the following question:

```
Reset macro?
```

```
Yes No
```

Pressing Yes will delete the Macro Table from memory, and then will return to the Disk Mode page. The Macro mode is set to Off, and the **(Macro on)** indicator is no longer displayed on the Disk Mode page.

Pressing No will return to the Macro page with no action taken. The "Reset macro?" question is displayed to allow you to change your mind about deleting the Macro Table, in case you have accidentally pressed the Off button.

When the Macro Table is first created it takes up a minimal size (14 bytes) in your non-volatile RAM. With each new entry that is added, the Macro Table will increase in size by approximately 40 to 100 bytes (or possibly more if the entry specifies an individual object list). When the Pause mode is entered, the Macro Table continues to remain in memory but does not change size since file operations are not being recorded. This is useful if you need to load some files into the K2vx but you don't want them to be entered into the Macro Table.

A Macro Table gets progressively larger as more file operations are recorded into it. Because of this, you will want to control when a Macro Table is stored in your K2vx's memory. If you are concerned about keeping enough free space for all of your other objects, you may want to practice the following strategy - when you want to record a Macro, enable Macro Record mode, and when you are finished (and possibly have saved some Macro files), turn Macro mode Off. This will reclaim the storage space taken up by the Macro Table.

If you are confident that you have enough K2vx object memory installed and that you can allow the Macro Table to increase in size as you work, then you may wish to leave Macro Record mode enabled all the time. This can be useful if you want to keep a running history of files you have recently loaded in. Both the Macro mode and the Macro Table are remembered between power-cycles of the K2vx via the battery-backed memory.

A Macro can hold as many entries as there is space for in your K2vx's non-volatile RAM.

## How to Make a Macro File

This section will take you through creating, saving, and loading in a Macro file. A simple example will be used. Afterwards, you will be able to apply the example and create your own Macro files.

The first step in making a Macro file is to turn on Macro Record mode (from the Macro Page, press On), if you have not already done so.

### Creating the Macro

Suppose you have the following four files on your disk (on SCSI ID 5, in the directory \ANALOG\) that contain analog style synthesizer patches, and you would like to have one Macro file that will load them all in:

```

Dir:\ANALOG\      Sel:4/4      Index: 1

File to load:
MULTIUDX .KRZ*    98K
NOISE     .KRZ*    36K
RESONANT  .KRZ*   109K
SYNAPSE   .KRZ*   421K
Total:664K
Select  Root  Parent  Open  OK  Cancel

```

Using multiple selection, you can select all four files, as shown. When you press OK you will see the "Load selected as:" dialog, but with the extra soft buttons **Macro** and **Insert**:

```

Load selected as: 200...299
                  300...399
                  400...499
                  500...599
Insert           Macro  OK  Cancel

```

The extra soft buttons are available only when a macro is being recorded. First, select the bank that you want, as usual. Pressing **OK** means to load all of the selected files into the K2vx, and add all of the files to the Macro list. If all you are doing is creating a Macro file, you may not want to load the files at this time. However, you still would like to enter the files into the Macro list. Pressing **Macro** does exactly that. It means *don't load the files, but do enter them into the Macro list*.

When you add entries to a Macro, they will be added to the end of the Macro list by default. **Insert** is for inserting file entries at any point in a Macro list. It is discussed a bit later, in the section on Macro editing.

Once you have pressed either Macro or OK, you will see the load modes soft button menu that contains all of the various load modes (OvFill, Overwrt, Merge, Append, and Fill), whether the bank is empty or not. The reason for this is that you might need to be able to specify a mode such as Overwrite even if at the current time, the selected bank is empty. Whatever you choose here for the load mode (as well as your choice of bank) will be saved in the Macro list. For the sake of this example, let's choose the 200's bank and Fill mode. As soon as you choose the mode, the K2vx will either load in the selected files or not, depending upon whether you pressed OK or Macro. In either case, they will be added to the Macro.

### Saving the Macro File

You have now created a Macro. You can immediately verify that the selected files have been added to the Macro Table by looking at the Macro page (from the Disk Mode page, press Macro).

But right now, let's proceed with this example by saving the Macro file. From the Disk Mode page, press **Save->Macro**. You'll see the following screen:

```

Func:SAVE MACRO  Sel:0/4  Index: 1

5:\ANALOG\MULTIVOX.KRZ  200:F:
5:\ANALOG\NOISE.KRZ    200:F:
5:\ANALOG\RESONANT.KRZ 200:F:
5:\ANALOG\SYNAPSE.KRZ  200:F:
Select                All  OK  Exit

```

This is called the **Save Macro page**. The soft buttons on this page control which Macro list entries (*Macro entries*) will appear in the Macro Table that will be stored in the Macro file. You can do multiple selection of entries with the **Select** soft button. Selected entries will have an asterisk on the first character of the display line, such as this:

```

*5:\ANALOG\RESONANT.KRZ  200:F:

```

You can use the following double-presses to select and deselect all entries in the list:

**Left/Right cursor double-press -> Select All Macro Entries**  
**Up/Down cursor double-press -> Clear All Selections**

The top line indicates how many total Macro entries are in the current Macro Table, and how many are selected.

Pressing **OK** will cause only the selected Macro entries to be saved in the file. If you press **OK** but have not selected any Macro entries, then the Macro file will only have one entry in it, that of the highlighted entry.

There isn't much use for a Macro file with only one entry in it, however it can be useful as a way to create a link to an often used file on a drive. For instance, you could create a Macro file called `\PERC.MAC` in the root directory on the disk that just loads in a file `\MYSOUNDS\PERC\GOOD\PERC.KRZ`. Then, you would have fast access to loading this file without having to go find it, yet the file would still be organized in a subdirectory.

If you know that you want to save all of the entries into the Macro file (as we do for this example,) just press **All**. The K2vx will go through the standard file saving dialog in which you choose a file name and select a directory to save the file in.

Let's save the file as `\ANALOG\SYNTH.MAC`. Macro files are automatically saved with the `.MAC` extension. You will see the following on the screen as the file is being saved:

```
Writing file SYNTH.MAC...
```

```
Table      35 Macro          - 162 b
```

### Loading the Macro File

So far, so good. We have created a Macro in memory and saved it to the disk, in the same directory as the files that are listed in the Macro.

While this example only loads files from within a single directory on a single drive, this is for the simplicity of the example. You can have files from any drive or directory in a Macro.

Now, let's go to the Load page and try to load the Macro file. This should load all of the files of the Macro one by one. When you return to the Load page, the file list will automatically highlight the Macro file that was just saved (as it would after any type of file that you save):

```
Dir:\ANALOG\      Sel:0/5      Index: 5
                RESONANT .KRZ      109K
                SYNAPSE  .KRZ      421K
File to load: SYNTH .MAC      .5K
```

```
Total:664K
Select Root Parent Open OK Cancel
```

Press **OK** to load in `SYNTH.MAC`. Now the display reads:

```

Load this macro as: Specified
                   300...399
                   400...499
                   500...599
Insert             Macro  OK  Cancel

```

There are a couple of things to notice here. The first is a new choice in the bank list, **specified**. "Load this macro as specified" means load all the files in the Macro following the exact instructions for the bank and load mode for each file. All of the files were specified to be loaded into the 200's bank using Fill mode. If this is acceptable at the time you want to load the macro, you can just press OK. Otherwise, you can override the bank and mode settings for the entire Macro by choosing a different bank and mode before pressing OK (this is called *rebanking* the Macro.)

The other thing noticeable about the above display is that the Macro and Insert buttons are still available. This is because the Macro mode is still "Record". It is also perfectly acceptable to load a Macro file "into a Macro". If you do this, then **the K2vx will add all of the Macro file's entries into the in-memory Macro list**. Later on, you will see how loading a Macro file into the in-memory Macro list is a convenient way to edit a Macro file or combine elements of several Macro files into one Macro.

In this example, the in-memory Macro is still there - it didn't go away after saving the Macro file (this only happens when you turn Macro Off, as described above.) Since Macro Record is still enabled, loading in the SYNTH.MAC file will have the effect of adding a duplicate set of entries into the in-memory Macro.

Whatever method of loading you choose (i.e., specified in the Macro or overriding the Macro), the K2vx will proceed to locate each file in the Macro in the exact order in which the entries were listed when the Macro was saved. If the files were on different drives in your disk system, you would be able to observe your various drives as they were selected in turn and files were loaded from them.

If the K2vx cannot locate one of the files, you'll see a "Not Found" error message. If a drive cannot be accessed (for instance, if the SCSI ID stored with the Macro entries in this example was no longer the current SCSI ID of the drive), you will see the message "Problem mounting disk.", to which you must press **OK**. If "Confirm" is set to "On" on the Master Mode Page, loading will stop on the first error message, giving you a chance to cancel the operation or keep going. If you answer "yes" at this point, the operation will continue, even if the K2vx encounters subsequent errors. If you run into a lot of errors due to loading an out of date Macro file, the Macro process can be discontinued using a special procedure described later in the section "Aborting a Macro Load."

When the Macro is done loading, you will see the screen:

```
Macro SYNTH.MAC completed...
```

The K2vx will then return to the Disk Mode page. You should now be able to go to Program Mode, or the Save Objects display, and verify that all of the objects from the files of the Macro are now in the K2vx's memory.

## Macro Entries

Each file load operation that is recorded into the Macro Table is called a Macro entry. While Macro Record is enabled, you have the option of either loading files into the K2vx or just adding files to the Macro without actually loading them.

Each Macro entry stores information about how a disk file should be loaded. Each entry is displayed as a single item in a scrollable list on the Macro page, with various fields indicating the parameters of the entry. This picture shows how the Macro page might look once 4 files have been entered into the table:

```

Func: MACRO [Record]          Index: 1

3:\DRUMS\REALKITS.KRZ      200:F:Obj
3:\BASSES\WALKING.KRZ     200:F:
3:\KEYS\CHROMA12.KRZ     200:F:
F:\SONG42.KRZ             200:F:
Select  Modify  Load  Pause  Off  Exit

```

The following chart describes the information stored in a Macro entry, and also tells how to understand the settings of a displayed Macro entry:

<b>Disk drive ID</b>	<p>This represents what drive the file is to be loaded from. There are 10 possible values:</p> <p>The numbers <b>0-7</b> represent SCSI 0 through SCSI 7.</p> <p>The letter <b>'F'</b> represents the Floppy drive.</p> <p>The letter <b>'U'</b> means Unspecified drive. (see below)</p> <p>The letter <b>'L'</b> means the Library drive. (see below)</p>
<b>Directory path/file name</b>	<p>This is the directory path and the file name of the file on the disk to be loaded by this Macro entry. This is displayed to the right of the Disk drive ID, following a colon, using as many characters as needed that will fit in the display (up to 28). The internal Macro Table can store any path length, however.</p>
<b>Bank</b>	<p>The bank that the file should be loaded into. This will have a value from 0 through 900 (by 100s), or the letter <b>'E'</b> for Everything. This field is the first field to the right of the path/file name in the display.</p>

<b>Mode</b>	<p>The mode specified for loading in the file. The following one-letter codes are shown in the display:</p> <p>'O' means Overwrite mode (Overwrt)</p> <p>'V' means Overwrite/Fill mode (OvFill)</p> <p>'M' means Merge mode</p> <p>'A' means Append mode</p> <p>'F' means Fill mode</p> <p>This field is to the right of the bank field, after a colon.</p>
<b>Object indicator</b>	<p>If a list of objects will be loaded from the file, the Macro entry display line will have an "Obj" indicator as the last field on the right. (see above, for the file \DRUMS\REALKITS.KRZ)</p>
<b>Object list</b>	<p>If the Obj indicator is present for a Macro entry, it means a list of individual object types and IDs are stored in the Macro entry for the particular file. When the Macro is loaded in, only these individual objects will be loaded from this file.</p>
<b>Disk Format type</b>	<p>If the disk is a third-party SCSI format disk such as Akai or Roland, a format indicator will appear on the Macro entry display line in the last field on the right. The following codes are used:</p> <p>'Aka' means Akai format</p> <p>'Rol' means Roland format</p> <p>Ensoniq format disks are not currently supported in Macros.</p>

## Using the Bank and Mode Fields

The bank and mode fields will only be used if a Macro file is loaded "as specified". (This is one of the options when loading in a .MAC file) This means that each file listed in the Macro will be loaded in exactly as the fields of the Macro entry are specified. These fields can optionally be overridden in the Load dialog, and a new bank and mode can be chosen for the entire Macro.

It is not always important what the bank and mode are set to in a Macro entry, since the Macro can really be loaded anywhere by overriding the bank and mode in the Load dialog. These fields are more important when you want to automatically configure the banks of your K2vx the same way each time you load a Macro, with different files being explicitly loaded into different banks.

## Viewing the Object List for a Macro Entry

If a Macro entry contains an object list, it can be examined by scrolling the Macro list display until the item with the **Obj** indicator is highlighted, and then by pressing either the Left or Right cursor button on the front panel. You will see a display that looks like this:



```
Macro Object List (load dependents)
Program 210
Program 211
Program 212
Program 213
```

```
OK
```

### Left or Right cursor button -> Displays Macro entry object list

The "Macro Object List" display, a scrollable list, shows what objects are to be loaded from the file specified in the currently indexed Macro entry. You will not see the names of the objects in this display, because they are not stored in the Macro Table. The objects are referenced only by object type and object ID. The "(load dependents)" indicator means that the Macro process should also load all dependents of the objects in this list.

If you need to know the names of objects in a Macro entry object list, it is possible to begin a disk function (such as Load), find the file specified in the Macro entry, press Open to display the file's objects, and look up what the objects are, using the information in this display.

## Unspecified Disk Drive ID

Before the time a Macro file is saved to disk, you can specify that a file's location will not be tied to a specific SCSI ID number. Instead, the file should be loaded from the same drive that contains the Macro file itself. This is called the "Unspecified" drive ID, since the actual ID is not known until the time of loading the Macro.

The Unspecified drive ID will typically be used by people who distribute K2vx sound files and Macro files on removable disk media such as **CD-ROMs** or **Magneto-Optical Cartridges**. This feature makes it possible to avoid specifying the exact SCSI ID used in a Macro file that is to be loaded on someone else's disk drive, since there is no way to predict the SCSI ID of the drive in advance.

Another use would be, if you are consistently working with Macro files that load mostly files from the same drive, then if you set the Macro's drive IDs to Unspecified you won't have to edit your Macro files if you happen to change the drive's SCSI ID.

Setting Macro entries to have an Unspecified drive ID is done from the Modify Macro page (press **DISK->Macro->Modify**), described later in this section.

## The Library Disk

If a Macro entry is set to the "Library" drive ID, it means that the file to be loaded should be found on the drive at the SCSI ID set by the **Library parameter** on the Disk Mode page. This designation is similar in purpose to the Unspecified drive ID, because it is a way to avoid needing to hard-wire the SCSI ID in advance. The Library drive ID is intended to be used in Macro files that reside on removable media such as a Floppy disk, whose Macro entries are supposed to load in particular files on a specific sample CD.

The main purpose of this feature is so that Macro files can be distributed on a floppy disks containing programs that link up with sample files from CD-ROMs. Formerly, a user would have to copy the CD-ROM file to their own hard drive if they wanted to make any new programs that use a CD-ROM file's sample and keymap data.

There are a lot of possibilities in the K2vx to come up with new programs for using with many available sample CDs. Now it is possible to distribute programs or sequences based on a floppy to other K2vx users who already have a certain sample CD in their collection. They

should be able to load the correct sequences of files just by loading a Macro, without those nasty SCSI ID conflicts.

A macro of this type would first load the CD-ROM files containing samples and keymaps (off the "Library" disk), and then load in files from the Floppy disk . The program files from the floppy should correctly link up to the pre-loaded keymaps and samples via the Relink-by-Name mechanism.

In order for the files in the Macro with a Library drive ID to be found, the user of the Macro must have already set up their Library parameter to be the SCSI ID of their CD-ROM drive.

In the following picture, notice that the Library parameter has been set to SCSI 3.

```
DiskMode Samples:10022K Memory: 42K
Path = \
(Macro on)
CurrentDisk:Floppy Startup:Off
Library:SCSI 3

<more Load Save Macro Delete more>
```

Assuming SCSI 3 is the user's CD-ROM drive, the user is now prepared to load a Macro file from on a floppy disk that contains the following entries:

```
L:\PIANO4MB.KRZ 200:F:Obj
F:\PNOPROGS.KRZ 200:F:
```

When the above Macro entries are loaded, the file \PIANO4MB.KRZ will be looked for on the drive with address SCSI 3, since this is what the Library parameter on the Disk Mode page has been set to. If the correct CD-ROM has been placed in the drive, the K2vx will load the file, and then continue and load \PNOPROGS.KRZ from the floppy disk.

If the K2vx executes the above Macro, but the Library parameter has not been set (is set to Off) an error message

### Library has not been set

will be displayed when the Macro process attempts to load \PIANO4MB.KRZ.

The above example is a typical case where the Relink-by-Name feature is used to have one file containing programs depend on another file containing samples and keymaps, yet the files are on different drives. The CD-ROM in the example might just as well have been an Akai or Roland format, and the file from the floppy would contain the latest custom K2vx programs making good use of the imported data.

The Library parameter is remembered by the K2vx across power-cycles, via the battery backed memory. This means that if you use this feature to load Macros you only have to set it up once.

Setting Macro entries to have a Library drive ID is done from the Modify Macro page (press **DISK->Macro->Modify**), described later in this section.

Although the Unspecified and Library drive IDs are meant to be used with distributable media such as CD-ROMs and floppy disks, these features will work with any supported disk drives.

## Loading Selected Entries from a Macro File

It is possible to examine the contents of a Macro file from any disk function page - the same way you would open a .KRZ file to check out what objects are stored in it - by highlighting the .MAC file and pressing **Open**:

```
Dir:\ANALOG\      Sel:4/5      Index: 5
                RESONANT .KRZ    109K
                SYNAPSE  .KRZ    421K
File to load:SYNTH .MAC      .5K
```

```
Total:664K
Select Root Parent Open OK Cancel
```

The K2vx will need to read the Macro file into a temporary area of internal memory, which means there needs to be enough free RAM to accommodate it. When Open is pressed from the Load function, a dialog similar to the Macro page and the Save Macro page will be entered, called the **Load Macro page**:

```
Func:LOAD MACRO  Sel:2/4      Index: 1

5:\ANALOG\MULTIUX.KRZ      200:F:
*5:\ANALOG\NOISE.KRZ      200:F:
5:\ANALOG\RESONANT.KRZ    200:F:
*5:\ANALOG\SYNAPSE.KRZ    200:F:
Select Check All OK Exit
```

From the Load Macro page, you can select one or more individual Macro entries for loading, instead of having to load the entire Macro. This is done using the **Select** soft button, identical to the method of saving Macro entries. In fact, this dialog operates identically to the Save Macro dialog, with one exception, the **Check** soft button.

The Check button will cross-check all of the Macro entries in this opened Macro file against the currently in-memory Macro Table, if there is one. Any entries in the opened Macro file that are not in the in-memory Macro Table will be selected when you press the Check button. The selected Macro entries can then be loaded in by pressing OK. This can be helpful to avoid loading in duplicate files if you...

- a) use Macro Record to keep a running history of files that you have already loaded into the K2vx, and
- b) have a lot of Macro files that load in a similar lists of files.

If the Open button is pressed from a disk function other than Load, you will see the View Macro page:

```
Func:VIEW MACRO Sel:0/4 Index: 1
5:\ANALOG\MULTIVOX.KRZ 200:F:
5:\ANALOG\NOISE.KRZ 200:F:
5:\ANALOG\RESONANT.KRZ 200:F:
5:\ANALOG\SYNAPSE.KRZ 200:F:
Exit
```

The only function of this dialog is to view the Macro list entries stored in a Macro file. This feature is useful when, for instance, you are about to delete a Macro file and want to know what information is contained in the file before you remove it from the drive.

### Editing Macros

The in-memory Macro Table can be edited from the Macro page. You can select one or more Macro entries and do any of the following operations to them:

1. Change the Drive ID.
2. Change the Bank and Mode settings.
3. Delete the selected Macro entries.

To edit a Macro file already saved on your disk, it is necessary to first load the Macro file into the in-memory Macro. If you recall, this is done by the following procedure:

Making sure that Macro Record mode is enabled, go to the Load function, highlight the Macro file you wish to edit, and then either select certain entries from the Macro file (by pressing Open to get to the Load Macro page), or just press OK to load the entire Macro file. When you see the following display, press the Macro soft button, so that the K2vx will not try to load the files listed in the Macro.

```
Load this macro as:specified
300...399
400...499
500...599
Insert Macro OK Cancel
```

If you want, you can rebank the Macro on the way in by scrolling the bank list to something other than "specified". Similarly, if you override the load mode, it will be reflected in the in-memory Macro Table as well.

To edit entries from the Macro Table, return to the Macro page. For this example we will edit all the entries at once (like other similar dialogs, if you only are concerned with one list entry, it does not need to be explicitly selected with the Select soft button.) You can use the following double-presses to select and deselect all entries in the list:

- Left/Right cursor double-press -> Select All Macro Entries
- Up/Down cursor double-press -> Clear All Selections

With all the entries selected, our display looks like this:

```

Func:MACRO [Record]      Index:    1
*4:\NEWMIX\TRASHX12.KRZ  200:F:
*4:\POTS\TEHPOT.KRZ     300:F:
*4:\PANS\FRYING.KRZ     400:F:
*4:\KITCHEN\SINK.KRZ    500:F:Obj
Select Modify Load Pause Off Exit

```

Two soft buttons have yet to be explained. They are **Modify** and **Load**.

Press the **Modify** soft button to change any items mentioned at the top of this section. You will see the following display:

```

Modify Macro Entries
Modify:Drive
Drive :SCSI 4
4 entries selected.
Delete OK Cancel

```

This shows the screen that allows you to change the SCSI drive for the selected Macro entries. If you increment the **Modify:** parameter, the display switches to let you modify Bank and Mode information:

```

Modify Macro Entries
Modify:Bank/Mode
Bank :200's      Mode:Fill
4 entries selected.
Delete OK Cancel

```

The initial settings of the parameters on these screens are always taken from the lowest indexed Macro entry that is selected on the Macro page. In addition, every time you return to the "Modify Macro Entries" page, both the **Modify** parameter and the cursored parameter field will be the same.

Pressing **OK** will set all of the selected Macro entries to have drive ID or Bank and Mode settings according to the parameters set up on this page. The display will return to the Macro page with the same entries still selected. Any modifications to the parameters will be visibly apparent. Selecting multiple entries for editing allows you to change those entries in a uniform way. In our example above, you could change the Macro so that all the files were loaded into a single bank, instead of the separate banks they had previously been loaded to.

Press **Delete** to remove the Macro entries from the Macro list. You will see the display:

## Delete macro entries?

**Yes** **No**

If you answer **Yes**, the display returns to the Macro page and all of the previously selected entries will be gone from the list. If you answer **No**, the display will return to the Modify Macro Entries dialog.

Pressing **Cancel** in the Modify Macro Entries dialog will return to the Macro page with everything that was selected still selected, but with no parameter changes made to any Macro entries.

Here are the parameter values for Modify Macro Entries:

Parameter	Values
Modify	Drive, Bank/Mode
Drive	Floppy, SCSI 0, SCSI 1, SCSI 2, SCSI 3, SCSI 4, SCSI 5, SCSI 6, SCSI 7, Unspecified, Library
Bank	000's, 100's, 200's, 300's, 400's, 500's, 600's, 700's, 800's, 900's, Everything
Mode	Append, Merge, Fill, Overwrite, OvFill

Once you have made the necessary changes to the in-memory Macro, you can go to the Save Macro page to write selected (or all) entries to a new disk file (or replace over the original Macro file you may have loaded in for editing.)

### Executing the In-Memory Macro

You can load any group of files listed in the in-memory Macro. This is done using the **Load** soft button on the Macro page:

```

Func: MHRU [Record] Index: 1
1:\STRINGS.KRZ 200:F:
*1:\PIANO.KRZ 300:F:
 1:\DRUMS.KRZ 400:F:
*1:\NOISE.AIF 500:F:
Select Modify Load Pause Off Exit
    
```

Pressing the **Load** button gives you the choice of loading either **all of the files** in the Macro list or loading **only the files that are selected**:

```
Load selected items or all items?
```

```
All Selected Cancel
```

If you don't have any items explicitly selected (with an asterisk), the message you see when you press load is slightly different:

```
Load current item or all items?
```

This word "current" means the highlighted entry in the display will be loaded.

The files that have been selected for loading will be loaded in their respective order in the Macro list, using the bank and mode parameters that are specified in the list. In the above screen, if you were to load selected items, first \PIANO.KRZ would be loaded in the 300's bank, and then \NOISE.AIF would be loaded in the 500's bank.

## Macro Insert

It is possible to insert new Macro entries into the middle of the Macro Table if necessary. This is done by pressing the **Insert** soft button at the "Load this file as:" prompt, when loading in a file (if Macro Record is enabled):

```
Load this file as: 200...299
                  300...399
                  400...499*
                  500...599*
Insert            Macro OK Cancel
```

When you press Insert, you will see a dialog displaying the current in-memory Macro list:

```
Set Macro Insert Point      Index: 2
5:\ARCEILE.KRZ              200:F:
5:\METALIC.KRZ              200:F:
5:\STEREOZX.KRZ             300:F:0b3
5:\STRINGS\DBLBASS.KRZ      400:F:
5:\STRINGS\CELLOS.KRZ       400:F:
                          OK Cancel
```

Scroll the Macro list until the entry before which you want to insert is highlighted. In the above display, any new macro entries added by this load operation will be inserted in the Macro list just before the entry for METALIC.KRZ.

Press **OK** to enable the insert point. Press **Cancel** to disable the insert point.

The display will return to the "Load this file as:" dialog. If a Macro insert point has been set, an indicator will appear at the top left of the screen:

(Macro insert)

```
Load this file as: 200...299
                   300...399
                   400...499*
                   500...599*
Insert             Macro OK Cancel
```

The insert point can be disabled before loading, by pressing Insert again and pressing Cancel from within the "Set Macro Insert Point" dialog. The "(Macro insert)" indicator will disappear. Similarly, the insert point can be changed before loading by pressing Insert again (the screen will highlight the current insert point,) scrolling to a different insert point, and pressing OK.

### Saving and Loading a Macro Table in a .KRZ file

Macro Table objects can be explicitly saved or loaded (without being "executed") using Save Individual Object and Load Individual Object. If for some reason you wanted to save a Macro Table that was being worked on "in-progress", and then be able to load it back in later on to be worked on some more, you would use this method. The Macro Table can be selected for saving and loading just like any other object. When you load in a Macro Table using Load Individual Object, it will overwrite any Macro Table already in memory. Once it is loaded in, you may have to go to the Macro page and enable Macro Record mode to continue to record further load operations into the Macro.

The "Save Everything" feature of the Save dialog will not include the Macro Table. This is done to prevent inadvertent distribution of what would most likely be a meaningless Macro Table to other users.

### Aborting a Macro Load

There is a way to abort the process of loading a Macro file. Aborting can only be done "in between" files that are being loaded, and not during the load of any one file (short of powering off or soft-resetting the machine with [simultaneously pressing +/- 0 CLR]...but this is not recommended!).

Aborting a Macro load is done by pressing and holding down either of the Plus (+) or the Minus (-) buttons that are located just below the Alpha wheel. This should be done at least one-half second before you anticipate the current file to finish loading, or else the K2vx will not sense that you wish to abort the load.

You will see the following question after the current file being loaded is completed:

Abort the macro?

Yes No



It may be a good idea to practice using this method of aborting a macro file load, so that when the time comes that you load in an out of date Macro with 25 entries all at the wrong SCSI ID. you will remember how to abort the process. This same method (of holding the Plus or Minus buttons down) is also used to abort the Backup feature and the multiple file load feature.

If you run out of object or sample RAM, you will have one opportunity to abort the Macro as explained above. However, if you continue from that point you may end up seeing the same error message "Memory is full" for each file that you had selected. This can be a rather tedious process, however it is still possible to abort out of this by holding down the Plus or Minus button simultaneously while pressing "Yes" when you see the following question:

Abort this Partial load?

Yes No

If the Macro that you abort was loaded by multiple selection together with other files (that may or may not be Macro files), you will have to "abort twice", once to get out of the current Macro file load, and a second time to get out of the multiple file load process.

If you are aborting a Macro because the Disk Drive ID is incorrectly specified (as evidenced by lots of "Problem mounting disk" errors) you will need to hold down either the Plus or the Minus button simultaneously while pressing OK to satisfy the error prompt. The screen may "blink" while holding down the repeating Plus or Minus button, but as soon as you have pressed OK you will see the "Abort the macro?" question.

### Disk Utilities

The Disk Utility functions provide certain necessary information about disks and their files and directories. These functions are useful when you want to know how your disks are organized and how much disk space you have available. They also help you to locate files and directories.

To access the Disk Utilities page, press the Util button from the Disk Mode page. The Disk Utilities page looks like this:

DiskUtil: SCSI 2

Select utility function:

Find List Free Done

The functions on the Disk Utilities page are used for finding out information about the Current Disk. The Current Disk is always indicated on the top line of this page. If you want to use the utility functions for a different disk, you must first set it to be the Current Disk on the Disk Mode page.

Here is a brief description of each soft button:

- Find** Find files utility. Allows you to search for files that match a certain character string in their file names.
- List** Lists an expanded directory tree from any level of the hierarchy. This function can be used to determine the total size of files within any tree of subdirectory. It is also helpful for finding files on the disk.
- Free** Calculates the available space on the disk.
- Done** Exits back the Disk Mode page.

## Find Files

The Find files utility first prompts you to enter a character string to be searched for:

```
Search string: RAT_
```

```
Delete Insert >>End Choose OK Cancel
```

You can use the **Choose** button to grab the text of a file name from the current disk , as described previously.

If you press **OK**, the K2vx will begin to search the disk for any files or directories that contain the search string in their names. The search starts in the root directory and searches the entire disk. When a matching file or directory is found, you'll see one of the following:

```
Found file:
```

```
\BABYTOYS\RATTLE.KRZ
```

```
FindNext Go To Cancel
```

```
Found directory:
```

```
\CRATES
```

```
FindNext Go To Cancel
```

If the search string is found anywhere within a file name it will be matched. The search algorithm independently checks both the file name and the extension. For instance, if you wanted to find any Macro file on the disk, you could enter in "MAC" for the search string. This would find any Macro files as well as any files or directories that have "MAC" in their filename.

When a match is found, there are three choices displayed:

**FindNext** Continue searching the disk for another file or directory that matches the search string.

**Go To** Exit to the Disk Mode page, setting the current directory and file index of the K2vx to the location of the found file or directory.

The next disk function you choose will display the current directory with the found file already highlighted. If a directory was found, then the first file in the directory list will be highlighted.

**Cancel** Exit to the Disk Mode page.

When the search has checked all of the items on the disk, the K2vx displays this screen:

```
Search completed.
```

```
OK
```

If no matching files were found, you will also see

```
(No files found)
```

The K2vx will remember the last search string that you entered. This makes it easy to repeat the same search. If you press **Util->Find** again, the "Search string:" dialog will contain the previously used string.

## List

The List utility allows you to view directories on a disk with the expanded contents of all sub-directories. Each line is indented according to its directory level, so that you can get sense of how your directory tree is organized.

```
Dir:\ [ 1968K] Index: 1
ELEPHANT.KRZ 148K
BSOUNDS\
  TRUMPETS\
    JSBACH.KRZ 712K
Root Up Down Go To Exit
```

The **Dir:** parameter shows the directory that is being listed. The size value displayed on the top line of the display represents the total size of the directory subtree that is currently being viewed. The **Root**, **Up**, and **Down** soft buttons control going from one directory to another in the hierarchy:

## Disk Mode

### Disk Utilities

---

- Root**      Display the disk from the top level, meaning that all of the files on the disk will appear in the scrollable list.
- Down**      Set the display to the level of the highlighted file or directory.  
For example, scrolling to JSBACH.KRZ in the above list, and pressing Down , would focus the list on the contents of the TRUMPETS directory.
- Up**         Set the display up one directory level.
- Go To**      Exit to the Disk Mode page, setting the current directory and file index to the location of the highlighted file or directory.
- Cancel**      Exit back to the Disk Mode page. The current directory is unchanged.

The files are listed in the order that they appear on the disk, unalphabetized. The traversal order of the directories is the same one that is also used for the Backup function.

## Free

The Free utility calculates the amount of free space on the current disk and displays the result on the Disk Utilities page. This may take a small amount of time to calculate, depending on the drive.

```
Computing free space on SCSI 2
```

```
Please wait...
```

```
Disk Utility SCSI 2 (DOS)  
Used: 23%    Free: 94814K    Total: 124396K
```

```
Select utility function:
```

```
Find   List   Free   Done
```

The parameters tell you the following:

**Used**      The percentage of the disk that is taken up by the existing information stored on it.

**Free**      The amount of disk space available for new files.

**Total**     The size of the disk. This will be the size of the usable partition if MS-DOS format.

For SCSI drives, if the current disk was formatted on a PC or a Mac in MS-DOS format and contains at least one primary partition, you will see the "(DOS)" indicator on the top line. Using this format is described in the section on "MS-DOS compatibility".

For floppy drives, the density (HD or DD) will be displayed. All K2vx floppy disks are MS-DOS format compatible.

## Moving Files Between Directories

Files and directories can be moved from one directory to another on a given disk. You can either choose multiple files to move using the Select soft button, or just move the single highlighted file or directory (if no other files are selected). As you would probably expect, moving a directory also moves all the files within the directory. To use this function, press the **Move** soft button from the Disk Mode page. Then choose the file or files that you want to move:

```
Dir:\                               Sel:0/15   Index:    1

File to move:ATOMTOM .KRZ           98K
              BLOWFISH.KRZ          36K
              COLORS .KRZ           109K
Total:6846K   DRUMHITS (dir)
Select Root Parent Open OK Cancel
```

Press **OK** when you have made your selection. Press **Cancel** to return to the Disk Mode page.

The K2vx remembers the most recent destination directory that a file was moved to. If the current directory is different from the most recent destination directory, you will see the question:

```
Use default directory for ATOMTOM.KRZ?
(Path = \DRUMHITS\)
```

```
Change OK Cancel
```

Press **OK** to use the default.

Press **Change** if you want to select a different destination from the default shown. The K2vx will then display a directory selection dialog (described early in this chapter), and you can select the move destination directory from there:

```
Dir:\                               Sel:0/15   Index:    4
              BLOWFISH.KRZ          36K
              COLORS .KRZ           109K
Select dest dir:DRUMHITS (dir)
              EARTHLNG.KRZ          144K
              HANDCLAP.KRZ          645K
Total:6846K   INDUSTRY (dir)
Root Parent Open Current SetDir Exit
```

The move operation begins when you press either **Current** or **SetDir**.

If the default destination directory is the same as the source directory, the K2vx will skip the "Use default directory?" question and instead go right to the "Select dest dir:" dialog.

A good way to organize files into subdirectories, with a disk drive that was being used with previous versions of K2vx software, is by using the Move command. First, create the subdirectories you need, using the **NewDir** function. Then, use multiple file selection to select the files to be grouped into a particular subdirectory. The files can be moved to their new location in one operation.

For each file that is moved, you will see a confirmation message:

```
\ATOMTOM.KRZ moved to  
\DRUMHITS\ATOMTOM.KRZ
```

Note the following:

- You can select multiple files for moving within a directory. However, you cannot move files from more than one directory at a time. If you select any files and then switch to another directory, the files you had chosen will be deselected.
- If you are moving a directory, you cannot move it in to a subdirectory of itself.
- If the source and destination directories are the same, the file will not be moved, and an error message such as the one below is displayed. This would happen if you pressed **Current** above.
- The same message will be displayed if there is a file in the destination directory with the same name as the file to be moved.

```
File \ATOMTOM.KRZ not moved.
```

## Renaming Files

Press the **Rename** soft button (from Disk Mode) to rename files or directories without loading them. When you press **Rename**, the K2vx will prompt you to select the file to be changed, by showing you a list of the files found on the current disk.

When you've selected the file to be renamed, press **OK**, and the K2vx will ask you to enter the new file name. When you've done this, press **OK**, and the file name will be changed.

```
Dir:\          Sel:0/1      Index: 1
```

```
File to rename:FILE9876 .KRZ 2048K
```

```
Total:2048K  
  Root  Parent  Open  OK  Cancel
```

This function can only be used to change the 8-character file name and not the extension. When you press **Rename**, the File List dialog is displayed and you can navigate through the directories to choose the file or directory you wish to rename. Unlike the other disk functions that use the File List dialog, you will not see the **Select** soft button. This is because you can only rename one file at a time. Therefore you simply choose the file you want and press **OK**. The K2vx will then ask you to enter a new name. Once you've done this, press **OK** again, and the file name will be changed.

## Deleting Files and Directories

Press the **Delete** soft button (from Disk Mode) to delete files and directories. The Delete function supports multiple selection of files for deletion. Select the file(s) and/or directory(s) to be deleted, and press **OK** (or **Cancel** to abort). Be careful! You don't get a second chance to change your mind once you've pressed **OK**. Once a file is deleted, it's gone. Remember the fundamental directive of computer users: Save early, save often; make backups.

```
Dir:\                               Sel:2/4           Index: 2
                                     BLUES           (dir)
File to delete: MOTOR .KRZ*         98K
                                     QUACKS         .KRZ           344K
                                     ZAPPER        .KRZ*         802K
Total:1244K
Select Root Parent Open OK Cancel
```

A directory must be empty in order to delete it (i.e. it must not contain any files or subdirectories.)

## Backup and Copy Functions

### File Backup

To access the Backup function from the Disk Mode page, first make sure that the Current Disk is set to be the drive that you want to make a backup of. Next, press the **Backup** soft button:

```
Dir:\                               Sel:0/15         Index: 1
Set backup dir: ANIMALS .KRZ       1097K
                                     BREAKAGE      (dir)
Total:9040K                          LOWINST      (dir)
Root Parent Open Current Exit        PLANKTON     (dir)
```

On this screen, choose the directory tree that you want to backup by selecting a directory (see *Selecting Directories*, near the beginning of this chapter.) Backup allows to you copy all of the files within a directory tree from one disk to another. All of the files within the directory that you choose, plus all of its subdirectories and files within them will be copied to the new drive. If you want to backup the entire disk, then make sure the current directory is the root directory (as in the picture), and press **Current**.

Next, you will see a dialog for choosing the destination disk:

```
Destination disk:SCSI 0
                  SCSI 1
                  SCSI 2
                  SCSI 3
                  [OK] [Cancel]
```

Select the disk you wish to transfer files to. It must be a different disk than the current disk.

Next, you can select a directory on the destination disk that will receive the transferred files.

```
Use default directory on SCSI 0?
(Path = \)
```

```
[Change] [OK] [Cancel]
```

The default is always the root directory on the destination disk. Press **OK** to select the default. To select a different directory, press **Change**.

Next, select the **Backup mode** when you see the following question:

```
RePlace or increment mode?
```

```
[Help] [RePlace] [Incr] [Cancel]
```

**Replace** Any files to be transferred that already exist in the destination directory will be replaced (overwritten.)

**Incr** Any files to be transferred that already exist in the destination directory will be skipped (not transferred.)

**Help** Displays a reminder about the meaning of Replace and Increment modes.

After you select the Backup mode, you will see a confirmation screen with all of your selections so far:

```
Press [OK] to start backup:SCSI 4->SCSI 0
Mode =RePlace
Source=\
Dest =\
```

```
[SetFile] [OK] [Cancel]
```

**OK** Begin the Backup function according to the parameters on this page.



- Cancel** Exit to Disk Mode.
- SetFile** Set Backup starting file. This is mainly used when you are backing up files from a larger SCSI drive to several smaller removable media disks. Once one of the removable destination media is full, note the last file that was successfully copied. Enter the Backup function again, setting up the source and destination path parameters exactly as before. But this time use the **SetFile** button to start from the file after the last file copied.

You will notice that the file list displayed when SetFile is pressed is not in alphabetical order, but in the actual order the files occur in the directory on the disk:

```
Dir:\                               Sel:0/15   Index:    1

Set start file:ZVXWUUS .KRZ   418K
                PLANKTON      (dir)
                ANIMALS      .KRZ   1097K
Total:9040K      VEGGIES      .KRZ   2801K
                Root  Parent  UPen  OK  Cancel
```

Find the last file copied from the previous partial backup. Set the list index to one entry past the location of this file. If this file is already the last entry in the file list, the "next" file to continue the multi-part backup would be the file or directory that comes after the file list entry for the currently viewed directory. To find this file or directory, you will need to look in the Parent directory.

A large SCSI drive backed up in this way can be restored by individually backing up the removable media in the reverse direction.

Backup and Copy will transfer files much faster if you have cleared your object RAM first using the Delete Everything command (save any objects in RAM to disk first.)

When using the Backup function to transfer files from SCSI to Floppy, directories are ignored (i.e. not created on the floppy.) This "flattens" out any directory structure so that multi-disk files can be created on the floppy disks.

## File Copy

The Copy function lets you copy a single file from a directory on one disk to a specified directory on the destination disk.

## Creating a Startup File

You can create a Macro file that will automatically be loaded in when you power-on your K2vx. This file, called the **Startup file**, or **Boot Macro**, can be on a disk drive at any SCSI ID or else on the Floppy disk. See the section on Macros for background information.

The steps needed to create a Startup file are:

First, create a Macro file called **BOOT.MAC** in the root directory of the SCSI or Floppy disk that you will use as the Startup disk. Specify in the Macro the exact ordering of files that you would like to have loaded into the K2vx when powered on. When you save the Macro file, just name the file "BOOT", and the K2vx will add the ".MAC" extension.

Second, set the **Startup** parameter on the Disk Mode page to be the drive ID of the Startup disk. So, if your BOOT.MAC file was on a floppy disk, set the Startup parameter to **Floppy** and make sure that you have the correct floppy in the drive when you next turn the K2vx on.

When the K2vx is powered on, it will display the following message (after the introductory VAST logo):

```
About to load startup file...
```

```
Cancel
```

The K2vx looks for a file BOOT.MAC in the **root directory** on the drive specified by the Startup parameter. If the file is not found, or the drive cannot be accessed, you will get an error message. The Startup load can be bypassed in the first few seconds after the K2vx is turned on, by pressing the **Cancel** button that is displayed on the screen.

If BOOT.MAC is found by the K2vx, it will begin to load the Macro file as if you had loaded it explicitly from the Load function in Disk Mode. When the Macro has completed, you will see the following:

```
Macro BOOT.MAC completed...
```

The K2vx will go directly to Program Mode afterwards.

### Deleting Banks in a Startup File

You may want the Startup file to clear out one or all banks in the K2vx before loading files. This could help overcome the problem of having "silent" copies of programs in your RAM that depend on samples that are no longer there (because they disappeared the last time the K2vx's power was turned off.) The following trick will allow a Macro entry to essentially function as a Delete Bank or Delete Everything command:

Create a file somewhere on (preferably) your Startup disk, by saving an empty bank from the K2vx. Call the file NULL.KRZ. Now, insert this file at the beginning of a boot Macro you are creating - load the file, specify the bank you want to delete in the Startup file (choose 'Everything' to clear out all of the object RAM), and specify **Overwrite** for the load mode. Make sure you press **Macro** and not **OK**, so that the overwrite doesn't take place until you use the Startup file.

Here is what that Macro entry for this file might look like on the Macro page, if you were doing a "Delete Everything"

```
3: \NULL.KRZ
```

```
E:O:
```

The "E:O" stands for "Load as Everything, using Overwrite mode".

## MS-DOS File System Compatibility

The K2vx is compatible with fixed and removable disk drives that use the MS-DOS hard disk and floppy disk formats. If you want to use this feature, **you must first format the disk media on a computer** such as a PC compatible or a Mac running appropriate MS-DOS conversion software.

The MS-DOS hard disk format is structured so that the disk can be split up into multiple partitions. **The K2vx can only use the first partition that it finds on the disk.** Therefore, it is usually best to format the media with only one partition taking up all usable space on the disk. Working from the K2vx front-panel with an MS-DOS formatted drive will appear the same as working with a drive that has been formatted with the K2vx's own Format function (on the Disk Mode page.) The Free utility (Disk Mode->Util->Free) can be used to identify whether a disk is DOS format or standard K2vx format. If the "DOS" indicator is displayed, it means the K2vx has determined that the disk is a DOS format hard drive with at least one primary DOS partition. Floppy disks will not display the DOS message.

Some advantages of working with an MS-DOS compatible drive format over the standard K2vx format are:

- easier sharing of K2vx files with other users over computer communications lines,
- being able to use graphical file management interfaces for organizing files and directories,
- being able to back up K2vx data using a PC compatible or Mac with commercially available software, and
- easier transfer of data using standard file formats such as AIFF, WAVE, and MIDI Type 0, for importing and exporting samples and sequences.

### File Name Compatibility

**DOS format does not support space characters in file names.** The K2vx, though, allows spaces to be used within file names. If you plan to transfer files between the K2vx and a DOS compatible computer, it is recommended that you use only file names without space characters in them. Otherwise, a computer may have trouble identifying the files.

## Importing and Exporting Data using Standard File Formats

The K2vx supports three common data interchange file formats, Apple Interchange File Format (AIFF), Microsoft RIFF WAVE, and MIDI Type 0. The first two are used to transfer sample data, and the latter is used for sequences.

The K2vx can recognize these file types automatically on loading, regardless of the file extension. You can load these files as you would any standard K2vx file, and also as part of a Macro file load. The most recent sample file loaded will become the "preview" sample, which means you can quickly access it for playing or editing on the Sample page (**MASTER->Sample**). Similarly, the most recent MIDI Type 0 file loaded in will become the current song on the Song Mode page.

You can save files in these formats on the **Export** page. This page is accessible from the Disk Mode page by pressing **Save->Export**.

```

Save AIFF file                               Index: 1

Sample 200 NewSample                          C 4S 100K
Sample 201 Babaganoush                       C 4S 444K
Sample 202 YaYaYa                            C 4S 190K
Sample 203 GregorianChant                    C 4S 6515K
AIFF    WAVE    MIDI                          OK    Cancel

```

The Export page allows you to save one sample or song object per file. Choose the format you wish to save in, and press the corresponding soft button. For AIFF and WAVE, only sample objects are listed. For MIDI Type 0, only song objects are listed. Scroll to the object that you wish to save, and press **OK**. The dialog will proceed the same as if you were saving a K2vx file. You will be prompted for a file name, and will have the option to select a different default directory to save the file in.

The K2vx will automatically place a standard extension on the file when it is created on the disk. These extensions are sometimes necessary when transferring files to external programs that can recognize the file format based on the extension. They also help you to recognize what files are of a what format when looking at a directory listing. You can use the Find Files utility (Disk Mode->Util->Find) to search for files that match a certain extension. The standard extensions used on the files are:

```

.AIF    AIFF
.WAV    WAVE
.MID    MIDI Type 0

```

The first time you enter the **Export** page after powering on (or after a soft-reset of the K2vx), the format defaults to AIFF. After that, the K2vx remembers the most recent format that you used. For example if you save a MIDI Type 0 file, and then go back to the Song Mode page to record more sequences, the next time you return to the Export page, the file format will still be set to "MIDI", and all of the song objects will be listed. You can audition the samples and songs the same way as you would on the Save Object or Object Utility pages (by pressing the Left or the Right cursor buttons on the front-panel.).

## AIFF and AIFF-C Files

The K2vx can read 8 or 16 bit AIFF files, mono or stereo. The sample rate, sustain loop, loop mode, base note, sample name, and sample detuning are supported. **AIFF-C** files that do not use compression can also be read by the K2vx. The K2vx will save 16-bit AIFF files, either mono or stereo.

## WAVE Files

The K2vx can read 8 or 16 bit WAVE files, mono or stereo. WAVE files do not support looping. The K2vx will save 16-bit WAVE files, either mono or stereo.

## MIDI Type 0 Files

Use this format to transfer a song to and from an external sequencer. Tempo information is supported. Time Signature is not supported. System Exclusive data is not supported.

# Chapter 14

## DSP Functions

This chapter presents explanations of the DSP functions that can be inserted into the algorithms in the Program Editor. As you configure each algorithm, the DSP functions you select determine the type of synthesis you apply to your sounds. Deciding which algorithm to use depends on what you want to do; there's no hard and fast rule. If you want to create a classic analog sound, for example, you'll choose one of the algorithms containing one or more blocks that can have filter functions assigned to them. If you want realtime panning effects, choose an algorithm that includes the PANNER function in the F3 block. Your best approach is to study the algorithm charts in the *Reference Guide*, and choose the algorithm that includes the functions you want to work with.

Before we get to the explanations of the DSP functions, we've included a brief discussion of a few general concepts of sound synthesis. This should help you understand the workings of the DSP functions. We'll refer to these concepts repeatedly as we go along.

Any single sound waveform is composed of numerous sine wave components, each at a different frequency. These components are called partials. The lowest frequency is perceived by the ear as the pitch of the sound, and is called the fundamental. The other components are called harmonics. The relative amplitudes (volume) of each of the partials in a sound determine its timbre, its most recognizable characteristic. When you think of the difference between the sound of a piano and a saxophone, you're thinking about their different timbres. A dull sound has a strong fundamental and weak harmonics, while a bright sound has strong harmonics.

Sound synthesis can be most simply described as the manipulation of either the amplitude or phase of one or more of the partials constituting a sound. The K2vx's various DSP functions give you a variety of methods for manipulating those partials. We've grouped our explanations of the DSP functions according to the types of specialized manipulation they enable you to perform on a given sound. The categories are:

- FILTERS
- EQUALIZATION (EQ)
- PITCH / AMPLITUDE / PAN POSITION
- MIXERS
- WAVEFORMS
- ADDED WAVEFORMS
- NON-LINEAR FUNCTIONS
- WAVEFORMS WITH NON-LINEAR INPUTS
- MIXERS WITH NON-LINEAR INPUTS
- SYNCHRONIZING (HARD SYNC) FUNCTIONS

## Introduction to Algorithm Programming

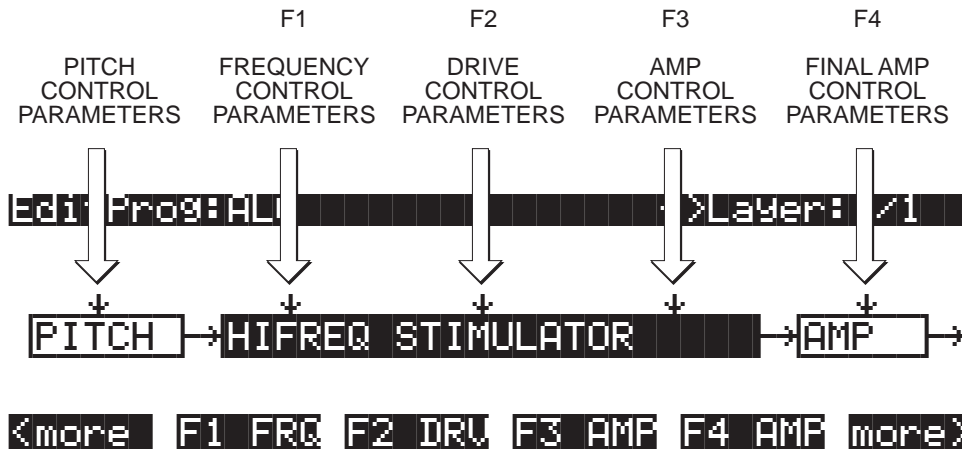
Programming the algorithms is a multi-step process. The first step is selecting an algorithm. Changing the algorithm of an existing program's layer is likely to alter the sound of the layer dramatically. As a rule, then, you won't want to change a layer's algorithm unless you're building a sound from scratch. Furthermore, when you change a layer's algorithm, the values for each of the DSP functions within the algorithm may be set at non-musical values; you should lower the K2vx's volume slider before changing algorithms.

Deciding which algorithm to use for a new sound is primarily a process of planning a layer's signal path through the sound engine. The real sound manipulation is done by the DSP functions you insert into the algorithm. The algorithm simply lays a framework that determines how the DSP functions interact.

Once you know which algorithm you're going to work with, you'll assign various DSP functions to each of the stages of the algorithm. These stages, as you recall, are represented by the rectangular blocks you see on the ALG page. The arrows pointing down at the blocks represent control inputs that affect the behavior of the DSP functions. For each arrow, there's a page of parameters controlling some aspect of the DSP function's behavior. Every DSP function has at least one control input; several have two or three.

The ALG page is where you select algorithms and assign DSP functions to the algorithm's various stages. To assign a DSP function, move the cursor to select the stage you want to modify, then use any data entry method to scroll through the list of available DSP functions for that stage. You'll normally hear the effect of each selection as soon as you make it. If you don't hear a difference, it's because the function's control parameters aren't set to significant values. Once you adjust some of these parameters, the function will have a noticeable effect on the sound. Keep in mind that not all DSP functions are available at every stage of every algorithm.

When you have each stage of the current algorithm set up to your liking, you can begin to program the control inputs of each DSP function. This is done by selecting the control input page(s) for the currently selected DSP function, and adjusting the parameters on the page. There are two ways to select the control input pages: you can move the cursor to select the DSP function you want to tweak, and press EDIT. The selected DSP function's control input page will appear (if it's a multi-stage DSP function, its first control input page will appear). Or you can use the soft buttons to select the pages. The PITCH soft button always selects the pitch control input page, since the first stage of every algorithm is invariably the pitch control. The F1—F4 soft buttons select the control input pages corresponding to the remaining four arrows, which point down at the subsequent four variable control inputs.



Each control input page contains several parameters, which affect some aspect of the behavior of the DSP function named on the top line of the page. Most of these parameters are the common DSP control parameters; for a review, see "Common DSP Control Parameters" in Chapter 6.

The possibilities are truly enormous, given the number of different combinations of functions you can assign to any particular layer (not to mention multi-layer programs, each layer of which has its own algorithm). You can create completely new sounds just by tweaking the parameters on the control input page for a single DSP function. When you begin adjusting these parameters, it's a good idea to start with all of them set to 0 (or the value that minimizes their effects), then adjust them one by one. This will help you understand exactly what effect each parameter has, and will give you an idea of the variety of effects each parameter can produce. Then you can start combining the effects of multiple parameters, and quite possibly take your sound in a whole new direction. You'll quickly become familiar with the control input pages for the DSP functions, since most of them contain the same parameters, with just a few variations. You'll find that they all look much alike. The top line of each page, however, will indicate which DSP control input it represents (PITCH, or F1–F4).

For example, on the page below, the top line tells you that the currently selected DSP function is the HIGH FREQUENCY STIMULATOR—its name is abbreviated and enclosed in parentheses. You can also see that you’re looking at F1, which in this case controls the frequency of the HIGH FREQUENCY STIMULATOR. So the top line of these pages always shows three things: 1) the currently selected control input (PITCH or F1–F4); 2) the aspect of the current DSP function controlled by the input (this varies depending on the current DSP function); 3) the currently selected DSP function (usually abbreviated, and in parentheses). Items 1) and 2) match the label of the soft buttons that select each page. The page below, for example, is selected with the soft button labeled “F1 FRQ.”

```

EditProg: F1 FRQ(HIGH FRQ STIM) <> LOWER 1/1
Coarse: C 4 262Hz Src1 : OFF
Fine : Oct Depth : Oct
KeyTrk: Oct/key Src2 : OFF
VelTrk: Oct DptCt1: OFF
Pad : 0dB MinDpt: Oct
MaxDpt: Oct
<more> F1 FRQ F2 DRU F3 AMP F4 AMP >more>
    
```

### Additional Parameters

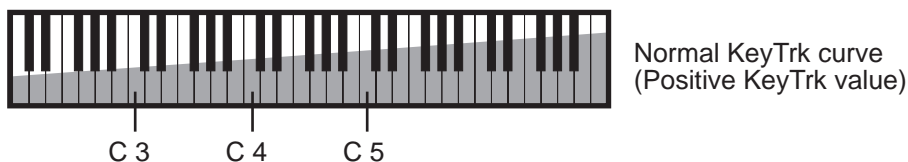
In addition to the common DSP control parameters you’ll find on each page, you’ll also see a few others on various pages. They’re described here, since programming them is the same regardless of the page on which they appear. Depending on the DSP function they affect, they’ll have different ranges of values and different units of measurement (% , dB, etc.).

#### Pad

Many of the DSP functions boost the signal as it passes through. Depending on the signal’s input level and the amount of gain (boost) introduced by any given DSP function, its output may clip, which will alter the sound considerably. Clipping may also occur as a result of phase shifting, but this is not as common as clipping caused by gain. While you may find clipping to be a useful component of some sounds, you’ll want to remove it from others. The Pad parameter, which appears on the control input pages of many DSP functions, lets you attenuate (reduce the amplitude of) the signal by 6, 12, or 18 dB at the input of those functions. Use the Pad parameter to reduce or eliminate any undesired clipping caused by the currently selected DSP function.

#### Key Track Start (KStart)

This parameter appears on many control input pages, and gives you added control over the effect of Key Tracking. For each note you play, it multiplies the value of the KeyTrk parameter by a number that varies with the note’s MIDI key number. If KeyTrk is set to 0, this parameter will have no effect. When KeyTrk is a non-zero value, KStart will modify the normal key tracking curve, which is shown in the diagram below. The effect of normal key tracking reaches its minimum at C -1, and its maximum at C 9. You can use KStart to dampen the effects of key tracking at one end of the keyboard. If key tracking causes a sound to clip or distort toward the high end of the keyboard, for example, you can use KStart to reduce the effect of the key tracking at the upper end without changing its effect on the lower end. To do this you would set a negative value for KeyTrk, and a unipolar value for KStart.

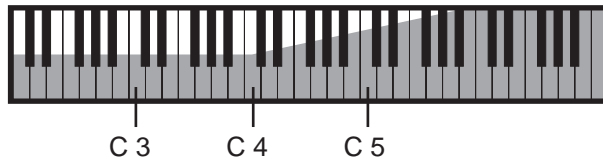


UNIPOLAR KEYSTART

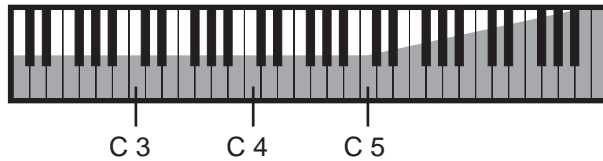
The range of values for KStart is C -1–C 9 unipolar, and C -1–C 9 bipolar. Unipolar and bipolar values have different effects on the key tracking. The next three diagrams illustrate the effect of three different *unipolar* keystart values on the key tracking curve when a positive value is assigned for the KeyTrk parameter. At a KStart value of C 4, for example, there is no key tracking effect below Middle C (it multiplies the key tracking amount by a key number value of 0). The key tracking value is multiplied by 0 at C 4 (normal key tracking), by 1 at C# 4, by 2 at D 4, and so on to a maximum of 64 at 5 1/3 octaves above the KStart value. For higher notes, the key tracking effect would still increase on its own, but the effect of keystart would not increase further. At a KStart value of C 3, the key tracking value would be multiplied by 0 for C 3 and all notes below, by 1 for C# 3, and so on. The key number value would reach its maximum of 64 before reaching C 9. When KStart is set above C 4, its effect on key tracking will continue to increase up to C 9, but will not reach full scale at C 9.

You’ll use unipolar values for KStart when you want to cancel the key tracking effect on a DSP function over a sizable portion of the keyboard, but have it increase or decrease throughout the rest of the keyboard’s range. Set high unipolar values for KStart when you want to remove key tracking from the lower notes, applying it only to the higher notes. If you have set a positive value for KeyTrk, set low unipolar values when you want to apply key tracking to the lower notes and pin it at its maximum throughout the upper range of the keyboard. You may want to use low values of key tracking in this case, depending on the DSP function you’re applying.

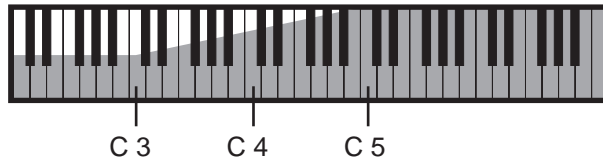
When the value of the KeyTrk parameter is negative, remember that the key tracking is at its minimum effect at C 9, and maximum at C -1. In this case, the key tracking effect will be reduced for notes above the KStart setting. For notes below the keystart note, the normal key tracking amount will apply.



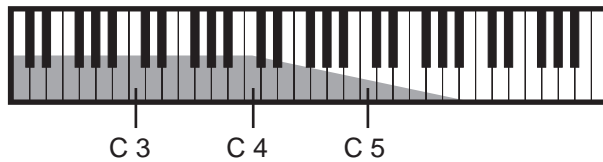
Positive KeyTrk value with KStart value at C4 Unipolar



Positive KeyTrk value with KStart value at C5 Unipolar



Positive KeyTrk value with KStart value at C3 Unipolar



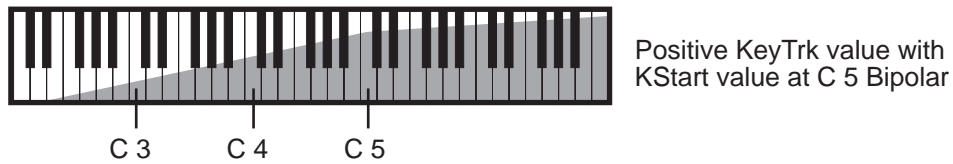
Negative KeyTrk value with KStart value at C4 Unipolar

BIPOLAR KEYSTART

For bipolar KStart values with positive key tracking values, the effect on key tracking depends on whether the KStart parameter is set above or below C 4. When it’s above, the effect on key tracking will be minimum at C -1, reaching its maximum effect on key tracking at the keystart



setting. The normal key tracking curve applies above the keystart setting. When KStart is set below C 4, the effect on key tracking is maximum at C 9, decreasing with each successive note closer to the keystart setting, and remaining constant at the keystart setting and below. The normal key tracking curve applies below the keystart setting.



Use bipolar settings for KStart when you want to gradually increase or decrease the key tracking effect of the currently selected DSP function across the entire keyboard range. With KStart at C 4 bipolar, playing C 4 will apply the DSP function at the level you set with the Adjust parameter, and will increase or decrease with higher or lower notes, depending on your settings for KeyTrk.

When KeyTrk is set to a negative value, the effect on key tracking is reversed. For keystart settings above C 4, the effect on key tracking will be maximum at C -1, decreasing with each note closer to the keystart setting, and remaining constant at and above the keystart setting. For keystart settings below C 4, the effect on key tracking will be minimum at C 9, increasing with each note closer to the keystart setting, and remaining constant for notes at and below the keystart setting.

KStart is available for many of the *non-linear DSP functions*, like SHAPER and WRAP. If you like the control it gives you, you can simulate its effect by using the FUNs. To simulate unipolar keystart, assign Key Number (KeyNum) as one of the inputs to a FUN, and select one of the diode equations for the FUN's Function parameter. To simulate bipolar keystart, assign Bipolar Key Number (BKeyNum) as one of the inputs of a FUN. Then assign those FUNs to some other control source parameter.

## The DSP Functions

### Filters

- ONE-POLE LOWPASS
- TWO-POLE LOWPASS
- TWO-POLE LOWPASS, -6 dB RESONANCE
- TWO-POLE LOWPASS, +12 dB RESONANCE
- FOUR-POLE LOWPASS WITH SEPARATION
- GATED LOWPASS
- ONE-POLE HIGHPASS
- TWO-POLE HIGHPASS
- FOUR-POLE HIGHPASS WITH SEPARATION
- ONE-POLE ALLPASS
- TWO-POLE ALLPASS
- TWO-POLE NOTCH
- TWO-POLE NOTCH, FIXED WIDTH
- DOUBLE NOTCH WITH SEPARATION
- TWO-POLE BANDPASS
- TWO-POLE BANDPASS, FIXED WIDTH
- TWIN PEAKS BANDPASS

Filters are widely used in synthesis to change the timbre of a sound by manipulating the amplitude of specific partials. When using filters, you always set a reference point (cutoff or center frequency) that determines which partials the filters affect. Here's a quick summary of the effects of the filter functions.

**Lowpass** filters cut the levels of all partials above the cutoff frequency without affecting the partials at or below the cutoff frequency (the low frequencies pass through). **Highpass** filters do the opposite; they cut the levels of all partials *below* the cutoff frequency without affecting the partials at or above the cutoff frequency.

**Notch** filters, as the name implies, cut the levels of partials in a range between high and low frequency. Consequently the "cutoff" frequency is referred to as the center frequency. With notch filters, the levels of partials at the center frequency are cut, while the levels of partials above and below the center frequency are unaffected. **Bandpass** filters are the opposite of notch filters; they leave the levels of partials at the center frequency unchanged, and cut the levels of partials above and below the center frequency.

The use of lowpass, highpass, notch, and bandpass filters is often referred to as subtractive synthesis, since the timbre of a sound is changed by removing certain partials.

Allpass filters, instead of cutting or boosting the partials of a sound, change the phase of the partials as their frequencies pass through the center frequency.

### **Filter Terminology**

**Rolloff** - Filters do not usually cut all frequencies precisely at their cutoff point. Instead, the amplitude of the frequencies above (or below, in case of a hi pass filter) the cutoff decrease by a fixed amount per octave - for example, 6 dB per octave. This curve of lessening amplitude is called a rolloff.

**Poles** - The number of poles in a filter affect how sharp the rolloff is. The more poles there are, the sharper the rolloff, meaning that the cutoff will have a more dramatic effect on the sound. The K2vx has one pole, two pole, and four pole filters available. A one pole filter has a 6 dB per octave cutoff; a two pole is 12 dB per octave; and a four pole is 24 dB per octave.

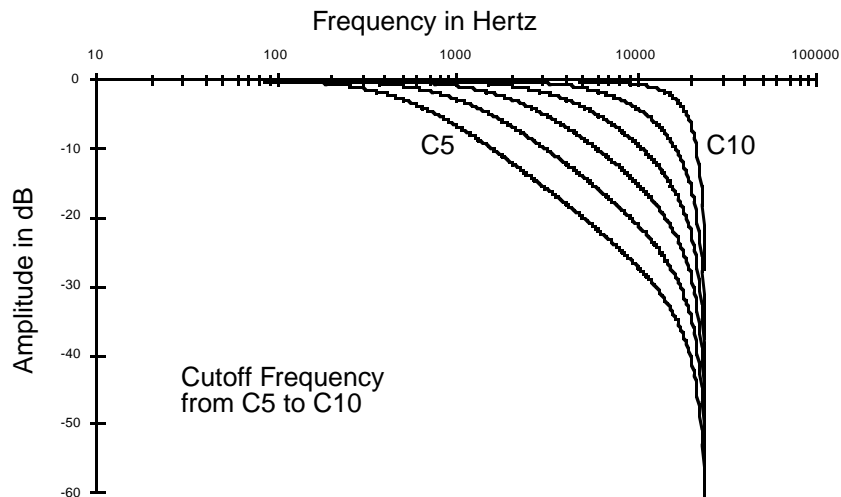
**Resonance** - In a filter that has resonance, the frequencies near the cutoff are given an increase or decrease in amplitude. If you decrease these frequencies, you are essentially creating a longer rolloff. But if you increase those frequencies thereby emphasizing them, it creates a distinctive sound that you will very likely recognize. Resonance is also sometimes called Emphasis or Q on various synthesizers. Resonance on the K2vx is implemented in one of two ways. On some filters, the resonance is fixed, adding or subtracting a specific amount of dB to the affected frequencies (the ones near the cutoff). On other filters, you can control the amount of resonance applied. In the case of these filters, there will always be a separate control page for the resonance.

**Separation** - Four of the filters in the K2vx (both Four Pole filters, the Double Notch, and the Twin Peaks) are actually two filters combined into one DSP function. For these filters, you will find a control page called Separation. This allows you to shift the cutoff frequency of the second filter, creating a separation in the cutoff frequencies of the two filters. In the case of the Notch and Band Pass filters, this can be used to create two separate notches or band passes. In the case of the four pole filters, it affects the shape of the roll off. For the four pole filters, separation set to 0 creates sharp rolloff of 24dB per octave.

## **How to read the graphs**

The graphs show the rolloff curve, using several different values to show how they change the shape of the curve. Amplitude is always on the vertical axis. Frequency is always on the horizontal axis. You will notice on several graphs that the curve becomes more dramatic as the cutoff frequency is set at a higher value. This is because the highest frequency the K2vx can produce is 20Khz, so as the cutoff is set to higher values, there are fewer frequencies available before it is past the range of the K2vx.

### One-pole Lowpass Filter (LOPASS)



Frequencies below the cutoff frequency are unaffected by this filter. At the cutoff frequency, the signal is attenuated 3 dB. There's a rolloff of 6 dB per octave above the cutoff frequency—that is, the signal is attenuated 6 dB with each octave above the cutoff. The resonance—the amount of cut or boost at the cutoff frequency—is fixed at -3dB. When the cutoff frequency is well below the the lowest-frequency partials of a sound, lowering the cutoff further will not affect the timbre of the sound, but will reduce its overall amplitude.

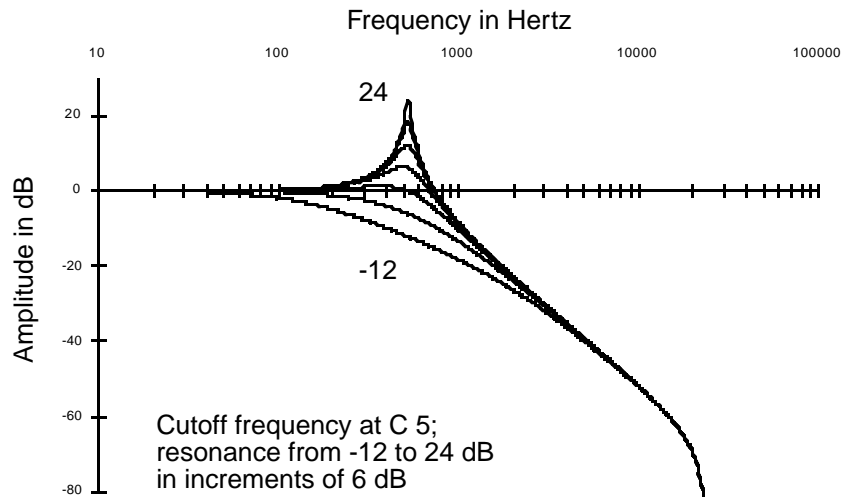
```

EditProg: F1 FRQ(LOPASS) <> Layer: 1/1
Coarse: C 4 262 Hz Src1 : OFF
Fine : 0ct Depth : 0ct
KeyTrk: 0ct/key Src2 : OFF
VelTrk: 0ct DptCt1: OFF
Pad : 0dB MinDpt: 0ct
MaxDpt: 0ct
<more F1 FRQ F2 F3 F4 AMP more>
    
```

PARAMETER	RANGE OF VALUES
COARSE ADJUST	C 0 16 Hz to G 10 25088 Hz
FINE ADJUST	± 100 cents
KEY TRACKING	± 250 cents per key
VELOCITY TRACKING	± 10800 cents
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 10800 cents
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 10800 cents
MAXIMUM DEPTH, SOURCE 2	± 10800 cents

The Coarse Adjust parameter sets the cutoff frequency in terms of a key name. The remaining parameters (except Pad) alter the cutoff frequency in increments of cents. You'll notice that positive values for key tracking have an interesting effect on the function of lowpass filters; positive key tracking values raise the cutoff frequency for high notes and lower it for low notes. More specifically, a value of 100 cents per key on this page, when filtering a constant waveform like a sawtooth, would result in waveforms of exactly the same shape for all pitches of the waveform. The cutoff frequency moves in sync with the frequencies of the waveform's partials as different pitches are generated. Negative key tracking values will steepen the rolloff of lowpass filters above the cutoff. The Pad parameter, as always, attenuates the signal at the input to the function. These parameters affect all the lowpass filters similarly.

**Two-pole Lowpass Filter (2POLE LOWPASS)**



The two-pole lowpass filter has a rolloff of 12 dB per octave above the cutoff frequency. This is a two-stage function, so it has two control input pages. The first affects the cutoff frequency, and has the same parameters as the one-pole lowpass. The second control input page (F2 RES) affects the resonance of the filter. Resonance is a cut or boost in amplitude of the partials in the vicinity of the cutoff frequency.

Set the resonance with the Adjust parameter on the F2 RES control input page, and use the other parameters to assign various controls to alter it. If a boost is applied at frequencies where there are high-amplitude partials, the signal may clip. The Pad parameter on the F1 FRQ page will reduce the clipping, but there's no harm in keeping it if you like the sound.

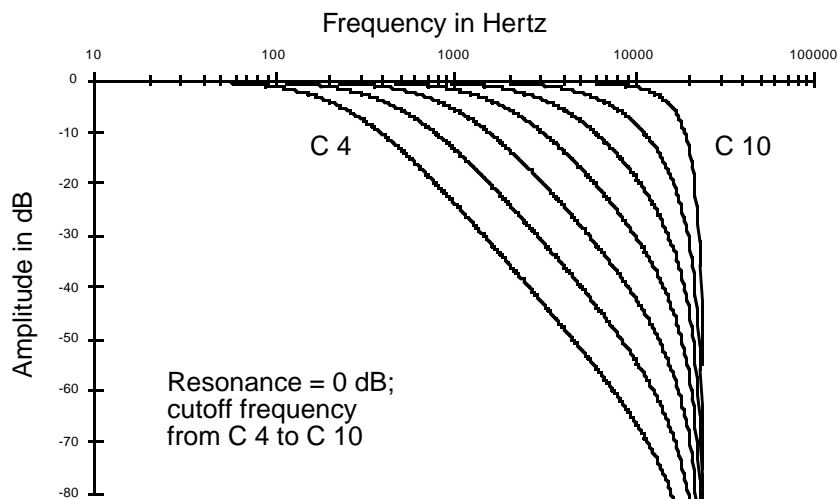
```

EditProg: F2 RES(2P LOWPASS) (<) Layer:1/1
Adjust: 0.0dB Src1 :OFF
Depth : 0.0dB
Src2 :OFF
KeyTrk: 0.00dB/key DptCt1:OFF
VelTrk: 0.0dB MinDpt: 0.0dB
MaxDpt: 0.0dB
<more F1 FRQ F2 RES F3 F4 AMP more>
    
```

PARAMETER	RANGE OF VALUES
-----------	-----------------

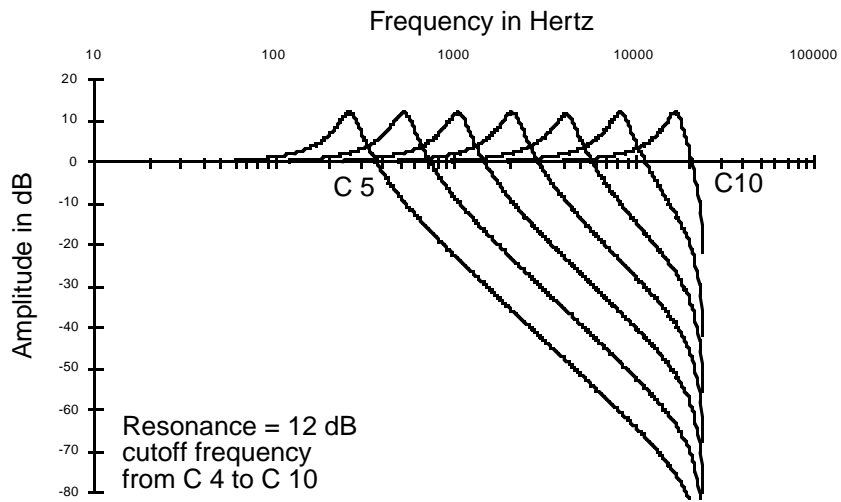
<b>ADJUST</b>	<b>-12 to +24 dB</b>
<b>KEY TRACKING</b>	<b>± 2.00 dB per key</b>
<b>VELOCITY TRACKING</b>	<b>± 30 dB</b>
<b>SOURCE 1</b>	<b>Control Source list</b>
<b>SOURCE 1 DEPTH</b>	<b>± 30 dB</b>
<b>SOURCE 2</b>	<b>Control Source list</b>
<b>SOURCE 2 DEPTH CONTROL</b>	<b>Control Source list</b>
<b>MINIMUM DEPTH, SOURCE 2</b>	<b>± 30 dB</b>
<b>MAXIMUM DEPTH, SOURCE 2</b>	<b>± 30 dB</b>

**Two-pole Lowpass Filter, -6 dB Resonance (LOPAS2)**



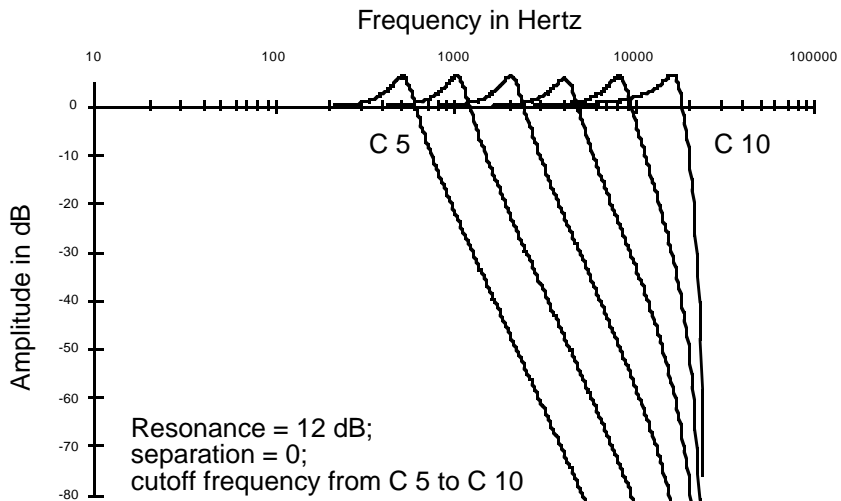
Using this filter is the same as using two one-pole lowpass filters in successive algorithm blocks. Since its resonance is fixed at -6 dB, it's also the same as using 2POLE LOWPASS with the resonance set to -6 dB. You'd use this filter when you want a 12 dB per octave rolloff but don't need to set a resonance level. This would leave you free to use another DSP function in the algorithm, since it's a one-stage function.

### Two-pole Lowpass Filter, +12 dB Resonance (LP2RES)

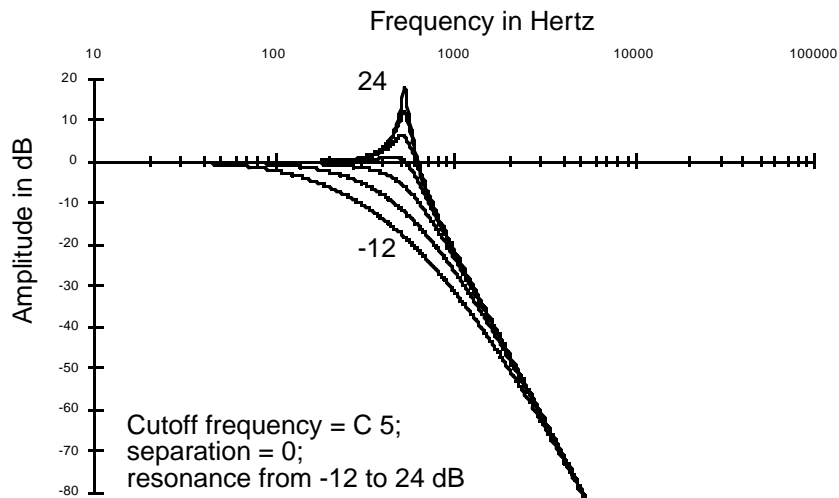


This is similar to LOPAS2; the only difference is that its resonance is fixed at +12 dB.

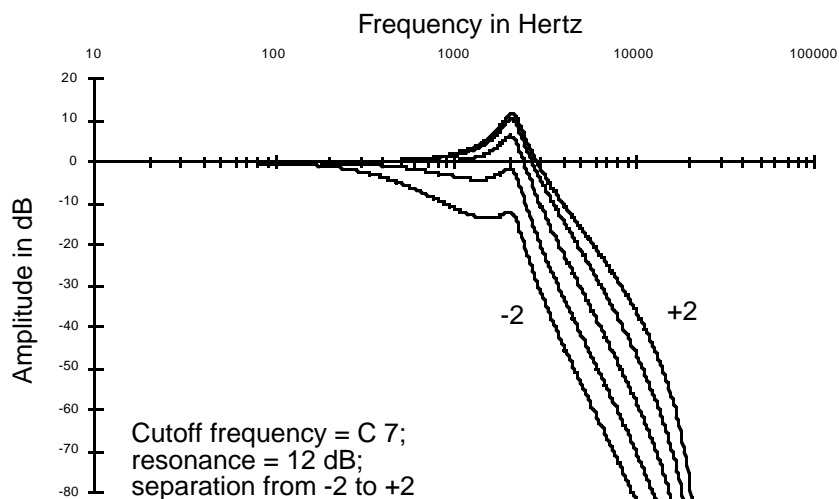
### Four-pole Lowpass Filter with Separation (4POLE LOPASS W/ SEP)



4-Pole Lowpass Filter:  
Resonance



4-Pole Lowpass Filter:  
Separation in Octaves



This combines 2POLE LOWPASS and LOPAS2 in one three-stage function. The parameters on the F1 FRQ control input page affect the cutoff frequencies of both filters. The parameters on the F2 RES page affect the resonance of 2POLE LOWPASS. The parameters on the F3 SEP page shift the cutoff frequency of LOPAS2, creating a separation between the cutoff frequencies of the two filters. Positive values raise the cutoff frequency of LOPAS2, while negative values lower it. If no separation is applied, there's a 24 dB per octave rolloff above the cutoff frequency.

```

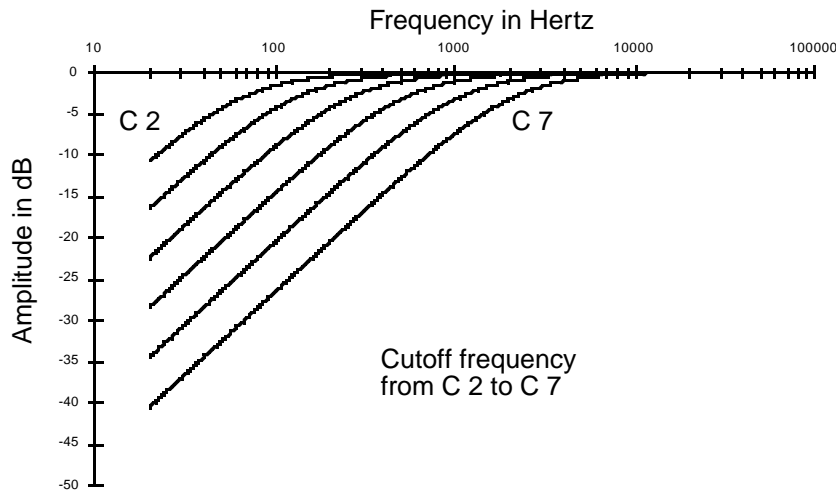
|<|>|Prog: F3 SEP(4P LOWPASS) <>|Layer: 1/1
Adjust: 0ct          Src1 : OFF
Fine   : 0ct        Depth : 0ct
KeyTrk: 0ct/key     Src2 : OFF
VelTrk: 0ct        DptCt1: OFF
                               MinDpt: 0ct
                               MaxDpt: 0ct
|<more| F1 FRQ | F2 RES | F3 SEP | F4 AMP | more>
    
```

PARAMETER	RANGE OF VALUES
COARSE ADJUST	± 10800 cents
FINE ADJUST	± 100 cents
KEY TRACKING	± 250 cents per key
VELOCITY TRACKING	± 10800 cents
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 10800 cents
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 10800 cents
MAXIMUM DEPTH, SOURCE 2	± 10800 cents

**Gated Lowpass Filter (LPGATE)**

You may be familiar with gates as applied to effects like reverb, where the effect shuts off abruptly after a specified time. The gated lowpass filter produces a somewhat similar effect in terms of the sound’s amplitude. The filter’s cutoff frequency is controlled by the AMPENV. When the AMPENV is at 100%, the cutoff frequency is high, so most of the partials are heard. When the AMPENV decays or releases to 0%, the cutoff frequency is low, so only the lowest partials are heard. You’ll hear the distinct effect of the filter closing as the amplitude envelope releases.

**One-pole Highpass Filter (HIPASS)**



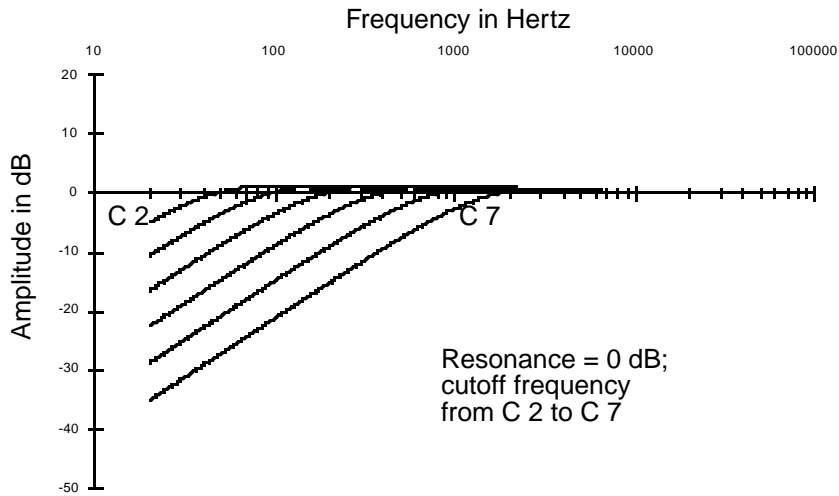
High-frequency partials pass through this filter unaffected. At the cutoff frequency, the signal is attenuated 3 dB. There’s a roll-off of 6 dB per octave below the cutoff frequency. The resonance is fixed at -3dB. When the cutoff frequency is well above the the highest-frequency partials of a sound, raising the cutoff further will not affect the timbre of the sound, but will merely attenuate it further.

The Coarse Adjust parameter sets the cutoff frequency in terms of a key name. The remaining parameters (except Pad) alter the cutoff frequency in increments of cents. Positive key tracking values raise the cutoff frequency for high notes and lower it for low notes. More specifically, a value of 100 cents per key on this page, when filtering a constant waveform like a sawtooth, would result in waveforms of exactly the same shape for all pitches of the waveform. The cutoff frequency moves in sync with the frequencies of the waveform’s partials as different pitches are generated. Negative key tracking values will steepen the rolloff of highpass filters below the



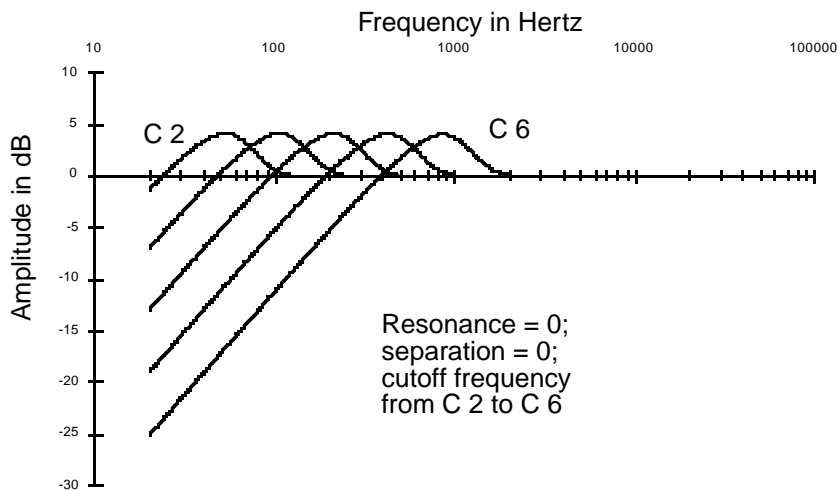
cutoff. The Pad parameter, as always, attenuates the signal at the input to the function. These parameters affect all the highpass filters similarly.

