Hawthorne Technology

TinyGiant HT68k

MANUAL

68000 TinyGiant

Low Cost 16-bit SBC and Operating System

by Art Carlson

There has been a lot of interest in a low priced 68000® single board computer for experimenting or process control, and a group of us got together at SOG last year to discuss the situation. We are interested in the 68000 because it is big, fast, cheap, and well supported. The 86 family is dominated by MS-DOS/PC-DOS® machines which are designed as office machines, and the other 32 bit micro processors are not very well supported. We wanted to work with a processor for which books are available, and we also wanted other companies to provide software for the new machine.

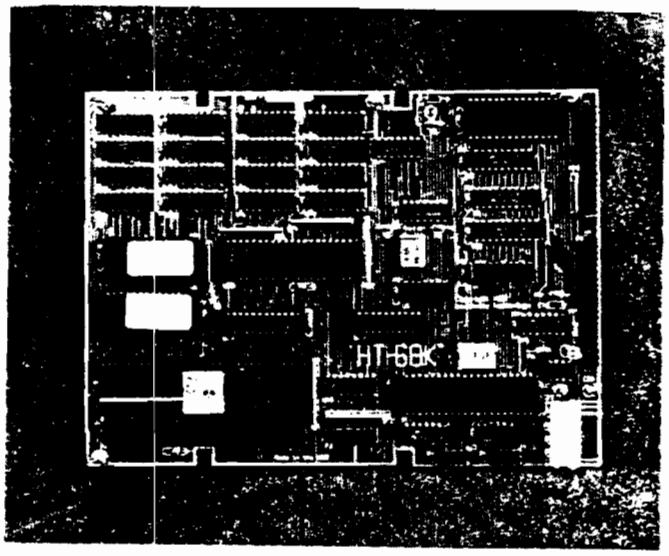
The major problem in experimenting with the 68000 has been the lack of a readily available low-cost operating system and programming utilities. As a result of our discussions, Joe Bartel (Hawthorne Technology) offered to supply a single-user version of his 68000 operating system complete with the source code and the compiler for his HTPL® language (in which the system is written) for $50. He's even throwing in a line editor and a 68000 assembler.

Once we had the operating system, we started searching for suitable hardware since not all of us are interested in designing and wirewrapping the main CPU board. I understand that some people are porting Hawthorne's K-OS ONE® operating system over to the Atari ST® (which is a great hardware bargain), but we also need a small low cost board which can be mounted on robotic arms and other places where an ST is too big or is technically unsuitable. After all, for many controller applications we don't need sound, graphics, drives, keyboard, or monitor; but we do need complete bus access so that we can provide lots of I/O. Joe came up with another winner which is the 68000 single board computer he calls "The TinyGiant® ."

Every computer is designed to fill a particular goal or market niche, and The TinyGiant was designed to be as low cost as possible while still providing the needed features. The target price was set at $395, and when a choice had to be made they looked at the original goal. Then they looked at what was needed to make a usable small developement system. There had to be a disk of some kind. There also had to be enough I/O to support a terminal and some kind of printer and a modem. They didn't put a video controller on the board because they felt that many of these would be used in places where video was not needed. Also video would have made the board much larger and more costly. Dynamic RAMs were used because of the lower cost.

The TinyGiant is a complete 68000 system on a single board. The main component is the 68000 that runs at 8 Mhz. with no wait states. There are two EPROMS, a dynamic RAM controller, four RAM chips, and a 68681® dual serial chip. There is also an unbuffered expansion bus provided.

The 68681 is almost a complete I/O system on a single chip. There are two serial ports and a timer on the 68681. There are also some extra input and output bits that can be used. Each serial port has its own internal baud rate generator that can be set from 50 to 9600 baud. The system startup assumes that the console terminal on the first port is set to 9600 baud. The other port is assumed to be for a modem and is set to 1200 baud. This port could be used to support a serial printer if a modem is not needed. The timer is used as the basic system timer for a time of day clock. When the PRN device is selected, a parallel printer port made from a latched buffer is used.



