MICROKIT

MICROKIT MICROCOMPUTER
DEVELOPMENT SYSTEMS

Contact for
Info and Demo.
NOUSOU-INC.
BOX 667, DIABLO,
CALIFORNIA  94528
(415) 820-2588

Northern California Representative
NOUSOU - Incorporated
BOX 667 - DIABLO - CA  94528
(415) 820-2588

MICROCOMPUTER DEVELOPMENT SYSTEMS
# TABLE OF CONTENTS

1. MICROKIT-8/16 Microcomputer Development Systems  
   for the 8080 and the 6800 ......................................... 1-1
2. MICROKIT-8/16 System Specifications ................................. 2-1
3. MICROKIT-8/16 System Software ....................................... 3-1
   - Monitor/Debugger ................................................... 3-1
   - Editor ................................................................. 3-3
   - Assembler ............................................................. 3-5
4. MICROEMULATOR/8080 In-Circuit Emulator ........................... 4-1
5. OS/8080 Memory Resident Software System ............................ 5-1
6. MICROPRINT/65 Line Printer ........................................... 6-1
7. MICROKIT Disk Operating System ...................................... 7-1
8. System Module Specifications ........................................... 8-1
   - 8080 CPU ............................................................... 8-1
   - 8K RAM Memory ....................................................... 8-3
   - CRT-Keyboard Interface ............................................ 8-5
   - Tape-EIA Interface ................................................ 8-9
9. MICROKIT Word Processor ............................................... 9-1
Fig. 1 MICROKIT-8/16 System
MICROKIT INC. MAY 1976

MICROKIT-8/16 MICROCOMPUTER DEVELOPMENT SYSTEMS

FIRST UNIVERSAL MICROCOMPUTER DEVELOPMENT SYSTEM FOR BOTH THE 8080 AND THE 6800

The MICROKIT-8/16 is a complete stand-alone development system for writing, debugging, and executing programs on the 8080 or the 6800 microprocessor. The system can be ordered as an 8080 based system or a 6800 based system. Conversion packages consisting of a plug-in processor module and software are available so that either system can be switched to handle the other processor.

Either basic system comes complete with an 8K memory, an alphanumeric CRT display, an ASCII keyboard, two cassette tape units, and software. Its excellent software includes a monitor/debugger, editor, and assembler designed to take full advantage of the high-speed multi-line CRT display and the tape I/O.

The MICROKIT-8/16 CRT display holds 960 characters formatted as 24 lines of 40 characters. Since the display is refreshed directly from the microcomputer memory, the full 960 character screen can be written at 20,000 cps. Using the fast CRT the interactive debugging program can generate a full screen hexadecimal memory dump display instantaneously. This kind of quick system response speeds the debugging task.

The MICROKIT editor also makes use of the fast CRT display. The screen based editor is quickly learned and easy to use, because changes are displayed instantly and in full context on the CRT.

To keep cost low, the MICROKIT-8/16 uses a standard television set for the high-speed display, and audio cassette units for mass storage. Because of MICROKIT's proprietary recording technique, data is transferred to the audio cassette units at the rate of 2000 bps with data reliability comparable to digital cassette units.

MICROKIT-8/16 FEATURES

The MICROKIT-8/16 incorporates a number of standard features, some of which are unique among microcomputer systems:

8K bytes of RAM memory—expandable to 32K bytes.

Memory write protect where each 1K page of memory can be write protected under software control.

960 character CRT display refreshed from system memory.
53 key ASCII keyboard with 2-key roll over.

Two audio cassette tape units, fully supported by the software.

A crystal controlled programmable real-time clock with 32 microsecond resolution.

Interrupt driven I/O. All I/O devices including the real-time clock can interrupt the CPU. All interrupts can be masked under software control.

Vectored interrupt.

Bootstrap loader in PROM.

Two EIA RS-232C serial interfaces - one for modem, one for teleprinter.

1 megabyte/second DMA capability.

Six extra card slots for custom interfaces.

OPTIONS AND ENHANCEMENTS

The MICROKIT-8/16 is backed up with a full line of enhancements. These include In-Circuit Emulators, EPROM programmers, add-on RAM or EPROM memories, line printers, "semiconductor disk" memory resident operating systems, and single or dual drive floppy disks with disk operating systems.

With all these peripherals, the MICROKIT-8/16 is also well suited for laboratory and industrial applications such as: 1) a complete inexpensive computer, 2) a programmable controller, or 3) a data acquisition system.

MICROKIT-8/16 HARDWARE CONFIGURATION

The MICROKIT-8/16 system consists of the following elements: mainframe, keyboard, CRT display, and two cassette tape units. The mainframe is packaged in a compact desk-top enclosure and contains power supplies, a card cage, and the four system modules: CPU, RAM, CRT-Keyboard I/O, Tape-EIA I/O.

The MICROKIT system is built around a universal system bus through which all the system modules communicate. The CPU module interfaces to this bus and controls the operation of the system. The 8080 and 6800 are only the first of a series of processors to be supported by the MICROKIT-8/16 system. The modular nature of the system makes it adaptable to other popular 8 and 16 bit
The 8K byte memory of the standard system is contained on one memory module. The system memory is easily expanded by plugging in additional 8K modules. Each 1K page of memory is separately write protectable under software control.

Input/output from the MICROKIT-S/16 is provided by the two I/O modules. The CRT-keyboard I/O module contains the interface logic for the keyboard and the CRT display. The keyboard interface supports the mini-operator console located on the keyboard. The console functions include two switches: "system reset" and "initial program load" (from cassette tape), and two status displays: "run", and "interrupt enabled". All other front panel functions and debugging aids are provided by an interactive debugger through the CRT display.