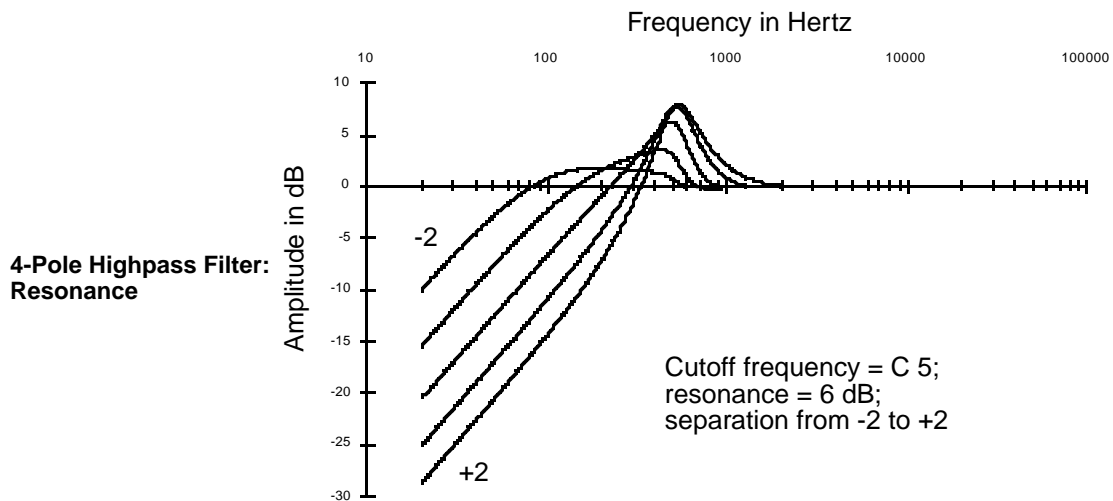
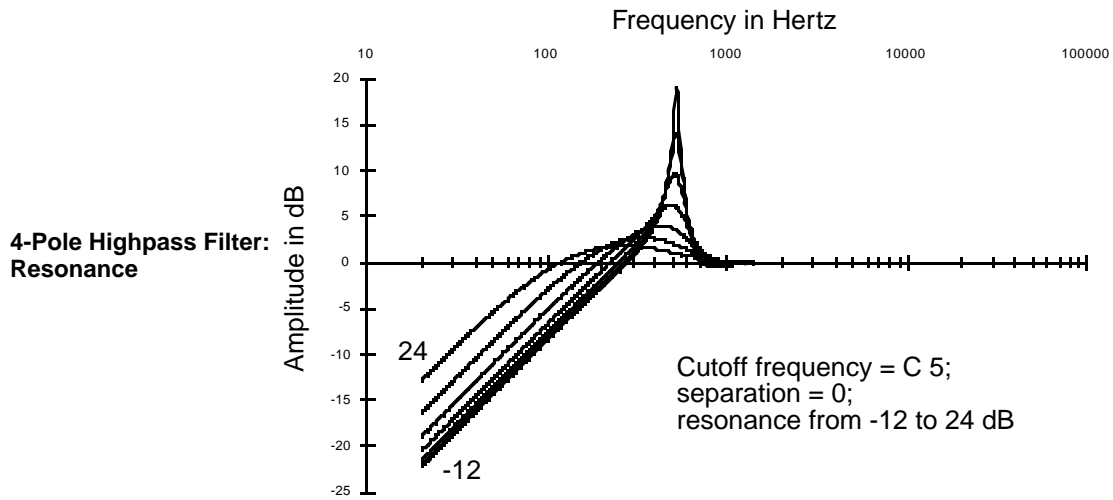
**Two-pole Highpass Filter (HIPAS2)**



This is very similar to HIPASS. The primary difference is in the steepness of the rolloff at the cutoff frequency. Below the cutoff frequency, the rolloff is similar to that of HIPASS, except that there's a one-octave shift—that is, HIPASS with a cutoff frequency of C 3 will sound nearly the same as HIPAS2 with a cutoff of C 4. In other words, HIPASS gives you greater attenuation of low frequencies when set to the same cutoff frequency as HIPAS2.

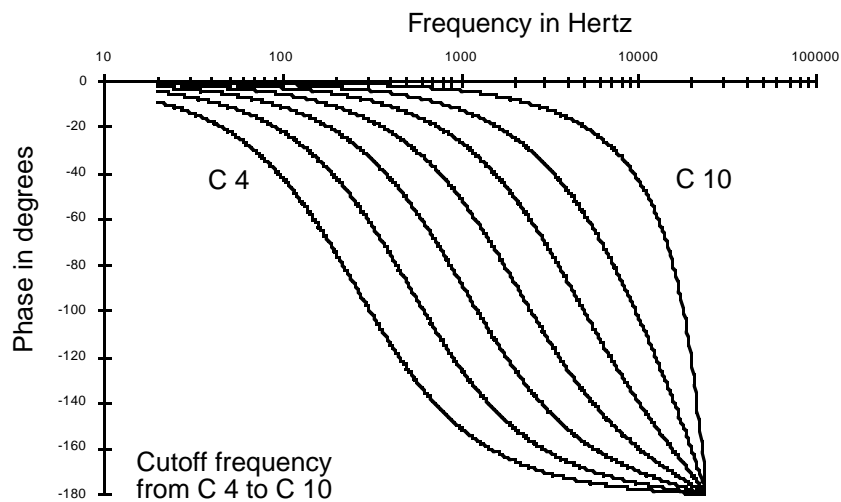
**Four-pole Highpass Filter with Separation (4POLE HIPASS W/ SEP)**





This combines two of the 2POLE HIPASS filters into one three-stage function. It has a rolloff of 6 dB per octave below the cutoff frequency. The parameters on the F1 FRQ control input page affect the cutoff frequencies of both filters. The parameters on the F2 RES page affect the resonances of the first filter. There will always be a slight extra boost of partials at the cutoff frequency, even at low resonance settings. The parameters on the F3 SEP page shift the cutoff frequency of the second 2POLE HIPASS, creating a separation between the cutoff frequencies of the two filters. Positive values raise the cutoff frequency of the second 2POLE HIPASS, while negative values lower it. If no separation is applied, there's a 24 dB per octave rolloff below the cutoff frequency. A variety of responses can be produced by adjusting the resonance and separation settings.

## One-pole Allpass Filter (ALPASS)

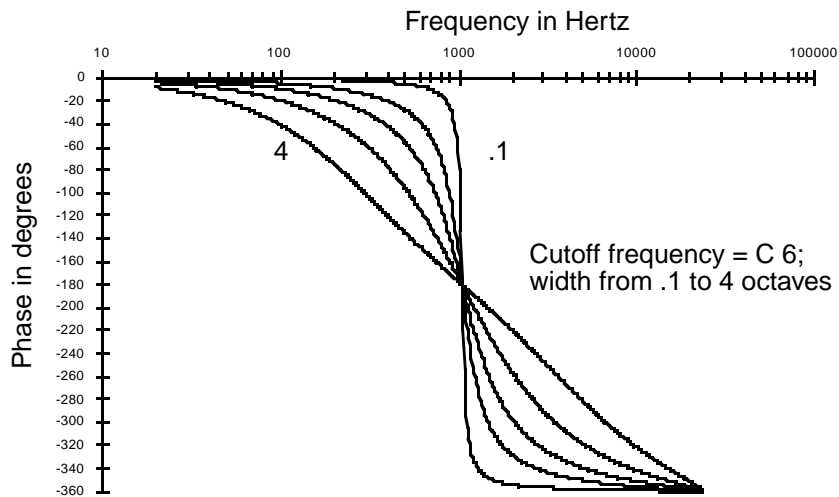
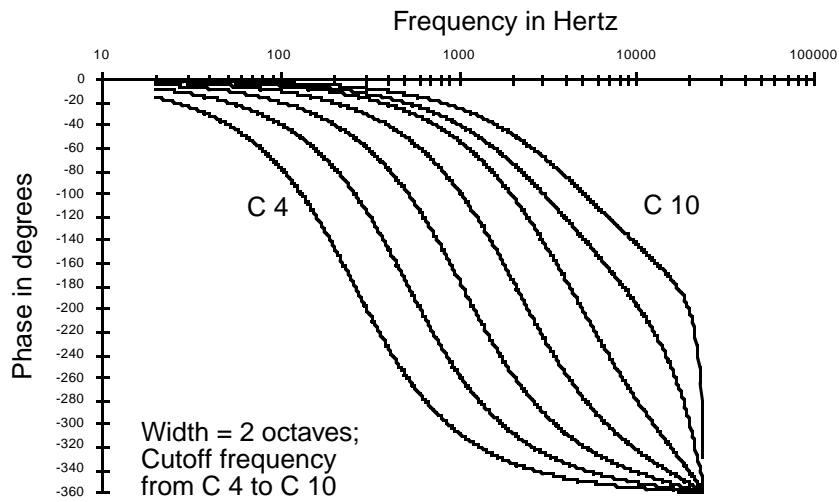


Allpass filters do not affect a sound's frequency response (the amplitude of partials at various frequencies), but change the phase of each partial depending on its proximity to the center frequency. The phase shift is -90 degrees for partials at the center frequency. It rises toward 0 degrees for partials at frequencies below the center, and falls toward -180 degrees for partials at frequencies above the center. With low-frequency waveforms, you'll be able to hear this phase shift. As a rule, however, the ear is not sensitive to phase shifts unless they're changing, so you'll usually want to use Source 1 or 2, and assign an LFO to sweep the center frequency up and down.

Periodic phase shifts like those induced by an LFO on the center frequency will cause a vibrato-like variation in the pitch of a sine wave input. This vibrato effect will be less regular for more complex partials. The greater the depth setting of the control source using the LFO, the greater the vibrato effect.

The amount of vibrato effect also depends on the speed and amount of the phase shift. Try adjusting the rate of the LFO controlling the center frequency. Another way to increase the amount of phase shift is to use the two-pole allpass filter, or to use the one-pole allpass filter in more than one algorithm block.

### Two-pole Allpass Filter (2POLE ALLPASS)



Using 2POLE ALLPASS is very similar to using ALPASS, with two differences. First, the phase shift is -180 degrees for partials at the center frequency, approaching 0 degrees for partials at low frequencies, and approaching -360 degrees for partials at high frequencies.

Second, since this is a two-stage function, there's an additional control input page (F2 WID) which controls the filter width. The parameters on this page affect the frequency range, measured in octaves, where most of the phase shifting occurs. Small values cause a drop from 0 to -360 in the phase shift to occur near the center frequency, while large values spread the drop in the phase shift over a broader frequency range. Small values tend to affect just a few partials, leaving others mostly untouched. The affected partials seem to become detached from the others, creating the illusion of an additional sound source.

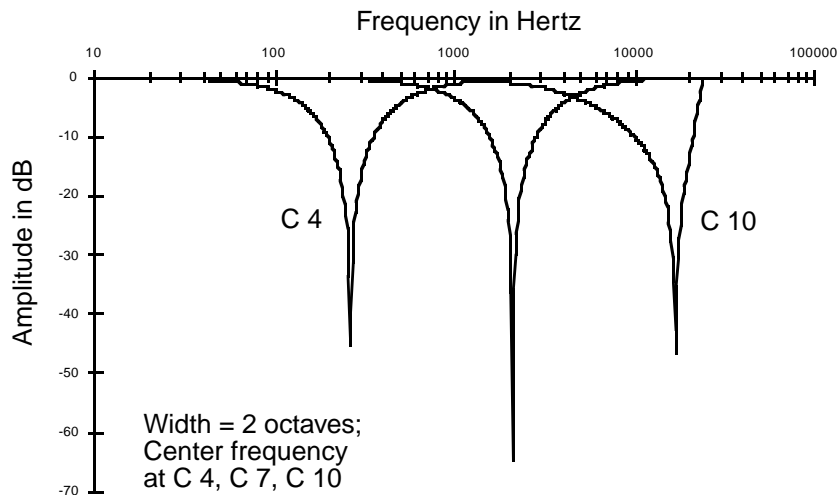
If you leave the center frequency constant and assign an LFO to vary the width, partials with frequencies above the center will shift their pitches in the opposite direction of partials below the center frequency.

```

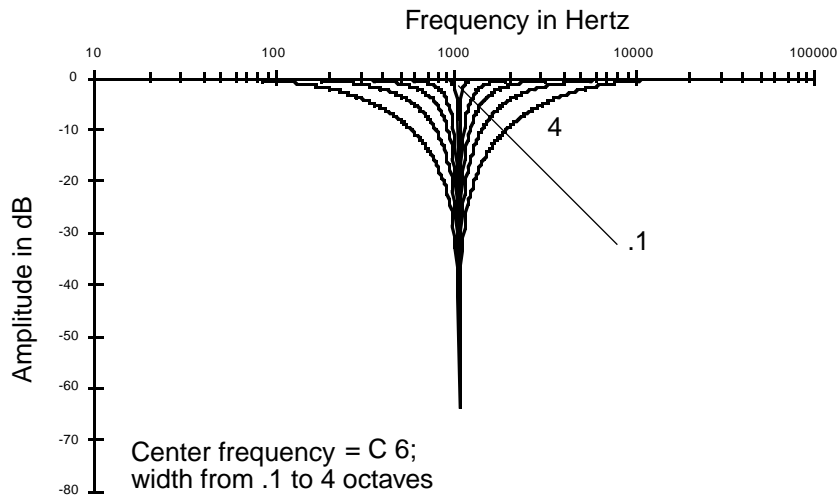
EditProg:F2 WII(2P ALPHSS) <>Layer:1/1
Adjust:0.010oct Src1 :OFF
Depth :0.00oct Src2 :OFF
KeyTrk:0.000oct/key DptCtl:OFF
VelTrk:0.000oct MinDpt:0.00oct
MaxDpt:0.00oct
<more F1 FRQ F2 WII F3 AMP F4 AMP more>
    
```

PARAMETER	RANGE OF VALUES
ADJUST	0.010 to 5.000 octaves
KEY TRACKING	± .200 octaves per key
VELOCITY TRACKING	± 5.00 octaves
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 5.00 octaves
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 5.00 octaves
MAXIMUM DEPTH, SOURCE 2	± 5.00 octaves

**Two-pole Notch Filter (NOTCH FILTER)**

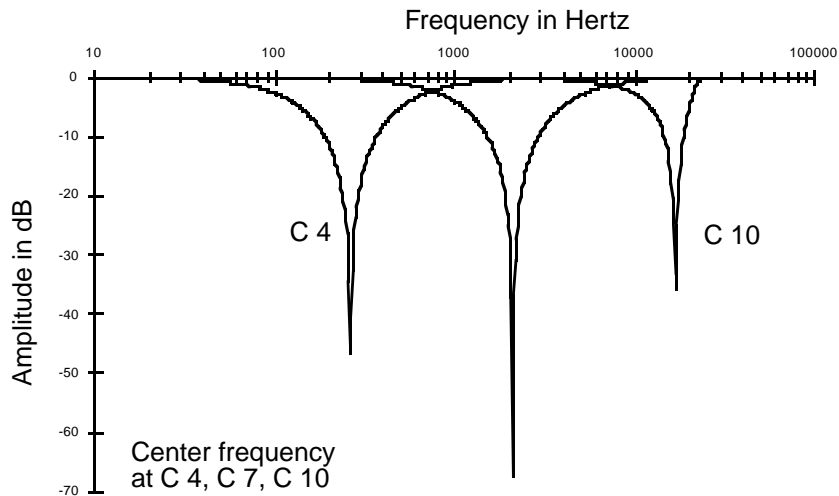


**2-Pole Notch Filter:  
Width in octaves**



The two-pole notch filter has two control input pages, one for center frequency, one for width. Partials with frequencies above or below the notch will be unaffected. Within the notch, partials will be attenuated according to the width of the notch. The width is defined in terms of the number of octaves between the points on the signal's attenuation curve where the attenuation is 3 dB (see the explanation of F2 WID for the PARAMETRIC EQ function). For example, if the width is set at four octaves, then the attenuation will be 3 dB at two octaves in either direction from the center frequency. There's no attenuation of partials at more than two octaves in either direction from the center frequency.

**Two-pole Notch Filter, Fixed Width (NOTCH2)**



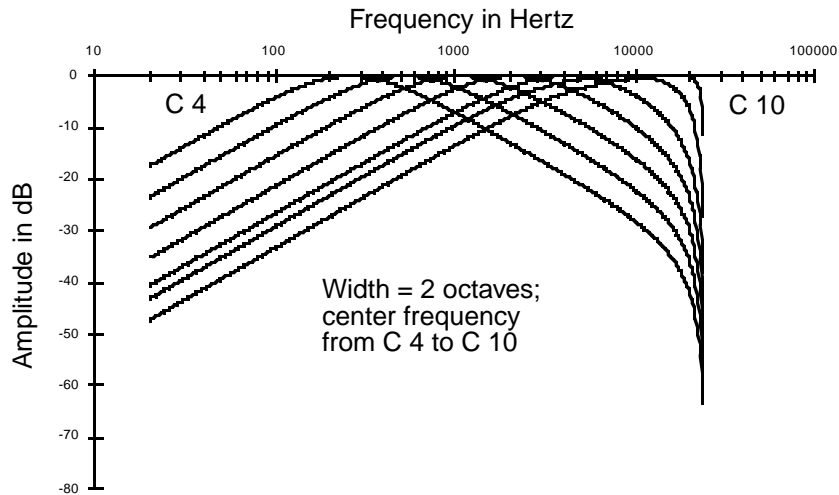
The only functional difference between NOTCH2 and NOTCH FILTER is that the width of NOTCH2 is fixed at 2.2 octaves. This gives you a one-stage notch filter function.

**Two-pole Bandpass Filter (BANDPASS FILTER)**

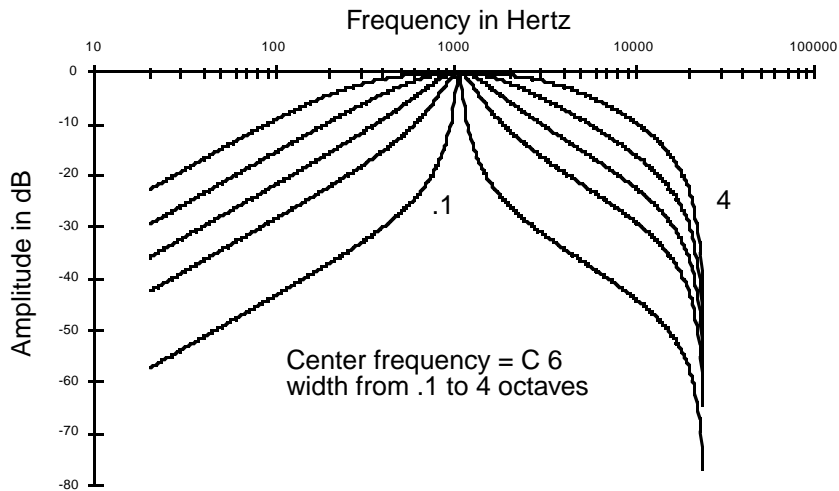
This is essentially the opposite of a notch filter; it passes all partials at the center frequency, and cut the levels of partials above or below the center frequency. The width is defined the same as for the double notch filter.

The gain at the center frequency is 0 dB. Small values for width (a narrow bandpass) may produce a very quiet signal unless the center frequency matches the frequency of a strong sine wave partial. Wide bandpasses may result in a quiet signal if they're centered in a region of the sound where the partials are weak. You can easily boost these quiet signals with the parameters on the F4 AMP page.

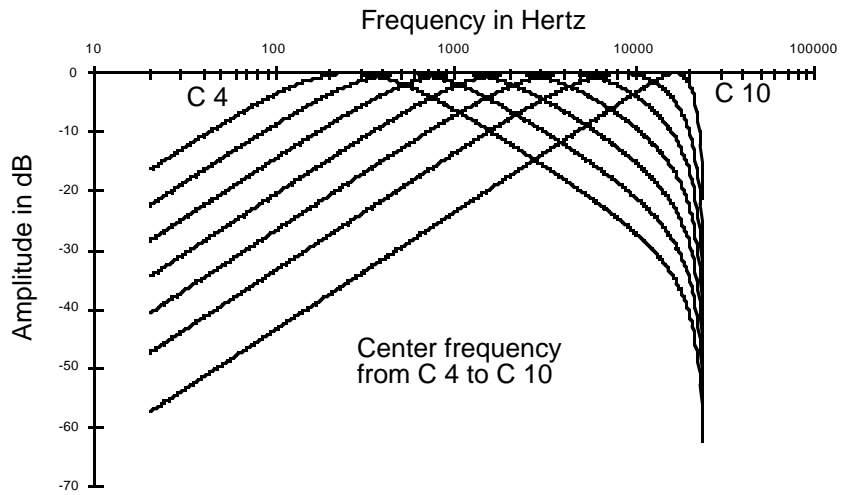
**Bandpass Filter:  
Frequency**



**Bandpass Filter:  
Width in octaves**

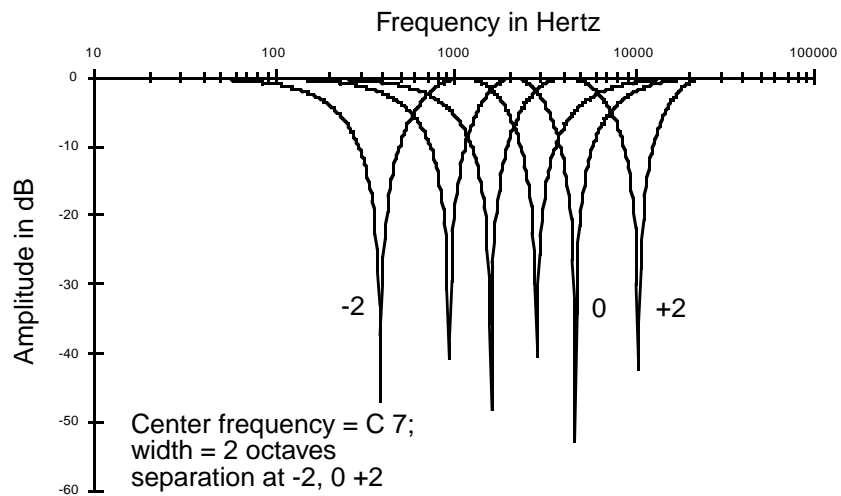


**Two-pole Bandpass Filter, Fixed Width (BAND2)**

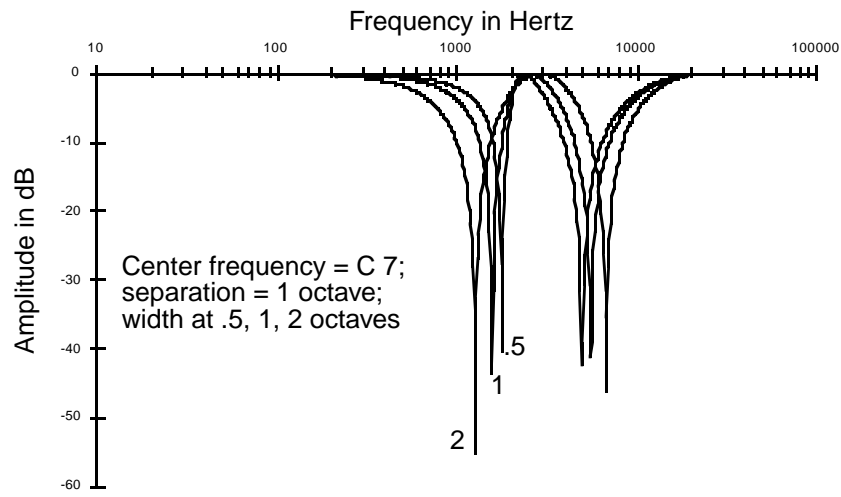


The only functional difference between BAND2 and BANDPASS FILTER is that the width of BAND2 is fixed at 2.2 octaves. This gives you a one-stage bandpass filter function.

**Double Notch Filter with Separation (DOUBLE NOTCH W/ SEP)**







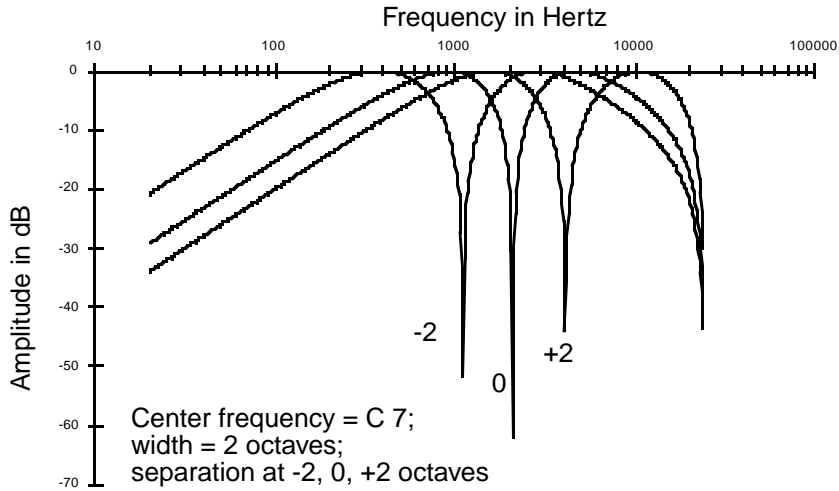
This is a three-stage function that puts two notches in the frequency response. As with NOTCH FILTER and NOTCH2, there are control input pages for frequency and width. A third control input page affects the separation of the notches.

Setting the center frequency on the F1 FRQ page defines the frequency that's halfway between the notches. The settings for the separation affect the behavior of the width control parameters. When the separation is 0, the notches are close to the center frequency, and the width control parameters control the widths of both notches equally. Positive values for separation move the notches apart, and cause the width control parameters to affect the width of the higher-frequency notch more than the width of the lower-frequency notch. Negative values for separation will move the notches apart to the same extent, but will cause the width control parameters to affect the width of the lower-frequency notch more than the width of the higher-frequency notch.

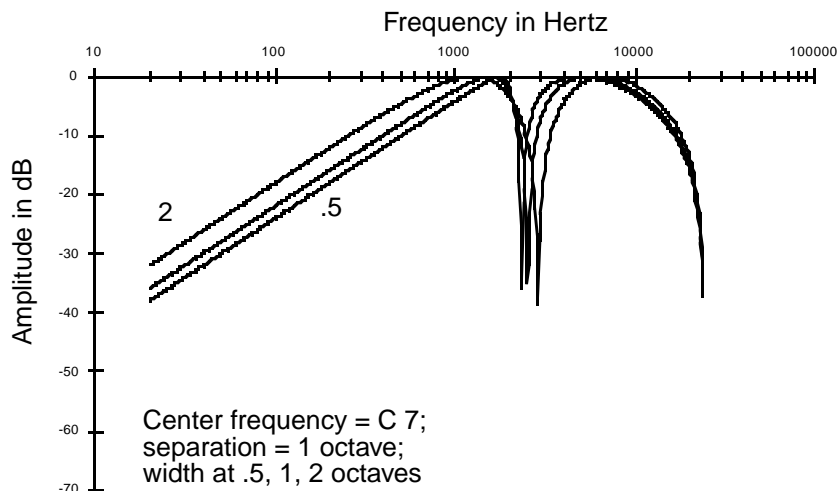
### **Twin Peaks Bandpass Filter (TWIN PEAKS BANDPASS)**

The control parameters for TWIN PEAKS BANDPASS work the same way as for DOUBLE NOTCH FILTER, but of course, you get peaks instead of notches—that is, the amplitudes of partials near the center frequency are high, and the amplitudes are increasingly attenuated at frequencies farther from the center.

**Twin Peaks  
Bandpass Filter:  
Separation**



**Twin Peaks  
Bandpass Filter:  
Width**



## Equalization (EQ)

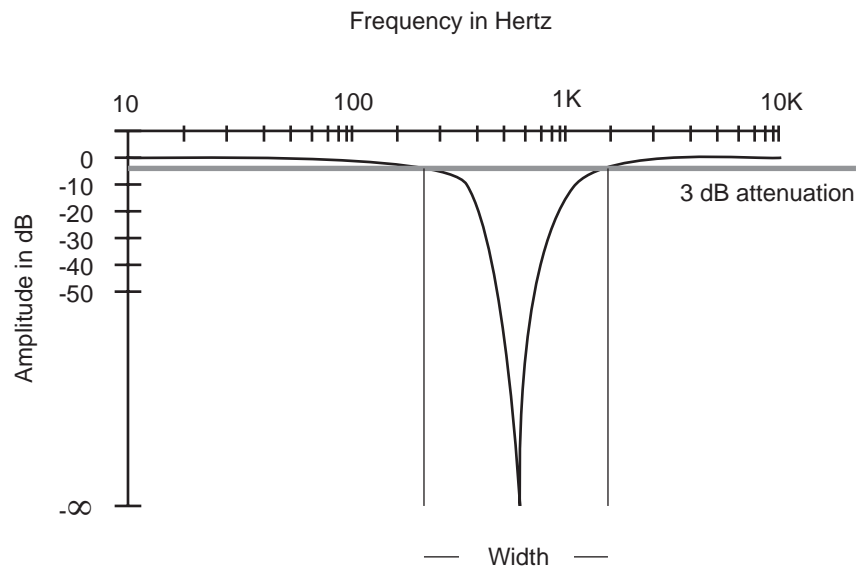
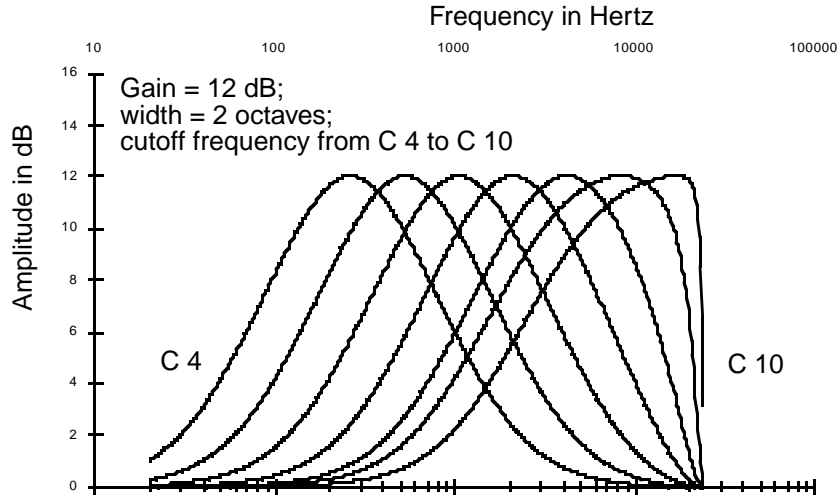
Equalization is a specialized filtering process that lets you boost or cut the amplitude of a specified range of frequencies.

- PARAMETRIC EQ
- MID-RANGE PARAMETRIC EQ
- BASS TONE CONTROL
- TREBLE TONE CONTROL
- STEEP BASS TONE CONTROL

### Parametric Equalizer (PARAMETRIC EQ)

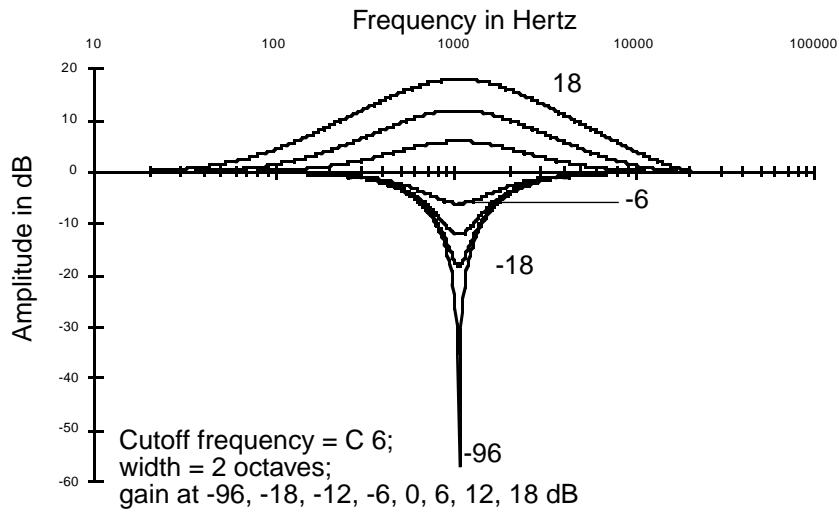
This function has three interacting parameters, each with its own control input page: center frequency, width, and amplitude. The center frequency is the center of the range of frequencies that will be boosted or cut by the amplitude setting. The width is the entire range of frequencies

that will be affected by the amplitude setting. For the K2vx, the width is defined by imagining an amplitude curve with a level (in dB) of -infinity (minus infinity) at the center frequency, then measuring the distance (in octaves) between the points on the curve where the amplitude is attenuated by 3dB. See the diagram below.

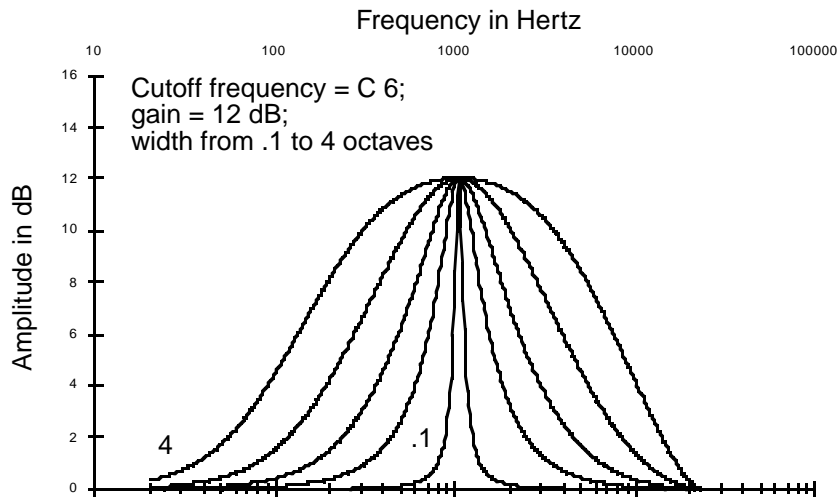


**DSP Functions**

Equalization (EQ)



When you're using the Parametric EQ, you might use the following sequence. Set the center frequency (press the F1 FRQ soft button to select its control input page). The frequency is measured in terms of each note of the keyboard. The frequency in Hertz of each note appears with the note name as the value for the Adjust parameter. Next, select the width control input page (the F2 WID soft button) to determine how wide a range will be affected by the amplitude adjustment. Then select the amplitude control input page (F3 AMP soft button), and adjust the amplitude of the range you specified with the center frequency and width settings. You'll probably jump back and forth between these three pages until your ear is satisfied with the sound.



```

EditProg: F1 FRQ (PARAM EQ) <> Layer: 1/1
Adjust: C 4 262Hz Src1 : OFF
Fine : 0ct Depth : 0ct
KeyTrk: 0ct/key Src2 : OFF
VelTrk: 0ct DptCt1: OFF
Ped : 0dB MinDpt: 0ct
MaxDpt: 0ct
<more> F1 FRQ F2 WID F3 AMP F4 AMP <more>
    
```

PARAMETER	RANGE OF VALUES
COARSE ADJUST	C 0 16 Hz to G 10 25088 Hz
FINE ADJUST	± 100 cents
KEY TRACKING	± 250 cents per key
VELOCITY TRACKING	± 10800 cents (9 octaves)
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 96 dB
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 10800 cents
MAXIMUM DEPTH, SOURCE 2	± 10800 cents

The Fine Adjust parameter gives you one-cent precision in setting the center frequency.

```

EditProg: F2 WII(PARA EQ) <> Layer: 1/1
Adjust: 0.010oct Src1 : OFF
Depth : 0.00oct
Src2 : OFF
KeyTrk: 0.000oct/key DptCt1: OFF
VelTrk: 0.00oct MinDpt: 0.00oct
MaxDpt: 0.00oct
<more> F1 FRQ F2 WII F3 AMP F4 AMP >more>

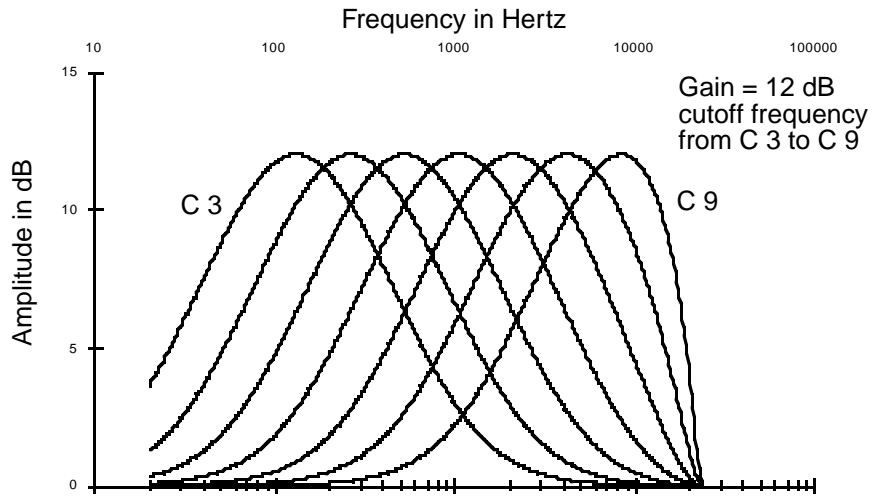
```

PARAMETER	RANGE OF VALUES
ADJUST	0.010 to 5.000 octaves
KEY TRACKING	± .200 octaves per key
VELOCITY TRACKING	± 5.00 octaves
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 5.00 octaves
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 5.00 octaves
MAXIMUM DEPTH, SOURCE 2	± 5.00 octaves

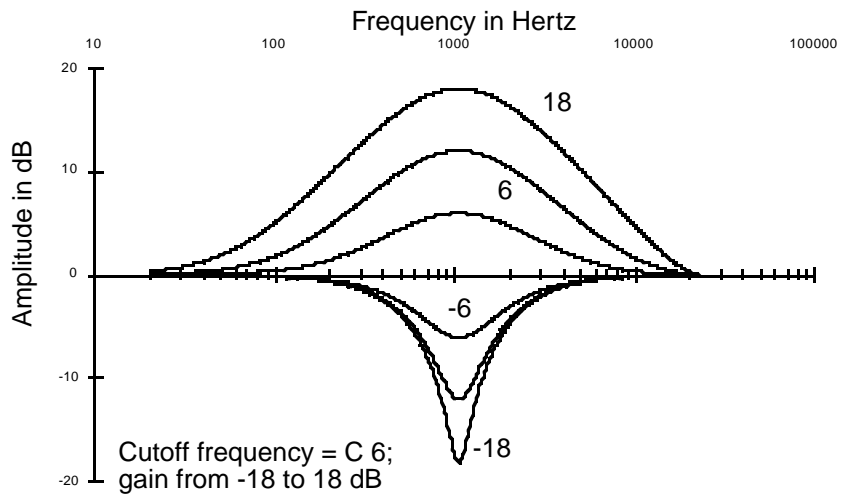
The control input page for the amplitude stage (F3 AMP) is identical to the AMP page described previously, except that there's no Pad parameter.

### Mid-range Parametric EQ (PARA MID)

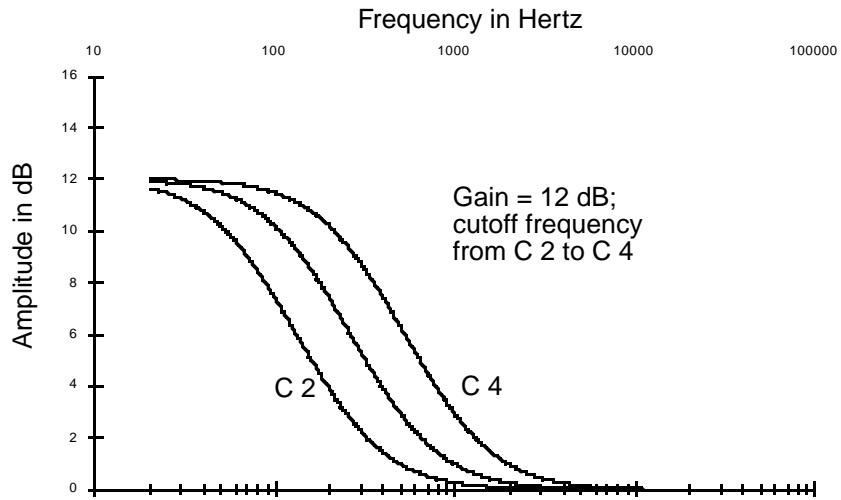
This two-stage function is almost identical to the three-stage Parametric EQ function. The only difference is that the width of PARA MID is fixed at 2.2 octaves. Consequently there's no control input page for the width.



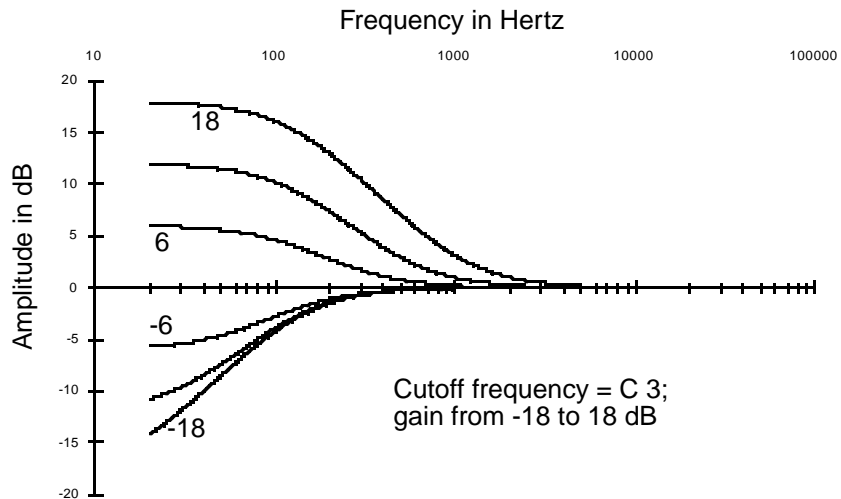
Para Mid: Gain



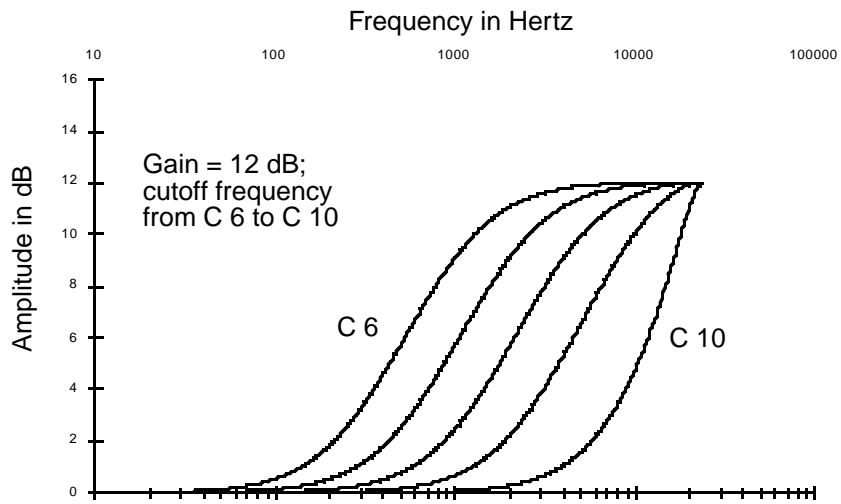
### Bass Tone Control (PARA BASS)



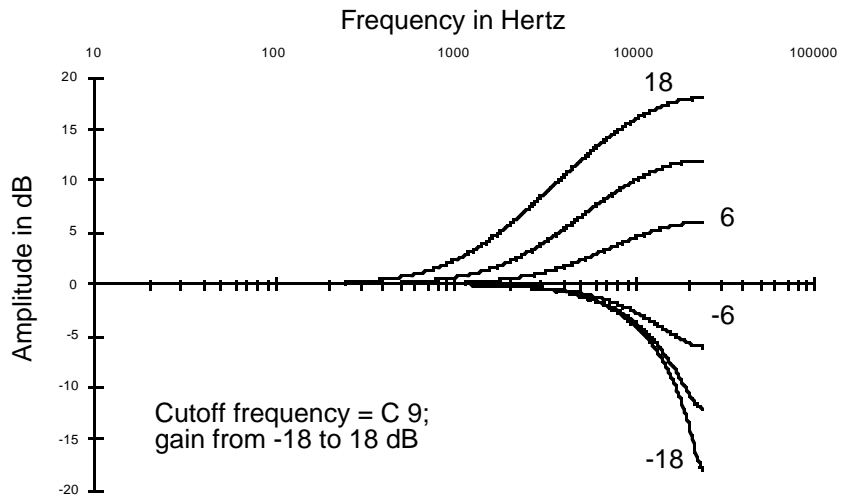
This is a two-stage function, with control input pages for frequency and amplitude. These pages are the same as those for frequency and amplitude in PARA EQ. On the frequency control input page, you'll set the cutoff frequency. For notes above this frequency, the amplitude setting has a diminished effect. On the amplitude control input page, you'll set the amount of cut or boost that's applied to notes below the cutoff frequency. There's a gradual increase in the bass response for each successively lower note. The location of the cutoff frequency will change somewhat as you change the amplitude settings, although the value for the Adjust parameter on the frequency control input page will not reflect the change.



### Treble Tone Control (PARA TREBLE)

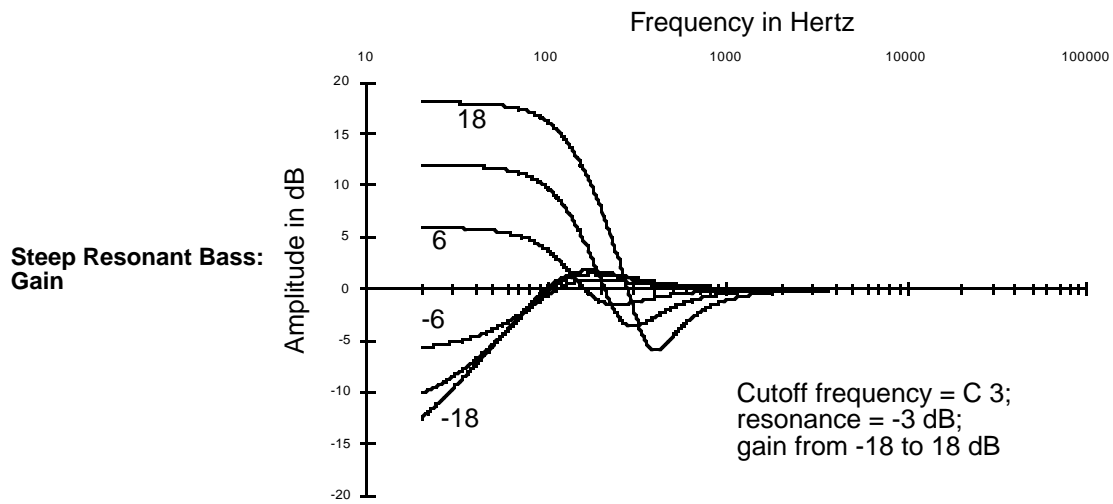
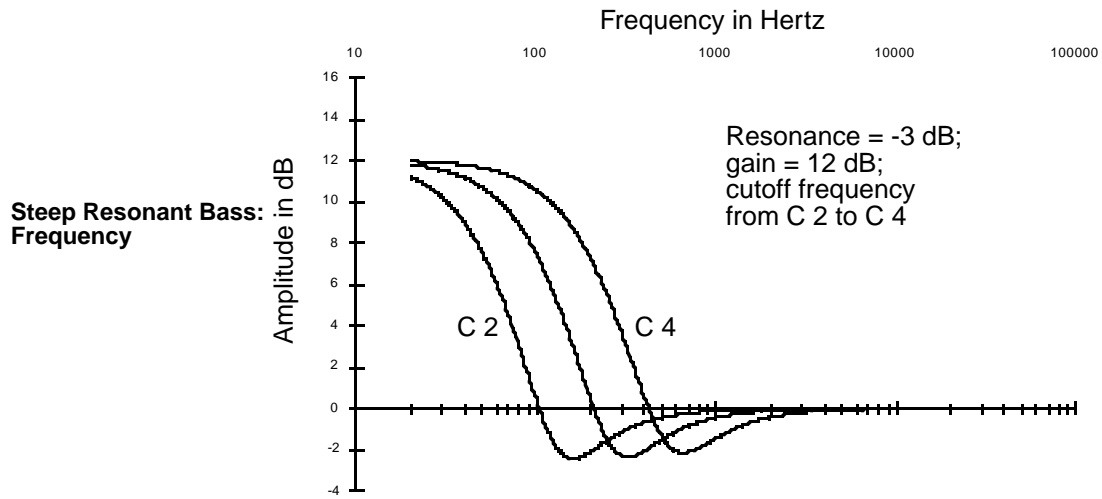


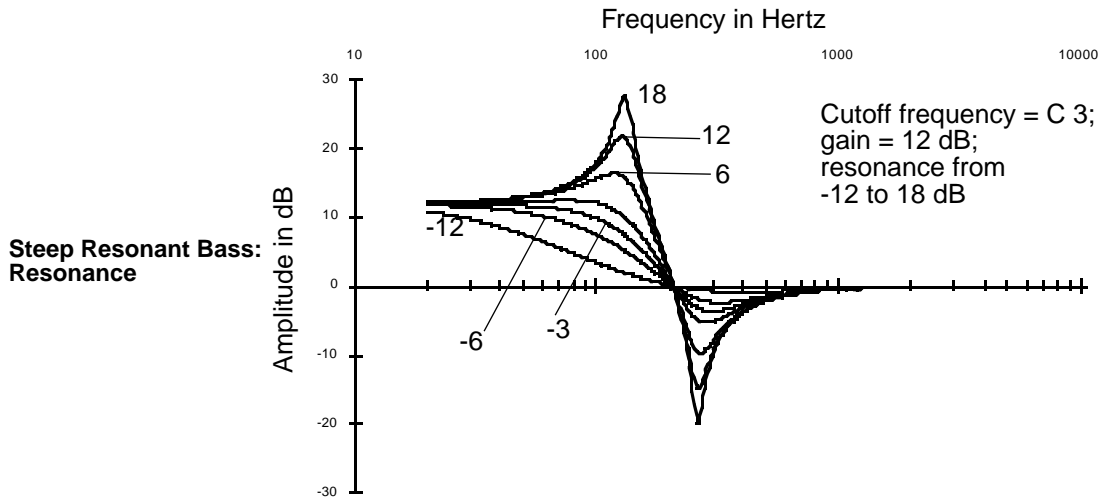
PARA TREBLE is very similar to PARA BASS; the only difference is that the amplitude setting affects notes *above* the cutoff frequency.





### Steep Bass Tone Control (STEEP RESONANT BASS)





This function uses a two-pole lowpass filter to give you a sharper transition in bass response than PARA BASS. Like PARA BASS, there are control inputs pages for cutoff frequency and amplitude, which are identical to those for PARA BASS. There is also a control input page for resonance (also known as “q”), which can boost or cut the amplitude of the partials near the cutoff frequency.

```

EditProg: F2 RES (Steep Bass) Layer: 1/1
Adjust: 0dB Src1 : OFF
Depth : 0.0dB
Src2 : OFF
KeyTrk: 0.00dB/key DptCt1: OFF
VelTrk: 0.0dB MinDpt: 0.0dB
MaxDpt: 0.0dB
<more F1 FRQ F2 RES F3 AMP F4 AMP more>
    
```

PARAMETER	RANGE OF VALUES
ADJUST	-12 to 24 dB
KEY TRACKING	± 2.00 dB per key
VELOCITY TRACKING	± 30 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 96 dB
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 30 dB
MAXIMUM DEPTH, SOURCE 2	± 30 dB

You’ll get the best transition in bass response with a resonance setting of -6 dB. There’s a small frequency range above the cutoff frequency where the response reverses direction (if you’re cutting the amplitude, for example, you’ll get a slight boost just above the cutoff frequency). The higher you set the resonance, the larger this reversal will be, resulting in unusual—but possibly useful—response curves at high resonance values.

## Pitch / Amplitude / Panner

- PITCH
- AMP
- PANNER
- UPPER AND LOWER AMP
- BALANCE AND AMP
- GAIN

### PITCH

We used the PITCH control input page as an example to introduce the common DSP control parameters in Chapter 6, so we won't add much here. The PITCH function modifies the pitch of the layer's keymap as it passes through the sound engine. The PITCH stage of each algorithm is always the first stage. Algorithms 26—31, the Sync algorithms, don't show the PITCH stage on the ALG page, since these algorithms generate their own sawtooth waves, and do not use keymaps.

### AMP

The AMP function is the final stage in every single-output algorithm, and controls the overall amplitude (volume) of the layer. This is an easy way to boost the signal to a more desirable level if it's not loud enough for your purposes. Large values for the Adjust parameter can cause a sound to clip, which will distort most sounds considerably. You may want this effect, and using it won't damage anything, but as a rule, you'll want to avoid clipping your sounds with the AMP (or GAIN) function. There are many other ways to distort your sounds, like DIST, SHAPER, and WRAP, to name a few.

The settings for the parameters on the F4 AMP page affect the gain level for the currently selected layer. So do the settings on the AMPENV page. Compare this with the effect of GAIN, described later in this section.

```

┌──────────┐ Prog: F4 AMP (FINAL AMP) <> Layer: 1/1
Adjust: 0dB          Src1 : OFF
                   Depth : 0dB
                   Src2 : OFF
KeyTrk: 0.00dB/key  DptCt1: OFF
VelTrk: 0dB         MinDpt: 0dB
Pad : 0dB           MaxDpt: 0dB
<more F1 F2 F3 F4 AMP more>
    
```

PARAMETER	RANGE OF VALUES
ADJUST	-96 to 48 dB
KEY TRACKING	± 2.00 dB per key
VELOCITY TRACKING	± 96 dB
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 96 dB
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 96 dB
MAXIMUM DEPTH, SOURCE 2	± 96 dB

**PANNER**

This single-stage function converts a single wire at its input into a double wire at its output, splitting the signal between an “upper” and “lower” wire. This creates the double-output algorithm we discussed in Chapter 6. The parameters on the PANNER page enable you to modify the signal’s routing through the upper and lower wires. By itself the PANNER doesn’t change the pan position of the sound. It just defines what percentage of the currently selected layer’s sound goes to each wire. When you select one of these double-output algorithms, the OUTPUT page for the layer changes to enable you to make pan settings for each wire independently. So when you use the PANNER function, you’ll also want to adjust the Pan parameters on the OUTPUT page, setting the upper wire’s pan fully right, and the lower wire’s pan fully left. This will enable you to hear the effect of the PANNER function.

The PANNER function is available only in algorithms 2, 13, 24, and 26, and always appears in the block before the final AMP function. Consequently, it will always be selected with the F3 soft button, which is labeled “F3 POS” (position).

```

EditProg: F3 POS(PANNER) <> Layer: 1/1
Adjust: 0% Src1 : OFF
Depth : 0% Src2 : OFF
KeyTrk: 0.0%key DptCt1: OFF
VelTrk: 0% MinDpt: 0%
Pad : 0dB MaxDpt: 0%
<more F1 F2 F3 POS F4 AMP more>
    
```

PARAMETER	RANGE OF VALUES
ADJUST	± 100%
KEY TRACKING	± 16%/key
VELOCITY TRACKING	± 200%
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 200%
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 200%
MAXIMUM DEPTH, SOURCE 2	± 200%

The Adjust parameter sets the initial routing of the layer to the upper or lower wire. -100% is the lower, and 100% is the upper. The KeyTrk parameter lets you shift the layer’s sound from one wire to the other based on the MIDI key number of each note. For positive values of KeyTrk, the higher above Middle C, the more sound goes to the upper wire.

The remaining parameters have ranges from -200% to 200%. This lets you start with a sound that’s fully on the lower wire, for example, and shift it completely to the upper wire. The VelTrk parameter shifts notes between wires based on the attack velocity of each note. For positive values, the higher the attack velocity, the more sound goes to the upper wire. The Src1 and Src2 parameters let you assign controls to reroute the sound relative to the initial routing. Setting their depth parameters to positive values will shift the sound to the upper wire when the controls assigned to them approach their maximum values.

**Upper and Lower Amp (AMP U AMP L)**

This two-stage function is similar to the AMP function described above, but it appears in algorithms that have split the signal to two wires and has sent them through different DSP functions in the F2 and F3 blocks. This function enables you to set the final amplitude

independently for each wire, and keeps the two signals separate at its output, giving you added flexibility for mixing and panning. Like the AMP function, UPPER AND LOWER AMP always appears as the last block in an algorithm. Since it's a two-stage function, it has two control input pages. F3 selects the control input page for the lower wire, and F4 the control input page for the upper wire.

```

EditProg: F3 AMP(AMP U/L) <> Layer: 1/1
Adjust: 0dB Src1 : OFF
Depth : 0dB
Src2 : OFF
KeyTrk: 0.00dB/key DptCt1: OFF
VelTrk: 0dB MinDpt: 0dB
Pad : 0dB MaxDpt: 0dB
<more F1 F2 F3 AMP F4 AMP more>
    
```

PARAMETER	RANGE OF VALUES
ADJUST	-96 to 48 dB
KEY TRACKING	± 2.00 dB per key
VELOCITY TRACKING	± 96 dB
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 96 dB
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 96 dB
MAXIMUM DEPTH, SOURCE 2	± 96 dB

**Balance and Amp (BAL AMP)**

This function has a two-wire input and a two-wire output. The parameters on its control input page affect the amount of gain applied to each wire between input and output. A value of 0% applies equal gain to both the upper and lower wires; at a value of 100% only the upper wire's sound will be audible, at -100%, only the lower wire's sound will be audible. This works like the balance control on any stereo system; as the gain increases for one wire, it decreases for the other. It's also similar to the PANNER and XFADE functions. The F3 soft button selects the control input page for the balance stage of this function.

```

EditProg: F3 POS(BAL/AMP) <> Layer: 1/1
Adjust: 0dB Src1 : OFF
Depth : 0%
Src2 : OFF
KeyTrk: 0.0%/key DptCt1: OFF
VelTrk: 0% MinDpt: 0%
Pad : 0% MaxDpt: 0%
<more F1 F2 F3 POS F4 AMP more>
    
```

PARAMETER	RANGE OF VALUES
ADJUST	± 100%
KEY TRACKING	± 16.00% per key
VELOCITY TRACKING	± 200%
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 200%

<b>SOURCE 2</b>	<b>Control Source list</b>
<b>SOURCE 2 DEPTH CONTROL</b>	<b>Control Source list</b>
<b>MINIMUM DEPTH, SOURCE 2</b>	<b>± 200%</b>
<b>MAXIMUM DEPTH, SOURCE 2</b>	<b>± 200%</b>

The AMP stage sets the overall amplitude applied to both wires, and is programmed exactly like the AMP function described above. Their control input pages are almost identical, including the ranges of values. The only difference is that there's no Pad parameter for the AMP stage of the BAL AMP function. The F4 soft button selects the control input page for the AMP stage.

**GAIN**

This function, like AMP, can boost or cut the amplitude of the signal as it passes through the algorithm. Unlike AMP, however, the layer's amplitude envelope doesn't effect the gain settings. GAIN can be used to introduce clipping into a signal, or for adjusting the amplitude of an added waveform. Use a gain function when you want to boost or cut all of a sound's partials uniformly. The control input page for GAIN is identical to the one for AMP.

## Mixers

- +AMP
- +GAIN
- CROSSFADE

The functions in the Mixer category combine two-wire signals in various ways. They have double wires at their inputs, and they mix and amplify the signals from the two wires, then combine them for output to a single wire. Depending on where you assign one of these functions, they can be used to combine two-wire signals for the F4 AMP block, or to enable you to apply another DSP function to the combined signals before the F4 AMP block.

There's a Pad parameter on the control input pages for these functions, which attenuates the lower wire's signal at its input to the function.

**+AMP**

The two input signals to this function are multiplied by .5 (to reduce the likelihood of clipping), then added together. The resulting signal is then multiplied by a gain factor (the combined values for the parameters on the control input page), and multiplied by 2. Any clipping that occurs can be eliminated by lowering the value of the Adjust parameter. If the Adjust value is -6 dB or lower, the signal will never clip. The control parameters are affected by the settings and controls on the AMPENV page.

**+GAIN**

This function operates in almost the same way as +AMP, the only difference being that the signal is not affected by the settings on the AMPENV page, since it occurs before the final AMP block.

**Crossfade (XFADE)**

This function adds the signals from the upper and lower wires after evaluating the combined values of the parameters on its control input page. If those values add up to -100%, only the lower wire's signal is sent to the output. If they add up to 100%, only the upper wire's signal is sent to the output. If they add up to 0%, both signals are attenuated 6 dB, then added and sent to the output.

## Waveforms

SINE  
LOW FREQUENCY SINE  
SAWTOOTH  
LOW FREQUENCY SAWTOOTH  
SQUARE  
LOW FREQUENCY SQUARE

In this category of DSP functions are three standard synth waveforms—Sine, Sawtooth, and Square—with high- and low-frequency variations of each. These are all one-stage functions. They can be assigned in several different positions and combinations in many of the algorithms.

One important fact to keep in mind is that assigning one of these waveforms to a layer's algorithm may remove the original sample from the signal, since they don't have input signals to send to their outputs (they send only the waveform that they generate themselves). If, for example, you were editing the Classical Piano program, and you assigned SINE in the F1 block, you would no longer hear the piano timbre, only the sine wave (unless the signal splits before the F1 block, as in Algorithm 10). Consequently, you'll tend to use these waveforms when you want to build a sound from scratch. If you want to add a waveform to the original timbre of a sound, use one of the added waveform functions described in the next section, or use one of the split signal algorithms.

These waveforms can range in frequency from .1 Hz to 20 KHz. They're not samples like the instrumental sounds and other waveforms; they're generated by oscillators. Since the DSP function waveforms aren't produced by playing back multi-sample keymaps, there are no sample root transitions as you play notes in different keyboard ranges. This makes them especially suitable for use with portamento and wide pitch bend ranges.

Since these waveform functions generate an output signal only, and don't receive an input signal to pass along, the algorithms are arranged so you won't inadvertently assign a series of waveforms that interfere with each other. You'll usually find, for example, that if you can assign a waveform in the F1 block, all subsequent blocks will allow you to assign only the added waveforms. Or, if the subsequent blocks allow you to assign the "regular" waveforms, it's because the wiring of the algorithm is split so that the two waveforms pass through in parallel (as in Algorithm 10).

This next point is a small one, but important, and may make it easier for you to understand the way the waveform functions operate, especially if you've been carefully studying the wiring paths of the algorithms. In several algorithms where the waveforms are available the wiring paths of the algorithms (the horizontal arrows) appear to send a signal to an input of the waveforms. This is *not* the case, and anywhere one of these waveforms is assigned, you should view the algorithm as if there were no horizontal arrow pointing to the left (input) side of the block where the waveform is assigned. The diagrams below will clarify this point. The only difference in the DSP function assignments is in the F1 block, where the first and second diagrams show the SAW waveform, and the third diagram shows the SAW+ added waveform (described in the next section). In the first diagram, the PITCH function's output (passing the sample signal from the keymap) appears to be connected to the input of the F1 block (the SAW function), as well as splitting and passing to the +GAIN function in the F2 block. This is what you would see on the ALG page.

In fact, the actual signal path does *not* pass from the PITCH function through the SAW function; it splits and bypasses the SAW function, as shown in the second diagram. The third diagram shows the same algorithm with the SAW+ added waveform assigned to the F1 block. In this case, the diagram is accurate; the signal passes from the output of the PITCH function, and splits into a two-wire signal. The upper wire passes through the F1 block where the sawtooth wave is added, and into the +GAIN function in the F2 block. The lower wire bypasses the F1 block, and passes directly to the F2 block, where it is combined with the upper wire signal.

Algorithm 24



Algorithm 24



Algorithm 24



The six waveforms in this category are Sine, Sawtooth, Square, Low Frequency Sine, Low Frequency Sawtooth, and Low Frequency Square. The control input pages for all six waveforms affect the pitch of the waveforms. The control input pages for the first three waveforms are identical, as are the control input pages for the three low frequency waveforms are identical.

**SINE, Sawtooth (SAW), SQUARE**

There's only one parameter on this control input page that may still be unfamiliar to you: Fine Hz. We discussed this in Chapter 6. It can tune the pitch of the waveform in terms of its actual frequency in Hertz, as opposed to the usual method of tuning by key names. The advantage to using the Fine Hz parameter is that you can maintain constant beat frequencies across much of the keyboard when you have a program with slightly detuned multiple layers (or multiple waveforms in one layer).

```

EditProg: F1 PCH(SINE) <> Layer: 1/1
Coarse: 0ST Src1 : OFF
Fine : 0ct Depth : 0ct
FineHz: 0.00Hz Src2 : OFF
KeyTrk: 0ct/key DptCt1: OFF
VelTrk: 0ct MinDpt: 0ct
Pad : 0dB MaxDpt: 0ct
<more F1 PCH F2 F3 F4 AMP more>
  
```

PARAMETER	RANGE OF VALUES
COARSE ADJUST	-120 to 60 semitones
FINE ADJUST	± 100 cents
FINE ADJUST IN HERTZ	± 6.00 Hertz
KEY TRACKING	± 2400 cents per key (2 octaves)
VELOCITY TRACKING	± 7200 cents (6 octaves)
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 7200 cents
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 7200 cents
MAXIMUM DEPTH, SOURCE 2	± 7200 cents

If you want the waveform to play with standard twelve-tone octaves, set the KeyTrk parameter to 100 cents per key. Different values for KeyTrk will result in non-standard tunings.



**Low Frequency Waveforms: Sine (LF SIN), Sawtooth (LF SAW), Square (LF SQR)**

These can be used like the waveforms above, since their frequency ranges are similar, but they're intended to be used not for their timbres, but for the shape of their waveforms. By using low frequency values for these waveforms, you're basically getting extra LFOs with very precise control parameters. They're intended to be used as inputs to drive the DSP functions in the subsequent algorithm blocks. They're especially useful with the non-linear DSP functions, such as xAMP.

The parameters on this page affect the pitch of the low frequency waveform in a slightly different manner. They're all tied to the value of the Coarse Adjust parameter, so when you're working with this page, you'll want to set the Coarse Adjust first, then set the values of the other parameters to modify the initial setting. The Coarse Adjust value is multiplied by the values of the other parameters to determine the effect on the pitch, as indicated by the "x" after the parameters' values. More parameter descriptions follow below.

```

|-----| Prog: F1 PCH(LF- SQR) |-----| Layer: 1/1
Coarse: 100.0Hz          Src1 : OFF
Fine : 4.00x             Depth : 1.000x
                          Src2 : OFF
KeyTrk: 2.00x/oct       DptCt1: OFF
VelTrk: 1.000x          MinDpt: 1.000x
Pad : 0dB               MaxDpt: 1.000x
<more> F1 PCH F2 F3 F4 AMP <more>
    
```

PARAMETER	RANGE OF VALUES
COARSE ADJUST	0.1, 1.0, 10.0, 100.0, 1000.0 Hertz
FINE ADJUST	1.00 to 20.00 x
KEY TRACKING	0.1 to 10.0 x per octave
VELOCITY TRACKING	0.010 to 32.000 x
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	0.010 to 32.000 x
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	0.010 to 32.000 x
MAXIMUM DEPTH, SOURCE 2	0.010 to 32.000 x

**Coarse Adjust (Coarse)**

There are only five values to choose from. They're expressed in terms of their frequencies in Hertz. Each value has a frequency ten times higher or lower than adjacent values.

**Fine Adjust (Fine)**

Set this parameter's value to 1.00 x if you don't want it to affect the Coarse Adjust setting. Doubling this value (2.00 x, 4.00 x, etc.) will raise the pitch in octaves. By using Fine Adjust in tandem with Coarse Adjust, you can achieve frequencies from .1 Hz (well below the audible range) to 20 KHz.

**Key Tracking (KeyTrk)**

A value of 1.00 x per octave will keep the waveform's pitch uniform across the entire keyboard. A value of 2.00 x per octave will give you the normal twelve-tone octave. Other values will give you non-standard tunings.

## Added Waveforms

SINE+  
SAW+  
NOISE+

There are three DSP functions that add waveforms to a layer's existing sample: SINE+, SAW+, and NOISE+.

The parameters on the control input page for the SINE+ function affect the pitch of the sine waveform without affecting the pitch of the existing sample. The control input page for the SINE+ function is similar to those for the regular waveforms above. There are parameters for coarse adjust, key tracking, velocity tracking, Source 1 and Source 2, and a pad. There are also parameters for fine adjust and fine Hertz adjust.

The SAW+ function is virtually identical to the SINE+ function; the only difference is in the shape of the waveform.

The NOISE+ function is tied to the level of the sample to which it's added. It will generate fairly white noise (that is, nearly equal amplitude at all audible frequencies) as long as the amplitude of the sample is non-zero. The amplitude of the noise is multiplied by its gain control (the Adjust parameter on its control input page), then added to the signal. To add a short burst of noise at the beginning of a sound, assign ENV2 as the value of one of the Source parameters, then edit ENV2 to produce an envelope with a rapid decay.

The control parameters for NOISE+ are similar to those for SINE+ and SAW+, except that there are no parameters for fine adjust or fine Hertz adjust.

## Non-linear Functions

HIGH FREQUENCY STIMULATOR  
DISTORTION  
SHAPER  
DOUBLE SHAPER  
TWO-PARAMETER SHAPER  
WRAP  
LOWPASS FILTER WITH CLIPPING  
PULSE WIDTH MODULATION

The functions in this category have a variety of effects on the signal. What they have in common is that they can add partials to the signal that were not present at their inputs.

The non-linear functions can produce dramatic changes in timbre, resulting in all sorts of new and modified sounds. One thing to keep in mind is that sounds with a large number of high-frequency partials can be subject to distortion at the high end of the keyboard, especially when you're using more than one of the non-linear DSP functions. You might also hear a bit of *aliasing* with some sounds. Aliasing refers to unintended partials that occur below the fundamental pitch of a sound. The easiest way to remove this distortion or aliasing is to reduce the level of the Adjust parameter on the control input page for whichever non-linear DSP functions you're working with. When you're using PWM followed by DIST or SHAPER, you'd reduce the level of the Adjust parameter for the DIST or SHAPER function. You can also use key tracking (KeyTrk, usually with a negative value), and key tracking in combination with the Keytrack Start (KStart) parameter that's described at the beginning of this chapter.

Even with the damping effects of KeyTrk and KStart, you may come up with sounds that are fantastic in the low range, but gritty in the high range. You can transpose the keymap down to

counteract this, but that's the nature of the non-linear functions. In extreme cases, you can lower the HiKey of the layer to disable the high end completely.

### High Frequency Stimulator (HIFREQ STIMULATOR)

The overall effect of this three-stage function is to boost the high frequency partials of the signal, and depending on the settings of the control inputs, it can add high-frequency partials to the signal as well. It's useful for building sounds that cut through the mix, and have a bright crisp nature.

There's more to the High Frequency Stimulator than meets the eye. It works like this: the signal is run through a high-pass filter, then through a distortion function, then through a second high-pass filter. Finally, it's mixed with the original signal after passing through the final AMP stage of the algorithm. The three control input pages let you adjust the cutoff frequency of the first high-pass filter (F1 FRQ), the amount (drive) of the distortion function (F2 DRV), and the mix (relative amplitude) of the stimulated signal with the original (F3 AMP).

```

|E3|UProg:F1 FRQ(HIFREQ STIM) <> HWER1/1
Coarse:C 4 262Hz Src1 :OFF
Fine :0ct Depth :0ct
KeyTrk:0ct/key Src2 :OFF
VelTrk:0ct DptCt1:OFF
Pad :0dB MinDpt:0ct
MaxDpt:0ct
<more F1 FRQ F2 DRV F3 AMP F4 AMP more>
    
```

PARAMETER	RANGE OF VALUES
COARSE ADJUST	C -1 16 Hz to G 10 25088 Hz
FINE ADJUST	± 100 cents
KEY TRACKING	± 250 cents per key
VELOCITY TRACKING	± 10800 cents
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 10800 cents
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 10800 cents
MAXIMUM DEPTH, SOURCE 2	± 10800 cents

```

|E3|UProg:F2 DRV(HIFREQ STIM) <> HWER1/1
Adjust:0dB Src1 :OFF
Depth :0dB
KStart:C -1 uniPola Src2 :OFF
KeyTrk: 0.00dB/key DptCt1:OFF
VelTrk:0dB MinDpt:0dB
MaxDpt:0dB
<more F1 FRQ F2 DRV F3 AMP F4 AMP more>
    
```

PARAMETER	RANGE OF VALUES
ADJUST	-96 to 48 dB

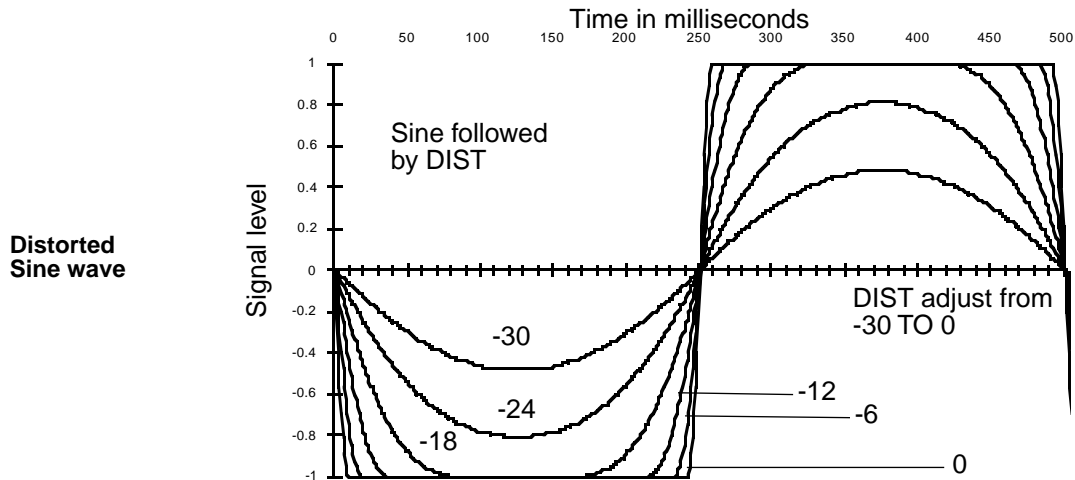
KEYTRACK START	C -1 to C 9 unipolar, C -1 to C 9 bipolar
KEY TRACKING	± 2.00 dB per key
VELOCITY TRACKING	± 96 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 96 dB
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 96 dB
MAXIMUM DEPTH, SOURCE 2	± 96 dB

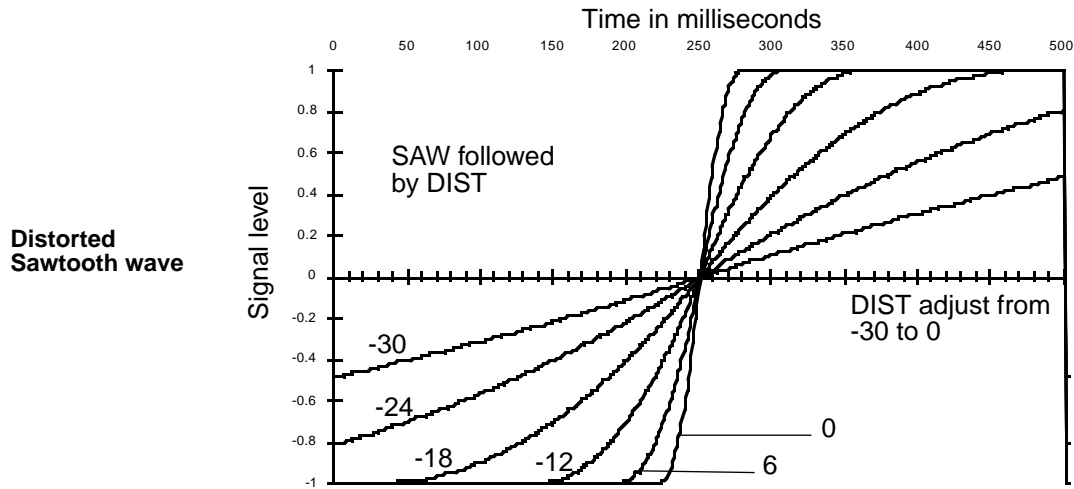
```

EditProg: F3 AMP(HI)FRQ SIMUL<>LAYER1/1
Adjust: 0dB Src1 : OFF
Depth : 0dB Src2 : OFF
KeyTrk: 0.00dB/key DptCt1: OFF
VelTrk: 0dB MinDpt: 0dB
MaxDpt: 0dB
<more F1 FRQ F2 DRU F3 AMP F4 AMP more>
    
```

PARAMETER	RANGE OF VALUES
ADJUST	-96 to 48 dB
KEY TRACKING	± 2.00 dB per key
VELOCITY TRACKING	± 96 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 96 dB
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 96 dB
MAXIMUM DEPTH, SOURCE 2	± 96 dB

**Distortion (DIST)**





Using this function is much like overdriving an ordinary keyboard or guitar amplifier. The input signal is multiplied by the gain control (the combined values of the parameters on its control input page labeled DRV, for Drive), then passes into a distortion mapper. Large Adjust values will cause serious amounts of distortion.

Different sounds are affected differently by DIST. Waveforms that are static (waveforms with shapes that repeat regularly and are not evolving) when they enter the DIST function will undergo more of a timbre change than the familiar sound of distortion.

The DIST function distorts each note separately, unlike a fuzz box, which adds several notes together then applies a uniform amount of distortion to all of them. Consequently your power chords may sound a little different from your expectations, but you can also get some great effects with key and velocity tracking (not to mention Sources 1 and 2!) that aren't possible with other distortion devices.

The page below shows the DIST function in the F1 block, but it can appear in other blocks as well.

```

EditProg: F1 DRU(DIST) <> Layer: 1/1
Adjust: 0dB Src1 : OFF
Depth : 0dB
KStart: C -1 unipola Src2 : OFF
KeyTrk: 0.00dB/key DptCt1: OFF
VelTrk: 0dB MinDpt: 0dB
Pad : 0dB MaxDpt: 0dB
<more F1 DRU F2 F3 F4 AMF more>
    
```

PARAMETER	RANGE OF VALUES
ADJUST	-96 to 48 dB
KEYTRACK START	C -1 to C 9 unipolar, C -1 to C 9 bipolar
KEY TRACKING	± 2.00 dB per key
VELOCITY TRACKING	± 96 dB
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 96 dB
SOURCE 2	Control Source list

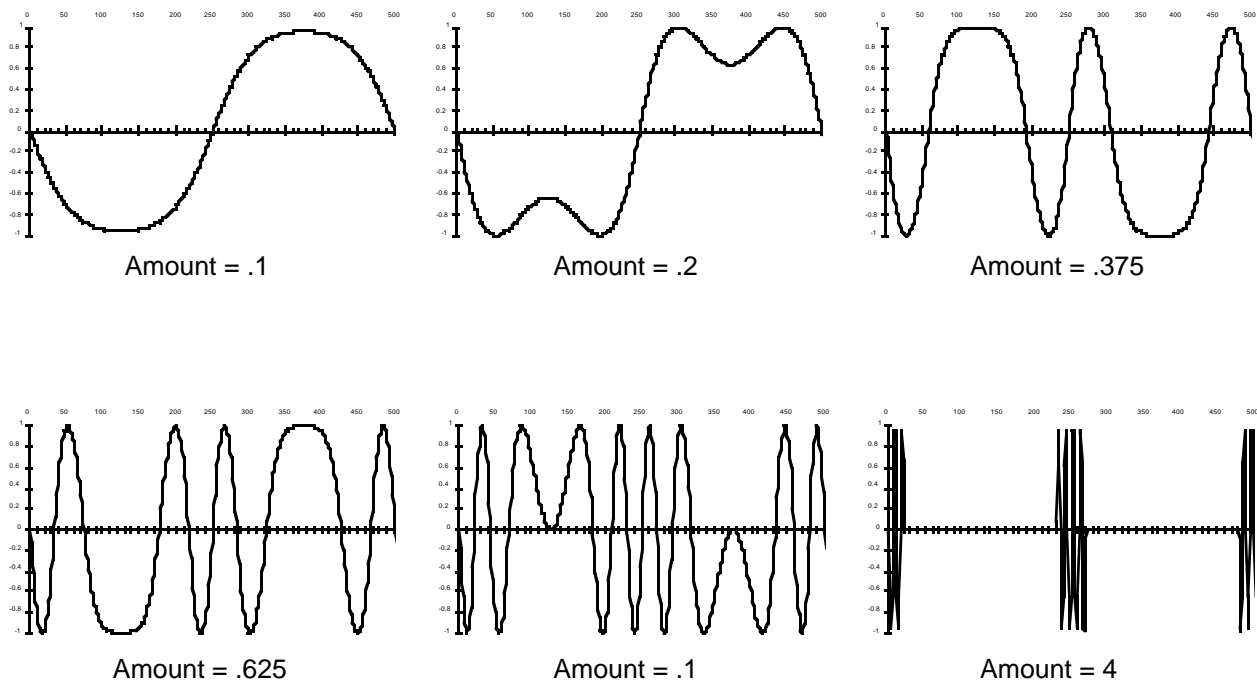
**SOURCE 2 DEPTH CONTROL**  
**MINIMUM DEPTH, SOURCE 2**  
**MAXIMUM DEPTH, SOURCE 2**

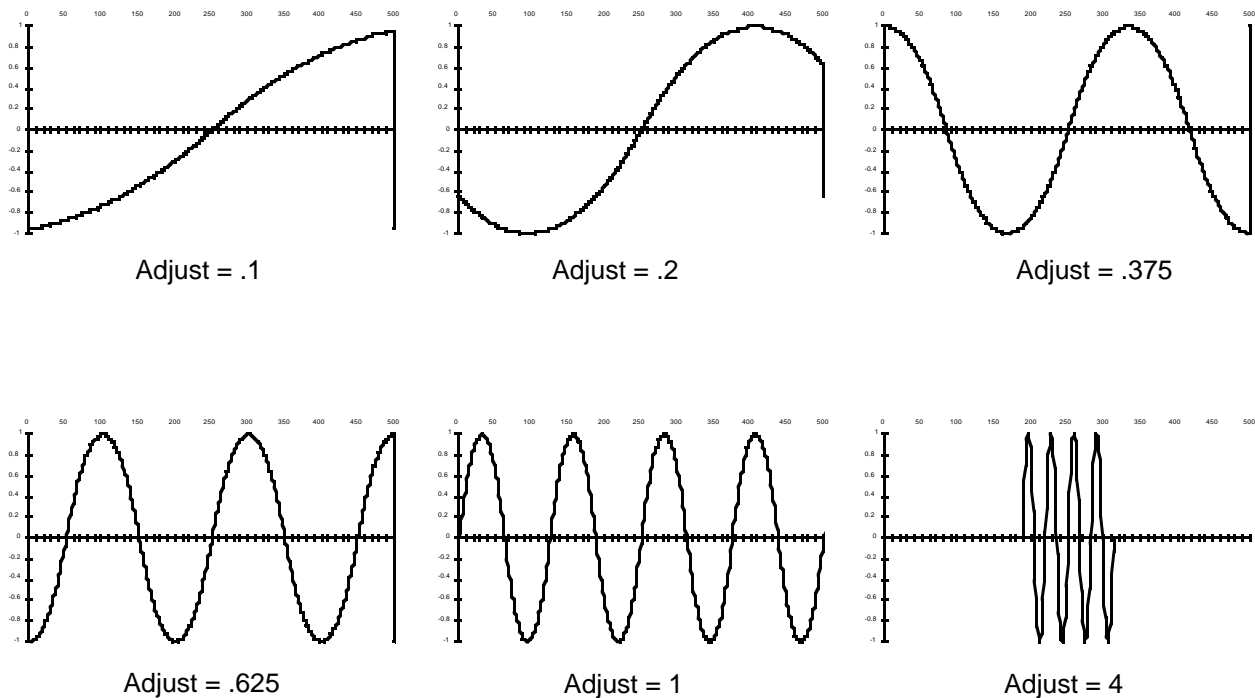
**Control Source list**  
**± 96 dB**  
**± 96 dB**

### SHAPER

The effect of SHAPER can be very unpredictable, and the mechanics of its operation lend themselves toward explanations that are more numerical than verbal. The best way for you to get a feel for the SHAPER is to start with single-cycle waveform keymaps and experiment with different values for the parameters on its control input page (labeled AMT, for Amount), and listen to the results. SHAPER tends to work best with the single-cycle waveform sounds (keymaps with IDs 112—167), and is usually less effective with acoustic instrumental sounds. SHAPER often produces numerous peaks throughout the frequency range, even at frequencies that didn't have much amplitude to begin with. These peaks can sound like resonant filters, and can even sound voice-like.

The two series of graphs that follow show the effect of SHAPER on two typical single-cycle waveforms. The first set of six graphs just below shows the evolution of a sine wave input as the value of the Amount parameter is increased. The following set of six graphs shows the effect when the Adjust parameter is increased. Each graph plots a 500-millisecond segment of waveforms cycling at frequencies of 2 Hz. Of course, these are just a few of the countless modulations you can apply to different waveforms at different frequencies.





As the SHAPER receives input signals, it evaluates the signal's level according to its own internal scale. When the SHAPER's Adjust value is at .25, an input signal moving from negative full scale to positive full scale (a sawtooth) will map to an output curve with a single-cycle sine wave shape. At an adjust value of .5, the same input signal would map to a 2-cycle sine wave output signal. Adjust values of .75 and 1.0 for the SHAPER would map to 3- and 4-cycle sine wave output signals, respectively. Beyond values of 1.0, some portions of the output will pin at zero-scale.

Small Adjust values for the SHAPER can sound much like the DIST function, but larger values will introduce dramatic changes in timbre, while DIST will have a less pronounced effect on timbre.

The following LCD page shows the SHAPER function in the F1 block, but it can appear in other blocks as well.

```

EditProg: F1 AMT (SHAPER) <> Layer: 1/1
Adjust: 0.100x Src1 : OFF
Depth : 0.00x
KStart: C -1 unipola Src2 : OFF
KeyTrk: 0.000x/key9 DptCt1: OFF
VelTrk: 0.00x MinDpt: 0.00x
Pad : 0dB MaxDpt: 0.00x
<more F1 AMT F2 F3 F4 AMP more>
    
```

PARAMETER	RANGE OF VALUES
ADJUST	0.100 x to 4.000 x
KEYTRACK START	C -1 to C 9 unipolar, C -1 to C 9 bipolar
KEY TRACKING	± 0.200 x
VELOCITY TRACKING	± 4.00 x

<b>PAD</b>	<b>0, 6, 12, 18 dB</b>
<b>SOURCE 1</b>	<b>Control Source list</b>
<b>SOURCE 1 DEPTH</b>	<b>± 4.00 x</b>
<b>SOURCE 2</b>	<b>Control Source list</b>
<b>SOURCE 2 DEPTH CONTROL</b>	<b>Control Source list</b>
<b>MINIMUM DEPTH, SOURCE 2</b>	<b>± 4.00 x</b>
<b>MAXIMUM DEPTH, SOURCE 2</b>	<b>± 4.00 x</b>

The values for each of the parameters on the SHAPER’s control input page are expressed in arbitrary quantities that represent a multiplication of the amount of shaping applied.

**Double Shaper (SHAPE2)**

This is simply a series of two SHAPERS. The first is programmed exactly like SHAPER. The values of the control parameters for the second are fixed at .75 times those of the first. This can produce effects that the single SHAPER can’t. If, for example, you set the Adjust parameter of SHAPE2 to 1.000, it will process the input signal with a value of 1.000, then again with a value of .75. This is not the same as processing the input signal once with an Adjust value of 1.75.