68000 TinyGiant

The 1770® floppy controller can control up to four 5¼ or 3½ inch floppy disk drives. At the present time the software is only supplied on 5¼ inch floppy disks, but it will also be supplied on 3½ disks in the near future. The 1770 controller has an internal clocked digital data separater so no adjustments are needed. There is no DMA controller so all I/O is done with interrupts.

The board is supplied with 128k of RAM but can be expanded to 512k by simply adding the memory chips. The memory is controlled with an MB1422® dynamic RAM controller which takes care of all the timing and refresh for the 64k x 4 dynamic RAMS. There are four RAM chips in the basic machine, 12 more can be added on board, and an expansion board can be added to provide additional memory. There are two EPROMS that contain the boot code and a very simple hardware debug monitor, and there is a lot of empty space in the PROMS that could later be used for code.

There are two expansion connectors that provide all the needed address, data, and control lines. The 64 pin expansion bus connector has all the raw signals just as they are on the 68000 itself and it is an unbuffered bus which means that buffers have to be provided for the bus lines on the first expansion board. The other expansion connector has some signals derived from logic on board. There is a priority interrupt encoder that is

connected to autovector interrupts.

To use the TinyGiant you have to connect a power supply, a terminal, and a disk drive. The board requires only 5 volts at about 2 amp and some at +12 volts for the RS-232 (the −12 for the RS-232 is generated on board), and the power connector is the same as a floppy drive. The EPROM monitor is set up to boot the K-OS ONE operating system from a floppy disk when the power is applied. All of this should be in a proper case, and any of the cases for the Ampro Little Board® should work fine because the board is the same size. The cases and power supplies for the PC clones should also work and are available at very low prices.

The software that comes with the TinyGiant has many features that will remind you of CP/M® or MS-DOS. The disks are compatible with MS-DOS which means that you can move data files from your new machine to your PC easily. The standard command processor is modeled after MS-DOS but does not have batch files. The editor is a line editor but is much better than ED® or EDLIN® . The assembler is a two pass absolute assembler that has include files but no macros. The compiler is for HTPL, the language used to write the K-OS ONE operating system.

People who currently use a Z-80® and are interested in moving up will be interested in the board. Also anyone who wants to learn about the 68000 without having to dig through a difficult operating system will like it. The price of the board is competitive with the 80 and 86 family boards and so is the price of the software. The small size and power make it great for control applications. The use of MS-DOS compatible disks makes it very convenient for data capture. The operating system makes it good for a personal computer or for experimenting.

At the present time there are a limited number of programs that will run under K-OS ONE. This will improve in the next few months as the companies that purchased the first copies of K-OS ONE bring their products to market. For any new system there is a delay between when it is first delivered and when lots of software is available for it. Some of the software that will be available is code that is being ported from other systems.

For languages there is the assembler and the HTPL compiler that come with the basic machine, and the operating system is written in HTPL so that it can be recompiled on the system. Most of the development is being done in HTPL, but the source code for the runtime library is written in assembler and is supplied in source form. For many applications most of the code can be written in HTPL with the speed critical portions written in assembler. Also there are some functions that are much easier to write or much smaller in assembler. I understand that a C compiler is being ported over to the system.

Externally, the operating system can look like anything you want it to. The command processor provided looks a lot like MS-DOS but you can change it because it is a separate program like the CCP is in CP/M and you are provided with the source code. In a dedicated application the command processor could be replaced by a menu to load a set of application programs. A command processor could also be written that would make it look like a Unix system was being used if that was desired.

Internally, the operating system resembles Unix® or MS-DOS 2.0 except that it is greatly simplified. All system calls are done in a manner similar to MS-DOS where the required information is placed in a parameter block and then a trap instruction gives control to the system. All error information is returned in memory in the parameter block. This makes it very easy to use system fuctions directly in a high level language. The use of Unix like files instead of file control blocks like CP/M means that any application written will still run even if there are radical changes to the internal structure of the disks.