The second I/O module, which is the Tape-EIA module, contains a number of interfaces as follows:

Tape Interface - The tape interface connects the microcomputer to the two audio cassette tape units.

Dual RS-232C Interface - This allows bit serial communication with both a terminal and a modem. Speeds up to 1200 Baud are possible, under program control.

Crystal Controlled Real Time Clock - The real time clock is used by the RS-232C interfaces for accurate bit timing. The clock is also available for other timing applications.

8-Bit Parallel I/O Port - This TTL port is unassigned and allows for interfacing a user designed device. It provides a bidirectional test port that can interface with user equipment.

Bootstrap PROM - The 256 byte PROM contains a bootstrap loader which may be invoked by the initial load button on the keyboard-console. It loads the system software from cassette tapes.
Fig. 2 MICROKIT KEYBOARD
MICROEMULATOR

The Microemulator provides the design engineer and programmer with editing, assembling and debugging facilities directly into the hardware system that is being designed. The designer/programmer is able to exercise all of the user system I/O devices, memory, interrupt structure and direct memory access facilities through the Microemulator plug. With the Microemulator, the MICROKIT-8/16 will respond to user system interrupts, ready line, reset line, hold line and clocks. It will generate addresses, status signals and strobos and it will send to and receive data from the user system.

The complete Microemulator consists of four distinct modules. These are: 1) the Microemulator logic board, which mounts in the MICROKIT and controls the connection between the microprocessor within the MICROKIT and the user system; 2) the Microemulator plug assembly, which provides the processor at the end of a remote cable that can be plugged directly into the system microprocessor socket; 3) the hardware break single step and PROM programmer interface module, which also mounts in the MICROKIT card cage; and 4) the PROM programming assembly, which is connected via 26 pin flat cable to the PROM interface module.

The Microemulator package can then be considered to provide three separate functions. These are: 1) the extension of the MICROKIT-8/16's microprocessor into the user system; 2) the analysis features of hardware breakpoint and single step; and 3) the ability to program PROMs -- specifically the 2704 (512 x 8) and 2708 (1024 x 8) PROMs. This combination of tools is especially useful in the task of combined hardware and software development. It should be particularly valuable during product development, hardware software integration, production test, and depot maintenance of microprocessor systems. With such features as the MICROKIT RAM operating system or disk operating system, it provides tools unmatched in systems costing many times more.

BASIC OPERATION OF THE MICROEMULATOR

The Microemulator uses the same microprocessor for both the host functions of program development: including source code editing, program assembly, program loading, PROM programming and peripherals management; and the user functions of emulating the microprocessor in the system under test. Commands are included in the microemulator monitor that allow the designer to switch over various of the microprocessor functions into the user system. Thus, there is separate software control of 1) the source of clocks; 2) the source and response to interrupts, reset line and ready line; 3) response to user DMA requests; and 4) memory
mapping functions.

The Microemulator software package is extremely simple to use and easy to learn. In addition to the basic MICROKIT monitor commands (DISPLAY, STORE, EXECUTE, WRITE, FIND and BREAKPOINT SETTING), the Microemulator monitor adds three additional functions: SINGLE STEP, CONTROL (for setting the mode of operation) and TRACE.

During the initial phases of a design program (before the hardware is available), the MICROKIT will be used in the host mode to develop software. Gradually, as the hardware system is brought up, various functions can be switched into the user system.

Thus, first the clocks would be switched over to see if the clock circuits, sync., etc., were working properly.

Second, control would be passed to the user system and response to interrupts, reset and ready line can be tested. User DMA mode would be connected and enabled as required.

Third, a NOP program would be executed to verify microprocessor operation in the user environment.

Fourth, diagnostic programs would be developed and run using the Microemulator feature to test the operation of I/O devices and verify the hardware design.

Fifth, the user system would be exercised with all program and data memory in the MICROKIT using PROM simulator and RAM simulator memory within the MICROKIT system.

Sixth, the memory functions would be switched over to the user system with user RAM and user PROM taking the place of MICROKIT PROM and RAM simulator memory.

Finally, seventh, the MICROKIT would be completely disconnected and the user microprocessor will be installed in the system.

This stepwise approach to debugging hardware and software is extremely effective since each step is a relatively small extension of the preceding step.
MICROEMULATOR DEBUG COMMANDS

Only three commands have been added to the basic Monitor-Debugger described on pages 3-1 and 3-3 of MICROKIT MICROCOMPUTER DEVELOPMENT SYSTEMS. These three commands are:

C P, P, P  Set operating mode (Control). Three parameters can be specified.

C   switch to user clocks.
U   switch to user control lines.
D   enable user DMA request.

T   Begin trace execution at present cursor location.

T addr   Begin trace execution at hexadecimal address "addr".

T$  Recall last trace display.

E(control)   Single-step the program.
MEMORY RESIDENT OPERATING SYSTEM FOR MICROKIT-8/16 MOD 8088
MICROCOMPUTER DEVELOPMENT SYSTEM

OS/8088 is a complete "in memory" operating system for developing 8088 microcomputer programs using the MICROKIT-8/16. OS/8088 provides a program development tool unmatched by any other software development system. It consists of a monitor/debugger, editor and assembler all co-resident in memory along with a 14K byte source code workspace and 4K byte object code workspace. This is sufficient to store a 1000 line source program.

Since the system software, the source program and the object code are simultaneously in memory, the user is able to instantaneously switch from editing to assembling to debugging. This facilitates rapid programming in two ways since (1) overhead for loading programs and data is reduced to zero and (2) the user is able to rapidly test many changes and options.