**Two-Parameter Shaper (2PARAM SHAPER)**

This function is similar to the SHAPERS described above, but it has two control input pages instead of one. The F1 EVN control parameters enable you to add distortion to sine wave partials that are even harmonics of the input signal, and the F2 ODD control parameters let you add distortion to sine wave partials that are odd harmonics of the input signal.

In simpler terms, the control parameters behave like those of the regular SHAPER, but they can shape the signal about six times more than SHAPER can. 2PARAM SHAPER works by separately multiplying the input signal by the combined values of the two sets of control parameters, adding the resulting signals, and multiplying that sum by a constant, then wrapping the signal values that exceed positive or negative full scale (see the WRAP function below).

Experimentation is the key here. Start with very low values for each of the Adjust parameters, and increase them until you begin to hear an effect. Some values will create a DC offset in the signal—that is, the signal won’t oscillate around the normal zero-point of the scale, but will be shifted toward positive or negative full scale. This may cause a click or thump in sounds with rapid attacks, decays, or releases. To reduce the click or thump, you can edit the AMPENV to produce a more gradual envelope.

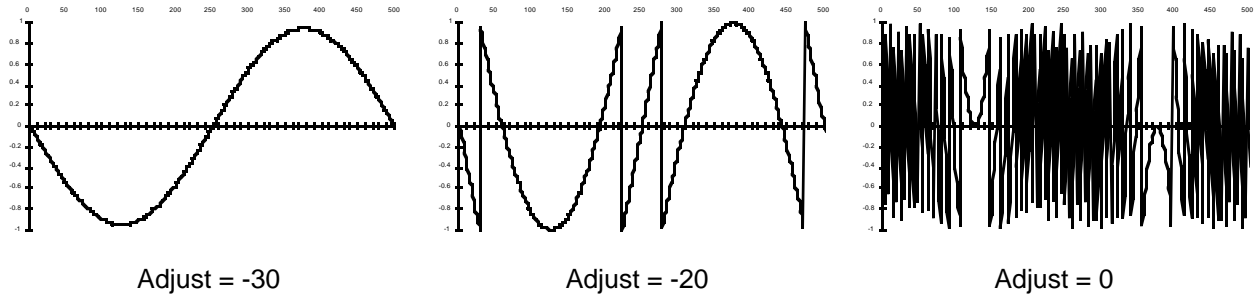
2PARAM SHAPER works best with the single-cycle waveform keymaps (IDs 112—167).



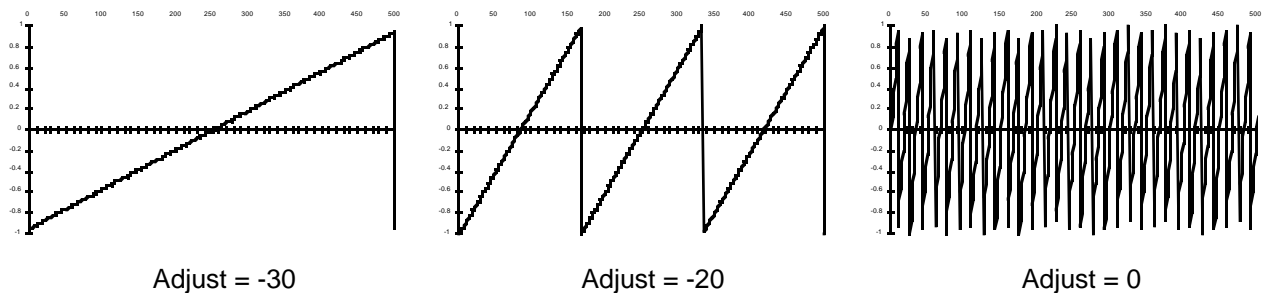


### Waveform Wraparound (WRAP)

The next three graphs show the effect of various amounts of WRAP on a sine wave.



The following three graphs show the effect of WRAP on a sawtooth wave at the same frequency.



With this function you can completely mutilate a sound, and with large amounts of wrap, turn anything into white noise. At the input of the WRAP function, the signal is multiplied by the combined values of the parameters on the WRAP control input page, then multiplied by an additional gain factor of 30. If the resulting value is greater than full scale (in other words, if it's sufficiently high to clip), then instead of clipping, the waveform "wraps" back around to negative full scale, and it continues to evolve from that point. Likewise, if the resulting value is less than negative full scale, it wraps to positive full scale and proceeds from there. For any waveform, several of these wraparounds can take place before the waveform fits into the allowable range.

You'll want to try different values of the Adjust parameter to get a feel for the results of different amounts of wraparound. Look for the value that introduces a very slight amount (it will tend to be well below 0). The sound will start to buzz here and there, as a few segments of the input wrap around. As you increase the Adjust value, the buzz will increase, and the pitch of the sound will begin to disappear. Keep adding to the Adjust value, and you'll end up with white noise, regardless of the starting timbre.

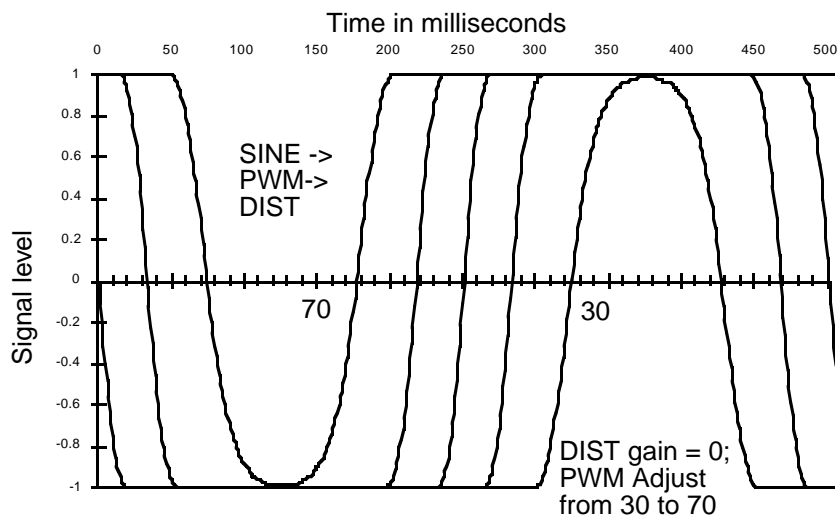
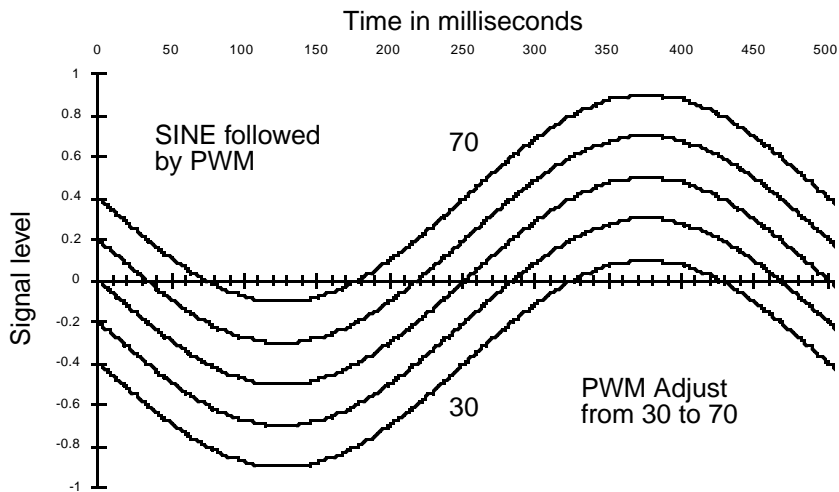
The bright buzzy nature of wrapped sounds is due to the discontinuities in the waveforms of the partials as they wrap around from positive full scale to negative full scale and vice versa. If you want, you can reduce or eliminate the buzz by sending the signal through SHAPER after it goes through WRAP (assign SHAPER as the DSP function in the following algorithm block). Set the SHAPER's Adjust parameter to .25. This will map both positive full scale and negative full scale amplitudes to a level of zero, eliminating the wraparound discontinuities, but preserving the alterations in the waveform produced by WRAP. SHAPER will add its own effects as well.

The control input page for WRAP uses the same set of parameters and ranges of values as DIST.

### Lowpass Filter with Clipping (LPCLIP)

This is a one-pole filter, which is programmed just like LOPASS. The difference with LPCLIP is that the amplitude of the input signal is multiplied by 4 before the filter. This can cause the signal to clip, which can produce interesting results. Naturally, you'd use this function only when you want to induce clipping intentionally as a component of your sound.

### Pulse Width Modulation (PWM)



Pulse width modulation can produce some classic synth sounds, and can break new sonic ground as well. Its operation is very simple; it adds an oscillating DC offset to the input signal (shifts it toward positive or negative full scale). Unless it causes the signal to clip, it won't have much effect on the signal. It's designed to be followed by DIST, with the Adjust parameter for DIST set to a fairly high level. The DIST function will drive all positive elements of the signal toward positive full scale, and negative elements toward negative full scale. The result is a rectangular wave with a width that varies according to the Adjust level of PWM.

PWM won't affect a square wave input. Sine and triangle waves will produce familiar PWM sounds. More complicated waveforms will result in discontinuous rectangular waveforms.

You can also follow a PWM algorithm block with SHAPER, since SHAPER's output is affected by the DC level of the signal.

The parameters on the PWM control input page affect the DC offset of the signal, in terms of the percentage of shift from no offset to maximum offset. At a value of 0%, there is an offset positive full scale. At 100%, the offset is negative full scale, and at 50%, there is no offset.

A typical control configuration for PWM has the Coarse Adjust set to 50%, an LFO assigned to Src1, and Src1's Depth parameter set to a value of 25%.

The page below shows the PWM function in the F1 block, but it can appear in other blocks as well.

```

EditProg: F1 WID(PWM) Layer: 1/1
Adjust: 50% Src1 : LFO1
Depth : 25%
Src2 : OFF
KeyTrk: 0.0%/key DptCt1: OFF
VelTrk: 0% MinDpt: 0%
Pad : 0dB MaxDpt: 0%
<more F1 WID F2 F3 F4 AMP more>
    
```

PARAMETER	RANGE OF VALUES
ADJUST	0 to 100%
KEY TRACKING	± 8.00% per key
VELOCITY TRACKING	± 100%
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 100%
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 100%
MAXIMUM DEPTH, SOURCE 2	± 100%

## Waveforms Combined with Non-linear Functions

- ADDED SAWTOOTH WITH NON-LINEARITY (SW+DST)
- ADDED SAWTOOTH PLUS SHAPER (SW+SHP)
- SHAPE-MODULATED OSCILLATOR
  - x SHAPE MOD OSC
  - + SHAPE MOD OSC
- AMP-MODULATED OSCILLATOR

The six functions in this category do one of two things: they combine samples and waveforms with a non-linear DSP function, or they use a waveform or sample as inputs to drive non-linear functions.

### Added Sawtooth Wave with Non-linearity (SW+DST)

This function starts by adding a sawtooth wave to the layer's sample input. When the input signal from the sample is added to the sawtooth, the signal may exceed full scale, so a wraparound function similar to WRAP is performed. The result is then squared to remove any

discontinuities from the wraparound. The resulting signal has a large DC offset, so a constant of  $3/8$  is subtracted.

The parameters on the control input page for SW+DST control the pitch of the sawtooth wave.

### **Added Sawtooth Wave Plus SHAPER (SW+SHP)**

For this function, the sample input is combined with a sawtooth wave, then passed into the SHAPER function. The SHAPER has a constant Adjust value of .25. First, the sample input is multiplied by a constant, which may cause it to clip. Any clipping becomes part of the signal. This result is added to the sawtooth wave, which may cause the waveform to exceed full scale. If it does, the signal wraps around as in the WRAP function. This result is then put through the SHAPER.

You may want to use the Pad parameter with this function, depending on the sample you use. Pastorius fans should try this function using one of the Electric Bass keymaps. Use Algorithm 8, and start with the second and third algorithm blocks set to NONE. Set the fourth block to SW+SHP. Try setting the Keymap transposition to -12, and the SW+SHP key tracking to 100 cents per key.

### **Shape-modulated Oscillator (SHAPE MOD OSC)**

This function combines the sample input with a sine wave at  $1/4$  scale, does a combination of additions and multiplications on both signals, then puts the result through a SHAPER function. The amount of shaping depends on the levels of both the input signals. First, the SINE value is multiplied by the sample input value, then multiplied by a constant—any samples exceeding full scale will wrap around. The result is added to the wrapped product of the SINE value times a constant. The entire resulting waveform is then passed through the SHAPER, whose Adjust value is set by the level of the sample input. You might think of this function as an oscillator whose shape is controlled by the sample input signal.

Even if the depth value goes to zero, there will still be a non-zero final output, due to the addition of the multiplied SINE value. In this case, the output would be a slightly distorted sine wave. As the value of depth increases, the harmonic content of the output signal will rise, assuming the pitches of the sample input and the sine wave are simply related—unison, octaves, etc. Slight detuning between the two pitches will cause a slow beat frequency.

The parameters on the F2 PCH control input page affect the pitch of the sine wave. Those on the F3 DEP page affect the level of the sample input, and consequently the amount of SHAPER that's applied. If the DEPTH values exceed +5 dB, then the product of SAMPLE INPUT  $\times$  DEPTH may clip, adding further harmonics through a mechanism that differs from the addition of harmonics below the +5 dB DEPTH level.

### **x SHAPE MOD OSC**

Available only in Algorithm 18, this function is similar to SHAPE MOD OSC, except that it multiplies its two input signals and uses that result as its input.

### **+ SHAPE MOD OSC**

Also available only in Algorithm 18, + SHAPE MOD OSC is similar to x SHAPE MOD OSC, except that it adds its two input signals and uses that sum as its input. With this and all the modulated oscillators, let your ears be your guide.

### **Amp Modulated Oscillator / Final Amp (AMP MOD OSC)**

This function is available only in Algorithm 17. The sample input is multiplied by the output of a sine wave oscillator. This result is scaled by the parameters on the F3 DEP control input page, and the result is added to the original sample input and sent to the final AMP function. The parameters on the F2 PCH control input page affect the pitch of the sine wave, and consequently all subsequent results.

## Mixers with Non-linear Inputs

x AMP  
x GAIN  
! AMP  
AMPLITUDE MODULATION

### x AMP

This function can be used in the final algorithm block when it mixes two input wires into a single output. The two input signals are multiplied. The control input parameters affect the gain of the multiplied signals. The final amplitude is also affected by the settings for AMPENV and ENVCTL. Multiplying the two signals can result in outputs that differ dramatically from the input signals. You can get a wide variety of effects from this function, for example, turning an acoustic sample and a waveform (or two waveforms) into a sound that has little resemblance to the input sounds.

When two signals are multiplied, the resulting signal consists of the sums and differences of the frequencies of each partial of each signal. The frequencies of the original signals do not come through, unless they have one or more DC components (non-oscillating partials). And of course, if one of the signals has zero amplitude, the resulting signal also has zero amplitude.

If the fundamental frequencies of the two input signals are related by simple fractions (that is, if the ratios between their frequencies are something like  $1/1$ ,  $2/1$ ,  $3/1$ ,  $4/1$ ,  $1/2$ ,  $1/3$ ,  $1/4$ ,  $2/3$ ,  $3/2$ ) the resulting signal will be a harmonic sound. Its partials will be multiples one of the original fundamentals, or possibly a new fundamental. If the frequency ratios of the original signals are nearly but not quite one of these fractions, some beat frequencies will be perceived, which may or may not be useful. Of course, with equal temperament, the ratios given above are not perfectly precise (a perfect fifth, for example, has a frequency ratio of 1.4983, not  $3/2$ ). If the frequencies of the original signals are not at least closely related, the result of X AMP will be, shall we say, less than harmonious.

If the frequency of one of the original signals is below the audible range, then the result of X AMP is not a matter of harmony, but of amplitude. In this case, a tremolo effect (amplitude modulation) would be heard, because the resulting signal would periodically dip below the audible range. In fact, when you're using X AMP in the final algorithm block, you can use any sample as an LFO source by setting the Adjust parameter on the PITCH page to its minimum. To make this work, the algorithm must use one of the waveform functions in one of the blocks, and the sample signal must be routed to the x AMP block. Results will vary.

### x GAIN

This function operates like X AMP, except that it is not affected by the settings for AMPENV, since it occurs before the F4 AMP block.

### SHAPER / Final Amp (! AMP)

This function also appears in the final algorithm block when it mixes two input wires to a single output wire. The two inputs are added, then put through the SHAPER function with a fixed Adjust value of .25, then amplified according to the values for the parameters on the F4 AMP control input page.

### Amplitude Modulation (AMP MOD)

The AMP MOD function multiplies its two input signals, and the result is multiplied by a gain value that is determined by the parameters on the AMPMOD's control input page. This result determines the balance between the upper and lower wires. AMP MOD can clip the signal, so you may need to use the Pad parameter.

## Hard Sync Functions

### SYNC M AND SYNC S

These two functions appear in Algorithms 26—31, and always work in tandem. Each is a rising sawtooth oscillator. SYNC M is the “master” waveform, and SYNC S is the “slave.” These terms stem from the fact that the pitch (frequency) of the master waveform determines the repetition rate, and thus the shape, of the slave’s waveform. These functions generate their own waveforms, and do not pass the sample input through the algorithm. Consequently the PITCH function does not appear for these algorithms.

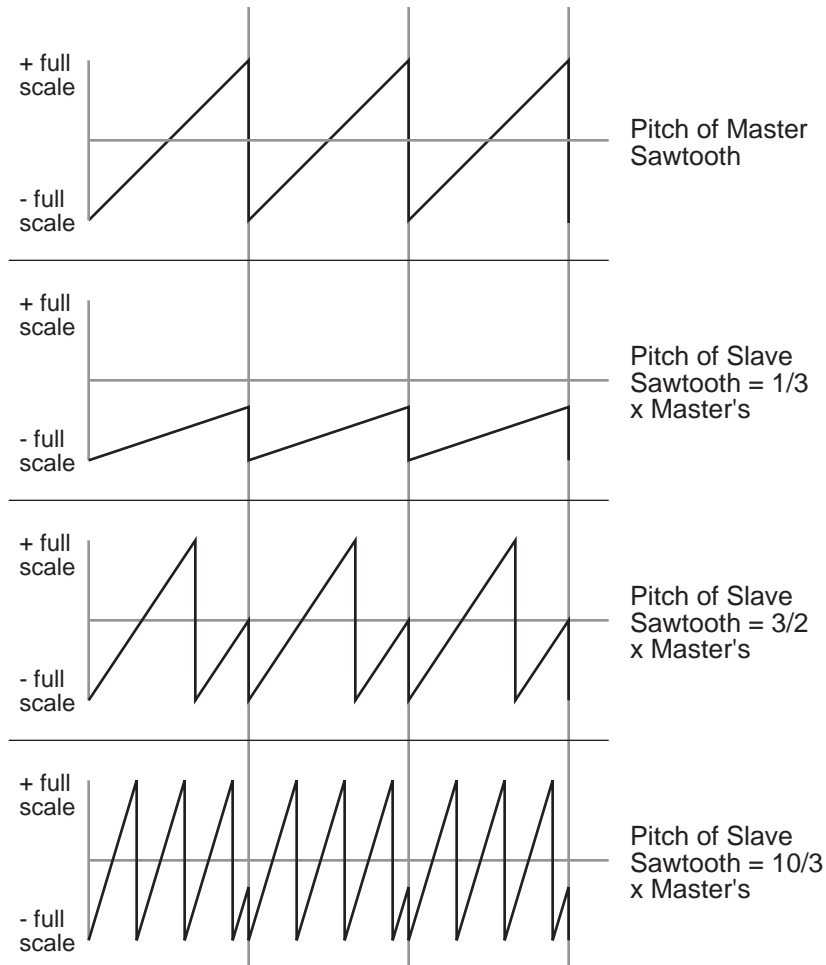
Every time the master waveform falls from positive full scale to negative full scale, the slave waveform is forced to negative full scale. You can create a wide variety of timbres by adjusting the pitches of the waveforms relative to each other. This is done with the parameters on the F1 PCH and F2 PCH control input pages. F1 is for the master, and F2 is for the slave. Pitch control is really a bit of a misnomer for the slave waveform, because its pitch is determined by the pitch of the master waveform. The fundamental of the slave waveform is forced to be the same as that of the master, since they always have the same frequencies, although the shapes of their waveforms differ.

To clarify this, assume for now that the pitch of the master waveform remains constant. When you trigger a note, both waveforms start at negative full scale. If the slave’s “pitch” control is set to a much lower value than that of the master, the master waveform will reach positive full scale before the slave. So the shape of the slave waveform will be that of a more slowly rising sawtooth wave with a relatively large negative DC offset (most of the waveform will be in the negative portion of the scale).

If the slave’s “pitch” control is set to a value only slightly lower than that of the master, the waveforms will be very similar, and the slave waveform will have a small negative DC offset. When the pitch settings are identical, the waveforms are identical.

If the slave’s pitch setting is higher than that of the master (which usually gives more interesting results), the slave’s waveform will alternate between a complete sawtooth cycle and a fraction of the subsequent cycle. At twice the frequency of the master, the waveform will have twice the frequency, and only the even harmonics of the master frequency will be pronounced. When the slave/master frequency ratio is nearly, but not exactly three, all harmonics will be present, but the 3rd, 6th, 9th (all multiples of 3) harmonics will be much louder than the others. This will sound like a resonant filter with multiple resonance peaks.

Because the pitch of the slave waveform is forced to be nearly that of the master, you can adjust the key tracking of the slave to values less than 100 cents per key without affecting the pitch. This will help reduce some of the harshness at the high end of the keyboard.



## DSP Functions

---

Hard Sync Functions



# Chapter 15

## Sampling and Sample Editing

### Setting Up For Sampling

Before you begin sampling, you'll need to connect the proper cables from your sample source to your K2vx. The cables and input jacks you use depend on the sample format you choose, and the output configuration of your sample source. Note that sampling requires the K2vx Sampling Option.

#### Cables and Input Jacks

If you're going to be sampling from an analog source, connect a standard 1/4-inch mono or stereo cable (a typical guitar cable) from the output of your sample source to the 1/4-inch analog input jack on the K2vx. If you have a K2vxR or K2vxRS, use the 1/4-inch jacks if you're sending an unbalanced signal, or the XLR jacks on the front panel if you're sending a balanced signal. Although it is possible to send a balanced signal on a 1/4-inch cable, avoid sending a balanced signal to the 1/4-inch jack when you're making stereo samples, since doing so can cause phase cancellation in your signals.

Using a mono cable will send the signal to the K2vx's left channel. If you use a mono cable, be sure to set the Mode parameter on the Sample mode page to a value of Mono(L).

If you're using a digital sample source, you can use either a coaxial cable or an optical cable, depending on the output format of your sample source. The coaxial input on the K2vx accepts a standard male XLR fitting. The K2vx's XLR input is configured as follows: Pin 1 is Common, Pin 2 is High, and Pin 3 is Low. Depending on your sample source (a commercial DAT deck, for example), you may need to use a cable (or a cable and adaptor) with an RCA connector on one end and an XLR connector on the other. In this case, you should tie the Common wire to the Low wire at the RCA end.

If your digital sample source has an optical output, connect your cable to the optical input jack above the coaxial jack on the K2vx's rear panel. This jack is covered by a small plug which is easily removed. This plug should be kept in place whenever the optical input is not in use, since dust or dirt can cause the optical input to malfunction.

#### Entering The Sampler - Two Different Ways

There are two different methods of entering the Sampling page. Which method you choose depends on the type of sampling you are doing - how many samples you are making and if you need custom keymaps.

The difference between the two methods primarily concerns the ease of accessing the keymap editor. Once you have made your samples, you will need to have them assigned to a keymap and have that keymap assigned to a layer in a program. Refer to the section entitled "Building a Keymap" on page 15-37, for a step by step explanation of how to create keymaps.

##### From Master Mode

The simplest way to enter the Sampling page is from Master Mode. Press the soft button labelled **Sample** on the Master page. This is a good method to use if you are making only a couple of samples, or if you want to assign each sample to its own keymap and program. Once you have created and saved your sample, you can press the **Preview** soft button. This button will allow you to quickly create a program and keymap, with that sample assigned across the

---

entire range of the keyboard. The program is a one layer program which uses the settings from the Default program 199.

#### From the Keymap Editor

This is a better method to use if you are going to be doing lots of multi-sampling, or if you need to create custom keymaps in which you have your new samples assigned across the keyboard in one keymap. Call up program 199, Default Program. Press **Edit**, then **Keymap**. Select Keymap 168, Silence, then press **Edit** again. This brings you to the Keymap editor. (In fact you can choose any program and keymap you want to start with, but by choosing these, you are starting with a "blank slate".) Now from the Keymap editor, press the MIDI Mode button. This will jump you to the Sampling page. Once you have created and saved your samples, press Exit. You will now return to the Keymap Editor page, where you can immediately assign those samples across the keyboard. Once you have created and saved your keymap, you can either exit the Keymap editor and create a program which uses your new keymap, or you can return to the Sampling page for another round of sampling

## Sampling Analog Signals

The K2vx's analog sampling input is optimized for a low-impedance line level signal (-10 dBm). With a line level signal, an input gain setting of 0 dB should prevent any clipping of the sample even at maximum output from the source. You can compensate for lower input levels with the Gain parameter on the Sample mode page.

If you're sampling through a microphone, you'll probably want to use a preamp to optimize your signal-to-noise ratio. If you don't have a preamp, you can adjust the Gain parameter. A setting of 21 dB will give you reasonable results for many applications. This will increase the noise level as well, however.

Running your sample signal through a mixer before sending it to the K2vx will give you the most flexibility in controlling your signal level, since you can use its gain or pad if needed. This may add noise to the signal, however. For the cleanest possible signal, you'll want to connect your sample source directly to the K2vx. The best results will be achieved by sampling from a digital source, using one of the K2vx's digital sample inputs.

Assuming your connections are made, you're ready to set up your first sample recording. Select the Sample mode page (refer to "Entering The Sampler - Two Different Ways," above). The top line of the sample mode page gives you the amount of free sample memory, and the amount of free program memory.

### Input

On the Sample mode page, you'll set the conditions for your sample recording. Depending on the input type you select, a different set of parameters will appear on this page. When you've selected analog input, the page appears as in the diagram below. The differences between analog and digital sampling are discussed in the section called "Sampling Digital Signals."

```
SampleMode Samples:8192K Memory:752K
Sample:None Src:Int
Input :Analog Time :1s Mon:Off
Gain :0 dB
Rate :44.1KHz
Mode :Mono(L)
Thresh:Off -dB 60 40 * 16 * 8 4 0
Record Auto Timer Preview
```

The digital meters at the lower right of the display give a good indication of your sample level. When you send a signal from your sample source, you should see the meters respond.

### Src (Rack models only)

The K2vxR and K2vxRS have an additional parameter related to input: Src. The possible values for the Src parameter are Internal(Int) or External (Ext). Choose a value of Ext when you want to sample the signal from an external source that's connected to one of the K2vx's sampling inputs. Use a value of Int if you want to sample the K2vx's own output. (This can be done on the keyboard models of the K2vx by selecting Analog input and removing any cable connected to the analog sampling input). For sampling the K2vx's own output, the Monitor parameter must be Off to prevent signal attenuation.

### Gain

The meters are calibrated in -dB units. A level of 0 dB indicates the maximum signal without clipping. The sample will be free of clipping as long as the meter levels don't exceed 0 dB. For optimum results, you should adjust the K2vx's Gain parameter (or the gain from your sample source) so that the signal stays below 0 dB. Otherwise, the signal will be clipped, causing the loss of sample data, and usually resulting in audible distortion of the resulting sample. A few clips (fewer than 100) may not cause any appreciable distortion. You'll get the best signal-to-noise ratio with meter levels as close to 0 dB as possible, although you'll find that samples with maximum meter readings as low as -12 dB can sound remarkably noise-free.

The relatively slow LCD output of the meter levels cannot show every peak in the incoming signal, therefore, you won't necessarily see every transient in every sample you take. You will be able to see any transient that is clipped, however, since whenever a clip occurs, the K2vx will display the word "CLIP" above the meters, and will flash the Master mode LED. It will also give you the number of clips in each sample before you save it.

The meters are inoperative during actual sampling, so make a few tests of your levels before you begin to record.

### Rate

After you've set your levels, you need to select the sample rate. You have four rates to choose from. The trade-offs that determine your best sampling rate are frequency response and storage requirements. Higher sample rates will capture more frequency content from your samples, but will take up more memory. Lower rates give you more sample time, but don't give the same frequency response as higher rates. Rates of 29.4 or 32 KHz will yield a flat response up to about 14 and 15 KHz, respectively. 44.1 and 48 KHz yield a flat response up to 20 KHz, which is the upper limit of audibility for most humans. The lower rates may be adequate for most sounds, since many sounds have little content above 15 KHz. Sounds with a great deal of high-frequency content, such as cymbals, should probably be sampled at the higher rates. You can save memory by using lower sample rates for sounds without much high-frequency content—acoustic or electric bass, for example.

Another consideration in selecting sample rate is the K2vx's transposition range during sample playback. The K2vx transposes samples by changing the sample playback rate; the higher the playback rate, the higher the pitch of the sample. The K2vx can achieve a maximum sample playback rate of 96 KHz. A sample made at 48 KHz can be transposed up only one octave, since the playback rate doubles for every octave of upward transposition. A sample made at 29.4 KHz can be transposed up approximately 21 semitones (an octave and a sixth). There is no limit on downward transposition, regardless of the sample rate.

Each portion of a sample (each individual sample element made by the K2vx during the sampling process) takes up two bytes of sample memory. A one-second stereo sample at 48

KHz consists of 96,000 individual samples (48,000 x 2), taking up 192,000 bytes (about 188K) of sample memory. The same sample taken at 32 KHz takes up about 125K. A one-second mono sample taken at 32 KHz takes up about 63K.

If you plan to do a lot of sampling, you may want to consider adding sample memory to your K2vx. SIMMS (Single In-line Memory Modules) are available at your dealer, or at most computer stores or mail-order houses. Be sure to read "Choosing SIMMs for Sample RAM" in Chapter 8 of the *K2vx Reference Guide* before you go SIMM shopping, though.

At a sampling rate of 44.1 KHz, each megabyte you add increases your sample time by about 12 seconds. The chart below lists a few standard sample RAM configurations and their total sample time capacity (in seconds) at various sample rates.

		Sampling rate in KHz			
		29.4	32.0	44.1	48.0
2 x 1M	Mono	35	32	23	21
	Stereo	17	16	11	10
2 x 4M	Mono	140	128	92	84
	Stereo	70	64	46	42
2 x 16M	Mono	560	512	368	336
	Stereo	280	256	184	168

### Mode

Use the Mode parameter to select mono or stereo sampling. (Keep in mind that stereo samples take up twice as much memory as mono samples.) Use a value of Mono(L) for a mono signal. You can use either Mono(L) or Mono(R) to isolate either the left or right side of a stereo signal.

Audio Sampling Input doubles as a two channel "drum" trigger, allowing audio signals to trigger samples. On the Sample page, set Mode to Trigger. Adjust Thresh to control triggering sensitivity. This triggers the currently assigned click program. The left input will trigger click key note number +1, right input will trigger click key +2. The click key and click program can be accessed on the Song Mode MISC page.

### Threshold (Thresh)

The Thresh parameter controls when the K2vx actually begins sampling incoming signals. If you set it to a value of Off, sampling begins immediately when you press the Record soft button. Otherwise the K2vx waits for the incoming signal to exceed a specified threshold before beginning to record. You can set the threshold from -90 to 0 dB, in 6 dB increments.

The K2vx has a pre-trigger feature—that is, the K2vx records the 3,000 samples immediately before the point at which the threshold is exceeded. This prevents early transients from being missed. This feature is automatic whenever the Thresh parameter is set to a non-zero value. You'll want to adjust this parameter if your samples are missing these transients; the lower the threshold, the more certain you'll be of capturing every element of the signal. In some cases, you may get better results if you set the Thresh parameter to a value of Off, then press the

Record soft button and wait briefly before beginning the sound to be sampled. This will guarantee that every element of the sample is captured. After you've recorded the sample, you can always adjust the starting point on the TRIM page, to remove any silence before the start of the actual sample.

Sampler recording can also be triggered via the keyboard. On the Sample page, set Thresh to "key", then press Auto. Striking a MIDI note event now will trigger the sampler, set the sample placement on the keyboard and relative tuning all in one easy step making sample mapping easy and intuitive.

## Time

The Time parameter lets you determine how long the sample will be. The available sample time is a function of the sample rate and the amount of available sample memory. The K2vx calculates this automatically, and limits the maximum value of the Time parameter accordingly. At a value of zero for this parameter, the K2vx will not record.

## Sample

The Sample parameter lets you select any sample in memory for auditioning. This is a convenient way to listen to the samples you've made without having to manually create keymaps and programs for them. With a value of None for this parameter, the K2vx plays the last program or setup you selected before entering Sample mode. The list of values includes all ROM and RAM samples.

When you select a sample for auditioning, the K2vx automatically creates a temporary keymap and program, based on the settings for Default Program 199—which is a simple single-keymap program with few controller assignments, and the effects set to 0% wet (100% dry). Any edits you've made to Program 199 are reflected in the sample you audition. When you exit the Sample mode page, the temporary keymap and program disappear until the next time you audition a sample. You can create regular RAM keymaps and programs using the Preview soft button; see the discussion of the Preview button in the section below called "Recording Samples."

If you don't have enough free program RAM, you may be unable to audition samples, since the K2vx won't have enough RAM to create the temporary keymap and program. In this case, deleting a few objects from RAM will restore the audition feature.

## Monitor (Mon)

The Monitor parameter provides a convenient way to listen to what you're recording. When set to a value of On, any signal received at the analog sample input will appear at the K2vx's MIX outputs and the headphone jack. Adjusting the input gain will affect the monitor gain as well. A clean monitor signal, however, does not guarantee a distortion-free sample. Always check the meters on the Sample mode page and look for the CLIP indicators to ensure that your sample is free of clipping. Note that the Mon parameter is not available when the Input parameter is set to a value of Digital. The Monitor feature applies only to the analog sampling inputs. You should monitor digital sources from the sources themselves.

For sampling the K2vx's own output, the Monitor parameter must be Off to prevent signal attenuation.

## Recording Samples

Pressing the Record soft button will begin the sample recording process. If the Thresh parameter is set to a value of Off, recording will begin immediately, and will continue for the number of seconds indicated by the Time parameter. The display will indicate that recording is in process. Any other value for the Thresh parameter will cause the K2vx to wait until the

specified threshold is exceeded, then recording will proceed normally. The display will indicate that you're making a threshold recording, but won't actually begin recording until the threshold is exceeded.

You can abort the recording of a sample by pressing the **-/+**, **0** and **CLR** buttons simultaneously.

When recording is complete, the K2vx will prompt you to strike a root key. The sample is assigned to the key you strike. This "root" is the key at which the sample will be played back without transposition. When sampling pitched sounds, it generally makes sense to assign a root key that matches the pitch of the original sample, although you can set the root key anywhere you like. If you press the **Default** soft button, the K2vx uses **C 4**. You can change the root key at any time on the **MISC** page in the Sample Editor.

When the root key has been assigned, the K2vx asks you if you want to save the sample. At this point the display will show one of two things—the number of clips, or if no clips occurred, the maximum level (in dB) of the sample signal.

You can listen to the sample before deciding whether to save it. If you decide not to keep the sample, press the **No** soft button, and you'll return to the Sample mode page. If you press **Yes**, you'll see the normal Save dialog. When you've saved the sample, you'll return to the Sample mode page. You'll also have the opportunity to name the sample. A recommended convention for naming samples is to include the root key as part of the name. This is particularly useful for pitched samples. Including the root key in the sample name helps when you are creating a keymap, because it tells you how much transposition of the sample you will hear depending on its key assignment.

Once the sample is recorded and saved, you may want to edit it, using the **TRIM** page, **LOOP** page, or any of the sample DSP functions.

### The Auto Soft Button

To save time when sampling with either the Analog or Digital inputs, you can use the **Auto** soft button. This will begin sampling immediately (or when the sample threshold is exceeded if you have the **Thresh** parameter set to a value other than **Off**), and will return you to the Sample Recording page when the sample is complete. The root key is automatically assigned at **C 4**, and the sample is automatically saved at the next available ID above 199.

### The Timer Soft Button

If you need to delay the beginning of your sample recording, you can press the **Timer** soft button instead of the **Record** or **Auto** soft buttons. This will begin a ten-second countdown before sample recording actually starts. The display will show the countdown. When the countdown reaches zero, The Program, Setup, MIDI, and Master mode LEDs will flash three times. If you have the **Thresh** parameter set to a value of **Off**, sample recording will start immediately after the LEDs flash. If you have the **Thresh** parameter set to a value other than **Off**, sample recording will begin when the countdown has finished and the threshold level is exceeded.

### The Preview Soft Button

When you've finished taking a sample, you can press the **Preview** soft button to automatically create a keymap and program using the new sample. It uses the settings for the Default program 199 as a template. Unlike the temporary keymap that's created when you audition a sample (and disappears when you select another sample), the preview keymap and program are stored in RAM and can be selected at a later time. The program and keymap will have the same name as the sample.

When you press the **Preview** soft button, the Bank dialog appears, prompting you to select a bank where the preview program will be stored. Select a bank, then press the OK soft button. The K2vx will create a keymap and a program, using the lowest available ID numbers in that bank for both the keymap and the program. The display will tell you the ID number of the new program.

### Sampling the K2vx's Output

You can sample the K2vx's own sounds when in Analog sampling mode. If there are no cables connected to the Analog sampling input, the K2vx will record its output automatically. To do so, set the Src parameter on the Sample mode page to a value of Int. Then, just press the **Record** soft button and start playing. The K2vx will sample the note(s) as played, without any controller information (just the raw sounds).

### Sampling Digital Signals

The process for sampling through either of the digital inputs is essentially the same, although there are a few additional parameters associated with digital sampling formats.

You'll notice that the Sample mode page changes considerably when you change the value of the Input parameter from Analog to Digital. There are a few more settings to be made before you start recording.

```

SampleMode Samples:8192K Memory:118K
Sample:None
Input :Digital Time :1s
Cable :Coaxial
Format:AES/EBU
Mode :Stereo
Thresh:Off -dB 60 40 * 16 * 8 4 0
Record Auto Timer Preview

```

The first difference is the fact that there are no parameters for gain and sample rate. There's no need for a gain parameter because with digital sampling, since you're making an exact digital copy of the source signal. The Rate parameter is excluded because the K2vx automatically recognizes the source sample's rate and sets its own rate accordingly. Also, the Mon parameter does not appear when sampling digitally. Any monitoring you wish to do must be done from the sample source.

### Cable

Set the Cable parameter to a value of Coaxial or Optical, depending on the type of cable you're using. Many consumer products with digital output provide an optical jack. There is a small plug covering the K2vx's optical input; this plug must be removed before you can connect the optical cable to the K2vx. The plug should be replaced whenever the optical input is not in use.

If you plan to use a coaxial cable, select a value of Coaxial for the Cable parameter. If you're sampling from a consumer product like a CD player, you'll probably need to get an adaptor, or a cable with an RCA fitting on one end and an XLR (male) fitting on the K2vx end. Some professional products have an XLR fitting for their outputs, so an ordinary microphone cable will be suitable in most cases. Longer cables can cause signal loss, however, so if you need a long cable, you may need to get a special cable designed for digital information transfers. There's detailed information about cables for sampling on page 15-1.

#### Format

Use the Format parameter to tell the K2vx the format of the incoming sample. Most consumer products use SPDIF (Sony / Philips Digital Interface Format), while most professional machines use the AES/EBU (Audio Engineering Society / European Broadcast Union) format. Refer to the owner's manual of your sample source for information regarding its digital format.

The Mode, Time, and Thresh parameters function for digital sampling just as they do for analog sampling.

When you begin recording, a box appears in the display that indicates that the K2vx has locked on to the digital signal. You'll see the word "LOCK," and the rate at which the signal was sampled will be shown. The K2vx automatically sets itself to this rate. If you don't see this box, check to make sure that the signal is getting to the K2vx.

The K2vx may not display the rate when you're using a rate other than the frequently-used rates of 32, 44.1 or 48 KHz. This is normal, and although the rate may not be displayed, the K2vx will still lock on to it.

## Using the Digital Outputs

The K2vx features digital audio outputs with XLR connectors. There are two points to keep in mind if you wish to use them. First, the Mode parameter on the Sample mode page must be set to a value of Analog to enable the digital outputs. Second, the analog input must be disconnected from any cables.

If you have a cable connected to the analog input, that input will be routed to the digital outputs. This effectively makes the K2vx into an analog-to-digital convertor, which can be very useful. The digital out will have the same signal that is sent to the MIX outs. So if you have all of your separate outs plugged in (and therefore have no signal coming out of the MIX outs), you will not have any signal coming out of the digital outs.

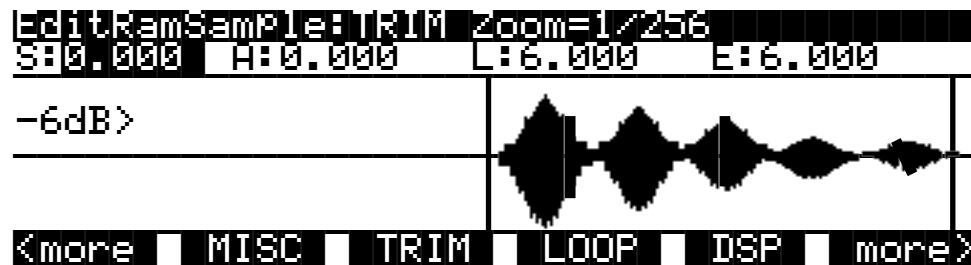
## Editing Samples

Most of the functions within the Sample Editor follow a general pattern. There are two ways to enter the Sample Editor. If you start from Master mode and press the Sample soft button, then select a sample and press EDIT, you can hear the isolated sample. If you want to hear the sample in the context of a program, start by entering the Sample Editor through the Program Editor: start by selecting a program in Program mode—usually the program containing the sample you want to edit. Press the EDIT button to enter the Program Editor. Press the KEYMAP soft button to view the KEYMAP page. The KeyMap parameter is selected (highlighted) when the page appears. Press the EDIT button to enter the Keymap Editor. The KeyRange parameter is selected when the page appears. The notes within the currently selected key range are the only ones that will be affected by your edits. You can hold the Enter button and trigger notes to select different key ranges.

If you want to select a different sample, use the cursor buttons to select the Sample parameter. Use the Alpha Wheel to select a sample. Press the EDIT button once more, and you'll enter the Sample Editor. (Pressing the EDIT button while in the Keymap Editor will enter the Sample Editor regardless of which parameter is selected.) The effects of the current program will be applied to the sample.

The TRIM page appears when you enter the Sample Editor. A representative TRIM page is shown below.





There are three basic sample editing pages—TRIM, LOOP, and MISC (Miscellaneous). The soft buttons for these pages are visible when you enter the Sample Editor. The DSP soft button is visible as well if you're editing a RAM sample; pressing it will take you to the DSP function page, where you can select a DSP function with the Alpha Wheel or Plus/Minus buttons.

The DSP soft button does not appear if you're viewing a ROM sample. Instead you'll see a **Link** soft button.

The **<more>** soft buttons will take you to the soft buttons for the other functions. You can trigger notes at any time while you're editing, to hear your changes as you make them.

## The Function Soft Buttons in the Sample Editor

In addition to the MISC, TRIM, and LOOP soft buttons, which select Sample Editor pages, there are several function soft buttons. As with other K2vx editors, the function soft buttons are labeled with upper and lower case letters, to distinguish them from the page selection soft buttons, which are labeled in all capital letters. The **<more>** soft buttons give you access to the other soft buttons that are available.

### Zoom- and Zoom+

These buttons are active only when you're viewing the TRIM and LOOP pages. They increase or decrease the resolution of the waveform display, enabling you to see a larger or smaller segment of the waveform of the currently selected sample. The top line of the display indicates the zoom position in terms of a fraction—for example, 1/256—which indicates the number of individual sample elements represented by each display pixel. A value of 1/256 means that each pixel represents 256 individual sample elements. The maximum zoom setting of 1 shows you a very small segment of the sample. The minimum setting of 1/16384 shows you the largest possible segment of the sample. Each press of a Zoom soft button increases or decreases the zoom by a factor of 4.

As a convenience, the Program and Setup mode buttons also serve as zoom buttons while in the Sample Editor. You can press the two left soft buttons together to toggle between the default zoom setting and your current zoom setting.

### Gain- and Gain+

Also active only for the TRIM and LOOP pages, these buttons increase or decrease the magnification of the currently displayed sample waveform, enabling you to see the waveform in greater or lesser detail. At the left of the display, you'll see the magnification setting, which is expressed in dB units. You can adjust the magnification from -72 dB (maximum magnification) to 0 dB. This doesn't affect the actual amplitude of the sample, only the magnification of its display.

As a convenience, the MIDI and Master mode buttons also serve as gain adjustment buttons while in the Sample Editor.

The simplest way to think of the Zoom and Gain buttons is to remember that the Zoom buttons control the left/right magnification of the waveform, while the Gain buttons control the up/down magnification. Neither button has any affect on the sound of the sample. You'll often use the Zoom and Gain soft buttons together to focus in on a particular sample segment, then magnify it to see it in close detail.

For example, you might want to zoom out to view an entire sample waveform, to decide which segment you want to edit. You could then zoom in to focus on a particular segment. Once you've zoomed in, you may want to boost the Gain to enable you to set a new Start (S) point with greater precision, or ensure that you get a smooth loop transition.

### Abort

Use the Abort soft button to cancel a sample dump before it's complete. You'll be prompted to verify whether you really want to cancel the dump.

### Split

The Split soft button enables you to create two mono samples from a single stereo sample, or to split up a multi-root block of samples. This button is available only when the currently selected sample is a stereo sample. When you press this button, the K2vx will prompt you: Split this sample? When you press the Yes soft button, you'll be prompted to enter the ID for the first sample. Select an ID with the Alpha Wheel or Plus/Minus buttons, then press the OK soft button. The split samples will automatically be assigned IDs, starting with the ID you select.

Splitting stereo samples enables you to use the separate sides individually, or to phase the samples by assigning each side to a separate keymap, then delaying one of the layers slightly.

### Units

With the Units soft button you can change the units used to display the locations of the current sample's Start, Alt, Loop and End points. The default setting displays these points in seconds, that is, the number of seconds from the physical start of the sample. Pressing the Units soft button will change the units to samples, that is the number of individual sample elements from the physical start of the sample. Press it again to return to a view of the sample in seconds.

As a convenience, the Quick Access mode button also serves as a units button while in the Sample Editor.

### Link

The Link soft button lets you fix the interval between the Start, Alt, Loop and end points of the current sample, so it remains constant when you move one or more of the points. This is done by selecting the desired parameter with the cursor buttons, then pressing the Link soft button. The colon (:) following the parameter's name will change to an arrow (>) to indicate that it is linked. You can link any or all of the four sample points. When sample points are linked, moving one of them will move the linked points correspondingly. For example, suppose the current sample's Start (S) point is 0.0 seconds, and its Alt (A) point is 0.5 seconds. The interval between the sample's Start and Alt points is exactly half a second. If you select the Start parameter, then press the Link soft button, the Start point will be linked. This won't have any effect until you link at least one more point. If you select the Alt parameter and press the Link soft button, the Start and Alt points will be linked. Now if you move the Start point to 1.0 seconds, the Alt point will automatically move to 1.5 seconds, preserving the half-second interval between Start and Alt. To remove the link on any of the points, select the point again,

and press the **Link** soft button again. The arrow will change to a colon, indicating that the link has been removed.

As a convenience, the Song mode button also serves as a link button while in the Sample Editor.

## Name, Save, Delete, and Dump

These soft buttons are similar to the Name, Save, Delete, and Dump soft buttons in the other editors, initiating the corresponding dialogs to name, save, delete, or dump the currently selected sample.

When you press the Save soft button, and choose to save the sample to a new ID, the K2vx will ask you if you want to copy the sample data. If you answer Yes, the K2vx will make a separate copy of the sample. If you answer No, the K2vx will simply mark the location of the original sample data and share the sample between the original and the edited sample. This can save a great deal of memory space. If you delete a sample that's partially or completely shared with another, the K2vx deletes only the portions that are unused by the shared sample, always optimizing its memory for maximum storage capacity.

Note that if you use the Utility function to view the objects currently stored in the K2vx, you'll see each object listed separately, including shared samples. The shared samples will each indicate their size, even though they're referring to the same memory location. This might lead you to believe that you're using more memory for samples than you actually are. If you use the Utility function to calculate your total sample memory usage, remember not to include any shared samples in the total.

## The Page Buttons

The soft buttons labeled in all capital letters select the various pages in the Sample Editor: MISC, TRIM, and LOOP (as well as DSP when you're editing RAM samples).

## Miscellaneous (MISC)

On the MISC page, you'll set several parameters that affect the behavior of the current sample. These parameters affect the entire sample. The diagram of the MISC page shows a ROM sample, and since the DSP functions cannot be applied to ROM samples, the DSP soft button is not available. In its place is the Link soft button, which enables you to maintain equal times between various points in the sample—Start and Alt, for example.

The default values shown in this diagram reflect the settings for the Default program 199.

```

EditRomSample:MISC
RootKeyNum   :G#1      LoopSwitch:On
PitchAdjust  :-12ct    Playback   :Normal
VolumeAdjust :0.0dB    AltSense   :Norm
AltVolAdjust :-0.5dB   IgnRelease:Off
DecayRate    :36B/s    SampleSize=101Kb
ReleaseRate  :106dB/s  SampleRate=19200Hz
<more>  MISC  TRIM  LOOP  Link  more>

```

PARAMETER	RANGE OF VALUES
Root Key Number	C -1 to G 9
Pitch Adjust	Variable (depends on sample rate)
Volume Adjust	-64.0 to 63.5 dB
Alternative Volume Adjust	-64.0 to 63.5 dB

<b>Decay Rate</b>	<b>0 to 5000 dB per second</b>
<b>Release Rate</b>	<b>0 to 5000 dB per second</b>
<b>Loop Switch</b>	<b>Off, On</b>
<b>Playback Mode</b>	<b>Normal, Reverse, Bidirectional</b>
<b>Alternative Sample Sense</b>	<b>Normal, Reverse</b>
<b>Ignore Release</b>	<b>Off, On</b>

### Root Key Number

The root key number represents the key at which the sample will play back without transposition (at the same pitch as the pitch of the original sample). When you're creating your own samples, the key you strike as the root key will be the key you see as the value for this parameter.

### Pitch Adjust

Use this parameter to change the pitch of the sample relative to the key from which it's played. Setting a value of 100 cts, for example, will cause the sample to play back one semitone higher than normal. This parameter is handy for fine tuning samples to each other if they're slightly out of tune.

### Volume Adjust

Uniformly boost or cut the amplitude of the entire sample. Compare this to the DSP Volume Adjust parameter, which lets you boost or cut the amplitude of a specified segment of a RAM sample.

### Alternative Start Volume Adjust (AltVolAdjust)

This parameter sets the amplitude of the sample when the alternative start is used. See Chapter 6 for a discussion of AltSwitch.

### Decay Rate

Use this parameter to specify how long the sample will take to decay (fade) to zero amplitude when a note is sustained, either by holding the key (or other note trigger), or with the sustain pedal. The higher the value, the faster the sample will decay. The sample starts decaying as soon as the note is triggered, and decays with a linear curve. This decay affects each sample individually, and begins as soon as the note starts. It operates independently of the decay in the AMPENV, which affects every note in the layer.

### Release Rate

The release rate determines how long the sample will take to decay to zero amplitude when the note trigger is released. The higher the value, the faster the release rate. This release affects each sample individually, and begins as soon as the note is released. It operates independently of the release in the AMPENV, which affects every note in the layer.

To create an extended sample loop that will play data after the sample's loop on key-up, set the Alt sample pointer after the sample end pointer, then set a relatively low value for the release rate.

### Loop Switch

This parameter activates or deactivates the looping of the currently selected sample. When set to On, the sample will loop according to the settings on the LOOP page. When set to Off, the sample will play through to its End point and stop.

### Playback Mode (Playback)

This parameter lets you modify the direction in which the sample is played. Set it to a value of Reverse if you want the sample to play from its End point to its Start (S) point. Choose a value of Bidirectional to cause the sample to play from Start to End, then reverse direction and play again from End to Loop and back, repeating until the note trigger is released (this works only when the Loop Switch parameter is set to On).

### Alternative Sample Sense (AltSense)

This provides a convenient way to activate the alternative start of a sample. When set to Normal, the alternative start will be used when the Alt Switch control is On (this is set on the KEYMAP page), or when the control source assigned to it is above its midpoint. When set to Reverse, the alternative start will be used when the Alt Switch control is Off, or when the control source assigned to it is below its midpoint.

### Ignore Release (IgnRelease)

When set to a value of Off, the sample will release normally when the note trigger is released. When set to On, the note will not release, even when the note trigger is released. This setting should be used only with samples that normally decay to silence; non-decaying samples will play forever at this setting. This parameter is equivalent to the IgnRelease parameter on the LAYER page, but affects only the currently selected sample.

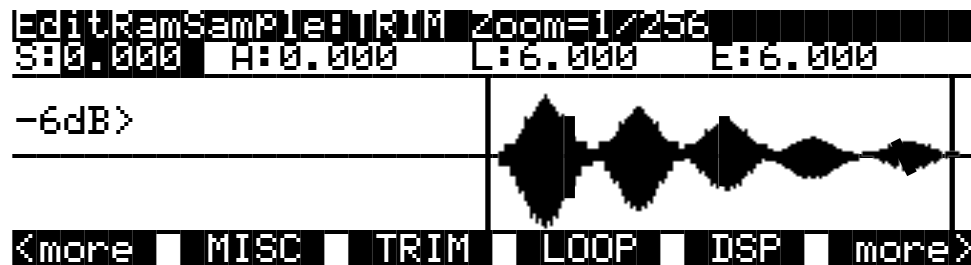
### Sample Size and Sample Rate

These are not editing parameters. They're put on this page to give you a convenient place to check the size in Kilobytes of the current sample, and the rate at which it was sampled.

## TRIM

The TRIM page lets you set the Start, Alt, Loop, and End points of the current sample. The top line tells you whether you're editing a ROM or RAM sample, and indicates the zoom setting. At the left of the display is the Gain (display magnification) setting. This Gain setting doesn't affect the amplitude of the sample, just the view in the display.

In the diagram below, the sample points are expressed in individual sample elements. Pressing the Units soft button (or the Quick Access mode button) will display them in seconds.



The four parameters on this page are Start (S), Alternative Start (A), Loop (L), and End (E). Selecting these parameters and adjusting their values enables you to modify how the sample plays back when notes are triggered.

There are four vertical lines that indicate the settings of the four parameters. You'll see all four lines only if the values for each of the four parameters are different; otherwise, the lines will overlap. Selecting one of the parameters will move the line corresponding to that parameter to the center of the display, where it will flash to indicate the parameter's position. The line corresponding to the currently selected parameter will flash. Turning the Alpha Wheel will move the sample waveform relative to the line. The line remains in the center of the display, and the waveform shifts to indicate the new position of the point. You can also use the alphanumeric buttonpad to enter new values directly. Press the ENTER button to register the values you enter.

The Start (S) point determines the beginning of the current sample. You can truncate the beginning of the sample by increasing the value of the Start (S) parameter. You might do this to remove silence at the beginning of a sample, or to remove some or all of the attack. You can't decrease the Start point of RAM samples below zero, but if you want to add silence at the beginning of a RAM sample to create a delay, you can use the Insert Zero DSP function to add as much silence as you like. The start points of ROM samples can be set lower than zero (you can set negative numbers for the Start (S) and Alt (A) parameters). Doing this will cause portions of other samples to be played, which can create interesting effects.

Note that for RAM samples, you won't see any waveforms displayed to the left of the Start point, but you will for ROM samples.

The Alt (A) parameter lets you set a second, optional start or end point for the current sample. The Alt will be used when the Alt Switch parameter on the KEYMAP page is set to On, or when it's set to a specific control source and that control source is generating a value of more than +.5 (for example, if you assign MWheel as the control source for the Alt Switch parameter, the Alt will be used when the Mod Wheel—or whatever control source you have set to send MWheel—is above its halfway point). The Alt can be set before, after, or at the same point as the Start or end.

If you set the Alt after the End, you can extend the play of looped samples. Normally, looped samples will play through to the End, then will loop back to the Loop point, and continue looping like this until the note is released, when they go into their normal release. If the Alt is set after the End, looped samples will loop in the same way while notes are sustained. As soon as you release the notes, however, the samples will play through to the Alt point before going into release.

The Loop (L) parameter sets the beginning of the looped portion of the current sample. Although you can adjust this parameter while you're on the TRIM page, you'll normally want to adjust it while viewing the LOOP page, so you can see the loop transition points, enabling you to create as smooth a loop as possible. The Loop can be set at any point before the End,

including before the Start and Alt. If you try to move it after the End, the End will be moved with it.

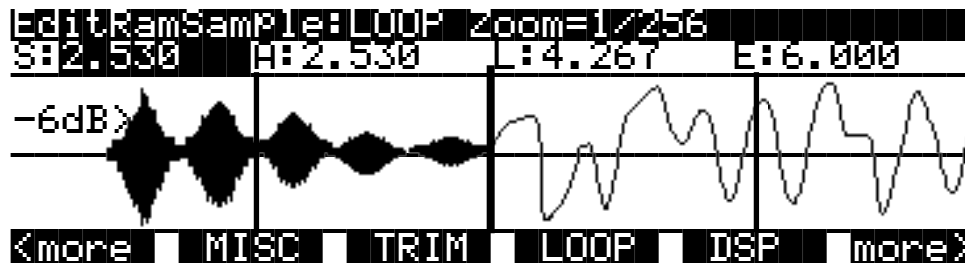
The End (E) parameter sets the point at which the current sample will stop playback. Typically you'll use this parameter to trim unwanted silence off the end of a sample, although you can use it to shorten a sample as much as you want.

If you want to truncate a sample to save memory, there are two points to keep in mind. First, if the Alt parameter is set before the Start, you won't save any memory by truncating the Start. Likewise, you won't save memory by truncating the End if the Alt is set after the End. You won't save memory by truncating a sample until you save the sample and exit the Sample Editor.

You can also use the Truncate DSP function to automatically truncate your samples at a specified noise floor. As with the TRIM page, you'll recover memory after saving the sample and exiting the Sample Editor.

## LOOP

The LOOP page features the same four parameters as the TRIM page, but the waveform display is quite different. The best way to understand what you see on the LOOP page is to switch back and forth between the TRIM and LOOP pages and study the waveform displays.



On the TRIM page you see the entire waveform—or as much of the waveform as your current zoom setting allows. When you move to the LOOP page, you'll notice that the page is split into two sections, left and right, divided by a vertical bar in the center. This bar is thicker than the vertical lines representing the Start, Alt, Loop, and End points, and does not move when you adjust any of these points.

To the left of the dividing bar you see the same segment of the current sample that you see on the TRIM page. The four vertical lines representing the Start (S), Alt (A), Loop (L), and End (E) points are visible. (Remember, you'll see all four vertical lines only if the values for the Start, Alt, Loop, and End parameters are different.) To the right of the dividing bar you see the entire loop segment of the sample.

In the center of the loop segment is a dotted vertical bar that represents the loop transition point—that is, the point at which the sample reaches its End point and loops back to the Loop point. You can visualize the loop segment by starting at the vertical transition point; this is the beginning of the loop, as defined by the setting for the Loop parameter. The waveform progresses to the right, representing the initial portion of the loop segment. The waveform “disappears” off the far right side of the display, and “reappears” at the thick dividing bar at the center of the display. The waveform again progresses to the right, representing the final portion of the loop segment. It reaches the dotted vertical transition line, representing the End point of the sample, where it jumps once again to the loop point and repeats the cycle.

If you select the Loop (L) parameter and change its value, you'll see the segment of the waveform to the right of the transition point shift its position. If you select the End parameter and change its value, you'll see the segment of the waveform to the left of the transition point shift its position.

When you're setting a loop segment for a sample, you'll want to adjust both the Loop and End parameters so the two ends of the waveform meet (or come as close as possible) at the transition point. You'll notice an audible click in your sample loop if the ends of the waveform do not meet at the transition point. The closer you can get the two ends of the waveform, the better the sound quality of your loop will be. With a bit of experimentation, you'll develop the ability to create smooth loop transitions.

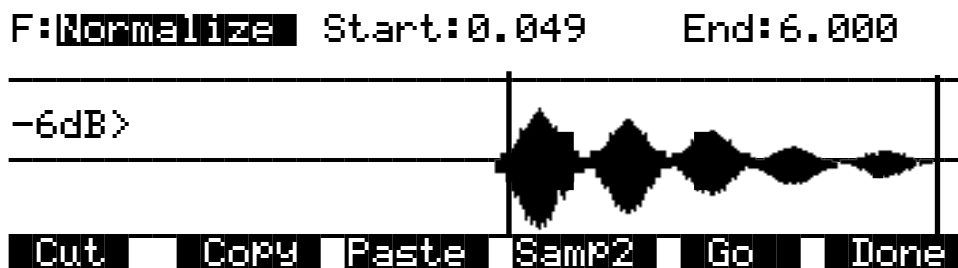
You'll also want to try to set the loop point at a zero-crossing—a point where the sample waveform is neither positive or negative. Pressing the Plus/Minus buttons together will search (from left to right) for the sample's next zero-crossing. You can usually press these buttons several times for any given sample without noticeably affecting the sound of the sample. If you press the Minus button, you'll reverse the direction of the search, and the next time you press the Plus/Minus buttons together, the K2vx will search for the next zero-crossing to the left. Press the Plus button again to search toward the right.

If you adjust the display Gain and Zoom of the sample while on the LOOP page, you'll notice that the Gain affects the waveform on both sides of the loop point, while the Zoom affects only the left side of the page. You can't zoom in on the loop transition point in the right half of the display.

You can also use the crossfade loop (XfadeLoop) DSP function to get a smooth transition between loop points. As with the TRIM page, you'll recover memory after saving the sample and exiting the Sample Editor.

### DSP (RAM samples only)

Select the DSP page with the DSP soft button. This gives you access to a long list of non-real time DSP functions, with which you can modify your RAM samples. The first time you select a DSP function, you'll see the Normalize function, shown below. Afterward, the most recently selected DSP function will appear when you select the DSP page.



All of the DSP functions operate on a segment of the current sample that you select before executing the function. In most cases, you'll use the Start and End parameters to define the start position and end position of the segment you want to modify. There are a few exceptions to this rule, which will be explained as applicable.

Please keep in mind that the Start and End parameters on the DSP pages are not the same as the sample Start (S) and End (E) parameters that you set on the TRIM and LOOP pages. When you're working on one of the sample DSP functions, Start and End position refer to the range of the sample that you want to process. Adjusting these parameters does not affect the overall



start and end of the sample. It affects only the portion of the sample that you want to process. When you audition the sample by triggering a note, you'll hear only the range of the sample within the Start and End parameters on the current DSP page. To hear the entire sample, press the Done soft button to return to the Sample Editor page. Similarly, moving the S and E parameters on the TRIM or LOOP pages have no effect on the Start and End parameters on the DSP pages.

Use the Plus/Minus buttons or the Alpha Wheel to select the starting and ending positions of the selected sample segment (the Start and End parameters). You can audition the selected sample segment by triggering any note within the current key range.

The actual processing of the sample begins when you press the Go button. The K2vx will display a row of scrolling dots while it's processing. When it's finished, it will prompt you : "Keep these changes?" Trigger a note to hear the result. Press the No soft button to return to the page for the currently selected DSP function. This will undo any changes you've made. Press the Yes soft button to make the change. The Save dialog will appear (except for the Truncate function). You can save to a different ID than the one displayed, if you want to preserve the original sample. Pressing the Cancel soft button will return you to the current DSP page without saving the sample. (the sample reverts to its original condition.)

Six of the DSP functions ( Mix, Mix Beat, Mix Echo, Insert, Replicate, and Beat Volume Adjust) involve selecting a second sample segment (Sample 2) to be processed with the currently selected sample. In these cases, use the **Samp2** soft button. When you press it, another page appears, enabling you to select a second RAM sample using the Alpha Wheel or the alphanumeric buttonpad. You can audition Sample 2 by triggering a note.

Once you've selected Sample 2, use the Start and End parameters to define the start and end positions of the segment of Sample 2 that you want to process. While you're on the Samp2 page, you'll hear the currently selected Samp2 when you trigger a note. When these positions are defined, press the OK soft button to return to the DSP page where you can continue your editing.

Once you've selected a sample on the Samp2 page, it remains selected until you return to the Samp2 page and select another sample. You can always use the Samp2 page to audition a second sample.

You can use the Copy soft button on the Samp2 page to copy the selected segment to a buffer, then paste the segment into Sample 1 when you return to the DSP page.

## The Soft Buttons on the DSP Page

### Cut

This button will cut (remove) the currently selected sample segment from the currently selected sample, and store it in a buffer. This is equivalent to cutting a section out of a piece of audio tape and splicing the remaining ends together. The cut segment is then available for pasting elsewhere in the sample. The segment you cut will remain in memory until you replace it with another cut or copy command, or until the K2vx is shut off. If you accidentally cut a sample segment, you can restore it by immediately pressing the Paste soft button.

Compare this function to the Delete DSP function, which removes the selected sample segment, but doesn't store it in a buffer. If you delete a sample segment, it's *gone*.

#### Copy

Use this button to store the selected sample segment in a buffer without altering the current sample. The copied segment will remain in memory until you replace it with another cut or copy command, or until the K2vx is shut off.

#### Paste

This button has an effect only after you've cut or copied a sample segment using the Cut or Copy soft buttons. The Paste soft button inserts the contents of the Cut/Copy buffer after the start position of the currently selected sample segment. This is like splicing a section of audio tape into another section; it extends the length of the sample. You can undo a paste by pressing the Cut soft button immediately after pasting.

When you cut or copy a sample, it's stored in a buffer. Whatever is stored in the buffer will remain available for repeated paste commands until replaced by another cut or copy command, or until the K2vx is shut off or reset (soft or hard). The buffer is also cleared if you execute a master delete, and select "Delete everything."

#### Samp2

The Samp2 button, as described earlier in the DSP page section, enables you to select a segment of a second RAM sample to be processed with the currently selected sample.

#### Go

Press the Go soft button when you want to execute the currently selected DSP function on the currently selected sample. When processing is completed, the K2vx will prompt you to keep the changes. You can audition your changes and decide whether to keep them, then press Yes or No.

#### Done

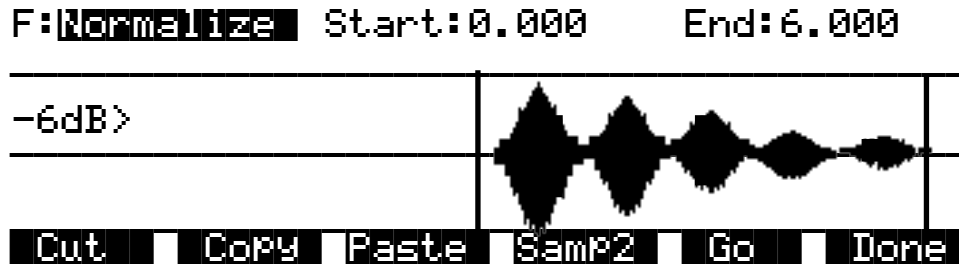
Press the Done soft button to return to the previously selected Sample Editor page when you're finished with the DSP functions.

## DSP Functions

Once you've entered the Sample Editor, press the DSP soft button to gain access to the DSP functions. The DSP function parameter will be highlighted, allowing you to scroll through the list of functions with the Alpha Wheel or Plus/Minus buttons.

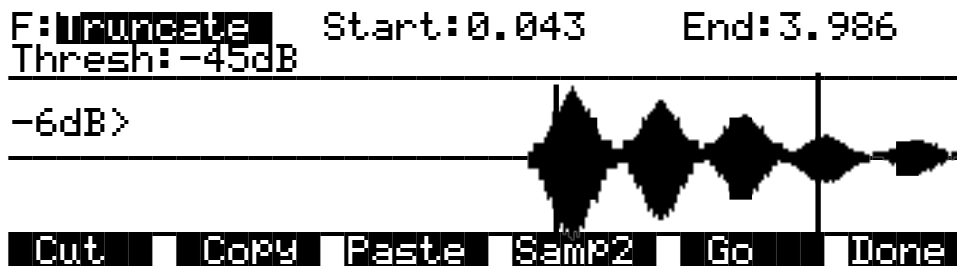
Remember that the DSP functions operate only on RAM samples. You'll notice that if you're editing a ROM sample, there is no DSP soft button; instead, there's an extra Link soft button for your convenience.

## 1 Normalize



With the Normalize function, you can rescale the amplitude of the selected sample segment to optimize its level relative to other samples. The Normalize function will uniformly boost the amplitude of the current sample range as high as possible without clipping, stopping just before the loudest element of the sample would be clipped. You might want to use the Volume Adjust function to boost the current segment manually, but the Normalize function does it automatically, and prevents you from boosting the amplitude too much.

## 2 Truncate

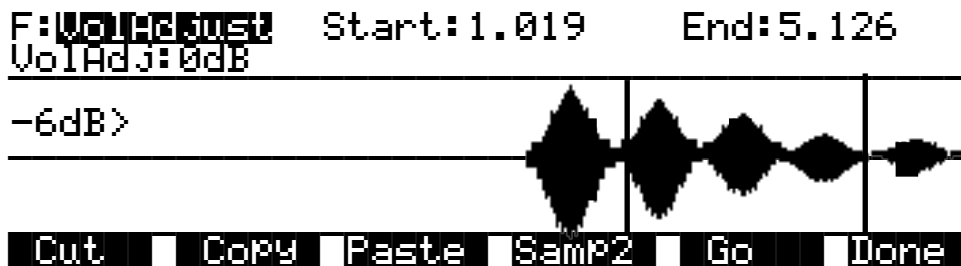


The Truncate function will automatically reset the Start (S) and End (E) points of the sample. This can be quicker than trimming the sample manually on the TRIM page.

Use the Start and End parameters to select a specific sample range. Set the Thresh parameter from -96 to 0dB to set the noise floor. When you press the Go soft button, the K2vx will search inward from the start and end points you set, until it finds the first sample that exceeds the noise floor. Everything outside the range will be left out. You'll want to experiment with different thresholds to find the noise floor that suits each sample you truncate. If the new End point is inside the current loop point, the loop will be disabled.

When you press the Go soft button, you'll be prompted "Keep this change?" Unlike the other DSP functions, answering Yes does not bring up the Save dialog. It registers the new Start and End positions as the Start (S) and End (E) points (as set on the TRIM or LOOP page). In fact, you can go to either of these pages to readjust the S and E points if you want to include more of the sample. To save the new S and E points, press the Done soft button (if necessary) to return to the Sample mode page, then press either the Save soft button, or EXIT.

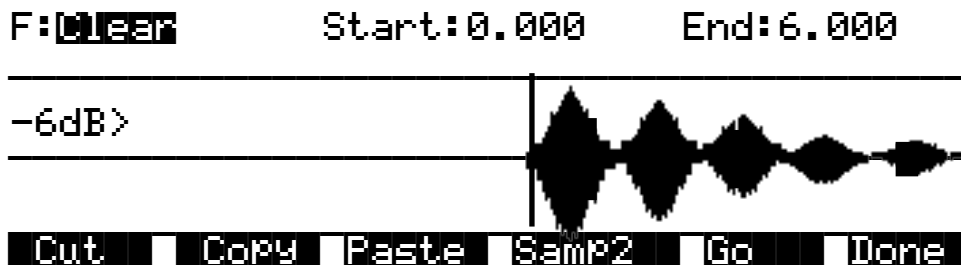
### 3 Volume Adjust



Use this function for a uniform cut or boost in the amplitude of the selected sample segment. This function will clip samples if you adjust the volume too high. This won't hurt the K2vx, and you may find it useful in some applications. In any case, you'll need to choose your start and end points carefully, if you want to avoid abrupt changes in volume.

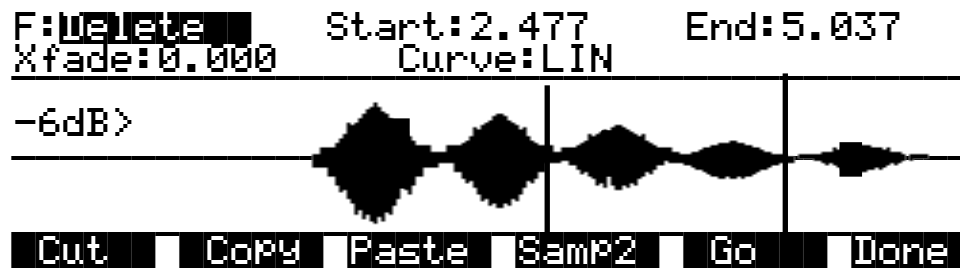
When you've selected the range to be adjusted, select the VolAdjust parameter with the cursor buttons, and use the Alpha Wheel to adjust the volume of the selected range. You can cut/boost the volume from -96 to 96 dB.

### 4 Clear



The result of this function is like erasing a section of recording tape. Use it to create sections of silence without changing the overall length of the sample. If you want to completely remove a segment and shorten the sample, you can do it with the Delete function.

## 5 Delete



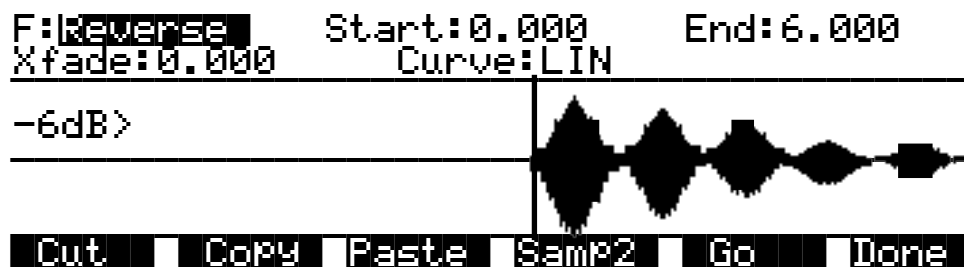
Unlike the Clear function, this will erase the samples within the selected range and shorten the sample, like cutting a section out of a tape and splicing the ends. If you want to silence a segment of the sample without shortening it, use the Clear function.

The Crossfade (Xfade) parameter enables you to smooth the transition from the deleted sample segment to the remaining sample segments, and can create overdub effects. The value of the Xfade parameter defines the amount of time the sample will take to fade to silence at the beginning of the deleted segment, and to ramp back up to the volume of the remaining portion of the sample.

The Curve parameter lets you choose from a variety of crossfade curves that affect the nature of the crossfade. A value of LIN gives a straight linear curve that fades uniformly from start to finish. A value of EXP sets an exponential curve that starts gradually and steepens toward the finish. A value of COS sets a Cosine curve. A value of EQL applies an equal crossfade that is the same at both ends of the deleted segment. A value of MIX assigns a curve that will start the crossfade before the start point of the deleted segment, and will end after the end point of the deleted segment. This differs from the other curves, which apply crossfade only to the deleted segment.

Crossfading a sample will shorten it by half the time you specify. The greatest length of crossfade you can get is the entire length of the shorter crossfaded segment (even though you can enter greater amounts with the Alpha Wheel or alphanumeric buttonpad). A crossfade of this length would crossfade the two samples over the entire length of the shorter segment.

## 6 Reverse

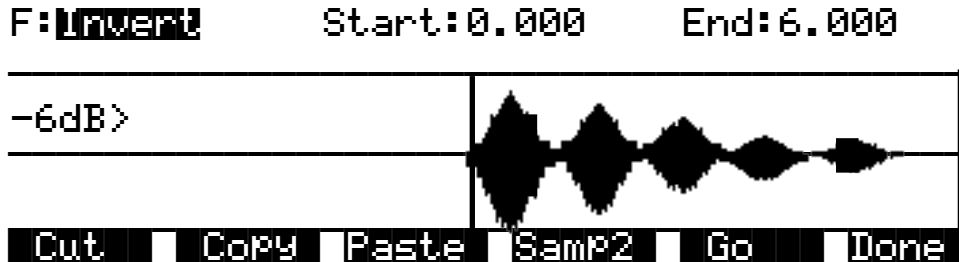


With this function you can reverse the order of the individual samples between the start and end positions you set. The Xfade parameter lets you apply a crossfade to the start and end of the reversed segment. The Curve parameter lets you select a crossfade curve. The available

values are LIN, EXP, COS, EQL, and MIX. These curves are described at the end of the section on DSP functions.

Like the crossfade parameter in the Delete function, this crossfade will also shorten the sample. The maximum crossfade length is half the length of the reversed segment.

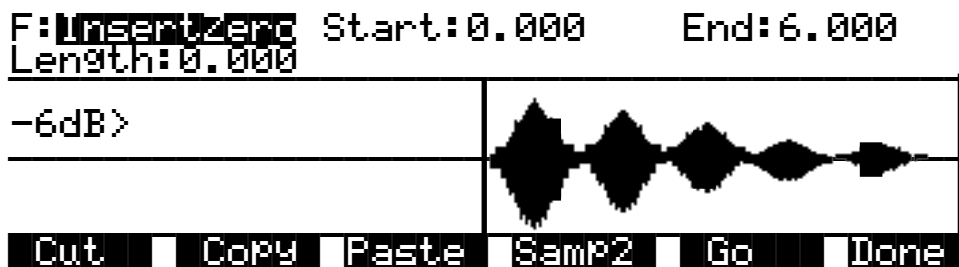
### 7 Invert



Use this function to invert the waveform of the selected sample range. This reverses the phase of the sample. You won't hear any difference unless it is played in reference with the original sample.

The results will vary depending on the type of sample. For an interesting phase effect, make a copy of a sample, then invert the copy, then assign it to the keymap of a second layer in a program that uses the original sample in the keymap of the first layer (or set them as the Left and Right samples in a stereo keymap). Add a delay to one of the layers or keymaps to create the phase effect.

### 8 Insert Zero



This function will insert a period of silence of any length into the selected sample range. This function is equivalent to splicing a section of blank tape into an existing segment of recorded tape. It's useful for creating a delay on the start of a sample, or creating a silent spot in the middle or at the end of the sample. Adjust the Start parameter to determine the point at which the period of silence will begin. Adjust the Length parameter to specify how long the silence will be. The End parameter has no effect for this function.

## 9 Mix



With this function you select a segment from Sample 2, and merge it with the selected segment from Sample 1, beginning at the point you set with the Start parameter. This is equivalent to mixing two audio signals through a mixing board. If the Sample 2 segment is longer than the segment from the Start (S) to the End (E) of Sample 1, the resulting sample will be longer than the original Sample 1.

Use the In and Out parameters to specify the length of time it takes Sample 2 to reach full amplitude and to fade to silence. The Curve parameter selects the curve of the fades. The available values are LIN, EXP, COS, EQL, and MIX. These curves are described at the end of the section on DSP functions.

The Volume Adjust parameter will cut or boost the amplitude of Sample 2 from -96 to 96 dB before merging.

If the sample rate of Sample 2 is different from that of Sample 1, the K2vx will alert you that the sample rates differ. If you mix samples with different rates, you'll hear a pitch shift in the mixed sample. If you don't want this pitch shift, use the Resample function to match the sample rates of the two samples before mixing.

### **Mixing Samples, Step-by-step**

First, use the Start parameter of Sample 1 to set the point at which the mix will begin. Then press the Samp2 soft button, and a page will appear enabling you to view and audition the list of RAM samples. Select a sample with the Sample parameter, then use the cursor keys to select the Start parameter. Turn the Alpha Wheel to set the beginning of the segment to be mixed in. Repeat this process for the end position of the sample. You can audition the sample as you're setting the range. Once the start and end points are set, press OK to return to the DSP page. Then press Go to initiate the mix.

## 10 Insert



Use this function to insert the selected segment from Sample 2 into Sample 1. This is like splicing a section of tape into an existing tape. This differs from the Mix function, which merges the two samples into one.

Use the Crossfade parameter to control the crossfades at the start and end of the inserted sample. The Curve parameter selects the curve of the crossfade. The available values are LIN, EXP, COS, EQL, and MIX. These curves are described at the end of the section on DSP functions.

The Volume Adjust parameter will cut or boost the amplitude of Sample 2 from -96 to 96 dB before merging.

If the sample rate of Sample 2 is different from that of Sample 1, the K2vx will alert you that the sample rates differ. If you insert a sample with a different rate, you'll hear a pitch shift in the inserted sample. If you don't want this pitch shift, use the Resample function to match the sample rates of the two samples before inserting.

### 11 Volume Ramp



This function lets you apply a ramp to the volume of the selected sample range. The Start Level and End Level parameters let you set the amount of cut (negative value) or boost (positive value) at the start and end points of the segment. The Curve parameter determines the shape of the ramp that scales the amplitude of the sample between the start and end amplitudes. The available values are LIN, EXP, COS, EQL, and MIX. These curves are described at the end of the section on DSP functions.

The VolRamp function affects only those samples within the start and end points. The sample will clip if you apply large amounts of volume ramp. If you want to ramp the volume of a sample segment up or down, then keep the volume at that level, use the Crescendo function.

### 12 Crescendo/Decrescendo (Crescendo)





Similar to Volume Ramp, this function applies a curve that scales the amplitude of the selected sample segment. Unlike Volume Ramp, however, you simply select a start and end point, and a single level. The amount of cut or boost starts at 0dB at the start point of the ramp, and reaches the level you specify when it reaches the end point of the ramp. The samples after the end point are scaled to the amplitude level of the end point.

When you set a negative end level (decrescendo), the sample's volume is cut by the amount you specify. If you set a positive end level (crescendo), the sample segment is first cut by the amount you specify, then is boosted back to its original level. This enables you to add large crescendos without clipping the sample. You may want to adjust the gain of the layer using the sample to match its level with other sounds.

Use the Curve parameter to select the shape of the curve within the selected segment. The available values are LIN, EXP, COS, EQL, and MIX. These curves are described at the end of the section on DSP functions.

### 13 Resample



Use this function to change the sample rate of the samples in the selected segment. This is convenient for converting samples to new rates matching those of other samples to be mixed with or inserted into. It's also useful for saving memory and altering the timbre of a sample.

If you include the entire sample in the segment, the new rate will be applied to the entire sample, and will be saved with the sample. If you select a shorter segment, only that segment will be modified, and it will sound pitch shifted relative to the remainder of the sample. To resample so that you'll hear a higher pitch for the selected segment, select a lower sample rate. For a lower pitch at the selected segment, choose a higher sample rate. This is because the K2vx applies the same playback rate to the entire sample, and doesn't compensate for the differing sample rates of the sample segments.

If the loop points of a looped sample are included in the segment to be converted, the K2vx will ask you if you want to adjust the rate slightly to optimize the loop. Press the No soft button if you don't want the rate adjusted.

You can use the Quick parameter to select from two resampling routines. Use Quick 0 to get an idea of the sound, then use Quick 1 for your final take.

### 14 Time Warp



With this function you can change the length of the selected sample segment without affecting the pitch. This function applies sophisticated routines that lengthen or shorten the selected sample segment to play it back over an different time period, modifying the playback rate so the pitch remains unchanged.

The Start and End parameters define the segment to be processed. Use the New Length (NewLen) parameter to specify how long you want the resulting sample to be While the function is in progress, the display will indicate the percentage of individual sample segments that have been processed.

The Quick parameter lets you select one of three warping routines. Use Quick 0 to audition your new sample. then use Quick 1 or 2 for your final take. Quick 2 takes more time to process, but gives you a better final result.

This function is extremely useful for fine adjustments in the length of a sample. You can also apply greater amounts of warp for a wide range of effects. Experimentation will give you an idea of the amount of alteration you want to apply.

### 15 Pitch Shift



The PitchShift function is the counterpart of the TimeWarp function; it shifts the pitch of the selected sample segment without changing the playback time—very useful for tuning samples when the playback time is crucial. Use the Start and End parameters to define the segment to be shifted. The Shift parameter determines the amount of pitch shifting, up to  $\pm 30000$  cents.

Like the crossfade parameter in the Delete function, this crossfade will also shorten the sample. The maximum crossfade length is half the length of the reversed segment.

The Quick parameter lets you select one of three shift routines. Use Quick 0 to audition your sample, then use Quick 1 or 2 for the final take. Quick 2 takes longer to process, but gives you better results.

## 16 Mix Beat



With this function you can mix the selected range of Sample 2 into Sample 1 at regular intervals. The most natural application of this function is to add percussion samples, but it can be used to mix any sample at regular programmable intervals (beats).

The Tempo parameter sets the interval between repetitions in beats per minute. The Of parameter establishes the number of beats per measure, and the Beat parameter sets which beat gets mixed. For example, if you set the Tempo parameter to 120, you'll have two beats per second. If you set the Beat parameter to 1, and the Of parameter to 1, Sample 2 will be mixed in twice per second, for the duration of the selected range of Sample 1. The first mix of Sample 2 occurs at the Start (S) point of Sample 1. If you change the Of parameter to 4, Sample 2 will be mixed into Sample 1 on the downbeat of every measure of 4.

The length of the ranges you set for Sample 1 and Sample 2 affects the results of the mix. In the above example, if Sample 1 is two seconds in length, and the mix of Sample 2 is on Beat 1 of 1, you'll mix four segments of Sample 2, at half-second intervals. On beat 1 of 4, you'd hear just one mix of Sample 2, right at the top.

To set up a MixBeat, first select the desired segment of Sample 1 using the Start and End parameters. Then press the Samp2 soft button to select the sample to be mixed in, and the selected range of that sample. You can use the Incr parameter to shift the starting position of Sample 2. If you set a positive number, the mix range of Sample 2 will start later each time it's mixed. A negative value will mix sample ranges that start progressively earlier.

To optimize processing time, keep the range of Sample 2 shorter than the interval between mixed-in segments. In the example above, if the mix were on beat 1 of 1, you'd want to keep Sample 2 at a range of a half second or less. If it were on beat 1 of 4, the range of Sample 2 could be as much as two seconds.

Next, press the OK soft button to return to the Mix Beat page, then use the Tempo parameter to select the rate at which Sample 2 will be mixed in. You can choose a tempo from 1 to 9999 beats per minute. Try setting the tempo at 9000 or more, and mix a very small range of Sample 2 into Sample 1 to create a wide variety of periodic waveforms.

Now set the values of the Beat and Of parameters to determine how the mixed sample will repeat. The Beat parameter determines the beat (s) on which Sample 2 will be mixed—from 1 to 9999. The Of parameter determines the measure length—also from 1 to 9999.

Finally, use the VolAdjust parameter to set the volume of the mixed sample segment—from -96 to 96 dB.

If the Of parameter is set to a value of 0, the Sample 2 segment will be mixed in on every beat, regardless of the setting for the Beat parameter. If the Beat parameter is set to a negative value, the segment of Sample 2 that's mixed in will be moved forward in time by the length of one beat each time it's mixed in; that is, you'll hear a later portion of the sample. Another way to accomplish this is to use the Increment (Incr) parameter on the Samp2 page. Set it to a positive value to use a later portion of Sample 2 with each repetition, or a negative value to use an earlier portion. The Beat parameter must be set to a value of 0 or higher for this to work.

For example, suppose you've chosen a six-second sample as Sample 1, and you use the entire sample as the segment. You also select a half-second segment of Sample 2 to be mixed in. If you choose a Tempo value of 120 beats per minute (2 beats per second), there will be 12 beats within your six-second Sample 1 segment. If you set the Beat parameter to 1 and the Of parameter to 4, then Sample 2 will play on the first, fifth, and ninth beat of your six-second Sample 1 segment. The result will look something like the diagram below.