HT68K TinyGiant Hardware Description

There are 6 connectors on the TinyGiant board that connect it to the real world. Two of these are for expansion, one for the console serial port, one for the auxillary serial port, one for the printer, and finally one for the disk drives.

The size and shape of the board is such that it can be bolted to the side of a standard 5¼ inch floppy disk drive, and the power connector for the +5 and +12 is the same as used on a floppy disk drive. The +12 volts is converted to −12 for RS-232 by an onboard power converter. Most of the cheap PC clone power supplies already have the connectors wired for floppy disks so one of them can be used.

The expansion bus is divided between two connectors, P1 and P2. The P1 connector brings out all the basic 68000 control signals. These come directly from the 68000 cpu chip and are not buffered. The remaining signals come from the P2 connector. Any signal should be buffered before it is used on an expansion board.

When power is applied the PROM is at location 0. As soon as the serial port is addressed the PROM is shifted into high memory. This is taken care of in the boot PROM so all the user will see is RAM memory at location 0. The BIOS is in PROM so it is assumed to run at $380000. All interrupts are auto vectored, some are dedicated to devices on the board and some are available on the expansion connector.

P1 CONNECTOR -- EXPANSION

1 = D4	33 = D5
2 = D3	34 = D6
3 = D2	35 = D7
4 = D1	36 = D8
5 = D0	37 = D9
6 = AS*	38 = D10
7 = UDS*	39 = D11
8 = LDS*	40 = D12
9 = R/W*	41 = D13
10 = DTACK*	42 = D14
11 = BG*	43 = D15
12 = BGACK*	44 = GND
13 = BR*	45 = A23
14 = VCC, 5 V	46 = A22
15 = CLK, 8 MHZ	47 = A21
16 = GND	48 = VCC, 5V
17 = HALT*	49 = A20
18 = RESET*	50 = A19
19 = VMA*	51 = A18
20 = E	52 = A17
21 = VPA*	53 = A16
22 = BERR*	54 = A15
23 = IPL2*	55 = A14
24 = IPL1*	56 = A13
25 = IPL0*	57 = A12
26 = FC2	58 = A11
27 = FC1	59 = A10
28 = FC0	60 = A9
29 = A1	61 = A8
30 = A2	62 = A7
31 = A3	63 = A6
32 = A4	64 = A5

```
P2 CONNECTOR -- EXPANSION

 1 = EXIO2*          9 = EXIO1*
 2 = +12            10 = DREQ*
 3 = -12            11 = NMI*
 4 = VCC, 5 V       12 = EXIR1*
 5 = VCC            13 = EXIR5*
 6 = GND            14 = EXIR6*
 7 = GND            15 = GND
 8 = GND            16 = 16 MHZ CLOCK
P3 CONNECTOR -- DISK

 1 = GND             2 = NC
 3 = GND             4 = NC
 5 = GND             6 = DS0
 7 = GND             8 = IP
 9 = GND            10 = DS3
11 = GND            12 = DS2
13 = GND            14 = DS1
15 = GND            16 = MO
17 = GND            18 = DIR
19 = GND            20 = STEP
21 = GND            22 = WD
23 = GND            24 = WG
25 = GND            26 = TK0
27 = GND            28 = WP
29 = GND            30 = RD
31 = GND            32 = SIDE SEL
33 = GND            34 = NC

P4 CONNECTOR -- TERMINAL SERIAL CONNECTOR
 1 = GND             6 = NC
 2 = RXD             7 = GND
 3 = TXD             8 = NC
 4 = CTS             9 = NC
 5 = RTS            10 = NC

P4 CONNECTOR -- MODEM SERIAL CONNECTOR
 1 = GND             6 = NC
 2 = RXD             7 = GND
 3 = TXD             8 = DCD
 4 = CTS             9 = NC
 5 = RTS            10 = RTS

PR CONNECTOR -- PRINTER
 1 = STROBE
 2 = D0
 3 = D1
 4 = D2
 5 = D3
 6 = D4
 7 = D5
 8 = D6
 9 = D7
10 = ACK -- BUSY

MEMORY MAP

$000000 - $07FFFF   RAM ON BOARD
$080000 - $08FFFF   X1
$100000 - $10FFFF   X2
$180000 - $17FFFF   PRT
$200000 - $27FFFF   SLK- FDC
$280000 - $2FFFFF   SIO
$300000 - $37FFFF   FDC
$380000 - $3FFFFF   PROM ON BOARD

INTERRUPT LEVELS
0 = not an interrupt at all
1 = external
2 = Printer, internal
3 = Serial, internal
4 = Floppy disk, internal
5 = external
6 = external
7 = NMI, external
```