OS/8088 makes use of MICROKIT’s unique 8K RAM memory boards with write protection registers. This allows each 1K page of memory to be individually protected and unprotected under program control. Without this feature, the user could inadvertently "clobber" his source program during execution and destroy hours of work.

Non-volatile storage of debugged programs (source and object) is provided by the MICROKIT-8/16 system cassette tape units. Loading and dumping to cassette tape takes less than 3 minutes and is usually done only once per session. Thus OS/8088 provides the microcomputer programmer with tools that were previously available only on the largest of computer systems, and makes the MICROKIT-8/16 superior to systems costing more than twice as much. Yet the MICROKIT-8/16 with OS/8088 is still the least expensive 8088 development system with peripherals.

MEMORY CONFIGURATION

OS/8088 runs in 32K of main memory. User programs can occupy the first 4K of memory. (X‘0000’ -- X‘0FFF’) The systems’ programs, Editor work area, and Symbol Table occupy the region from X‘1000’ to X‘7FFF’.

OS/8088 SOFTWARE

The OS/8088 software package uses the standard MICROKIT Editor, Assembler and Monitor programs with a series of enhancements as will be described. These fall into four categories: 1) System commands, 2) Editor commands, 3) Assembler commands, and 4) Monitor commands. These commands will be described in that order.
SYSTEM COMMANDS

Three System Commands have been added. These allow the user to switch between the three system programs: Editor, Assembler and Monitor. They are:

JA - Jump to the Assembler when in Editor or Monitor mode.
JM - Jump to the Monitor when in Assembler or Editor mode.
JE - Jump to Editor when in the Assembler or Monitor mode.

These commands allow the user to rapidly switch from one of the system software packages to the other. This means that when errors are detected in the Monitor program, they can be immediately corrected in the source program rather than using machine language patches.

NEW EDITOR COMMANDS

A small number of commands have been added to the Editor program. In most cases, they are designed to reduce the opportunity for operator error.

The commands are as follows:

CLR - Clear the Workspace. This avoids the possibility of inadvertently clearing the workspace by typing a "C".

REW - Rewind the Input Tape. This avoids the possibility of inadvertently rewinding the tape with an "R".

A RETURN during READ will interrupt the READ routine.

A RETURN during WRITE will generate the message RESTART WRITE?. This allows the operator to interrupt a WRITE command without losing the contents of the workspace. Thus, if the cassette has been inserted upside down, or the recorder is not in the write mode, correction is possible without losing the workspace contents. After the RESTART WRITE message appears, the programmer may type either a "Y", in which case the output tape will be rewound and the work area will not be cleared, or the programmer may type RETURN, in which case the write will be interrupted and may be restarted.

EC - Outputs to tape and writes and end-of-file, but does not delete the data from memory. Useful for checkpointing work in progress without deleting it so that it has to be read in again.
Three commands have been included that allow the programmer to write two tapes on the initial entry of the program. This provides the same sort of redundancy that the "old" and "new" tapes provide after the first set of tapes have been written. In this case, the programmer inserts cassettes into both recorders and puts both recorders into the Write mode. He then has available the following three commands:

- **WB** - Output to both tapes. A write command, just like any other write command.
- **WB#$** - Write the top half of the workspace to both tapes.
- **EB** - Output to both tapes and End-file.

Two more commands have been added that allow the user to scroll through the workspace. They are:

- **Control A** - Advance one line.
- **Control H Repeat** - Scroll up.
- **Control B** - Back-up one line.
- **Control B Repeat** - Scroll down.

### NEW ASSEMBLER COMMANDS

A series of commands have been added to the Assembler in OS/8080. These primarily are used to accomplish assemblies out of memory. The default input is from memory, the default object is to memory. Pass 1 and Pass 2 are both run in response to a RETURN after invoking the Assembler. The following input/output option parameters have been added to Pass 1 and Pass 2:

- **1** - 1 is a parameter used to run Pass 1 from tape. It is used in conjunction with the I command that indicates a tape input.
- **2** - 2 is a parameter, again required only for input from tape. It can be used with assemblies from memory also to avoid re-running Pass 1.
- **I** - I is a parameter required both in Pass 1 and Pass 2 when input is from tape rather than from memory.

The following parameters apply to Pass 2 only. NOTE: when assembling from memory, the 1 and 2 parameters are not necessary. Also, it is not necessary to have both the L and T commands in order to get a truncated listing on the CRT. The T command alone will produce a truncated listing. The parameters are as follows:

- **L** - List the program to the CRT; non-truncated.
- **T** - List the program to the CRT; truncated.
- **E** - List errors only.
- **O** - Produce an object tape.
- **N** - Inhibit the object load to memory.
P - Print the listing on an Okidata Printer (the default print option is now for the Okidata Printer).
Bn - Print to EIA Port 1. The n following is an optional parameter specifying Baud rate as described in the assembly manual. NOTE: The default Baud rate is 110 Baud.

One other modification of the Assembler involves a missing end statement. If the Assembler encounters an End-of-file without an End statement, it will print the message NEXT TAPE . . . "E" TO END. If the programmer then types "E" the Assembler will automatically insert an End statement. Otherwise, the programmer may insert another tape into the reader and continue the assembly.

NEW MONITOR COMMANDS

A number of new debugging commands have been added to the Monitor program. They are:

FD,D, . . . - Find command, used to search memory for hexadecimal strings.
F - Find command without parameter, uses the last parameters loaded.
B Sn - Set breakpoint 1 through 4, allows the programmer to set up to 4 breakpoints in the program. The instructions clobbered with the software breakpoint (FF) will be saved so that they may be restored.
B Dn - Display breakpoint command, positions the cursor at the specified breakpoint.
B R - resets the breakpoint at the breakpoint location just displayed.
B C - Clears all temporary storage of breakpoints. Allows the programmer to again set from 1 to 4 breakpoints.