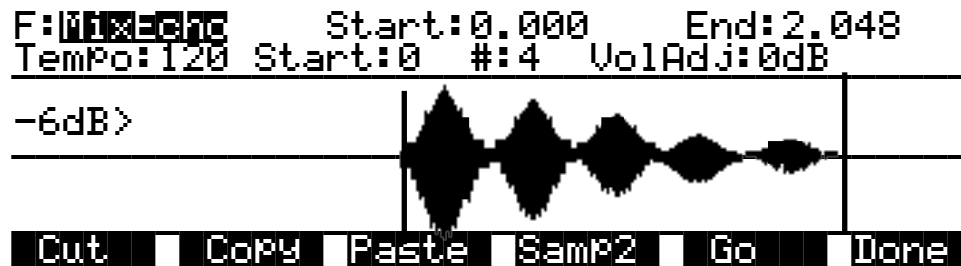
If you change Beat to 3, Sample 2 will play on beats 3, 7, and 11. If you set Beat to 1 and Of to 6, Sample 2 will play on beats 1 and 7. If you set Beat to -1, Sample 2 will still play on beats 1 and 7, but the sound of the sample will change each time, as the start point of the Sample 2 segment moves forward (later) by one beat each time it is mixed in.

### 17 Replicate



With the exception of two differences, the replicate function is similar to the Mix Beat function. The primary difference is that the replicate function uses the selected Sample 2 segment to overwrite (replace) the selected Sample 1 segment, instead of merging the two segments like the Mix Beat function. This makes it run faster. The other difference is that there is a crossfade parameter instead of a Volume Adjust parameter. The crossfade parameter lets you smooth the transition points from Sample 1 to Sample 2.

## 18 Mix Echo



This function operates much like Mix Beat, but instead of the Beat and Of parameters, you have Start and “#” parameters. The Start parameter sets the beat at which the selected Sample 2 segment begins being mixed with the selected Sample 1 segment. Sample 2 is repeated on every beat after this starting point. The “#” parameter determines how many times the Sample 2 segment is repeated.

The Volume Adjust parameter will affect the volume of each repetition of Sample 2. It sets the relative level of the last beat, and boosts or cuts each repetition by a proportionate amount from the starting level. This is a linear adjustment, measured in decibels, evenly spaced over the total number of mixed-in beats.

If you set the “#” parameter to a positive value, the adjustment in volume begins on the second mixed-in beat (the first beat plays at the initial level). If you set a negative value, the first mix of Sample 2 will be cut or boosted by the Vol adjust amount.

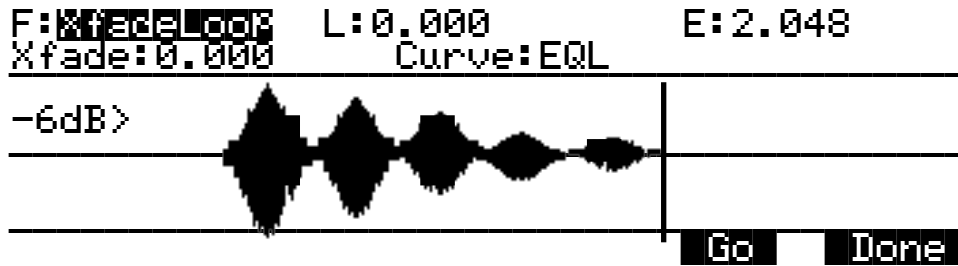
## 19 Beat Volume Adjust



This function works much like Mix Beat. The difference is that you’re not mixing in any samples, you’re just volume adjusting Sample 1 at regular intervals. Each beat is adjusted in volume by the amount specified for the VolAdj parameter. This is useful for mixing part of a sample within itself.

Set the Start and End positions of Sample 2 to define the length of time over which the volume will be adjusted. For example, if you had a six-second sample, and you set the Tempo to 120, the Beat parameter to 1, and the Of parameter to 1, you’d have twelve beats defined. If you then set the range of Sample 2 at a half-second, and set the VolAdj parameter to -3 dB, then you’ll insert 12 volume adjustments into Sample 1, each lasting a half second, and bringing the level down 3 dB.

### 20 Crossfade Loop (XfadeLoop)

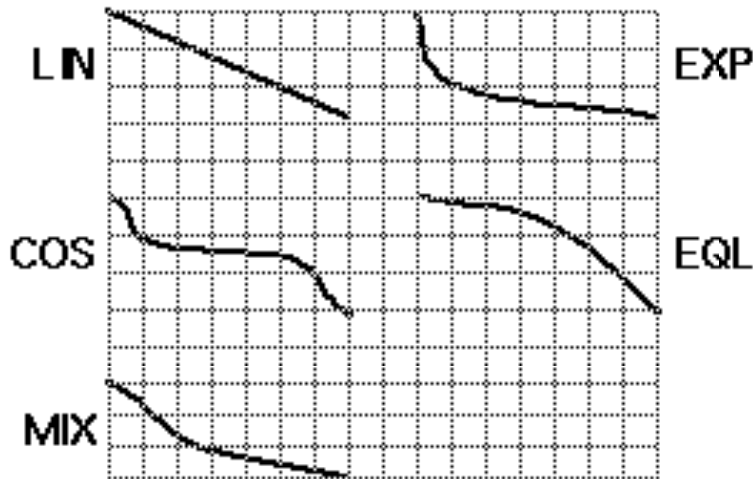


The Crossfade Loop function lets you create smoother loops by crossfading the beginning segment of the loop with a segment of equal length at the end of the loop. These segments can be defined by the Loop and End parameters as set on the TRIM or LOOP page for the current sample, or with the Loop and End points on the XfadeLoop page. Changing the Loop and End parameters on the XfadeLoop page will change them on the TRIM and LOOP pages, and vice versa. Using this function is equivalent to setting the loop on the LOOP page, but with the added feature of a crossfade at the loop transition point.

The Xfade parameter determines the length of the crossfade, while the Curve parameter sets the shape of the crossfade curve. The available values are LIN, EXP, COS, EQL, and MIX. These curves are described below.

### Crossfade and Volume Adjust Curves

There are five curves that can be applied to a number of DSP functions: LIN, EXP, COS, EQL, and MIX. The LIN curve is a straight linear curve, which will create an even cut or boost. The EXP curve is exponential, that is, gradual at one end and steep at the other. The COS curve is a segment of a cosine curve, which is relatively flat at both ends and steeper in the middle. The EQL curve steepens at an even rate, approximating an equal power fade curve. The MIX curve is a gradual curve that approximates manually dropping or raising the faders on a mix board. The diagram below shows each of the curves as it would be applied to a cut in amplitude.



## Reading Samples

The K2vx will load numerous samples from Akai, Roland, and Ensoniq EPS and EPS-16 Plus SCSI drives and floppies (including ASR-10 "Ensoniq" format), using Version 2's enhanced Disk mode operations. The displays you see will vary depending on the samples you're loading, but several features are the same. We'll describe the similarities first, then elaborate on the differences.

First, enter Disk mode by pressing the Disk mode button. Select the disk to be loaded from, as described in Chapter 13. The K2vx automatically recognizes the type of disk when you select it.

Press the Load soft button, and you'll see a page prompting you to select something to load (we'll call them objects, since different manufacturers give them different names). The top line of the display will tell you the number of objects available of the currently selected type, as well as the index number of the currently selected object. You can select any object in the list by typing its index number on the alphanumeric buttonpad and pressing Enter. The next step is to use the soft buttons to select the type of object to be loaded.

Once you've selected the type of object to load, press the OK soft button, and the bank dialog will appear, enabling you to select the bank into which the object(s) will be loaded. When you've selected a bank, press the OK soft button, and the loading process will begin. At the center of the display you'll see the object currently being loaded. The top line of the display will fill with asterisks to indicate the status of the current object. The bottom line will tell you the total number of kilobytes to be loaded.

The K2vx will create layers as necessary when you load objects. These layers have the same settings as Layer 1 of Program 199.

When the load is complete, the Disk mode page will reappear. You can now proceed with another load, or go to any other mode. If you exit Disk mode, the K2vx will remember the file that you selected most recently. When you return to Disk mode, this file will be selected.

Once you've loaded a sample or program file, you can save it as a Kurzweil object. You'll find it it can be loaded and backed up much faster as a Kurzweil object than in its original format.

### Akai

The first page to appear is the page for loading files. The soft buttons name the operations: HDrive, Volume, and File on the left, and OK and Cancel on the right.

The hierarchy of objects is shown by the three soft buttons on the left. The display prompts you: "File to load:" The HDrive button selects the partition on the currently selected disk. The Volume button selects volumes within the currently selected partition. The File button selects an individual sample file from within a volume. The OK button, toward the right, executes the displayed function: partition selection, or loading a volume or file. The Cancel button returns you to the Disk mode page.

When you press the HDrive button, the center of the display's top line shows the currently selected volume in the currently selected partition. The prompt at the center of the display will read: "HD Partition." The list of available partitions will appear following the prompt. They're usually named A through F. Use the cursor buttons or numeric entry to highlight a different partition. Pressing the OK soft button will select the highlighted partition.

Pressing the Volume button will change the prompt to "Volume to load:" The list of available volumes in the current partition will appear. The center of the top line will show the current partition. The Layer buttons will scroll through the list of available partitions. Use the cursor buttons or numeric entry to select a different volume. Pressing the OK button will load the entire highlighted volume, unless the volume is larger than your available sample RAM, in which case, the K2vx will load as many files as will fit.

The Bank dialog will appear, enabling you to select the bank that will receive the volume. Press OK again, and you'll be prompted to press either the Progs soft button, which will load program information in addition to the samples, or the Samps soft button, which will load only the sample information. Programs are identified by the suffix ".p," and are stored in program

RAM. Samples have the suffix “.s,” and are stored in sample RAM. You can press Cancel to return to the Disk mode page without loading the volume.

If you load sample objects, you’ll see the following prompt: “Create preview program/keymap?” If you answer Yes, the K2vx will load the samples into a program that it creates based on Layer 1 of Program 199.

Loading program objects will load multi-layer samples and keymaps, and sometimes velocity switches. The K2vx will create layers as necessary when you load program objects. These layers have the same settings as Layer 1 of Program 199. In some cases, the K2vx will also create stereo keymaps to preserve the separation of stereo samples.

If you press the File button, the prompt will change to “File to load:” You can view the list of files with the cursor buttons, or use numeric entry. The top line of the display will show the currently selected volume. Select different volumes with the Layer buttons. The size of the currently selected file, in Kilobytes, is shown just above the soft buttons on the left. Press the OK button to load the highlighted file.

Press OK, and the Bank dialog will appear. Press OK again, and the file will be loaded into the highlighted bank.

Sometimes you may find an Akai floppy disk that your K2vx can’t read. In this case, check the format of the disk. Akai allows you to format a double sided-double density (DD) disk for high density (HD). This is not standard in the computer industry, since DD disks are not verified for use as HD disks. If this is the case, you will need to load the disk into an Akai, then save it out to a properly formatted disk.

## Roland

For Roland disks, the hierarchy is a bit different; the objects that can be loaded are called Volumes, Performances, Patches, and Samples. The page that was selected last time a SCSI load was executed will appear when you initiate the load operation. Following the prompt is the list of available objects, with the size of the object in Kilobytes displayed as well. The top line of the display will show the number of available objects of the selected type.

Use the soft buttons to highlight the object to be loaded. The layer buttons will take you through the current object list in increments of 100. Press OK to execute the load. The Bank dialog will appear. Press OK again, and the object will be loaded. The display will update you on the progress of the load.

## EPS

For EPS disks, the hierarchy consists of files and directories. Directories can be nested several layers deep. When you press the Load soft button, you’ll be prompted to select a file or directory to load from the list of available files and directories. The currently highlighted object will be either a file or a directory. If it’s a file, its name and size will be shown following the prompt. If it’s a directory, its name appears, followed by “(dir)” to indicate its type. The Layer buttons will take you to the first and last files of the currently selected directory. Note that if you’re loading from floppy disks, you won’t see any of the displays or soft buttons relating to directories, since the directory architecture is not supported for floppies.

When a file is highlighted and you press OK, the Bank dialog appears; press OK again to load the file. When a directory is highlighted and you press OK, you enter that directory, and the list of files and subdirectories in that directory appears, each file followed by its size, and each subdirectory, if any, followed by “(dir).” The top item in every list you select is always the parent directory of the files below it. Select the top item in a list to go up one directory level.

Pressing the Exit soft button will take you one level back up the hierarchy. Pressing it repeatedly will take you to the root directory—the directory at the top of the hierarchy. The quickest way to the root directory is to press the Root soft button. The top line of the display shows you the name of the currently selected directory (or subdirectory).

Pressing the All soft button will load all files in the current directory (but not any subdirectories). The Bank dialog will appear, and when you press OK, you’ll be prompted to



press the Progs button to load program information in addition to the samples, or the Samps button to load only the samples.

## The Keymap Editor

The Keymap Editor lets you customize the K2vx's factory preset keymaps and save them to RAM. You can also build your own keymaps from scratch.

Keymaps are an integral part of every layer of a program. Each keymap contains a set of parameters determining which sample(s) the K2vx will play when you trigger a note. Each layer has at least one keymap, but it can have two keymaps when you're working with stereo samples. Each of these stereo keymaps uses two of the 48 available voices.

Each keymap consists of a set of key (note) ranges—C 4 to G 4, for example. The entire span of each keymap is from C 0 to G 10. Each range has a sample root assigned within the range. Each sample root is a distinct ROM or RAM sample. Within each key range, the sample root is transposed up and down to play on each of the range's notes. You can view each range by changing the value of the Key Range parameter on the Keymap Editor page. You can mix samples of different timbres within a single keymap, and even tune individual keys to any pitch by defining key ranges to single notes and assigning samples to each of those notes.

When you trigger a note, the K2vx identifies the key range where the Note On event occurred. It also checks the attack velocity value of the note. It then addresses its memory, and retrieves the sample root that's assigned to that key range and attack velocity value. If the note that's triggered is not the note where the sample root is assigned, the sample is transposed to play at the correct pitch. The K2vx then generates the digital signal that represents the sound of the note. At this point the keymap's job is done, and the signal proceeds through the layer's algorithm and on to the audio outputs.

You can assign as many key ranges to a keymap as you like, even creating a separate range for each note. This would allow you to tune each key independently, to create microtonal tunings. For keymaps that use a single timbre, like the Grand Piano, there's a key range for each sample root stored in memory. For acoustic instrumental sounds, the more key ranges you have for a keymap, the more realistic the sound will be, since there will be less pitch shifting of the sample root within the key range.

Of course, you can assign sample roots with different timbres within the same keymap. Many of the drum kit keymaps in ROM, for example, have about 20 key ranges, with several different timbres assigned as the sample roots. You can also create a keymap with a single key range that spans from C 0 to G 10, if you want to stretch a single sample root from C 0 to G 10. Keep in mind, however, that samples can be transposed upward only to a limit of 96 KHz for the playback rate. For 48K samples, that's an octave of upward transposition. Samples can be transposed downward without limit.

Think of a keymap as if it were a single piece of string, divided into different sections that adjoin one another. Sections can not overlap. If you have one range that goes from C4 to F4 and another that goes from F#4 to C5, then if you change the first range to be C4 to G4, the second one will change to be G#4 to C5.

Also, you can't have "nothing" assigned to a key range. Even if it is Silence (#168), there will always be a sample assigned to every range in the keymap. This is something to watch out for when creating drum programs. For example, let's say you are creating a program with 20 layers. Each layer has its own keymap, which has just one sample assigned to part of the keyboard with the rest of the key range assigned to Silence. Make sure that you limit the note range of each layer using the LoKey and HiKey parameters on the Layer page in the Program Editor. If each layer covers the entire range, then each note you played would trigger 20 voices (one for each layer). You would only hear one drum per note because all the other layers are triggering "Silence". Because of the voice stealing algorithms in the K2vx, the voices would almost

immediately become available again, since they have no amplitude. But for one brief instant, the voice would be triggered, which could cause other voices to be cut off.

You can also create multi-velocity keymaps—that is, keymaps that will play different timbres depending on the attack velocities of your Note On events. The program Dual Electric Piano, for example, uses a keymap with two velocity ranges. Each key range in a multi-velocity keymap contains two or three distinct sample roots that the K2vx chooses between, according to the attack velocity of the note. When in the Keymap Editor, you can select the different velocity ranges within a keymap using the CHAN/BANK buttons. To create your own multi-velocity keymaps, start in Program mode by selecting a program that contains a multi-velocity keymap. Most of these programs are identified by the word “Dual” in their names. Then select the KEYMAP page, and press EDIT to enter the Keymap Editor. From there you can change the sample root assigned to each velocity level in each key range. The description of the Velocity Crossover parameter on page 15-36 has more information.

The Keymap Editor is nested within the Program Editor. The first step in using the Keymap Editor is to select the keymap you want to edit. This is done on the KEYMAP page in the Program Editor, using the Keymap parameter. Once you’ve done this, just press the EDIT button, and you’ll enter the Keymap Editor. If you want to edit a different keymap, return to the KEYMAP page in the Program Editor and select the desired keymap. If you want to build a keymap from scratch, start with the keymap 168 Silence (see “Building a Keymap” on page 15-37). This keymap template contains one key range from C 0 to G 10, and is a convenient starting point for adding key ranges and assigning sample roots. The Keymap Editor page looks like this:

```

EditKeymap      <>VelocityRange:PPP-fff
MasterXPose    :0S1
Key Range      :C 0-A 1   Lo:C 0   Hi:A 1
Sample         :1 Grand Piano-G#1
Coarse Tune    :0ST
Fine Tune      :0ct
VolumeAdjust   :0dB      VelCrossover:None
Name Save     Delete Dump NewRng Assign
    
```

PARAMETER	RANGE OF VALUES
MASTER TRANSPOSE	-128 ST -127 ST
KEY RANGE	Variable
LOW KEY	C 0 to G 10
HIGH KEY	C 0 to G 10
SAMPLE	Sample Root list
COARSE TUNE	-120 to 60 semitones
FINE TUNE	-49 to 50 cents
VOLUME ADJUST	± 48 dB
VELOCITY CROSSOVER	None, ppp to fff

The top line of this page tells you which velocity range you’re currently looking at. If the current keymap is a multi-velocity keymap, the CHAN/BANK buttons let you select between the velocity ranges. The velocity range is set with the Velocity Crossover parameter(s), described on page 15-36.

### The Soft Buttons in the Keymap Editor

The first four soft buttons execute the basic library functions, enabling you to name, save, or delete the current keymap, or dump it via a MIDI SysEx message.

### **New Range (NewRng)**

The NewRng button lets you define a range to edit, whether it's to assign a different sample, or to adjust the pitch or volume. Just press NewRng, then trigger the note you want as the low note, then the high note. The K2vx will prompt you for each note. When you trigger the high note, you'll return to the Keymap Editor page, and the edit range you defined will be selected. The next change you make will affect only that edit range.

There's more than one way to use this function. If you set an edit range that's completely within an existing key range, you can modify the edit range without affecting the rest of the key range or the adjacent key ranges. If you set an edit range that overlaps part or all of another key range, the sample assigned to the lower key range will be applied to the entire edit range. This is an easy way to define a new key range that replaces one or more existing key ranges.

### **Assign**

The Assign soft button lets you select a sample, then specify the key range to which it's assigned. This enables you to insert a new key range within the current keymap. When you press the Assign soft button, a dialog appears that prompts you to select a keymap from the Keymap list. Scroll through the list, then press the OK soft button. You'll then be prompted to define the new key range by triggering the notes you want to be the lowest and highest notes of the range. (Press the Cancel soft button if you change your mind.) When you trigger the low and high notes, the new key range is inserted. If the new key range partially overlaps an adjacent key range, the existing key range will be adjusted to accommodate the new range. If the new key range completely overlaps an existing key range, the original key range will be replaced.

## **The Keymap Editor Parameters**

### **Master Transpose**

This parameter does not really pertain to the keymap itself. Instead it is identical to the Transpose parameter found on the MIDI Transmit page. If you change the value here, the same value will be reflected on the MIDI Transmit page, and vice versa. It transposes the entire instrument globally. The reason it is placed on this page is that it will allow you to assign samples across the entire keyboard easily, when you are using a keyboard which has fewer than 88 notes.

As with the Transpose parameter on the MIDI Transmit page, if you have the rack, this parameter will not transpose without using the Local Keyboard Channel parameter (refer to page 10-6 for more on the Local Keyboard Channel).

The transposition amounts from this page and the Master page are cumulative. For example, if you set this parameter to a value of +1ST and then set the Transpose parameter on the Master page to -1ST, the two will cancel each other out, and the K2vx will actually be back at 0ST.

### **Key Range**

This parameter shows you which key range you're currently viewing or editing. Changing the value of this parameter selects a different key range, and shows you the sample assignment of that key range, as well as other parameters related to that key range. Use this parameter to move from one key range to another within the keymap, and assign different samples without adding new key ranges, as well as editing the transposition, tuning, volume and velocity crossover (if any) of the sample root assigned to the key range.

### Low Key (Lo), High Key (Hi)

With these parameters you can use any of the data entry methods to change the low and high notes of the current range. These parameters let you extend or shorten the width of a key range. You can extend a key range to the full capacity of the K2vx (C 0 to G 10).

The setting for the low key cannot be higher than the setting for the high key. Similarly, the setting for the high key cannot be lower than the setting for the low key.

### Sample

This is where you assign a sample root to the current key range. For RAM samples, each sample will have its own ID number. For ROM samples, it's a little different. Because of the large number of ROM samples, we did not want to use up too many of the available IDs. So we have organized them as follows. Each sample's name consists of three parts—for example, "1 Grand Piano G#1." The first numeral indicates what we call the sample block, which represents groupings of similar sample roots in memory. Next comes the name of the sample, which describes the sample's timbre. The final part of the sample's name refers to the pitch at which it was originally sampled. For many timbres, multiple samples are made at various pitches. As you scroll through the Sample list, you'll see only the pitch of the sample change until you reach the next sample block.

Highlight the Sample parameter, hold the ENTER key, then play a note on your controller to see the complete name of the sample played by that note. For example, when you play middle C on a piano program you'll see a name such as "1 Grand Piano C4". Move down the keyboard an octave and the sample will be "1 Grand Piano G#2".

### Coarse Tune

Coarse Tune allows you to transpose a sample for a given range. This is extremely useful when you have set the Root key of the sample for one note but want to assign the sample to a different part of the keyboard and still be able to play it without transposition. For example, if you originally set the Root key at C4 but want the sample assigned to C3, you would set Coarse Tune to 12ST, transposing it up one octave. Now the original pitch will play at C3, one octave down. If you examine the drum and percussion kit keymaps in ROM, you will see that we have done this. Most of our ROM drum samples have the Root key set at C4.

### Fine Tune

This gives you further pitch control. Once the sample's pitch is close to the desired level, use the Fine tune to sharpen or flatten it as much as a half-semitone.

### Volume Adjust

Here you can adjust the volume of the notes in the current key range. This enables you to make each key range play at the same volume even if the samples in the various ranges were recorded at different volumes.

### Velocity Crossover (VelCrossover)

This parameter applies only when the keymap assigned to the currently selected program is a multi-velocity keymap. The name of the keymap usually indicates whether it's of the multi-velocity variety (*Dual Electric Piano*, for example). Multi-velocity keymaps have a predetermined number of velocity levels, each of which can be assigned a different sample.

The K2vx supports keymaps with one, two, or three velocity levels. You can't add velocity levels to existing keymaps; if you want to create your own multi-velocity keymaps, select an existing multi-velocity keymap in the Program Editor before entering the Keymap Editor. Then you can select the different velocity levels with the CHAN/BANK buttons, and assign samples to the different levels. The currently selected velocity range is shown in the top line of the

display. Keymaps 2, 10, and 53 are dual-velocity keymaps. Keymaps 164–166 are triple-velocity keymaps.

When the current keymap is a single-velocity keymap, the VelCrossover parameter does not appear on the Keymap Editor page. When the current keymap is dual-velocity, the value for the Crossover parameter will be one of the eight dynamic markings from ppp to fff. The K2vx translates each of your Note Ons into one of these dynamic values, using the settings for the VelTouch or VelocMap parameters. When this translated value exceeds the setting for the VelCrossover parameter, the K2vx plays the sample assigned to the upper velocity range.

When the currently selected keymap has three velocity ranges, the VelCrossover parameter becomes two parameters: LowCrossover and HiCrossover. The K2vx plays the sample assigned to one of these ranges depending on the translated value of each note's attack velocity.

## Building a Keymap

If you used the Keymap Editor to enter the sampling page, then just press Exit from the Sampling page and you are ready to begin creating a keymap. If you entered the Sampling page from from Master Mode, do the following. Start in Program mode, and select Program 199, the Default program. Press the EDIT button, and you'll enter the Program Editor. Press the KEYMAP soft button, and the KEYMAP page will appear. The Keymap parameter will be automatically selected. Press 1, 6, 8, ENTER on the alphanumeric pad to assign the keymap "Silence." This isn't absolutely necessary, but it makes it easier to recognize the key ranges that have samples assigned to them when you start assigning samples. You can actually choose any program you want to start with, but by choosing these, you are starting with a "blank slate".

With the Keymap parameter still selected, press the EDIT button, and you'll enter the Keymap Editor. The Key Range parameter will be automatically selected, and you see its values: C 0 to G 10 (the entire MIDI keyboard range). The Sample parameter will have a value of 168 Silence C 4.

Now you're ready to start assigning samples to key ranges within the keymap. We'll assume that you've loaded samples with roots at C 1, C 2, C 3, etc. and that you plan to assign a root to each octave. To begin, press the Assign soft button. The display will prompt you to select a sample. Use the Alpha Wheel to scroll to one of your samples, or type its ID on the alphanumeric pad and press ENTER. When you've found the sample you want to use, press the OK soft button. The display will say "Strike low key..." Trigger A 0 (MIDI note number 21, the lowest A on a standard 88-note keyboard) from your MIDI controller. The display will change to say "Strike High Key..." Now trigger F 1 (MIDI note number 29) from your MIDI controller. The display will return to the Keymap Editor page. The Key Range parameter will show A 0–F 1, and the Sample parameter will show the sample you selected when you started the range assignment.

One more time...Press the **Assign** soft button. Select another sample root at the prompt, and press the **OK** soft button. Now trigger F# 1 for the Low Key prompt, and F 2 for the High Key prompt. At this point you've defined two key ranges, the first from A 0 to F 1, and the second from F# 1 to F 2. You can repeat the process as many times as you want, creating a new key range each time.

Once you have your samples assigned, you may need to transpose them so that they play back at the correct pitch within the range you have chosen. To do this, highlight the Keyrange parameter, scroll to the range you need, then highlight the Coarse Tune parameter. Adjust Coarse Tune to bring the sample to the proper pitch within that keyrange. Then scroll back up to the Keyrange parameter, select the next range, and continue as needed.

Here's a fairly important point that may or may not affect your keymap construction. Suppose you want to build a keymap that uses the same sample in several adjacent key ranges, and you plan to add a bit of detuning to the samples in each range. You might think that you could build

the keymap first, then go into the Sample Editor and tweak the samples when the keymap is finished. Yes, but...

Suppose you used the technique we described above to assign a vocal sample whose root was C 4 to a key range from A 3 to E 4. Then you assigned the same sample to a key range from F 4 to B 4. You might be surprised to find that when you finished the F 4–B 4 key range and the Keymap Editor page reappeared, the current key range would not be F 4 to B 4, but A 3 to B 4! This is because the K2vx automatically merges adjacent key ranges that are identical (this is done to save memory). Therefore, some parameter must be different in each adjacent key range you create if you want to build keymaps using the technique we just described. So if you want to use the same samples in adjacent key ranges with, for example, minor pitch or volume modification, you should make those changes to the current sample on the Keymap Editor page before assigning the next range.

## Chapter 16

# FUNS

The name “FUNs,” although it sounds like a pun, is simply the best abbreviation for “Function.” You’ll definitely have a good time with them, however, if you’re looking for a variety of ways to control your sounds.

We’ve discussed various control sources throughout this manual, from the physical controls like the Mod Wheel to the software control sources like LFOs and attack velocity. You can assign them to affect your sounds in all sorts of ways.

The FUNs take the control sources one level further. By setting up a FUN as a control source, you can mix the signals of two control sources, and perform one of 50 functions on the combined signals. The result of that function becomes the new control source value. Because the FUNs can radically change the combined input values, the FUNs can have a profound effect on your sounds.

You may find that experimenting with the various FUN equations gives you a better idea of their effects than reading the explanations. Although there’s a great deal of mathematics behind the FUNs, the most important consideration is how they affect your sounds. The more you play around with them, the better you’ll understand how powerful they are.

## The Mechanics of Control Sources

We’ll return for a minute to the notion that the K2vx is an integrated system consisting of a MIDI-driven sound engine and a MIDI-driven effects processor. The sound engine responds to MIDI messages received at the MIDI In port and from the front panel, as does the effects processor.

The K2vx’s control sources use their own internal signal format for interpreting control messages and communicating them to the sound engine. Every control source sent from your MIDI controller to the K2vx’s sound engine is translated to a value in the range from -1 to +1. This consistency enables the sound engine to process control source signals very efficiently. Conversely, the K2vx’s internal control source signals are translated to MIDI values before being sent to the MIDI Out port.

A control signal value of 0 represents minimum effect; it’s equivalent to the control source being turned off or disconnected. A control signal value of +1 represents the maximum positive effect of a control source, while a value of -1 represents the maximum negative effect of a control source.

### Unipolar and Bipolar Control Sources

There are two kinds of control source signals: unipolar and bipolar. A unipolar signal has a value between 0 and +1. A bipolar signal has a value between -1 and +1.

A switch pedal is unipolar; its control signal value will never go below 0. Since it’s a switch control, it has only two possible values: 0, which corresponds to off or minimum, and +1, which corresponds to on or maximum. When you depress your MIDI controller’s sustain pedal, for example, it sends a control signal value of +1 to the K2vx’s sound engine.

Continuous controls can be unipolar or bipolar. Consider your MIDI controller’s Mod and Pitch Wheels as examples. Normally, the Mod Wheel affects the K2vx as a unipolar control

source; it sends a control signal value that's interpreted as 0 when it's fully down, and values interpreted between 0 and +1 as you push it up. When fully up, it sends a value that's interpreted as +1. It can be used as a bipolar control source by assigning a value of Bi-Mwl to any control source parameter.

The Pitch Wheel is normally bipolar; it sends a control signal value that's interpreted as 0 when it's centered, values interpreted between 0 and -1 as it's pulled downward, and values interpreted between 0 and +1 as it's pushed upward. It can be used as a unipolar control source by assigning a value of AbsPwl to any control source parameter.

The FUNs can act as unipolar or bipolar control sources; it depends on the values of the input signals and the nature of the function you choose. Depending on the function you choose to process the input signals, the output signal value can exceed +1 or -1. Normally the signal merely pins at +1 or -1; that is, it won't go any higher or lower. In some cases, however, the output signal value is wrapped around instead of pinning; we'll mention these cases as we get to them. You can assume that the output signal values of the functions listed below will pin at -1 or +1, unless specified otherwise.

## Programming the FUNs

Start by entering the Program Editor, then use the soft buttons to select the FUN page. Setting up a FUN as a control source is a two-step process: assigning a FUN as the value for one or more control source parameters in the Program Editor, then programming the FUN on the FUN page, by assigning control sources to two inputs—**a** and **b**, and choosing a function (equation) that will process the combined signals from **input a** and **input b**.

```

EDIT:PROG:FUN          <>Layer:1/1
FUN1:  Input a: Input b: Function:
FUN1:  OFF      OFF      a+b
FUN2:  OFF      OFF      a-b
FUN3:  OFF      OFF      (a+b)/2
FUN4:  OFF      OFF      a/2+b
<more  LFO  ASR  FUN  UTRIG  more>
    
```

There are four FUNs; you can combine and process four different pairs of control source signals. FUNs 1 and 3 are always local, that is, they affect each note in their respective layers independently. FUNs 2 and 4 are local by default, but they can be made global by setting a value of On for the Globals parameter on the COMMON page in the Program Editor. A global FUN affects all notes in its layer equally and simultaneously.

The best way to understand the use of the FUNs is to set up a simple test model, then plug in the different equations and listen to their effects. We'll walk you through the programming of a FUN and assigning it to control pitch. Then you can scroll through the list of equations at your leisure.

Start in Program mode and select program 199. Press EDIT to enter the Program Editor. Select the KEYMAP page, and change the keymap to 152 Dull Sawtooth. Then select the PITCH page, and assign a value of FUN1 for the Src1 parameter (a shortcut is to press 1, 1, 2, ENTER on the alphanumeric pad). Select the Depth parameter and change the value to 1200 cents. Next, select the FUN page, and select the **Input a** parameter for FUN1. Assign a value of MWheel (the quickest way is to hold the K2vx's ENTER button and move your MIDI controller's Mod Wheel). Next, select the **Input b** parameter for FUN1, and assign a value of Data. This assumes your MIDI controller either has a data slider, or a programmable control that you set to send Data messages (MIDI 06). If you don't have a data slider or a programmable control, you can set the value of **Input b** to AbsPwl, and use your Pitch Wheel to control **Input b**. If you do this,



you'll need to go to the LAYER page and set the PMode parameter to a value of Off to keep Pitch Wheel messages from interfering with the test model.

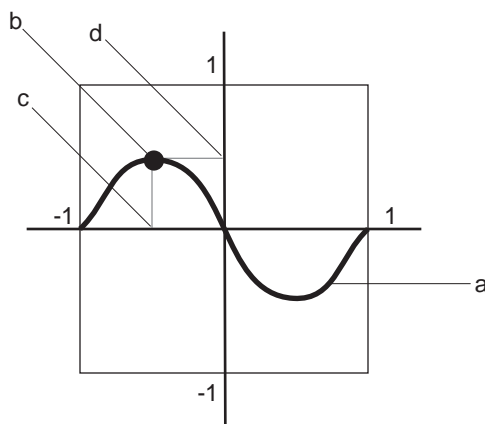
Now select the Function parameter, and scroll through the list of equations. Move your MIDI controller's Mod Wheel and Data slider as you play, and listen to their effects. Actually listening to the various effects while reading the explanations below will help your understanding. In the model we've set up here, **inputs a** and **b** are both unipolar. The effect of each equation will differ depending on the type of controls you assign to the inputs. There are four possible combinations: both inputs unipolar; both inputs bipolar; **input a** unipolar with **input b** bipolar; **input a** bipolar with **input b** unipolar.

## The FUN Equations

In this section we'll describe how each of the FUN equations works. In some cases, a small graph will accompany the explanation. Here's how to interpret the graphs.

Each graph shows a curve illustrating the effect of the equations on the input signals. The horizontal axis represents the possible values of the input to the FUN (the combined control signals of **inputs a** and **b**). The vertical axis represents the possible values of the FUN's output signal. The four elements in the diagram below show you how to read these graphs:

- the curve representing the effect of the FUN's equation on every possible input value
- one point on that curve, representing a single input value and the corresponding output value generated by the FUN's equation
- the input value represented by the point
- the output value represented by the point



For any point on the equation's curve, you can determine the input value by tracing a line from the point to the horizontal axis. Similarly, you can determine the output value by tracing a line from the point to the vertical axis. For the point shown in the example above, the combined values of the control signals of **inputs a** and **b** equal about  $-0.5$ , which translates to an output value of  $+0.5$ . An input value of  $-1$  gives an output value of  $0$ , as do input values of  $0$  and  $+1$ . An input value of  $+0.5$  gives an output value of  $-0.5$ .

## The List of Equations

The first six equations are weighted sums and differences—that is, the signal values of **inputs a** and **b** are added to or subtracted from each other, and are divided in turn by various amounts to alter their effects relative to each other. These equations give you several different types of mixers for combining the signals of the two inputs.

### **a + b**

The values of **inputs a** and **b** are added, creating a simple mixer. For example, you might have LFO1 assigned for the Src2 parameter on a layer's PITCH page, and a FUN assigned for the DptCtl parameter. On the FUN page, if you set **input a** to a value of MWheel, and **input b** to a value of MPress, then this equation will let you modulate the depth of the LFO's pitch modulation with your MIDI controller's Mod Wheel or with mono pressure. You could set a fixed initial depth with the Mod Wheel and alter it further with mono pressure. In this case the output signal would pin at +1 or -1 fairly quickly.

### **a - b**

This operates similarly to the previous equation, but the value of **input b** is subtracted from the value of **input a**. This equation will reverse the normal effect of the control source assigned to **input b**. For example, if **input a** is off, and **input b** is assigned to a unipolar control source like MWheel, then the Mod Wheel will generate a control signal of -1 when fully down, and 0 when fully up.

### **(a + b) / 2**

The values of **inputs a** and **b** are added, and the sum is divided by 2. This gives you the same kind of control as the previous two equations, but the output signal will reach +1 or -1 half as often as with the equation  $a + b$ .

### **a / 2 + b**

The value of **input a** is divided by 2, and the result is added to the value of **input b**. **Input a** has half the effect of **input b**.

### **a / 4 + b / 2**

The value of **input a** is divided by 4, and the value of **input b** is divided by 2. The two results are added to give the output value. **Input a** has half the effect of **input b**, and the total result has half the effect of the previous equation.

### **(a + 2b) / 3**

The value of **input b** is multiplied by 2, and the result is added to the value of **input a**. This sum is then divided by 3. **Input a** has half the effect of **input b**, and the total result has somewhat more effect than the previous equation, but less effect than  $a + b$ .

### **a \* b**

The values of **inputs a** and **b** are multiplied. If you like using Src2 and DptCtl, this equation can be used to create a similar type of control source (it's equivalent to the Src2/DptCtl pair with the MinDpt parameter set to 0).

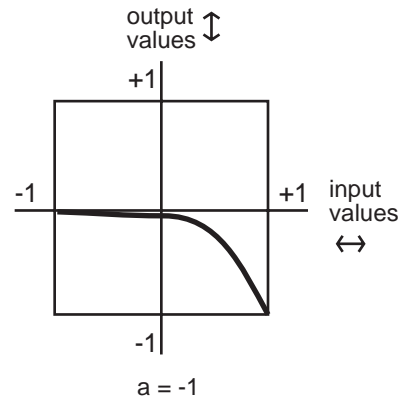
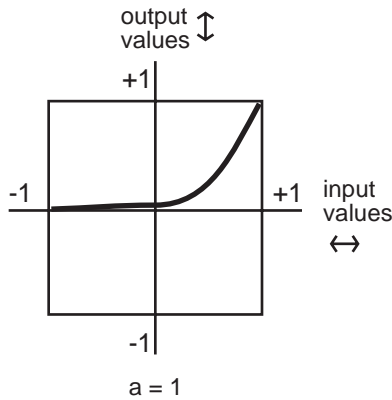
### **-a \* b**

The value of **input a** is multiplied by -1, then multiplied by the value of **input b**. This will reverse the normal effect of the control source assigned to **input a**. This equation also produces an effect like that of Src2 and DptCtl with the MinDpt parameter set to 0.

**a \* 10^b**

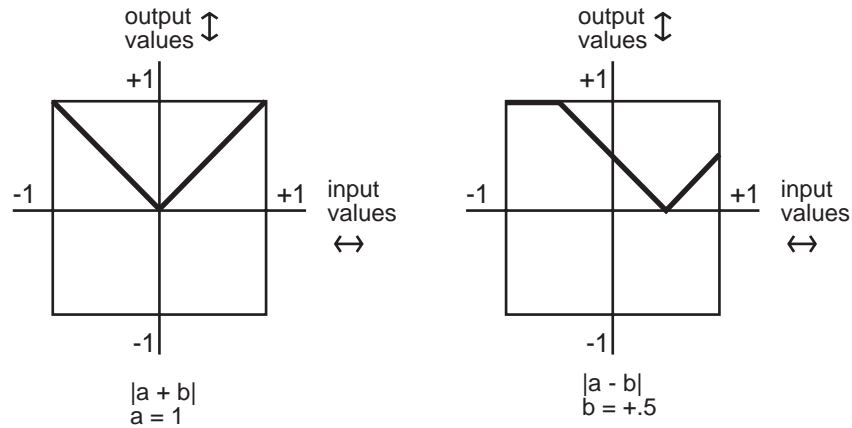
The actual equation is:  $a \times (10^{(2 \times b)} \div 100)$ . This is an exponential curve. 10 is raised to the (2 x b) power, then divided by 100. This result is then multiplied by a. Another way to express this is as follows: a change of 1 in the value of input b results in a hundredfold change in the output value. Here are a few possible output values:

Input values	Output values
a = +1, b = +1	+1
a = +1, b = 0	.01
a = +1, b = -1	.0001
a = 0, b = +1	0
a = 0, b = 0	0
a = 0, b = -1	0
a = -1, b = +1	-1
a = -1, b = 0	-.01
a = -1, b = -1	-.0001



**| a + b |**

The values of **inputs a** and **b** are added, and the absolute value of the sum is taken. If the sum is negative, it is multiplied by -1. This makes the FUN a unipolar control source. (See the illustration on the following page.)



**| a - b |**

The value of **input b** is subtracted from the value of **input a**, and the absolute value is taken. If the difference is negative, it is multiplied by -1. This also makes the FUN unipolar.

**min (a, b)**

The values of **inputs a** and **b** are compared, and the smaller value becomes the output value. This can be used to limit the value range of a control source. If, for example the value of the control source assigned to **input b** is left at +.5, then when the value of the control source assigned to **input a** is between -1 and +.5, its value will be used. As soon as its value exceeds +.5, the value of **input b** is used.

**max (a, b)**

This is the opposite of the previous equation. The values of **inputs a** and **b** are compared, and the larger value becomes the output value.

**Quantize b to a**

This turns the control source assigned to **input b** into a stepped control source. Instead of smooth transitions from minimum to maximum, it will jump from minimum to maximum in some number of equal steps. The number of steps is determined by the value of **input a**. The normal realtime application of this is to set a stationary value for input a to set the number of steps in the effect. Then use the control source assigned to **input b** as a realtime control to induce the stepped effect. Changing the value of **input a** in realtime will produce an extraneous (but possibly useful) effect.

Range of values for <b>input a</b>		Total number of steps as <b>input b</b> moves from min to max	
from	to	(when <b>input b</b> is bipolar)	(when <b>input b</b> is unipolar)
0	.0625	1 (no effect)	1
.0625	.125	2	1*
.125	.1875	3	2
.1875	.25	4	2*
.25	.3125	5	3
.3125	.375	6	3*
.375	.4375	7	4
.4375	.5	8	4*
.5	.5625	9	5

.5625	.625	10	5*
.625	.6875	11	6
.6875	.75	12	6*
.75	.8125	13	7
.8125	.875	14	7*
.875	.9375	15	8
.9375	1	16	8*

As an example, consider the FUN we set up at the beginning of the previous section: the Mod Wheel was assigned as **input a**, and the data slider as **input b**. The FUN was assigned as Src1 on the PITCH page, and the depth of Src1 was set to 1200 cents. If you push the Mod Wheel all the way up, the value of **input a** will be +1. This will set the number of steps at 8, since the data slider sends a unipolar control signal. With your MIDI controller's data slider at minimum, play and sustain a note. Then move the data slider slowly up. The pitch of the note will jump up an octave in 8 steps as you move the data slider all the way up.

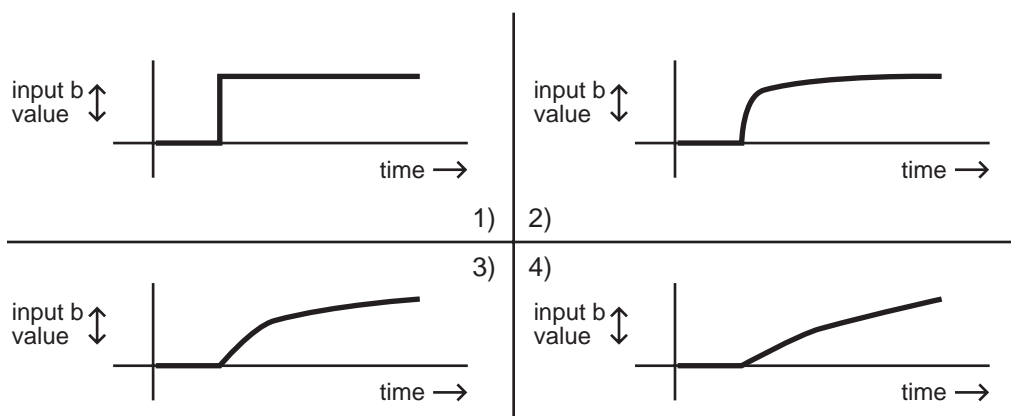
If the value of **input a** is negative, it's multiplied by -1, so its value always falls within the ranges above. When **input b** is bipolar and the resulting number of steps is an odd number, the steps are centered around a value of 0—that is, the center step is equivalent to no effect from **input b**. When the number of steps is even, a value of zero is not included in the steps. This is also true for the values marked by an asterisk when input b is unipolar.

### lowpass (f = a, b)

This equation might be called a lag equation. Its effect is to introduce a delay in the K2vx's response to changes in the value of **input b**. It works by filtering (reducing) higher values of **input b**. The value of **input a** determines the degree to which the values of **input b** are filtered. Low values for **input a** will induce a long lag when the value of **input b** changes. High values will shorten the lag. When **input b** remains constant at a high level, low values of **input a** will cause the FUN to sweep up slowly from 0 to the value of **input b**. Higher values for **input a** will cause the FUN to sweep more rapidly.

The four graphs below show the effect of different values for **input a** on the change of **input b**. In each graph, the value of **input b** jumps from 0 to +1. In graph 1, the value of **input a** is +1. Each successive graph represents the same change in the value of **input b**, at successively lower values for **input a**.

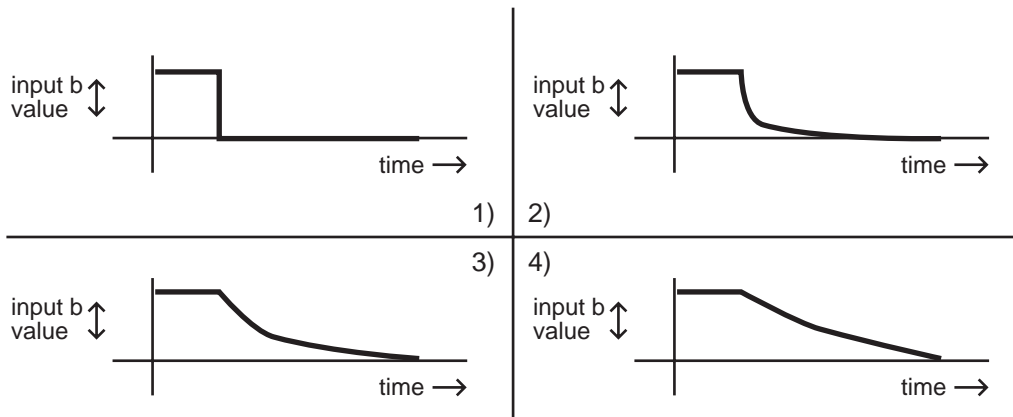
This equation works as intended only when the value of **input a** is 0 or positive. Negative values for **input a** will result in a much less predictable response than positive values. You might like the effect, but it won't be anything like what we've just described.



**hipass (f = a, b)**

With this equation the low values of **input b** are filtered according to the value of **input a**. This causes somewhat different results compared with the lowpass equation above. At low values for **input a**, low values for **input b** will have little effect, while high values for **input b** will cause the FUN to quickly reach full effect then slowly sweep down to its starting level. At high values for **input a**, a rapid change in the value of **input b** will have little effect. At low values for **input a**, rapid changes in the value of **input b** will cause the FUN to respond quickly to the change, then slowly fade back to minimum effect. Listening to the effects at different values for each input will give you the best understanding.

The four graphs below show the effect of different values for **input a** on the change of **input b**. In each graph, the value of **input b** drops from +1 to 0. In graph 1, the value of **input a** is +1. Each successive graph represents the same change in the value of **input b**, at successively lower values for **input a**.



**b / (1 - a)**

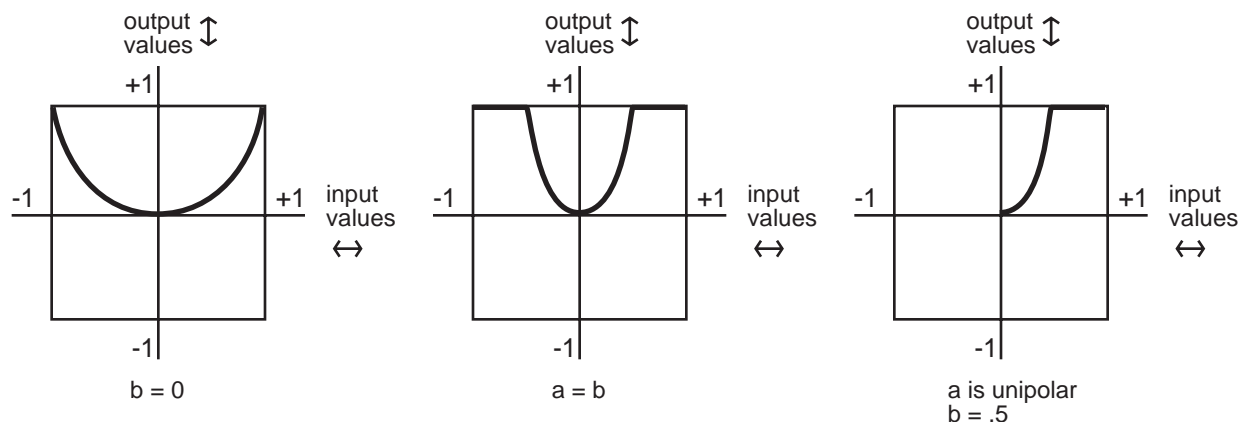
This is another weighted difference equation similar to the first six. The value of **input a** is subtracted from 1. The value of **input b** is then divided by the difference. You'll get considerably different results for different input values of **a** and **b**.

**a(b-y)**

Think of this equation as reading "y is replaced by the result of the function a(b-y)." The value of y indicates the value of the FUN's output signal. Every 20 milliseconds, the K2vx takes the current value of y, runs the equation, calculates a new value of y, and inserts the new value into the equation. Consequently the value of y will change every twenty milliseconds. Here's an example. When you play a note, the K2vx starts running the FUN. The first value for y is always 0. We'll assume the value of **input a** is +.5, and the value of **input b** is +1. The first time the K2vx evaluates the FUN, the result of the equation is .5 x (+1 - 0), or .5. So the FUN's output value after the first evaluation is .5. This becomes the new value for y, and when the K2vx does its next evaluation of the FUN, the equation becomes .5 x (+1 - .5), or .25. The resulting output value is .25, which becomes the new value for y. For the next evaluation, the equation is .5 x (+1 - .25), or .375.

**(a + b)^2**

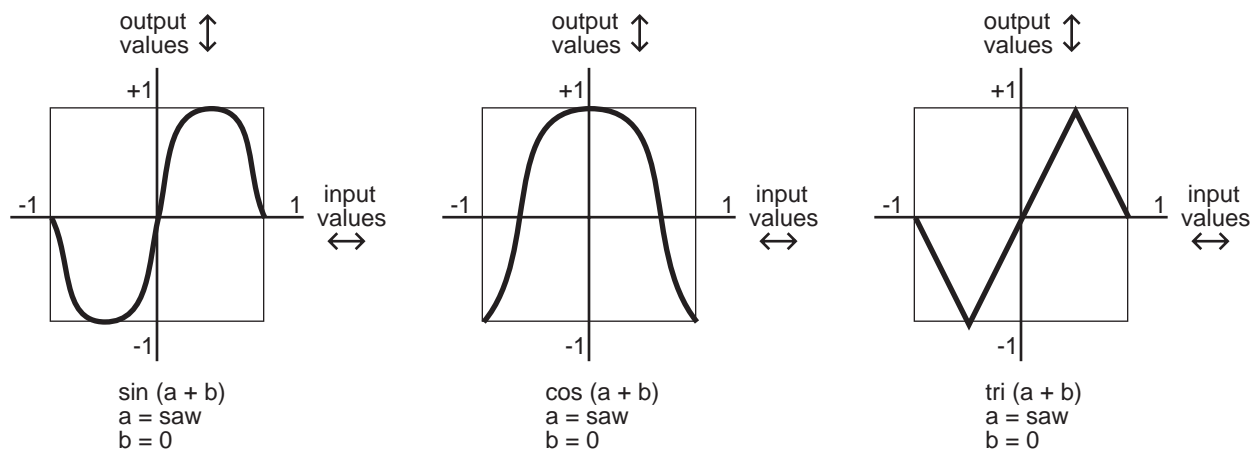
The values of **inputs a** and **b** are added, and the result is squared (multiplied by itself). This will change the linear curve of a unipolar control signal into a curve that's lower at its midpoint (by a factor of 2). Bipolar control signals will generate curves that are high at both ends, and 0 in the middle.

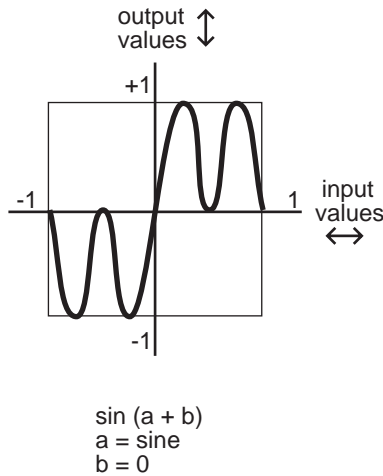


### sin (a + b), cos (a + b), tri (a + b)

These equations are intended to be used with inputs that are sawtooth waves—for example, **input a** might be LFO1 with its shape set as a sawtooth. Each equation will map a sawtooth-shaped input into a sine-, cosine-, or triangle-shaped output. Other input waveform shapes will result in outputs with more complex waveform shapes.

Other ways to get sawtooth shapes as inputs to these FUNs are to use other FUNs as the inputs, with their equations set as any of the ramp equations described later in this section (see the note at the end of this chapter about the evaluation order of the FUNs). You could also use LFOph1 or LFOph2 as inputs. The first three graphs below show the result of these functions when **input a** is a rising sawtooth wave, and the value of **input b** is 0. The fourth shows the result of the  $\sin (f=a + b)$  equation when the value of **input b** is 0 and **input a** is a sine wave.



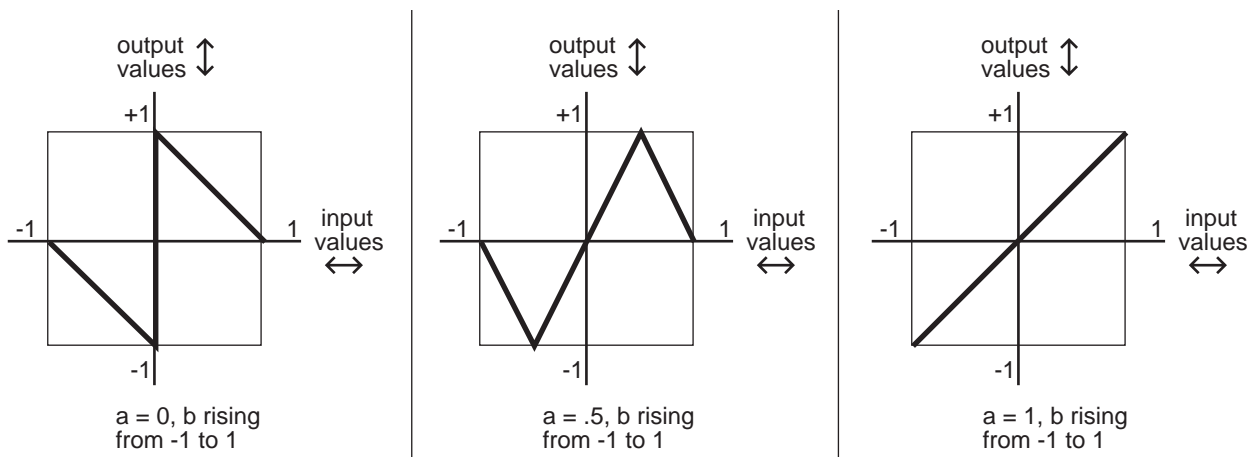


### Warp Equations

The next five equations all behave similarly, and are intended to be used as follows: the value of **input a** is the controlling value, and normally remains constant, although it doesn't have to. The value of **input b** is expected to change over time; **input b** might be an LFO, for example. The value for **input a** affects how the FUN calculates its output value while the value of **input b** changes.

#### warp1(a, b)

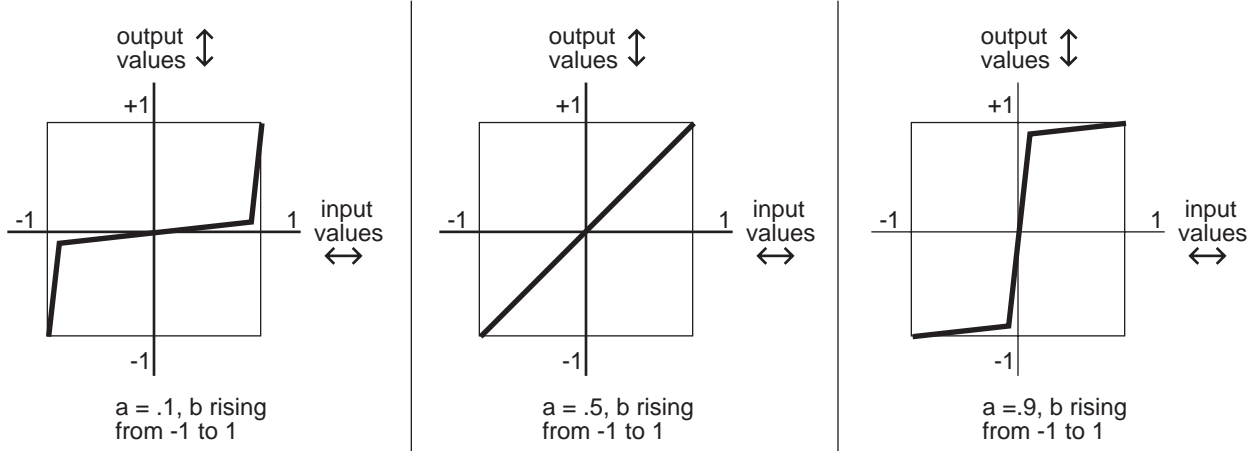
We call this the Vari-slope™ equation. The value of input a controls the mapping of values for **input b**. If **input b** is a sawtooth wave, different values for **input a** will change it into a triangle wave. If **input b** is a more complicated waveform, the output waveform is also more complicated.





**warp2(a, b)**

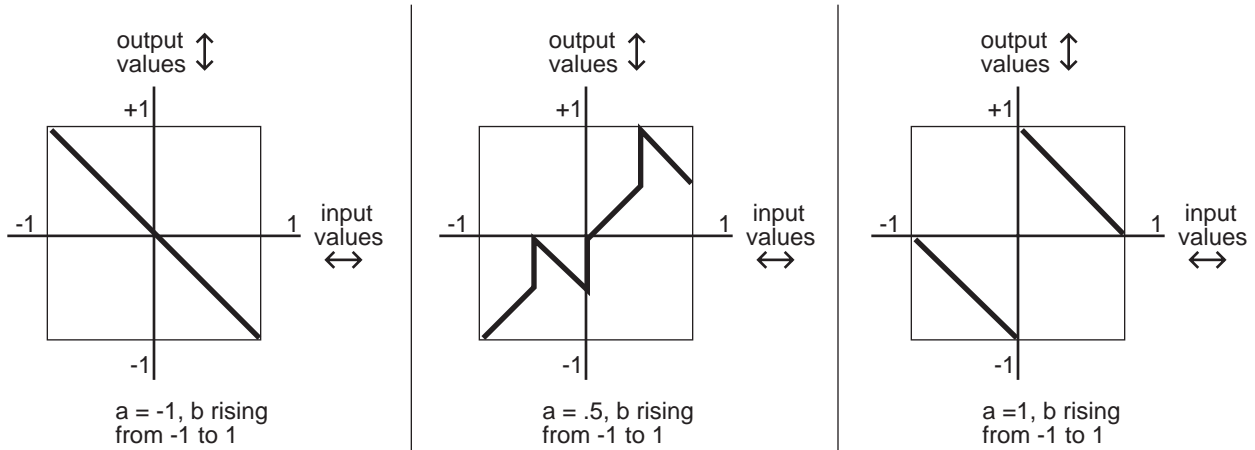
We call this equation Slant-square.™ Again, the value of **input a** controls the mapping of values for **input b**. If **input b** is a sawtooth wave, different values for **input a** will turn it into a number of variations on square waves.



**warp3(a, b)**

We call this one the Variable Inverter.™ It looks at the binary numbers that represent the values of **inputs a** and **b**, compares the corresponding bits in each number, and performs an XOR operation on them (we'll explain that below). The resulting number is converted into the output value. This can produce some erratic results, but if variety is what you're after, this equation will give it to you. You'll get your best results when **input b** is an LFO with a slow rate.

The XOR operation is a subprogram that applies a truth table to each of the digits in the binary numbers that represent the values of **inputs a** and **b**. Each of these numbers is a string of 16 digits (bits); each bit is either a 0 or a 1. The subprogram looks at the first bit of each number. If they're both 0s, the resulting value is 1. If one is a 0 and the other is a 1, the result is 1. If they're both 1s, the result is 0. This process is repeated for the remaining 15 bits of each number, and a new 16-bit number is generated. This number represents the output value of the FUN.

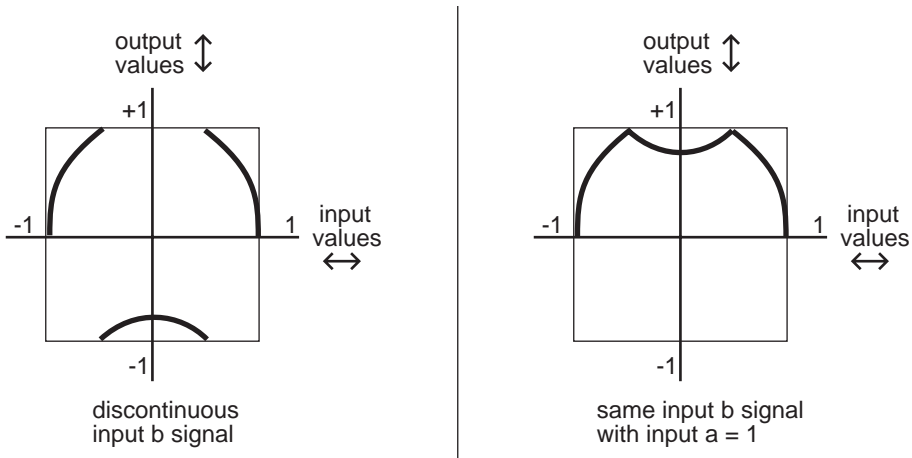


**warp4(a, b)**

This equation, the Period Inverter,™ is based on repeated evaluations of the value of **input b**. The K2vx compares each new value of **input b** with the value from the previous evaluation. If the absolute value (always a positive number) of the difference between the two is greater than the value of **input a**, the current value of **input b** is multiplied by -1.

The primary feature of this equation is that it will take a discontinuous signal and make it continuous. If, for example, FUN1 uses an equation like  $a(y + b)$ , its output can wrap around from +1 to -1, or vice versa. You could set FUN1 as **input b** for FUN2, set **input a** of FUN2 to ON (+1), and FUN2 would remove the discontinuity from the signal. The first graph below shows a hypothetical output signal with such a discontinuity, and the second shows how FUN2 in this case would make the signal continuous without drastically changing its shape.

If, on the other hand you *want* the signal to become discontinuous, you can use the warp4(a, b) equation in a single FUN, with **input a** set to OFF (0), and the signal would be multiplied by -1 with each evaluation of **input b**.



**warp8(a, b)**

This relatively simple equation is  $a \times b \times 8$ . If the result is beyond the range of -1 to +1, it wraps around from +1 to -1 (or vice versa), until it's within the allowable range. The table below shows some examples of how this works.

$a \times b \times 8 =$	final output value
-7.4	.4
-4.2	-.2
-1.8	.8
-.6	-.6
.4	.4
1.2	-.2
2.6	.6
5.4	-.4

**a AND b**

The values of **inputs a** and **b** are interpreted as logical quantities—they're considered TRUE if they're greater than +.5, and FALSE otherwise. This turns the FUN into an on/off switch. In the

model we set up in the previous section, FUN1 was set to control Src1 on the PITCH page, and Src1's depth was set to 1200 cents. With this equation, both **input a** (the Mod Wheel in this case) and **input b** (the data slider in this case) would have to be more than halfway up for the FUN to switch on. The pitch would jump 1200 cents as soon as both control sources moved above their halfway points. As soon as one of them moved below its halfway point, the pitch would jump back to its original level.

This equation can be used to trigger ASRs, or as a layer enable control, or for any control source that toggles on and off. If you set one of the inputs to an LFO, the FUN would switch on and off every time the LFO's signal went above +.5 (as long as the other input was also above +.5).

### **a OR b**

This equation is very similar to a AND b. The only difference is that the FUN will switch on when the value of either **input a** or **input b** moves above +.5.

## **Sawtooth LFOs**

The next six equations case the FUN to generate a sawtooth LFO as its output signal. Each performs a different operation on the values of **inputs a** and **b**, and the resulting value is multiplied by 25. The result determines the frequency of the LFO. If the value is a positive number, the LFO has a rising sawtooth shape. If the value is negative, the LFO has a falling sawtooth shape. When the resulting values are large (above 10 or so), the output waveform is not a pure sawtooth; a bit of distortion occurs.

### **ramp(f=a + b)**

The values of **inputs a** and **b** are added, then multiplied by 25.

### **ramp(f=a - b)**

The value of **input b** is subtracted from the value of **input a**, and the difference is multiplied by 25.

### **ramp(f=(a + b) / 4)**

The values of **inputs a** and **b** are added, and the sum is divided by 4. This value is multiplied by 25.

### **ramp(f=a \* b)**

The values of **inputs a** and **b** are multiplied, and the result is multiplied by 25.

### **ramp(f=-a \* b)**

The value of **input a** is multiplied by -1, then multiplied by the value of **input b**. The result is multiplied by 25.

### **ramp(f=a \* 10^b),**

10 is raised to the power of **b**, then multiplied by the value of **input a**. The result is multiplied by 25.

## **Chaotic LFOs**

The next five equations function somewhat like the equation  $a(b-y)$  described earlier, in that they start with a value of 0 for  $y$ , evaluate the equation, and use the result as the new value of  $y$  for the next evaluation. Although they all can function as LFOs (they can have a repeating cycle of output values), they can become chaotic depending on the input values.

**$a(y + b)$** 

The values of  $y$  and  $b$  are added, then multiplied by the value of  $a$ .

 **$ay + b$** 

The values of  $a$  and  $y$  are multiplied, then added to the value of  $b$ .

 **$(a + 1)y + b$** 

1 is added to the value of  $a$ . The sum is multiplied by the value of  $y$ . The result is added to the value of  $b$ .

 **$y + a(y + b)$** 

The values of  $y$  and  $b$  are added. The sum is multiplied by the value of  $a$ . The result is added to the value of  $y$ .

 **$a |y| + b$** 

The absolute value of  $y$  is taken (if it's a negative value, it's multiplied by -1). The absolute value of  $y$  is multiplied by the value of  $a$ . The result is added to the value of  $b$ .

**Sample b On a**

This is a sample and hold function. The values of **inputs a** and **b** are interpreted as logical quantities, as described for the equations a AND b, a OR b. When the value of **input a** changes from FALSE to TRUE (goes above +.5), the value of **input b** at that moment is sampled (recorded), and becomes the FUN's output value. This value remains constant until **input a** makes another transition from FALSE to TRUE.

**Sample b On ~a**

This works like the previous equation, but the value of **input b** is sampled whenever the value of **input a** makes a transition from TRUE to FALSE.

**Track b While a**

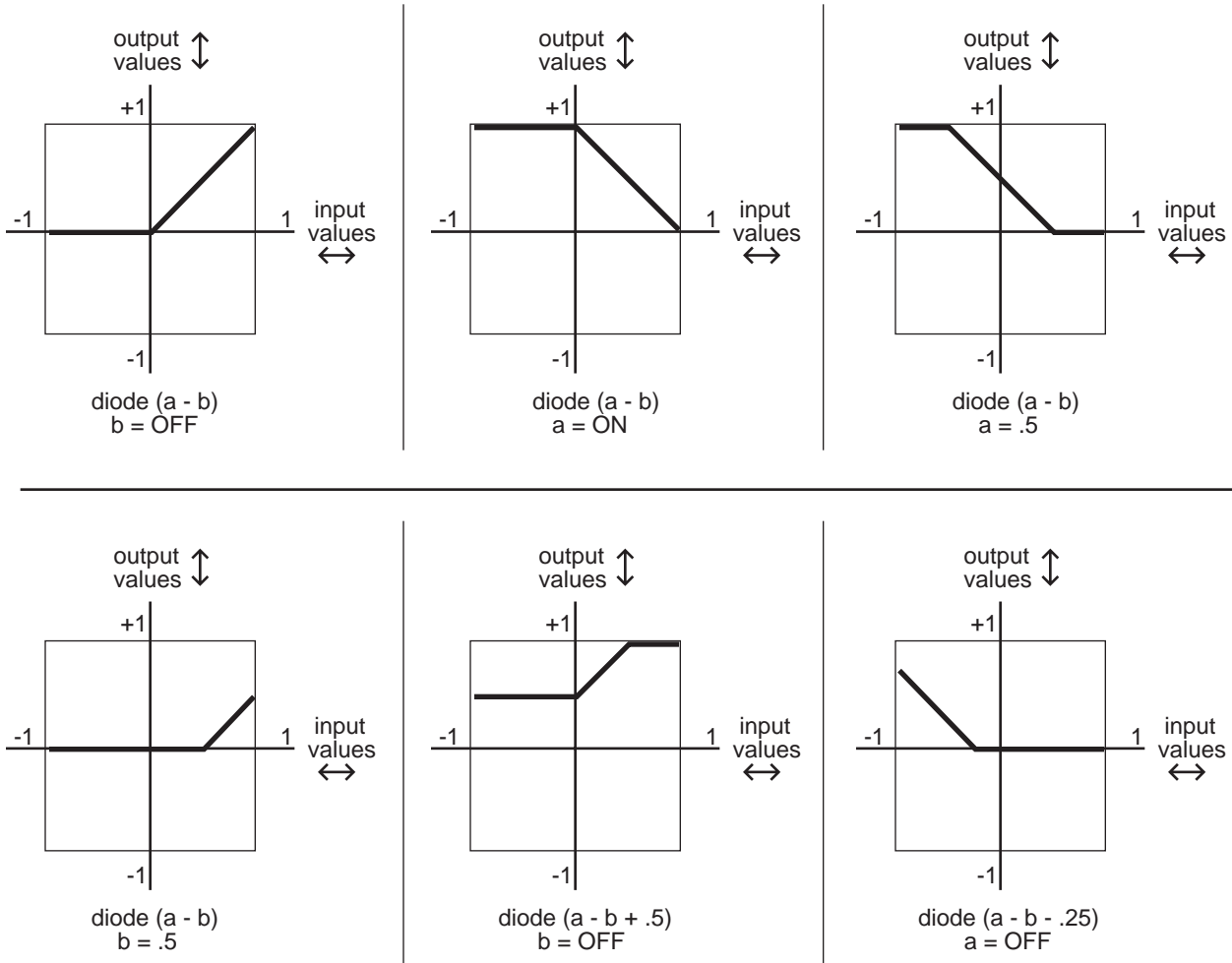
This equation also interprets the values of **inputs a** and **b** as logical quantities. While the value of **input a** is TRUE, the value of **input b** is used as the FUN's output value. The output value changes exactly as the value of **input b** changes. When the value of **input a** goes FALSE, the FUN's output value freezes and remains constant until the value of **input a** becomes TRUE again. The FUN's output value then continues to track the value of **input b**.

**Track b While ~a**

This is the opposite of the previous equation. The FUN's output value tracks the value of **input b** as long as the value of **input a** is FALSE.

## Diode Equations

The remaining equations perform a diode-like function; only positive input values are significant. If the result of the equation is negative, the FUN's output value is 0. You can use these equations to limit bipolar control signals to unipolar values. Normally you'll use these by setting input a or b to ON or OFF, and assigning some control source to the other input. These will enable you to produce a variety of output curves.



Diode (a - b) simply subtracts the value of **input b** from the value of **input a**. If the difference is less than 0, the output value is 0.

Diode (a - b + .5) adds a constant of +.5 to the difference of (a - b), then maps all negative values to 0. The curve is the same shape as Diode (a - b), but shifted upward 1/4 of the range between -1 and +1.

Diode (a - b - .5) is the same curve as diode (a - b), shifted downward 1/4 of the range.

Diode (a - b + .25) shifts the curve up 1/8 of the range, and diode (a - b - .25) shifts the curve down 1/8 of the range.

## The Order of Evaluation for FUNs

The K2vx is a computer, and processes information at very high speeds. Every 20 milliseconds, it checks the condition of every active parameter, evaluates any changes, and processes the new information. This is done according to a rigid set of priorities that determines the sequence in which the parameters are evaluated.

The status of each control source is evaluated in turn, according to the sequence in which they appear in the Control Source list. In the case of the FUNs, they are evaluated in the following order: FUN1, FUN2, FUN3, FUN4 (although there are a few control sources that get evaluated between FUN2 and FUN3).

This sequence of evaluation becomes significant if you assign a FUN as the input for another FUN. You should always assign FUNs as inputs for higher-numbered FUNs. For example, you might assign FUN1 as input a for FUN2. Since the K2vx needs to know the value of FUN1's output before it can evaluate FUN2, assigning the FUNs in this way will ensure that the K2vx can evaluate both FUNs as quickly as possible.

If you were to assign FUN2 as an input for FUN1, then when the time came for the K2vx to evaluate FUN1, it wouldn't know the current value for FUN2's output, so it would evaluate FUN1 according to the previous value of FUN2. There would be a delay of one evaluation cycle before a change in FUN2 would be reflected in FUN1. This might have an adverse affect on the start of notes as you play. You won't harm anything, but you might not hear what you expect to hear.

## Chapter 17

# Other Editors

In addition to the editors accessible from their respective modes, there are three editors that enable you to modify other performance parameters of the K2vx. They are the Intonation Table Editor, the Velocity Map Editor, and the Pressure Map Editor.

### The Intonation Table Editor

Intonation tables define the interval between the notes in each octave. The default intonation table is "1 Equal," which sets precisely equal intervals between notes—the standard for modern western music. If you're interested in playing other styles of music, you can use any of the K2vx's other factory intonation tables, or create you own with the Intonation Table Editor.

The intonation table is selected in Master mode with the Intonation parameter. Here's the list of available factory intonation tables (check the *Reference Guide* for brief descriptions):

- |    |               |
|----|---------------|
| 1  | Equal         |
| 2  | Classic Just  |
| 3  | Just b/7th    |
| 4  | Harmonic      |
| 5  | Just Harmonic |
| 6  | Werkmeister   |
| 7  | 1/5th Comma   |
| 8  | 1/4th Comma   |
| 9  | Indian Raga   |
| 10 | Arabic        |
| 11 | 1Bali/Java    |
| 12 | 2Bali/Java    |
| 13 | 3Bali/Java    |
| 14 | Tibetan       |
| 15 | CarlosAlpha   |
| 16 | Pyth/ aug4    |
| 17 | Pyth/ dim5    |

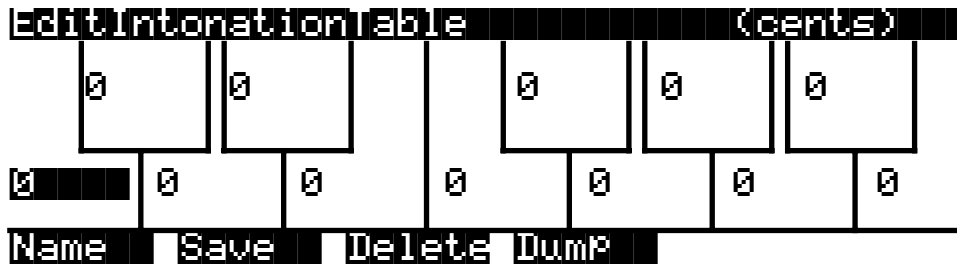
As you scroll through the options for the Intonation parameter, you'll notice an eighteenth entry with a name like "Obj *vn.n*". This isn't really an intonation table; it's simply a convenient place for the K2vx to keep a record of the current version of ROM objects. If you ever need to find out the version of ROM objects loaded into your unit, this is the place to check. Just remember, if you're using a non-standard intonation table, that you'll need to reset the Intonation parameter after you check the ROM object version list at "table" 18.

It's important to emphasize that the detuning values for each parameter affect the intervals in each octave independently. That is, notes that are an octave apart will remain perfectly tuned regardless of the detuning between notes within the octave. Consequently, editing intonation tables will not enable you to create tunings with more (or fewer) than 12 notes per octave. If you want to do this, use the Keymap Editor, assign each key to its own key range, then tune the samples in each key range.

When you're ready to edit an intonation table, select Master mode, then select the Intonation parameter. Use any data entry method to select the intonation table you want to edit. Check the

value of the intonation key parameter (IntonaKey). This sets the tonic, or reference note for the parameters on the Intonation Table Editor page. We'll explain this below.

Enter the Intonation Table Editor by pressing the Edit button. The Intonation Table Editor page displays a graphic representation of a C octave, with the low C always representing the tonic. The values for these parameters indicate the amount of detuning applied to each note relative to perfectly equal intonation. If you look at the values for the intonation table 1 Equal (as illustrated below), you'll see that the value of each parameter is 0. None of the notes is detuned, so the intervals between each note are exactly the same. Take a minute to look at the values for some of the other intonation tables, and listen to the effect on the tuning.



The top line of the display gives you the usual reminder of your location. The bottom line labels the soft buttons for the Intonation Table Editor, which let you perform the usual librarian functions of naming, saving, and deleting the currently selected intonation table, or dumping it via SysEx.

Assume for now that the IntonaKey parameter is set to a value of C (which is the default). This means that each C is the reference point for defining the intervals of all the other notes.

Let's say you want to create an intonation table that flats the 5th in each octave. Select the value on the G note displayed by the Intonation Table Editor, and set its value to a negative number—for example, -12 cents. This will reduce the interval between the tonic (C, in this case) and the 5th (G, in this case) by 12/100ths of a semitone. All the Gs will remain exactly an octave apart from each other, but they'll be 12 cents flat from their normal pitches.

Press the Compare (Disk mode) button to hear the difference between the edited table and the original. Press the **Save** soft button to begin the save dialog, where you can rename and save your edited intonation table.

The changes we've discussed here are based on a value of C for the IntonaKey parameter. If you were to select Master mode and change the value of the IntonaKey parameter, you'd change the tonic note that the intonation table uses to define the intervals of the other notes. Let's say you change IntonaKey from C to G. The tonic changes to G, and the intervals in each octave are defined relative to G. The minor 2nd is now G#, the major 2nd is A, and so on. In the example we gave above, all the Ds would now be 12 cents flat.

A good reference source for descriptions of various alternative intonations is a book by Scott R. Wilkinson entitled *Tuning In: Microtonality in Electronic Music*, available from the Hal Leonard Publishing Corporation.



## The Velocity Map Editor

Velocity maps affect the way the K2vx interprets attack velocity values. Velocity maps are assigned as the values for two different parameters: `VelocMap` on the `XMIT` page in MIDI mode, and `VelocityMap` on the `RECV` page in MIDI mode. (They also apply to the `VelTouch` parameter in Master mode, but since the `VelTouch` parameter doesn't affect the K2vx, you shouldn't change its default value.) There's only one list of velocity maps, and all three of these parameters use this list. Even so, certain maps are intended for use with the `VelTouch` parameter in the keyboard version of the K2vx, and others for use with the MIDI mode parameters. You can think of these as internal maps and MIDI maps. With the K2vx, you should use only the MIDI maps, and unless you have a problem with the K2vx's velocity response, or that of MIDI devices connected to the K2vx's MIDI Out port, you should leave both the Receive Velocity map and the Transmit Velocity map set to their default values.

Every attack velocity value generated by your MIDI controller is mapped through the velocity map assigned on the MIDI mode `RECV` page before being passed to the sound engine. If you're using the Local Keyboard Channel feature, attack velocity values received at the K2vx's MIDI in port go first through the Receive Velocity map, then to the sound engine, then through the Transmit Velocity map (as selected with the `VelocMap` parameter on the `XMIT` page in MIDI mode), and to the MIDI Out port.

The `VelocMap` parameter on the `XMIT` page in MIDI mode affects only the velocity values the K2vx sends to its MIDI Out port. Normally you'll leave its value set to 1 Linear, especially when you're recording sequences with a personal computer or hardware sequencer. You might want to adjust this parameter, however, when you need to change the response of MIDI slaves that you're driving from the K2vx. If, for example, you're overdriving a DX7 that you have slaved to the K2vx, try the map 6 Hard2.

The `VelocityMap` parameter on the `RECV` page in MIDI mode affects how the K2vx responds to attack velocity values received at its MIDI In port. Normally you'll leave this set to Linear, as with the `VelocMap` parameter on the `XMIT` page. You might want to adjust this parameter, for example, to boost or cut the volume response of the K2vx during playback in Song mode, or when playing a sequence from an external sequencer.

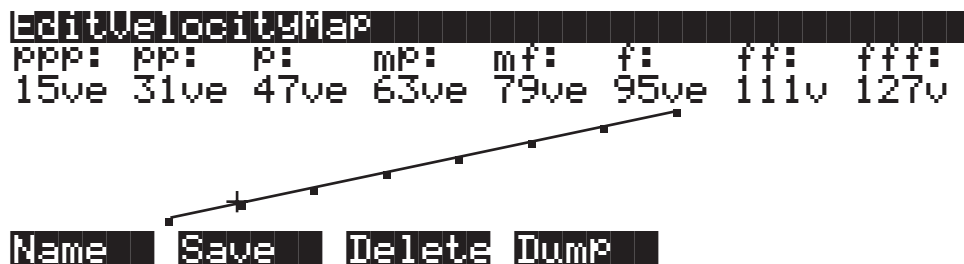
Here's a list of the velocity maps stored in ROM (the factory presets). Keep in mind, however, that you should use only the linear MIDI maps; the internal maps are meant for the K2vx.

- 1 Linear (the default on the `XMIT` and `RECV` pages in MIDI mode): Velocity values are passed through this MIDI map unchanged.
- 2 Light1
- 3 Light2.
- 4 Light3.
- 5 Hard1.
- 6 Hard2.
- 7 Hard3.

The attack velocity value of every note received by the K2vx's sound engine is routed through the map assigned to the `VelocityMap` parameter on the `RECV` page in MIDI mode). The velocity maps convert each attack velocity value into a new value, depending on the curve of the map. By editing velocity maps, you can change the attack velocity value that the maps calculate.

## Using the Velocity Map Editor

To get to the Velocity Map Editor, select any of the parameters we've mentioned above—`VelTouch` (Master mode) `VelocMap` (`XMIT` page, MIDI mode), or `VelocityMap` (`RECV` page, MIDI mode). Use any data entry method to select the map you want to edit, then press `Edit`.



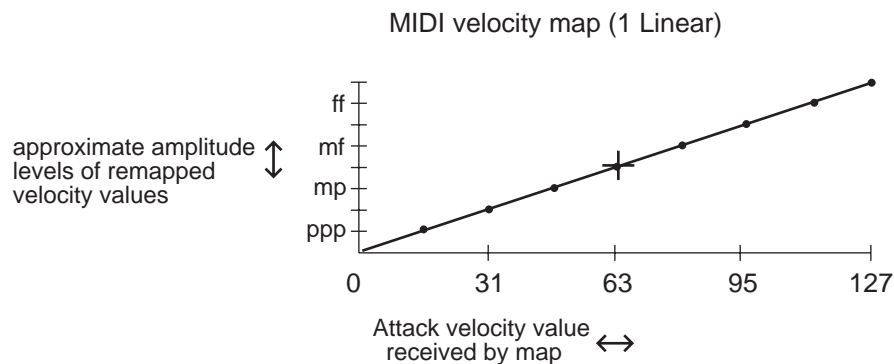
The top line of the display gives you the usual reminder of your location. The bottom line labels the soft buttons, which execute the normal librarian functions of naming, saving, or deleting velocity maps, or dumping them via SysEx. The values you see in this diagram are the settings for the velocity map 1 Linear.

The eight parameters on the Velocity Map Editor page are expressed in terms of the eight dynamic levels of standard musical notation. As far as the velocity maps are concerned, the dynamic levels represented by the parameter names are simply reference points—for example, when using the map 1 Linear, an attack velocity value of 63 will result in a remapped velocity value at the mp level of the K2vx’s dynamic range.

The values for these parameters are expressed in “vels,” which correspond to the standard MIDI attack velocity values of 0—127. The values you set for each parameter determine the velocity value that must be generated to achieve the dynamic level for that parameter. For example, if you’re using the map 1 Linear as the value for the VelocityMap parameter, then the K2vx will play notes at full volume only when the attack velocity values for those notes are 127. Notes with attack velocities from 0 to 15 will be played at the lowest volumes. They wouldn’t all play at the *same* volume; their volumes would be graduated, but would be at the low end of the scale.

The values for these parameters have another function as well. They determine the velocity thresholds for the velocity triggers (VTRIG), as well as the velocity crossovers for multi-velocity keymaps. Once again using the example of the map 1 Linear, if you have the VelCrossover parameter in the Keymap Editor set to a value of mp, then attack velocity values higher than 63 will trigger the high-velocity range of a dual-velocity keymap, while velocity values of 63 or lower will trigger the low-velocity range.

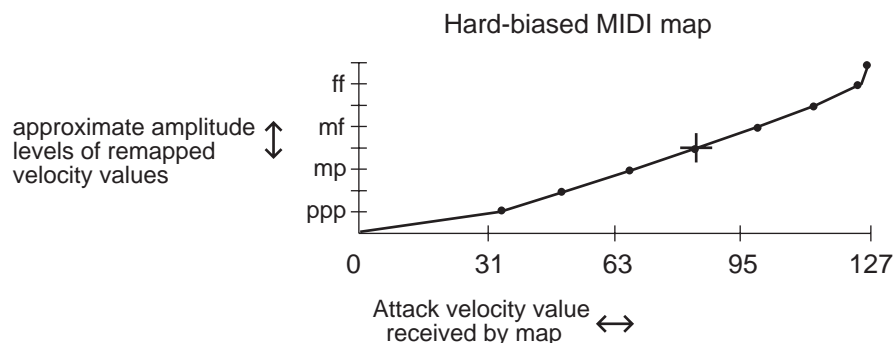
The curve that represents the velocity map is shown in the center of the page. Think of this curve as a plot on a graph, as shown in the diagram below. The horizontal axis gives the attack velocity values received by the velocity map. The vertical axis gives the approximate dynamic level at which the K2vx will play the note (in response to the remapped velocity value).



When you select a parameter, the small crosshairs move to indicate the position on the curve of the dynamic level represented by that parameter. Each of the eight levels is marked on the curve by a small dot. The crosshairs jump to one of the dots when you select the corresponding parameter. The dots always have the same vertical location, but move to the left or right depending on the velocity value assigned to them. The farther to the right that one of these dots is located, the higher the attack velocity value required to play a note at that level. As the curve suggests, velocity values between those set for the eight dynamic levels will be remapped to a new velocity represented by the height of the curve.