Where Do We Go From Here?

It's very likely that the 68000 will replace the Z80 as the choice for hobby and experimental projects, and the combination of the TinyGiant and the K-OS ONE operating system will provide a very good development platform.

The lack of software is a major problem with any new hardware or operating system — developers won't write the programs until a lot of the systems are sold, and people won't buy the systems until a lot of software exists — at least that's true with a mass market appliance type computer. But this is initially aimed at a different market, and I believe that Hawthorne has made several very smart moves. For one, they're supplying the system with the board, and the system (whether obtained separately or with the board) includes the OS source code plus the language compiler, a 68000 assembler, and a line editor so that you have the utilities to start writing programs. Another very good move is that even though Hawthorne is selling a single board computer, they still want to sell their operating system to hardware companies even if they are directly competing with Hawthorne! Joe has also been in contact with a lot of potential 68000 programers who were frustrated with the lack of a low cost operating system, and there will soon be a rapidly expanding base of software.

One of the most significant points is that you get the operating system with the source code (plus the other goodies) for only $50. Has any one else ever supplied the OS source code? I doubt it, because many CP/M hardware vendors wouldn't even supply the source code for their customized CBIOS. With the low cost OS and its source, you can customize it for your own particular application, and it's cheap enough so that you can include it with your software.

I feel that the use of K-OS ONE is going to expand very rapidly, and we at TCJ are going to fully support this growth. ■

68000 Reference Book List

68000 16/32-bit Microprocessor Programmer's Reference Manual, by Motorola, Prentice Hall

68000, 68010, 68020 Primer, by Kelly-Bootle and Fowler, The Waite Group, Howard W. Sams

The 68000: Principles and Programming, by Leo Scanlon, Howard W. Sams

68000 Assembly Language: Techniques for Building Programs, by Donald Krantz and James Stanley, Addison-Wesley Publishing Company

68000 Assembly Language Programming, by Lance Leventhal, Doug Hawkins, Gerry Kane and William D. Cramer, Osborne/McGraw-Hill

The 68000 Microprocessor: Architecture, Software and Interfacing Techniques, by W. Triebel, and A. Singh, Prentice Hall

68000 Microprocessor Handbook, by William Cramer, AVM Systems, Gerry Kane, Osborne/McGraw-Hill

68000 User's Manual, by J. Carr, Prentice Hall

MC68020 32 bit Microprocessor User's Manual, by Motorola, Prentice Hall

Basic Microprocessors and the 68000, by Ron Bishop & the Motorola Semiconductor Group, Hayden Books

Dr. Dobb's Toolbook of 68000 Programming, Edited, Prentice Hall

Programming the 68000, by Rosenzweig and Harrison, Hayden Books

Programming the 68000, by Steve Williams, Sybex Computer Books

Self Guided Tour through the 68000, by M. Andrews, Prentice Hall

The above books can be ordered from:

Magrathea
8842 Southeast Stark
Portland, OR 97216
Phone (503) 254-2005