Control A - Advance the memory display 8 locations.
Control B - Backup the memory display 8 locations. NOTE: The repeat key will scroll in combination with these commands.

The Store (S), Find (F) and Zap (Z) commands have been modified to allow either HEX or ASCII characters as arguments. Thus, the following commands are valid:

S 'A'
F 'B', 'C'
Z A='A'

The Display (D), Execute (E), and Write (W) commands have been modified to allow for expression evaluation of the address using the following operators: "+" for adding parameters, "-" for subtracting parameters, "*" to point to an indirect address, and
finally a new operator "#" indicating a symbolic label. This last operator allows the programmer to position the cursor at the location of a label stored in the Label Table. For example, if the label LOOP has the hexadecimal address 113, then the command D#LOOP will cause the cursor to be positioned at address 113. All operators will be scanned from left to right. There is no hierarchy in any of the operators.
MICROPRINT/65

65 LPM PRINTER FOR MICROKIT-8/16 MICROCOMPUTER DEVELOPMENT SYSTEM

The MICROPRINT/65 is a 65 lpm line printer for use with MICROKIT-8/16 Microcomputer Development System. The MICROPRINT/65 prints bidirectionally at 65 lines-per-minute on standard TTY roll paper. The printer utilizes a 5 x 7 dot matrix to produce highly legible characters. Standard TTY ribbons are also used, thus the user is free from the bother and cost of special paper or special ribbons.

The MICROPRINT/65 is a simple, proven and reliable line printer capable of heavy duty operation. It can print single and multipart forms. Options are available for form feed fan-fold paper, and upper and lower case printing.

The MICROPRINT/65 is delivered complete with interface and software to run with the MICROKIT-8/16 microcomputer system. Optional software includes a Word Processor package for use with the upper/lower case printer.
MICROKIT DISK OPERATING SYSTEM

The MICROKIT Disk Operating System -- MICRODISK -- consists of a dual floppy disk drive, interface and software package.

With MICRODISK the user is now able to edit and assemble large programs 160 times faster than is possible with TTY based systems. The MICRODISK combined with the unique screen based editor, interactive debugger, high speed assembler and In-Circuit Emulator make the MICROKIT-8/16 the most powerful microcomputer development system available.

Each diskette has 77 tracks. The first track is always reserved by the system for the directory of files. Thus, there are 76 tracks available for system and user files. Each track is 26 sectors of 128 bytes.

DISK OPERATING SYSTEM FEATURES

The MICROKIT Disk Operating System incorporates a number of features which make it unique among microcomputer systems:

- Overlapped I/O and high-speed assembler doubles assembly throughput.
- File management system provides up to 76 files per diskette.
- Dual drive system provides 500 kilobytes of on line storage.
- Minimum resident Monitor reduces main memory requirements.
- Disk management functions allow user to create "load and go" system files for production programs.
- High speed CRT display and interactive debugger automatically generate full screen "snapshot" register and memory dumps faster than the blink-of-an-eye.
- Interactive screen based Editor eliminates blind editing. User receives instantaneous feedback on the results of each editing command.
- Utility package is provided for translation of media to and from: telecommunication ports, paper tape, cassette tape, EPROM's, and diskette. Utility package is provided in source form allowing users to include all or part in their programs.
- All Monitor functions -- creating, accessing for read and
write, deleting, copying and renaming of disk files -- are accessible by user programs. These powerful file management functions provide the complete capabilities of disk operating system for any application the user may envision.

Bootstrap loader in EPROM eliminates clumsy manual operations.

Disk interface board also contains a full interface for high-speed paper tape reader.

MICRODISK SOFTWARE

The software package is organized into six modules as follows:

MONITOR Provides the basic disk management functions that are used to: create, delete, copy and rename files.

EDITOR Creates and updates source (text) files stored on diskette.

ASSEMBLER Converts editor created source files to machine executable object files.

LOADER Loads and dumps object files.

DEBUGGER Generates the monitor "snapshot" dump display and provides interactive program debugging features.

UTILITY The utility module allows the user to transfer files from media to media. Includes routines for cassette tape, diskette, EPROM, telecommunication lines, and paper tape.

OPERATING PROCEDURES

The MICRODISK system allows the user to invoke any of the system modules from the present module he is using with the following commands:

JM Jump to Monitor
JE Jump to Editor
JA Jump to Assembler
JL Jump to Loader
JD Jump to Debugger
JU Jump to Utility

The user may create up to 38 other system files of this type.
MONITOR

The disk monitor is loaded from diskette into the top end of memory by the EPROM bootstrap loader. The MICRODISK Monitor is used for file management functions. The Monitor commands are summarized below.

C filename, init, ext
Creates a file with name "filename", size "init", and extension size "ext".

S filename
Delete (scratch) "filename".

A filename, attributes
Change the attributes of "filename". attributes are:

O - object file
S - source file
W - write protected file
P - permanent file
Z - system file

D (unit)
Display the directory for "unit" which may be 0 or 1. Unit may be omitted with default to unit 0.

I
Initialize and clear all files on unit 1.

M fromfile, tofile
Copies (move) "fromfile" to "tofile" the destination file "tofile" may have the prefix "N:" to indicate the file is to be created.