As you change the value of the currently selected parameter, the shape of the curve changes to reflect the new value. As an example, let's assume that you're driving a MIDI slave synth from the K2vx's MIDI Out port, and you get full-amplitude sounds from the slave, even when you play softly on your MIDI controller. To fix this, you'd edit the map to give it a harder bias (or use one of the preset MIDI maps with a hard bias). That is, you want to have to strike your controller's keys harder to get full amplitudes.

The diagram below shows a hard-biased linear map similar to the Hard3 map. Although the even progression from low to high remapped values remains, the entire curve is shifted so that a given input velocity value will remap to a lower value.



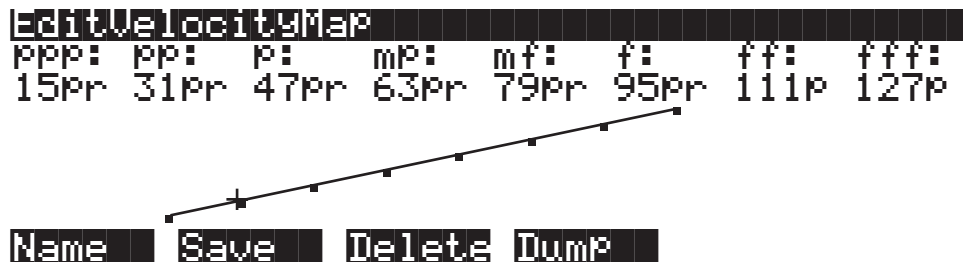
## The Pressure Map Editor

Pressure maps function in much the same way as velocity maps, but they affect the K2vx's response to mono pressure messages received from your MIDI controller. Like the velocity maps, pressure maps convert pressure values of 0—127 to new values that are sent to the sound engine. Also like the velocity maps, pressure messages received at the K2vx's MIDI In port from your MIDI controller pass through the Receive Pressure map, then to the sound engine. If you're using the Local Keyboard Channel feature, the pressure messages pass first through the Receive Pressure map, then through the Transmit Pressure map, then to the MIDI Out port.

Pressure maps are assigned as the values for the PressTouch parameter in Master mode, the PressMap parameter on the XMIT page in MIDI mode, and the PressureMap parameter on the RECV page in MIDI mode. There's one list of pressure maps, which is used to select the pressure map for all three parameters. As with the velocity maps, you should use only the MIDI map (ID 1) for the MIDI mode parameters, leave the value of the PressTouch parameter at its default value. Here's the list of all seven maps. Keep in mind, however, that you should use only the linear MIDI maps; the internal maps are meant for the K2vx.

- 1 Linear: The default internal and MIDI pressure map.
- 2 Easy.
- 3 Easier.
- 4 Easiest.
- 5 Hard.
- 6 Harder.
- 7 Hardest.

To edit a pressure map, select one of the three pressure map parameters, and use any data entry method to set its value to the pressure map you want to edit. Edit the map 1 Linear if you want to use your edited map with the MIDI pressure map parameters. Press EDIT, and the Pressure Map Editor page will appear.



The top line of the display gives you the usual reminder of your location. The bottom line labels the soft buttons, which execute the normal librarian functions of naming, saving, or deleting the current pressure map, or dumping it via SysEx.

The eight parameters are expressed in terms of the dynamic levels of standard musical notation. They don't correspond to actual volume levels, they're just a reference point (ppp corresponds to minimum pressure effect, and fff corresponds to maximum). Their values are expressed as "prs" units, and range from 0—127, corresponding to the values of mono or poly pressure messages. The values you set for these parameters determine the pressure message values that must be generated to achieve the dynamic levels expressed by the parameters. Values in between those indicated by the parameters will be remapped as indicated by the curve.

# Chapter 18

## Audio Outputs

### Audio Configurations

The K2vx features ten audio jacks, enabling you to route the K2vx's sounds through its own internal effects processor, through an external effects device, or through combinations of the two. Of course the K2vx's effects processor can be bypassed as well.

On the rear panel are a pair of stereo mix jacks (MIX L and R), and eight additional jacks—A (L and R), B (L and R), C (L and R), and D (L and R). These can be configured as four stereo pairs or as eight separate outputs. Your audio configuration is determined by two factors. The "hardware" side has to do with which jacks you use and what kind of cables you connect to them. The "software" side depends on the settings you make for parameters on the OUTPUT page in the Program Editor, and on the Master mode page.

The simplest configuration would involve connecting the left and right MIX outputs directly to a mixing board or keyboard amplifier. If you're using only one input to your sound system from the K2vx, use the left MIX output for a mono signal. We'll assume for now that you're using only the MIX outputs. The use of the separate outputs is discussed later in this chapter.

Note: we recommend that you make the cable connection to the K2vx (or any instrument) after you've made your other audio connections, since this reduces the chance of creating static electricity that can cause an audible "pop" (and, in extreme cases, cause equipment damage).

### Using the MIX Outputs

If you plan to use the K2vx's global effects processor to add dimension to your sounds, you'll need to use the MIX outputs. The audio output to the separate outputs is always dry—that is, the audio signal is sent to the separate outputs without passing through the global effects processor. The MIX outputs and the headphone jack are the only jacks where the signal will appear with effects.

The signal is routed using the OUTPUT page in the Program Editor. While in Program mode, press the EDIT button, then press one of the <more> soft buttons until you see "OUTPUT" on the display's bottom line. Press the corresponding soft button, and the OUTPUT page will appear. Set the value of the Pair parameter to A(FX) if you want the current layer's sound to be routed through the effects processor before appearing at the MIX outputs.

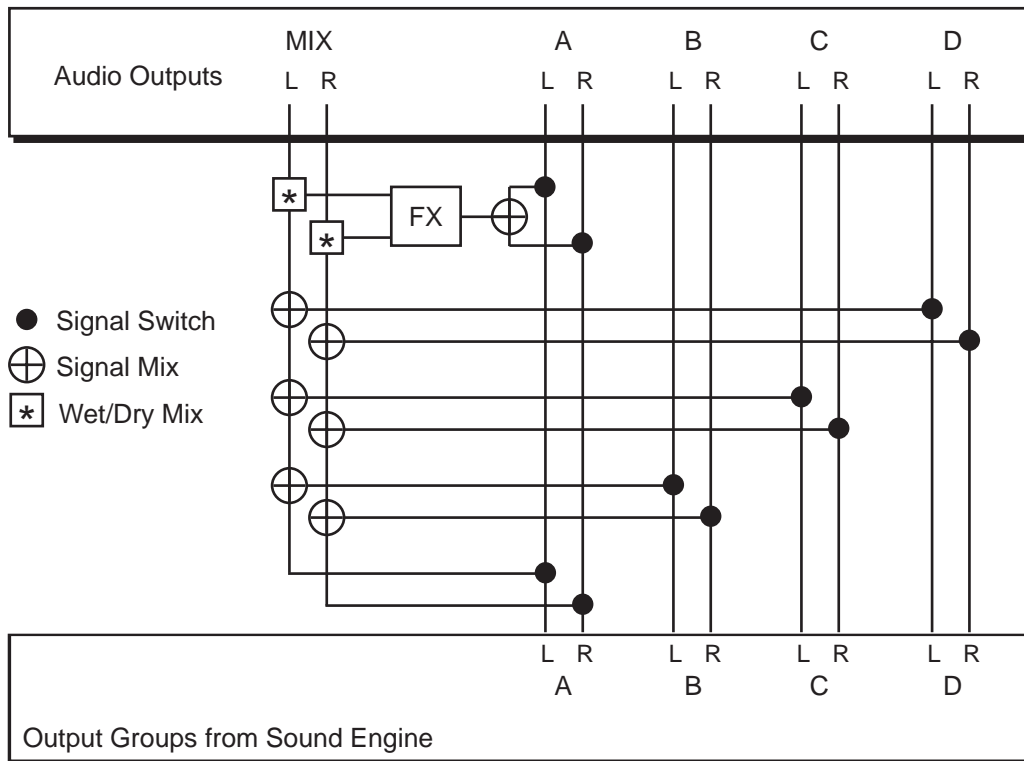
A value of A(FX) means two things: first, the layer's sound will appear with effects at the MIX outputs, and second, that the layer's sound will appear *without* effects at the Group A outputs if standard audio cables are inserted into the Group A output jacks. Set the value of the Pair parameter to B(DRY), C(DRY), or D(DRY) if you want the layer's sound to appear without effects at the MIX outputs. It will also appear at the Group B, C, or D outputs if standard audio cables are inserted into the respective output jacks. Connecting audio cables to the separate outputs will remove part or all of the signal from the MIX outputs. Be sure to set the output pair for every layer in the program; each layer can be routed independently (use the CHAN/Bank buttons to scroll through the layers). You'll also want to set the pan position for each layer while you're on the OUTPUT page. Depending on the program you're working with, you'll have several different options for panning. See "The OUTPUT Page" in Chapter 6 for an explanation of these options. Also see "PANNER" in Chapter 14 to learn how to gain even more control over a layer's pan position.

## Using the Separate Outputs

The K2vx's eight separate outputs are arranged in four pairs, A, B, C, and D. These are mono output jacks, and you'll normally plug a standard audio cable into one or more of them. Depending on the OUTPUT page settings you use for a given program's layers, they can function as four stereo output pairs, or as eight separate outputs.

The diagram below illustrates the wiring of the K2vx's audio system. From the sound engine, the audio signal can be routed to any of the four output groups. If the signal is routed to Output Group B, C, or D it will appear at the MIX outputs if no audio cables are plugged into those outputs. Plugging cables into any of the separate outputs will divert the signal from the MIX outputs to that output.

If the signal is routed to Output Group A, and audio cables are plugged into the A outputs, the signal will pass directly to the A outputs. If no cables are plugged into the A outputs, the signal will pass from the sound engine to the global effects processor. The left and right sides are mixed before passing to the effects processor. The effects processor splits the signal into left and right sides again after the effects are applied. Effects will be applied to both sides of the signal unless the Out A->FX parameter is set to L Only, in which case only the left side will have effects applied. In either case, the signal will pass to the MIX outputs. Remember that the effects processor isn't stereo; effects can't be panned the way dry sounds can.



If a layer's pan position is centered, then the outputs to which it's assigned will carry an equal mix of the layer's sound in its left and right outputs. This makes the audio outputs a stereo pair with respect to that layer. If the layer is panned all the way to the left or right, the separate outputs function as individual monophonic outputs with respect to that layer. Of course, you could have a multi-layer program with all its layers assigned to the same output group, and pan each layer so that some of them appear at both outputs, while others appear only at the left or right output. If you're using your K2vx for recording and multi-timbral sequencing, you'll want to use the separate outputs to increase your mixdown options, and for adding outboard effects to your sounds. The separate outputs give you a great deal of added flexibility.

## Hardware Audio Switching

It's important that you understand the effect on the MIX outputs when you use the separate outputs. When you connect a standard audio cable to any of the separate outputs, *every* sound (or part of a sound) that's assigned to the output you connect will be removed from the MIX outputs. For example, imagine that you plug a cable into the left output of Group A. The left side of every program layer assigned to Output Group A will disappear from the left side of the MIX outputs, and will appear only at the left Group A output. If a layer is panned all the way left, it will disappear from the MIX outputs entirely. If it's centered, the left side will disappear, but the right side will remain at the right MIX output. A layer that's panned hard right will not be affected in this case. If you use all eight separate outputs, the MIX outputs will not carry any signal.

The pan settings you make on each program's OUTPUT page apply to the separate outputs just as they do for the MIX outputs. Keep in mind, however, that the signal at the separate outputs will always be dry; only the MIX outputs make use of the global effects processor.

Once you have the output groups assigned on the OUTPUT page, you can go to Master mode to specify how the signal will appear at the MIX outputs. On the Master mode page are three parameters that affect the routing of all signals that are sent to the MIX outputs. The settings for these parameters will affect each layer in every program.

The parameters Out A->Mix and Out B->Mix determine the panning of the A and B Group's audio signal at the mixed outs. If they're set to Stereo, then whatever panning is applied to each program (as set on the OUTPUT page in the Program Editor) will show up at the MIX outputs. Output groups C and D are permanently set to Stereo.

If either of these parameters is set to Mono, then programs routed to the corresponding audio output group will send all of their sound at equal levels to both outputs, regardless of any panning that might be applied to them on the OUTPUT page. This is a quick and convenient way to convert the MIX outputs into two mono outputs.

The parameter Out A->FX is intended for special cases when you want an additional separate output that doesn't go through the K2vx's effects processor. A value of L Only bypasses the effects processor for all sounds assigned to the right side of Output Group A. There's an example in Chapter 11 that describes how to use this parameter.

## Using Stereo Insert Cables

The preceding description of the separate outputs is only part of the picture. You can make the separate outputs even more useful by using stereo insert cables. A stereo insert cable has a single stereo plug on one end; the other end is split into two mono cables fitted with mono plugs. By plugging the stereo end of a stereo insert cable into one or more of the K2vx's separate outputs, you can create all sorts of audio configurations with outboard gear. The tip of the stereo end carries the signal to the send side of the split end, and the ring of the stereo end receives the signal from the return side.

### Effects Return

Plug the stereo end of a stereo insert cable into one of the K2vx's separate outputs, then plug the send (tip) side of the split end into an input on an outboard effects box. If you connect the effects box's output to the return (ring) side of the split end, you've created a loop that will send the K2vx's output into the effects box, and return it wet to the K2vx. Since the loop returns the signal to the same output jack on the K2vx, the K2vx's MIX outputs act as if there were no cable at all plugged into the separate output, and the signal will appear at one or both of the MIX outputs. If you're using one of the Group A outputs, you can apply the effect of both the outboard box and the K2vx's global effects processor to the corresponding side of the K2vx's MIX outputs. If you don't want to double the effects in this way, you can go to Effects mode, set the FX mode parameter to Master, and set the Wet/Dry Mix parameter to 0%. You can also set the Pan parameter of programs routed to Output Group A fully right, then go to Master mode

---

and set the Out A->Mix parameter to mono, and the Out A->FX parameter to L Only. This will send programs routed to Output Group A to the MIX outputs without going through the K2vx's effects processor, but will still send the signal to both the left and right MIX outputs.

#### INPUT TO K2vx

Suppose you want to bring the outputs from a microphone and another synth into the K2vx's MIX outputs so you can apply K2vx effects to them. You have two options for setting this up. (You may need to run the mic output through a pre-amp before it gets to the K2vx.)

The most versatile way to do this requires an external mixing board. Use standard audio cables to connect the outputs from the other synth and the mic to two of the input channels on the mixing board (for this example, we'll use Channel 1 for the other synth—we'll assume it's a mono unit—and Channel 2 for the mic). Then plug the stereo end of a stereo insert cable into the A Left output of the K2vx. Connect the send (tip) side of this cable to Input channel 3 on your mixing board. Connect the return (ring) side of this cable to the left side of a subgroup output on the mixing board. Similarly, connect the stereo end of a second stereo insert cable to the A Right output of the K2vx. Connect the send side of this cable to Input channel 4 on the mixing board, and the return side to the right side of the same output subgroup on the mixing board. Pan Channel 3 fully left, and Channel 4 fully right. Pan Channels 1 and 2 as desired.

Here's what happens. Every K2vx sound routed to Output group A will be initially removed from the K2vx's MIX outputs, and will pass instead to Channels 3 and 4 on the mixing board. These signals, along with the signals from Channels 1 and 2 on the mixing board (the outputs from the other synth and the mic), will be routed to the mixing board's output subgroup, and will pass to the A outputs of the K2vx, which now act as inputs. Since there's a closed circuit between the K2vx and the mixing board, the K2vx handles the input at its A outputs as if it were its own output, and passes it along to the effects processor, then to the MIX outputs. The signals from the other synth and the mic, as well as the K2vx's Output group A sounds, will have effects applied to them before they appear at the MIX outputs. (The dry portions of all these sounds will be panned according to your pan settings for Channels 1–4 on the mixing board and the pan settings for the K2vx sounds routed to Output Group A, while the wet portions of all the sounds will appear equally at the MIX outputs.)

A second, simpler method is to pan all the K2vx sounds routed to Output Group A to the left, then connect the output from the other synth or the mic into the K2vx's A Right output using a standard audio cable. The K2vx sounds routed to Output Group A will appear at the MIX Left output, and the signal from the other synth or the mic will appear at the MIX Right output. This restricts your panning options somewhat, but is easier to set up.

## Output Groups and MIDI Channels

You can also set a program's output group on the basis of the MIDI channel it's assigned to. This is helpful for mixing purposes, in both studio and live situations. When you're using the MIX outputs, it's an easy way to apply or remove effects without editing the program.

To set the output group on the basis of MIDI channel assignments, select MIDI mode by pressing the MIDI mode button. Then press the CHANLS soft button. Use the CHAN/BANK buttons to select the channel you want to edit. The OutPair parameter controls the output group assignment for all programs assigned to the channel. If you choose a value of "Prog" (the default), then any program assigned to that MIDI channel will be routed according to its own output settings on the OUTPUT page in the Program Editor. Otherwise, programs on that channel will be routed according to the settings for the Output Group parameter on the CHANLS page.

Say for example that you set the OutPair parameter for Channel 1 to a value of A(FX). Now when you select MIDI Channel 1 as the current MIDI channel in Program mode, *any* program you select will appear at the A outputs, or at the MIX outputs (with effects) if no cables are connected to the A outputs. Set the OutPair parameter to B(DRY), and any program you assign to Channel 1 will appear at the B outputs, or dry at the MIX outputs if no cables are connected to the B outputs.



## Chapter 19

# Programming Examples

The other chapters in this manual have described the K2vx's features in detail. This tutorial chapter will take you step-by-step through several programming operations.

Each of the following examples will begin from the same starting point: the default program with ID 199. This program is included specifically for the purpose of giving you a programming template. Most of its parameters have been set at values that don't affect the sound of the program.

You may want to adjust some of the parameters of Program 199, to create your own customized programming template. Even if you don't, it's a good idea to begin with Program 199 when you're building a new sound, so you'll know exactly what you have from the start.

### Example 1

#### Trumpet with Delayed Vibrato and Velocity-triggered Stabs

Vibrato is a regular oscillation in pitch that adds dimension to any sound. Brass players will often "stab" a note, punching it then letting the pitch roll down smoothly or in small fast steps.

To create these effects, we'll use an LFO to control the pitch, (this is the typical way to create vibrato), and delay it with an ASR. This way you'll hear the vibrato only on notes that you hold for a second or so. The stab will be done with a second ASR controlling pitch and amplitude. The stab's ASR will be triggered by a velocity trigger (VTRIG), so only those notes you play at fortissimo will stab.

Start by selecting Program 199 and pressing EDIT. The ALG page will appear. The first task is to change the keymap. Press the KEYMAP soft button to select the KEYMAP page.

```

EditProg:KEYMAP          <>Layer:1/1
KeyMap:17 Trumpet      Stereo:Off
XPose :0ST             TimbreShift :0ST
KeyTrk:100ct/key      AltSwitch  :OFF
VelTrk:0ct            PlaybackMode:Normal
<more  ALG  LAYER  KEYMAP  PITCH  more>

```

The KeyMap parameter is already selected, and as you can see, the Default program uses the Grand Piano Keymap. Use any data entry method to change the Keymap to Trumpet, which has ID 17. The KEYMAP page should look like the diagram above when you're done.

Remember that you can play your MIDI controller's keyboard at any time while editing, so you can listen to each change as you make it.

Next set up the vibrato. Start by selecting the PITCH page (press the PITCH soft button). Use the cursor keys to move the cursor to the Src2 parameter. Use any data entry method to select LFO1 as its value (pressing 1, 1, 4, ENTER on the alphanumeric pad is the quickest). This assigns LFO1 to control the pitch of the trumpet sample.

## Programming Examples

### Example 1

The next step is to set the depth of the vibrato. Select the MaxDpt parameter and assign a value of 10 cents (1, 0, ENTER). Since the default program is preset to have your controller's Mod Wheel control the depth of Src2, you can hear the vibrato by pushing the Mod Wheel fully up (LFO1 has non-zero default values in the default program, otherwise, you wouldn't hear the vibrato). If you're not sure you hear the vibrato, try setting the MaxDpt parameter to a larger value.

Next, select the DptCtl parameter and assign a value of ASR2 (1, 1, 1, ENTER). This will cause ASR2 to control the depth of the vibrato. At this point, the default values for ASR2 will cause the vibrato to fade in and out.

There are two more steps to programming the delayed vibrato: adjusting the rate of LFO1 and setting up ASR2 to control the vibrato's delay. First, select the LFO page by pressing the <more soft button (the leftmost) three times. You'll see the LFO soft button. Press it, and the LFO page will appear. The default value for LFO1's minimum rate (the MnRate parameter) is 2 seconds. Select this parameter with the cursor buttons, and set its value to .16 seconds (1, 6, ENTER). Select the MxRate parameter, and set its value to 4.40 Hz (4, 4, 0, ENTER). Select the RateCt parameter, and assign a value of ASR2 (1, 1, 1, ENTER). The vibrato will still fade in and out because of the default settings of ASR2.

The LFO page should now look like this:

```
EDITPROG:LFO <>Layer:1/1
MnRate:MxRate:RateCt:Shape: Phase:
LFO1: 0.16H 4.40H ASR2 Sine Odeg
LFO2: 2.00H 0.00H OFF Sine Odeg
<more LFO ASR FUN UTRIG more>
```

Now select the ASR page to adjust the settings for ASR2. Since the cursor is highlighting ASR2 as the value for the RateCt parameter, you can select the ASR page by pressing EDIT.

To program a realistic delayed vibrato, you need to adjust the Mode, Delay, and Attack parameters. Select the Mode parameter and change its value to Hold (use the Alpha Wheel or Plus/Minus buttons). This will prevent the vibrato from fading as it did. (This fading was caused by the ASR repeating, which was the default setting). Now select the Delay parameter and set its value to .4 seconds (4, 0, ENTER). Select the Attack parameter and change its value to .48 seconds (4, 8, ENTER). The vibrato should now begin to fade in gradually after a short delay, then remain constant at a rate of 4.40 cycles per second. The ASR page should look like the page below. We're finished with the delayed vibrato; next is the velocity stab.

```
EDITPROG:ASR <>Layer:1/1
Trig: Mode: Delay: Attack:Releas:
ASR1: ON Rept 1.00s 1.00s 1.00s
ASR2: ON Hold 0.40s 0.48s 1.00s
<more LFO ASR FUN UTRIG more>
```

We want the stab to drop the trumpet's pitch *and* amplitude, but only when notes are played fortissimo or louder. This is done by using ASR1 as a control source on both the PITCH and AMP pages, then using a velocity trigger (VTRIG) to control ASR1.

First return to the PITCH page (if you're still on the ASR page, press the more> soft button three times, and the PITCH soft button will appear). Press PITCH, then select the Src1 parameter, and set its value to ASR1 (1, 1, 0, ENTER). Then select the Depth parameter, and set its value to -1200 cents (+/-, 1, 2, 0, 0, ENTER). ASR1 has non-zero default values in Program 199, so you'll hear the pitch drop an octave if you play a note. The PITCH page should look like this:

```

Edit Prog:PITCH <>Layer:1/1
Coarse:051 Src1 :ASR1
Fine :0ct Depth :-1200ct
FineHz: 0.00Hz Src2 :LF01
KeyTrk:0ct/key DptCt1:ASR2
VelTrk:0ct MinDpt:0ct
MaxDpt:-10ct
<more> ALG LAYER KEYMAP PITCH >more>

```

The next step is to adjust the characteristics of the stab by programming ASR1. Select the Src1 parameter again, and press EDIT to select the ASR page.

First, select the Trig parameter, and assign a value of VTRIG1 (1, 0, 6, ENTER). Select the Mode parameter and set its value to Hold. Select the Delay parameter and set its value to .34 seconds (3, 4, ENTER). Set the Attack parameter to a value of .78 seconds. The ASR page should now look like this:

```

EditProg:ASR <>Layer:1/1
ASR1: Trig: Mode: Delay: Attack:Releas:
      VTRIG1 Hold 0.34s 0.78s 1.00s
ASR2: ON Hold 0.40s 0.48s 1.00s
<more> LFO ASR FUN UTRIG >more>

```

To make the stab sound realistic, we'll drop the amplitude at the same rate as the pitch. To do this, select the F4 AMP page (press the more> soft button once, then press the F4 AMP soft button). Select the Adjust parameter, and assign a value of 8 dB. This will give the trumpet a bit more punch. Next, select the Src1 parameter, and set its value to ASR1. Then select the Depth parameter and set its value to -68 dB.

Here's an important point to remember. ASR1 is being used to control both the drop in pitch for the stab, and the drop in amplitude as well. All of the control sources can be similarly assigned for as many parameters as you like.

The final step in this example is to set the velocity threshold of the stab. Right now the stab is occurring on almost every note, but we want it to happen only when playing fortissimo or louder. To do this, press the more> soft button three times, then press the VTRIG soft button to select the VTRIG page. The VTrig1 Level parameter is already selected, so just turn the Alpha Wheel until the VTrig1 Level value is ff. Now you can play softly without triggering the stab.

That's it for Example 1. If you want to save your work, the easiest way is to press EXIT. The K2vx will ask you if you want to save. You should probably press the Rename soft button, give your program a new name, then save it with a new ID. See Chapter 5 if you need help with the Save dialog.

## Example 2

### Lowpass Filter, Envelopes

This example will show you how to assign a DSP function to an algorithm block (the 4-pole lowpass filter), and adjust its control parameters. You'll also set up an envelope to control the cutoff frequency of the filter.

Start with Program 199, press EDIT, and select the KEYMAP page (This is explained in a bit more detail in the previous example, if you haven't read it). Change the value to "6 Ensemble Strings." String sounds are especially responsive to lowpass filtering, because they have a great deal of activity in the higher frequencies. Lowpass filters, depending on their cutoff frequencies, attenuate high frequencies, so a bit of adjustment can alter a string sound considerably.

Next, press the ALG soft button to select the ALG page. The center DSP function block will already be selected. Set its value to "4POLE LOWPASS W/ SEP." Notice how the sound changes. Press the more> soft button, and the three soft buttons that select the control input pages for the lowpass filter will appear. Press the F1 FRQ soft button.

The Coarse Adjust parameter will already be selected. Try a few different values for this parameter, to get a feel for its effect on the sound. Then set it to a value of G#3 208 Hz (Try holding down the ENTER button and striking G#3 on your controller's keyboard). Use the cursor buttons to select the Src1 parameter, and set its value to MPress (3, 3, ENTER). Cursor down once to the Depth parameter, and set it to a moderate value, like 3800 cents. Now try playing and applying pressure to the keys. (Obviously, this will have no effect if your MIDI controller doesn't send mono pressure.)

Cursor down to the Src2 parameter, and set its value to ENV2 (1, 2, 1, ENTER). Cursor down to the DptCtl parameter and set its value to ON. Cursor down to the MaxDpt parameter, and set its value to 5800 cents. As soon as you set a depth, you'll hear the envelope sweep the cutoff frequency. We'll adjust it further in a minute. The F1 FRQ page should now look like this:

```

edit Prog#11 FRQ04P LUT#55) >Layer:1/1
Coarse:G#3 208Hz Src1 :MPress
Fine :0ct Depth :3800ct
KeyTrk:0ct/key Src2 :ENV2
VelTrk:0ct DptCtl:ON
Pad :0dB MinDpt:0ct
MaxDpt:5800ct
<more F1 FRQ F2 RES F3 SEP F4 AMP more>
    
```

Next, press the F2 RES soft button to select the page for adjusting the resonance of the lowpass filter. Select the Src1 parameter, and set its value to Data (6, ENTER). Select the Depth parameter and set its value to 30.0 dB. This means that the partials at the cutoff frequency will be boosted 30 dB. The F2 RES page should look like the following diagram. If your MIDI controller has a data slider (or a control that can be programmed to send MIDI 06), the data slider (or programmable control) will affect the resonance of the filter.

```


Edit Prog: F2 RES(4F LDPASS) >Layer: 1/1
Adjust: 0.00dB          Src1 : Data
                        Depth : 30.00dB
                        Src2 : OFF
KeyTrk: 0.00dB/key    DptCtl: MWheel
VelTrk: 0.00dB        MinDpt: 0.00dB
                        MaxDpt: 0.00dB
<more F1 FRQ F2 RES F3 SEP F4 AMP more>
    
```

Next press either of the <more> soft buttons until you see the ENV2 soft button. Press it to select the ENV2 page. Here you'll program Envelope 2 to control the filter's cutoff frequency.

The Att1 time parameter will already be selected; set its value to 0.50 seconds. Press the right cursor button once to select the Att2 time parameter and set its value to 1.06 seconds. Cursor down once to select the Att2 level parameter and set its value to -1%. Cursor right once, then up once, selecting the Att3 time parameter. Set its value to 3.52 seconds. The ENV2 page should look like the diagram below. Play around with the sound a bit. Hold a few notes through the full length of ENV2; the sound will continue to change for several seconds. Work your controller's Mod Wheel and data slider through several different positions to get an idea of the range of variation you can add by using the two control sources in tandem.

```

Edit Prog: ENV2 [1/1] <>Layer: 1/1
Att1:Att2:Att3:Dec1:Rel1:Rel2:Rel3:Loop:
0.50 1.06 3.52 0s 0s 0s 0s Off
100% -1% 100% 0% 0% 0% 0% Inf
    
```



```

<more AMPENV ENV2 ENV3 ENUCTL more>
    
```

## Example 3

### Sample and Hold; Using a FUN

This example will use one of the FUNs to create a sample and hold program. As usual, start with Program 199, and press EDIT. While you're on the ALG page, select a value of "PARAMETRIC EQ" for the center DSP function block. Then select the KEYMAP page, and select the keymap 152 Dull Sawtooth.

Now press the more> soft button, then the F1 FRQ soft button. This enables you to set the frequency for the parametric EQ, which will set the depth of the modulation for the sample and hold function. The Coarse parameter will already be selected; set its value to D#4 311 Hz. Cursor down to the KeyTrk parameter, and set its value to 100 cents per key. Cursor over to the Src1 parameter, and set its value to FUN1 (1, 1, 2, ENTER). Cursor down once to the Depth parameter, and set its value to a fairly large negative number, like -4000 cents.

Next, press the F3 AMP soft button, to set the amplitude for the Parametric EQ (make sure you don't select F4 FINAL AMP). Set the value of the Adjust parameter to about 17 dB.

Now press either <more> soft button until you see the LFO soft button. Press the LFO button to select the LFO page. We'll program LFO1 as a square wave, then use it as one of the FUN's inputs.

Select the MnRate parameter for LFO1 and set it to a value of 5.00 Hz. Select the MxRate parameter and set its value to 9.40 Hz. Select the RateCt parameter and set it to a value of Data. Select the Shape parameter and set its value to Square. That's it for the LFO page.

Next press the FUN soft button to select the FUNs page. Select the Input A parameter for FUN1, and set its value to LFO1 (1, 1, 4, ENTER). Select the Input B parameter and set its value to RandV1 (1, 0, 8, ENTER). Finally, select the Function parameter, and set its value to "Sample B On A." Now when you play and hold a note, you'll hear the sample and hold effect. Use your MIDI controller's data slider (or a programmable control set to MIDI 06) to control the rate of the effect.

Here's what's happening. Every time the square wave LFO cycles, the value of the random signal generator (RandV1) modulates the frequency of the parametric EQ. There are lots of ways to set up sample and hold effects, but the FUN is the basic element. In this case, every time the value of the FUN's "A" input becomes greater than +.5, the value of the "B" input is sampled. That value becomes the FUN's output until the next time the value of the "A" input becomes greater than +.5 (see Chapter 16 for more). In this example, the FUN's output modulates the frequency of the EQ, and since it happens so rapidly, the effect is almost like a random tremolo.

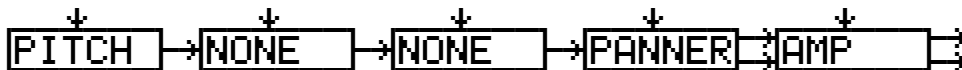
Now we'll apply the FUN to control the pitch of the sound as well. Press either of the <more> soft buttons until the PITCH soft button is visible, and press it to select the PITCH page. Select the Src1 parameter and set its value to FUN1. Select the Depth parameter and set its value to about 400 cents. Now the sound's pitch will fluctuate in sync with the EQ effect.

## Example 4

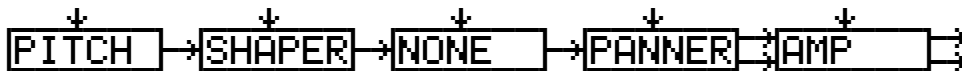
### SHAPER and PANNER

Our next example incorporates two of the DSP functions, and will give you a general overview of using the algorithms to build sounds.

Starting with Program 199, press EDIT, and while you're on the ALG page, cursor up to the Algorithm parameter, and select Algorithm 13. Select values of NONE in the F1 and F2 blocks. The value of the F3 block is already set to PANNER, as shown below.



Next press the KEYMAP soft button to select the KEYMAP page, and select 163 Sine Wave as the keymap. Play a few notes to accustom your ear to the sound of the unshaped sine wave. Then return to the ALG page, and select SHAPER for the algorithm's F1 block. You'll notice a bit of an effect right away,



Press the more> soft button, then press the F1 AMT soft button. This enables you to adjust the amount of shaping applied to the sine wave. Try different values for the Adjust parameter, then set it to its minimum—0.100x. Then cursor down to the VelTrk parameter, and set a value of 0.70. Listen for the variation in the effect as you play with different attack velocities.

Now go back to the KEYMAP page, and select different keymaps so you can hear the SHAPER's effect on different sounds. When you're finished experimenting, set the Keymap parameter at 152 Dull Sawtooth.

Notice how notes on the high end tend to break up. If you return to the F1 AMT page you can reduce this distortion and/or *aliasing* with the KStart and KeyTrk parameters.

Select the KStart parameter and set a value of C 2 Unipolar. This limits the amount of shaping applied to notes above or below C 2, depending on the value of the KeyTrk parameter. Next set the value for the KeyTrk parameter to -0.018x / key. Since we're using a negative value, the amount of shaping will decrease with higher notes.

Next we'll program an envelope to change the SHAPER in realtime. While still on the F1 AMT page, select the Src2 parameter, and set it to a value of ENV2 (1, 2, 1, ENTER). Cursor down to the DptCtl parameter and set it to a value of AttVel (1, 0, 0, ENTER). Set the MinDpt parameter to 0.00x and the MaxDpt parameter to a value of 1.70 x. This will let you use attack velocity to vary the speed of the envelope.

Next, use the <more> soft buttons to locate the ENV2 soft button, then press it to select the ENV2 page. Set up the parameters as shown in the diagram on the following page.

```

[Edit]:pos:=ENV2 [1/1] <>Layer:1/1
Att1:Att2:Att3:Dec1:Rel1:Rel2:Rel3:Loop:
0.10 0.82 0.86 1.04 0s 0s 0s Off
51% -23% 42% 0% 0% 0% 0% Inf

```

This can still be a little harsh on the high end when you play with high attack velocities. One way to smooth it out would be to go back to the ALG page, select a lowpass filter for the F2 block, and adjust its cutoff frequency to about F# 6. This is done by pressing EDIT when the F2 block is selected, then selecting the Adjust parameter and changing the value with any data entry method.



That’s it for the SHAPER example; we’ll continue with this program to describe the PANNER. You’ll notice that in Algorithm 13, PANNER is the only value available for the F3 block. The PANNER function takes a single signal from the sound engine, and splits it into two. These are referred to as the upper and lower wires. The upper and lower wires pass independently into the final Amp block, and from there to the audio outputs.

The parameters on the control input page for the PANNER let you distribute the signal between the upper and lower wires. You can send the signal all to the lower wire (an Adjust value of -100%), all to the upper wire (100%), or anywhere in between. This in itself won’t necessarily change the pan position of the current layer. It works in tandem with the Pan parameter on the layer’s OUTPUT page.

When a layer uses an algorithm that contains the PANNER function, you always have two wires going through the final Amp and to the audio outputs. Consequently, on the layer’s OUTPUT page, there are parameters to assign the output pair and pan position of each wire. When you have one wire panned hard left, and the other hard right, changing the parameters on the PANNER control input page will enable you to move the layer’s pan position in realtime. The closer a layer’s output is to the center of the stereo field, the less effective the PANNER function will be.

The first step in our PANNER example, therefore, will be to select the OUTPUT page. Select the Pan parameter for the upper wire, and set it all the way to the right. Select the Pan parameter for the lower wire and set it all the way to the left.

Now you can select the F3 POS page, and program it to move the sound around. First try changing the value of the Adjust parameter. You should hear the sound move to the left when the value is negative, and to the right when it’s positive. Set it back to 0%, then select the Src1 parameter. Select LFO1 as the value, then select the Depth parameter and set it to 100%. Now select the Src1 parameter again, and press EDIT. You’ll jump right to the LFO page. Set the MnRate parameter for LFO1 to a value of 0.1 Hz. Set the MxRate parameter to a value of 0.5 Hz, and the RateCt parameter to Data. Leave the Shape parameter set to Sine. You should hear the sound shift slowly from left to right as the LFO cycles. You can adjust the speed of the shift with your MIDI controller’s data slider (or a programmable control set to MIDI 06).



## Example 5

### Building a Drum Program; Using the Keymap Editor

With our next example, you'll learn how to build a drum program using the Program and Keymap Editors. To keep the example as brief as possible, we'll include only a few timbres and DSP examples. This won't make for a terribly realistic drum program, but it will give you the basic ideas you need to build your own. In this example, you'll create a four-layer program, with a different percussion timbre in each layer, each timbre having a different set of DSP functions applied.

Start with the default program 199. Press EDIT, then press the KEYMAP soft button. Select a value of 168 Silence. This gives you a keymap with a single key range from C 0 to G 10. Select the KeyTrk parameter, and change the value to 0. This will make the pitches of all the samples you assign the same on each key (you won't hear anything until you assign the samples). Next, press the <more soft button once, then press the DupLyr soft button. Layer 2 will be created. Repeat this twice, until you have four layers. If the K2vx tells you that you can't add any more layers, exit the Program Editor (don't bother to save at this point), check the current MIDI channel shown at the top right of the Program mode page, then go to Master mode and set the DrumChan parameter so that one of the eight available drum channels matches the current MIDI channel. Now you can start over, and the K2vx will allow you to add the fourth layer.

When you've created the four layers, you'll notice that the top line of the display tells you that you're looking at Layer 4 of a 4-layer program. Press the Up CHAN/BANK button to return to Layer 1. Press the LAYER soft button to select the LAYER page. Set the LoKey and HiKey parameters to C 4 and D 4 respectively. The easiest way to do this is to select the LoKey parameter, hold the ENTER button, and strike C 4 on your MIDI controller. Do the same for the HiKey parameter, striking D 4. Press the Up CHAN/BANK button to select Layer 2. Set its LoKey and HiKey to D# 4 and F 4, respectively. Repeat this for Layer 3, setting its LoKey and HiKey parameters to F# 4 and G# 4 respectively. Do the same for Layer 4, setting its LoKey and HiKey parameters to A 4 and B 4 respectively. This might be a good time to save what you've done so far.

Next, return to Layer 1 (CHAN/BANK button), press the KEYMAP soft button, select the Keymap parameter, and press EDIT to enter the Keymap Editor. Select the Sample parameter, and select a value of 20 14in Dry Tom-C 4. Press the Save soft button, and the K2vx will prompt you to save the keymap. Rename it as Tom, and save it. Press EXIT to return to the Program Editor, and select Layer 2 with the CHAN/BANK buttons. Press EDIT to return to the Keymap Editor, select the Sample parameter, and assign a value of 20 22in Dry Kick C 4. Save the keymap, renaming it Kick. Press EXIT to return to the Program Editor, and select Layer 3. Return to the Keymap Editor, select the Sample parameter, assign a value of 20 8in Dry Snare C 4, save the keymap—renaming it Snare. Return to the Program Editor, select Layer 4, select the Keymap parameter, and assign a value of 20 Vol Ft Cl Hat C 4. Save the Keymap, renaming it HiHat. You now have a 4-layer program, each layer having its own keymap with a different sample assigned to each one.

This is the basic process for creating any keymap and incorporating it into a program. In this case, we don't want the layers to overlap, and we want each layer to use a distinct keymap with its own sample assignment. In other programs, you might want to create a keymap with different timbres in a single layer, and you might want the layers to overlap.

As an example of how to quickly set up a multi-sample keymap, we'll change the sample assignment in Layer 1. Return to the Program Editor, and select Layer 1. Return to the Keymap Editor (by pressing EDIT while the Keymap parameter is highlighted on the EditProg\*KEYMAP page), then press the NewRng soft button. The K2vx will prompt you to strike a low and high key on your MIDI controller. Strike C 4 and D 4. Notice that the value of the Key Range parameter changes to reflect the new range assignment. Now select the sample

parameter, and turn the Alpha Wheel one click to the right, to select the sample 20 12in Dry Tom C 4. Save the keymap, replacing the earlier version. You can repeat this process to create as many new key ranges as you like (in this example, doing so would have no effect, since we've limited each layer to a narrow 3-key span).

If you wanted the layers to overlap, you would simply set each layer's LoKey and HiKey parameters (on the LAYER page in the Program Editor) to the same respective values. For example, you might set each LoKey parameter to C 2, and each HiKey parameter to C 7, causing all layers to play across five octaves.

Now we'll add some processing to some of the layers in our drum program example. The fact that each sound is on a different layer enables us to use a different algorithm for each layer, giving us enormous control over each sound.

Return to the Program Editor and select Layer 1. Press the PITCH soft button, and select the Coarse transpose parameter. Set its value to -3 ST. Select the KeyTrk parameter, and set its value to 400 cents per key. This will give you a different pitch on each key.

Next, select the PITCH page and select the Src2 parameter. Set its value to ENV2 (1, 2, 1, ENTER). Select the MaxDpt parameter and set its value to 300 cents. Select the DptCtl parameter, and set its value to VTRIG1 (1, 0, 6, ENTER). Press EDIT to select the VTRIG page, and set the VTrig1 Level parameter to a value of fff. Return to the PITCH page, select the Src2 parameter, and press EDIT to select the ENV2 page. Set it up as shown in the page below.

```

EditProg: ENV2 [1/1] <> Layer: 1/1
Att1:Att2:Att3:Dec1:Rel1:Rel2:Rel3:Loop:
0.02 0.16 0s 0.38 0s 0s 0s Off
28% -100 0% -76% 0% 0% 0% Inf
.....
↓
<more> AMPENV ENV2 ENV3 ENVCTL more>

```

Next, select Layer 2 and select the F4 AMP page. Set the Adjust parameter to 8 dB to give the kick a little more presence. Select the Src1 parameter and set it to a value of AttVel (1, 0, 0, ENTER). Select the Depth parameter and set a value of 10 dB. The kick will get considerably louder as you strike the keys harder.

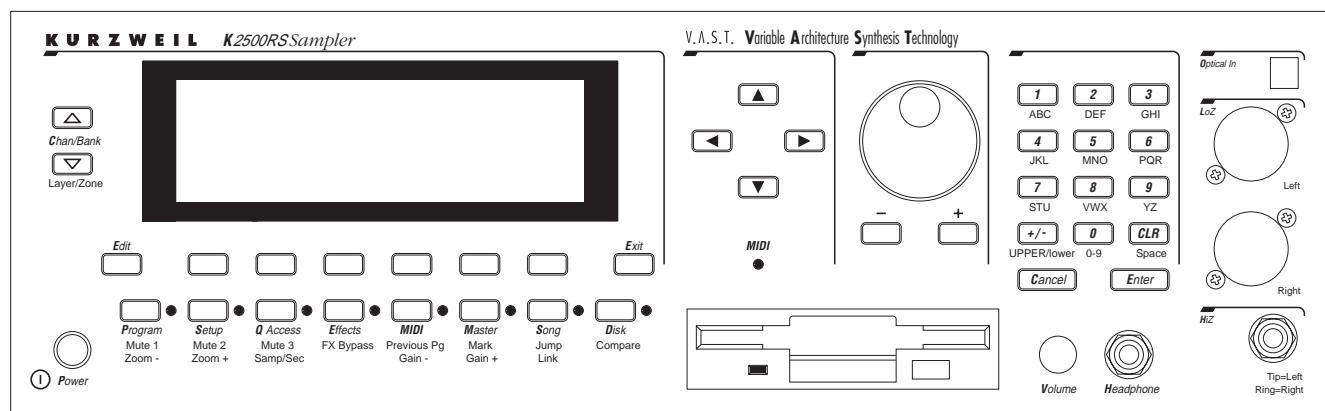
Select Layer 3, and press the ALG soft button. Select the center, 3-stage DSP function block and assign a value of HIFREQ STIMULATOR. Press the more> soft button once, then press the F1 FRQ soft button. Select the Coarse Adjust parameter, and assign a value of G 10 25088 Hz. Select the Src1 parameter and assign a value of MWheel. Select the Depth parameter and set it to a value of -10800 cents. Move your MIDI controller's Mod Wheel to bring out the snare's high end.

# Chapter 20

## Front Panel

### Front Panel Quick Reference

**Note:** Rack model is shown; keyboard models generally have the same features.



#### Volume Knob

Controls mixed audio outputs and headphone jack only. Does not send MIDI Volume (MIDI 07).

#### Mode Buttons

Press any of these eight buttons to enter the corresponding mode.

#### Chan/Bank Buttons

Scroll through the layers of the current program while in the Program Editor. Scroll through the zones in the current setup while in Setup mode. Scroll through the Quick Access banks while in Quick Access mode.

#### Edit Button

Functional in most modes. Press Edit to modify the currently selected object or parameter. If it's not editable, pressing Edit will do nothing.

There are editors available from every mode but Disk mode. The effect of pressing Edit in each of the modes is listed below.

#### When in this mode—Pressing the Edit button...

Program mode— ...enters the Program Editor, where you can edit the currently selected program. Chapter 6 in the *Performance Guide* covers the Program Editor.

Setup mode—	...enters the Setup Editor, where you can edit the currently selected setup. Chapter 7 in the <i>Performance Guide</i> describes the Setup Editor.
Quick Access mode—	...enters the Quick Access Editor, where you can change the program or setup assigned to the bank slot that was selected when you entered the Quick Access Editor. See Chapter 8 in the <i>Performance Guide</i> .
Effects mode—	...enters the Effects Editor, where you can edit the currently selected effects preset. Chapter 9 in the <i>Performance Guide</i> explains the Effects Editor.
MIDI mode—	...enters the Velocity Map or Pressure Map Editor if the Velocity or Pressure Map parameter is selected on either the XMIT page or the RECV page. See Chapter 17 in the <i>Performance Guide</i> . Enters the Program Editor if the Program parameter is selected on the CHANLS page. See Chapter 6 in the <i>Performance Guide</i> .
Master mode—	...enters the Velocity Map, Pressure Map, or Intonation Table Editor if the VelTouch, PressTouch, or Intonation parameter is selected.
Song mode—	...enters the Song Editor. The Song Editor is discussed in Chapter 12 in the <i>Performance Guide</i> . Enters the Program Editor if the Program parameter is highlighted when Edit is pressed.
Disk mode—	...has no effect.

### **Soft Buttons**

Functions change depending on current display page. Function of each button is displayed on bottom line of display.

### **EXIT Button**

Press to leave various editors. If you've made any changes while in the editor, you will be prompted to save them.

### **Cursor Buttons**

Press the corresponding button to move the cursor up, down, left, or right in the display. Different parameter values will be highlighted as buttons are pressed.

### **Alpha Wheel**

For data entry. Rotate clockwise to increase value of currently selected parameter, counterclockwise to decrease.

### **Plus / Minus Buttons (- and +)**

Under the Alpha Wheel. Press to increase or decrease the value of the currently selected parameter by the smallest possible amount.

### **Alphanumeric Pad**

#### ***For Numeric Characters***


Enter the value numerically instead of using the Alpha Wheel or Plus/Minus buttons. Press ENTER when finished. Press CANCEL to restore a parameter to its previous value. Pressing CLEAR is equivalent to pressing 0 without pressing ENTER.

#### ***For Alphabetic Characters***

When naming objects, you can use the alphanumeric pad to enter letters instead of numbers. If you're renaming a program, for example, just position the cursor under the character you want

to change, then press the corresponding numeric button, as labeled. Press the button as many times as necessary to enter the desired character. Pressing CLEAR will enter a space before the selected character. The "0" button will enter the numerals 0–9 when pressed repeatedly.

Here's an example. To enter the letter "C" in a blank space, press "1" three times. You can press the +/- button before or after entering the letter.

The CANCEL button is equivalent to the  soft button, and ENTER is the same as OK. The CLEAR button replaces the currently selected character with a space. The "+/-" button toggles between uppercase and lowercase letters.

When you press the +/- button on the alphanumeric pad, the currently selected character (the one with the cursor under it) will switch from upper case to lower case, and vice versa. The +/- button is a toggle; that is, if you switch from lower to upper case, all further entries will be in upper case until you press the +/- button again.

There are several punctuation characters available as well, but they can be entered only with the Alpha Wheel or Plus/Minus buttons. The punctuation characters are between "z" (lower case) and "0."

### Special Alphanumeric Pad Functions

When you're in Quick Access mode, the Alphanumeric pad can be used to select the entries in the current Quick Access bank. The layout of the alphanumeric pad corresponds to the layout of Quick Access bank entries as seen on the Quick Access mode page.

There's also a shortcut for selecting different QA banks while in QA mode. Just press the +/- or CLEAR button on the alphanumeric pad, and you'll be prompted to enter a bank number. Type the desired number on the alphanumeric pad, then press ENTER. The bank will be selected, and you'll return to the Quick Access page.

You can also use the alphanumeric pad to select strings to search for in the currently selected list of objects, and to enter new strings to search for. The search function is described fully in Chapter 3 in the *Performance Guide*.

### The Display

You may want to adjust the contrast of the display for different lighting conditions. The Contrast parameter in Master mode lets you set the contrast to your liking.

### MIDI LED

Lights when the K2vx is receiving MIDI information at its MIDI In port.

## Mode Button Functions

The mode buttons do more than select modes; when you're in the Program or Setup Editor, they have special functions, as indicated by the green labeling under each button. They also work as track mutes on the Mixer page of Song Mode.

PROGRAM / Mute 1

When you're in the Program Editor, this button will mute Layer 1 of the current program or the currently displayed layer for drum programs. While in the Setup Editor, it will mute Zone 1 of the current setup. On MIXER page of Song mode, mutes either track 1 or 9.

SETUP / Mute 2	When you're in the Program Editor, this button will mute Layer 2 of the current program, if any. For drum programs, solos currently displayed layer. While in the Setup Editor, it will mute Zone 2 of the current setup. On MIXER page of Song mode, mutes either track 2 or 10.
QUICK ACCESS / Mute 3	When you're in the Program Editor, this button will mute Layer 3 of the current program, if any. For drum programs, solos currently displayed layer. While in the Setup Editor, it will mute Zone 3 of the current setup. On MIXER page of Song mode, mutes either track 3 or 11.
EFFECTS / FX Bypass	When you're in the Program Editor, pressing this button will bypass (mute) the preset effect assigned to the current program, letting you hear just the sound of the layer(s) without effects. On MIXER page of Song mode, mutes either track 4 or 12.
MIDI / Prev pg	In the Program Editor, pressing this button will take you to the previously selected editing page. The K2vx remembers the four most recently selected pages, so you can press this button up to four times to backtrack through the pages you've viewed. Pressing it a fifth time will take you back to the ALG page. On MIXER page of Song mode, mutes either track 5 or 13.
MASTER / Mark	This is handy for marking Program Editor pages that you use frequently. Pressing this button will mark the currently selected page. You can mark as many pages as you like. Then you can use the Jump button to select the marked pages in the order you marked them. Marked pages will show an asterisk in the top line of the display, just before the name of the page. A marked page can be unmarked by pressing the Mark button while the page is visible. On MIXER page of Song mode, mutes either track 6 or 14.
SONG / Jump	Use this button to jump to pages in the Program Editor that you've marked with the Mark button. This will cycle through all the currently marked pages in the order they were marked. On MIXER page of Song mode, mutes either track 7 or 15.
DISK / Compare	This button works in most editors, and lets you compare your edits with the original version of the object you're editing. When you press the Compare button, the display changes to remind you that you're listening to the original version. Press any button to return to the currently selected page of whatever editor you're in. On MIXER page of Song mode, mutes either track 8 or 16.

# Chapter 21

## Programs, Setups, and Keymaps

### K2vx Program List

The 200 preset programs in the K2vx are organized by instrument category. You will find a few representatives of each instrument sampled for the base ROM soundset, as well as synthesized instrument emulations, commonly used synthesizer timbres, and templates for new programming. We hope you find it a good starting point for your own work.

There are many ways to put expressivity and variety in a single program by assigning MIDI controllers to the various DSP functions in its layers. This list describes how each of the 200 factory preset programs can be modulated or altered by the various MIDI controls. Only those controls which may not be immediately evident are listed. Controls such as attack velocity and keynumber are understood to be assigned to most programs.

Pgr#	Program Name	Mod Wheel	Data	MPress	Comments
<b>KEYBOARDS</b>					
1	Acoustic Piano				Soft Ped.
2	Stage Piano				Soft Ped.
3	BriteGrand				Soft Ped.
4	ClassicPiano&Vox	Choir Balance			Soft Ped.
5	Ballad Pno&Str	Strings Balance			Soft Ped.
6	Rock Piano 1				
7	Honky-Tonk	Tremolo			
8	E Grand & Pad	Pad Balance			
9	Classic E Piano				Soft Ped.
10	Dyno E Piano	Tremolo			
11	E Piano PF				
12	Suitcase E Pno	Vibrato Depth	Vibrato Rate		
13	Brite Klav	Layer Balance	8vb		
14	Match Stick	Vibrato		Vibrato, W/D Mix	

Programs, Setups, and Keymaps

K2vx Program List

Pgr#	Program Name	Mod Wheel	Data	MPress	Comments
<b>COMPING SYNTHS</b>					
15	<b>Big PWM</b>	Vibrato	Filter & Env Ctl	Vibrato	
16	<b>Matrix 12</b>	Vibrato	Filter & Env Ctl	Vibrato	
17	<b>OBX Braz 4</b>	Vibrato	Filter & Env Ctl	Vibrato	
18	<b>Memorymoog 4</b>	Vibrato	Filter & Env Ctl	Vibrato	
19	<b>Prophet Pulse 2</b>	Vibrato	Filter & Env Ctl	Vibrato	
20	<b>Prophet Square 2</b>	Vibrato	Filter & Env Ctl	Vibrato	
21	<b>New Shaper</b>	Vibrato	Env Ctl	Vibrato	
22	<b>Klicomp tree</b>	Vibrato, Filter Env	Detune, Env Ctl	Vibrato, Filter Env	
23	<b>Digicomp</b>	Vibrato	Env Ctl	Vibrato	
24	<b>Da Clav</b>		Env Ctl		
25	<b>Simpilton</b>	Vibrato	Env Ctl	Vibrato, Filter Ctl	
26	<b>Synth Caliope</b>	Vibrato	Env Ctl		
27	<b>Chiffloots</b>	Vibrato	Chiff Pitch	Vibrato	
28	<b>Bamboo Voices</b>	Vibrato	Alt Atk	Vibrato	
29	<b>Hyper Guitar</b>	Dly Vib		Dly Vib	
30	<b>Dreamers</b>	Tremolo		Tremolo	
31	<b>Pluxichord</b>	Env Ctl			
<b>LEAD SYNTHS</b>					
32	<b>Fluty Lead</b>	Vibrato		Vibrato	
33	<b>Gooshy Lead</b>	Distance	W/D Mix		
34	<b>Orient Wind</b>	Vibrato	W/D Mix	Vibrato, Filter Ctl	
35	<b>DC Lead</b>	Vibrato	Timbre Ctl	Vibrato	
36	<b>Duke's lead</b>	Vibrato	W/D Mix, Filter	Vibrato	
37	<b>FM Harmonica</b>	Vibrato		Tremolo	
38	<b>Mini Lead Poly</b>	Vibrato	Pitch	Vibrato	
39	<b>AlaZawi</b>	Timbre Ctl	Filter, Resonance	Vibrato	
40	<b>JR's Lead</b>	Vibrato	Timbre Ctl	Feedback	
41	<b>Funky Lead</b>	Vibrato	W/D Mix	Vibrato	
42	<b>Hammeron Synth</b>	Filter 1		Filter 2	
43	<b>Synthitar Lead</b>	Vibrato	W/D Mix	Vibrato	
44	<b>Modular Lead</b>	Vibrato	8 ve's	Vibrato	



Pgr#	Program Name	Mod Wheel	Data	MPress	Comments
45	<b>Prophet Sync</b>	Vibrato	Slave Osc Pitch		
46	<b>Brt Saxy Lead</b>	Vibrato	W/D Mix	Vibrato	
47	<b>Don Corllione'</b>	Vibrato		Vibrato	
<b>DRUMS</b>					
48	<b>Studio Kit 1</b>		W/D Mix		
49	<b>Studio Kit 2 MW</b>	Alt Atk	W/D Mix		
50	<b>2 Live Kits MW</b>	Alt Kit	W/D Mix		
51	<b>Rock Kit</b>	Alt Toms	W/D Mix		
52	<b>Jazz Kit</b>	Alt Atk	W/D Mix		Att Vel controls Hi Hat's decay
53	<b>Reggae Kit</b>		W/D Mix		Att Vel controls Hi Hat's decay
54	<b>Light Kit</b>		W/D Mix		
55	<b>Garage Kit MW</b>	Alt Kit	W/D Mix		Att Vel controls Hi Hat's decay
56	<b>Techno Kit</b>	Alt Kick	W/D Mix(Flange)		
57	<b>General MIDI Kit</b>		W/D Mix		
58	<b>Slam'n Drums</b>		W/D Mix(Flange)		
59	<b>Perc Section MW</b>	Cowbell, Shaker Enable	W/D Mix		
60	<b>Industry Set</b>	Resonance (A#4-C5)		Resonance (A#4-C5)	
61	<b>Techno Loops</b>	Loop Tempo	Loop Tempo		
62	<b>Rhythmic</b>	Disable Multilayer	Rhythm Tempo		
<b>PERCUSSION</b>					
63	<b>Dualimba</b>	Vibrato	Amp	Vibrato	
64	<b>TouchDrums</b>	Vibrato	Amp	Vibrato, Pitch	
65	<b>Hand Drums</b>		W/D Mix, Pitch Envelope		
66	<b>Dynamic Perc</b>	Heartbeat (C2) Fil- ter	W/D Mix, Heartrate (C2)		Mod Wheel Engages Non- tracking Congas
67	<b>Mark Tree</b>		W/D Mix, Env Ctl		
68	<b>Bell Player</b>	Vibrato		Vibrato	
69	<b>Marimba</b>	EQ Amp			
70	<b>Excited Marimba</b>	Alt Atk			

## Programs, Setups, and Keymaps

### K2vx Program List

Pgr#	Program Name	Mod Wheel	Data	MPress	Comments
<b>BASSES</b>					
71	Dual E Bass	Vibrato		Vibrato	
72	Warm Bass	Vibrato		Vibrato	
73	Sustain E Bass	Vibrato		Vibrato	
74	Ripper Bass	Vibrato		Vibrato	
75	Yama Bass		Layer Balance		
76	Synth Fretless	Vibrato		Vibrato	
77	Fretless Lead	Vibrato		Vibrato	
78	Moogy Bass 1	Vibrato	Filter	Vibrato	
79	Moogy Bass 2	Filter Depth	Resonance Depth	Vibrato	
80	Mix Bass	Filter Depth	Layer balance		
81	Tite Rave Bass	W/D Mix	Filter, Depth		
82	Synth Bass	Filter Ctl			
83	House Bass	Vibrato	Filter	Vibrato	
<b>GITARS</b>					
84	Acoustic Guitar				SoftPd
85	Steel Str Guitar	Vibrato	EQ	Vibrato	
86	12-str Guitar		EQ		SoftPd
87	Strummer Guitar	Vibrato	EQ	Vibrato	
88	Slo Chorus Gtr	Tremolo	EQ		
89	Captain Crunch	Vibrato	Filter , W/D Mix	Vibrato	PBend goes +2 and -12ST
90	Smooth Lead		Shaper, W/D Mix	Feedback	
91	Dist Harmonics	Tremolo		W/D Mix	
92	Kotolin	EQ		Vibrato	
93	Cee Tuar	Vibrato	Alt Sound	Vibrato	
94	Green Acres				
<b>ORGANS</b>					
95	Perc Organ 2500	Rotary Speaker	Perc Balance		
96	Ballad Organ 2	Rotary Speaker			
97	Gospel Organ	Rotary Speaker	Perc Balance		
98	Drive Organ	Rotary Speaker	Distortion Ctl		
99	Rotating B's & M's	Rotary Speaker			

Pgr#	Program Name	Mod Wheel	Data	MPress	Comments
100	<b>Cheeze</b>	Vibrato Depth	Env Ctl	Vibrato Depth	
101	<b>Tamborgan</b>	Vibrato	Perc Pitch	Vibrato	
102	<b>Organ Pad</b>	Tremolo			
103	<b>Chiffy Pipes</b>	Decrescendo			
104	<b>Offertory</b>	Layer Balance			
105	<b>Pedal Pipes</b>	Decrescendo			
106	<b>Church Organ</b>	Vibrato	Layer Balance	Vibrato	Velocity Sensitive
107	<b>Resorgan</b>	Dynamics			Sost Pdl Does Release Ctl
<b>STRINGS/CHOIR</b>					
108	<b>Fast Strings</b>	Filter			
109	<b>Att ctl Fast Str</b>	Filter			For Fast Solo Lines& Active Comping
110	<b>Att ctl Med Str</b>	Env Ctl			For Med or Slow (MW) Solo Lines
111	<b>SfzTrem Strings</b>				Sfz Envelope Triggered by Increased Att Vel
112	<b>ClassicalStrings</b>			Vibrato Depth	For Light, Active Comping
113	<b>SloClassical Str</b>	Decrescendo			For Chordal Comping
114	<b>Silk Strings</b>	Decrescendo	W/D Mix	Vibrato Depth	For Chordal Comping
115	<b>Fast Violin</b>			Vibrato Depth	
116	<b>Slo Solo Cello</b>	Quick Fade	W/D Mix	Vibrato Depth	
117	<b>Stereo Slo Str</b>	Filter			Velocity Controls Timbre Shift
118	<b>Cathedral Choir</b>	Decrescendo			Att Ctl Allows Smooth Voice Leading in 4-Part Playing
119	<b>Mixed Choir</b>	Layer Balance			
120	<b>The Choir</b>	Vibrato	Release Env	Vibrato	
<b>WINDS</b>					
121	<b>Wendy's Flute</b>				Higher Att Vel Equals Less Tremolo
122	<b>Treble Flute</b>				
123	<b>Baroque Flute</b>				
124	<b>Soft Tenor Sax</b>		W/D Mix	Vibrato, Filter	
125	<b>Fast Solo Tenor</b>	EQ		Vibrato	For Fast Legato Lines

Programs, Setups, and Keymaps

K2vx Program List

Pgr#	Program Name	Mod Wheel	Data	MPress	Comments
<b>BRASS</b>					
126	<b>Dynamic Trumpet</b>	Swell	W/D Mix	Vibrato	
127	<b>Miles Unmuted</b>	Vibrato	Timbre Ctl	Vibrato	
128	<b>Strght Mute Trpt</b>	Vibrato Defeat	W/D Mix	Vibrato Rate	
129	<b>Almost Muted</b>	Vibrato, Amp	Timbre Ctl	Vibrato	
130	<b>Solo Trombone</b>		W/D Mix	Vibrato Depth	
131	<b>Sfz Bone</b>				
<b>ENSEMBLES</b>					
132	<b>Trumpet Section</b>			Swell	
133	<b>Hip Brass</b>	Vibrato	W/D Mix	Swell	
134	<b>Brt Miami Brass</b>	Bright Swell			
135	<b>Orchestral Brass</b>			Swell	
136	<b>Sax Section</b>	Swell	W/D Mix	Swell	Sfz Envelope
137	<b>Dyn Big Band</b>	Softer		Swell	Sfz Envelope
138	<b>Flute &amp; Slo Str</b>	Solo String Swell		Vibrato Depth	SostPd disables Solo Str. Release Velocity Controls Strings Release
139	<b>Horn&amp;Flute w/ Str</b>	Strings Balance			Rel. Velocity Controls Strings Rel.
140	<b>DynamicOrchestra</b>  Light playing engages a horn, flute & string ensemble. SostPed holds a chord and engages an ensemble suited to solo lines. At forte, bigger brass is enabled, with pressure controlling a swell.				
141	<b>Touch Orchestra</b>  Mod wheel replaces RH trumpets with solo flute. Each Velocity level brings in a new instrument: At forte, horns are doubled; at double forte, kettle drums play; at <i>fff</i> , crash cymbal plays.				
142	<b>Slo Ensemble</b>	Synstring Res	Synstring Filter	Vibrato Depth	
143	<b>W Tell Orchestra</b>	Swell		Swell	For Active Marcato Comping SostPD Latches and Disables Brass
144	<b>Jazz Band</b>  LH Bass layered with ride for walking rhythm section. Data slider switches from guitar to horn section; SostPed holds brass and adds solo Tenor. Throw the ModWhl for drum solo.				
145	<b>Rock Quartet</b>  LH Bass layered with hihat for driving rhythm section. At forte, kick, snare, and rhythm guitar are added. Mod wheel does rotary speakers for the organ.				

Pgr#	Program Name	Mod Wheel	Data	MPress	Comments
<b>HYBRID SYNTHS</b>					
146	<b>Gargantuanism</b>	Vibrato	Layer Balance	Vibrato	SostPd Disables Strings
147	<b>Tranquil Pluck</b>	Vibrato	Release Ctl	Vibrato	
148	<b>The Chase</b>	Vibrato	W/D Mix	Vibrato	
149	<b>Enterprize</b>	Tremolo	Bell Pitch	Tremolo	
150	<b>Magic Orchestra</b>		Piano Balance		SostPd Disables Cymbal
151	<b>Passion Source</b>	Detune		Detune, Swell	
152	<b>Microwave</b>	Vibrato	Release Ctl	Vibrato	
153	<b>Fuzz Lite</b>	Vibrato	Release Ctl	Vibrato	
154	<b>Solina Phaze</b>	Vibrato	Phaser Rate	Vibrato	
155	<b>Arystal</b>		Layer Balance		
156	<b>Timershift</b>	Vibrato	Release Ctl	Vibrato	
157	<b>Aurora</b>	Mod Speed	W/D Mix		
158	<b>Gongers</b>	Vibrato	Pitch, W/D Mix	Vibrato	
159	<b>Arrakis Grand</b>	Vibrato Depth	Detune	Vibrato Depth	
160	<b>Sisternal</b>	Vibrato	Release Ctl	Vibrato	
161	<b>PPG 4</b>	Vibrato	W/D mix, Env Ctl	Vibrato	
162	<b>Pseudomento MW</b>	Pseudomento Rate			
163	<b>Big Strings</b>	Vibrato		Vibrato	
164	<b>Spaced</b>	Dly Sweep	Mod Rate		
165	<b>Digital Choir</b>	Res Mod	Filter Ctl		
166	<b>Meditation Pad</b>	Res Mod	Filter Ctl	Vibrato	Move data slider from top to bottom and throw the MW for best resonance effect.
167	<b>The Cymbal Sings</b>	Vibrato		Filter	
168	<b>Slo FlangeStrngs</b>	Flange Rate			Template: Using allpass to create flanging within a layer
169	<b>Lushlife</b>				
170	<b>Crystal too</b>	Vibrato	Sine Wave Pitch	Vibrato	
171	<b>Multi Marimba</b>	Vibrato	Delay Time	Vibrato	
172	<b>Wave Power</b>	Vibrato		Vibrato	
173	<b>Noo Mutes</b>	Pitch Effect	Filter Ctl, W/D Mix		
174	<b>Hammer Violin</b>	Pitch Effect	Env Ctl, W/D Mix	Vibrato	

**Programs, Setups, and Keymaps**

K2vx Program List

<b>Pgr#</b>	<b>Program Name</b>	<b>Mod Wheel</b>	<b>Data</b>	<b>MPress</b>	<b>Comments</b>
175	<b>Mallet Flutes</b>	Chiff Pitch	W/D Mix		
176	<b>Malletoo</b>	Pitch Mod Depth	Pitch Mod Rate		
177	<b>Ethereal Echoes</b>	Disable Bell			
<b>PAD SYNTHS</b>					
178	<b>Padifier</b>	Vibrato	Filter	Vibrato	
179	<b>Spaced Inn</b>	Vibrato		Vibrato	
180	<b>Glass Bow</b>	Vibrato	Env Ctl	Vibrato	
181	<b>Angel Pass</b>				
182	<b>In the Air</b>	Vibrato	Filter Ctl	Vibrato	
183	<b>Matrix Mellostr</b>	Vibrato	Env Ctl	Vibrato	
184	<b>Ethereal Strings</b>	Filter Sweep		Filter	
185	<b>Dawning</b>	Sweep Fade			
186	<b>Synth Strings</b>				
187	<b>Choir Fixer</b>	Vibrato, Pan	W/D Mix	Vibrato	
188	<b>Shine On</b>	Vibrato	Filter Ctl	Vibrato	
189	<b>ChoirStrings</b>				
190	<b>Launch Pad</b>	Timbre Ctl		Timbre Ctl	
<b>EFFECTS SOUNDS</b>					
191	<b>cymbal thing</b>				
192	<b>a no way CS</b>	Mod Depth	Mod Rate		
193	<b>Environments</b>				
194	<b>Gremlin Group</b>	Timbre Ctl, Env Ctl	Vibrato>Pitch, Resonance		
195	<b>Thunder Storm</b>				Play Sparse Staccato Notes in LH for Thunder. RH is Rain
196	<b>Northern Winds</b>			Pan Rate, Filter	
197	<b>Doomsday</b>	Pitch			
<b>UTILITY</b>					
198	<b>Click</b>				Assigned to output group B (dry)
199	<b>Default Program</b>				Used in New Lyr, Sample audition, and Preview Program

## Setup List

The Performance Setup, or "Setup" is a combination of three zones, each with independent MIDI channel and controller transmission assignments. Designed initially for models with built-in keyboards, Setups can be played on a K2500R via the Local Keyboard Channel feature: Find this parameter in MIDI mode on the RECV page, change it from None to a channel of your choice, and set your controller to send on only that channel. Now, any note that comes in on that channel will be re-mapped according to the display channel (in program mode) and according to the Setup (in Setup mode).

Here is a list of the 100 Setups provided in the Factory Objects ROM:

ID#	NAME	ID#	NAME	ID#	NAME
1	Majesty	35	Age of Atlantis	69	FatSo
2	Sahara	36	Tinged Piano	70	Festival Brass
3	Full Orchestra	37	Combi Clav	71	WindyOrch MWPR
4	Hyper Funk	38	Dynomatch	72	8ve Slo Orch
5	LowCsGroove	39	Clav EP Organ	73	Slo tutti Orch
6	Waviator	40	Mallet Pad	74	Fast tutti Orch
7	Air Guitar	41	Organomics	75	Pno & Stereo Ens
8	PowerLead	42	Match Point Splt	76	PastaMan
9	TotalEnvironment	43	Pipes 1	77	HammerFlange
10	Combi Pluck	44	Pipes 2	78	Spaced Strings
11	Stop Time	45	Phantom Organ	79	Mist Strings
12	Ultra Clav	46	Shadow Pad	80	Crunch&Lead
13	Thru The Void	47	Spiral Pad	81	Ethnic & Analog
14	BigBee	48	Crystal Pad	82	Dulcimer Gtr
15	Dirty BigBee	49	Tranquil Waves	83	Floyd Wheel
16	Daze of Analog	50	Melody Voyage	84	Man of Steel
17	Bell Bliss	51	RoundTheSquare	85	2 Hand Comp
18	Echo in Time	52	Desert Song	86	Guitar Pad
19	Royal Prog	53	Soaring Solo	87	Big Gtr/CymSings
20	Choirz	54	FewshunLead	88	GB Band
21	SummerSnows	55	Joyful Lead	89	StThomas
22	Armenian Song	56	Perc 5ths	90	RastaSauce
23	Vital Wave	57	Big Combi Perc	91	Gtrs & Lead
24	Chordal Passion	58	Swing Trio	92	HoDown
25	Bell Voyage	59	EP Trio	93	Piano/Cello
26	Micro Magic	60	Pno Trio	94	3 Martini Shade
27	E Piano Pipes	61	Power Play	95	Synth Ensemble
28	Avalon	62	Big Jazz	96	Rhythm/Chiff
29	Celeste Pad	63	LivDrum/DualBass	97	Rhythm/SynLead
30	Starlight Piano	64	TekDrum/RaveBass	98	Noo Dance
31	Windowpane EP	65	BigTekDrm/RavBas	99	Earth & Sky
32	Grand & Electric	66	The Falling Sky	100	Basic Setup
33	EP & Strings	67	Inner Wave		
34	MalleTines	68	GigBrass		

## Storing Objects in the Memory Banks

The number of available IDs differs between object types, and depending on whether you are storing the object to the Zeros bank or one of the other 9 banks.

OBJECT TYPE	NUMBER OF OBJECTS AND ID RANGE	
	IN ROM	IN RAM
Sample, Keymap, Program, Setup	100 1—99	100 200—299
	100—199	300—399 . . 900—999
A total of 999 objects of these types can be stored, 99 of each type in the Zeros bank, and 100 of each type in every other bank.		

Quick Access Banks, Songs, Velocity Maps, Pressure Maps, Intonation Tables	75 1—75	20 100—119
		200—219 . . 900—919
A total of 255 objects of these types can be stored, 75 of each type in the Zeros bank, and 20 of each type in every other bank.		

Preset Effects	37 1—37	10 100—109
		200—209 . . 900—909
A total of 127 preset effects can be stored, 37 in the Zeros bank, and 10 in every other bank.		



## K2500 ROM Keymaps

ID#	Keymap	ID#	Keymap	ID#	Keymap	ID#	Keymap
0	None	58	Ambient Snare 2	110	Solo Strings	158	Buzz Wave
1	Grand Piano	59	Ambient Snare 3	111	Six String Mutes	159	Hi Formant Wave
2	Dual Elec Piano	60	Cross Stick	112	Oboe Wave	160	PrimeNumberWave
3	Hard Elec Piano	62	10in Dry Tom	113	Clav Wave	161	Triangle Wave
4	Soft Elec Piano	63	12in Dry Tom	114	Elec Piano Wave	162	Third Wave
5	Voices	64	15in Dry Tom	115	Bell Wave	163	Sine Wave
6	Ensemble Strings	65	13in Amb Tom	116	Ping Wave	164	ExtDynPrtls1
7	Elec Jazz Guitar	66	15in Amb Tom	117	Drawbars 1-3	165	ExtDynPart2
8	Acoustic Guitar	67	16in Amb Tom	118	Drawbars 1-4	166	ExtDynSaw
9	5 String Guitar	68	Reversals	119	Drawbars 1-3 Dist	167	Mellow Vox
10	Dual E Bass	69	Reverse Bell	120	Full Drawbars	168	Silence
11	Elec Pick Bass	71	Bidir Amb Kick 1	121	Drawbars 1-4,8	169	Synflute Brt
12	Elec Slap Bass	72	Reverse Snare	122	Organ Wave 1	170	Synflute mello
13	Finger Atk Bass	73	Conga Bass	123	Organ Wave 2	171	SlapBass/Guitar
14	Flute	74	Conga Slap	124	Organ Wave 3	172	Mello Vox 2
15	Tenor Saxophone	75	Conga Tone	125	Organ Wave 4	173	Shift Guitar 2
16	Sax no Altissimo	76	Syn Conga Tap	126	Organ Wave 5	174	Single Mute
17	Trumpet	77	Timbale	127	Organ Wave 6	177	Fingered Bass 2
18	Trombone	78	Timbale Shell	128	Organ Wave 7	178	Ext Dual Bass
19	Trombone/Trumpet	79	Cabasa	129	Partials 1-3	179	Syn Bass Pick
20	Bone/Trp 2	80	Clave	130	Partials 4-7	180	Syn Bass Slap
21	Trombet	81	Cowbell	131	Partials 8-12	181	Shift Guitar
22	Trumpbone	82	Tambourine	132	Partials 13-20	182	Syn Guitar
23	Preview Drums	83	Handclaps	133	Partials 21-30	183	Syn Voices
24	Dry Kit1	84	Reverse Crash	134	Partials 1&2	184	Syn voices 2
25	Dry Kit2	85	Reverse Clsd Hat	135	Partials 3&4	185	Perc Voice
26	Amb Kit 1	86	Reverse Open Hat	136	Partials 5-7	186	Synstrings 1
27	Amb Kit 2	87	Reverse hat loop	137	Partials 8-10	187	Synstrings 2
28	Amb Kit 3	88	Chiff	138	Partials 11-15	188	Syn Piano
29	2 8ve Dry Kit	89	Chirp	139	Partials 16-21	189	Funny Perc
30	General MIDI Kit	90	FM Bell Trans	140	Partials 2-4	190	TechnoLoops
39	Ride Rim Cymbal	91	Waterphone	141	Partials 5,7,9,11	191	Hat Loop
40	Ride Bell Cymbal	92	Metal Clank	142	Partials 1,2,4	192	[for New Piano]
41	Crash Cymbal	93	TimbaleShell Atk	143	Partials 1,2,4,6	193	[for New Piano]
42	Closed Hihat	94	Cowbell Atk	144	Partials 3-5	194	[for New Piano]
43	Slt Open Hihat	95	Timbale Atk	145	Partials 1-3	195	[for New Piano]
44	Open Hihat	96	Bell Attack	146	Partials 1,3,5	196	[for New Piano]
45	Open>Close Hihat	97	Clave Atk	147	Partials 1&4	197	[for New Piano]
46	Foot Close Hihat	98	Wood Bar Atk	148	Partials 1&6	198	[for New Piano]
47	Dry Kick 1	99	Conga Tone Atk	149	Partials 1&8	199	Silence
48	Dry Kick 2	100	Conga Slap Atk	150	Partials 1&12		
49	Amb Kick 1	101	Elec Pno Atk	151	Sawtooth		
50	Amb Kick 2	102	Brass Attack	152	Dull Sawtooth		
51	Amb Kick 3	103	Bow Attack	153	Very Dull Saw		
52	DrySnare 1 Soft	104	Jazz Guitar Atk	154	Square Wave		
53	DrySnare 1 Hard	105	Steel Guitar Atk	155	Dull Square		
54	Dual Dry Snare 1	106	Perc Atk 1	156	Very Dull Square		
55	Dry Snare 2	107	Perc Atk 2	157	Buzzy Square		
56	Dry Snare 3	108	Perc Atk 3				
57	Ambient Snare 1	109	Wood Bars				



## Chapter 22

### Effects

#### List of Factory Preset Global Effects and Their Configurations

ID#	Name	Configuration
1	Sweet Hall	Ultimate Reverb
2	Small Hall	Room Simulator
3	Mediuim Hall	Ultimate Reverb
4	Large Hall	Ultimate Reverb
5	Big Gym	Room Simulator
6	Bright Plate 1	Ultimate Reverb
7	Opera House	Ultimate Reverb
8	Live Chamber	Room Simulator
9	Bathroom	Ultimate Reverb
10	Med Large Room	Room Simulator
11	Real Room	Ultimate Reverb
12	Drum Room	Room Simulator
13	Small Dark Room	Room Simulator
14	Small Closet	Ultimate Reverb
15	Add Ambience	Room Simulator
16	Gated Reverb	Gated Reverb
17	Reverse Reverb	Reverse Reverb
18	Non-Linear	Ultimate Reverb
19	Slapverb	Room Simulator
20	Full Bass	Chorus+Delay+Room+Mixer
21	Room & Delay	Delay+Room+Mixer
22	Delay Big Hall	Delay+Hall+Mixer
23	Chorus Room	Chorus+Room+Mix
24	Chorus Smallhall	Chorus+Hall+Mix
25	Chorus Med Hall	Chorus+Hall+Mix
26	Chorus Big Hall	Chorus+Hall+Mix
27	Chor-Delay Room	Chorus+Delay+Room+Mixer
28	Chor-Dly Hall	Chorus+Delay+Hall+Mixer
29	Flange-Dly Room	Flange+Delay+Room+Mixer
30	Flange-Dly Hall	Flange+Delay+Hall+Mixer
31	Stereo Chorus	Stereo Chorus
32	Stereo Flanger	Stereo Flange
33	Stereo Delay	4-Tap Delay
34	4 Tap Delay	4-Tap Delay
35	Chorus Delay	Parametric EQ+Chorus+Delay+Mixer
36	Flange Delay	Parametric EQ+Flange+Delay+Mixer
37	Chorus 4 Tap	EQ+Chorus+4 Tap Delay+Mixer
100	Flange 4 Tap	EQ+Flange+4 Tap Delay+Mixer
101	Chorus Echo	EQ+Chorus+4 Tap Delay+Mixer
102	Chorus Echoverb	EQ+Chorus+4 Tap Delay+Mixer
103	Fast Flange	Stereo Flange
104	Wash	Chorus+Delay+Hall+Mixer
105	Into the Abyss	Chorus+Delay+Hall+Mixer
106	Space Flanger	EQ+Flange+4 Tap Delay+Mixer
107	Flange Room	Flange+Delay+Hall+Mixer
108	Predelay Hall	Delay+Hall+Mixer
109	Flange Echo	EQ+Flange+4 Tap Delay+Mixer

## Effects Controller Numbers

The K2vx uses the Digitech 256 chip for its effects processor. When in Program Mode, the operating system allows you to assign any MIDI controller to Wet/Dry Mix plus two additional parameters related to the effect. (The choice of controllable parameters changes depending on the effect configuration. They are the parameters you see when you go to edit the effect.)

But if the FX Mode parameter on the Effects page is set to Master, then the remapping that takes place within the program is not applied. Instead, you use a predefined set of controller numbers. So to control the effects processor in real time when FX Mode is set to Master, you must use the following controller numbers.

To control the effects processor in this manner, press the EFFECTS button. Change FX Mode to Master, and set FX Chan to the channel you will use to send the controller info. (You can also send program changes on this channel to switch effects, so it is usually best to pick a channel that is not being used for notes.) These settings are remembered as long as the Power Mode parameter on the MIDI receive page is set to User. Otherwise, you will have to re-enter the settings each time you power up.

In the following chart, the parameters are grouped by Configuration. Multi FX will contain parameters found in more than one configuration.

### Ultimate Reverb

Decay time	77
Room volume	78
HF damping	14
Envelopment	32
Early Delay	29
Early Diffusion	30
Later Delay	n/a
Later Diffusion	93
Dry Level	28
Early Level	31
Later Level	92

### Room Simulation

Gross Size	90
Decay Time	15
Listening Position	79
HF Damping	14
Dry Level	28
Reverb Level	88

### Reverb in Multi FX

RevPre-Delay	80
Hi-Freq Damp	83
Reverb Decay	82
Revin dry	86
Revin flange	87
Revin Delay	85

**Gated Reverb**

Pre-Delay	80
Decay Time	38
Envelope	39
Accent Dly	01
Dry Level	57
Accent Level	02
Right Level	62
Left Level	94

**Reverse Reverb**

Pre-Delay	80
Reverse Time	89
Accent Delay	01
Accent Level	02
Dry Level	57
Right Level	66
Left Level	65
Accent Lvl L	50
Accent Lvl R	51

**Parametric EQ**

Band 1 Freq	03
Band 1 level	04
Band 2 Freq	05
Band 2 level	06
Band 3 Freq	07
Band 3 level	08
EQ level	58

**Graphic EQ**

63 Hz	41
125 Hz	43
250 Hz	45
500 Hz	47
1.0 Khz	42
2.0 Khz	44
4.0 Khz	46
8.0 Khz	48

**Stereo Chorus**

Chorus Delay	10
LFO Speed	13
LFO Depth	12
Dry Level	57
Right Level	54
Left Level	52
Chorus Level	53

**Stereo Flange**

Flange Delay	33
LFO Speed	37
LFO Depth	36
Feedback	35
Dry Level	57
Right Level	61
Left Level	59
Flange Level	60

**Stereo Delay**

Delay Time	22
Feedback	17
Dry Level	57
Right Level	56
Left Level	55
DelayDry In	19
Delay Chr In	18
Delay Flg In	21
Delay EQ In	20
Delay EQ Src	16

**4 Tap Delay**

Tap 1 Delay	24
Tap 2 Delay	25
Tap 3 Delay	26
Tap 4 Delay	27
Feed Delay	23
Feedback	17
Dry Level	57
Tap 1 Level L,R	67, 68
Tap 2 Level L,R	69, 71
Tap 3 Level L,R	72, 74
Tap 4 Level L,R	75, 76
Tap 2 Level	70
Tap 4 Level	73

**Other**

Wet/Dry Mix	91
Bypass	09
LowPass Filter	49

**NOTES:**

In the interest of signal to noise performance, the effects dry level parameter should be left at 0 and the Wet/Dry mix (91) should be used instead.

Some of the above parameters may only be found in a Multi FX patch.