X fromunit, tounit, parm
Copies all files from diskette in "fromunit" to diskette in "tounit". "Parm" indicates which files are to be copied as follows:

No "Parm" copy all user files
A "Parm" copy all files
S "Parm" copy all system files

R oldname, newname
Rename the file "oldname" to "newname"

EDITOR

The MICRODISK Editor has all of the features of the tape and 0/S
8080 Editors described on pages 3-3 and 3-4 of "MICROKIT MICROCOMPUTER DEVELOPMENT SYSTEMS". In addition the MICRODISK Editor has the following file management commands:

- **W filename** Write all data from the editor work area to "filename".
- **W** Write all data from the editor work area to the previously opened write file.
- **W*** Write all data from the editor work area to the previously opened write file. Do not clear the work area.

Note that combinations of the above parameters are allowed.

- **L filename** Load the editor work area from "filename".
- **L** Load additional data from previously specified read file.
- **N** Write work area to previously specified write file and load additional data from previously specified read file.

**ASSEMBLER**

The MICRODISK Assembler is quite similar to the tape and O/S 8080 Assemblers described on pages 3-5 and 3-6 of "MICROKIT MICROCOMPUTER DEVELOPMENT SYSTEMS". The MICRODISK Assembler will request (upon assembly process request) the name of the Input File (source file) and the name of the Output File (object file). The MICRODISK Assembler will allow the user to create a new file for the object code without return to the Monitor.

**LOADER**

The MICRODISK Loader is used to load and dump object files. The loader has three commands:

- **L filename** Loads data from "filename" into memory.
- **E filename** Loads data from "filename" into memory and begins execution at entry point.
- **W filename, addr1,addr2,...** Writes data from memory at specified addresses into "filename". File may be created without return to the Monitor.
DEBUGGER

The MICRODISK Debugger has the same functions as the Tape System Monitor described on pages 3-1 and 3-3 of "MICROKIT MICROCOMPUTER DEVELOPMENT SYSTEMS"; with the exception of the tape commands. These functions have been implemented in the Loader.

UTILITY

The MICRODISK Utility contains a number of functions which provide for the transfer of data between the various media (diskette, cassette tape, paper-tape, telecommunication ports and EPROM programmer) supported by the MICROKIT system.

The only function required is a copy function and the operator need only specify input device, source or object data, output device and "filename" if the disk is to be used.
8080 CPU Module Specifications

Processor: 8080

Instruction Set: 78 including conditional branching, binary and BCD arithmetic, logical operations, subroutine stack, 7 registers

Memory Addressing: 16 bits, up to 64K bytes

I/O Addressing: 8 bits, up to 256 input and 256 output addresses

System Clock: Crystal Controlled, 2.0 MHz +/- 0.1%

Interrupts: 8 level vectored priority interrupts

Power-on Restart: Automatic bootstrap load

Connector: Dual 50-pin on 0.125 centers, Wirewrap P/N CPH8100-100 (SAE)

Board Dimensions: 6.00" x 8.00" x 0.062"

Operating Temp.: 10 C to 40 C

DC Power Requirements: +5 +/- 5% @ 0.75A
+12 +/- 5% @ 0.2A
-12 +/- 5% @ 0.05A
8080 CPU BLOCK DIAGRAM (M8-80)

DMA REQUEST
DMA ACKNL
READY
DELAY 1
DELAY 2
RESET

F/F
F/F
F/F
F/F

8080 CPU

CRYSTAL CLOCK

HIGH ORDER DECODE

ADDR BUFFER

DATA BUFFER

CONTROL LOGIC

STATUS LATCH

PRIORITY ENCODER

INTERRUPT REQUESTS 0-7

READ STROBE
WRITE STROBE
MEMORY CYCLE
I/O CYCLE
READ/WRITE

POWER ON

F/F

HOLD

HLDA

F/F

RDY

RES

INT

WR

DBIN

MEMR
WR
INP
OUT
HLT
INTA

8

16

8

8

3

8
**RAM Memory Module Specifications**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory Size</td>
<td>8K bytes</td>
</tr>
<tr>
<td>Word Size</td>
<td>8 or 16 bits (16 bits requires two modules)</td>
</tr>
<tr>
<td>Cycle</td>
<td>460 ns read; 970 ns write using 4K dynamic RAM chips</td>
</tr>
<tr>
<td>Refresh</td>
<td>64 us every 512 us accomplished by CRT access</td>
</tr>
<tr>
<td>Connector</td>
<td>Dual 50-pin on 0.125 centers Wirewrap P/N CPH8100-100 (SAE)</td>
</tr>
<tr>
<td>Board Dimensions</td>
<td>6.00&quot; x 8.00&quot; x 0.062&quot;</td>
</tr>
<tr>
<td>Operating Temp.</td>
<td>10 C to 40 C</td>
</tr>
<tr>
<td>DC Power Requirements</td>
<td>+5 +/- 5% @ 0.5A</td>
</tr>
<tr>
<td></td>
<td>+12 +/- 5% @ 0.25A</td>
</tr>
<tr>
<td></td>
<td>-12 +/- 5% @ 0.05A</td>
</tr>
<tr>
<td>Specification</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>------------------------------------------------------------------</td>
</tr>
<tr>
<td>CRT Display Size</td>
<td>24 lines of 40 characters</td>
</tr>
<tr>
<td>Character Set</td>
<td>Standard 64 character ASCII set</td>
</tr>
<tr>
<td>Character Generation</td>
<td>5 x 7 dot matrix</td>
</tr>
<tr>
<td>Display Memory</td>
<td>Accesses main memory using DMA</td>
</tr>
<tr>
<td>I/O Transfer Rate</td>
<td>20,000 cps</td>
</tr>
<tr>
<td>Refresh Rate</td>
<td>60 Hz</td>
</tr>
<tr>
<td>Video Output</td>
<td>Composite video without interlace</td>
</tr>
<tr>
<td>Interrupts</td>
<td>Maskable interrupts for keyboard strobe and break interrupt (BRK) key</td>
</tr>
<tr>
<td>Keyboard Interface</td>
<td>7 bit ASCII with character strobe</td>
</tr>
<tr>
<td>Mini-Console Interface</td>
<td>Switch closures for system &quot;reset&quot; and bootstrap &quot;load&quot;; TTL outputs for driving &quot;run&quot; and &quot;interrupt enabled&quot; LED's.</td>
</tr>
<tr>
<td>Connector</td>
<td>Dual 50-pin on 0.125 centers</td>
</tr>
<tr>
<td></td>
<td>Wirewrap P/N CPH8100-100 (SAE)</td>
</tr>
<tr>
<td></td>
<td>20-pin Scotchflex header</td>
</tr>
<tr>
<td></td>
<td>Connector P/N 3421 (3M)</td>
</tr>
</tbody>
</table>
CRT & KEYBOARD I/O BLOCK DIAGRAM (M8-20)

DATA

DMA REQUEST

LINE REG.

CHAR. GEN.

DOT SHIFT

VIDEO DRIVER

VIDEO

CRystal OSC.

CNTR. CHAIN

H. SYNC

V. SYNC

ADDR.

CNTR.

BANK SEL.

LATCH

I/O DECODE

10

16

ADDR.

F/F

KEYB. STROBE

KEYB. DATA

1

7

CONTROL

8-6
<table>
<thead>
<tr>
<th>Specification</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Board Dimensions</strong></td>
<td>6.00&quot; x 8.00&quot; x 0.062&quot;</td>
</tr>
<tr>
<td><strong>Operating Temp.</strong></td>
<td>10 C to 40 C</td>
</tr>
<tr>
<td><strong>DC Power Requirements</strong></td>
<td>+5 +/- 5% @ 1.0A</td>
</tr>
<tr>
<td></td>
<td>-12 +/- 5% @ 0.2A</td>
</tr>
</tbody>
</table>

**Microkit-8/16 I/O Connectors**

![Diagram of I/O Connectors]
TAPE & EIA I/O BLOCK DIAGRAM (M8-21)

Diagram showing the connection between TAPE 1, TAPE 2, EIA 1, EIA 2, CPU CLOCK, 8-BIT I/O, TAPE READ DECODER, TAPE WRITE ENCODER, TAPE MOTOR CONTROL, EIA INTERFACE, I/O DECODE, RTC COUNTER, BUFFER, LATCH, BOOT PROM 256 x 8, DATA, ADDR., and CONTROL.
<table>
<thead>
<tr>
<th>Specification</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cassette Type</td>
<td>Philips type audio cassettes</td>
</tr>
<tr>
<td>Tape Units</td>
<td>Panasonic RQ-309AS</td>
</tr>
<tr>
<td>No. of Heads</td>
<td>One</td>
</tr>
<tr>
<td>No. of Gaps</td>
<td>One</td>
</tr>
<tr>
<td>No. of Tracks</td>
<td>One</td>
</tr>
<tr>
<td>Recording Technique</td>
<td>Phase encoded (Harvard)</td>
</tr>
<tr>
<td>Bit Transfer Rate</td>
<td>2000 bps</td>
</tr>
<tr>
<td>Recording Density</td>
<td>1067 +/- 2% bpi</td>
</tr>
</tbody>
</table>
| Character Transfer Rate             | 225 cps max.  
|                                    | 100 cps average including overhead |
| Error Detection                     | 16-bit checksum |
| Error Correction                    | Redundant recorded data |
| Error Rate                          | Recoverable: 1 in 1,000,000 bits  
|                                    | Unrecoverable: 1 in 100,000,000 bits |
| Record Size                         | Variable (1 to 32K bytes) |
Inter Record Gap 1.875 inch (1 second)
BOT, EOT Recorded codes
Manual Rewind End-to-end in 60 seconds
Write Protect Knock out tab
No. of EIA lines One to terminal, one to modem
Modem Control Request to send, clear to send, carrier detect
Baud Rates 50 to 1200 software generated
Real Time Clock Crystal controlled, 32 us resolution; overflows at 2.048 ms
I/O Port 8 bit parallel, general purpose TTL, input and/or output for party line I/O or interprocessor communication.
PROM Memory 256 bytes for loader
Interrupts Clock overflow and I/O port request; all maskable
Board Dimensions 6.00" x 8.00" x 0.062"
Connector Dual 50-pin on 0.125 centers
Wirewrap P/N CPH8100-100 (SAE)
20-pin Scotchflex header
Connector P/N 3421 (3M)
Operating Temp. 10 C to 40 C
Power Requirements +5 +/- 5% @ 1.0A
+12 +/- 5% @ 0.2A
-12 +/- 5% @ 0.2A
Note on the preparation of this document:

MICROKIT WORD PROCESSOR

This report was prepared using the MICROKIT Word Processing system. The MICROKIT Word Processor allows the user to print neatly formatted reports, manuals, documents etc. It is used in conjunction with the MICROKIT Editor program. The text to be printed is first entered, along with format commands, using the Editor. The Word Processor program is then used to read the text from the Editor tape, format the data according to the imbedded commands, and print it.

The Word Processor recognizes a series of format commands. These format commands are of the form ↑AA (where AA represents two alphabetic characters). The Word Processor also recognizes /A (where A is any alphabetic character) as a case shift. Thus if the user is in the normal lower case mode then "/The" will print "The."