# Chapter 23

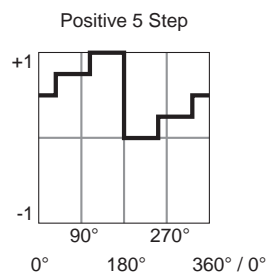
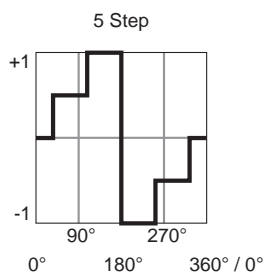
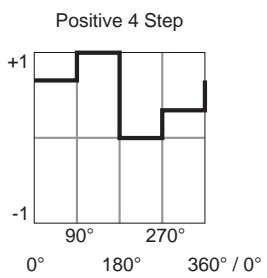
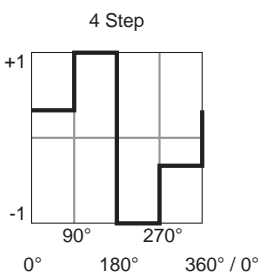
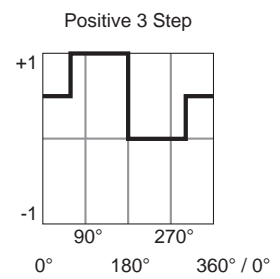
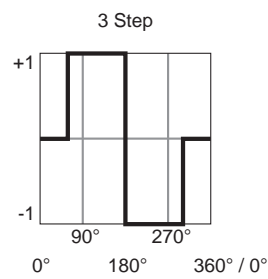
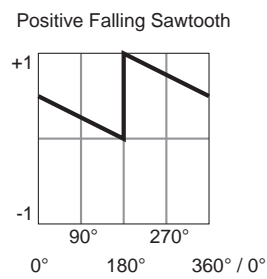
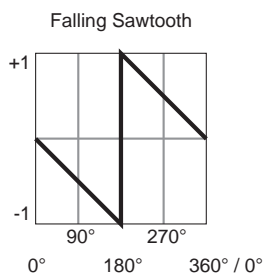
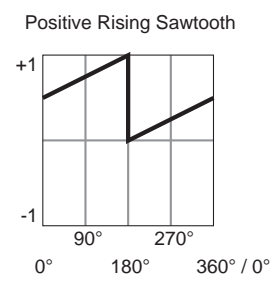
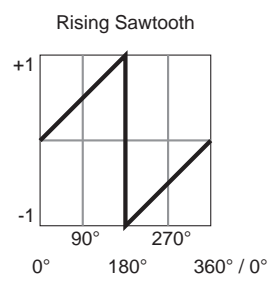
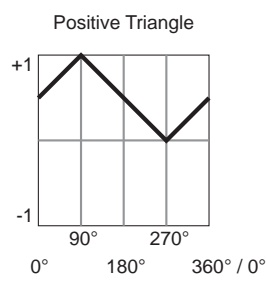
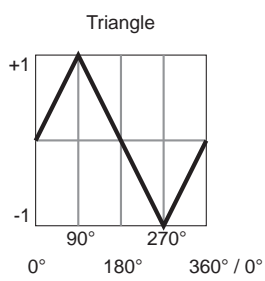
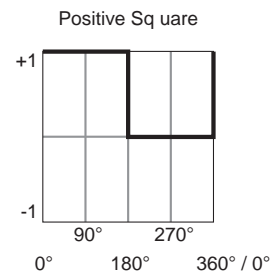
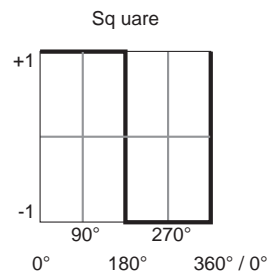
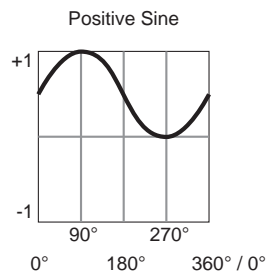
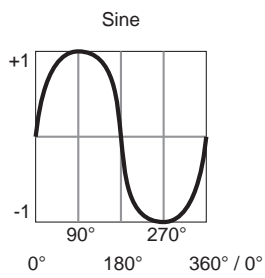
## LFOs

### LFO Shapes

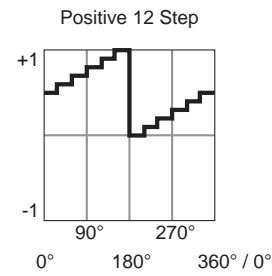
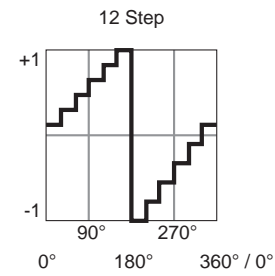
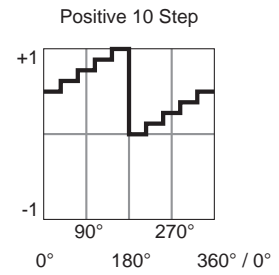
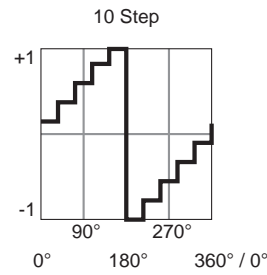
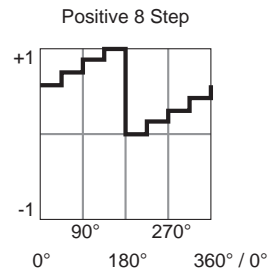
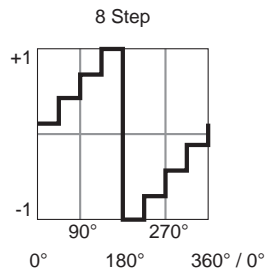
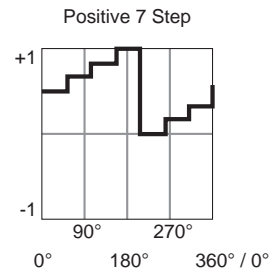
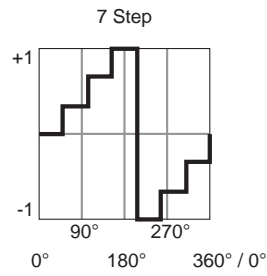
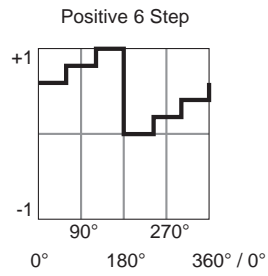
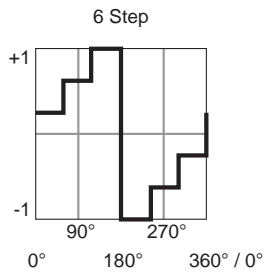
LFO Shape	Displayed:
Sine	Sine
Positive Sine	+Sine
Square	Square
Positive Square	+Squar
Triangle	Triang
Positive Triangle	+Trian
Rising Sawtooth	Rise S
Positive Rising Sawtooth	+Rise
Falling Sawtooth	Fall S
Positive Falling Sawtooth	+Fall
3 Step	3 Step
Positive 3 Step	+3 Ste
4 Step	4 Step
Positive 4 step	+4 Ste
5 Step	5 Step
Positive 5 Step	+5 Ste
6 Step	6 Step
Positive 6 Step	+6 Ste
7 Step	7 Step
Positive 7 Step	+7 Ste
8 Step	8 Step
Positive 8 Step	+8 Ste
10 Step	10 Ste
Positive 10 Step	+10 St
12 Step	12 Ste
Positive 12 Step	+12 St

# LFOs

## LFO Shapes







**LFOs**

---

LFO Shapes

# Chapter 24

## Note Numbers and Intonation Tables

### K2vx Note Numbers and MIDI Note Numbers

K2vx	MIDI
C -1—B -1	0—11
C 0—B 0	12—23
C 1—B 1	24—35
C 2—B 2	36—47
C 3—B 3	48—59
C 4 (Middle C)—B 4	60—71
C 5—B 5	72—83
C 6—B 6	84—95
C 7—B 7	96—107
C 8—B 8	108—119
C 9—G 9	120—127

You can assign samples to keymaps in the range from C 0 to G 9. The K2vx will respond to MIDI events in the octave from C -1 to B -1. If a Note On event is generated in the range from C -1 to B -1, the K2vx will respond by setting the Intonation key correspondingly (C -1 will set it to C, C# -1 will set it to C#, etc.)

### Note Numbers for Percussion Keymaps

Most of the K2vx's percussion programs have keymaps that place the various percussion timbres at standardized key locations. There are eight drum keymaps: Preview Drums, five 5-octave kits (two dry and three ambient), a 2-octave kit, and the General MIDI kit. The keymap 30 General MIDI Kit adheres as closely as possible to the General MIDI standard for placement of timbres. As a rule, programs that use this keymap can be assigned in percussion tracks for prerecorded sequences and will play appropriate timbres for all percussion notes.

The timbres are located consistently within the 5-octave kit keymaps so you can interchange keymaps within percussion programs freely without changing the basic timbres assigned to various notes (snare sounds will always be at and around Middle C, for example). The note assignments for the timbres in the 5-octave kit and 2-octave kit keymaps are listed below. MIDI note number 60 (Middle C) is defined as C 4.

#### 5-Octave Percussion Keymaps (C2 - C7)

MIDI NOTE NUMBER	KEY NUMBER	SAMPLE ROOT
36-37	C2-C#2	Low Tom
38-39	D2-D#2	Low Mid Tom
40-41	E2-F2	Mid Tom
42-43	F#2-G2	Hi MidTom
44-45	G#2-A2	Mid Hi Tom
46	A #2	Hi Tom
47-51	B 2-D# 3	Kick
52-54	E3-F#3	Snare (Sidestick)

## Note Numbers and Intonation Tables

### Note Numbers for Percussion Keymaps

55-56	G3-G#3	Low Snare (dual vel. on Dry Kit 1)
57-59	A3-B3	Mid Snare (dual vel. on Dry Kit 1)
60-61	C4-C#4	Hi Snare (dual vel. on Dry Kit 1)
62-64	D 4-E 4	Closed HiHat
65-67	F 4-G 4	Slightly Open HiHat
68-69	G# 4-A 4	Open HiHat
70-71	A# 4-B 4	Open to Closed HiHat
72	C 5	Foot-closed HiHat
73-74	C#5-D5	Low Crash Cymbal
75-78	D#5-F#5	Pitched Crash Cymbals
79	G5	Splash Cymbal
80	G#5	Ride Cymbal (Rim)
81-82	A5-A#5	Ride Cymbal (Rim and Bell)
83-84	B5-C6	Ride Cymbal (Bell)
85	C# 6	Cowbell
86	D 6	Handclap
87	D# 6	Timbale
88	E 6	Timbale Shell
89	F 6	Conga Tone
90	F#6	Conga Bass Hi
91	G 6	Conga Slap
92	G#6	Conga Bass Low
93	A 6	Clave
94	A# 6	Cabasa
95-96	B 6-C 7	Tambourine Shake

## 2-Octave Percussion Keymaps (C3 - C5)

MIDI NOTE NUMBER	KEY NUMBER	SAMPLE ROOT
48-49	C 3-C# 3	Kick
50	D 3	Low Tom
51	D# 3	Cowbell
52	E 3	Low Tom
53	F 3	Mid Tom
54	F# 3	Cowbell
55	G 3	Mid Tom
56	G# 3	Timbale
57	A 3	High Tom
58	A# 3	Snare (Sidestick)
59	B 3	High Tom
60-61	C4-C#4	Snare (dual velocity)
62	D 4	Closed HiHat
63	D#4	Ride Cymbal (Rim and Bell)
64	E 4	Closed HiHat
65	F 4	Slightly Open HiHat
66	F# 4	Crash Cymbal
67	G 4	Slightly Open HiHat
68	G# 4	Crash Cymbal
69	A 4	Open HiHat
70	A# 4	Crash Cymbal
71	B 4	Open to Closed HiHat
72	C 5	Foot-closed HiHat

## List and Description of Intonation Tables

- |    |                         |  |
|----|-------------------------|--|
| 1  | Equal                   | No detuning of any intervals. The standard for modern western music.   |
| 2  | Classic Just            | Tunings are defined based on the ratios of the frequencies between intervals. The original tuning of Classical European music.   |
| 3  | Just 7 <sup>th</sup>    | Similar to classic Just, but with the Dominant 7 <sup>th</sup> flatted an additional 15 cents.   |
| 4  | Harmonic                | The perfect 4 <sup>th</sup> , Tritone, and Dominant 7 <sup>th</sup> are heavily flatted.   |
| 5  | Just Harmonic           |  |
| 6  | Werkmeister             | Named for its inventor, Andreas Werkmeister. It's fairly close to equal temperament, and was developed to enable transposition with less dissonance.                       |
| 7  | 1/5 <sup>th</sup> Comma |  |
| 8  | 1/4 <sup>th</sup> Comma |  |
| 9  | Indian Raga             | Based on the tunings for traditional Indian music.   |
| 10 | Arabic                  | Oriented toward the tunings of Mid-Eastern music.  |
| 11 | 1Bali/Java              | Based on the pentatonic scale of Balinese and Javanese music.  |
| 12 | 2Bali/Java              | A variation on 1Bali/Java, slightly more subtle overall.   |
| 13 | 3Bali/Java              | A more extreme variation.  |
| 14 | Tibetan                 | Based on the Chinese pentatonic scale.   |
| 15 | CarlosAlpha             | Developed by Wendy Carlos, an innovator in microtonal tunings, this intonation table flats each interval increasingly, resulting in an octave with quarter-tone intervals. |
| 16 | Pyth/aug4               | This is a Pythagorean tuning, based on the Greek pentatonic scale. The tritone is 12 cents sharp.  |
| 17 | Pyth/dim5               | This is a Pythagorean tuning, based on the Greek pentatonic scale. The tritone is 12 cents flat.   |

In general, you should select a non-standard intonation table when you're playing simple melodies (as opposed to chords) in a particular musical style. When you use intonation tables based on pentatonic scales, you'll normally play pentatonic scales to most accurately reproduce those styles. An excellent reference source for further study of alternative tunings is *Tuning In: Microtonality in Electronic Music*, by Scott R. Wilkinson.

## Note Numbers and Intonation Tables

---

List and Description of Intonation Tables

# Chapter 25

## Control Sources

### Control Sources

Control sources are assigned as values for control source parameters, like Src1 and Src2, Depth Control for Src2, and LFO rate control. Assigning a control source to one of these parameters is like connecting control source outputs to various inputs on early modular synthesizers. You can think of each control source parameter as the input to a synthesizer module, and the values for those parameters as the outputs of modules generating control signals.

For the control sources to have an effect, two things have to happen. First, the control source must be assigned as the value for (patched to) a control source parameter like Src1. In other words, for a control source parameter to have an effect, it must be programmed to respond to a particular control message. Second, the control source must generate a signal. The level of the control source's signal determines how much effect it has on the control source parameter to which it's assigned.

In terms of generating signals, there are two types of control sources. The first, which might be called hardware control sources, require some physical movement to transmit them. The control source called MWheel (MIDI 01) is probably the most prominent example of this type of control source. When you move your MIDI controller's Mod Wheel, it sends a Modulation message (MIDI 01), unless you've programmed it to send something else. By default, when the K2vx receives a MIDI 01 message, it responds by sending a control signal to whatever control source is assigned as the value for the ModWhl parameter on the MIDI mode RECV page. Of course, you can program the Mod Whl parameter to send any available control source signal in response to MIDI 01 messages.

Some of these hardware control sources have physical controls "hard-wired" to transmit them. That is, there are certain physical controls that *always* generate these control signals. Every time you strike one of your MIDI controller's keys (or pluck a string, or whatever), for example, a Note on message is generated, along with an Attack velocity message. So any time you strike a key, any control source parameter that has AttVel assigned as its value will be affected by the Attack velocity message. Similarly, every time you move the physical Pitch Wheel, a PWheel message is generated. Whether this affects anything depends on whether you have assigned any control source parameters to respond to the PWheel message (in other words, whether any control source parameter has PWheel assigned as its value).

On the MIDI XMIT page (and in the Setup Editor) you'll find six parameters that correspond to the standard physical controls found on many keyboard controllers: Mod Wheel, Foot Switches 1 and 2, the Control Pedal (CPedal), the Controller Slider (Slider), and mono pressure (Press). As long as the LocalKbdCh parameter on the RECV page in MIDI mode matches the transmit channel of your MIDI controller, these parameters will always respond to specific MIDI control messages: ModWhl always responds to Modulation messages (MIDI 01); FtSw1 always responds to Sustain (MIDI 64); FtSw2 always responds to Sostenuito (MIDI 66); CPedal always responds to Foot (MIDI 04); Slider always responds to Data (MIDI 06); Press always responds to mono pressure.

The values you assign for these six parameters determine which control messages will be transmitted to the K2vx and to its MIDI Out port when you move the corresponding controls on your MIDI controller. If you look at the MIDI XMIT page, you'll see that the parameter called ModWhl has a default value of MWheel. You can interpret this as follows: "Moving the Mod Wheel on my MIDI controller sends the MWheel (Modulation, MIDI 01) message to the

K2vx's sound engine, and, if the K2vx's LocalKbdCh parameter matches my controller's transmit channel, to the K2vx's MIDI Out port."

If you change the value of the ModWhl parameter, the Mod Wheel will no longer send the MWheel message, and any control source parameter with MWheel assigned as its value will no longer respond to movement of the Mod Wheel. All of the control assignment parameters on the MIDI mode XMIT page (and in the Setup Editor) can be programmed to send any of the MIDI controller numbers. For example, if you assign Foot (MIDI 04) as the value for the Press parameter, then generating mono pressure messages from your MIDI controller will send a Foot (MIDI 04) message to the K2vx's sound engine, and will affect any control source parameter that has Foot assigned as its value. If the value for the K2vx's LocalKbdCh parameter matches your MIDI controller's transmit channel, then in this case the Foot message will be sent to the K2vx's MIDI Out port as well, when you generate mono pressure messages from your MIDI controller.

The other type of control source is independent of the movement of physical controls. These control sources generate their control signals internally, and might be called software control sources. They either run automatically (like A Clock and RandV1), or they're programmed to generate their signals according to parameters of their own (as with the LFOs and FUNs). The software control sources must have some non-zero value set for one or more of their parameters before they'll generate control signals.

To summarize, there are two different cases in which you'll assign control sources. One, the transmit case, determines what control message will be sent by a particular physical control. For example, MWheel is set by default to be transmitted by the Mod Wheel. The other case, the receive case, determines which control message will activate a particular control source parameter. For example, if you assign MPress as the value for the Src1 parameter on the PITCH page in the Program Editor, then that layer's pitch will be affected whenever an MPress message is generated by any physical control.

### Control Source Lists

There's one long list of control sources stored in the K2vx's memory, although not all control sources are available for all control source parameters. With time you'll become familiar with the types of control sources available for various control source parameters.

The available list of control sources varies depending on the type of control source parameter you're programming. There are four basic types: MIDI control sources, local control sources, global control sources, and FUNs.

When you're setting the control assignment parameters on the MIDI mode XMIT page or in the Setup Editor, you'll see only the portion of the Control Source list that has values appropriate to MIDI controller messages. Consequently we refer to this subset of the Main Control Source list as the MIDI Control Source list.

You'll see variations on the Main Control Source list as you program the other control source parameters. We'll explain these variations, but it's not important that you memorize each variation. The lists differ to prevent you from assigning a control source where it would be ineffective. All you have to do is scroll through the list of control sources available for any given control source parameter, and choose from the available values.

If you're programming one of the FUNs, you'll see the Main Control Source list, which includes almost every control source from the MIDI Control Source list (with the exception of Data Inc, Data Dec, and Panic, which belong exclusively to the MIDI Control Source list). The list for the FUNs also includes a set of constant values, that set an unvarying control signal level for one or both of the FUN's inputs.

For most other control source parameters, you'll see the Main Control Source list (without the FUN constants and the three special MIDI control sources we mentioned above). There are two



exceptions to this rule, which have to do with global control source parameters. Globals affect every note in each program's layer(s). Consequently they can't use local control sources as their values, since local control sources affect each note independently.

Four of the control source parameters are always global: the Enable parameter on the LAYER page (Program Editor), and the three control source parameters on the EFFECT page, (Program Editor). When programming these parameters, you'll see the Main Control Source list minus the three special MIDI control sources, minus the following local control sources:

Note St  
 Key St  
 KeyNum  
 BKeyNum  
 AttVel  
 InvAVel  
 PPress  
 BPPress  
 RelVel  
 Bi-AVel  
 VTRIG1  
 VTRIG2  
 RandV1  
 RandV2  
 ASR1  
 LFO1  
 FUN1  
 FUN3  
 Loop St  
 PB Rate  
 AtkSt  
 Rel St

Finally, if you've turned on the Globals parameter on the COMMON page in the Program Editor, the available values for GLFO2, and the values for GASR2's trigger will lack the local control sources listed above, as well as the three special MIDI control sources and the FUN constants. The available values for GFUN2 and GFUN4 will exclude the same list of local control sources, but will include the FUN constants.

## Descriptions of Control sources

This section is organized into two sets of descriptions: the MIDI Control Source list, and the rest of the control sources. The numeral preceding the name of each control source can be entered on the alphanumeric pad to select the control source directly (press ENTER after typing the numeral).

Many of the MIDI control sources are assigned as default values for the control assignment parameters on the MIDI mode XMIT page and the Setup Editor page. We'll indicate these assignments as they appear, simply by mentioning that they're the default control source for a control assignment parameter.

### MIDI Control Source List

With a few exceptions, the MIDI control sources correspond to the standard MIDI controller numbers used by every MIDI device.

- 128 OFF**  
This value eliminates the effect of any control source parameter to which it's assigned.
- 0, 33 MONO PRESSURE (MPress)**  
Many of the K2vx's factory programs are assigned to modify parameters such as pitch, filter cutoff frequency, and depth control when MPress messages are received. The mono pressure (Press) control assignment parameters in MIDI and Setup modes are set by default to transmit MPress messages when mono pressure messages are received from a controller.
- 1 MIDI 01 (MWheel)**  
Many factory programs are assigned to respond to MWheel messages. The Mod Whl parameter in MIDI and Setup modes is set by default to transmit MWheel.
- 2 MIDI 02 (Breath)**
- 3 MIDI 03**
- 4 MIDI 04 (Foot)**  
This is the standard MIDI controller number for continuous control foot pedals. It's the default value for the CPedal control assignment parameter, so a control pedal on your MIDI controller which sends MIDI controller 04 messages will send MIDI controller 04 messages to the K2vx by default.
- 5 MIDI 05 (PortTim)**  
This is the standard MIDI controller number for portamento time control. The K2vx always responds to this control message. For any program that has portamento turned on (on the COMMON page in the Program Editor), MIDI Portamento Time messages received via MIDI will affect the rate of the program's portamento.
- 6 MIDI 06 (Data)**  
MIDI 06 is the standard MIDI controller number for data entry. The Slider parameter on the MIDI mode XMIT page and in the Setup Editor is set by default to transmit this message, and can be used to select programs and edit parameters on MIDI slaves if your controller can send it.
- 7 MIDI 07 (Volume)**  
This is the standard MIDI controller number for volume. The Volume parameter on the CHANLS page in MIDI mode will respond to MIDI controller 07 unless the VolLock parameter is turned on.
- 8 MIDI 08 (Balance)**
- 9 MIDI 09**
- 10 MIDI 10 (Pan)**  
MIDI controller 10 is defined as Pan control. The Pan parameter on the CHANLS page in MIDI mode will respond to MIDI controller 10 unless the PanLock parameter is turned on.

**11        MIDI 11 (Express)****12—15    MIDI 12—15****16—19    MIDI 16—19 (Ctl A—D)****20—31    MIDI 20—31****64        MIDI 64 (Sustain)**

This is the standard MIDI controller number for Sustain. The control assignment parameter FootSw1 is set by default to MIDI controller 64, so a switch pedal on your MIDI controller which sends MIDI 64 will send sustain messages to the K2vx by default. The K2vx will always respond to sustain messages by sustaining currently active notes.

**65        MIDI 65 (PortSw)**

This is the standard MIDI controller number for Portamento Switch. The Portamento parameter on the COMMON page in the Program Editor always responds to this controller, and will turn Portamento on for monophonic programs when the controller signal is at 64 or above. It won't affect polyphonic programs.

**66        MIDI 66 (SostPD)**

MIDI controller 66 is defined as Sostenuto Switch. The control assignment parameter FootSw2 is set by default to MIDI controller 66, so a switch pedal on your MIDI controller which sends MIDI 66 will send sostenuto messages to the K2vx by default. The K2vx will always respond to sostenuto messages.

**67        MIDI 67 (SoftPd)**

This is the standard MIDI controller number for Soft Pedal. The K2vx will always respond to Soft pedal messages.

**68        MIDI 68****69        MIDI 69 (FrezPd)**

The K2vx will always respond to this message. It causes all notes to be frozen at their current amplitude levels while the function is on.

**70—74    MIDI 70—74****75        MIDI 75 (LegatoSw)**

The K2vx always responds to this message. When a MIDI controller 75 message with a value above 64 is received, the K2vx will force polyphonic programs to be monophonic.

**76—79      MIDI 76—79**

**80—83      MIDI 80—83 (Ctl E—H)**

**84—90      MIDI 84—90**

**91            MIDI 91 (FXDep)**

The MIDI specification defines this controller as External Effects Depth. If the FX Mode parameter is set to Master, and the FX Channel parameter is set to a specific MIDI channel, the K2vx will respond to this message when it is received on the FX channel. It responds by adjusting the Wet/Dry mix of the current preset effect.

**92—95      MIDI 92—95**

**96            MIDI 96 (DataInc)**

This is defined as Data Increment. It's intended to be assigned to a switch control. When the control is on (value 127), the currently selected parameter's value will be increased by one increment. This could be assigned to FootSw2, for example, to scroll through the program list while in Program mode.

**97            MIDI 97 (DataDec)**

This is defined as Data Decrement. It's intended to be assigned to a switch control. When the control is on (value 127), the currently selected parameter's value will be decreased by one increment.

**123          MIDI 123 (Panic)**

The K2vx always responds to this message by sending an All Notes Off and All Controllers Off message on all 16 MIDI channels.

## Main Control Source List

This list contains all but the last three control sources in the MIDI Control Source list. It also contains the following control sources. All are local unless specified as global.

**32            Channel State (Chan St)**

Chan St refers to whether any notes are currently active on a given MIDI channel. Chan St switches on whenever a note is started, and switches off when a Note Off has been received for each current note on that channel, even if notes are sustained.

**33            Mono pressure (MPress)**

This is the same as the MPress control source in the MIDI Control Source list, but is assigned by entering 33 on the alphanumeric pad when used with a parameter that takes its values from the Main Control Source list.

**34            Bipolar mono pressure (BMPress)**

This control source generates a control signal of -1 when the value of the control to which it's assigned is at its minimum, and +1 when the control is at its maximum. For example, if you had the MPress control assignment parameter assigned to send BMPress, and you had Src1 on a program layer's PITCH page assigned to BM-Press, with its depth parameter set to 1200 cents, then the layer would be transposed down an octave when no pressure (value 0) was applied to your controller's keys (assuming it sends mono pressure). Maximum pressure (value 127) would

transpose the layer up an octave, while a pressure level of 64 would leave the pitch unchanged.

**35 Pitch Wheel message (PWheel)**

The K2vx is hard-wired to respond to this message. Any parameter with PWheel assigned as its value will be affected when your MIDI controller's Pitch Wheel is moved.

**36 Bipolar Mod Wheel (Bi-Mwl)**

This control source will always respond to MIDI controller 01 (MWheel). Control source parameters set to this value will generate control signals of -1 when the MIDI controller 01 message value is 0, and will generate a control signal of +1 when the MIDI controller 01 message is at 127, scaling all values in between. For example, you might set Src1 on a program layer's PITCH page to a value of Bi-Mwl, and its depth parameter to 1200 cents. Then as long as the ModWhl control assignment parameter is set to a value of MWheel, your controller's Mod Wheel will be bipolar; in this case it will bend the layer's pitch down as you move the Mod Wheel toward minimum, and bend the pitch up as you move the Mod Wheel toward maximum.

**37 Absolute value of Pitch Wheel (AbsPwl)**

This control source always responds to movement of your MIDI controller's Pitch Wheel, but makes the Pitch Wheel unipolar. Whereas pulling the Pitch Wheel fully down usually generates a control signal value of -1, this control source generates a value of +1 when the Pitch Wheel is pulled fully down.

**38 Global ASR (GASR2)**

When the Globals parameter on the COMMON page is turned on, ASR2 becomes global, and is labeled GASR2. The functions of ASRs are explained in Chapter 6 in the *Performance Guide*, in the section "The ASR Page." This control source does not appear in the Control Source list for parameters whose functions are local.

**39 Global FUN2 (GFUN2)**

When the Globals parameter on the COMMON page is turned on, FUN2 becomes global, and is labeled GFUN2. The functions of FUNs are explained in Chapter 16 in the *Performance Guide*. This control source does not appear in the Control Source list for parameters whose functions are local.

**40 Global LFO (GLFO2)**

When the Globals parameter on the COMMON page is turned on, LFO2 becomes global, and is labeled GLFO2. The functions of LFOs are explained in Chapter 6 in the *Performance Guide*, in the section "THE LFO PAGE." This control source does not appear in the Control Source list for parameters whose functions are local.

**41 Global LFO Phase (GLFO2ph)**

When the Globals parameter on the COMMON page is turned on, LFO2 becomes global, and is labeled GLFO2. The functions of LFOs are explained in Chapter 6 in the *Performance Guide*, in the section "THE LFO PAGE." This control source does not appear in the Control Source list for parameters whose functions are local.

**42 Global FUN 4 (GFUN4)**

When the Globals parameter on the COMMON page is turned on, FUN 4 becomes global, and is labeled GFUN4. This control source does not appear in the Control Source list for parameters whose functions are local.

**43 Volume Control (VolCtl)**

This control source will always respond to MIDI controller 07 messages. Assign this value to a parameter when you want MIDI volume messages to affect the parameter.

**44 Pan Control (PanCtl)**

This control source always responds to MIDI controller 10 messages. Assign this value to a parameter when you want MIDI pan messages to affect the parameter.

**45 Balance Control (BalCtl)**

This control source will always respond to MIDI controller 08 messages. Assign this value to a parameter when you want MIDI balance messages to affect the parameter.

**46 Channel Count (ChanCnt)**

This control source keeps track of the total number of active voice channels (how many notes are playing), and converts the number into a control signal between 0 and +1. The control signal's value is 1 when all 48 voice channels are active, and 0 when no voice channels are active.

You can use this control source in several ways. One example is to limit the volume of each note so that you have a more nearly constant volume regardless of how many notes you're playing (this is independent of the effect of attack velocity on volume). To set this up, you would go to the F4 AMP page in the Program Editor, and set the Src1 parameter to a value of ChanCnt. Then set the Depth parameter to a negative value. This will decrease the overall amplitude of each note as you play more simultaneous notes. This example works best with short-release sounds. It's great for an organ program, for example.

Channel count is also useful for controlling the modulation applied to a sound. For example, you may have a sound what you use both as a lead and for rhythm. Suppose you want a deep vibrato when you're soloing, but less vibrato when you're playing chords. Set up the vibrato by using LFO1 as the value for the Src2 parameter on the PITCH page in the Program Editor. Set the MinDpt parameter to 72 cts, and the MaxDpt parameter to 12 cts. Then set the value of the DptCtl parameter to ChanCnt, and You'll get maximum vibrato depth when only one note is active. (Channel count outputs a control signal of 0 when no notes are playing, so with only one note playing, its value is near 0, which causes the DptCtl parameter to generate a value near its minimum: 72 cents in this case.)

If you want to increase the depth of the vibrato as you increase the number of active notes, set the value of the MaxDpt parameter higher than that of the MinDpt parameter.

NOTE: There are no control sources that correspond to the numeric entries 47—54.

- 55      Sync State (SyncSt)**  
This unipolar control source responds to MIDI clock messages received from an external MIDI device. Sync State switches on (+1) at each clock start, and switches off (0) with each clock stop.
- 56      A Clock**  
This is a unipolar square wave that responds to MIDI clock messages. It switches to +1 and back to 0 with every clock beat. This control source looks first for externally received MIDI clock messages, and if none is received, it responds to the K2vx's internal clock, which is always running. The internal clock speed is set with the Tempo parameter in Song mode.
- 57      Negative A Clock (~A Clock)**  
This is the opposite of A clock, that is, it switches from 0 to +1 with every clock beat (the square wave is 180 degrees out of phase with that of A Clock).
- 58      B Clock**  
This is similar to A Clock, but it's bipolar—it switches from +1 to -1 with every clock beat.
- 59      Negative B Clock (~B Clock)**  
The opposite of B Clock, this bipolar control source switches from -1 to +1 with every clock beat (the square wave is 180 degrees out of phase with that of B Clock).
- 60, 61      Global Phase 1 and 2 (G Phase 1, G Phase 2)**  
These bipolar global control sources are both rising sawtooth waves that rise from -1 to +1 with each MIDI clock beat. Like A Clock and B clock, they look for an external clock signal, and if none is received, they respond to the K2vx's internal clock.
- 62, 63      Global Random Variant 1 and 2 (GRandV 1, GRandV 2)**  
These are also bipolar and global, and generate random control signal values between -1 and +1 when assigned to a control source parameter. There is a subtle difference in the randomness of the signals they generate, therefore choosing between them is a matter of preference.
- 96      Note State (Note St)**  
At any moment, any given note is either on or off; this is its Note State. Note State can be used as a unipolar control source that responds to each note that's played. It switches to +1 when the note starts, and stays on as long as the note is held on (by the sustain pedal, for example), or by holding down the trigger for that note. It switches to 0 when the note is no longer sustained by any means. For example, if you play a note, then hold it with the sustain pedal, its Note State is still on (+1) even if you've released the key that triggered the note. As soon as you release the sustain pedal, the note's Note State switches to off (0), even if it has a long release and you can still hear the release section of the note.
- 97      Key State (Key St)**  
This is a unipolar control source that responds to the motion of your MIDI controller's keys. It switches to +1 when a key is pressed, and switches to 0 when the key is released. Its effect differs from Note State in that when the key that switched it

on is released, it will switch off even if the note is sustained. If you're using a non-keyboard MIDI controller, Key State will switch to 0 when the equivalent of a key release is sent.

**98 Key Number (KeyNum)**

This is a unipolar control source that generates its signal value based on the MIDI key number of each note triggered. That is, it generates a value of 0 in response to MIDI key number 0, a value of 64 in response to MIDI key number 64, and so on.

**99 Bipolar Key Number (BKeyNum)**

This is like KeyNum, but generates a signal value of -1 in response to MIDI key number 0, a value of 0 in response to MIDI key number 64, and a value of +1 in response to MIDI key number 127.

**100 Attack Velocity (AttVel)**

This unipolar control source responds to Attack velocity values received at the K2vx's MIDI In port. Velocity values of 0 cause it to generate a signal value of 0, while velocity values of 127 will generate a value of +1. All other velocity values will result in signal values proportionally scaled between 0 and +1.

**101 Inverse Attack Velocity (InvAttVel)**

This is the opposite of AttVel, generating a signal value of 0 in response to attack velocity values of 127.

**102 Polyphonic pressure (PPress)**

This unipolar control source responds to poly pressure (aftertouch) messages received via MIDI. It generates a signal value scaled from 0 to +1 based on the poly pressure value range of 0—127.

**103 Bipolar polyphonic pressure (BPPress)**

This is like PPress, but scales its signal value from -1 to +1.

**104 Release Velocity (RelVel)**

Also unipolar, this control source scales its signal value from 0 to +1 in response to release velocity values from 0—127.

**105 Bipolar Attack Velocity (Bi-AVel)**

This is similar to AttVel, but scales its signal values from -1 to +1.

**106, 107 Velocity Triggers 1 and 2 (VTRIG1, VTRIG2)**

These unipolar control sources are switch controls, that is, they generate signal values of either 0 or +1. These must be programmed in order to have an effect; their programming parameters are found on the VTRIG page in the Program Editor. When a VTRIG's Sense parameter is set to normal, it switches to +1 when a note plays at a dynamic level exceeding the dynamic level set for its Level parameter. See "THE VTRIG PAGE" in Chapter 6 in the *Performance Guide* for more information.



- 108, 109 Random Variants 1 and 2 (RandV1, RandV2)**  
These are similar to GRandV1 and GRandV2, but are local, so will affect each control source parameter independently.
- 110, 111 ASR1, ASR2**  
These are programmable envelopes with three segments, Attack, Sustain, and Release. Their control source signals are unipolar. See "The ASR Page" in Chapter 6 in the *Performance Guide* for a thorough explanation.
- 112, 113 FUN1, FUN2**  
These generate their control source signals by combining the control signal values of two programmable inputs, and performing a mathematical function on the result. Their control signals can be unipolar or bipolar, depending on the control sources assigned as their inputs. See "The FUN Page" in Chapter 6 in the *Performance Guide*. FUN2 becomes global (GFUN2) when the Globals parameter on the COMMON page in the Program Editor is set to On.
- 114 LFO1**  
LFO1 can be unipolar or bipolar depending on the value set for the Shape parameter on its programming page. See "The LFO Page" in Chapter 6 in the *Performance Guide*.
- 115 LFO1 Phase (LFO1ph)**  
This bipolar control source generates its signal based on the cycle of LFO1. When the phase of LFO1 is 0 degrees, the signal value of LFO1ph is 0. When the phase of LFO1 is 90 degrees, the signal value of LFO1ph is 1. When the phase of LFO1 is 180 degrees, the signal value of LFO1ph is 0. When the phase of LFO1 is 270 degrees, the signal value of LFO1ph is -1.
- 116 LFO2**  
This functions exactly the same as LFO1, when the Globals parameter is set to Off (on the COMMON page in the Program Editor). When the Globals parameter is set to On, LFO2 becomes global (GLFO2).
- 117 LFO2 Phase (LFO2ph)**  
This functions exactly the same as LFO1ph, responding to the cycle of LFO2.
- 118, 119 FUN3, FUN4**  
These function exactly the same as FUNs 1 and 2, when the Globals parameter is set to Off (on the COMMON page in the Program Editor). When the Globals parameter is set to On, FUN4 becomes global (GFUN4).
- 120 Amplitude Envelope (AMPENV)**  
This programmable unipolar control source lets you vary the effect of a control source parameter over time. See "The AMPENV Page" in Chapter 6 in the *Performance Guide*.
- 121, 122 Envelopes 2 and 3 (ENV2, ENV3)**  
These are programmed in the same way as AMPENV, but they can be bipolar.

**123      Loop State (Loop St)**

This unipolar control source switches to +1 when the currently playing sample reaches its LoopStart point. If you've programmed a sound with a User amplitude envelope, Loop St will always be on (+1) for that sound. See Chapter 15 in the *Performance Guide* for more about sample loops.

**124      Sample Playback Rate (PB Rate)**

The signal value of this bipolar control source is determined by the sample playback rate of each note. The playback rate is a function of the amount of transposition applied to a sample root to play it at the proper pitch for each note. If you trigger a note where a sample root is assigned, the PB Rate signal value for that note is 0. If the note is above the sample root, the sample is transposed upward, and its playback rate is higher than that of the sample root. Consequently the PB Rate signal value for that note will be positive. If the note is below the sample root, the PB Rate signal value will be negative.

**125      Attack State (Atk State)**

This unipolar control source switches to +1 and back to 0 very quickly with each note start.

**126      Release State (Rel State)**

This unipolar control source switches to +1 when a note is released, and stays on until the note has completed its release (faded to silence), then it switches to 0. It will stay on if a note is sustained, even if its trigger (key, string, whatever) is released.

**127      ON**

This generates a constant control signal value of +1.

**128      -ON**

This generates a constant control signal value of -1 (the numeric entry 128 selects a value of OFF in the MIDI Control Source list).

**129      GKeyNum**

Uses the key number (global) to modify whatever it is patched into. Higher notes will have a very different effect than will lower notes. Users can use this new Source to control any K2vx parameters such as F/X depth, or to scale amplitude or pitch.

**130      GAttVel**

This is updated every time you strike another key (kind of a multi-trigger function). Users can patch this new Source to control parameters such as F/X Depth.

In addition to enabling (triggering) layers from any controller (works like an on/off switch), users may now set the assigned controller's threshold (value, or range of values from 0-127), thus defining the controller's active range where it will enable the layer.

For example, you could create a 32 layer nylon guitar where each layer is assigned to a different V.A.S.T. Algorithm and each layer is enabled by discrete narrow velocity ranges. This would produce 32 different sounding layers with 32 cross switch points emulating a picked guitar where no two attacks are exactly alike. If

the layers' velocity ranges were very close together yet not overlapping, you could create very subtle non-repeating changes. This kind of power usually eludes most sample playback devices, as this technique uses only one layer of polyphony, due to cross switching versus cross fading.

### 131, 132 **GHiKey, GLoKey**

These control sources work the same as GKeyNum except that they track the highest key currently held and the lowest key currently held respectively. By using one of these as the only source for pitch tracking, you can create "mono-like" layers within a polyphonic program.

The remaining control sources are constants, which appear only when you're assigning control sources as inputs for the FUNs. Assigning one of these values fixes the input's control signal value at a steady level.

133	-0.99
134	-0.98
135	-0.97
136-140	-0.96 to -0.92
141	-0.91
142	-0.90
143-145	-0.88 to -0.84
146-150	-0.82 to -0.74
151-155	-0.72 to -0.64
156-160	-0.62 to -0.54
161-165	-0.52 to -0.44
166-170	-0.42 to -0.34
171-175	-0.32 to -0.24
176-180	-0.22 to -0.14
181	-0.12
182	-0.10
183	-0.09
184	-0.08
185	-0.07
186-190	-0.06 to -0.02
191	-0.01
192	0.00
193	0.01
194	0.02
195	0.03
196-200	0.04 to 0.08
201	0.09
202	0.10
203	0.12
204	0.14
205	0.16
206-210	0.18 to 0.26
211-215	0.28 to 0.36
216-220	0.38 to 0.46
221-225	0.48 to 0.56

## Control Sources

### Descriptions of Control sources

---

226-230	0.58 to 0.66
231-235	0.68 to 0.76
236-240	0.78 to 0.86
241	0.88
242	0.90
243	0.91
244	0.92
245	0.93
246-250	0.94 to 0.98
251	0.99

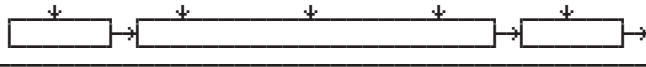
Note: There are no control sources that correspond to numeric entries 252—254.

256	OFF
-----	-----

# Chapter 26

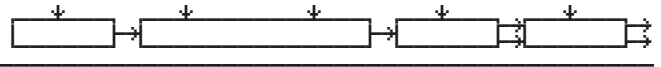
## DSP Algorithms

Algorithm 1



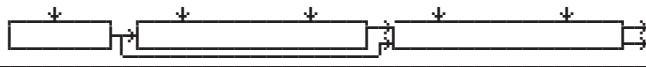
- |       |                     |     |
|-------|---------------------|-----|
| PITCH | HIFREQ STIMULATOR   | AMP |
|       | PARAMETRIC EQ       |     |
|       | STEEP RESONANT BASS |     |
|       | 4POLE LOPASS W/SEP  |     |
|       | 4POLE HIPASS W/SEP  |     |
|       | TWIN PEAKS BANDPASS |     |
|       | DOUBLE NOTCH W/SEP  |     |
|       | NONE                |     |

Algorithm 2



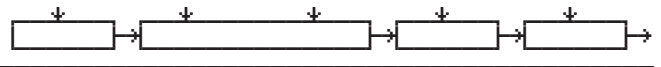
- |       |               |        |     |
|-------|---------------|--------|-----|
| PITCH | 2PARAM SHAPER | PANNER | AMP |
|       | 2POLE LOWPASS |        |     |
|       | BANDPASS FILT |        |     |
|       | NOTCH FILTER  |        |     |
|       | 2POLE ALLPASS |        |     |
|       | PARA BASS     |        |     |
|       | PARA TREBLE   |        |     |
|       | PARA MID      |        |     |
|       | NONE          |        |     |

Algorithm 3



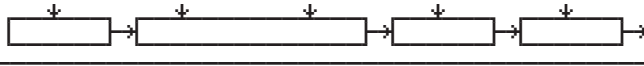
- |       |               |       |       |
|-------|---------------|-------|-------|
| PITCH | 2PARAM SHAPER | AMP U | AMP L |
|       | 2POLE LOWPASS | BAL   | AMP   |
|       | BANDPASS FILT |       |       |
|       | NOTCH FILTER  |       |       |
|       | 2POLE ALLPASS |       |       |
|       | NONE          |       |       |

Algorithm 4



- |       |               |        |     |
|-------|---------------|--------|-----|
| PITCH | 2PARAM SHAPER | LPCLIP | AMP |
|       | 2POLE LOWPASS | SINE+  |     |
|       | BANDPASS FILT | NOISE+ |     |
|       | NOTCH FILTER  | LOPASS |     |
|       | 2POLE ALLPASS | HIPASS |     |
|       | PARA BASS     | ALPASS |     |
|       | PARA TREBLE   | GAIN   |     |
|       | PARA MID      | SHAPER |     |
|       | NONE          | DIST   |     |
|       |               | SW+SHP |     |
|       |               | SAW+   |     |
|       |               | SW+DST |     |
|       |               | NONE   |     |

Algorithm 5



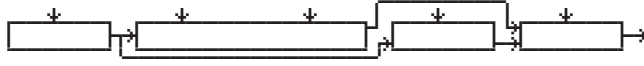
PITCH	2PARAM SHAPER	LP2RES	AMP
	2POLE LOWPASS	SHAPE2	
	BANDPASS FILT	BAND2	
	NOTCH FILTER	NOTCH2	
	2POLE ALLPASS	LOPAS2	
	PARA BASS	HIPAS2	
	PARA TREBLE	LPGATE	
	PARA MID	NONE	
	NONE		

Algorithm 6



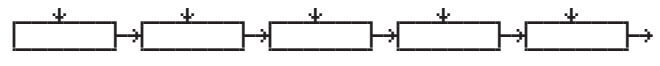
PITCH	2PARAM SHAPER	LPCLIP	x AMP
	2POLE LOWPASS	SINE+	+ AMP
	BANDPASS FILT	NOISE+	! AMP
	NOTCH FILTER	LOPASS	
	2POLE ALLPASS	HIPASS	
	NONE	ALPASS	
		GAIN	
		SHAPER	
		DIST	
		SW+SHD	

Algorithm 7



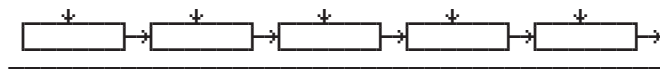
PITCH	2PARAM SHAPER	LPCLIP	x AMP
	2POLE LOWPASS	SINE+	+ AMP
	BANDPASS FILT	NOISE+	! AMP
	NOTCH FILTER	LOPASS	
	2POLE ALLPASS	HIPASS	
	NONE	ALPASS	
		GAIN	
		SHAPER	
		DIST	
		SINE	
		LF SIN	
		SW+SHD	
		SAW+	
		SW+DST	
		NONE	

Algorithm 8



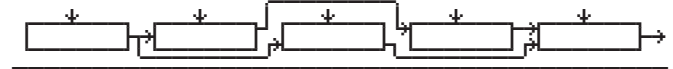
PITCH	LOPASS	LOPASS	LPCLIP	AMP
	HIPASS	HIPASS	SINE+	
	ALPASS	ALPASS	NOISE+	
	GAIN	GAIN	LOPASS	
	SHAPER	SHAPER	HIPASS	
	DIST	DIST	ALPASS	
	PWM	SW+SHD	GAIN	
	SINE	SAW+	SHAPER	
	LF SIN	WRAP	DIST	
	SW+SHD	NONE	SW+SHD	
	SAW+		SAW+	
	SAW		SW+DST	
	LF SAW		NONE	
	SQUARE			
	LF SQR			
	WRAP			
	NONE			

Algorithm 9



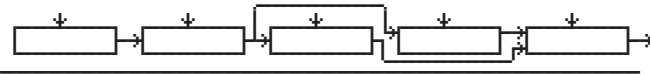
PITCH	LOPASS	LOPASS	LP2RES	AMP
	HIPASS	HIPASS	SHAPE2	
	ALPASS	ALPASS	BAND2	
	GAIN	GAIN	NOTCH2	
	SHAPER	SHAPER	LOPAS2	
	DIST	DIST	HIPAS2	
	PWM	SW+SHP	LPGATE	
	SINE	SAW+	NONE	
	LF SIN	WRAP		
	SW+SHP	NONE		
	SAW+			
	SAW			
	LF SAW			
	SQUARE			
	LF SQR			
	WRAP			
	NONE			

Algorithm 10



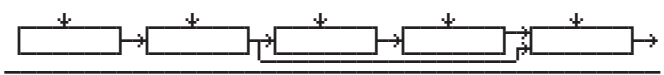
PITCH	LOPASS	LOPASS	LPCLIP	x AMP
	HIPASS	HIPASS	SINE+	+ AMP
	ALPASS	ALPASS	NOISE+	! AMP
	GAIN	GAIN	LOPASS	
	SHAPER	SHAPER	HIPASS	
	DIST	DIST	ALPASS	
	PWM	SINE	GAIN	
	SINE	LF SIN	SHAPER	
	LF SIN	SW+SHP	DIST	
	SW+SHP	SAW+	SW+SHP	
	SAW+	SAW	SAW+	
	SAW	LF SAW	SW+DST	
	LF SAW	SQUARE	NONE	
	SQUARE	LF SQR		
	LF SQR	WRAP		
	WRAP	NONE		
	NONE			

Algorithm 11



PITCH	LOPASS	LOPASS	LPCLIP	x AMP
	HIPASS	HIPASS	SINE+	+ AMP
	ALPASS	ALPASS	NOISE+	! AMP
	GAIN	GAIN	LOPASS	
	SHAPER	SHAPER	HIPASS	
	DIST	DIST	ALPASS	
	PWM	SINE	GAIN	
	SINE	LF SIN	SHAPER	
	LF SIN	SW+SHP	DIST	
	SW+SHP	SAW+	SINE	
	SAW+	SAW	LF SIN	
	SAW	LF SAW	SW+SHP	
	LF SAW	SQUARE	SAW+	
	SQUARE	LF SQR	SW+DST	
	LF SQR	WRAP	NONE	
	WRAP	NONE		
	NONE			

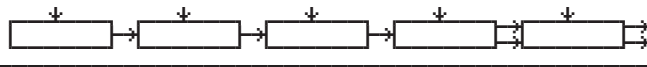
Algorithm 12



PITCH	LOPASS	LOPASS	LPCLIP	x AMP
	HIPASS	HIPASS	SINE+	+ AMP
	ALPASS	ALPASS	NOISE+	! AMP
	GAIN	GAIN	LOPASS	
	SHAPER	SHAPER	HIPASS	
	DIST	DIST	ALPASS	
	PWM	PWM	GAIN	
	SINE	SINE	SHAPER	
	LF SIN	LF SIN	DIST	
	SW+SHP	SW+SHP	SW+SHP	
	SAW+	SAW+	SAW+	
	SAW	SAW	SW+DST	
	LF SAW	LF SAW	NONE	
	SQUARE	SQUARE		
	LF SQR	LF SQR		
	WRAP	WRAP		
	NONE	NONE		

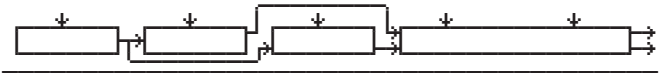


Algorithm 13



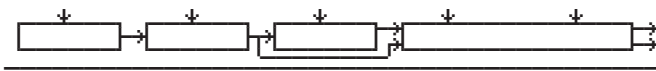
PITCH	LOPASS	LOPASS	PANNER	AMP
	HIPASS	HIPASS		
	ALPASS	ALPASS		
	GAIN	GAIN		
	SHAPER	SHAPER		
	DIST	DIST		
	PWM	SW+SHP		
	SINE	SAW+		
	LF SIN	WRAP		
	SW+SHP	NONE		
	SAW+			
	SAW			
	LF SAW			
	SQUARE			
	LF SQR			
	WRAP			
	NONE			

Algorithm 14



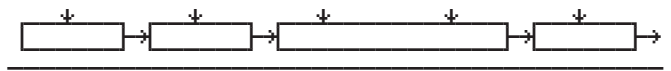
PITCH	LOPASS	LOPASS	AMP U	AMP L
	HIPASS	HIPASS	BAL	AMP
	ALPASS	ALPASS		
	GAIN	GAIN		
	SHAPER	SHAPER		
	DIST	DIST		
	SINE	SINE		
	LF SIN	LF SIN		
	SW+SHP	SW+SHP		
	SAW+	SAW+		
	SAW	SAW		
	LF SAW	LF SAW		
	SQUARE	SQUARE		
	LF SQR	LF SQR		
	WRAP	WRAP		
	NONE	NONE		

Algorithm 15



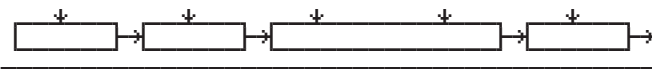
PITCH	LOPASS	LOPASS	AMP U	AMP L
	HIPASS	HIPASS	BAL	AMP
	ALPASS	ALPASS		
	GAIN	GAIN		
	SHAPER	SHAPER		
	DIST	DIST		
	PWM	SINE		
	SINE	LF SIN		
	LF SIN	SW+SHP		
	SW+SHP	SAW+		
	SAW+	SAW		
	SAW	LF SAW		
	LF SAW	SQUARE		
	SQUARE	LF SQR		
	LF SQR	WRAP		
	WRAP	NONE		
	NONE			

Algorithm 16



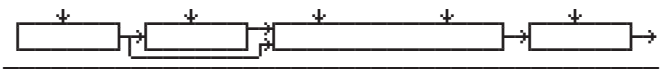
PITCH	LOPASS	PARA BASS	AMP
	HIPASS	PARA TREBLE	
	ALPASS	NONE	
	GAIN		
	SHAPER		
	DIST		
	SINE		
	LF SIN		
	SW+SHP		
	SAW+		
	SAW		
	LF SAW		
	SQUARE		
	LF SQR		
	WRAP		
	NONE		

Algorithm 17



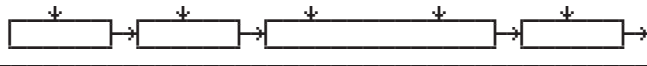
PITCH	LOPASS	SHAPE MOD OSC	AMP
	HIPASS	AMP MOD OSC	
	ALPASS	NONE	
	GAIN		
	SHAPER		
	DIST		
	PWM		
	SINE		
	LF SIN		
	SW+SHP		
	SAW+		
	SAW		
	LF SAW		
	SQUARE		
	LF SQR		
	WRAP		
	NONE		

Algorithm 18



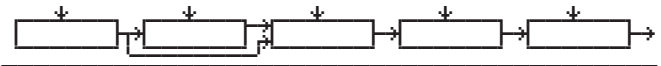
PITCH	LOPASS	x SHAPEMOD OSC	AMP
	HIPASS	+ SHAPEMOD OSC	
	ALPASS	NONE	
	GAIN		
	SHAPER		
	DIST		
	SINE		
	LF SIN		
	SW+SHP		
	SAW+		
	SAW		
	LF SAW		
	SQUARE		
	LF SQR		
	WRAP		
	NONE		

Algorithm 19



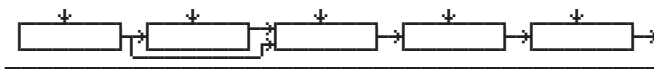
- |       |        |               |     |
|-------|--------|---------------|-----|
| PITCH | LOPAS2 | SHAPE MOD OSC | AMP |
|       | NONE   | NONE          |     |

Algorithm 20



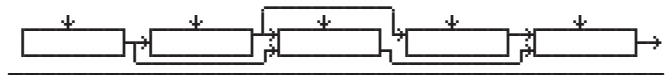
- |       |        |        |        |     |
|-------|--------|--------|--------|-----|
| PITCH | LOPASS | x GAIN | LPCLIP | AMP |
|       | HIPASS | + GAIN | SINE+  |     |
|       | ALPASS | XFADE  | NOISE+ |     |
|       | GAIN   | AMPMOD | LOPASS |     |
|       | SHAPER | NONE   | HIPASS |     |
|       | DIST   |        | ALPASS |     |
|       | SINE   |        | GAIN   |     |
|       | LF SIN |        | SHAPER |     |
|       | SW+SHP |        | DIST   |     |
|       | SAW+   |        | SW+SHP |     |
|       | SAW    |        | SAW+   |     |
|       | LF SAW |        | SW+DST |     |
|       | SQUARE |        | NONE   |     |
|       | LF SQR |        |        |     |
|       | WRAP   |        |        |     |
|       | NONE   |        |        |     |

Algorithm 21



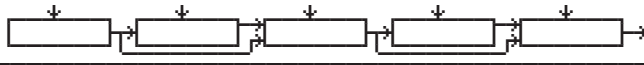
PITCH	LOPASS	x GAIN	LP2RES	AMP
	HIPASS	+ GAIN	SHAPE2	
	ALPASS	XFADE	BAND2	
	GAIN	AMPMOD	NOTCH2	
	SHAPER	NONE	LOPAS2	
	DIST		HIPAS2	
	SINE		LPGATE	
	LF SIN		NONE	
	SW+SHP			
	SAW+			
	SAW			
	LF SAW			
	SQUARE			
	LF SQR			
	WRAP			
	NONE			

Algorithm 22



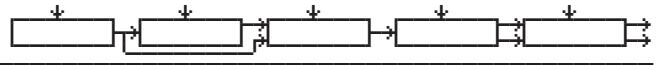
PITCH	LOPASS	x GAIN	LPCLIP	x AMP
	HIPASS	+ GAIN	SINE+	+ AMP
	ALPASS	XFADE	NOISE+	! AMP
	GAIN	AMPMOD	LOPASS	
	SHAPER	NONE	HIPASS	
	DIST		ALPASS	
	SINE		GAIN	
	LF SIN		SHAPER	
	SW+SHP		DIST	
	SAW+		SINE	
	SAW		LF SIN	
	LF SAW		SW+SHP	
	SQUARE		SAW+	
	LF SQR		SW+DST	
	WRAP		NONE	
	NONE			

Algorithm 23



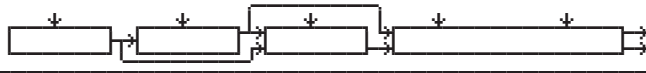
PITCH	LOPASS	x GAIN	LPCLIP	x AMP
	HIPASS	+ GAIN	SINE+	+ AMP
	ALPASS	XFADE	NOISE+	! AMP
	GAIN	AMPMOD	LOPASS	
	SHAPER	NONE	HIPASS	
	DIST		ALPASS	
	SINE		GAIN	
	LF SIN		SHAPER	
	SW+SHP		DIST	
	SAW+		SINE	
	SAW		LF SIN	
	LF SAW		SW+SHP	
	SQUARE		SAW+	
	LF SQR		SW+DST	
	WRAP		NONE	
	NONE			

Algorithm 24



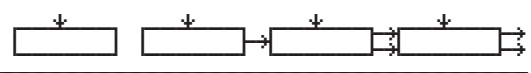
PITCH	LOPASS	x GAIN	PANNER	AMP
	HIPASS	+ GAIN		
	ALPASS	XFADE		
	GAIN	AMPMOD		
	SHAPER	NONE		
	DIST			
	SINE			
	LF SIN			
	SW+SHP			
	SAW+			
	SAW			
	LF SAW			
	SQUARE			
	LF SQR			
	WRAP			
	NONE			

Algorithm 25



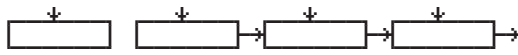
- |       |        |        |       |       |
|-------|--------|--------|-------|-------|
| PITCH | LOPASS | x GAIN | AMP U | AMP L |
|       | HIPASS | + GAIN | BAL   | AMP   |
|       | ALPASS | XFADE  |       |       |
|       | GAIN   | AMPMOD |       |       |
|       | SHAPER | NONE   |       |       |
|       | DIST   |        |       |       |
|       | SINE   |        |       |       |
|       | LF SIN |        |       |       |
|       | SW+SHP |        |       |       |
|       | SAW+   |        |       |       |
|       | SAW    |        |       |       |
|       | LF SAW |        |       |       |
|       | SQUARE |        |       |       |
|       | LF SQR |        |       |       |
|       | WRAP   |        |       |       |
|       | NONE   |        |       |       |

Algorithm 26



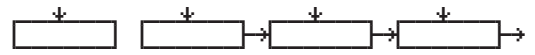
- |        |        |        |     |
|--------|--------|--------|-----|
| SYNC M | SYNC S | PANNER | AMP |
|--------|--------|--------|-----|

Algorithm 27



- 
- |        |        |        |     |
|--------|--------|--------|-----|
| SYNC M | SYNC S | LPCLIP | AMP |
|        |        | SINE+  |     |
|        |        | NOISE+ |     |
|        |        | LOPASS |     |
|        |        | HIPASS |     |
|        |        | ALPASS |     |
|        |        | GAIN   |     |
|        |        | SHAPER |     |
|        |        | DIST   |     |
|        |        | SINE   |     |
|        |        | LF SIN |     |
|        |        | SW+SHP |     |
|        |        | SAW+   |     |
|        |        | SW+DST |     |
|        |        | NONE   |     |

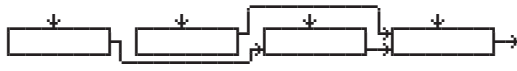
Algorithm 28



- 
- |        |        |        |     |
|--------|--------|--------|-----|
| SYNC M | SYNC S | LP2RES | AMP |
|        |        | SHAPE2 |     |
|        |        | BAND2  |     |
|        |        | NOTCH2 |     |
|        |        | LOPAS2 |     |
|        |        | HIPAS2 |     |
|        |        | LPGATE |     |
|        |        | NONE   |     |

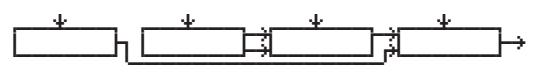


Algorithm 29



- |        |        |        |       |
|--------|--------|--------|-------|
| SYNC M | SYNC S | LPCLIP | x AMP |
|        |        | SINE+  | + AMP |
|        |        | NOISE+ | ! AMP |
|        |        | LOPASS |       |
|        |        | HIPASS |       |
|        |        | ALPASS |       |
|        |        | GAIN   |       |
|        |        | SHAPER |       |
|        |        | DIST   |       |
|        |        | SINE   |       |
|        |        | LF SIN |       |
|        |        | SW+SHP |       |
|        |        | SAW+   |       |
|        |        | SW+DST |       |
|        |        | NONE   |       |

Algorithm 30



- |        |        |        |       |
|--------|--------|--------|-------|
| SYNC M | SYNC S | LPCLIP | x AMP |
|        |        | SINE+  | + AMP |
|        |        | NOISE+ | ! AMP |
|        |        | LOPASS |       |
|        |        | HIPASS |       |
|        |        | ALPASS |       |
|        |        | GAIN   |       |
|        |        | SHAPER |       |
|        |        | DIST   |       |
|        |        | SINE   |       |
|        |        | LF SIN |       |
|        |        | SW+SHP |       |
|        |        | SAW+   |       |
|        |        | SW+DST |       |
|        |        | NONE   |       |

Algorithm 31



- |        |        |       |       |
|--------|--------|-------|-------|
| SYNC M | SYNC S | AMP U | AMP L |
|        |        | BAL   | AMP   |



## Chapter 27

# Memory Upgrades and Other Options

### Program RAM vs. Sample RAM

If you're creating a lot of your own programs, and using samples loaded from disk, there are a few things you should be aware of to avoid perplexity. First of all, there's an important distinction between what we call Sample RAM and what we call Program RAM. Sample RAM refers to any SIMMs you may have had installed in your K2vx. This RAM is reserved exclusively for sample storage; nothing else is stored there. Sample RAM is volatile; that is, when you power down your K2vx, the data stored there will "evaporate" almost immediately. That's why you have to load RAM samples every time you power up.

The amount of sample RAM in your K2vx is indicated in the center of the top line of the Disk mode page. If the center of the display's top line is blank when you're on this page, it means that there is no sample RAM installed in your K2vx (or that the K2vx isn't recognizing it, in which case you should see your dealer or service center).

Program RAM is where all the other RAM objects you create (programs, setups, QA banks, songs, keymaps, etc.) are stored. The K2vx comes from the factory with approximately 240K of available Program RAM. The amount of free Program RAM is indicated at the right side of the top line of the display in Song mode and Disk mode. You can add a Program RAM (P/RAM) option to increase your total available Program RAM to about 1250K. Ask your dealer.

Program RAM is battery-backed, so anything that's stored there will be preserved even when you power down (as long as your batteries have enough juice). Fresh alkaline batteries will last up to two years, so you'll have very few worries about losing your RAM program information. Nonetheless, we recommend that you back up your programs, songs, etc. by saving them to disk. This offers insurance in case the RAM becomes corrupted. This is unlikely, but still a possibility.

If you create a program that uses a disk-loaded sample, the program information (number of layers, keymap assignment, output group, algorithm, etc.) is stored in Program RAM. All RAM samples associated with the program are stored in Sample RAM. This means that when you power down, the RAM samples associated with your programs will disappear. The program information, however, will remain in Program RAM indefinitely. When you power up again, your RAM programs will still appear in the display as you scroll through the program list, but they won't play if they use RAM samples, because the RAM samples are lost when you power down.

### Viewing RAM Objects

If you're a heavy Disk mode user, you'll often be faced with the decision to overwrite, merge, or append objects when you load files from disk. If you're loading into a memory bank that's nearly full, this can be a tricky call, because if you decide to merge or append, there may not be enough open slots in the memory bank to accommodate the objects you load. In this case, the extra objects will be loaded into the next-higher memory bank.

Things get even trickier if you save dependent objects when you save to disk. (A dependent object is any object that's associated with another object stored in a different memory bank—for example, a RAM sample with ID 301 that's used in a program with ID 200. See the discussion of dependent objects in Chapter 13 in the *Performance Guide*.) If you load a file that contains a number of dependent objects, some of them may be loaded into a higher memory bank than

the one you specified in the Bank dialog before you loaded the file. A quick way to see where the objects you loaded ended up is to use the "Objects" Utility function in Master mode.

Select Master mode and press the **Utility** soft button. Press the **Objects** soft button, and a list of RAM objects will appear. Use the Alpha Wheel to scroll through the list of objects. You'll see the type, ID, name, and size (in bytes) of each object.

## Choosing SIMMs for Sample RAM

Single In-Line Memory Modules, commonly referred to by the acronym "SIMM", are the small memory cards that the K2vx uses for Sample RAM. You can add up to eight SIMMs to your K2vx, and since they range in available size up to 16 Megabytes, that means you can add up to 128 Megabytes of Sample RAM to your machine.

The K2vx will accept 30-pin non-composite SIMMs, in sizes of 256K Bytes (obsolete), 1 Megabyte, 4 Megabytes, and 16 Megabytes, in either 8-bit or 9-bit configurations. The SIMMs must have an access time of 80 nanoseconds (ns) or faster. The maximum height and width of a SIMM for the K2vx is 30mm x 90mm (approximately 1.2 inches x 3.5 inches). Below is a list of some SIMMs that will work with your K2vx:

Hitachi HB56A48A; 4Mx8  
Hitachi HB56A49A; 4Mx9  
TI TM124EU9B, TM124EU9C; 1Mx9  
TI TM497EAD9B, TM4100EAD9; 4Mx9  
TI TM4100GAD8, TM497GAD8A; 4Mx8  
TI TM16100GBD8;16Mx8  
TI TM16100EBD9;16Mx9  
NEC MC-421000A8B; 1Mx8  
NEC MC-424100A8B; 4Mx8  
NEC MC-421000A9B; 1Mx9  
NEC MC-424100A9B; 4Mx9  
Tosh THM81000AS, Tosh THM81000BS, Tosh THM81070AS; 1Mx8  
Tosh THM91000AS, Tosh THM91000BS, Tosh THM91070AS; 1Mx9

SIMMs are always installed in pairs, and must be installed by an authorized Kurzweil facility.



**CAUTION:** *You must not use composite SIMMs in your K2vx. A composite SIMM is one that uses a PAL or other additional circuitry to make multiple DRAM chips act like bigger chips. Non-composite SIMMs (the kind you may use in your K2vx) have no chips other than DRAM memory chips soldered to the board. SIMMs with PALs, buffers, or other logic components will not work in your K2vx, and must not be used.*

## Using Headphones with the K2vx

A good pair of headphones can be indispensable when you want to play but need to keep the volume down. You'll get optimum performance from headphones with at least 50 ohms impedance, but anything over 8 ohms is adequate. Headphone volume decreases as the impedance decreases.

# Chapter 28

## Maintenance and Troubleshooting

### Preventive Maintenance

With a modicum of care, your K2vx will give you years of use and enjoyment. There are just a few important points to keep in mind.

Proper installation is essential to the health and welfare of your K2vx. It should be mounted in a standard 19-inch MIDI rack, or should rest on a hard flat surface. In this case it *must* rest on its rubber feet, and **NOT** on the bottom panel. **NEVER** block the ventilation openings on the rear panel; doing so can cause overheating that will seriously damage your K2vx! To provide adequate ventilation, the rear panel should be at least four inches from any vertical surface. If you install an internal hard disk, the ventilation opening on the underside of the K2vx must remain unobstructed so the cooling fan can operate properly. Care should be taken to minimize the amount of dust in the environment.

There are no user-serviceable parts in the K2vx. Under no circumstances should you attempt to remove any panels (except for battery replacement). If you attempt to open your K2vx, you'll risk electric shock, and you'll void your product warranty.

#### Cleaning your K2vx

It's a good idea to remove dust from your K2vx occasionally. You may also want to remove fingerprints. You can clean the K2vx's front panel with a soft damp cloth, and use a mild soap or detergent. Never use strong cleaners or solvents, and never spray anything on the front panel or into the ventilation holes! Any cleaners you may want to use should be applied to your cleaning cloth; you can then carefully wipe the surfaces of the K2vx.

#### Floppy Disk Drive Maintenance

As long as you're reasonably careful to keep dirt and dust out of the floppy disk drive, you shouldn't have any problems. If, however, you start to experience errors or failures in loading or saving, it may be due to dirt in the floppy drive mechanism. See your dealer for information regarding products and techniques for floppy drive cleaning.

### Battery selection and Replacement

The K2vx uses batteries to preserve its internal memory when the power is turned off. The original batteries should last up to two years before they need replacing. Replacement is necessary when the LCD says "BATTERY VOLTAGE IS LOW" during power-up, or when you notice that the LEDs flash three times instead of once during power-up. Once these warnings begin to occur, the batteries should be replaced within a couple of weeks to ensure continued safety of your RAM objects.

To replace the batteries, you'll need access to the rear panel. Remove the battery compartment cover, which is located at the lower right corner of the rear panel as you face it. You may wish to use a small screwdriver. Replace the batteries with three high quality AA size "heavy duty" or alkaline batteries. A capacitor will keep the memory alive for about 30 seconds while changing the batteries, so don't remove the old ones until the new ones are available. Alternatively, it is permissible to have the power cable plugged in and the power on while changing the batteries, in which case memory will be retained as long as power is on. Be sure to insert the new

batteries in the proper direction (the positive terminals should be pointing out). Incorrect insertion won't damage anything but the memory won't receive any power from the batteries.

Most quality brand-name batteries now have "sell by" dates printed on their package. Carbon-zinc batteries will last for at least a year after installation while alkaline batteries should last for at least 2 years provided they are installed before the date on the package. Rechargeable batteries should not be used; the K2vx will not recharge them and their life after charging on an external charger will be only a few months. Battery life is not significantly increased by leaving your K2vx on all the time; batteries may even suffer heat degradation if the K2vx is left on continuously.

## User-callable Diagnostics

There's an onboard diagnostic program that will enable you to check your battery and confirm front panel button functions.

To enter the diagnostic program, simply press the 4, 5, and 6 buttons all at once when in Program Mode. The K2vx will respond by lighting each LED in sequence and then displaying text such as the following on the LCD:

```
K2vx SCANNER DIAGNOSTICS VERSION 2.1
(PRESS "EXIT" AND "ENTER" TO EXIT)
BATTERY=4.3VOLTS, WHEEL CENTER=128
XXXXXXXXXXXXXXXXXXXX
```

The battery voltage and wheel center values may be different on your unit. You can ignore the Wheel Center reading; it doesn't apply to the K2vx. The fourth line (represented by XXXX) gives a readout identifying the buttons you press.

The diagnostic program can also be used to check out the front panel components. If you move the Alpha Wheel clockwise, the numbers will go 0-1-2-3-0-1-2...while counterclockwise should produce 3-2-1-0-3-2-1... If you press a button, its name will be shown and if it is one of the mode buttons, its associated LED should flash.

The third line of the display shows the results of two measurements that are made whenever your K2vx is turned on. The battery voltage will be about 4.3 volts for new batteries, gradually declining over time to 3.2 volts, at which point you will begin to receive warnings (see "Battery Selection and Replacement" above). The line referring to the Wheel is only relevant to the keyboard models of the K2vx.

## Maximizing Music and Minimizing Noise

Your K2vx quite possibly has the lowest noise and widest dynamic range of any instrument in your studio. The following tips will enable you to make the most of this, and optimize the K2vx's audio interface to your other equipment.

Setting your audio levels appropriately is the key to optimizing the signal-to-noise ratio of any piece of equipment. You may have noticed that the K2vx's output signal seems less "hot" than most other synths when using the unedited factory sounds. This is to allow virtually any configuration of voice assignment to be used (up to 96 oscillators directed to one output!), and played very loudly with almost no chance of overload distortion. For more controlled

adjustments, it's best to increase the output level *digitally* (by editing programs) instead of increasing the gain of your amplifier or mixing board. This is because a digital gain increase is completely noiseless whereas an analog increase will proportionally increase hum and noise from the connecting cabling and from the K2vx itself.

Increasing the volume digitally can be accomplished in three different ways. You can increase the volume of all programs assigned to a given MIDI channel by selecting the CHANLS page in MIDI mode and setting the OutGain parameter to the desired level (in 6dB steps). For multi-timbral sequences (on multiple MIDI channels), you will need to do this for each channel. Alternatively you can increase the volume of a single program by going to the OUTPUT page in the Program Editor and setting the Gain parameter to the desired level, again in 6dB steps. For finer adjustment, the Adjust parameter on the F4 AMP page can be used.

Increasing the level too much can cause clipping distortion when multiple notes are triggered with high attack velocity. For dense sequences all played through the same outputs, you will probably only be able to increase the volume by 6dB or so without risk of distortion. For monophonic instruments (lead guitar) or single instrument tracks (such as drums), a substantially greater boost is generally possible.

For the absolute maximum signal quality, the individual outputs should be used. These are connected almost directly to the 18-bit digital-to-analog converters with a minimum of noise-inducing processing circuitry. A total dynamic range of over 100dB is available at these outputs. The MIX outputs are naturally somewhat noisier because they represent the noise of the individual outputs all mixed together, and the signal must travel through more circuitry to reach them.

Programs that are routed through the K2vx's global effects processor (Output Group A) will also be slightly noisier than programs routed to Output Groups B, C, or D. As with an external effects unit, maximizing the input signal level (using the methods described above) will improve the signal-to-noise ratio of the effects processor. When in Effects mode, you'll see an internal Wet/Dry mix parameter; in the Effects Editor you'll find numerous parameters for setting the level of the various effects. Your best signal-to-noise ratio will be achieved by setting the effects level parameters to maximum and adjusting the Wet/Dry Mix parameter to set the overall effects mix. If you are only using the effects unit for EQ functions, one of the EQ or Tone Control functions accessible through the Program Editor will produce quieter results than the global effects processor's EQ functions (Parametric EQ, Para Bass, Para Mid, or Para Treble, for example).

## Ground Hum

A common problem with all electronic musical gear is the hum that can occur in connecting cables due to AC ground loops. Although "3-prong to 2-prong" AC adaptors are frequently used to break ground loops, they also break the safety ground that protects you from electric shock. **Using these adaptors is dangerous, and SHOULD NOT be done!** Furthermore, although using these adaptors may reduce low-frequency hum, high-frequency line noise (such as motor switching noise) is likely to get worse in this case, since the K2vx's AC noise filter will have no outlet for the noise it filters if you disable the ground.

To reduce ground hum, you can increase your output signal levels as described earlier in this section. Other safe procedures include plugging your mixing board and amplifier into the same outlet as your K2vx, and making sure that all your gear is properly grounded. If you're using an external SCSI device, plug it into the same outlet as well. AC isolation transformers are extremely effective at eliminating ground loops, and are recommended for critical installations. A 75-watt transformer is sufficient for the K2vx.

For studio applications, where the utmost signal purity is required, using audio unbalanced-to-balanced line transformers will give you the best results. Each of the K2vx's audio outputs can easily drive a 600-ohm transformer.

Finally, magnetic fields can be a source of interference. The area surrounding the K2vx's Alpha Wheel and alphanumeric pad is sensitive to fields from large transformers in power amps; keep them at least a foot away from the K2vx's front panel. Smaller gear like drum machines and hardware sequencers can also cause interference.

## Power Problems and Solutions

The K2vx is quite tolerant of voltage fluctuations, noise, and transients in the AC power it receives. The input line filter and grounded power cable will protect against even large amounts of noise from motors and the like while the built-in filter coupled with the fuse will protect against all but the largest transients. If your installation is actually suffering from line noise or transients, most likely your other equipment will be suffering more than your K2vx.

Very low line voltage or severe voltage dips are a problem for any computer-based instrument. When the K2vx is set for 120 volt input (the normal North American setting), it should function down to 90 volts. If the line voltage drops below 90 volts, a special circuit halts all activity to protect against software crashes or damage. When the line voltage returns to and stays at an acceptable level for at least one second, the computer will automatically restart. The net effect is just as if you had performed a soft reset. Continuous low line voltage or transient dips will never produce symptoms other than unexpected soft resets as just described. Any other problems such as distortion, disk errors, or lost data are caused by something other than line voltage fluctuations.

Soft resets from line voltage dips are most common. These are easily identified because the reset occurs coincident with the building lights dimming, stage lights or power amps being switched on, or air-conditioning equipment starting up. The solution in all cases is to get a more direct connection between your K2vx (and any other computer-based equipment) and the building's power. Floodlights, large power amplifiers, and motor-operated devices should use a separate extension cord; preferably they should be plugged into a separate outlet.

Chronic low line voltage is best confirmed by measurement. Readings below 100-105 volts mean that even small dips could cause resets, while readings below 95 volts (accounting for meter inaccuracies) are a definite problem. Again, the best solution is to separate your heavy lighting and amplifier loads from your K2vx and other synths on separate extension cords or separate circuits when possible. If the actual building voltage is that low, use of an external step-up transformer or voltage regulator is recommended. We DO NOT recommend changing the line voltage selector to 100 volts (or 220 volts in Europe) because overheating or blown fuses may occur if you leave the K2vx at the lower setting and use it later at a normal voltage level.

## Troubleshooting

Naturally, we've done everything possible to ensure that your K2vx arrives free of defects. And there's a good chance that there's nothing wrong, even if you're not seeing the proper display or hearing the sounds. Carefully check the following things:

Make sure that your power supply is at the right voltage, and is functioning properly.

Make sure the power cable is connected properly.

Adjust the display contrast if necessary (with the Contrast parameter in Master mode). If for some reason you have trouble reading the display, even after adjusting the Contrast parameter, you can also adjust the contrast by holding down the ENTER button and turning the Alpha Wheel. If this improves the contrast, immediately return to the Contrast parameter and adjust it slightly. This will cause the K2vx to remember the current display contrast level, and should take care of any difficulties you may have been having. If this procedure doesn't work, it's time to contact your dealer.



Make sure your audio cables are fully connected to the K2vx and to your sound system. You may want to switch your audio cables, unless you're sure they're functioning properly.

Make sure that your MIDI connections are correct, and that your MIDI cables are functional. You should have at least one MIDI cable, which should be connected from the MIDI Out port of your MIDI controller to the MIDI In port of the K2vx.

Check that the K2vx's Volume slider is at least partially up.

Check the volume level of your sound system.

Lower the volume of your sound system, and turn the K2vx off, then on again (this is called a power cycle).

Press the +/-, 0, and Clear buttons (on the alphanumeric pad at the far right of the front panel) at the same time. This is called a soft reset.

As a last resort, save to disk any RAM objects you've created, and perform a hard reset. Do this by pressing the Master Mode button, then pressing the "Reset" soft button (at the lower right of the display). The K2vx will warn you about deleting everything (only RAM objects will be deleted). Press Yes. After a few seconds, the power-up display should appear.

Also check the suggestions on the following pages. If it's still not happening, the next step is to shut off the power and call your dealer.

## Other Possible Problems

Condition	Possible Cause
<i>No sound, no display, no LEDs illuminated.</i>	<ol style="list-style-type: none"> <li>1 AC line cord not fully inserted into outlet or unit. If using a multiple outlet box, check its plug.</li> <li>2 Power not on at AC power source (wall outlet). Check with a different appliance.</li> <li>3 Power switch not on (either the unit or multiple outlet box).</li> <li>4 Incorrect voltage selection setting. REFER TO QUALIFIED SERVICE PERSONNEL.</li> </ol>
<i>No sound.</i>	<ol style="list-style-type: none"> <li>1 Volume control turned all the way down on the K2vx or on amplifier or mixer.</li> <li>2 Amplifier or mixer not turned on.</li> <li>3 Cabling is not correct - see Chapter 2 in the <i>Performance Guide - Startup</i>, and Chapter 18 in the <i>Performance Guide - Audio Outputs</i>. Also check that amplifier, mixer and speaker cabling is correct.</li> <li>4 MIDI volume has been assigned to a control source which has sent a value of 0. Pressing the Panic soft button will reset all controls, and resolve this problem.</li> </ol>
<i>No sound at MIX outputs or headphones.</i>	<ol style="list-style-type: none"> <li>1 Audio cables are plugged into some or all of the separate outputs. Cables plugged into of the separate outputs will remove some or all of the audio signal from the MIX and headphone outputs. See Chapter 18 in the <i>Performance Guide</i>—"Audio Output" for output configurations.</li> </ol>

*Left MIX output seems louder than Right MIX output when used individually.*

- 1 This is normal. When a cable is plugged into the left MIX output alone, both the left AND the right audio signals are routed to the jack. When a cable is plugged into right MIX output alone, only the right channel audio signal is heard.

*Volume knob has no effect.*

- 1 Separate outputs are in use - the volume knob does not affect the separate outputs.
- 2 MIDI volume has been assigned to a control source which has sent a value of 0.

*Programs, Setups, Songs, etc. are missing.*

- 1 Batteries have run down or have been disconnected. If the batteries have failed, the message "Battery voltage is low - X.X volts" (where X.X is less than 3.0) will appear in the display on power-up. All user data will be permanently lost if this occurs. See the information on battery selection and replacement elsewhere in this manual.

*LCD is too dark or too light to read.*

- 1 Contrast not adjusted. Select Master mode and adjust the Contrast parameter. If this fails, hold the ENTER button and turn the alpha wheel clockwise to make display darker; counterclockwise to make it lighter. Then adjust the Master mode Contrast parameter to a higher value if the LCD was too light, or to a lower value if it was too dark.

*Cannot mount or read disk.*

- 1 Disk is not MS-DOS (or Akai, Ensoniq, or Roland) format.
- 2 Disk is damaged.

*Cannot write floppy disk.*

- 1 Disk is not MS-DOS formatted.
- 2 Disk write protected.
- 3 Sample is copy protected.
- 4 Disk is damaged.

*Cannot format disk.*

- 1 Disk is damaged.
- 2 Disk is write protected.
- 3 You have instructed the K2vx to format a Double density (720K) disk as a high-density (1.4M) disk. NOTE: Punching a hole in a double-density disk case to try to make the K2vx read it as a high-density disk is not recommended.

# Chapter 29

## MIDI and SCSI Sample Dumps

### SCSI Guidelines

The following sections contain information on using SCSI with the K2vx, as well as specific sections dealing with the Mac and the K2vx.

Here are some basic guidelines to follow when configuring a SCSI chain:

1. According to the SCSI Specification, the maximum SCSI cable length is 6 meters (19.69'). You should limit the total length of all SCSI cables connecting external SCSI devices with Kurzweil products to 17 feet (5.2 meters). To calculate the total SCSI cable length one must add up the lengths of all SCSI cables, plus 8" for every external SCSI device connected. No single cable length in the chain should exceed 8 feet.
2. The first and last device in the chain must be terminated.

Poor termination is a common cause of SCSI problems. If you install an internal drive that is terminated, the termination resistors should be removed from the K2vx Engine Board. Having more than two terminators on the bus will overload the bus drivers, but this should not cause permanent damage to the hardware. Poor termination can corrupt the data on your disk, however, as can bad SCSI cables.

For the K2vxR, if it is not located at one end of a SCSI chain all internal termination, including the terminator resistor network on the K2vx Engine Board plus terminator resistors in the internal SCSI drive must be removed. It is much simpler to just make sure that the K2vx is at one end of the SCSI chain.

A note about active termination - The K2vx uses active termination of the SCSI bus. Active termination has some benefits over traditional passive termination. Some people have positioned active termination as a panacea for SCSI problems, but this is more hype than reality. Active terminators are available to be used at the end of one's SCSI chain and all APS SR2000 series external drives use internal active termination that can be switched on or off.

3. Each device in the chain (including internal hard drives) must have its own unique SCSI ID. The default K2vx ID is #6. Macintoshes use ID #7 & #0.
4. Use only true SCSI cables - high quality, twisted pair, shielded SCSI cable. Do not use RS432 or other non SCSI cables.

The majority of SCSI cables we've tested were poorly made and could damage data transferred to and from the disk. Nearly all the SCSI data problems Young Chang's engineering department has had have been due to bad cables that didn't twist pairs of wires properly. Correctly made SCSI cables have one ground wire for every signal wire and twist them together in signal/ground pairs. Cables made by APS Technologies (800-233-7550) are very good and are highly recommended. Young Chang manufactures 1 and 2 meter 25-25 SCSI cables, that we can also recommend. Good cables are essential to reliable data transfers to and from the disk drive.

5. One should buy all SCSI cables from a single source to avoid impedance mismatch between cables.

6. Theoretically all eight SCSI IDs can be used. However, feedback from users has shown us that many people have problems with more than 5-6 devices in a chain. If you have 7 or 8 devices and are having problems, your best bet is to make sure you have followed all of the previous information, especially with respect to cables.
7. Connect all SCSI cables before turning on the power on any equipment connected by SCSI cables. Plugging or unplugging SCSI cables while devices are powered on can cause damage to your devices or instrument.
8. Authorized service centers should remove termination from the K2vx when installing an internal drive, set its ID correctly, and terminate the drive.
9. When using a Macintosh, power up the K2vx and other devices first.
10. The K2vx file format is a proprietary format; no other device will be able to read or write a Kurzweil file.
11. The floppy disk format of the K2vx is DOS. The SCSI disk format is a proprietary form that is close to DOS, but it is not DOS. Nonetheless, the K2vx can read and write to a DOS formatted disk provided it was formatted on the PC with no partitions.
12. It is possible to view, copy, move, name, delete files on a K2vx formatted floppy disk or removable media hard drive, with a PC or Macintosh running a DOS mounting utility program such as Access PC.
13. As long as the SCSI bus is properly terminated there is no way you can damage your hardware simply by operating it. There are a few hazards K2vx users should be aware of, however:

The only damage that usually occurs to SCSI hardware comes from static electricity "zapping" SCSI connector pins when the cables are disconnected. The silver colored shell of the SCSI connector on the end of the cable is connected to ground and is safe to touch, but the brass colored pins inside eventually lead to the SCSI interface chip and are vulnerable. One should discharge static from one's body before touching SCSI connectors by touching the 1/4" jacks on the rear of the K2vx or another grounded metal object. Any devices connected to the SCSI bus should be turned off when plugging or unplugging SCSI cables.

If the K2vx is connected to a Macintosh or PC you should make sure that the computer cannot access a SCSI disk at the same time the K2vx does (see below for more information on this). Those who occasionally want to share a drive, but don't want to take any risks would be best served by disconnecting and connecting devices as needed. If you want to share drive(s) often and cannot constantly disconnect and reconnect devices, make sure the Mac or PC is really done with the disk before using the K2vx. Furthermore, you should quit or exit from all running programs and disable screen savers, email, network file sharing, and any INITs or TSR's that run in the background. If the computer and K2vx access the disk at the same time there will be no damage to the hardware, but the bits on the disk, K2vx, and computer memory can easily be corrupted. You may not know that damage has been done to these bits until weird things start to happen for no apparent reason.

## K2vx and Macintosh Computers

1. The Mac really does not like having another SCSI master on the bus (i.e., the K2vx). It assumes that it owns the bus and its drives, therefore it will not tolerate the situation where the K2vx is trying to talk to its (the Mac's) disk. This suggests that you never want to select the ID of any drive mounted on the Mac's desktop. Even more fundamental is the problem that the Mac assumes that the bus is always free, so if it tries to do anything via

SCSI when the K2vx is doing anything via SCSI, the Mac will freak. The only solution is, wait until your Mac is completely idle before accessing SCSI from the K2vx.

2. The Mac and the K2vx cannot share a drive in any way, with or without partitions. If you are using a drive with removable media, you cannot easily switch back and forth between a Mac formatted volume and a K2vx formatted volume. To prevent problems, you will need to unmount the drive from the Mac desktop before switching to a K2vx format volume. The Mac will basically ignore the volume if it's not Mac format, but once you insert a Mac format volume, the Mac owns it. Don't forget about #1 above; inserting a cartridge will cause the Mac to access SCSI, so don't try to use the K2K at that moment.
3. The only good reason for connecting the Mac and the K2K on the same SCSI bus is to use Alchemy or equivalent. If you're using a patch editor or librarian, you can just hook up via MIDI. Connecting via SCSI will allow fast sample transfers through the SMDI protocol. In this type of configuration the easiest solution is to let the K2K have its own drive, and the Mac have it's own drive.

However, we have discovered that when using a K2vx with a Mac and a removable media drive in the middle of the chain, the following scenario will work:

Start with a Mac formatted cartridge in the drive. When you want to use the K2vx, put the drive to sleep from the K2vx. You can then change to a K2vx formatted cartridge and perform whatever disk operations you need. When you want to go back to the Mac, put the drive to sleep again, switch cartridges, and then wake up the disk by pressing **Load**. Of course the K2vx will tell you it can't read the cart, but the Mac will now access it fine.

## Accessing a K2vx Internal Drive from the Mac

Access PC is one of the many programs for the Mac which allow it to format, read, and write to DOS floppy disks and removable SCSI cartridges. However, we have discovered that it is possible to format internal K2vx hard drives, even though the documentation claims to only support removable media (not a fixed drive). Because the program claims not to be able to do this, we do not necessarily recommend it.

The main thing to remember is:

Never change the disk contents (i.e., save or delete files) from the K2vx when the disk is mounted by the Macintosh. If you do, this could easily lead to trashed files, directories, or even the entire disk. Access PC has no way of knowing when the K2vx has modified the disk structure, and it can just overwrite any state of the disk it thinks should be there. The safest thing is to connect a drive to either the K2vx or the Mac, but not both at the same time. Of course, you can't always predict when a Mac will access its drive, and it doesn't do SCSI bus arbitration, so using the Mac while using the SCSI bus from the K2vx (e.g., doing a disk mode operation) is also a bad idea, and can cause the Mac to hang.

## The MIDI Sample Dump Standard

Samples can be transferred between the K2vx and most other samplers and computer sampling programs using the MIDI Sample Dump Standard.

Due to the relatively slow transfer rate of MIDI data, transferring samples into the K2vx via the MIDI Sample Dump Standard can take a long time, on the order of a coffee break for a long sample. Most samplers, synthesizers, and computer software will "freeze up" during this process, preventing other features of the machine or program from being used. Your K2vx, however, will allow you to continue playing the instrument or using any of its sound editing features during a MIDI Sample Dump! The transfer takes place in the background; the MIDI mode LED on the K2vx's front-panel will flash repeatedly during the transfer, so you will

always know if the MIDI Sample Dump is still proceeding. The MIDI mode LED will flash only when the K2vx is transmitting or receiving a MIDI Sample Dump, or when it receives a MIDI System Exclusive message.

Note: if you're using Sound Designer® to transfer samples, you'll have to offset the sample number by 2 to transfer the right sample. For example, if you want to dump sample ID 208 from the K2vx, then when you begin the sample fetching command from Sound Designer, instruct it to get sample 210.

### Loading Samples with the MIDI Standard Sample Dump

To load a sample into the K2vx from an external source such as a computer or sampler, first connect the MIDI Out port of the sampler (or computer) to the K2vx's MIDI In port, and connect the K2vx's MIDI Out to the MIDI In of the sampler. This is known as a closed-loop configuration.

Next, access the Sample Dump facility on the sampler. In addition to selecting which sample you wish to transfer over MIDI, you will need to set the correct sample dump channel number and destination sample number. The channel number should match the K2vx's SysX ID parameter (on the RECV page in MIDI mode). If the sampler has no facility for setting the Sample Dump channel number, try setting the K2vx's SysX ID parameter to 0 or 1. Alternatively, if you set the SysX ID to 127, the K2vx will accept a MIDI Sample Dump no matter what Sample Dump channel is used to send the sample dump.

If the sampler has a provision for setting the destination sample number, you can use it to specify the ID the K2vx will use for storing the sample. The K2vx sample number is mapped from the destination sample number as follows:

Sample Number	K2vx ID
0	uses lowest unassigned ID between 200 and 999.
1-199	adds 200 to the ID (i.e. 5 becomes 205 in the K2vx.)
200-999	ID is the same number.

If the sample number maps to a number already assigned to a RAM sample in the K2vx, the RAM sample will be deleted prior to the K2vx's accepting of the new sample load. The K2vx will always map sample number zero to an unassigned ID, and therefore no samples will be overwritten when zero is specified.

Some computer-based sample editing software limits the sample numbers to a low range such as 1-128. This conflicts with the K2vx, which reserves IDs 1-199 for ROM samples, which cannot be loaded or dumped. To get around this, the K2vx adds 200 to any numbers between 1 and 199. Therefore, if you want to load a sample into the K2vx at number 219, but your program can't transfer samples at numbers greater than 128, specify number 19 (There's an exception to this; please see "Troubleshooting a MIDI Sample Dump" later in this section).

At this point, you're ready to try loading a sample. See "Accessing a New K2vx Sample" later in this section to learn how to use samples once they've been dumped to the K2vx.

### Getting a Sample into a Sample Editor from the K2vx

Connect the MIDI ports of the K2vx and the computer/sampler in a closed-loop configuration as described for the Sampler/Computer to K2vx procedure above.

Access the computer software's "Get Sample" page (it might be called something different). As with loading a sample into the K2vx, the K2vx adds 200 to dump request sample numbers between 1 and 199. K2vx samples with IDs from 1 to 199 are ROM samples, and cannot be

dumped. Therefore, if you want to get sample number 219 from the K2vx but your program can't transfer samples at numbers greater than 128, specify number 19 (There's an exception to this; please see "Troubleshooting a MIDI Sample Dump" later in this section).

### **Loading a Sample into the K2vx from another K2vx**

Connect the MIDI ports of the two K2vxs in a closed-loop configuration as described for the Sampler/Computer to K2vx procedure above.

On the source K2vx, go to the Sample Editor and select the sample you wish to transfer. To do this, start in Program mode and press EDIT, followed by the KEYMAP soft button. Now you should be on the KEYMAP page. Now move the cursor to the Sample parameter, use any data entry method to select the desired sample, then press EDIT.

To start the sample transfer, press the Dump soft button. A dialog will appear, suggesting the ID for the sample to be dumped to the destination K2vx. The source K2vx will suggest the same ID as it uses for the sample, but you can change the destination ID with any data entry method. If you choose the default by pressing Yes, the sample will transfer to the same ID on the destination K2vx as it is on the source K2vx.

### **Dumping from the K2vx to a Sampler**

This procedure is the same as dumping a sample from one K2vx to another. This will work only if the sampler supports the MIDI Sample Dump Standard.

### **Dumping a Sample from the K2vx to a MIDI Data Recorder**

This can be accomplished in an open-loop configuration, by connecting the MIDI Out port of the K2vx to the MIDI In port of the MIDI Data Recorder. Go to the Sample Editor and select the K2vx sample you wish to transfer. Set up the MIDI Data Recorder to begin recording, and press the Dump soft button on the Sample Editor page. This will bring up a dialog allowing you to change the sample number in the dump if you wish. In most cases, you will just use the default value. The K2vx's MIDI mode LED will flash while the data transfer is in progress.

### **Loading a Sample into the K2vx from a MIDI Data Recorder**

Connect the MIDI Out port of the Data Recorder to the MIDI In port of the K2vx. Load the appropriate file containing the MIDI Sample Dump data into the Data Recorder, and send the file. The K2vx's MIDI mode LED will flash during this procedure.

### **Accessing a New K2vx Sample**

First, select the K2vx program you wish to play the new sample from, and press the EDIT button. Then select the layer you wish (using the CHAN/BANK buttons if necessary), press the KEYMAP soft button, and select a keymap. Use the default keymap called "168 Silence" if you don't want to alter any existing keymaps.

Now, enter the Keymap Editor by pressing EDIT once again. Use the Sample parameter to select the new sample. If the new sample was loaded from another K2vx, it will have the same ID as it did on the other K2vx. If the sample was loaded from any other source, its ID will be defined as described above (in the section called "Loading Samples with the MIDI Standard Sample Dump").

The name of the sample will be assigned by the K2vx if the sample has been assigned to a previously unused ID. In most cases, the sample will have a name of "New Sample - C 4".

The name will be “New Sample! - C 4” (note the exclamation point) if checksum errors were detected by the K2vx. Checksum errors are usually not serious, since they may just mean the source sampler doesn’t adhere to the MIDI Sample Dump Standard checksum calculation. In other cases, a checksum error could indicate that the MIDI data flow was interrupted during the sample transfer.

You can now press EDIT to edit the parameters of the new sample such as Root Key, Volume Adjust, Pitch Adjust, and Loop Start point. You can also rename the sample. Be sure to save the parameters you change when you press EXIT. Once the sample is adjusted to your liking, you can assign it to any Keymap.

### Troubleshooting a MIDI Sample Dump

This section will help you identify what has gone wrong if your MIDI sample dumps fail to work.

#### WHEN LOADING SAMPLES TO THE K2vx

There are two reasons a K2vx will not accept a MIDI Sample Dump. First, a dump will not be accepted if the destination sample number maps to a K2vx sample that is currently being edited—that is, if you’re in the Sample Editor, and the currently selected sample has the same ID as the sample you’re trying to dump. Second, a dump will not be accepted if the length of the sample to be dumped exceeds the available sample RAM in the K2vx. There may be samples in the K2vx RAM that you can save to disk (if not already saved) and then delete from RAM to free up sample RAM space. You can delete the current sample by pressing the Delete soft button while in the Sample Editor.

Note that when you’re loading a sample to an ID that’s already in use, the K2vx will not accept a MIDI Sample Dump if the length of the sample to be loaded exceeds the amount of available sample RAM *plus* the length of the existing sample. If the K2vx accepts the sample load, the previously existing sample will be deleted.

Also note that certain computer-based editing programs will subtract one from the sample number when performing MIDI sample transfers to remote devices. So if you instruct these programs to send a sample to the K2vx as sample ID 204, the program will send the sample as 203. The only way to know if your program behaves in this manner is to try a MIDI Sample Dump and see what happens.

#### WHEN DUMPING SAMPLES FROM THE K2vx

Certain computer-based sample editing programs subtract one from the sample number when performing MIDI Sample transfers to remote devices. For instance, if you tell these programs to get sample number 204, the programs will request that the K2vx dump sample ID 203, which would ordinarily dump a different sample from the one you intended, possibly causing the dump to fail. The K2vx automatically counteracts this offset by adding a number to sample requests. This was done because more sample editing programs create this offset than do not. If you find that the K2vx is sending samples with higher IDs than the ones you requested, you can compensate by requesting the sample ID one lower than the one you want. For example, if you want the K2vx to dump sample 205, ask for sample 204.

Some samples in the K2vx are copy-protected. These include all ROM samples and possibly some third-party samples. The K2vx will not dump these samples.

### Aborting a MIDI Sample Dump

The Abort soft button in the Sample Editor can be used to cancel any sample load into the K2vx from an external source (e.g. a computer or a sampler). This button will also halt a sample dump from the K2vx. The K2vx will ask for confirmation before it aborts the sample dump.



## SMDI Sample Transfers

You can use Passport's Alchemy® and Opcode's Max® SMDI-capable Macintosh® software packages to transfer mono and stereo samples to and from the K2vx. These applications use the SMDI data transfer format (SMDI stands for SCSI Musical Data Interchange—pronounced *smiddy*). SMDI is parallel, not serial, so sample transfers can be made much faster than with the MIDI sample dump standard.

Each of these applications has commands for getting and sending samples, which is how you'll make the transfer from your offline storage to the K2vx. Once the samples have been loaded to the K2vx, you can use the Keymap and Sample Editors as you would with any other sample. Check your manuals for Alchemy or Max for the specifics.

Keep in mind that when transferring samples via SMDI, the K2vx's sound engine is disabled, so you can't play it during a SMDI transfer as you can during a MIDI sample transfer. The average SMDI sample transfer time is about 20K per second.

As of this writing, the latest versions of Alchemy and Max are the only software packages supporting SMDI sample transfers to and from the K2vx. SMDI is a new technology, however, and many software developers are working on packages that will support K2vx SMDI sample transfers. Your Kurzweil/Young Chang dealer can let you know about new developments.

**MIDI and SCSI Sample Dumps**

---

SMDI Sample Transfers

# Chapter 30

## System Exclusive Protocol

### K2vx System Exclusive Implementation

The MIDI System Exclusive capabilities of the K2vx allow you to manipulate objects in the K2vx's memory from a computer system, another K2vx, or a MIDI data recorder. The following is a reference to the SysEx protocol used by the K2vx. This information can be used to build a simple object librarian software program. A word of advice—before you begin experimenting with SysEx, make sure you have saved anything of value in RAM to disk.



***NOTE:** To support new features and changes in the K2vx line of products, the internal program structure has been changed from that of the K2vx. Due to these changes, you cannot transfer a K2vx program to a K2vx, or a K2vx program to a K2vx via MIDI system exclusive. The K2vx software will continue to be enhanced, and in the future the K2vx will be capable of accepting K2vx programs over MIDI. As a result of this, computer based K2vx editor/librarians will not currently work with the K2vx, unless they have been revised to accommodate the changes.*

#### Common Format

In the following discussion, the fields of the K2vx System Exclusive Protocol messages are notated as...

#### **field(length)**

...where 'field' is the name of the particular information field in the message, and 'length' is either 1, 2, 3, or n, representing the number of sequential MIDI bytes that make up the field. A length of 'n' means that the field is of a variable length that is determined by its contents or sub-fields.

All K2vx SysEx messages have the common format:

**sox(1) kid(1) dev-id(1) pid(1) msg-type(1) message(n) eox(1)**

'sox' is always F0h, and represents start of System Exclusive.

'kid' must be 07h, and is the Kurzweil Manufacturer ID.

'dev-id' is Device ID. The K2vx will recognize a SysEx message if the 'dev-id' is the same as the SysX ID parameter from the MIDI Receive page (from the top level, press the MIDI mode button and the RECV soft button.) If the K2vx's SysX ID parameter is set to 127, it will recognize SysEx messages no matter what the 'dev-id' is.

'pid' is the Product Identifier, and must be 78h (120 decimal), indicating the SysEx message is for the K2vx.

'msg-type' is the identifier of one of the K2vx SysEx messages defined below, and 'message' is the variable-length message contents.

'eox' is always F7h, for end of System Exclusive.

#### Data Formats

K2vx SysEx messages are subdivided into fields that contain data in different formats. The various fields are shown in the Messages section below. Within a message, any fields for values that can be bigger than 7 bits are broken into 7 bit chunks. Thus two MIDI bytes gives 14 bits, three bytes gives 21 bits. The significant bits are right justified in the field. All bytes in a field must be present no matter what the value is. For example, an object type of 132 would be split into two MIDI bytes in a 'type' field as 01 04:

decimal: 132  
binary: 10000100  
binary encoding for type(2) field: 0000001 0000100  
decimal encoding for type(2) field: 1 4

Object name fields are sent as a string of ASCII values in a 'name' field, with one MIDI byte of zero as a string terminator. For example, the name "Glass Kazoo" would be sent as letters:

G l a s s \_ K a z o o <null>

hex encoding for 'name' field: 47 6C 61 73 73 20 4B 61 7A 6F 6F 00

Data sizes and offsets are sent in the 'size' and 'offs' fields. These values refer to quantities of 8-bit bytes in the K2vx's memory, which is packed in the 'data' field.

Binary data in the 'data' field is sent by in one of two formats, according to the value of the 'form' field. If the 'form' field equals zero, the data is transmitted as 4 bits or one "nibble" in every MIDI byte. If the 'form' field equals one, then the data is sent as a compressed bit-stream, with 7 bits per midi byte. The bit-stream format is more efficient for data-transmission, while the nibble format is easier to read (and write software for).

For example, to send the following four K2vx data bytes,

hex: 4F D8 01 29  
decimal: 79 216 1 41  
binary: 01001111 11011000 00000001 00101001

eight MIDI bytes are sent in "nibble" format:

hex: 04 0F 0D 08 00 01 02 09  
decimal: 4 15 13 8 0 1 2 9  
binary: 0000100 0001111 0001101 0001000 0000000 0000001 0000010 0001001

five MIDI bytes are sent in bit-stream format:

hex: 27 76 0 12 48  
decimal: 39 118 0 18 72  
binary: 0100111 1110110 0000000 0010010 1001000

The bit-stream format can be thought of as taking the binary bits of the K2vx data and, starting from the left, slicing off groups of 7 bits. Note that the trailing bits are set to zero.

After the 'data' field, there is another field, 'xsum'. This is a checksum field which is calculated as the least significant 7-bits of the sum of all of the MIDI bytes that make up the 'data' field.

**Messages**

This section defines the K2vx System Exclusive message formats. Each message has a message type (that goes in the 'msg-type' field; see Common Format, above), followed by the field definitions of the message.

**DUMP = 00h**      **type(2) idno(2) offs(3) size(3) form(1)**

...requests the K2vx to send a data dump of an object or portion thereof. 'type' and 'idno' identify the object. 'offs' is the offset from the beginning of the object's data and 'size' describes how many bytes should be dumped starting from the offset. 'form' indicates how the binary data is to be transmitted (0=nibblized, 1=bit stream). The response is a LOAD message:

**LOAD = 01h**      **type(2) idno(2) offs(3) size(3) form(1) data(n) xsum(1)**

...which writes data into the specified object, which must exist. Both load and dump operate on the object data only. The response to a load message will be

**DACK = 02h**      **type(2) idno(2) offs(3) size(3)**

...meaning "load accepted", or

**DNAK = 03h**      **type(2) idno(2) offs(3) size(3) code(1)**

...meaning "load not accepted." The 'code' field indicates the cause of the failure, as follows::

<b>code</b>	<b>meaning</b>
1	Object is currently being edited
2	Incorrect checksum
3	ID out of range (invalid)
4	Object not found (no object with that ID exists)
5	RAM is full

To request information about an object, use:

**DIR = 04h**      **type(2) idno(2)**

The 'type' and 'idno' identify the object. The response is an INFO message:

**INFO = 05h**      **type(2) idno(2) size(3) ramf(1) name(n)**

This is the response to DIR, NEW, or DEL. If object is not found, 'size' will be zero and 'name' will be null. 'ramf' is 1 if the object is in RAM.

**NEW = 06h**      **type(2) idno(2) size(3) mode(1) name(n)**

...creates a new object and responds with an INFO message of the created object. The object's data will not be initialized to any default values. If 'idno' is zero, the first available object ID number will be assigned. If 'mode' is 0, the request will fail if the object exists. If 'mode' is 1, and the object exists in ROM, a RAM copy will be made. If 'mode' is 1, and the object exists in RAM, no action is taken.

**DEL = 07h**      **type(2) idno(2)**

...deletes an existing object and responds with an INFO message for the deleted object. If there is only a RAM copy of the object, the response will indicate that the object doesn't exist anymore. However, if the deletion of a RAM object uncovers a ROM object, the INFO response will refer to the ROM object. A ROM object cannot be deleted.

**CHANGE = 08h    type(2) idno(2) newid(2) name(n)**

...changes the name and/or ID number of an existing object. If 'newid' is zero or 'newid' equals 'idno', the ID number is not changed. If 'newid' is a legal object id number for the object's type, then the existing object will be relocated in the database at the new ID number. This will cause the deletion of any object which was previously assigned to the 'newid'. If the 'name' field is null, the name will not change. Otherwise, the name is changed to the (null-terminated) string in the 'name' field.

**WRITE = 09h    type(2) idno(2) size(3) mode(1) name(n) form(1) data(n) xsum(1)**

...writes an entire object's data directly into the database. It functions like the message sequence DEL followed by NEW followed by a LOAD of one complete object data structure. It first deletes any object already existing at the same type/ID. If no RAM object currently exists there, a new one will be allocated and the data will be written into it. The object name will be set if the 'name' string is non-null. The response to this message will either be a DACK or a DNAK, as with the load message. The 'offs' field of the response will be zero. The K2vx will send a WRITE message whenever an object is dumped from the front-panel (using a "Dump" soft-button), or in response to a READ message.

The 'mode' field is used to determine how the 'idno' field is interpreted.

If 'mode' = 0:

The 'idno' specifies the absolute ID number to write to, which must exist.(must be valid)

If 'idno' equals zero, write to the first available ID number.

If 'mode' = 1

The object is written at the first available ID number after what is specified by 'idno'.

It doesn't matter if 'idno' is a legal ID number. Remember that for certain object types, the 100s through 900s banks allow fewer than 100 objects to be stored (for example, the 100s bank will store preset effects at IDs 100—109 only). In this mode, if 'idno' was 313, the object would be written to ID 400 if available.

**READ = 0Ah    type(2) idno(2) form(1)**

...requests the K2vx to send a WRITE message for the given object. No response will be sent if the object does not exist.

**READBANK = 0Bh type(2) bank(1) form(1) ramonly(1)**

...requests the K2vx to send a WRITE message for multiple objects within one or all banks.

'type' and 'bank' specify the group of objects to be returned in WRITE messages. The 'type' field specifies a single object type, unless it is zero, in which case objects of all user types will be returned (see object type table below). The 'bank' field specifies a single bank, 0-9, unless it is set to 127, in which case objects from all banks will be returned.

'form' requests the format of the binary data in the WRITE messages. If 'ramonly' is one, only objects in RAM will be returned. If 'ramonly' is zero, both RAM and ROM objects are returned.

The responses, a stream of complete WRITE messages, will come out in order of object type, while objects of a given type are in order by ID number, from lowest to highest. If no objects are found that match the specifications, no WRITE messages will be returned. After the last WRITE message, an ENDOFBANK message (defined below) is sent to indicate the completion of the bank dump.

The K2vx will insert a small delay (50ms) between WRITE messages that it issues in response to a READBANK message.

A bank dump can be sent in its entirety to another K2vx, which will add all of the objects contained in the dump to its own object database. IMPORTANT: If the K2vx receives a large bank dump for a bank or banks that already contain objects, errors may result unless the sender waits for the DACK message before sending the next object's WRITE message. One way to avoid transmission errors such as this is to make sure that the bank being dumped is clear in the K2vx before sending the dump, so that the K2vx will not miss parts of the dump while its CPU is busy deleting already existing objects. This can be done using the DELBANK message (defined below). If the destination bank in the K2vx is pre-cleared, it is not necessary to wait for the DACK before sending. Even if the sender chooses not to wait for the DACK before sending the next message, it may be necessary to preserve the 50ms delay between the WRITE messages.

Due to the large amount of incoming data during a bank dump containing many objects, the receiving K2vx may have a more sluggish response to front-panel use and keyboard playing during the data transfer. This is normal behavior and the machine will become fully responsive as soon as the dump is finished.

**DIRBANK = 0Ch type(2) bank(1) ramonly(1)**

This is similar to the READBANK message. The DIRBANK message requests an INFO message (containing object size, name, and memory information) be returned for each object meeting the specifications in the 'type' and 'bank' fields. Following the last INFO response will be an ENDOFBANK message.

**ENDOFBANK = 0Dh type(2) bank(1)**

This message is returned after the last WRITE or INFO response to a READBANK or DIRBANK message. If no objects matched the specifications in one of these messages, ENDOFBANK will be the only response.

**DELBANK = 0Eh type(2) bank(1)**

This message will cause banks of objects (of one or all types) to be deleted from RAM. The 'type' and 'bank' specifications are the same as for the READBANK message. The deletion will take place with no confirmation. Specifically, the sender of this message could just as easily delete every RAM object from the K2vx (e.g. 'type' = 0 and 'bank' = 127) as it could delete all effects from bank 7 (e.g. 'type' = 113, 'bank' = 7.)

**MOVEBANK = 0Fh type(2) bank(1) newbank(1)**

This message is used to move entire banks of RAM objects from one bank to another. A specific object type may be selected with the "type" field. Otherwise, if the "type" field is unspecified (0), all object types in the bank will be moved. The "bank" and "newbank" fields must be between 0 and 9. The acknowledgement is an ENDOFBANK message, with the "bank" field

equal to the new bank number. If the operation can't be completed because of a bad type or bank number, the ENDOFBANK message will specify the old bank number.

**PANEL = 14h      buttons(3n)**

...sends a sequence of front-panel button presses that are interpreted by the K2vx as if the buttons were pressed at its front-panel. The button codes are listed in a table at the end of this chapter. The K2vx will send these messages if the Buttons parameter on the XMIT page in MIDI mode is set to On. Each button press is 3 bytes in the message. The PANEL message can include as many 3-byte segments as necessary.

Byte 1      Button event type:

08h      Button up

09h      Button down

0Ah      Button repeat

0Dh      Alpha Wheel

Byte 2      Button number (see table)

Byte 3      Repeat count (number of clicks) for Alpha Wheel; the count is the delta (difference) from 64—that is, the value of the byte minus 64 equals the number of clicks. A Byte 3 value of 46h (70 dec) equates to 6 clicks to the right. A Byte 3 value of 3Ah (58 dec) equates to six clicks to the left. For example, the equivalent of 6 clicks to the right would be the following message:

(header) 14h 0Dh 40h 46 (eox)

For efficiency, multiple button presses should be handled by sending multiple Button down bytes followed by a single Button up byte (for incrementing with the '+' button, for instance.)

**Object Types**

These are the object types and the values that represent them in 'type' fields:

Type	ID (decimal)	ID (hex)	ID(hex, 'type' field)
Program	132	84h	01h 04h
Keymap	133	85h	01h 05h
Effect	113	71h	00h 71h
Song	112	70h	00h 70h
Setup	135	87h	01h 07h
Soundblock	134	86h	01h 06h
Velocity Map	104	68h	00h 68h
Pressure Map	105	69h	00h 69h
Quick Access Bank	111	6Fh	00h 6Fh
Intonation Table	103	67h	00h 67h



## Master Parameters

The Master parameters can be accessed as type 100 (00h 64h), ID number 16. Master parameters cannot be accessed with any of the Bank messages.

## Button Press Equivalence Table

<u>Button</u>	<u>Code (hex)</u>	<u>Button</u>	<u>Code(hex)</u>
<b><i>Alphanumeric pad</i></b>		<b><i>Soft-Buttons 'A-F'</i></b>	
zero	00	A (leftmost)	22
one	01	B	23
two	02	C	24
three	03	D	25
four	04	E	26
five	05	F (rightmost)	27
six	06	AB	28
seven	07	CD (two center)	29
eight	08	EF	2A
nine	09	YES	26
+/-	0A	NO	27
<b><i>Alphanumeric pad</i></b>		<b><i>Edit/Exit</i></b>	
CANCEL	0B	EDIT	20
CLEAR	0C	EXIT	21
ENTER	0D		
<b><i>Navigation</i></b>		<b><i>Mode Selection</i></b>	
Plus (+)	16	PROGRAM	40
Minus (-)	17	SETUP	41
Plus and Minus	1E	QUICK ACCESS	42
CHAN/BANK Inc	14	EFFECTS	47
CHAN/BANK Dec	15	MIDI	44
CHAN/BANK Inc/Dec	1C	MASTER	43
Cursor Left	12	SONG	46
Cursor Right	13	DISK	45
Cursor Left/Right	1A		
Cursor Up	10		
Cursor Down	11		
Cursor Up/Down	18		

The next four commands allow you to read the screen display, both text and graphics layers.

**ALLTEXT = 15h**

...requests all text in the K2vx's display.

**PARAMVALUE = 16h**

...requests the current parameter value.

**PARAMNAME = 17h**

...requests the current parameter name.

**GETGRAPHICS = 18h**

...requests the current graphics layer.

**SCREENREPLY = 19h**

This is the reply to ALLTEXT, PARAMVAL, PARAMNAME, GETGRAPHICS, or SCREENREPLY.

The reply to ALLTEXT will be 320 bytes of ASCII text (the display has 8 rows of 40 characters each). If you receive less than that, then the screen was in the middle of redrawing and you should request the display again.

The reply to PARAMVALUE will be a variable length ASCII text string. Some values (like keymaps, programs, samples, etc.) include their ID number in the text string (e.g., "983 OB Wave 1"). Some messages are also padded with extra spaces.

The reply to PARAMNAME will be a variable length ASCII text string. In cases where there is no parameter name (like on the program page) there will just be the single 00 null terminator.

The reply to GETGRAPHICS will be 2560 bytes of information. The 6 least significant bits of each byte indicate whether a pixel is on or off. If pixels are on over characters, the text becomes inverted. Characters on the K2vx display are a monospaced font with a height of 8 pixels and a width of 6 pixels.

# Chapter 31

## Glossary

<b>Algorithm</b>	In the K2vx, a preset configuration of programmable digital signal processing functions. Each of a program's layers uses its own algorithm, which determines the type of synthesis each layer uses to generate its sound.
<b>Aliasing</b>	A type of distortion that occurs in digitally sampled sounds when higher pitches (increased sample playback rates) introduce partials that were not present in the original sound. These partials may or may not be musically useful.
<b>Amplitude</b>	The intensity of a signal, perceived as loudness in the case of audio signals.
<b>Analog</b>	A term used widely in electronics-related fields to describe a method of representing information, in which the method of representation resembles the information itself. Analog synthesizers, for example, use gradual variations in electrical voltage to create and modify sounds. The oscillations in voltage are analogous to the waveforms of the sounds they generate. Compare Digital.
<b>Bandwidth</b>	In terms of sound generation, the range of frequencies within which a device functions. The human ear has a "bandwidth" of almost 20 KHz (it can distinguish sound at frequencies from 20 Hz to 20KHz). The K2vx's 20KHz bandwidth enables it to produce sounds that span the entire range of humanly audible sound.
<b>Bank</b>	There are two types of banks in the K2vx's memory: memory banks, which store and organize the programs and other objects you create, and Quick Access banks, where you can store programs and setups for one-button access while in Quick Access mode.
<b>Cent</b>	1/100th of a semitone. The standard increment for fine adjustment of pitch.
<b>Continuous control</b>	A device that converts motion into a range of 128 possible values that can modulate a sound source. The Mod Wheel, a standard volume pedal, and controllers like Breath and Aftertouch are continuous controls. Compare switch controls.
<b>Control Source</b>	Anything that can be used to modify some aspect of a program's sound. LFOs, envelopes, Mod Wheel messages (MIDI 01), and FUNs are just a few examples of the K2vx's control sources.
<b>DSP</b>	Digital signal processing (see)
<b>DSP Functions</b>	The K2vx's collection of digital signal processing functions are what give the Variable Architecture Synthesis system its flexibility. Within each layer's algorithm, you can select from a long list of DSP functions like filters, EQ, oscillators, and a few that are unique to the K2vx. Each DSP function has a corresponding page that enables you to assign numerous control sources to define how the DSP functions affect the sound of the program you're editing.
<b>Default</b>	The starting condition of a system. The settings for the K2vx's parameters are at their defaults when you unpack it, and they stay there until you change them. A hard reset will erase RAM and restore all parameters to their defaults.

- Dialog** A page that prompts you to enter information that the K2vx needs in order to execute an operation. Dialogs appear, for example, when you initiate a Save or Delete operation.
- Digital** A term used widely in electronics-related fields to describe a method of representing information as a series of binary digits (bits)—1s and 0s. Digital computers process these strings of 1s and 0s by converting them into an electrical signal that is always in one of two very definite states: “on” or “off.” This is much more precise than the analog method, therefore digital computers can operate at speeds unattainable by analog devices. Digital synthesizers like the K2vx are actually computers that process vast strings of digital information signals, eventually converting them (at the audio output) into the analog signals that flow into PAs and other audio systems. See also Analog.
- Digital Signal Processing** The term “Signal processing” refers to a vast range of functions, all of which have in common the fact that they act upon an electric current as it flows through a circuit or group of circuits. A simple form of signal processing is the distortion box used by many guitarists. *Digital* signal processing refers to similar processes that are performed by digital (see) circuitry as opposed to analog (see) circuitry. Many of the effects devices available today use digital signal processing techniques.
- Drum Program** The only difference between a drum program and an ordinary program is that a drum program can contain up to 32 layers instead of the usual maximum of three. Since each layer has its own keymap and algorithm (not to mention all the other control sources), this gives you enormous control over whatever sounds you assign to the layers in a drum program.
- Editor** The complete set of parameters used to modify a particular aspect of the K2vx, for example, the currently selected Program, which is modified with the Program Editor. The Program Editor spans several display pages, which can be viewed by using the soft buttons (the ones labeled “<more>.”
- Envelope** An aperiodic modifier. In other words, a way to cause a sound to change over time without repeating the change (unlike periodic modifiers like LFOs, which repeat at regular intervals).
- File** A group of objects stored to a floppy or hard disk, or loaded into the K2vx’s RAM from disk.
- Global** In this manual, used primarily in reference to control sources. A global control source affects all notes in a layer uniformly. If a layer uses a global control source, that control source begins to run as soon as the program containing it is selected. Its effect on each note will be completely in phase, regardless how many notes are being played. Compare Local.
- Hard Reset** Resets all parameter values to their defaults, and completely erases the contents of RAM. Press the Reset button in Master mode to do a hard reset. This is a quick way to restore the factory defaults to your K2vx, but EVERYTHING in RAM (all the objects you’ve created) will be erased, so objects you wish to keep should be saved to disk or SyxEx dump. A hard reset should not be used to recover if your K2vx is hung up, except as a last resort. See Soft Reset.
- Keymap** A keymap is a collection of samples assigned to specific notes and attack velocities. Keymaps usually contain numerous sample roots pitch-shifted across a range of several notes. When you trigger a note, the keymap tells the K2vx what sound to play, at what pitch, and at what loudness.

---

<b>LFO</b>	Low frequency oscillator. An oscillator is an electrical signal that cycles regularly between a minimum and maximum amplitude. The simplest oscillating waveform is the sine wave, but an LFO waveform can have almost any shape. The number of times each second that an oscillator repeats itself is called its frequency, which is measured in Hertz (Hz). Anything up to 50 Hz is considered low-frequency in musical applications. Use an LFO whenever you want to generate a <i>periodic</i> (repeating) effect. Adjusting the rate of the LFO will change the repetition rate of the effect.
<b>Layer</b>	A layer consists of a keymap processed through an algorithm. Layers can be stacked together within a program. Each layer uses one of the K2vx's 48 available voices. Each K2vx program can contain up to three layers—except drum channel programs, which can contain up to 32 layers.
<b>Leslie effect</b>	This classic vibrato effect was originally created by mounting a speaker in its cabinet so the speaker could be rotated at varying speeds. This applied a vibrato of varying rate to all sounds played through the rotating speaker.
<b>Local</b>	In this manual, used primarily in reference to control sources. A local control source affects each note in a layer independently. For example, if a local LFO is used as a control source, a separate LFO cycle will begin with each note start. The LFOs don't run in phase unless notes are started simultaneously. Compare Global.
<b>Memory banks</b>	The K2vx's memory is divided into ten spaces where you can store any object you edit. These spaces are called banks. Each bank can hold up to 100 objects of each type, so we refer to them as the 100s bank, the 200s bank, and so on. The ID of an object determines which bank it's stored in. An object with an ID of 399, for example, would be stored in the 300s bank. ROM objects are stored in the Zeros and 100s banks. RAM objects can be stored in any bank.
<b>MIDI</b>	Musical Instrument Digital Interface. A specialized format for representing musical information in terms of standardized computer data, which enables electronic musical instruments to communicate with computers
<b>MIDI device</b>	Any device—keyboard, computer, wind instrument, etc.—which is capable of transmitting and receiving MIDI messages.
<b>MIDI Master</b>	A MIDI device that is configured to control one or more other MIDI devices. The MIDI Out port of the master is connected by cable to the MIDI In port(s) of the slave device(s).
<b>MIDI Slave</b>	A MIDI device that is configured to receive MIDI messages from a master device. The MIDI In port of the slave is connected by cable to the MIDI Out port of the master.
<b>Non-linear DSP Function</b>	Without getting technical, non-linear DSP functions like SHAPER and WRAP add waveforms to those already present in a sound, while linear DSP functions act upon the existing waveforms without adding new ones.
<b>Note State</b>	Any K2vx note is either on or off; this is its note state. Normally, any given note's Note State switches on when you strike the key for that note. It switches off when you release the key, and any sustain controls you may have applied to the note (Sustain or Sostenuato pedal, etc.). Also see the index entry for Note State.
<b>Object</b>	A chunk of information stored in the K2vx's memory. Programs, setups, keymaps, and samples are all objects. There are several others as well. Also see the index entry for "Objects."

---

<b>Page</b>	A set of performance or programming parameters that appear as a group in the display. The entry level page for each mode appears when you select the mode. Most other pages are selected with the soft buttons, from within an editor.
<b>Parameter</b>	A programming feature. The name of the parameter describes the function it controls—transposition, for example. Each parameter has a value associated with it, which indicates the status of the parameter.
<b>Pixel</b>	A contraction of “picture element.” The K2vx’s display consists of a screen with small square dots (the pixels). Each pixel lets light through or blocks it depending on whether it is receiving an electrical charge. The combination of light and dark dots creates a pattern that you recognize as text or graphics. The K2vx’s display is 240-by-64 pixels, in other words, 64 horizontal rows, each containing 240 pixels, for a total of 15360 pixels.
<b>Program</b>	The K2vx’s basic performance-level sound object. Programs can consist of up to 3 layers (32 layers for programs on the drum channel); each layer has its own keymap (set of samples) and sound-processing algorithm.
<b>Program Editor</b>	The set of parameters that lets you modify the sound of ROM or RAM programs. Enter the Program Editor by pressing the EDIT button while in Program mode, or any time the currently selected parameter has a program as its value.
<b>RAM</b>	Random Access Memory, one of the two basic types of computer memory. RAM can be both read from and written to. When you load samples into the K2vx, or save a program you’ve created, you’re writing to RAM. Compare ROM.
<b>ROM</b>	Read Only Memory, one of the two basic types of computer memory. You can retrieve the information stored in ROM, but you can’t write (save) new information to it. The onboard sounds of your K2vx are stored in ROM.
<b>Sample</b>	A digital recording of a sound that can be assigned to a keymap as part of the process of building a program. Samples are stored in ROM (factory-installed) or in RAM (loaded from disk).
<b>SCSI</b>	Pronounced “scuzzy,” this acronym stands for Small Computer Systems Interface. It’s simply a standardized form of information exchange that allows any SCSI equipped device to communicate with any other SCSI device. Two or more SCSI devices—they can be computers, hard disks, printers, just about anything that sends or receives information in standardized form—are connected via special cables to their SCSI ports. This configuration is much faster than serial information exchange, the precursor to SCSI.
<b>SMDI</b>	Pronounced “smiddy,” this acronym stands for SCSI Musical Data Interchange. It’s a new format for data transfer, based on the SCSI format, which uses parallel input/output rather than serial, as used by MIDI and standard SCSI operations. This enables data to flow much faster. You can use SMDI to transfer samples to and from the K2vx using software packages from Passport and Opcode.
<b>Semitone</b>	In “Western” music, the standard interval between the twelve notes in the scale. There are twelve semitones to an octave. The interval between C and C# is one semitone.
<b>Setup</b>	A multi-timbral performance object. A setup consists of three zones, each of which can be assigned its own program, MIDI channel, and control assignments. These assignments control the K2vx’s operation while in Setup mode, as well as determining the Program Change numbers and controller messages the K2vx sends via MIDI.

- Soft Reset** Returns the K2vx to Program mode without affecting the contents of RAM. Press the +/-, 0, and CLEAR buttons to do a soft reset. If your K2vx is hung up for some reason, this will usually get take care of the problem. See Hard Reset.
- Switch control** A device that converts motion into discrete on/off signals. A switch control, like the Sustain pedal, is either on or off. Compare continuous control.
- Toggle** As a verb, to switch between (usually) two conditions using a device that makes the switch. As a noun, the device that makes the switch. For example, pressing the "View" soft button on the top level Program mode page toggles between small-type and large-type views of the current Program.
- Value** The current setting of a parameter. Each parameter has a range of available values, one of which you select while editing. The Transposition parameter on the Program mode page, for example, has a default value of 0. Change the value to change the parameter's effect on the current program.
- Variable Architecture Synthesis Technology (V. A. S. T.)** The term created by Kurzweil engineers to describe the multi-faceted capabilities of the K2vx, combining sample playback (ROM and RAM), and waveform generation with a broad array of processing functions. This architecture provides preset algorithms created by Kurzweil sound engineers, which include filters, distortion, panning, EQ, waveform oscillators, waveform shaper, hard sync oscillators, amplitude modulation, gain, crossfade, and more. V. A. S. T. is a registered trademark of Young Chang Akki Co. Ltd.
- Zero Crossing** Any of a number points in the digital representation of a sound's waveform where the digital signal is neither positive or negative. When looping samples, starting the loop at one of these points will reduce or eliminate the click or change in timbre that can occur in sample loops.





## Chapter 32

# Specifications

### K2vx FEATURES

- 240 x 64-pixel backlit graphic display with adjustable contrast
- 3.5-inch floppy disk drive, for DD or HD disks, DOS compatible
- MIDI In, Thru, and Out
- MIDI LED to indicate MIDI activity
  
- 24-note polyphony with dynamic voice allocation
- Multi-timbral, for multi-track sequencing and recording
- 199 factory preset programs, and 100 factory preset setups
- Up to 3 layers per program, up to 32 layers for programs on drum channels
- Receives mono (channel) pressure and poly (key) pressure
- 3-zone setups transmit on 3 MIDI channels with independent programmable controls
- Fully featured onboard sequencer for recording from keyboard or via MIDI; loads and plays MIDI Type 0 sequences
- Easy-to-use programming interface including soft buttons, Alpha Wheel, and alphanumeric pad
  
- 8 Megabytes of 16-bit sample ROM, including acoustic instrumental sounds, waveforms, and noise
- 20 KHz maximum bandwidth
- Optional stereo sampler with analog and digital inputs
- Sound ROM expandable to a total of 24 Megabytes
- 4 SIMM sockets for optional sample RAM—up to 64 Megabytes
- Stereo sample playback capability
- Akai® S1000 sample disk compatibility
  
- Two 1/4-inch mixed audio outputs (stereo pair)
- Four 1/4-inch audio outputs programmable as two stereo pairs or as four separate outputs, with insert capability for effects patching
- Stereo headphone jack
  
- 128K battery-backed RAM for user programs, setups and other objects, expandable to 750K
- Two SCSI ports for connection with external SCSI disks, CD-ROM drives, or Macintosh® personal computers
- Optional internal hard disk
  
- Realtime DSP for each voice: 31 programmable DSP algorithms incorporating filters, EQ, distortion, panning, pulse width modulation, and more; up to 3 programmable DSP functions per voice
- Filters: Lowpass, Highpass, Allpass, Bandpass, Notch, programmable resonance
- Programmable stereo multi-effects on MIX outputs, including simultaneous reverb, chorus, delay, flanging, EQ—and more

## Specifications

---

### K2vx FEATURES

- Realtime internal and MIDI control of effects parameters
- MIDI standard sample dump/load capability
- SMDI sample dump/load capability
- System Exclusive implementation
- MIDIScope™ for analyzing MIDI events

## Environmental Specifications

### Temperature ranges

For operation:	minimum	41° F (5° C)
	maximum	104° F (40° C)
For storage:	minimum	-13° F (-25° C)
	maximum	186° F (85° C)

### Relative humidity ranges (non-condensing)

Operation and storage:	5—95%
------------------------	-------

## Physical Specifications

### Overall dimensions

Width	16.9 in*	43 cm
Depth	13.9 in	35.4 cm
Height	5.1 in	13 cm
Weight:	24.65 lb	11.2 kg

\* Excluding the rack-mount brackets

## Electrical Specifications

AC supply: selectable; 100V, 120V, 220V, or 240V. 1.0 amps at 120 volts nominal

### Safe voltage ranges

Voltage setting:	100V	120V	220V	240V
Safe voltage range:	85—107	95—125	180—232	190—250
Safe frequency range:	48—65	48—65	48—65	48—65

If the voltage drops below the minimum safe level at any voltage setting, the K2vx will reset, but no data will be lost. If the voltage exceeds the maximum safe level, the K2vx may overheat.

**Specifications**

MIDI Implementation Chart

# MIDI Implementation Chart

**Model: K2vx**

**Manufacturer:  
Young Chang**

**Date: 3/21/95  
Version 1.0**

**Digital Synthesizers**

Function	Transmitted	Recognized	Remarks			
<b>Basic Channel</b>	Default	1	1	Memorized		
	Changed	1 - 16	1 - 16			
<b>Mode</b>	Default	Mode 3	Mode 3	Use Multi mode for multi-timbral applications		
	Messages Altered					
<b>Note Number</b>			0 - 127	0-11 sets intonation Key		
	True Voice	0 - 127	0 - 127			
<b>Velocity</b>	Note ON	O	O			
	Note OFF	O	O			
<b>After Touch</b>	Keys	X	O			
	Channels	O	O			
<b>Pitch Bender</b>		O	O			
<b>Control Change</b>	O	0 - 31 32 - 63 (LSB) 64 - 127	O	0 - 31 32 - 63 (LSB) 64 - 127	Controller assignments are programmable	
<b>Program Change</b>		O	1 - 999	O	1 - 999	Standard and custom formats
	True #		0 - 127		0 - 127	
<b>System Exclusive</b>		O	O*			
<b>System Common</b>	Song Pos.	O	O			
	Song Sel.	O	O			
	Tune	X	X			
<b>System Real Time</b>	Clock	O	O			
	Messages	O	O			
<b>Aux Messages</b>	Local Control	O	O			
	All Notes Off	O	O			
	Active Sense	X	X			
	Reset	X	X			
Notes	*Manufacturer's ID = 07 Device ID: default = 0; programmable 0 - 127					

Mode 1: Omni On, Poly  
Mode 3: Omni Off, Poly

Mode 2: Omni On, Mono  
Mode 4: Omni Off, Mono

O = yes  
X =no

# Appendix A

## K2vx Program Farm

The K2vx Program Farm, included on one of the accessory disks, provides you with a large collection of additional programs. You can either use these programs as they are, or edit them further to suit your own tastes.

### Overview of Program Files

#### **ANACOMPS.K25**

*Analog Comping Sounds*

- Prophets
- PWM Synth
- Saws w/ Resonance
- Sawtooth
- Synth Brass
- Vintage Synths
- Mellotrons

#### **ANALEADS.K25**

*Analog Lead Sounds*

- Fusion
- Intervals
- Minimoog
- Pseudamentos
- Rock Leads
- Syncro Sounds

#### **ANAPADS.K25**

*Analog Pads*

- Analog String Pads
- Classic Pads
- Philtered Pads
- Soft Sawtooth Pads

#### **BASS.K25**

*Bass*

- Dual Basses
- Picked or Fingered Basses
- Slap Basses
- Stylized Basses
- Analog Synth Basses
- House Basses

#### **BELLS.K25**

*Bell Sounds*

- Bell Pads
- Percussive Bell Comps

#### **BRASS.K25**

*Brass & Winds*

- Solo Trumpets
- Muted Trumpet Emulations
- Solo Trombones
- Bass Horn Emulations
- Solo Flutes
- Piccolo Emulations
- Sax Emulations
- Sax Sections
- Brass Sections
- Brass and Strings

#### **DIGITAL.K25**

*Digital Sounds*

- Digital Leads
- AM Sounds
- DigiBuzz
- Digital Clavs
- DigiDistort
- PPG's
- Miscellaneous Digital Sounds

#### **DKICKSNR.K25**

*Individual Kicks And Snares*

- Kicks
- Snares

#### **DRUMS.K25**

*Drum Kits And Percussion Sounds*

- Techno sounds
- Percussion
- Drums Kits
- Toms
- Mallet Sounds
- Multi Taps

#### **ENSEMBLE.K25**

*Acoustic Ensembles*

- Orchestral Ensembles
- Layered w/ Guitar
- Jazz and Rock Combos

#### **ETHEREAL.K25**

*Ethereal*

- Stepped Pads
- Vocal Pads
- Air Pads
- Glassy Pads
- Pads w/ Partial
- Hybrid Strings and/or Vocal Pads
- Generic Ethereal Sounds

#### **FXSOUNDS.K25**

*Effects Sounds/Textures*

- Sawtalk
- Water Sounds
- Weather Emulations
- Weird Percussive Things
- Natureal Sounds
- Strange Loops

## **GUITARS.K25**

### *Guitars*

- Steel String Guitars
- 12-String Guitars
- Stylized Steel String Guitars
- Guitars w/ Strings or Pads
- Combination Guitars
- Jazz Guitars
- Guitar Mutes
- Distorted Emulations
- Exotic Plucked Things

## **HYBPERC.K25**

### *Percussive Hybrids*

- Percussive Comps
- Percussive Pads

## **HYBRIDS.K25**

### *General Hybrid Sounds*

- Hybrid String Pads
- Hybrid String Pads w/ Resonance
- Hybrid Stacks
- Flute-Like Hybrids

## **ORGANS.K25**

### *Organs*

- Clean Rock Organs
- Clean Percussive Organs
- Pipe Organs
- Velocity Sensitive Organs
- Organ-Like Synths

## **PNOEPNO.K25**

### *Piano & E Pianos*

- Classical Pianos
- Rock Pianos
- Pianos w/ Strings or Choir
- Bright and House Pianos
- Stylized Pianos
- Electric Grands
- Rhodes Pianos
- Dyno Pianos
- FM Pianos
- Clavinet Emulations
- Harpichord Emulations

## **STRINGS.K25**

### *Strings*

- Straight Strings
- Attack Velocity Strings
- Sfz Strings
- Ensemble Strings
- Layered Solo and Ensemble Strings
- Solo Violin and Cello
- Pizz Emulations
- String Pads

## **VOX.K25**

### *Vocal Sounds*

## **Programs Within the Files**

The complete list of programs in the K2vx Program Farm is provided on the following pages. Each program is numbered as if it were loaded into the 200's bank.

## ANACOMPS.K25

Analog Comping Sounds  
(88 Programs)

### Prophets

200-Prophet Clav  
201-Prophet Clav #3  
202-Prophet Disco  
203-Prophet Disco 2  
204-Prophet Fuzz  
205-Prophet Fuzz #2  
206-Prophet Piano 1  
207-Prophet Pulse 1  
208-Prophet Pulse 2  
209-Prophet Pulse 3  
210-Prophet Pulse 4  
211-Prophet Pulse 5  
212-Prophet PWM Clav  
213-Prophet Square 1  
214-Prophet Square 2  
215-Prophet Square 3  
216-Prophet Square 4  
217-Prophet phase  
218-Brightsaw Prophet  
219-Neo-Prophet  
220-Mellow Prophet  
221-70's Synth

### PWM Synth

222-Big Mondo PWM  
223-Big PWM  
224-Big PWM #2  
225-Big PWM #5  
226-Env PWM  
227-PWM Ags  
228-Journey Synth

### Saws w/ Resonance

229-Ana's Saws  
230-OB-8 1  
231-Obyrez #1  
232-R&B Synth  
233-Clean Sweep  
234-Alaska  
235-Analog Saw Pad  
236-Poly Analog 1  
237-Poly Analog 2  
238-Poly Analog 5th  
239-Bright and Fat  
240-Saw Teeth  
241- 9 Osc Sweep  
242-Big Synth #1  
243-Saw Repeater

### Sawtooth

244-Sawz  
245-Big Jupiter  
246-Dry Pluck Pad

247-o b Waves  
248-Polysynth  
249-Thick Osc  
250-Oh Bee !!!

### Synth Brass

251-OBX Braz #1  
252-OBX Braz #2a  
253-OBX Braz #4  
254-Synth Brass  
255-Real OBX Braz  
256-Synbrass Sect  
257-Synth Brass 1  
258-Synth Brass 2  
259-Mellow Synbrass  
260-Dyno Synbrass  
261-Analog Brazz

### Vintage Synths

262-Memorymoog #4  
263-Memorymoog Split  
264-Poly 2600 #1  
265-Poly ARP #1  
266-Poly ARP #2  
267-Elka 4  
268-Elka II  
269-Univox Piano #2  
270-Univox Piano #3  
271-Matrix 12  
272-Matrix 12 2  
273-Matrix 12 House  
274-Matrix 12 Rez 5  
275-Matrix 12 Soft 5

### Mellotrons

276-Chiffatron  
277-Mellotron  
278-Mellotron 1  
279-Mellotron 2  
280-Mellotron 3  
281-Tron Flutes  
282-Tron Flutes 2  
283-Tron Vox  
284-Tron Vox 2  
285-Tron Tapes  
286-Tron Tapes 2  
287-Tron Strings

## ANALEADS.K25

### Analog Lead Sounds

(73 Programs)

#### **Fusion**

200-Chick Lead 1  
201-Chick Lick  
202-Duke Lead 1  
203-Duke Lead 2  
204-Duke's Lead 3  
205-Control Formants  
206-Ewi Lead  
207-Ewi Lead 2 (CS)  
208-Porta Lead (CS)  
209-Ballad Lead

#### **Intervals**

210-Wakeman #2  
211-Wakeman #3  
212-Duo Synth  
213-Zawinul 2600  
214-Zawinul 2600 #2  
215-Quadra #1  
216-Quadra #2  
217-Quadra #3  
218-Maj/Min Break  
219-Dr. Dre Lead  
220-Lonely 5ths

#### **Minimoog**

221-Mini Lead #1  
222-Mini Lead #2  
223-Mini Lead #3  
224-Mini Lead #4  
225-Mini Lead #5  
226-Mini Lead #6  
227-Mini Lead #7  
228-Mini Lead #8  
229-Poly Mini  
230-Mini 2  
231-Poly Mini 2

#### **Pseudamentos**

232-PseudamentoCS  
233-Pseudamento 2  
234-Pseudamento 3  
235-Pseudamento 5  
236-Pseudamento 6  
237-Pseudamento 7  
238-MonoLead 1  
239-PRS Wow Lead

#### **Rock Leads**

240-Rich Raunch 1  
241-Rich Raunch 2  
242-Wakeman  
243-Wakeman#1  
244-Wakeman#4  
245-JR Lead 2  
246-Later Emerson  
247-DC Lead

248-Modular Lead  
249-Lucky Lead  
250-Mono Brass  
251-OB Brass  
252-SynBass Lead PRS  
253-Clock S&H Lead  
254-Fun Delay Square  
255-Square Pad  
256-TimsDukeyDick 1  
257-TimsDukeyDick 2  
258-AlaZawinul  
259-Lead Program  
260-Lead Program 2  
261-RezoLead Program  
262-Skinny Lead  
**Syncro Sounds**  
263-Car's Sync  
264-Car's Sync 2  
265-Classic Sync  
266-Sync  
267-Sync It !  
268-Prophet Sync 2  
269-Yo Mama !  
270-Velocitync 2  
271-Prophet Sync  
272-Velocitync 3



## ANAPADS.K25

Analog Pads

(47 Programs)

### **Analog String Pads**

200-Orch Pad 4  
201-Lore Pad  
202-MatrixMelostring  
203-String Like 2  
204-Eerie Synth Pad  
205-Resonator  
206-ARP Omni  
207-String Machine 1  
208-Synstring  
209-Mixpad

### **Classic Pads**

210-Mello Analog  
211-Oberheim Pad  
212-Prophet Pad  
213-Memorymoog #2  
214-Mister Softy  
215-OB-8 Pad  
216-Slo Generic OBX  
217-Prophet 5 Mello  
218-Prophet 5 Mello5  
219-Slo PWM  
220-Soft PWM  
221-Soft PWM 7

### **Philtered Pads**

222-Slow Filter  
223-JB Synth Pad  
224-Sawphaze  
225-Sawphaze 7th  
226-Solina Phaze  
227-Solina Phaze 2  
228-Solina Phaze 7th  
229-Sweep Pad  
230-AM Square Synth  
231-Matrix 12 Pad  
232-Matrix 12 Pad 5  
233-Matrix 12 Pad 5 II  
234-Slo SawPad  
235-Pad 5th

### **Soft Sawtooth Pads**

236-Another Pad  
237-Bag Slush Pod  
238-Big Lush Pad  
239-Slo Lush Pad  
240-Lo Pad 7 Split  
241-Matrix SoftPad  
242-Matrix SoftPad 5  
243-In the Air  
244-Ride Sweep  
245-Pulsar  
246-Scanners

## BASS.K25

Bass

(61 Programs)

### Dual Basses

200-Dual E Bass 1  
201-Dual E Bass 2  
202-Warm Dual Bass

### Picked or Fingered Basses

203-Sustain EBass 2  
204-Fingered Bass  
205-Fingered Bass 2  
206-Picked Bass  
207-Finger Bass  
208-Warm Bass  
209-E Bass & Ride

210-Ripper

### Slap Basses

211-Slap Bass 1  
212-Slap E Bass Prs  
213-Funk Me Bass

### Stylized Basses

214-Hammer Bass  
215-Too Bad Bass  
216-Walking Bass  
217-No Frets

### Analog Synth Basses

218-Big Lo Bass  
219-Big Mono Bass  
220-2 Big Basses  
221-Bass & Lead  
222-Bass & Rhodes  
223-Dubb Bass  
224-Mooger Bass  
225-Big Res Bass  
226-AnalogBass D  
227-Saw Bass  
228-Synth Bass 1  
229-WonderSynth Bass  
230-Mondo Bass  
231-Mogue Bass 2  
232-Mogue Bass 3  
233-Doom Bass  
234-Matrix Big Bass  
235-Matrix BigBass 3  
236-Moogy Bass #1  
237-Moogy Bass #2  
238-Moogy Bass #3  
239-Moogy Bass #4  
240-Moogy Bass #5  
241-Moogy Bass #6  
242-Moogy Bass #8  
243-ProphetPulsBass 2  
244-ProphetPulsBass 3  
245-ProphetPulseBass  
246-Synth Bass 2  
247-Synth Bass 3

248-Tuch Bass

249-Tuch Bass 2

### House Basses

250-Unison House  
251-Pop Attack Bass  
252-Slap House Bass  
253-AnaHouse Bass  
254-House Bass #3  
255-House Bass #4  
256-House Bass #5  
257-House Bass #6  
258-House Bass #7  
259-Low BigBass 3  
260-Low BigBass 4

## BELLS.K25

Bell Sounds

(28 Programs)

**Bell Pads**

- 200-Larabell
- 201-Space Bell 1
- 202-Metallic Pad
- 203-Fantasia
- 204-Space Bell 6
- 205-d50 Voce 2
- 206-Leningrad
- 207-With Tinklers
- 208-Lullaby
- 209-Tranquility
- 210-Tranquil Bell
- 211-PM'S Bell Pad
- 212-Digiphaze
- 213-Blistener
- 214-Simmbell
- 215-Bell Tree > Big Bell

**Percussive Bell Comps**

- 216-Crystal
- 217-DigiBell
- 218-Wave Bells
- 219-Little Metals
- 220-Clangorous
- 221-Gong Layers
- 222-Mallet Pad
- 223-Clockbells
- 224-Balarimba
- 225-Pinger
- 226-Belles
- 227-Toy Box

## BRASS.K25

### Brass & Winds

(68 Programs)

#### **Solo Trumpets**

200-Dynamic Trumpet  
201-Solo trp mw vib  
202-Miles Unmuted  
203-Gentle trumpet

#### **Muted Trumpet Emulations**

204-Strght Mute Trpt  
205-Muted trumpet 2  
206-Sfz "mute" trp  
207-20's Trumpet  
208-Almost Muted

#### **Solo Trombones**

209-Trombone  
210-Solo Trombone  
211-Sfz Bone

#### **Bass Horn Emulations**

212-Tuba  
213-Solo Bass Horn

#### **Solo Flutes**

214-Legato Flute  
215-Legato Flute(prs)  
216-Orchestral Flute  
217-fast orch flute  
218-wendy's Flute 2  
219-Treble Flute  
220-Jethro's Flute  
221-JethroFlute(prs)  
222-Baroque Flute 1  
223-Baroque Flute 2  
224-EchoFlute

#### **Piccolo Emulations**

225-Piccolo  
226-Orch Piccolo 3

#### **Sax Emulations**

227-Tenor Sax  
228-Kenny's Tenor  
229-StreetCorner Sax  
230-Get Real Bari

#### **Sax Sections**

231-Fake Sax  
232-Hybrid Sax  
233-Reed Section  
234-Sax Section 8ve 1  
235-Section Sax 2

#### **Brass Sections**

236-Tijuana Brass  
237-Trp Section 1  
238-Trps & Bones  
239-Bri Trp & Bone  
240-Trumpet & Bone  
241-Hip Brass  
242-Brass Section  
243-Brass Band

244-Dyn Hi Brass B  
245-Dyn Lo Brass B  
246-Dyn Tbn & Hrn  
247-Huge Brass 2  
248-Huge Brass Too  
249-SoftLowBrass B  
250-Hall Horns  
251-Mello Orch Brass  
252-Chorale Brass 4  
253-Soft Section 1  
254-Soft BigBand 2  
255-Big Band 3  
256-Big Band 4  
260-Trombone Section  
261-Braz Sect  
262-Orchestral Brass  
263-SoftLowBrass  
264-Spit Brass  
**Brass and Strings**  
265-Stacc. Brass & St  
266-Brass & Strings  
267-Huge Brass

**DIGITAL.K25**

## Digital Sounds

(63 Programs)

**Digital Leads**

200-Carrie or Maud

201-FM Guitar 1

202-FM Guitar 2

203-FM Guitar 3

204-FM Guitar 5

205-FM Harmonica

206-FM Lead

207-Funfare Leadelay

208-Monolead 1

**AM Sounds**

209-Shape Mod Rules !

210-VS-Type

211-Wave Table II

212-Velveteen

213-Digi Strange

**DigiBuzz**

214-Buzz-a-Loo Too

215-Buzz-a-Loo

216-Digi Wet 2

217-The Buzz

218-Hybrid Sweep

219-Buzz Ofe

220-Cycler

221-Fuzz Lite 2

222-Growl

223-Cycler 2

224-Farr Feesah

225-Rezzysteppy

226-Whipstep

227-Deetuara II

228-Shap Mod Oscar

229-Backwards 2

230-Backwards 3

231-Backwards 4

**Digital Clavs**

232-Klikomp too

233-Klikomp tree

234-CS Clav

235-Talking Clav 3

236-Klav Ennette

237-Shape 3

238-Shape 1

**DigiDistort**

239-New Shaper

240-RAD wave 2

241-Buzz Slap

242-Rizzak

243-Chorusar II

244-Lowteeth

245-Digi Power

246-Microwave 2

247-Insect Klav Rise

248-Insectrise

249-FuzzFall

250-EP Lead

251-Ep Lead Too

**PPG's**

252-PPG 9

253-Slo PPG 10

254-PPG 1

255-PPG 2

256-PPG 4

257-PPG 8

258-Shape 2

259-Shape 4

**Miscellaneous Digital Sounds**

260-Cricketar

261-Piano Ring

262-StringBell

## DKICKSNR.K25

Individual Kick And Snare Programs

(95 Programs)

*These are single layer kicks and snares ready to be imported into your own custom drum programs. Programs whose names are in all capital letters are the unmodified kick and snare samples.*

### Kick Drums

200-AMB KICK 1  
201-Tight Amb Kick 1  
202-Sharp Amb Kick 1  
203-AMB KICK 2  
204-Room Kick Drum 2  
205-Big Kick Drum 2  
206-Dyn Kick Drum 2  
207-Gate Kick Drum 2  
208-Rock Kick Drum 2  
209-Amb Kick DR 2eKT  
210-Crack Kick 2  
211-AMB KICK 3  
212-Fat Kick Drum 3  
213-Pad Kick Drum 3  
214-DRY KICK 1  
215-Beater Kick Drum  
216-Soft Kick Drum1  
217-Low Kick 1  
218-Dance Kick 1  
219-Dead Kick 1  
220-DRY KICK 2  
221-HighnDry Kick Dr  
222-SoftnDry Kick Dr  
223-Sub Kick Drum 2  
224-Noise Kick Drum  
225-High'n Low Kick  
226-Noisy Kick 2  
227-Fried Kick 2f  
228-Jazz Kick Drum 1  
229-Jazz Kick Drum 2  
230-Techno Kick 1  
231-little kick  
232-Klik Kick  
233-Cut Kick  
**SNARES**  
234-Kick Drums  
235-AMB SNARE 1  
236-Rock Snare 1  
237-Pop Snare  
238-Dance Snare  
239-Sharp Snare  
240-AMB SNARE 2  
241-Big Snare 2

242-Short Snare 2  
243-Deep Cut Snare 2  
244-Low Snare 2  
245-AMB SNARE 3  
246-Big Hall Snare 3  
247-Metal Snare 3  
248-High Snare 3  
249-Hard Snare 3  
250-sLow Snare 3  
251-Tight Snare 3  
252-Gated Snare 3  
253-Amb/Dry Snare MW  
254-DRY SNARE 1SOFT  
255-Short Soft Snare  
256-Dyn Dry Snare 1  
257-Deep Dry Snare 1  
258-DRY SNARE 1HARD  
259-Hard Attk DrySna  
260-Cracked Dry Snar  
261-Snappy Dry Snare  
262-DRY SNARE 2  
263-Open Snare 2  
264-PunchySnare 2  
265-Studio Snare2  
266-DRY SNARE 3  
267-Ringer Snare  
268-Roll Snare  
269-Harm Snare3  
270-Dual Dry Snare 1  
271-Dual Dry Snare 2  
272-Dual Dry Snare 3  
273-Snare Drum 1  
274-Snare Drum 2  
275-Brush 1  
276-Brush 2  
277-Brush 3  
278-Deep Brush  
279-Hard Brush  
280-Sharp Brush  
281-Harmful Brush  
282-Techno Snare 1  
283-Techno Snare 2  
284-Res Snare  
285-Snare w/Ring  
286-Res Snare 2  
287-Deep Snare  
288-Deep Snare 2  
289-Deep Snare 3  
290-Deep Snare 4  
291-Deep Snare 5  
292-Dual Deep Snare  
293-Ringy Snare  
294-Snare Program

**DRUMS.K25**

## Percussion Sounds

(83 Programs)

**Techno Sounds**

200-CR 78 III  
 201-CR 78 J  
 202-CR 78 Kick  
 203-CR 78 Snare 2  
 204-CR 78 Hat  
 205-CR 78 Hat 2 o  
 206-CR 78 Hat o/c  
 207-CR 78 Splish  
 208-Fake Snare  
 209-Fake Toms  
 210-Fake Toms 2  
 211-Fake Hat  
 212-Fake Hat Open  
 213-Fake Hat o/c  
 214-Fake Splash  
 215-Rezit Klik  
 216-Rezit Tick  
 217-Rezit Klave  
 218-Rezit Bongos  
 219-Rezit Guiros  
 220-Rezit Guiros 2  
 221-Rezit Guiros 3  
 222-Rezit Sidestick  
 223-Kowbell 1  
 224-Kowbell 2  
 225-Kowbell 3  
 226-Kewbell 1  
 227-Kipbell 1  
 228-Tekno Kick  
 229-Triangle 1  
 230-Triangle 2  
 231-808 Toms  
 232-Noised  
 233-Noised Sweep  
 234-Zing 1  
 235-Zing 2  
 236-Thwick 1  
 237-Klap 1  
 238-Klap 2  
 239-Simmons  
 240-Synth Tom

**Percussion**

241-Log Drum 2  
 242-Shaker Thing  
 243-High Shaker  
 244-Native Drum  
 245-Dyn Perc  
 246-5 Drums Low

**Drum Kits**

247-Dyn Snare Kit  
 248-LightAmb DynKit  
 249-New Dance Kit

250-J Bottham  
 251-1 Layer dry kit  
 252-1 Layer dry kit2  
 253-1 Layer Amb kit1  
 254-1 Layer Amb kit2  
 255-1 Layer Amb kit3  
 256-Drums Program 1  
 257-Drums Program 2  
 258-Drums Program 3  
 259-Drums Program 4  
 260-Drums Program 5

**Toms**

261-Toms 1  
 262-Toms 2  
 263-Toms 3

**Mallet Sounds**

264-Malletone  
 265-Wood Bars  
 266-Metal Bars  
 267-Glockenspiel  
 268-Tine Mariba  
 269-Marimba Vibe  
 270-JARO bell Ens  
 271-JARO bell Ens 3  
 272-Cym Roll!  
 273-Gong Release  
 274-Cym Roll 2 Cmplx

**Multi Taps**

275-TouchmTones  
 276-Snappy JR  
 277-Perky Lizards  
 278-Touchy Rezoid  
 279-Killamon-Jaro  
 280-STEP-OOO-DOO  
 281-Multipercs  
 282-Killamon-Jaro 2

## ENSEMBLE.K25

Acoustic Ensembles

(21 Programs)

**Orchestral Ensembles**

200-Flute & Slo str  
201-Horn&Flute w /Str  
202-Winds&Strings 2  
203-W Tell Orchestra  
204-Touch Orchestra  
205-Orch Hit  
206-Slo Ensemble  
207-Mello Slo Ens  
208-Voice w/ upper Str  
209-ChoirStrings 1  
210-St Choir&Strings  
211-Syn Orch Winds  
212-Syn Orch Pad

**Layered w/ Guitar**

213- 12 Str Rhodes  
214-Williamsong  
215-40 Something  
216-Guitar / Flute  
217-Piano / Cello

**Jazz and Rock Combos**

218-A.Bass&Ride/ Piano  
219-Jazz Club  
220-Trio 1



## ETHEREAL.K25

### Ethereal Sounds

(74 Programs)

#### Stepped Pads

200-Aurora Part 2  
 201-Choir Jumps  
 202-Hipass Pad 6  
 203-Putthings  
 204-Smoothings  
 205-Stutterer  
 206-Timbre Steps  
 207-Solar System  
 208-Solar System 2  
 209-Solar System 3  
 210-Space Moves  
 211-Time Traveler 2  
 212-Bell Steps  
 213-Spac'd 1  
 214-Spac'd 2  
 215-Star Theme  
 216-Stringer W/01  
 217-Pair o' Pads  
 218-Soft fm 2  
 219-Wind Vox  
 220-Shimmerling  
 221-Stereo Sweeps  
 222-Multi-Texture  
 223-Touch Down  
 224-Choir Things  
 225-Heavens Voxx  
 226-Tinglethings  
 227-Mallet Choir 3

#### Vocal Pads

228-World's Order 1  
 229-World's Order 4  
 230-Angelia  
 231-DreamVox 2  
 232-Low World Vox  
 233-Enya Vocal Pad  
 234-Slo Syn Pad  
 235-DreamVox

#### Air Pads

236-Sisternal II  
 237-Sisternal 3  
 238-SloSynPad 2 (CS)  
 239-Snare Thing Pad  
 240-Flatliners  
 241-Fair Breath  
 242-Flutevox  
 243-Flutevox 2  
 244-Passion Source 2  
 245-Passion Base  
 246-Launch Pad 2

#### Glassy Pads

247-Waterphone  
 248-Glass Bow 2

249-Glasswaves  
 250-Glassy Eyes  
 251-Cycle 2  
 252-Harmonic Synth  
 253-Aliens 2

#### Pads w/ Partial

254-SloHarm  
 255-Slo HiHarm  
 256-Slo Vox Formant  
 257-Vectoring  
 258-Vectoring 2  
 259-Vectoring 3  
 260-Syn Tambura

#### Hybrid Strings and/or Vocal Pads

261-LoWorld Shift 2  
 262- Odyssey  
 263-Deep Atmosphere  
 264-Lush Life 1  
 265-Lush Life 2  
 266-Lovershift  
 267-PM's Lead Pad  
 268-PressFor Thunder!

#### Generic Ethereal Sounds

269-Bone Thing  
 270-Disaster  
 271-Aliens Voice  
 272-Space Notch 2  
 273-Launch Pad Water

## FXSOUNDS.K25

Effects Sounds/Textures

(47 Programs)

### **Sawtalk**

200-Fun Program

201-Fun 2

202-Hello 2 b

203-Talk Talk

### **Water Sounds**

204-Noise PWM Qnirp

205-NoizFalls

206-NoizFalls 2

207-Sub Space

208-Wavionics

### **Weather Emulations**

209-Winds 2

210-Downpour

211-Press Wind

212-Thunder/Rain

213-Thunder 3

### **Weird Percussive Things**

214-Aliens

215-Crashear

216-Pell ShakThing 2

217-Pell Thing 3

218-What!

### **Natureal Sounds**

219-Criks

220-Sinebird

221-Sinebird 2

222-Sineforest

223-RainforestCrunch

224-Chirps

### **Strange Loops**

225-Speilbergs

226-Captain Nemo

227-Meow Scratch

228-Qnirp

229-Shape lfo 1

230-Shape lfo 2

231-Shape lfo 3

232-Shape lfo MW

233-Xylo Lore

234-Notreallyrandom

235-Subotnick (CS)

236-' ndustry 2

237-The Night Shift

238-Ffich

239-Strike 2nd

240-Slider Spaceout

241-DeathToTheVoices

242-Freddy's Hands

243-Slay Bells

244-Slay Drum

245-Dream State 2

246-Con Ed

## GUITARS.K25

### Guitars

(73 Programs)

#### Steel String Guitars

200-Acoustic Guitar

201-Steel Str Guitar

202-Steel Str Guitar 2

#### 12-String Guitars

203-12-str Guitar 1

204-12-str Guitar 2

#### Stylized Steel String Guitars

205-Rich Guitar

206-Fluid Guitar

207-Fluid Guitar 2

208-Magic Guitar

209-Meditator 4

210-Atmosphere

211-Modern Harp

212-Sweetar

213-Steel String

214-12 String

215-Hybrid Guitar

216-Pluxichord 2

217-Clean Guitar

#### Guitars w/ Strings or Pads

218-AcGtr&StrPad

219-AcGtr&Strings 2

220-Para Gtr w/ Voice

221-Mod Lag City

222-Heaven Guitar

223-Oto Pad

#### Combination Guitars

224-Majic Guitar #2

225-Meditator E

226-Nylon Ensemble 2

227-A.Gtr.Ensemb.(CS)

228-3 Guitars (CS)

229-All Guitars (CS)

#### Jazz Guitars

230-Slo Chorus Gtr

231-Stereo Jazz Guit

232-Clean Lead Gtr

233-Jazz Dream

234-Stereojazzguitar

#### Guitar Mutes

235-Muted Guitar

236-Muted Guitar 2

237-Guitar Mutes 1

238-Guitar Mutes 2

239-Mutes 3

240-Mutes 4

241-Mutes 5

242-Fancy Mutes

243-Jungle Mutes

244-Press For Effect

#### Distorted Emulations

245-Distortion Gtr

246-xFadeDistGuitar 2

247-Nasty Lead Gtr

248-Rockin Lead

249-Press WahWah

250-HelterSkelter Gtr

251-GRUNGE

252-Attacker

253-Crank It Up

254-Harmonics Gtr.

255-Dist Harmonics

256-Soon

257-Charang

258-Smithereen

259-Guitar Lead

260-Optical Link

261-Meathead

#### Exotic Plucked Things

262-Kotolin

263-Twangy Lead

264-lectric twang

265-Ravitar

266-Cee Tuar

267-NewAge Guitar

268-Classical Gtr

269-Green Acres

270-Para Pick w/ Voice

271-Syncro Taps

272-Stratosphere

## HYBPERC.K25

### Percussive Hybrids

(38 Programs)

#### **Percussive Comps**

200-Night Ryder  
201-Ethnick 1  
202-Neastern  
203-Zawinul  
204-Industrial Komp  
205-Timber Shifter  
206-Mod Bel  
207-Klank 1  
208-Toy Store II  
209-Baribun  
210-Choir Stabs  
211-Resimallet  
212-Perc Flute  
213-Gateperc Too  
214-Islanders  
215-Driver 4  
216-Driver 5

#### **Percussive Pads**

217-D50 Voicebell  
218-Vox Marimba  
219-Wet Voices  
220-New Dawn 2  
221-Wood Pad  
222-Ensamble 1  
223-Perc Voices  
224-Mallet Voice  
225-Klakran  
226-Bella Voce  
227-Noo Marimbala  
228-Sweet Mallets  
229-Tranquil Pluck 2  
230-Dyn Marimba  
231-Arystal 1  
232-Arystal 2  
233-Arystal 3  
234-Dankness  
235-In the Well  
236-Orchestrar  
237-Orchestrar 2

## HYBRIDS.K25

### General Hybrid Sounds

(63 Programs)

#### Hybrid String Pads

200-Brt SynChoir  
 201-Angel Pad  
 202-Big TynthTex  
 203-Bush String Pad  
 204-Lush Strangs  
 205-Fake String  
 206-Stereo ChoirStr  
 207-Mirage Strings  
 208-FatMan Str II CS  
 209-Fatman Strings  
 210-Thick Low Pad  
 211-Synth Choir Ensemble

#### Hybrid String Pads w/Resonance

212-New Dawn  
 213-Big Single  
 214-Mella Tron  
 215-Froese String  
 216-String Reversal  
 217-DistortResonance  
 218-Hi Res Sweeper  
 219-7th World String  
 220-Sweeper  
 221-Lunar Dance  
 222-Stack Pad 4  
 223-String Machines

#### Hybrid Stacks

224-Fairlite Like  
 225-String Stack  
 226-Golck 'n Brass  
 227-Grand String  
 228-LA Stack  
 229-Rock Stack  
 230-All in the Fader  
 231-Utopian Comp  
 232-Gargantuan  
 233-Outside L / A  
 234-Stackoid  
 235-Ethnick 2

#### Flute-Like Hybrids

236-Hybrid Flute  
 237-Clave Flute  
 238-Xyliope  
 239-Fake Chiff  
 240-Fluty Lead 4  
 241-Perky Caliope  
 242-Koto Inside  
 243-Koto Inside 2  
 244-Chiff Lead  
 245-Marimba & Flutes  
 246-Flooter 2  
 247-E Pno & Lead 1  
 248-Bars & Lead

### Altered Acoustic Sounds

249-Vibe 5th  
 250-7-Sax Delay  
 251-7-Sax Delay 2  
 252-Mutant Brass  
 253-Neu Trumps  
 254-New Rumpett  
 255-Process Sax  
 256-RezTouch Sax 5th  
 257-S+H Violin  
 258-String Function  
 259-Violastic 2  
 260-BushKate Cellos  
 261-Aliens Wood  
 262-A Kordian

## ORGANS.K25

Organs

(41 Programs)

**Rock Organs**

200-Ballad Organ 2 pr  
201-Ballad Organ 3  
202-London Hammond  
203-London Hammond 2  
204-Drive Organ  
205-Drive Organ 2  
206-Rotating B&M's 2

**Clean Percussive Organs**

207-Perc Organ  
208-Clav Organ MW  
209-Bee3  
210-Tamborgan #2

**Pipe Organs**

211-Pipes 1  
212-Pipes 2  
213-Pipes 3  
214-Sanctuary Pipes  
215-Sanctuary Pipes2  
216-Flute Pipe1 C+MW  
217-Flute Pipes 2  
218-Cath.Pipes(C+MW)  
219-Cathedral Pipes2  
220-Pedal Pipes 2  
221-Pedal Pipes 3  
222-Pipes 2 (C+MW)  
223-Pipes 3 (C+MW)  
224-Pipes 4 (C+MW)

**Velocity Sensitive Organs**

225-Organ 1 (drawbar)  
226-Organ 2  
227-Organ 2 (perc)

**Organ-Like Synths**

228-Mello Perc  
229-Tamb Organ  
230-Organarimba  
231-Organellica  
232-Synth Pipe  
233-Mello organ  
234-Organ pad 2  
235-Organ 3 (perc)  
236-DrawbarPerc  
237-Diver  
238-Diver 2  
239-Diver 3  
240-Driver 3

**PNOEPNO.K25**

## Pianos &amp; E Pianos

(71 Programs)

**Classical Pianos**

200-ClassicalPiano  
 201-ClassPiano 2  
 202-DynamicPiano  
 203-BalladPiano 3

**Rock Pianos**

204-CP-70  
 205-CP-70 1 layer  
 206-Stereo Grand  
 207-Studio Piano 1  
 208-Studio Piano 3  
 209--Studio Piano 5  
 210-Rock Syn Piano  
 211-Rock Piano 1

**Pianos w/ Strings or Choir**

212-Piano&SloStrings  
 213-ClassPiano&Voice  
 214-ClassPiano&Str  
 215-Blld Piano & Str  
 216-Piano&FilterStr  
 217-Piano & Voicepad  
 218-Vox Tite Piano

**Bright and House Pianos**

219-Britegrand 2  
 220-Britegrand 3  
 221-Bright Piano  
 222-Bright Piano 2  
 223-House Piano

**Stylized Pianos**

224-Tight Piano  
 225-Lennon Piano 1  
 226-New Age CP-70  
 227-Honky-Tonk  
 228-Yama E Piano

**Electric Grands**

229-Electric Grand  
 230-Grand & Electric  
 231-Grand n Elec 4  
 232-Grando Elec 5  
 233-Grand&Elec&Choir  
 234-Warm E Grand 1

**Rhodes Pianos**

235-Classic E Piano  
 236-Classic E Pno 2  
 237-Fluid E Piano 1  
 238-Tine Waves  
 239-Dual Rhodes 1  
 240-Foster E Piano  
 241-Dual Rhodes 2  
 242-Phase Rhodes  
 243-EQ chrRhodes

**Dyno Pianos**

244-Celest EP (CS)

245-Tine Elec Piano  
 246-Dual Elec Piano  
 247-Suitcase E Pno  
 248-St Suitcase EP 2  
 249-Dyno E Piano

**FM Pianos**

250-Digital E Piano  
 251-FM EP  
 252-New EP  
 253-Yamaha E Pno 2  
 254-PF Elec Piano  
 255-Dx Rhodes  
 256-Dig E Piano  
 257-Elec Piano + Vox  
 258-Vollenweider  
 259-Dualin' pianos  
 260-E Gtrs & E Pno

**Clavinet Emulations**

261-Clavinetist  
 262-Brite Clav  
 263-DX Clav  
 264-ClavBassHarp  
 265-Clav 5

**Harpsichord Emulations**

266-Quillsichord  
 267-Harpsichord 1  
 268-Harpsichord 2  
 269-Harpschd & Str  
 270-Baroquen Trio

## STRINGS.K25

243-Stereo Str Pad 2

244-Stereo Str Pad 3

Strings

(45 Programs)

### **Straight Strings**

200-Straight Strings

201-Fast Strings

202-Fast Wet Strings

203-Strings eq 1

204-Strings eq 2

205-New Strings

206-New Strings 2

207-New Strings 3

### **Attack Velocity Strings**

208-Att Ctl Fast Str

209-Very Touch Str

210-Vel Strings B

211-AttCtl Med Str 1

212-AttCtl Med Str 2

### **Sfz Strings**

213-Sfz Strings MW

214-Sfz Trem Strings

### **Ensemble Strings**

215-Stereo Med Str

216-Grand Strings

217-Quick Strings MW

### **Layered Solo and Ensemble Strings**

218-Mixed Strings

219-Chmbr Strings

220-Baroque Strings

221-ClassicalStrSect

222-Silk Strings 1

223-Slo Classical St

224-Silk Strings 2

### **Solo Violin and Cello**

225-Prs Slo Violn

226-SloViolin prs

227-Violin att vib

228-MarcatoViolin MW

229-Marcato S . Strngs

230-Elec Violin

231-Slo Solo Str 2

232-Mellow Cello

### **Pizz Emulations**

233-Synth Pizz 1

234-Synth Pizz 2

235-Synth Pizz 3

236-Synth Pizz 4 lo

### **String Pads**

237-String Paddy

238-String Paddy 2

239-Melle Orchestra

240-Slo Ensemble

241-MelloStr & Choir

242-Stereo Str Pad 1



## VOX.K25

### Vocal Sounds

(21 Programs)

- 200-CathedralVoice
- 201-Bach Fixer
- 202-St Slo Voices
- 203-Smooth Choir
- 204-Dream Vox 2
- 205-Vox 2
- 206-Breath Pad 1
- 207-The Voice
- 208-Angels
- 209-Chant
- 210-Belle Orchestra
- 211-Fake Vox 3
- 212-Fake Vox II
- 213-PM's Choir Pad
- 214-Voice w/ Upper Str
- 215-Smooth Choir 2
- 216-Vox Piano
- 217-Emu Vox
- 218-Flutters
- 219-Fake Vox
- 220-Passover



# Appendix B

## K2vx Compatibility

### K2vx Compatibility Files

Included as part of your K2vx accessory disks are two disks of compatibility files, for your use when playing K2000 programs on the K2vx. The Kurzweil K2000 has been a widely used platform for several years, and the VAST architecture and programming interface is largely the same in the K2vx. Therefore, an attempt was made to organize the K2vx factory objects in a way most compatible with existing K2000 files. However, several improvements have been made to the Base ROM objects, and therefore not all K2vx support software will play correctly in a K2000 without some minimal translation.

The purpose of the files on the two compatibility disks is to allow you to play programs, sequences, and other objects that were created on a K2vx, so that they can be re-saved in a "native" K2000 format.

If you never owned a K2000 and you do not have existing material programmed on the K2vx, you probably do not need these files.

*Here are the main differences:*

**ROM Drum samples.** While most of the samples in the base ROM are compatible between the K2000 and the K2vx, the drum samples are not. The K2vx drums are made from new recordings, and a slightly different selection of drums is offered (e.g., three ambient snares instead of four). Furthermore, where all drums and percussion had been grouped in one multi-root sample (Sample #20 Drums and Percussion), they are now available as separate samples addressed by number.

**ROM Effects programs.** These were re-programmed for greater signal-to-noise ratio, and re-organized for ease of use. The Effect page in the program editor always points to an Effect program, and has several parameters for real-time control. Many programs developed for the K2000 series utilized those factory default effects. When these programs are loaded into a K2vx, they will not call the correct effect.

**ROM Keymaps.** An effort was made to keep instrument keymaps in the same order as in the K2vx, because the keymap must be correct for a program to sound correct. Keymaps 20-38, 61, 70, and 173-176 have been replaced or deleted, and subtle improvements in volume have been made to others.

*About the compatibility files:*

There is one main file on this two disk set, K2KBASE.K25. It contains all the necessary objects for a K2vx to play any program made on a late model K2000, including drum and percussion samples. If you do not have sample memory, you can still use this file for some compatibility, but the drums will not play.

The idea is to temporarily overwrite the ROM in the K2vx with these objects, so that K2000 programs can be loaded, played, and then re-saved with their dependent objects.

If you use the compatibility files often, you will find that sometimes you only need to load some of the objects from the big file. This can be done with the Load Object feature. As a convenience, we have provided a file which only contains effects programs, K2KFX.K25, for one such case.

We also included a file for Orchestral ROM compatibility, K2KROM1.K25. It should be loaded in tandem with the K2KBASE.K25 file only if you have the Orchestral ROM option installed. (There are very few differences between the K2000 and the K2vx in the Orchestral ROM bank, so this file will rarely be used.)

## **Converting K2000 Files to K2vx Files**

There are five steps to convert a K2000 file to a K2vx file:

Before you start, make sure you have saved all user objects to disk, because memory will be cleared.

Step 1 Load the compatibility file as Everything/ Overwrite. (Everything/Merge mode will work too)

Step 2 Load the file(s) you wish to port into any memory bank from 200 through 800.

Step 3 Save these objects with dependents to new files.

Step 4 Delete everything.

Step 5 Load the new files to make sure they play correctly.

Happy porting.

## Converting programs from the K2vx to K2000

There may be times when you wish to take a file you have created for your K2vx and load it into a K2000. As we have mentioned in the above section on loading K2000 files into the K2vx, most objects are compatible, but there are a few things you will need to be aware of:

### Programs using Drum samples

Since the K2vx has new drum samples, these programs will not translate correctly. The K2vx drum samples are not available on disk to be loaded into the K2000, so these programs simply can not be converted so that they will sound identical.

However, if you have some K2vx programs which take advantage of VAST programming and wish to use them in a K2000, you can load the program into the K2000, then edit the program to change the keymaps to the corresponding drum keymap. If the keymap is one of the 5 octave or 2 octave kit keymaps, you will find that for the most part, the type of percussion sound will match up, though there may be a few which don't. Keep in mind though that the sound itself may be quite different, since the samples are different.

### Effects Programs

The preset effects programs in the K2vx are different than in the K2000. However, since these effects programs consist simply of different values for the various editable parameters, a K2vx effect can be loaded into the K2000.