The basic unit of text is the paragraph. New paragraphs are signalled by ↑PP at the beginning. Each time the Word Processor encounters ↑PP it will skip two lines and begin a new paragraph. Text within a paragraph is entirely free form. The Word Processor will fit the maximum number of words on each line that it can without violating the line width limits. Thus all extra spaces will be suppressed on print out. Other commands such as ↑AI (As-Is Mode) are used to suppress this paragraph mode of operation.

The Word Processor has software interfaces for the MICROPRINT/65-LC line printer and the BCD version of the IBM 2741 "Selecetric" terminal.

The format commands recognized by the Word Processor are:

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>↑NL</td>
<td>Execute a carriage-return and line feed to position the print head at the beginning of a new line. Overides paragraph rules.</td>
</tr>
<tr>
<td>↑PG</td>
<td>Execute a form feed (line printer) or request a new page (2741 terminal).</td>
</tr>
<tr>
<td>↑UC</td>
<td>Set upper case mode (shift-lock).</td>
</tr>
</tbody>
</table>
MICROKIT INC. MAY 1976

↑LC
Set lower case mode (shift-unlock).

↑PP
Skip two lines. Position print head at the first column and start a new paragraph.

↑BS
Backspace on the 2741 terminal only. Used for underlining or overstriking.

↑SKnn
Skip nn lines.

↑IFnn
Conditional new page. Avoids printing of headings as the last lines on a page. Skips if line is within the last nn lines.

↑AI
Enter the As-Is mode. This will defeat the extra space suppression feature of the normal paragraph mode of operation.

↑EA
End the As-Is mode.

↑TI
Flags the following text as a title to be printed on each page. (Note: title must be followed by a ↑PG command to ensure printing title on the first page).

↑ET
Ends the title.

↑NT
Suppresses the title on all subsequent pages.

↑IM
Indents all subsequent text. The indent will be 5 spaces unless reset by the ↑IS command.

↑EI
Ends the indent mode.

↑ISnn
Set the indent to value nn.
Set the line width to nn. The default line width is 66 characters.

Set the number of printed lines per page to nn. The default number of printed lines is 53.

Start a label at the left margin during indent mode.

Flags the end of a label.

Position the print head to column nn. Overrides the paragraph mode.

Reset to default Page Width, Page Length and Indent.

EXAMPLE OF TEXT AS SEEN ON THE CRT SCREEN

\texttt{\textasciitilde PP/this is an example of the use of the \textasciitilde UCmicrokit\textasciitilde LC /word /processor /program. It uses many of the commands in the repertoire. For example: \textasciitilde NLT\textasciitilde IM\textasciitilde IS20 \textasciitilde PP/IL/label/EL /labeling an indented paragraph. This uses the /i/1 and /e/1 commands. \textasciitilde PP/IL/tab/command/EL \textasciitilde TB40 /moves the print head \textasciitilde NLT\textasciitilde TB40 to position 40. \textasciitilde EI \textasciitilde PP/these are a few examples of the use of the \textasciitilde UCmicrokit\textasciitilde LC /word /processor /program. With a little practice its use}
should become second nature.

EXAMPLE OF THE WORD PROCESSOR OUTPUT AS PRINTED OUT IN HARD COPY

This is an example of the use of the MICROKIT Word Processor Program. It uses many of the commands in the repertoire. For example:

Label                      Labeling an indented paragraph. This uses the IL and EL commands.

Tab Command                Moves the print head to position 40.

These are a few examples of the use of the MICROKIT Word Processor Program. With a little practice its use should become second nature.
The Most POWERFUL 8080 or 6800 Microcomputer Development System at any price . . . $3,850.00 COMPLETE.

Only a truly powerful microcomputer development system allows you to accomplish your design task quickly, efficiently, and at a minimum cost. MICROKIT has the powerful microcomputer development system you need.

OTHER SYSTEMS OBSOLETE
The MICROKIT-8/16 has a unique CRT display refreshed directly from the microcomputer memory so it can write the full 960 character screen at 20,000 cps, faster than the blink-of-an-eye. Using the fast CRT, our interactive debugger gives you full screen hexadecimal memory dump displays instantaneously. With this kind of fast system response, debugging is a snap.

To match our fast debugger we have a screen based editor that is quickly learned and easy to use, because it lets you see the changes you make instantly and in full context on the CRT display. Rounding out our outstanding software package is a complete microcomputer resident assembler. Any system without a display as fast as ours and without software like ours cannot possibly match the powerful debugging and editing features we offer.

EVERYTHING FOR $3,850
But how is it that the MICROKIT-8/16 has everything (keyboard, display, tape units, and software) included in the basic price? We keep your cost low by using a standard television set for the high-speed display, and by using audio cassette units for mass storage. And with MICROKIT's proprietary recording technique, you get data reliability comparable to digital cassette units while data is transferred to the cassettes at the rate of 2000 bps — 20 times faster than TTY paper tape.

BOTH 8080 AND 6800
The MICROKIT-8/16 can be ordered as either an 8080 or a 6800 based system. At $3,850 either system is the best buy in microcomputer development systems today because they both include the 8K memory, the display and keyboard, the two audio cassette units, and the full complement of development software — debugger/monitor, editor, and assembler. Furthermore, either system can easily be switched to the other processor with our conversion packages that consist of a plug-in processor module and software.

FULL LINE OF ENHANCEMENTS
Rest assured that we also back you up with a complete line of enhancements including In-Circuit Emulators, EPROM programmers, add-on RAM or EPROM memories, line printers, "semiconductor disk" memory resident operating systems, and single or dual drive floppy disks with a super disk operating system.