Here is the simplest way to include a K2vx effect in your file with the program. On the K2vx, call up the program that you will be porting to the K2000. Press edit and go to the EFFECT page. Press edit again to enter the Effects Editor. Now press save to save that effect to RAM. Once it is saved, press exit. You will now see that the RAM effect is assigned to the program. Press exit and save the program before leaving the editor. If the effect was saved to the same bank as the program, and you are saving the entire bank, both objects will be saved to the file. If the effect was saved to a different bank or you are selecting only individual programs to be saved to disk, be sure to answer yes to the "Save Dependent Objects?" question and the effect will be saved along with the program.

If you have a great number of programs that you want to convert and don't want to edit each of those programs, there is another method you can use. You can create your own K2000 effects compatibility file, similar to the K2vx effects file. This method will require more work initially, but once it is done, the file can easily be used again and again.

To do this, start with the K2vx cleared of all RAM objects. (Go to Master and delete Everything.) Now go to Effects Mode, call up each effect one by one, going into the Effects Editor and saving that effect to RAM. Save the effect back to the exact same number it was originally at, choosing Replace. For instance, save effect #17 back to location #17. (If you have an editor/librarian software program for your computer, you can get all the effects in one shot and save them to the same RAM locations.) Now save an Everything file to Disk. You now have a file similar to the K2KFX.K25 file on the compatibility disks. You can use the exact same set of five steps documented in the section on converting K2000 files to the K2vx, but this time you will be loading the files into the K2vx. (Don't forget to delete Everything in the K2vx when you are done creating the compatibility file.)

### Keymaps

The following keymaps are either different in the two instruments, or they do not exist in the K2vx: 20-30 & 173-191. (Keymaps 169-172 have different names in the K2vx but are identical to the ones in the K2000.) Keymaps 23-30 & 189-191 use drum samples and are therefore can not be converted to the K2vx (see the section on Drum Samples, above). But keymaps 20-22 & 173-188 can easily be ported to the K2vx. To do this, you will follow the same procedure used to convert effects programs, documented above. Follow those exact steps, but instead of going to the EFFECT page, go to the KEYMAP page in the Program Editor.

## **K2vx Compatibility**

---

Converting programs from the K2vx to K2000

## Appendix C

# Orchestral ROM

The Orchestral ROM Soundblock option adds 8 Megabytes of samples, including a full array of winds, brass, and strings. The Orchestral ROM Upgrade adds objects in the 900s bank. There you will find programs, keymaps, samples, effects, performance setups, and QA banks. All Orchestral ROM sounds can be combined with your existing 8 Megs of base sound ROM, 4 Megs of Stereo Piano ROM, and 8 Megs of (optional) Contemporary ROM.

### Orchestral ROM Effects

900	Rich Delay
901	Glass Delay
902	Real Plate
903	Real Niceverb
904	ClassicalChamber
905	Empty Stage
906	Long & Narrow
907	Far Bloom
908	New Hall w/Delay
909	With A Mic

## Orchestral ROM Programs

### Orchestras

900	TotalCntrl Orch1
901	TotalCntrl Orch2
902	BaroqueOrchestra
903	Oboe&Flute w/Str
904	Horn&Flute w/Str
905	Trp&Horns w/Str

### Winds

906	Piccolo
907	Orchestra Flute
908	Solo Flute
909	Orchestral Oboe
910	Solo Oboe
911	2nd Oboe
912	Orch EnglishHorn
913	Solo EnglishHorn
914	Orch Clarinet
915	Solo Clarinet
916	Orch Bassoon
917	Solo Bassoon
918	Woodwinds 1
919	Woodwinds 2

### Brass

920	Dynamic Trumpet
921	Copland Sft Trp
922	Orch Trumpet
923	Soft Trumpet
924	Strght Mute Trp
925	French Horn MW
926	Slow Horn
927	F Horn Con Sord
928	F Horn a2 MW
929	French Horn Sec1
930	French Horn Sec2
931	Solo Trombone
932	Tuba
933	Dyn Hi Brass
934	Dyn Lo Brass
935	Dyn Brass & Horn
936	Soaring Brass

### Solo Strings

937	MarcatoViolin MW
938	Solo Violin
939	2nd Violin
940	Orch Viola
941	Solo Viola
942	Slow Viola
943	Marcato Cello MW
944	Solo Cello
945	Slow Cello
946	Arco Dbl Bass
947	Slow Arco Bass
948	Brt Dbl Bass

### Section Strings

949	Touch Strings
950	Fast Strings MW
951	Chamber Section
952	Sfz Strings MW
953	Sweet Strings
954	Baroque Strg Ens
955	Big String Ens
956	Bass String Sec
957	Pizzicato String
958	Wet Pizz
959	Arco & Pizz

### Plucked Strings

960	Classical Guitar
961	Virtuoso Guitar
962	Acoustic Bass
963	Snappy Jazz Bass
964	Dynamic Harp
965	Harp w/8ve CTL
966	Harp Arps

### Keyboards

967	Celesta
968	Pipes
969	Pedal Pipes
970	Church Bells

### Percussion

971	Glockenspiel
972	Xylophone
973	Chimes
974	Timpani/Chimes
975	Timpani
976	Timpani & Perc
977	Big Drum Corp
978	Orch Percussion1
979	Orch Percussion2
980	Jam Corp
981	Conga & Perc
982	Woody Jam Rack
983	Metal Garden
984	Hot Tamali Kit
985	Funk Kit

### Synths

986	Magic Guitar
987	Glass Bow
988	Synth Orch
989	Nooage InstaHarp
990	AC Dream
991	Synth Dulcimer
992	Glistener
993	Afro Multi CTL
994	Tranquil Sleigh
995	Batman Strings
996	Ethnoo Lead
997	Orch Pad CTL
998	Choral Sleigh
999	Pad Nine



## Orchestral ROM Keymaps

### Reeds

900	<b>Oboe</b>
901	<b>English Horn</b>
902	<b>Bassoon</b>
903	<b>Clarinet</b>
904	Bassoon/Oboe
905	Bsn/EHrn/Oboe
906	Flute 2
907	Eng Horn/Oboe

### Brass

910	<b>Soft Trumpet</b>
911	<b>French Horn</b>
912	<b>French Hrn Sec</b>
913	<b>Tuba</b>
914	Tuba/Horn
915	Tuba/Hrn Sec
916	Tuba/Sft Trmpt
917	Trombet
918	Trumpbone
919	Trombone/SftTrmpt

### Orchestral Percussion

920	<b>Timpani</b>
921	<b>Snare Roll</b>
922	<b>Snare Hit</b>
923	<b>Orch Bass Drum</b>
924	<b>Orch Crash</b>
925	<b>Tam Tam</b>
926	<b>Triangle</b>
927	<b>Tambourine Roll</b>
928	<b>Tamb Hit</b>
929	<b>Sleigh Bells</b>
930	<b>Woodblock</b>
931	<b>Low Clave</b>
932	<b>Castanet Hit</b>
933	<b>Castanet Up</b>
934	Dry Snares
935	Amb Snare
936	Bass Drums
937	Orch Perc Units
938	Orch Perc Full
939	Misc Percussion
940	2Hand Amb Kit
941	2Hand Dry Kit
942	2H Kit Unit1
943	2H Kit Unit2
944	<b>Xylophone</b>
945	<b>Glockenspiel</b>
946	<b>Chimes</b>
947	2Hand DrumCorp

948	Lite Metal
949	Woody Perc
950	Celeste

### Pluck

951	<b>Plucked Harp</b>
952	Harp Gliss
953	<b>Nylon String Gtr</b>
954	Nylon Str noA2
955	Nylon for dulc
957	<b>Acoustic Bass</b>
960	<b>Pizz Strings</b>
961	Full Kbd DblBass

### Strings

962	<b>Solo Violin</b>
963	<b>Solo Viola</b>
964	<b>Solo Cello</b>
965	Fast Solo Cello
966	Solo Double Bass
967	Bass/Cello
968	Bass/Cello/Vio
969	Cello/Vla/Cello
970	Cello/Vla/Vln
971	Ens Strings 2
972	Solo Section 1
973	Solo Section 2
979	BassDrum/Timp

### Waveforms

980	<b>Organ Wave 8</b>
981	<b>Buzz Wave 2</b>
982	<b>Ahh Buzz Wave</b>
983	<b>OB Wave 1</b>
984	<b>OB Wave 2</b>
985	<b>OB Wave 3</b>

### Variations

986	Tenor tune alt
987	Dual Ride 1
988	Black Fills C
989	Orc Perc Preview
990	<GM>Standard Kit
991	<GM> Orch Kit
992	Castanets x 3
993	Tambourine x 3
994	Black Fills B
995	Black Fills A
996	2HandDrumCrp NB
997	Sleigh Loop
998	Bs Drm Rumble
999	Church Bell

Note: Items in **bold** represent the primary keymap for each instrument.

## Orchestral ROM Samples

900	Oboe	951	Harp
901	English Horn	953	Nylon String Gt
902	Bassoon	957	Acoustic Bass
903	Clarinet	960	Pizz Strings
904	DbI Reeds	962	Solo Violin
910	SoftTrump	963	Solo Viola
911	French Horn	964	Solo Cello
912	FrenchHrnSect	965	Fast Solo Cello
913	Tuba	966	Solo Double Bass
914	Synth Accord	967	Conga Tone ingrl
915	Tuba % Horn	968	Amb Kick 3 va
920	Timp	980	Organ Wave 8
921	Snare Roll	981	Buzz Wave 2
922	Snare Hit	982	Ahh Buzz Wave
923	Orch Bass	983	OB Wave 1
924	Orch Crash	984	OB Wave 2
925	Tam Tam	985	OB Wave 3
926	Triangle	988	Jackhammer
927	Tamb Roll	989	Scratch
928	Tamb Hit	990	Zap 1
929	Sleigh Bells	991	Alarm Bell
930	Woodblock	992	Deep House Clave
931	Low Clave	993	China Crash
932	Castanet Hit	994	Dry Sidestick
933	Castanet Up	995	Med Open Hi Hat
934	Bi TamTam<v2.0>	996	Syn Vibra Stick
935	Orch Crash ignf	997	Sleigh Loop
937	Dark Triangle	998	BD Rumble <v2.0>
938	MuteTriangle	999	Church Bell
939	Triangle (rel)		
944	Xylophone		
945	Glockenspiel		
946	Chimes		
950	Celeste		

## Orchestral ROM Programs with Controller Assignments

The preset programs in the K2vx Orchestral ROM are organized by category. You can either use them as they are or as a good starting point for your own work. There are many ways to put expressivity and variety in a single program by assigning MIDI controllers to the various DSP functions in its layers. This list describes how each of the preset programs can be modulated or altered by the various MIDI controls. Only those controls which may not be immediately evident are listed. Controls such as attack velocity and keynumber are understood to be assigned to most programs.

Prg #	Program Name	Mod Wheel	Data	MPress	Comments
<b>Pianos</b>					
788	<b>Piano Trio</b>		Ride cymbal fade	Vibrato - Bass	
789	<b>Pno &amp; Syn String</b>	String fade	String swell		
790	<b>Fluid Grand</b>		Wet/Dry mix		
791	<b>Haunted Piano</b>	Harp balance	Wet/Dry mix		
792	<b>Xylophano</b>	Release ctl	Wet/Dry mix		
793	<b>Grand,Harp&amp;Lead</b>	Lead tremolo	Lead fade	Lead tremolo	Sustain pedal does not affect the lead sound
<b>Orchestras</b>					
900	<b>TotalCntrl Orch1</b>	Layer bal	Adds brass & flute, boosts strings	Swell (trp out - ww solo)	
901	<b>TotalCntrl Orch2</b>	Layer bal, adds harp	Layer balance, adds horns/cuts woodwinds	Swell	
902	<b>BaroqueOrchestra</b>	None	None	Swell	Sost ped disables brass
903	<b>Oboe&amp;Flute w/Str</b>	Strings fadeout	Disables strings	None	
904	<b>Horn&amp;Flute w/Str</b>	Strings fadeout	Disables strings	None	
905	<b>Trp&amp;Horns w/Str</b>	Strings fadeout	Disables strings	None	
<b>Winds</b>					
906	<b>Piccolo</b>	None	Wet/Dry mix	None	
907	<b>Orchestral Flute</b>	Envelope control (slower)	Wet/Dry mix	None	
908	<b>Solo Flute</b>	Timbre (brighter)	Wet/Dry mix	None	
909	<b>Orchestral Oboe</b>	Swell	Wet/Dry mix, rate & depth	Vibrato	
910	<b>Solo Oboe</b>	Vibrato off	Wet/Dry mix	Swell	
911	<b>2nd Oboe</b>	Vibrato off	Wet/Dry mix	Swell	
912	<b>Orch EnglishHorn</b>	Swell	Wet/Dry mix, rate & depth	Vibrato	

**Orchestral ROM**

Orchestral ROM Programs with Controller Assignments

<b>Prg #</b>	<b>Program Name</b>	<b>Mod Wheel</b>	<b>Data</b>	<b>MPress</b>	<b>Comments</b>
913	<b>Solo EnglishHorn</b>	Vibrato off	Wet/Dry mix	Swell	
914	<b>Orch Clarinet</b>	Swell	Wet/Dry mix	Vibrato depth	
915	<b>Solo Clarinet</b>	Swell	Wet/Dry mix	Swell	
916	<b>Orch Bassoon</b>	Swell	Wet/Dry mix	Vibrato depth	
917	<b>Solo Bassoon</b>	Vibrato off	Wet/Dry mix	Swell	
918	<b>Woodwinds 1</b>	None	Wet/Dry mix	None	
919	<b>Woodwinds 2</b>	None	Wet/Dry mix, rate & depth	Swell, vibrato	
<b>Brass</b>					
920	<b>Dynamic Trumpet</b>	Swell	Wet/Dry mix	Vibrato depth	
921	<b>Copland Sft Trp</b>	Vibrato off	Wet/Dry mix	Swell	
922	<b>Orch Trumpet</b>	Timbre (darker)	Envelope Control	Swell, vibrato rate & depth	
923	<b>Soft Trumpet</b>	None	Wet/Dry mix	Vibrato depth	
924	<b>Strght Mute Trp</b>	Vibrato off	Wet/Dry mix	Swell	
925	<b>French Horn MW</b>	Timbre (brighter)	Wet/Dry mix	Vibrato rate & depth	
926	<b>Slow Horn</b>	Vibrato	Wet/Dry mix	None	
927	<b>F Horn Con Sord</b>	Timbre (brighter)	Wet/Dry mix	Vibrato depth	
928	<b>F Horn a2 MW</b>	Timbre (brighter)	Wet/Dry mix	None	
929	<b>French Horn Sec1</b>	None	Wet/Dry mix	Slight swell	
930	<b>French Horn Sec2</b>	None	Wet/Dry mix	Swell	
931	<b>Solo Trombone</b>	Selects legato layer	Wet/Dry mix	Slight swell when MW is off	
932	<b>Tuba</b>	Vibrato rate & depth	Wet/Dry mix	Vibrato rate & depth	
933	<b>Dyn Hi Brass</b>	Swell, legato	Wet/Dry mix	Swell	
934	<b>Dyn Lo Brass</b>	Swell, legato	Wet/Dry mix	Swell	
935	<b>Dyn Brass &amp; Horn</b>	Timbre (darker)	Wet/Dry mix	None	
936	<b>Soaring Brass</b>	None	Wet/Dry mix	None	
<b>Solo Strings</b>					
937	<b>MarcatoViolin MW</b>	Spiccato articula- tion	Wet/Dry mix	Vibrato rate & depth	
938	<b>Solo Violin</b>	Delays auto- vibrato	Wet/Dry mix	Vibrato rate & depth	
939	<b>2nd Violin</b>	Envelope control	Wet/Dry mix	Vibrato rate	
940	<b>Orch Viola</b>	Release time (shorter)	Wet/Dry mix	Vibrato depth	

Prg #	Program Name	Mod Wheel	Data	MPress	Comments
941	<b>Solo Viola</b>	Delays auto-vibrato	Wet/Dry mix	Vibrato rate & depth	
942	<b>Slow Viola</b>	Timbre (darker)	Wet/Dry mix	Swell, vibrato rate & depth	
943	<b>MarcatoCello MW</b>	Spiccato articulation	Wet/Dry mix	Vibrato rate & depth	
944	<b>Solo Cello</b>	Delays auto-vibrato	Wet/Dry mix	Vibrato rate & depth	
945	<b>Slow Cello</b>	Timbre (brighter)	Wet/Dry mix	Vibrato rate, swell	
946	<b>Arco Dbl Bass</b>	Bass boost	Wet/Dry mix	Vibrato depth	
947	<b>Slow Arco Bass</b>	Delays auto-vibrato	Wet/Dry mix	Swell, vibrato rate & depth	
948	<b>Brt Dbl Bass</b>	Decrescendo	Wet/Dry mix	Vibrato rate	
<b>Section Strings</b>					
949	<b>Touch Strings</b>	Timbre (brighter)	Envelope Control	Swell	
950	<b>Fast Strings MW</b>	Selects faster strings	Timbre (darker), Wet/Dry mix	Swell	
951	<b>Chamber Section</b>	None	Wet/Dry mix	Vibrato depth	
952	<b>Sfz Strings MW</b>	Tremolo	None	Swell	
953	<b>Sweet Strings</b>	Fade out	Wet/Dry mix	Vibrato depth	
954	<b>Baroque Strg Ens</b>	Bass boost, layer delay	Wet/Dry mix	Swell	
955	<b>Big String Ens</b>	None	Wet/Dry mix	Swell	
956	<b>Bass String Sec</b>	Bass boost on solo layer	Wet/Dry mix	None	
957	<b>Pizzicato String</b>	Timbre (darker)	Wet/Dry mix	None	
958	<b>Wet Pizz</b>	Treble boost	Wet/Dry mix	None	
959	<b>Arco &amp; Pizz</b>	Timbre (brighter), layer balance	Enables 2nd string layer, stereo panning	Swell	
<b>Plucked Strings</b>					
960	<b>Classical Guitar</b>	Fade/disables key-up layer	Wet/Dry mix	None	
961	<b>Virtuoso Guitar</b>	Vibrato rate & depth	Wet/Dry mix	None	Sost ped enables staccato envelope
962	<b>Acoustic Bass</b>	Vibrato rate & depth	Wet/Dry mix	None	
963	<b>Snappy Jazz Bass</b>	Vibrato rate & depth	Pitch of snap, disables ride	Vibrato rate & depth	Sost ped disables ride cymbal

## Orchestral ROM

### Orchestral ROM Programs with Controller Assignments

Prg #	Program Name	Mod Wheel	Data	MPress	Comments
964	<b>Dynamic Harp</b>	Release time (longer)	Wet/Dry mix	None	
965	<b>Harp w/8ve CTL</b>	Brightness	Enables octave	None	
966	<b>Harp Arps</b>	None	Selects diminished	None	
<b>Keyboards</b>					
967	<b>Celesta</b>	None	Wet/Dry mix	None	
968	<b>Pipes</b>	Timbre (hollow)	Wet/Dry mix	None	
969	<b>Pedal Pipes</b>	None	None	None	
970	<b>Church Bells</b>	Distance	Timbre (brighter)	None	
<b>Percussion</b>					
971	<b>Glockenspiel</b>	None	Wet/Dry mix	None	Sus ped enables key-up layer (for rolls)
972	<b>Xylophone</b>	Timbre (fuller)	Wet/Dry mix	None	Sus ped enables key-up layer (for rolls)
973	<b>Chimes</b>	None	Wet/Dry mix	None	
974	<b>Timpani/Chimes</b>	Alt attack (timp)	Wet/Dry mix	None	
975	<b>Timpani</b>	Alt attack	Wet/Dry mix	None	Sus ped enables key-up layer (for rolls)
976	<b>Timpani &amp; Perc</b>	Alt attack (timp)	None	None	Sost ped enables bass drum. Sus ped dampens.
977	<b>Big Drum Corp</b>	None	Enables both fill layers (black keys: f#3-a#4)	None	Sost ped switches layers. Sus ped dampens.
978	<b>Orch Percussion1</b>	None	Switches fill layers	None	Sus ped dampens
979	<b>Orch Percussion2</b>	None	Wet/Dry mix	None	Sus ped dampens
980	<b>Jam Corp</b>	Alt attack	Pitch control (black keys: f#3-a#4)	None	
981	<b>Conga &amp; Perc</b>	Pitch control	Wet/Dry mix	None	
982	<b>Woody Jam Rack</b>	Pitch control up to 1200ct	Enables random drum layer	None	
983	<b>Metal Garden</b>	Pitch control up to 1200ct	Pitch control down to -1200ct	None	
984	<b>Hot Tamali Kit</b>	Tunes drums, alt atk on snares	Switches to old drum map	None	
985	<b>Funk Kit</b>	Tunes drums	Switches to old drum map	None	

Prg #	Program Name	Mod Wheel	Data	MPress	Comments
	<b>Synths</b>				
986	<b>Magic Guitar</b>	Vibrato depth	Layer balance	Vibrato depth	
987	<b>Glass Bow</b>	Vibrato depth	None	Vibrato depth	
988	<b>Synth Orch</b>	Filter modulation	None	Vibrato depth	
989	<b>Nooage InstaHarp</b>	Vibrato depth	Echo rate	Vibrato depth	
990	<b>AC Dream</b>	Vibrato depth	Enables bell layer	Vibrato depth	
991	<b>Synth Dulcimer</b>	Filter modulation	Wet/Dry mix	None	Sus ped dampens
992	<b>Glistener</b>	Vibrato depth	None	Vibrato depth	
993	<b>Afro Multi CTL</b>	None	Pitch control	None	
994	<b>Tranquil Sleigh</b>	Panner rate	Bandpass width	None	
995	<b>Batman Strings</b>	Vibrato depth	None	Vibrato depth	
996	<b>Ethnoo Lead</b>	Vibrato depth	Pitch control	Vibrato depth	
997	<b>Orch Pad CTL</b>	Vibrato depth	Filter cutoff	None	
998	<b>Choral Sleigh</b>	Sleigh play	None	None	
999	<b>Pad Nine</b>	Vibrato depth	Filter cutoff	None	

## Orchestral ROM Setups

The Performance Setup, or "Setup" is a combination of three zones, each with independent MIDI channel and controller transmission assignments. Designed initially for models with built-in keyboards, Setups can be played on K2vxR via the Local Keyboard Channel feature: Find this parameter in MIDI mode on the RECV page, change it from None to a channel of your choice, and set your controller to send on only that channel. Now, any note that comes in on that channel will be re-mapped according to the display channel (in program mode) and according to the Setup (in Setup mode). Here are the Setups provided in the Orchestral ROM:

900	Common Manfare
901	Tutti Passage
902	Swan Lake
903	Magic Bell Pad
904	Take the L Train
905	Yankee Doodle
906	Lyric Cue
907	Orchestrator
908	Toy Jam
909	Ethereal Harp
910	Cemetary Night
911	Trumpet Trio
912	Overture
913	Snake Charmer
914	OrchScape 1
915	Glock & Celesta
916	Dual Pipes
917	Pastorale
918	Nylon Gtr/S. Cello
919	Concerto
920	DialogUnderscore
921	Mallet Justment
922	Octave Harp
923	Choir/Harp
924	Symphonic
925	Nighty Nite
926	Bela's Ensemble
927	Guitar & Strings
928	Astro Soundscape
929	Appalachian 1
930	Appalachian 2
931	Notre Organ
932	Plectrum
933	Harp/Chimes
934	TripleOctStrings
935	Folk Jam
936	OrchScape 2
937	ExoticPerc & Kit3
938	Bumper Orch
939	Madness Strings
940	Flute Trio
941	Tutti Str&Brs
942	Nylon Pad
943	Cartoon Score
944	Piano & Harp
945	Warm String Bed
946	Piano & Str/Cello
947	S.O.S. Pad
948	Piano Trio
949	TamaliKit/Bass
950	Finale Orch



## Mirror Image Drum Map

The Mirror Image Drum Map is a drumkit layout that enables a natural two-hand style of playing. The Mirror Image Drum Map gets its name by its instruments being laid out in a mirror image of itself with D4 being the point of reflection.

### Getting Started

Play the key, D4. You'll notice that snare drum is assigned to it. From there, play 1 semitone down (C#4), and up 1 semitone (D#4). Notice that the two are the same snare drum. Play 2 semitones down (C4) and up 2 semitones (E4). Notice that the same bass drum is assigned to both keys. (Bass drum is also repeated on E3 and C4, which is particularly useful in fast double bass drum playing.) Notes 3 semitones down, and up 3 semitones, have the same hi-hat, etc.

There are, however, two instances – G4 and E6 – where the left and right sides do not match. They deviate from the mirror image scheme to accommodate the more familiar one hand playing of hi-hat and tambourine.

The layout of the drums and various percussion instruments are easy to remember. Just keep in mind that the basic drumkit consisting of Snare, Bass Drum, Toms, and Cymbals are in the range of C3-E5, or the "inner core" range. The two remaining ranges (C2-B2 and F5-C7) which extend out to the left and right edges of the keyboard make up the "outer edge" range, and will generally consist of auxiliary percussion instruments. This "inner" and "outer" range structure is also maintained in the drum corps programs (#977, #980) and orchestral percussion programs (#978, #979).

It is easy to memorize the placement of instruments if you think of the double and triple groupings of the black keys as one instrument or instrument type. Look at the center group of black keys, C#4 and D#4. Think of that grouping as the snare drum. Fanning out on both sides to the next group of black keys, F#3, G#3, A#3 on the left hand side, and F#4, G#4, A#4 on the right hand side, are the toms. Fanning out farther to the next set of double black keys are the cymbals. The next set of triple black keys are the timbales, and the next set of double black keys are the congas. The four white keys under the toms are the hi-hats.

In the "outer edge" range, white keys are generally hand-held percussion toys or various useful articulations of the congas laid out such that one can play typical conga patterns with one hand.

Try playing in a straight eighth note beat D#2, E2, F2, D#2, E2, F2, C2, C#2 and repeat. For easy right-hand tambourine playing, try playing in the same eighth note beat C7, A6, E6, C7, A6, E6, C7, A6 and repeat. Now combine the left-hand conga part and the right-hand tambourine. A combination of easily fingered patterns will often yield a useful rhythm section.

### Sostenuto Pedal

One more bonus was added to the drumkit programs—the sostenuto pedal. Just hold down the sostenuto pedal and again play in a steady eighth note beat E3, G3, A3, B3, G3, A3, B3, G3 and repeat. The sostenuto adds percussion to the white keys ranging from F3 to C4, and C#4.

## Sticking

The Mirror Image Drum Map lets you simulate the sticking that a real drummer would use. Try playing a tom tom fill from hi tom to low tom using a paradiddle sticking (RLRRLRL). This should be very easy to execute with minimal physical motion. The symmetrical inward-outward motion also feels comfortable and smooth. Doubling or layering of instruments while maintaining the beat is as easy as grabbing chords. A good example of this can be illustrated with program 977, "Big Drum Corps".

Play a steady repeating sixteenth note snare drum pattern with your left and right thumbs alternating on the keys, C#4 and D#4. Keep that same left, right, left, right hand motion going but simply add your index fingers to play the next black keys which would be A#3 in the left hand and F#4 in the right hand. Finally, add the ring finger for the low tom on F#3 and A#4. If you look at your hands now the right hand is making an E flat minor chord and its mirror image chord, F# Major, is in the left hand. After playing with the drum programs you'll notice how easy it is to play multiple drums in unison or to add or drop a tom or crash cymbal while maintaining a continuous flow of rhythm.

For those of you who prefer the old Kurzweil drum map for the drumkit programs, it is available on the Controller slider.

Drumkit Programs: 984, 985

Drum Corps Programs: 977, 980

Orchestral Percussion Programs: 978, 979

## Appendix D

# Contemporary ROM

The Contemporary ROM Soundblock option adds 8 Megabytes of samples to your K2vx. These include ethnic percussion, electronic and processed drum sounds, electric guitars, synthesizer waveforms, contemporary keyboards, wind instruments, and much more. Combined with the powerful on-board Variable Architecture Synthesis Technology (VAST) capabilities of your K2vx, this new palette of sounds gives your instrument unmatched potential.

The Contemporary ROM Upgrade adds objects in the 800s bank. There you will find 100 programs, 94 keymaps, 94 samples, 10 effects, 51 performance setups, and 11 QA banks. All new sounds can be combined with your existing 8 Megs of base sound ROM, 4 Megs of Stereo Piano ROM and 8 Megs of (optional) Orchestral ROM.

## Contemporary ROM Programs

<b>Ethnic/World Instruments</b>	850	Shudder Kit
800 Jungle Jam	851	Crowd Stomper
801 Mbira Stack	852	Econo Kit
802 Ritual Metals	853	EDrum Kit 1
803 Prepared Mbira	854	EDrum Kit 2
804 Balinesque	<b>Loops</b>	
805 Ambient Bells	855	Dog Chases Tail
806 World Jam 1	856	Saw Loop Factory
807 World Jam 2	<b>Basses</b>	
808 India Jam	857	Two Live Bass
809 Slo Wood Flute	858	Dual/Tri Bass
810 Hybrid Pan Flute	859	Clav-o-Bass
811 Chiff Brass Lead	860	Chirp Bass
812 Bell Players	861	DigiBass
813 Prs Koto	862	Mono Synth Bass
814 Medicine Man	863	Touch MiniBass
815 Mbira	864	Ostinato Bass
816 Kotobira	865	House Bass
817 Cartoon Perc	866	Dubb Bass
818 CowGogiBell	<b>Guitars</b>	
819 Perc Pan Lead	867	Straight Strat
820 Trippy Organ	868	Chorus Gtr
821 Koto Followers	869	Strataguitar
822 Hybrid Horn	870	Elect 12 String
<b>Keyboards</b>	871	Dyn Jazz Guitar
823 Dyno EP Lead	872	Pedal Steel
824 ParaKoto	873	Strummer DistGtr
825 Super Clav	874	Rock Axe
826 StrataClav	875	Hammeron
827 Touch Clav	876	Rock Axe mono
828 Bad Klav	<b>Synth Timbres</b>	
829 Rad Rotor	877	Attack Stack
830 B-2001	878	Skinny Lead
831 Perc Organ	879	Q Sweep SynClav
832 Drawbar Organ CS	880	Anna Mini
<b>Brass/Reeds</b>	881	Ballad Stack
833 Bebop Alto Sax	882	Big Stack
834 Soft Alto Sax	883	BrazKnuckles
835 Soprano Sax	884	Hybrid Breath
836 Low Soft Sax	885	Hybrid Stack
837 Air Reeds CS	886	Eye Saw
838 Jazz Muted Trp	887	Mello Hyb Brass
839 Jazz Lab Band	888	Sizzl E Pno
840 Harmon Section	889	My JayDee
841 Sfz Cres Brass	890	Slo SynthOrch
842 Neo Stabs	891	SpaceStation
843 Gtr Jazz Band	892	Glass Web
844 Full Rock Band	893	Circus Music
<b>Drum Kits</b>	<b>Pads</b>	
845 World Rave Kit	894	Mandala
846 Punch Gate Kit	895	Slow Strat
847 Shadow Kit	896	Fluid Koto
848 Fat Traps	897	Koreana Pad
849 Generator Kit	898	Tangerine
	899	Planet 9

## Contemporary ROM Keymaps

### Synth Multi-Samples

800	Hybrid Pan
801	Glass Rim Tone
802	Synth Vox
803	Orch Pad
804	Koreana
805	Heaven Bells
806	MIDI Stack
807	Synth Brass
808	DigiBass
809	AnaBass
810	Mini Saw

### Instrument Multi-Samples

811	EBass Pick
812	EBass Slap
813	Clean Elec Gtr
814	Distorted Guitar
815	Dist Harmonics
816	Clav
817	Tone Wheel Organ
818	Muted Trumpet
819	Soft Alto Sax
820	Koto
821	Mbira

### Individual Percussion Roots

822	Tabla Ta
823	Tabla Tin
824	Tabla Dhin
825	Tabla/Bayan Dha
826	Bayan
827	Ghatam Bass Tone
828	Small Ghatam
829	Ghatam Shell
830	Ghatam Slap
831	Dumbek Open Tone
832	Dumbek Brt Tone
833	Dumbek Tek
834	Dumbek Snap
835	Dumbek Dry Dum
836	Djembe Tone
837	Djembe Cl Slap
838	Djembe Open Slap
839	Djembe Finger
840	Djembe w/ Stick
841	Muzhar
842	Talking Drum Lo
843	Talking Drum Hi
844	Luna Drum Dry
845	Luna Drum Hi
846	Log Drum Lo
847	Log Drum Hi
848	Shakers/Tamborim
849	Gankogui Bell Lo

850	Gankogui Bell Hi
851	Tibetan Cymbal
852	Tibetan Bowl
853	Indo Bowl Gong

### Percussion Kits

854	Prev Ethnic Perc
855	Cartoon Perc
856	Prev EDrum Map
857	Toms Map
858	ProcKick/Snr Map
859	EDrum Kit 1
860	EDrum Kit 2
861	1 Lyr Proc Kit
862	Industry Perc
863	Tuned Loops

### Custom Percussion Keymaps

870	PreparedMbira L1
871	PreparedMbira L2
872	World Jam 1 L1
873	World Jam 1 L2
874	World Jam 1 L3
875	India Jam L1
876	India Jam L2
877	World Jam 2 L1
878	World Jam 2 L2
879	World Jam 2 L3
880	World Jam 2 L4
881	World Jam 2 L5
882	World Jam 2 L6
883	World Jam 2 L7
884	World Jam 2 L8
885	CowGogiBell L1
886	Dual Log Drum
887	Jungle ProcDrms
888	JungleBrushTip1
889	JungleBrushTip2
890	Jungle Birds
891	Jungle Ghtm rel
892	Jungle Tabla
893	Jungle Dumbek
894	Jungle ProcDrms2
895	Jungle Ghtm Strgt

### Custom Keymap

896	Syn Bass Pick
-----	---------------

### Single-Cycle Waveforms

897	ARP SAW
898	ARP PW30%
899	OB PW25%

## Contemporary ROM Samples

800	Hybrid Pan	850	Gankogui Bell Hi
801	Glass Rim Tone	851	Tibetan Cymbal
802	Synth Vox	852	Tibetan Bowl
803	Orch Pad	853	Indo Bowl Gong
804	Koreana	854	EDrum1 Kick
805	Heaven Bells	855	EDrum1 Snare
806	MIDI Stack	856	EDrum1 Rim
807	Synth Brass	857	EDrum1 Hi Tom
808	DigiBass	858	EDrum1 Crash
809	AnaBass	859	EDrum1 Cowbell
810	Mini Saw	860	EDrum1 Clave
811	EBass Pick	861	EDrum1 Shaker
812	EBass Slap	862	EDrum2 Kick1
813	Clean Elec Gtr	863	EDrum2 Kick2
814	Distorted Guitar	864	EDrum2 Kick3
815	Dist Harmonics	865	EDrum2 Snare1
816	Clav	866	EDrum2 Snare2
817	Tone Wheel Organ	867	EDrum2 Snare3
818	Muted Trumpet	868	EDrum2 HH Open
819	Soft Alto Sax	869	EDrum2 HH Close
820	Koto	870	EDrum2 Clap
821	Mbira	871	EDrum2 Conga
822	Tabla Ta	872	Hi Proc Tom
823	Tabla Tin	873	Hi Mid Proc Tom
824	Tabla Dhin	874	Lo Mid Proc Tom
825	Tabla/Bayan Dha	875	Lo Proc Tom
826	Bayan	876	Syn Toms
827	Ghatam Bass Tone	877	Proc Kicks
828	Small Ghatam	878	Proc Snares
829	Ghatam Shell	879	Rvs Proc Kicks
830	Ghatam Slap	880	Rvs Proc Snares
831	Dumbek Open Tone	881	Bayan Mute
832	Dumbek Brt Tone	882	Alt Muzhar Rim
833	Dumbek Tek	883	Alt Tabla Ta
834	Dumbek Snap	884	Alt Maracas
835	Dumbek Dry Dum	885	Alt Shakere
836	Djembe Tone	886	Syn Bass Pick
837	Djembe Cl Slap	887	Alt Log Drum Lo
838	Djembe Open Slap	888	Alt Tibetan Cym
839	Djembe Finger	891	Dumbek Mute Slap
840	Djembe w/ Stick	896	ROM Loops
841	Muzhar	897	ARP SAW
842	Talking Drum Lo	898	ARP PW30%
843	Talking Drum Hi	899	OB PW25%
844	Luna Drum Dry		
845	Luna Drum Hi		
846	Log Drum Lo		
847	Log Drum Hi		
848	Shakers/Tamborim		
849	Gankogui Bell Lo		

## **Contemporary ROM Effects**

800	Percussive Room
801	Brt Empty Room
802	Mosque Room
803	New Gated
804	Chorus Slap Room
805	Chorus Bass Room
806	New Chorus Hall
807	Spacious
808	Wash Lead
809	New Hall w/Delay

## Contemporary ROM Programs with Controller Assignments

The 100 preset programs in the K2vx Contemporary ROM are organized by category. We hope you will find these programs to be a good starting point for your own work. There are many ways to put expressivity and variety in a single program by assigning MIDI controllers to the various DSP functions in its layers. This list describes how each of the 100 factory preset programs can be modulated or altered by the various MIDI controls. Only those controls which may not be immediately evident are listed. Controls such as attack velocity and keynumber are understood to be assigned to most programs.

Prg #	Program Name	Mod Wheel	Data	MPress	Comments
<b>Pianos</b>					
794	<b>Water Piano</b>	Vibrato	Wet/Dry mix	Vibrato	
795	<b>StPno &amp; OrchPad</b>	Pad balance			
796	<b>Grand &amp; Pad</b>	Pad balance	Bell release envelope		
797	<b>Pop Grand Stack</b>	Bell fade	Wet/Dry mix	Vibrato	
798	<b>Prepared Piano</b>	Alt switch - mbira	Wet/Dry mix		
799	<b>Tack Piano Stack</b>	Bell fade, Wet/Dry mix	Pitch env - mbira		
<b>Ethnic/World Instruments</b>					
800	<b>Jungle Jam</b>	This program uses the mirror image drum mapping, symmetrical around D4. Identical or similar drum articulations are found at equal distances above and below D4, with extras outside the center region. Mod wheel disables layered "chirps" and fades rain stick on A0. Data slider enables "screamers" on G5-C6.			
801	<b>Mbira Stack</b>	Vibrato			
802	<b>Ritual Metals</b>	Vibrato		Vibrato	
803	<b>Prepared Mbira</b>		Pitch change		
804	<b>Balinesque</b>	Pan flute fade			
805	<b>Ambient Bells</b>	Vibrato		Vibrato	
806	<b>World Jam 1</b>		Pitch change		Mirror image drum mapping
807	<b>World Jam 2</b>		Pitch change	Layer pitch	Mirror image drum mapping
808	<b>India Jam</b>	Tablas appear at center with the mirror-image mapping, tuned to C. Pressure controls pitch for the bayan and RH lead sound. LH drone may be played as broken chord C2,G2,C3,G3 and held with sustain or sostenuto. Mod Wheel fades the drone. Data Slider controls Wet/Dry mix.			
809	<b>Slo Wood Flute</b>	Less tremolo		Filter ctl	
810	<b>Hybrid Pan Flute</b>	Tremolo		Tremolo	
811	<b>Chiff Brass Lead</b>	Vibrato, Swell	Unison layers	Vibrato, Filter	
812	<b>Bell Players</b>	Muzhar fade	Tibetan cym env ctl		
813	<b>Prs Koto</b>			Pitch mod	



Prg #	Program Name	Mod Wheel	Data	MPress	Comments
814	<b>Medicine Man</b>				
815	<b>Mbira</b>	Release ctl	Tremolo		
816	<b>Kotobira</b>	Mbira balance			
817	<b>Cartoon Perc</b>		Wet/Dry mix		
818	<b>CowGogiBell</b>	Alt start	Layer select		
819	<b>Perc Pan Lead</b>	Vibrato			
820	<b>Trippy Organ</b>	Vibrato		Vibrato	
821	<b>Koto Followers</b>	Vibrato		Vibrato	
822	<b>Hybrid Horn</b>	Balance (bell)		Timbre ctl, Vibrato	
<b>Keyboards</b>					
823	<b>Dyno EP Lead</b>	Tremolo, Env ctl			
824	<b>ParaKoto</b>	Pad tremolo			
825	<b>Super Clav</b>	Phase clav enable	Disable release	Filter rate	
826	<b>StrataClav</b>	Vibrato		Vibrato	
827	<b>Touch Clav</b>	EQ, Vibrato	Disables release	Filter control	
828	<b>Bad Klav</b>				
829	<b>Rad Rotor</b>	Rotary speaker			
830	<b>B-2001</b>	Rotary speaker	Perc balance	Rotary speaker	
831	<b>Perc Organ</b>	Rotary speaker	Perc balance	Rotary speaker	
832	<b>Drawbar Organ CS</b>	Rotary speaker	Filter ctl		
<b>Brass &amp; Reeds</b>					
833	<b>Bebop Alto Sax</b>	Attack ctl		Vibrato	
834	<b>Soft Alto Sax</b>			Vibrato, Swell	
835	<b>Soprano Sax</b>	Vibrato, Swell		Vibrato, Swell	
836	<b>Low Soft Sax</b>			Vibrato	
837	<b>Air Reeds CS</b>	Vibrato	Harmonica enable	Harmonica vibrato	
838	<b>Jazz Muted Trp</b>				
839	<b>Jazz Lab Band</b>			Vibrato, Swell	
840	<b>Harmon Section</b>	Vibrato		Vibrato, Swell	
841	<b>Sfz Cres Brass</b>	Vibrato	Wet/Dry mix	Vibrato, Swell	
842	<b>Neo Stabs</b>	Vibrato		Vibrato, Filter ctl	
843	<b>Gtr Jazz Band</b>	LH bass is layered with ride for walking rhythm section. LH hard strikes trigger kick/snare. Data slider switches RH from guitar to horn section; SostPed holds horns and adds bright tenor.			
844	<b>Full Rock Band</b>	LH bass is layered with kick/snare for driving rhythm section. At <i>ff</i> , crash cymbal is triggered. Mod wheel and pressure enable rotary speaker for RH organ. Data slider switches LH to walking rhythm section, and RH to guitar solo.			

Contemporary ROM

Contemporary ROM Programs with Controller Assignments

Prg #	Program Name	Mod Wheel	Data	MPress	Comments
<b>Drum Kits</b>					
845	<b>World Rave Kit</b>	Disable chirps	Wet/Dry mix, Disable claps (G6-G#6)		
846	<b>Punch Gate Kit</b>		Wet/Dry mix		
847	<b>Shadow Kit</b>	Flanging (A#3-B3)	Wet/Dry mix		
848	<b>Fat Traps</b>	Filter (C2-A#2)	Wet/Dry mix		
849	<b>Generator Kit</b>	Disable claps (G3-G#3)	Wet/Dry mix		
850	<b>Shudder Kit</b>		Wet/Dry mix		
851	<b>Crowd Stomper</b>		Wet/Dry mix		
852	<b>Econo Kit</b>	Gate time (G3-C#4)	Wet/Dry mix		
853	<b>EDrum Kit 1</b>	Gate time (B2-D#3, G3-C#4), Pitch (D6)	Wet/Dry mix	Pitch (D6)	Sust ped chokes cym- bal (F#5)
854	<b>EDrum Kit 2</b>	Filter ctl (A#1-C2, F#6-C7)	Wet/Dry mix		
<b>Loops</b>					
855	<b>Dog Chases Tail</b>	Various loop effects	Tempo (pitch)		Loops below E4 are tuned to play together, as are loops above E4.
856	<b>Saw Loop Factory</b>	Layer balance	Tempo (pitch)		
<b>Basses</b>					
857	<b>Two Live Bass</b>	Vibrato	Layer select	Vibrato	
858	<b>Dual/Tri Bass</b>	Vibrato	Ghost note enable	Vibrato	
859	<b>Clav-o-Bass</b>	Vibrato	Wet/Dry mix	Vibrato	
860	<b>ChirpBass</b>	Vibrato	Wet/Dry mix	Vibrato	
861	<b>DigiBass</b>				
862	<b>Mono Synth Bass</b>		Filter		Pitch bend goes +2/- 12ST
863	<b>Touch MiniBass</b>	Vibrato		Vibrato, Swell	
864	<b>Ostinato Bass</b>		EQ		
865	<b>House Bass</b>	Vibrato	Release ctl	Vibrato	
866	<b>Dubb Bass</b>	Vibrato	Release ctl	Vibrato	

Prg #	Program Name	Mod Wheel	Data	MPress	Comments
<b>Guitars</b>					
867	<b>Straight Strat</b>	Tremolo	EQ		
868	<b>Chorus Gtr</b>		Wet/Dry mix	Detune	
869	<b>Strataguitar</b>	Alt start			
870	<b>Elect 12 String</b>	Detune	Wet/Dry mix, EQ	Vibrato	
871	<b>Dyn Jazz Guitar</b>		Wet/Dry mix		PBend gives fretboard slide
872	<b>Pedal Steel</b>	Vibrato		Vibrato	
873	<b>Strummer DistGtr</b>	Vibrato		Vibrato	
874	<b>Rock Axe</b>	Alt start	EQ	Feedback	
875	<b>Hammeron</b>	Timbre ctl		Timbre ctl	
876	<b>Rock Axe Mono</b>	Alt start	EQ, Delay	Feedback	
<b>Synth Timbres</b>					
877	<b>Attack Stack</b>	Vibrato	Wet/Dry mix	Vibrato	
878	<b>SkinnyLead</b>	Vibrato	Overdrive enable	Vibrato, Filter	
879	<b>Q Sweep SynClav</b>	Vibrato	Sweep rate ctl	Vibrato	
880	<b>Anna Mini</b>	Vibrato		Vibrato	
881	<b>Ballad Stack</b>	Swell		Swell	
882	<b>Big Stack</b>	Vibrato	Env ctl	Vibrato	
883	<b>BrazKnuckles</b>	Swell	EQ		
884	<b>Hybrid Breath</b>	Envelope ctl, EQ	Envelope ctl, Wet/Dry mix	Vibrato	
885	<b>Hybrid Stack</b>		Layer balance		
886	<b>Eye Saw</b>	Vibrato	Release ctl, Filter	Vibrato	
887	<b>Mello Hyb Brass</b>				
888	<b>Sizzl E Pno</b>	Pad balance			
889	<b>My JayDee</b>	Vibrato	Release ctl	Vibrato	
890	<b>Slo SynthOrch</b>	Filter effect			
891	<b>SpaceStation</b>	Vibrato	Envelope ctl	Vibrato	
892	<b>Glass Web</b>	EQ	Delay ctl		
893	<b>Circus Music</b>	Vibrato		Vibrato	
<b>Pads</b>					
894	<b>Mandala</b>	Filter ctl	Pitch change		
895	<b>Slow Strat</b>	Vibrato	Filter sweep enable	Vibrato	
896	<b>Fluid Koto</b>	Vibrato		Vibrato	
897	<b>Koreana Pad</b>	Tremolo	Filter, Wet/Dry mix		
898	<b>Tangerine</b>	Enable 5th	Envelope Ctl	Vibrato	
899	<b>Planet 9</b>				

## Contemporary ROM Setups

The Performance Setup, or "Setup" is a combination of three zones, each with independent MIDI channel and controller transmission assignments. Designed initially for models with built-in keyboards, Setups can be played on K2vxR via the Local Keyboard Channel feature: Find this parameter in MIDI mode on the RECV page, change it from None to a channel of your choice, and set your controller to send on only that channel. Now, any note that comes in on that channel will be re-mapped according to the display channel (in program mode) and according to the Setup (in Setup mode).

Here is a list of the Setups provided in the Contemporary ROM:

800	Dark Jungle	825	Plucky Toys
801	Lay It Down Funk	826	Big Pretender
802	World Bells	827	Low Octave Jam
803	Praise Hymn	828	Low C Groove
804	Perc Ensemble	829	Orient
805	Stackz	830	Earth Sing
806	Breath Stack	831	Bright Stack
807	Hybrid Jam	832	Ancient Future
808	Dance Split	833	Zen Divide
809	Perc Body & Pad	834	Muted Mood Split
810	TriplePress Lead	835	Spiced Out
811	Ethnic Machine	836	Loop & Groove
812	Hot Sauce	837	Stacks & Lead
813	Earth Drums	838	Club Date
814	Mystic	839	Bluegrass Split
815	Jazz Baby	840	E Guitar Trio
816	Trippy Planet	841	Hybrid Split
817	Big Brass	842	Trashed Trio
818	World Kit	843	Mega Drawbars
819	Way Bass Comp	844	Glory Days
820	World Ensemble	845	Retro Rock
821	Lullabye EP	846	Pedal Steel Splt
822	Amber Pad	847	Tender Split
823	Snake Dancer	848	Psycho Fun House
824	Lead Ensemble	849	In The Trenches
		850	Foggy Bells

## Symbols

! AMP 14-49  
+ SHAPE MOD OSC 14-48

## Numerics

2PARAM SHAPER 14-44  
2POLE ALPASS 14-16  
2POLE LOWPASS 14-8  
32 track sequencing 12-46  
4POLE HIPASS W/SEP 14-13  
4POLE LOPASS W/SEP 14-10

## A

A clock 25-9  
Aborting a Macro Load 13-54  
Aborting a Multiple File Load 13-21  
Aborting a sample recording 15-6  
Absolute Pitch Wheel 25-7  
Added waveforms 14-38  
Adding layers 6-33  
Adjust  
    Coarse 6-7  
    Fine 6-8  
Adjusting effects mix 9-2  
Adjusting sample volumes 15-36  
Aftertouch 11-2  
AIFF files 13-66  
AIFF-C files 13-66  
Akai 15-31  
Algorithm (ALG) page 6-9  
Algorithm basics 6-5  
Algorithm output 6-5  
Algorithm programming 14-1  
Algorithm wiring 6-5  
All Controllers Off 10-5  
All Notes Off 10-5  
Allpass filters 14-15  
ALPASS 14-15  
Alpha Wheel 2-4, 3-4  
Alphabetic entry 3-4, 5-3  
Alphanumeric pad 3-4  
Alternative attack 6-15  
AMP  
    Adjust 6-17  
    Key tracking 6-18  
    Pad 6-18  
    Sources 1 and 2 6-18  
    Velocity tracking 6-18  
Amp control 14-31  
AMP MOD 14-49  
AMP MOD OSC 14-48  
AMP U AMP L 14-32  
Amplitude envelope 25-11  
    Attack segment 6-25  
    Decay segment 6-26  
    Loop type 6-26  
    Natural 6-24  
    Number of loops 6-26  
    Release segment 6-26  
    User 6-24  
Amplitude envelope (AMPENV) page 6-24  
Append soft button 13-19  
Are you sure? dialog  
    suppressing with Confirm on Master page 11-3  
ARRANGE Page  
    Song Mode 12-46  
Arrangement Feature  
    Edit Song Common parameters used with 12-26  
ASCII characters 5-3  
ASR page 6-30  
ASR1, ASR2 25-11  
ASRs  
    Attack 6-31  
    Delay 6-31  
    Mode 6-31  
    Release 6-31  
    Trigger 6-31  
Assign sample 15-35  
Assigning samples to keymaps 15-36  
Attack portamento 6-23  
Attack segment 6-25  
Attack state 25-12  
Attack time for ASRs 6-31  
Attack velocity 11-2, 25-10  
Audio cables 2-2  
Audio Configurations 18-1  
Audio output 6-18  
Audio output groups and MIDI channels 18-4  
Auditioning Objects in RAM 13-25  
Auditioning Samples from a Disk File 13-17

## B

- B clock 25-9
- Backup 13-61
- Balance (MIDI 08) 25-4
- Balance and Amp (BAL AMP) 14-33
- Balance control 25-8
- BAND2 14-20
- BANDPASS FILTER 14-18
- Bandpass filters 14-18
- Bank Select
  - MIDI Receive page 10-8
- Bank Status Indicator 13-18
- Basic editing 5-1
- Basic MIDI channel 10-5
- Bass tone control 14-27
- Battery Installation 2-2
- Battery replacement 2-2, 28-1
- Beat Volume Adjust 15-29
- Bipolar attack velocity 25-10
- Bipolar control sources 16-1
- Bipolar key number 25-10
- Bipolar keystart 14-4
- Bipolar Mod Wheel 25-7
- Bipolar mono pressure 25-6
- Bipolar polyphonic pressure 25-10
- Boldface italic type 1-3
- BOOT.MAC 13-64
- Bottom line of display 3-2
- Bounce
  - Song Mode TRACK Page 12-35
- Breath (MIDI 02) 25-4
- Building a Drum Program 19-9
- Building a keymap 15-37
- Bypass effects 5-6, 20-4

## C

- Cable 15-7
- Cables and Input Jacks 15-1
- Cancel
  - soft button 13-10
- Cancel soft button 13-19
- CD-ROM 13-1
- CHAN/BANK buttons 3-3, 5-6
- Change
  - Song Mode TRACK Page 12-39
- Changing intonation key 11-3
- Changing song playback programs 12-49

- Channel count 25-8
- Channel state 25-6
- Channels on and off 10-9
- Chaotic LFOs (FUNs) 16-13
- Characters 5-3
- Choose File Name Function 13-37
- Cleaning your K2500 28-1
- Clear 15-20
- Clicking during portamento 6-23
- Clock
  - Song mode 12-21
- Coarse adjust 6-7
- Common DSP control parameters 6-6
- COMMON Page
  - Song Mode 12-25
- COMMON page 6-22
- Compare 5-6, 20-4
- Confirmations 11-3
- Connecting MIDI 2-2
- Connecting SCSI devices 13-2
- Contemporary ROM D-1
- Control input pages 14-2
- Control inputs 6-5
- Control messages
  - Balance (MIDI 08) 25-4
  - Breath 25-4
  - Data (MIDI 06) 25-4
  - Data decrement (MIDI 97) 25-6
  - Data increment (MIDI 96) 25-6
  - Effects depth (MIDI 91) 25-6
  - Expression (MIDI 11) 25-5
  - Foot (MIDI 04) 25-4
  - Freeze pedal (MIDI 69) 25-5
  - Legato switch (MIDI 75) 25-5
  - Mod Wheel (MIDI 01) 25-4
  - Mono pressure 25-4
  - Pan (MIDI 10) 25-4
  - Panic (MIDI 123) 25-6
  - Portamento switch (MIDI 65) 25-5
  - Portamento time (MIDI 05) 25-4
  - Soft pedal (MIDI 67) 25-5
  - Sostenuto (MIDI 66) 25-5
  - Sustain (MIDI 64) 25-5
  - Volume (MIDI 07) 25-4
- Control mode
  - setup 7-3
- Control parameters 2-6
- Control source lists 25-2
- Control sources 25-1
  - A clock 25-9
  - Absolute Pitch Wheel 25-7
  - Amplitude envelope 25-11
  - ASR1, ASR2 25-11

- Attack state 25-12
- Attack velocity 25-10
- B clock 25-9
- Balance control 25-8
- Bipolar attack velocity 25-10
- Bipolar key number 25-10
- Bipolar Mod Wheel 25-7
- Bipolar mono pressure 25-6
- Bipolar polyphonic pressure 25-10
- Channel count 25-8
- Channel state 25-6
- Constants for FUNS 25-13
- Envelopes 2 and 3 25-11
- FUN1, FUN2 25-11
- FUN3, FUN4 25-11
- GAttVel 25-12, 25-13
- GKeyNum 25-12
- Global ASR2 25-7
- Global FUN2 25-7
- Global FUN4 25-8
- Global LFO2 25-7
- Global LFOphase 25-7
- Global phase 1 and 2 25-9
- Inverse attack velocity 25-10
- Key number 25-10
- Key state 25-9
- LFO1 25-11
- LFO1 phase 25-11
- LFO2 25-11
- Loop state 25-12
- Mono pressure 25-6
- Negative A clock 25-9
- Negative B clock 25-9
- Note state 25-9
- ON 25-12
- ON 25-12
- Pan control 25-8
- Pitch Wheel 25-7
- Polyphonic pressure 25-10
- Random variants 1 and 2 25-11
- Release state 25-12
- Release velocity 25-10
- Sample playback rate 25-12
- Sync state 25-9
- Unipolar vs Bipolar 16-1
- Velocity triggers 1 and 2 25-10
- Volume control 25-8
- Control sources—Overview 16-1
- Controlling envelopes 6-27
- Controlling LFO rate 6-30
- Copy
  - disk mode 13-61
  - on Object Utilities page 11-8

- Song Mode TRACK Page 12-34
- Copying layers 6-33
- Creating a Macro 13-41
- Creating Directories 13-11
- Creating new layers 6-33
- Crescendo/Decrescendo (Crescendo) 15-24
- Crossfade 6-19, 14-34
  - Orientation 6-19
- Crossfade and Volume Adjust Curves 15-30
- Crossfade Loop (XfadeLoop) 15-30
- Cursor buttons 3-2

## D

- Data (MIDI 06) 25-4
- Data decrement (MIDI 97) 25-6
- Data entry 3-4
- Data increment (MIDI 96) 25-6
- DC offset 14-44, 14-46, 14-47, 14-48, 14-50
- Decay segment 6-26
- Delayed vibrato 19-1
- Delaying ASRs 6-31
- Delete 15-21
  - on Object Utilities page 11-9
  - Song Mode TRACK Page 12-36
- Deleting Banks in a Startup File 13-64
- Deleting Files and Directories 13-61
- Deleting layers 6-34
- Deleting objects 5-4, 11-4
- Deleting programs 6-33
- Dependent objects
  - saving 13-26
- Dependents
  - loading 13-17
- Descriptions of control sources 25-3
- Diagnostics 28-2
- Dialogs
  - Save 5-2
- Digital Outputs
  - using 15-8
- Diode equations (FUNs) 16-14
- Directories 13-4
  - creating 13-11
  - deleting 13-61
- Directory Selection Dialog 13-12
- DISK 5-6
- DISK button 5-6
- Disk Drive Information 13-5
- Disk format requirements 13-1
- Disk Function soft buttons 13-5

- Disk Mode 13-1
- Disk mode 4-4
- Disk Mode Functions 13-13
- Disk Utilities 13-55
- Display 3-1
  - Bottom line 3-2
  - Top line 3-2
- Display contrast 2-3, 11-3
- Distortion (DIST) 14-40
- Double button presses 3-5
- Double notch filter 14-20
- DOUBLE NOTCH W/SEP 14-20
- Double shaper 14-44
- Double-output algorithm 6-5
- Double-output algorithms 6-20
- Drum channel 11-1
- Drum Loop
  - example of recording 12-3
- Drum Program 19-9
- Drum programs 11-1
- Drum trigger 15-4
- DSP control inputs 6-5
- DSP control parameters 6-6, 6-9, 14-2, 14-3
- DSP Functions 15-18
- DSP functions 14-5
  - inputs and outputs 6-5
- Dump
  - on Object Utilities page 11-10
- Dump program 6-33
- Dumping samples via SMDI 29-7
- Duplicating layers 6-33
- Dynamic voice allocation 11-4

## E

- EDIT button 3-3, 5-6
- Edit compare 5-6, 20-4
- Edit Song
  - ARRANGE Page 12-46
  - COMMON Page 12-25
  - EVENT Page 12-28
  - STEP Page 12-43
  - TRACK Page 12-31
- Edit Song Pages 12-25
- Editing 5-1
- Editing Macros 13-50
- Editing Samples 15-8
- Effect changes when entering Effects mode 9-3
- EFFECT page 6-21
- Effect preset 6-21

- Effect Selection During Recording and Playback 12-49
- EffectChan
  - in Song mode 12-26
- Effects
  - selecting 9-3
  - Wet/Dry mix 6-21
- EFFECTS button 5-6
- Effects bypass 5-6, 20-4
- Effects configurations 9-6, 22-1
- Effects configurations and parameters 9-7
- Effects Controller Numbers 22-2
- Effects depth (MIDI 91) 25-6
- Effects Editor 9-5
- Effects in setups 7-3
- Effects mix 9-2
- Effects Mix in setups 7-3
- Effects mode 4-3, 9-1
  - Selecting effects 9-2
  - Soft buttons 9-1
- Effects mode page 9-1
- Effects mode parameters 9-2
- Effects return 18-3
- Electrical grounding 2-2
- Enable layer 6-12
- Enable MIDI channels 10-9
- Envelope control
  - Adjust 6-28
  - Key tracking 6-28
  - Source and depth 6-28
  - Velocity tracking 6-28
- Envelope control page 6-27
- Envelopes 19-4
- Envelopes 2 and 3 6-27, 25-11
- EPS 15-32
- EQ
  - Bass tone control 14-27
  - Mid-range parametric 14-26
  - Parametric 14-22
  - Steep bass tone control 14-29
  - Treble tone control 14-28
- Equalization 14-22, 14-30
- Erase
  - Song Mode TRACK Page 12-34
- Evaluation of FUNs 16-16
- EVENT Page
  - Song Mode 12-28
- Everything
  - loading objects as 13-20
- Everything files 13-23
- EXIT button 3-4
- Exporting data 13-65
- Expression (MIDI 11) 25-5
- Extended program changes 10-11



Extended sample loop 15-12  
Extensions  
    used by or accepted by the K2500 13-7  
Extra mono dry output 11-3

## F

F1-F3 pages 6-17  
F4 AMP page 6-17  
Farm A-1  
File Backup 13-61  
File Copy 13-63  
File Formats  
    supported by K2500 13-65  
File Index 13-8  
File List Dialog 13-6  
Files  
    deleting 13-61  
    Everything 13-23  
    Loading 5-5, 13-13  
    Master 13-23  
    moving between directories 13-59  
    renaming 13-60  
    Saving 5-5  
    saving 13-22  
    Split 13-23  
Fill Mode 13-18  
Fill soft button 13-19  
Filters 14-5  
    Allpass 14-15  
    Bandpass 14-18  
    Double notch 14-20  
    Highpass 14-12  
    Lowpass 14-7  
    Notch 14-17  
    Twin peaks bandpass 14-21  
Filters—A review 14-6  
Find Files 13-56  
Finding objects 3-6  
Fine adjust 6-8  
Fine Hz 6-17  
Floppy disk drive maintenance 28-1  
Floppy disks 13-1  
Floppy disks, formatting 13-2  
Foot (MIDI 04) 25-4  
Format 15-8  
Formatting floppy disks 13-2  
Formatting SCSI devices 13-4  
Four-pole highpass filter 14-13  
Four-pole lowpass filter 14-10

Free Utility 13-58  
Freeze pedal 6-13  
Freeze pedal (MIDI 69) 25-5  
Front panel MIDI 3-7  
Front panel navigation 3-1  
Front panel reference 20-1  
FUN diagram 16-3  
FUN equations 16-3  
FUN page 6-31  
FUN1, FUN2 25-11  
FUN3, FUN4 25-11  
Function soft buttons 6-32  
Function Soft Buttons in the Sample Editor 15-9  
FUNS  
    For sample and hold 19-6  
FUNs  
    Chaotic LFOs 16-13  
    Diode equations 16-14  
    Evaluation sequence 16-16  
    Sawtooth LFOs 16-13  
    Warp equations 16-10  
FUNs equation list 16-4, 16-15  
FX bypass 5-6, 20-4  
FX channel 9-4  
FX mode 9-3

## G

GAIN 14-34  
Gain 15-3  
Gated lowpass filter 14-12  
GAttVel control source 25-12, 25-13  
General synthesis concepts 14-1  
GKeyNum control source 25-12  
Glitches  
    with guitar or wind controller 11-5  
Global ASR2 25-7  
Global effects 22-1  
Global FUN2 25-7  
Global FUN4 25-8  
Global LFO phase 25-7  
Global LFO2 25-7  
Global phase 1 and 2 25-9  
Global preset effect 9-5  
Global random variant 1 and 2 25-9  
Globals 6-24  
Glossary entries 1-3  
Grab  
    Song Mode TRACK Page 12-41  
Ground hum 28-3

Grounding 2-2  
Guitar/Wind Controller Mode 11-5

## H

Hard reset 11-5  
Hard sync functions 14-50  
    SYNC M and SYNC S 14-50  
Hardware audio switching 18-3  
Headphones 27-2  
High frequency stimulator 14-39  
High key 6-11  
High velocity parameter 6-11  
Highpass filters 14-12  
HIPAS2 14-13  
HIPASS 14-12  
Hold through attack 6-13  
Hold until decay 6-13

## I

Ignore release 6-13  
Impact  
    ENVCTL page 6-29  
Importing data 13-65  
Importing layers 6-33  
Input 15-2  
Input Quantize 12-20  
Insert 15-23  
    Song Mode TRACK Page 12-35  
    with Macros 13-53  
Insert cables 18-3  
Insert Zero 15-22  
Installation 2-2  
Intonation key 11-3  
Intonation Table Editor 17-1  
Intonation tables 11-2, 24-3  
Introduction to editing 5-1  
Intuitive data entry 3-6  
Inverse attack velocity 25-10  
Invert 15-22

## J

Jump to page 5-6, 20-4

## K

K2000  
    SysEx compatibility 30-1  
K2000 Compatibility Disks B-1  
K2500 Features 1-1, 32-1  
K2500 Overview 1-1  
K2500 Program Farm A-1  
K2500 rear panel 2-2  
Key number 25-10  
Key numbers 24-1  
Key range 15-35  
Key state 25-9  
Key tracking 6-8, 6-14, 6-18, 6-28  
Keyboard  
    using to trigger sampler 15-5  
Keymap 15-37  
    Key tracking 6-14  
    Stereo 6-15  
    Transpose 6-14  
    Velocity tracking 6-14  
Keymap Editor 15-33, 19-9  
    Assign sample 15-35  
    New range 15-35  
Keymap Editor Parameters 15-35  
KEYMAP page 6-14  
Keymap parameter 6-14  
Keymaps 6-2  
Keystart  
    bipolar 14-4  
    unipolar 14-4  
Keytrack start (KStart) 14-3  
KRZ files  
    saving and loading a macro table in 13-54  
Kurzweil program changes 10-13

## L

Large disk files 13-23  
Layer  
    Delete 6-34  
    Duplicate 6-33  
    Hold through attack 6-13  
    Hold until decay 6-13  
    Ignore release 6-13  
    Import 6-33  
    new 6-33  
    Opaque 6-13  
    Set range 6-32

- Layer delay
  - Maximum 6-12
  - Minimum 6-12
- Layer delay control 6-12
- Layer enable 6-12
- Layer enable sense 6-12
- LAYER page 6-10
- Layers
  - moving between in multi-layer programs 3-6
  - Muting 5-6
- LCD 3-1, 11-3
- Legato play 6-23
- Legato switch (MIDI 75) 25-5
- LF SAW 14-37
- LF SIN 14-37
- LF SQR 14-37
- LFO page 6-29
- LFO shapes 23-1
- LFO1 25-11
- LFO1 phase 25-11
- LFO2 25-11
- LFO2 phase 25-11
- LFOs 6-29
  - Maximum rate 6-29
  - Minimum rate 6-29
  - Phase 6-30
  - Rate control 6-30
  - Shape 6-30
- Library 13-5
- Library Disk 13-47
- Line cord 2-2
- List Utility 13-57
- Load Function Dialog 13-18
- Loading a Macro File 13-43
- Loading Dependents of Selected Objects 13-17
- Loading files 5-5, 13-13
- Loading Individual Objects 13-13
- Loading Objects from Floppy Disk Files 13-18
- Loading Selected Entries from a Macro File 13-49
- Loading Songs From Disk 12-50
- Local control 10-2
- Local Keyboard Channel 2-4, 10-4
  - with Song Mode 12-1
- Lock parameters 10-10
- LOOP 15-15
- Loop state 25-12
- Looping amplitude envelopes 6-26
- LOPAS2 14-9
- LOPASS 14-7
- Low frequency waveforms
  - Coarse adjust 14-37
  - Sawtooth 14-37
  - Sine 14-37

- Square 14-37
- Low key 6-11
- Low velocity parameter 6-11
- Lowpass filter with clipping 14-46
- Lowpass filters 19-4
- LP2RES 14-10
- LPCLIP 14-46
- LPGATE 14-12

## M

- Macro Insert 13-53
- Macro Load
  - aborting 13-54
- Macro On Indicator 13-5
- Macros 13-38
  - editing 13-50
- Main Control Source list 25-6
- Maintenance and Prevention 28-1
- Mark pages 5-6, 20-4
- MASTER button 5-6
- Master files 13-23
- Master mode 4-4
- Master mode page 11-1
- Master mode—Soft buttons 11-4
- Master panning 11-2
- Master Transpose 11-1
- Maximum layer delay 6-12
- Maximum LFO rate 6-29
- Memory bank ID allotments 21-10
- Memory banks 5-5
- Memory display 11-4
- Memory Limits 12-50
- Memory management 27-1
- Merge soft button 13-19
- Microtonal tunings 11-2
- MIDI
  - All Notes Off 10-5
  - Audio output gain 10-10
  - Audio output groups 18-4
  - Audio output pair 10-9
  - Basic channel 10-5
  - Channel enable 10-9
  - Key and note numbers 24-1
  - LED 3-3
  - Pan 10-9
  - Parameter locks 10-10
  - Program change formats 10-10
  - Program changes 10-13
  - Receive mode 10-5

- Reset channels 10-18
  - Sample dumps 29-3
  - Sending from front panel 3-7
  - Song recording 12-50
  - Transmit parameters 10-1
  - Volume 10-9
  - MIDI button 5-6
  - MIDI channel 10-2
  - MIDI channel parameters 10-8
  - MIDI connections 2-2
  - MIDI control 10-2
  - MIDI control parameters 2-6
  - MIDI Control Source list 25-3
  - MIDI controls 10-4
  - MIDI Implementation Chart 32-4
  - MIDI loop 10-2
  - MIDI mode 4-4
    - Soft buttons 10-18
  - MIDI pitch bend 10-3
  - MIDI program changes 10-3, 10-16
  - MIDI Receive page 10-8
    - Power Mode 10-8
  - MIDI Receive parameters 10-5
  - MIDI sample dump standard
    - Aborting 29-6
    - Loading 29-4
    - New samples 29-5
    - Troubleshooting 29-6
  - MIDI Type 0 Files 13-66
  - MIDIScope 11-4
  - Mid-range parametric EQ 14-26
  - Minimum layer delay 6-12
  - Minimum LFO rate 6-29
  - Mirror Image Drum Map C-11
  - MISC Page
    - Song Mode 12-18
  - Miscellaneous (MISC) 15-11
  - Mix 15-23
  - Mix Beat 15-27
  - MIX outputs 18-1
  - MIX Page
    - Song Mode 12-23
  - Mixers 14-34
  - Mixers with non-linear inputs 14-49
    - !AMP 14-49
    - AMP MOD 14-49
    - x AMP 14-49
    - x GAIN 14-49
  - Mixing outboard effects with the K2500's 18-3
  - Mod Wheel (MIDI 01) 25-4
  - Mode 15-4
  - Mode buttons 3-1
  - Mode buttons—Program and Setup Editors 20-3
  - Mode Selection 3-1
  - Mode selection 4-1
  - Modes 4-1
    - Using 4-3
  - Monitor (Mon) 15-5
  - Mono pressure 25-6
  - Mono sound systems 2-2
  - Monophonic programs 6-22
  - Move
    - on Object Utilities page 11-7
  - Moving Files Between Directories 13-59
  - MS-DOS File System Compatibility 13-65
  - Multi-disk files 13-23
  - Multiple Object Selector
    - entering selection criteria in 13-34
    - soft buttons 13-33
  - Multiple Object Selector Page 13-32
  - Multiple Selection of Files to Load 13-21
  - Multi-velocity keymaps 15-36
  - Music workstation ideas 27-2, 30-1
  - Mute
    - Layers 5-6, 20-3
    - Setup zones 5-6, 20-3
  - Muting tracks 12-24
- ## N
- Name
    - on Object Utilities page 11-9
  - Name Table 13-27
    - not loading 13-31
  - Naming objects 5-2
  - Natural amplitude envelope 6-24
  - Navigation 3-1
  - Negative A clock 25-9
  - Negative B clock 25-9
  - Nested editors 4-2
  - New layer 6-33
  - Noise generator 6-16
  - Noise prevention and reduction 28-2
  - Non-linear functions 14-38
    - 2PARAM SHAPER 14-44
    - DIST 14-40
    - HIFREQ STIMULATOR 14-39
    - LPCLIP 14-46
    - PWM 14-46
    - SHAPE2 14-44
    - SHAPER 14-42
    - WRAP 14-45
  - Normalize 15-19

Notch filters 14-17  
NOTCH2 14-18  
Note numbers 24-1  
Note numbers, percussion keymaps 24-1  
Note State 20 25-9  
Note triggering 6-12  
Numeric entry 3-4

## O

Object type and ID 5-2  
Object Utilities 11-6  
    accessing from the editor 11-10  
Objects 5-1  
    auditioning 13-25  
    Deleting 5-4, 11-4  
    loading individually 13-13  
    Naming 5-2  
    Numbering in memory banks 21-10  
    RAM 5-2, 5-4  
    RAM, viewing 27-1  
    Renaming 5-3  
    ROM 5-2, 5-4  
    Saving 5-2  
    saving individually 13-24  
    soft button on Utility page 11-4  
    SysEx values 30-6  
Objects, storing 21-10  
Octave buttons don't work 6-3  
OK  
    soft button 13-10  
One-pole allpass filter 14-15  
One-pole highpass filter 14-12  
One-pole lowpass filter 14-7  
Opaque layer 6-13  
Open  
    soft button 13-9  
Orchestral ROM C-1  
Orchestral ROM Effects C-1  
Orchestral ROM Keymaps C-3  
Orchestral ROM Programs C-2  
Orchestral ROM Programs with Controller Assignments C-5  
Orchestral ROM Samples C-4  
Orchestral ROM Setups C-10  
Output  
    Assignment and control 6-18  
    Double-wire algorithms 6-20  
    Gain 6-19  
    Pan mode 6-19

Panning 6-19  
    Selecting output group 6-19  
    Stereo keymaps 6-20  
Output gain 10-10  
OUTPUT page 6-18  
Output pair 10-9  
Overview 1-1  
Overwrt soft button 13-19  
OvFill soft button 13-19

## P

Pad 6-18, 14-3  
Page Buttons 15-11  
Pages 3-2  
    Algorithm (ALG) 6-9  
    Amplitude envelope (AMPENV) 6-24  
    ASR 6-30  
    COMMON 6-22  
    EFFECT 6-21  
    Effects mode 9-1  
    Envelope control 6-27  
    Envelopes 2 and 3 6-27  
    F1-F3 6-17  
    F4 AMP 6-17  
    FUN 6-31  
    Jumping to 5-6  
    KEYMAP 6-14  
    LAYER 6-10  
    LFO 6-29  
    Marked 5-6, 20-4  
    Master mode 11-1  
    MIDI CHANLS 10-8  
    MIDI RECV 10-5  
    MIDI XMIT 10-1  
    OUTPUT 6-18  
    PITCH 6-7, 6-16  
    Previous 5-6  
    VTRIG 6-32  
Pan 6-19  
    Master 11-2  
    MIDI 10-9  
Pan (MIDI 10) 25-4  
Pan control 25-8  
Pan lock 10-10  
Pan Position  
    MIX page 12-23  
Panel play feature 3-7  
Panic (MIDI 123) 25-6  
Panic button 2-5, 10-18

- PANNER 14-32, 19-7
- PARA BASS 14-27
- PARA MID 14-26
- PARA TREBLE 14-28
- Parameter locks 10-10
- Parametric EQ 14-22
- Parent
  - soft button 13-9
- Parentheses around program name 6-2, 11-2
- Patch Through
  - with Song mode 12-1
- Path 13-4
- PCH 10-12
- Performance controls 2-6
- Phase of LFO waveforms 6-30
- Pitch
  - Fine Hz 6-17
- Pitch bend
  - MIDI 10-3
- Pitch bend in setups 7-4
- Pitch bend mode 6-11
- Pitch bend range 6-22
- Pitch control 14-31
- PITCH page 6-7, 6-16
- Pitch Shift 15-26
- Pitch Wheel 25-7
- Pitch, amplitude, panner 14-31
- Playback mode 6-16
- Playing notes from front panel 3-7
- Playing songs 12-49
- Plus/Minus buttons 3-4
- Polyphonic pressure 25-10
- Polyphony 6-2, 11-4
- Portamento 6-23
- Portamento click 6-23
- Portamento rate 6-23
- Portamento switch (MIDI 65) 25-5
- Portamento time (MIDI 05) 25-4
- Power cable 2-2
- Power Mode
  - MIDI Receive page 10-8
- Power problems 28-4
- Preset effect 6-21
- Preset effects in setups 7-3
- Pressure Map Editor 17-6
- Pressure maps 10-3, 10-6
- Pressure sensitivity 11-2
- Previous page 5-6, 20-4
- PROGRAM button 5-6
- Program change formats 10-10
- Program change type 10-2, 10-6
- Program changes
  - Extended 10-11

- Kurzweil 10-13
- MIDI 10-3, 10-13, 10-16
- Quick Access banks 10-13, 10-16
- Quick Access mode 8-1
- Program changes in Setup mode 7-4
- Program Editor
  - Pages 6-9
  - Soft buttons 6-4
- Program Farm A-1
- Program list 21-1
- Program lock 10-10
- Program mode 2-4, 4-3, 6-1, 6-2
  - Soft buttons 6-3
- Program mode page 6-2
- Program RAM vs. Sample RAM 27-1
- Program structure 6-2
- Program, monophonic 6-22
- Programming algorithms 14-1
- Programming FUNs 16-2
- Programs 2-4
  - Deleting 6-33
  - SysEx dumps 6-33
- Programs not playing 11-2
- Punctuation 5-3
- PWM 14-46

## Q

- QA 0-127 10-17
- Quantize
  - Input 12-20
  - Song Mode TRACK Page 12-36
- Quick Access bank program changes 10-13, 10-16
- QUICK ACCESS button 5-6
- Quick Access Editor 8-1
- Quick Access mode 2-5, 4-3, 8-1
- Quick Mute
  - Song Mode MIX Page 12-24
- Quick Mute Track Bank Indicator
  - MIX page 12-24
- Quick Scrolling to Subdirectories 13-10

## R

- RAM
  - Sample vs. Program 27-1
- RAM objects 5-2, 5-4
- RAM objects, viewing 27-1

- Random variants 1 and 2 25-11
- Rate 15-3
- Rate control of LFOs 6-30
- Reading Samples 15-31
- Realtime effects 6-21
- Rear panel 2-2
- Recording Multi-timbral Sequences via MIDI 12-50
- Recording songs via MIDI 12-50
- Reference Quantize
  - Song Mode TRACK Page 12-37
- Region/Criteria window
  - Song mode 12-31
- Release segment 6-26
- Release state 25-12
- Release time for ASRs 6-31
- Release velocity 25-10
- Releasing ASRs 6-31
- Relink-by-Name 13-28
- Remap
  - Song Mode TRACK Page 12-41
- Removing layers 6-34
- Renaming Files 13-60
- Renaming objects 5-3
- Repeating ASRs 6-31
- Resample 15-25
- Reset
  - hard 11-5
- Reset MIDI channels 10-18
- Reverse 15-21
- Reverse samples 6-16
- Roland 15-32
- ROM objects 5-2, 5-4
- Root
  - soft button 13-9

## S

- Safety precautions 2-2
- Sample 15-5
  - Playback loops 6-16
  - Playback mode 6-16
  - SMDI transfers 29-7
  - Stereo 6-15
- Sample and Hold 19-6
- Sample and hold 19-6
- Sample and Hold with FUNs 19-6
- Sample dumps 29-3
- Sample Editor 15-37
- Sample ID offset 29-6
- Sample playback rate 25-12
- Sample RAM vs. Program RAM 27-1
- Sample recording
  - aborting 15-6
- Sampler
  - triggering from keyboard 15-5
- Samples
  - Adjusting volume 15-36
  - Assigning to keymaps 15-36
  - triggering from audio signals 15-4
  - Tuning 15-36
- Sampling Analog Signals 15-2
- Sampling Digital Signals 15-7
- Save dialog 5-2
- Saving and naming 5-2
- Saving Dependent Objects 13-26
- Saving Files 13-22
- Saving files 5-5
- Saving Individual Objects 13-24
- Saving Master and Everything Files 13-23
- Saving objects 5-2
  - RAM 5-2, 5-4
  - ROM 5-2, 5-4
- Saving RAM objects 5-2, 5-4
- Saving ROM objects 5-2, 5-4
- SAW 14-36
- Sawtooth LFOs (FUNs) 16-13
- SCSI 1-1
- SCSI connections 2-3
- SCSI devices 13-1
  - Formatting 13-4
  - Termination 13-3
- SCSI Guidelines 29-1
- SCSI ID 10-6
- SCSI IDs 13-4
- SCSI Musical Data Interchange 1-1
- Search function 3-6
- Select soft button 13-8
- Selecting a Song for Playback 12-49
- Selecting effects 9-3
- Selecting modes 4-1
- Selecting parameters 3-1
- Selection by Dependents 13-36
- Selection by Type/Range 13-35
- Separate outputs 18-2
- Sequencer
  - tutorial 12-1
- Setting layer ranges 6-32
- Setting Up For Sampling 15-1
- SETUP button 5-6
- Setup control parameters 2-6
- Setup Editor 7-2
  - Soft buttons 7-4
- Setup Editor Parameters 7-2

- Setup mode 4-3, 7-1
- Setups 2-4
  - Control mode 7-3
  - Effect 7-3
  - Effects mix 7-3
  - list of 21-9
  - Low and high key 7-3
  - MIDI channel 7-3
  - Muting zones 5-6
  - Pitch bend 7-4
  - Program assignment 7-3
  - Program change 7-4
  - Selecting via MIDI 10-13, 10-16
  - Transpose 7-3
  - Transposing 7-1
- SHAPE MOD OSC 14-48
- SHAPE2 14-44
- SHAPER 14-42, 19-7
- Shift
  - Song Mode TRACK Page 12-38
- Signal-to-noise ratio 2-3
- SIMMs 1-1
  - for Sample RAM 27-2
- SINE 14-36
- Single-output algorithm 6-5
- Sleep soft button 13-6
- SMDI 1-1
- SMDI sample transfers 29-7
- Soft buttons 3-2, 6-3, 6-4
  - Disk Mode page 13-5
  - Effects mode 9-1
  - Keymap Editor 15-34
  - Master mode 11-4
  - MIDI mode 10-18
  - Setup Editor 7-4
  - Special functions 6-32
- Soft Buttons on the DSP Page 15-17
- Soft pedal (MIDI 67) 25-5
- Soft Through
  - with Song Mode 12-1
- Soloing current track 12-25
- SONG button 5-6
- Song Mode 12-1
  - MISC Page 12-18
  - MIX Page 12-23
- Song mode 4-4
- Song playback 12-49
- Songs
  - Loading 12-50
  - Memory limits 12-50
  - Recording via MIDI 12-50
  - Synchronizing 12-49
- Sostenuto (MIDI 66) 25-5
- Sostenuto pedal 6-13
- Source 1 6-8
- Source 2 6-8
- Special button functions 5-6
- Specifications
  - K2500 32-1
- Split files 13-23
- SQUARE 14-36
- Src parameter
  - Sample page 15-3
- Startup 2-1, 13-5
- Startup File
  - creating 13-63
  - deleting banks in 13-64
- Stealer
  - soft button on Utility page 11-4
- Steep bass tone control 14-29
- STEP Page
  - Song Mode 12-43
- Stereo insert cables 11-3, 18-3
- Stereo keymaps 6-15, 6-20
- Stereo samples 6-15
- Storing Objects in Memory Banks 21-10
- Sustain (MIDI 64) 25-5
- Sustain not working 10-5
- Sustain pedal 6-13
- Sustaining ASRs 6-31
- SW+DST 14-47
- SW+SHP 14-48
- SYNC M and SYNC S 14-50
- Sync state 25-9
- Synchronizing songs 12-21, 12-49
- Synthesis—General concepts 14-1
- System Exclusive 10-3
  - Button press values 30-7
  - Common format 30-1
  - compatibility with K2000 30-1
  - Data formats 30-1
  - Master parameters 30-7
  - Messages 30-2
  - Object types 30-6
- System exclusive 6-33
- System Exclusive dumps 11-10
- System Exclusive ID 10-6
- System Exclusive implementation 30-1

T

- Terminating SCSI devices 13-3
- Thin



- Song Mode TRACK Page 12-40
- Threshold (Thresh) 15-4
- Timbre shift 6-15
- Time 15-5
- Time Warp 15-26
- Top line of display 3-2
- Total size of all files 13-10
- Track mute 12-24
- TRACK Page
  - Song Mode 12-31
- Track Status Indicators
  - MIX page 12-24
- Transpose
  - Master 11-1
  - MIDI 10-2
  - Setups 7-3
  - Song Mode TRACK Page 12-39
- Transposing setups 7-1
- Treble tone control 14-28
- Triggering ASRs 6-31
- Triggering notes on startup 6-12
- Triggering sampler from keyboard 15-5
- Triggering samples from audio signals 15-4
- TRIM 15-13
- Troubleshooting 28-4
- Truncate 15-19
- Tuning samples 15-36
- Tuning to other instruments 11-1
- Twin peaks bandpass filter 14-21
- Two-parameter shaper 14-44
- Two-pole allpass filter 14-16
- Two-pole bandpass filter 14-18
- Two-pole bandpass filter, fixed width 14-20
- Two-pole highpass filter 14-13
- Two-pole lowpass filter 14-8
- Two-pole notch filter 14-17
- Two-pole notch filter, fixed width 14-18

## U

- Unipolar control sources 16-1
- Unipolar keystart 14-4
- Upper and lower amp 14-32
- User amplitude envelope 6-24
- Using the Digital Outputs 15-8
- Using the MIX Outputs 18-1
- Using the modes 4-3
- Using the Velocity Map Editor 17-3
- Using your K2500 in a SCSI System 13-4
- Utilities 11-4

## V

- V Parameter
  - on Setup Editor page 7-4
- V parameter
  - MIDI Receive page 10-4
- Variable Architecture Synthesis 1-1, 1-2
- VAST 1-1, 1-2
- Velocity crossover 15-36
- Velocity Map Editor 17-3
- Velocity maps 10-3, 10-6
- Velocity sensitivity 11-2
- Velocity stabs 19-1
- Velocity tracking 6-8, 6-14, 6-18, 6-28
- Velocity triggers 6-32
- Velocity triggers 1 and 2 25-10
- Ventilation 2-2
- View soft button 2-4
- Viewing RAM objects 27-1
- Viewing Selected Objects 13-25
- Voice allocation 11-4
- Voice channels 6-2
- Voltage levels 2-2
- Volume
  - MIDI 10-9
- Volume (MIDI 07) 25-4
- Volume Adjust 15-20
- Volume control 25-8
- Volume Knob 2-6
- Volume Level
  - MIX page 12-23
- Volume lock 10-10
- Volume Ramp 15-24
- VTRIG page 6-32

## W

- Warp equations 16-10
- WAVE Files 13-66
- Waveform wraparound 14-45
- Waveforms 14-35
  - Sawtooth 14-36
  - Sine 14-36
  - Square 14-36
- Waveforms for LFOs 6-30
- Waveforms with non-linear functions 14-47
  - + SHAPE MOD OSC 14-48
  - AMP MOD OSC 14-48
  - SHAPE MOD OSC 14-48

SW+DIST 14-47  
SW+SHP 14-48  
x SHAPE MOD OSC 14-48  
Wet/Dry mix 6-21  
Wet/dry mix 9-2  
Wind controller  
controlling K2500 with 11-5  
WRAP 14-45  
Wrong sample being dumped 29-6

## **X**

x AMP 14-49  
x GAIN 14-49  
x SHAPE MOD OSC 14-48  
XFADE 14-34  
XMIT page 10-1