MICROKIT-8/16'S ARRIVE READY TO GO TO WORK FOR YOU
But don't be misled by our name, our system comes fully assembled, fully tested, fully warranted, and ready to begin helping you with microcomputer development the very day it arrives. The MICROKIT-8/16 is a proven and reliable system which over the past year has received enthusiastic customer acceptance.

Our unbeatable features make the MICROKIT-8/16 an obvious choice for the designer requiring a microcomputer development system. Write or call MICROKIT INC. today, (213) 828-8539, to see just how well the MICROKIT-8/16 fits your microcomputer development requirements.

MICROKIT INCORPORATED
2180 COLORADO AVENUE  SANTA MONICA, CALIFORNIA 90404  (213) 828-8539
# MICROKIT

**Northern California Representative**

**NOUSOU - Incorporated**

**BOX 667 • DIABLO • CA 94528**

(415) 820-2588

---

**PRICE LIST**

Effective September 1, 1976

### PRODUCT DESCRIPTION

<table>
<thead>
<tr>
<th>PRODUCT</th>
<th>DESCRIPTION</th>
<th>PRICE</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MICROCOMPUTER SYSTEMS</strong></td>
<td>Microcomputer Development System for the 8080 microprocessor. Includes CRT display, keyboard, (2) cassette tape units, software and manuals.</td>
<td>$3,850.</td>
</tr>
<tr>
<td>MICROKIT-8/16 MOD 8080</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MICROKIT-8/16 MOD 6800</td>
<td>Microcomputer Development System for the 6800 microprocessor. Includes CRT display, keyboard, (2) cassette tape units, software and manuals.</td>
<td>$3,850.</td>
</tr>
</tbody>
</table>

### IN-CIRCUIT EMULATORS/PROM PROGRAMMERS

<table>
<thead>
<tr>
<th>PRODUCT</th>
<th>DESCRIPTION</th>
<th>PRICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MICROEMULATOR/8080</td>
<td>In-circuit emulator module for 8080, and PROM programmer module for 2708/2704 EPROM. Debugging features include hardware breakpoint, single step and trace.</td>
<td>$1,250.</td>
</tr>
<tr>
<td>MICROEMULATOR/6800</td>
<td>In-circuit emulator module for 6800, and PROM programmer module for 2708/2704 EPROM. Debugging features include hardware breakpoint, single step and trace.</td>
<td>$1,250.*</td>
</tr>
</tbody>
</table>

### CONVERSION PACKAGES

<table>
<thead>
<tr>
<th>PRODUCT</th>
<th>DESCRIPTION</th>
<th>PRICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>M8-68C</td>
<td>Conversion module and software for adding 6800 to 8080 system.</td>
<td>$950.*</td>
</tr>
<tr>
<td>M8-80C</td>
<td>Conversion module and software for adding 8080 to 6800 system.</td>
<td>$950.</td>
</tr>
</tbody>
</table>

### PERIPHERALS

<table>
<thead>
<tr>
<th>PRODUCT</th>
<th>DESCRIPTION</th>
<th>PRICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MICROPRINT/65-LC</td>
<td>65 lpm printer with upper and lower case.</td>
<td>$2,300.</td>
</tr>
<tr>
<td>MICROPRINT/65-TF</td>
<td>65 lpm line printer with upper/lower case and tractor feed.</td>
<td>$2,600.</td>
</tr>
<tr>
<td>MICRODISK/2</td>
<td>Dual drive floppy disk unit including interface module and DOS software.</td>
<td>$3,650.</td>
</tr>
</tbody>
</table>

### MODULES

<table>
<thead>
<tr>
<th>PRODUCT</th>
<th>DESCRIPTION</th>
<th>PRICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>M8-10</td>
<td>8K dynamic RAM module with memory write protect.</td>
<td>$800.</td>
</tr>
<tr>
<td>M8-11</td>
<td>8K PROM/2K static RAM module. Includes sockets for (8) 2708 EPROM's and (16) 2102 RAM's.</td>
<td>$150.</td>
</tr>
<tr>
<td>M8-30</td>
<td>Universal prototype board.</td>
<td>$40.</td>
</tr>
<tr>
<td>M8-31</td>
<td>Module extender.</td>
<td>$40.</td>
</tr>
<tr>
<td>M8-41</td>
<td>PROM programmer for 2708/2704 EPROM. Includes hardware breakpoint, single step and trace debugging features.</td>
<td>$750.</td>
</tr>
</tbody>
</table>

### SOFTWARE/HARDWARE PACKAGES

<table>
<thead>
<tr>
<th>PRODUCT</th>
<th>DESCRIPTION</th>
<th>PRICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS/8080</td>
<td>Software package with memory resident monitor, editor and assembler. Includes additional 24K RAM memory.</td>
<td>$1,850.</td>
</tr>
<tr>
<td>OS/6800</td>
<td>Software package with memory resident monitor, editor and assembler. Includes additional 24K RAM memory.</td>
<td>$1,850.*</td>
</tr>
<tr>
<td>BASIC/8080</td>
<td>BASIC interpreter. Includes additional 8K RAM memory. Program used for preparation of documents, reports and manuals.</td>
<td>$900.*</td>
</tr>
<tr>
<td>Word Processor/8080</td>
<td>Makes MICROI KIT-8/16 usable as an intelligent terminal and provides a loader for PL/M object code.</td>
<td>$100.</td>
</tr>
<tr>
<td>TERMINAL SIMULATOR &amp; PL/M LOADER</td>
<td></td>
<td>$15.</td>
</tr>
</tbody>
</table>

### TERMS:


### FOB:

Santa Monica, California 90404, U.S.A.

### WARRANTY:

All equipment is fully assembled, tested and warranted for ninety (90) days against defects in material and workmanship.

*Available 4th quarter 1976