Build the Circuit Cellar MPX-16 Computer System
Part 2

A continued description of an 8088-based system that shares its principles of operation with the IBM Personal Computer.

Steve Ciarcia
POB 582
Glastonbury, CT 06033

This article is the second of three describing the design and operation of my most ambitious construction project to date: the Circuit Cellar MPX-16 computer system. I've written these articles with the intent of giving you a grasp of the basic functional parts of a complicated piece of electronic equipment and how these parts work together.

Because the MPX-16 is somewhat more complex than the projects I normally write about, I've had to simplify the presentation of many details to fit them into the magazine, but if you're interested in building an MPX-16, you can get all the details you need from the *MPX-16 Technical Reference and User's Manual*, which comes with the printed-circuit board available from The Micromint (see the text box on page 78). This book includes timing diagrams and listings of the MPX-16's special software.

Last month I presented an overview of the system and a discussion of the coprocessors and bus structures. This month, I'd like to continue by explaining memory, interrupts, the expansion bus, and I/O (input/output) decoding. But first, here's a recap of the MPX-16's features.

System Features

The Circuit Cellar MPX-16 computer system fundamentally consists of a single 9- by 12-inch five-layer printed-circuit board (containing 120 integrated-circuit packages), to which various peripheral devices are attached. It has nine expansion slots and is completely compatible with the I/O-expansion bus of the IBM Personal Computer.

The MPX-16 uses the Intel 8088 microprocessor and the optional 8087 numeric coprocessor; the main circuit board has room for 256K bytes of user memory and contains two serial and three parallel I/O ports, a floppy-disk controller, and EPROMs (erasable programmable read-only memories) containing the BIOS (basic I/O system) module of Digital Research's CP/M-86 16-bit disk operating system. The MPX-16 can be readily expanded to provide a full 1 megabyte of user memory and several megabytes of hard-disk mass storage. A more detailed list of characteristics appears in table 1.

The MPX-16 was initially designed to run CP/M-86, but eventually Microsoft's MS-DOS operating system will be available for it, making it possible to run most software written for the IBM Personal Computer on the MPX-16, except software that uses unique features of the IBM PC. The principal difference is this: with the present BIOS, the MPX-16 communicates with the user through a serially interfaced display terminal instead of a memory-mapped video display. (You could theoretically install an IBM Color Graphics Display Adapter and a serial IBM-type keyboard for exact hardware emulation.)

The MPX-16 is well suited for use as a low-cost 8088-based computer for integration into a complete hardware/software package, chiefly because it combines so many functions on a single printed-circuit board. Putting together the hardware of a complete system, you need only add a power supply, a serial video-display or printing terminal, and one floppy-disk drive (either 5¼- or

IBM and IBM Personal Computer are trademarks of International Business Machines Corporation.
CP/M-86 is a trademark of Digital Research Inc.

1. designed to use a 5-MHz Intel 8088 microprocessor, which combines a 16-bit architecture with an 8-bit bus interface and has 20-bit addressing capability for up to 1 megabyte of system memory, operating in maximum mode to support multiprocessing
2. optional Intel 8087 math coprocessor
3. onboard space for four 64K-byte banks of dynamic RAM for a total of up to 256K bytes, with parity generation and error detection
4. sockets for up to 64K bytes of JEDEC 24- or 28-pin standard ROM or EPROM devices
5. two RS-232C serial I/O ports
6. two 8-bit general-purpose parallel I/O ports with handshaking control lines
7. one Centronics-compatible parallel printer port
8. four programmable timers (one for a real-time clock, two for data rates, one for memory-refresh requests)
9. four independent DMA (direct memory access) channels
10. sixteen levels of vectored, prioritized interrupt control
11. single- or double-density floppy-disk controller for controlling up to four 5¼-inch or 8-inch single- or double-sided drives
12. five 62-pin I/O-expansion-channel connectors (hardware compatible with the IBM Personal Computer) with space for four more
13. five-layer 9- by 12-inch printed-circuit board
14. BIOS for CP/M-86 in EPROM

Table 1: *Major characteristics of the MPX-16 computer system.*

Start Address			
Decimal	Hexadecimal	Bank	Function
0	00000	0	64K to 256K bytes of R/W memory on system board
64K	10000	1	
128K	20000	2	
192K	30000	3	
256K	40000	4	up to 704K bytes of expansion memory in I/O channel
320K	50000	5	
384K	60000	6	
448K	70000	7	
512K	80000	8	
576K	90000	9	
640K	A0000	10	
704K	B0000	11	
768K	C0000	12	
832K	D0000	13	
896K	E0000	14	
960K	F0000	R	64K bytes of system ROM/EPROM

Figure 1: *Map of memory-address-space allocation in the MPX-16, in 64K-byte increments.*

8-inch). Turn on the power, insert a CP/M-86 disk, and go. And by the time you read this, an enclosure for the circuit board should be available. Many applications need nothing more.

System Memory

The stars of the show in November's article were the Intel 8088 microprocessor and the 8087 numeric processor extension (NPX), with supporting roles played by the Intel 8284 clock generator/driver, the 8288 bus controller, and the 8237A-5 DMA (direct memory access) controller. This month we look at some less glamorous but equally necessary components, starting with a type of component so prosaic as to be called a commodity by the semiconductor industry: the memory.

The MPX-16 system circuit board contains both read-only and read/write memory. In addition to the possible 64K bytes of ROM, the MPX-16 circuit board contains sockets for up to 256K 9-bit words (an 8-bit byte plus a parity bit) of dynamic RAM (random-access read/write memory). Furthermore, to augment the onboard memory, as much as 704K bytes of expansion RAM or ROM can be added in the I/O-expansion slots using readily available memory-expansion boards such as the Quadram Quadboard or the Seattle Computer RAM-Plus card. A memory map of the 8088's 1-megabyte (1,048,576-byte) address space in 64K-byte increments is shown in figure 1. Two of the five sections of the schematic diagram are included in this article; section 2 appears as figure 2 on pages 48, 49, and 50; section 3 as figure 3 on pages 52, 53, and 54. A table of integrated circuits, giving their type, location, and power connections, appears on pages 56 and 60 as table 2.

ROM Configuration

Four integrated-circuit sockets, designated IC82 through IC85 in section 3 of the schematic diagram, are provided for holding ROM (read-only memory) chips, which most often are EPROM devices. These four JEDEC- (Joint Electron Device Engineering Council) standard 28-pin sockets can contain several sizes of EPROMs, any of the various "byte-wide" (8-bit word size) devices such as the 2716 (16K bits or 2K bytes), the 2732 (4K bytes), the 2764 (8K bytes) or the 27128 (16K bytes). EPROMs with 24-pin packages, such as the 2716s and 2732s, are plugged into the lower 24 pins of the sockets, with certain jumper connections set accordingly.

For proper operation, the MPX-16 circuit board must contain a ROM or EPROM device in the highest address space (socket IC85) and a bank of RAM in the lowest address space because the 8088 processor fetches its first instruction after a power-up reset from location hexadecimal FFFF0 (usually a jump instruction branching to an initialization routine) and uses interrupt vectors in the range hexadecimal 00000 to 003FF.

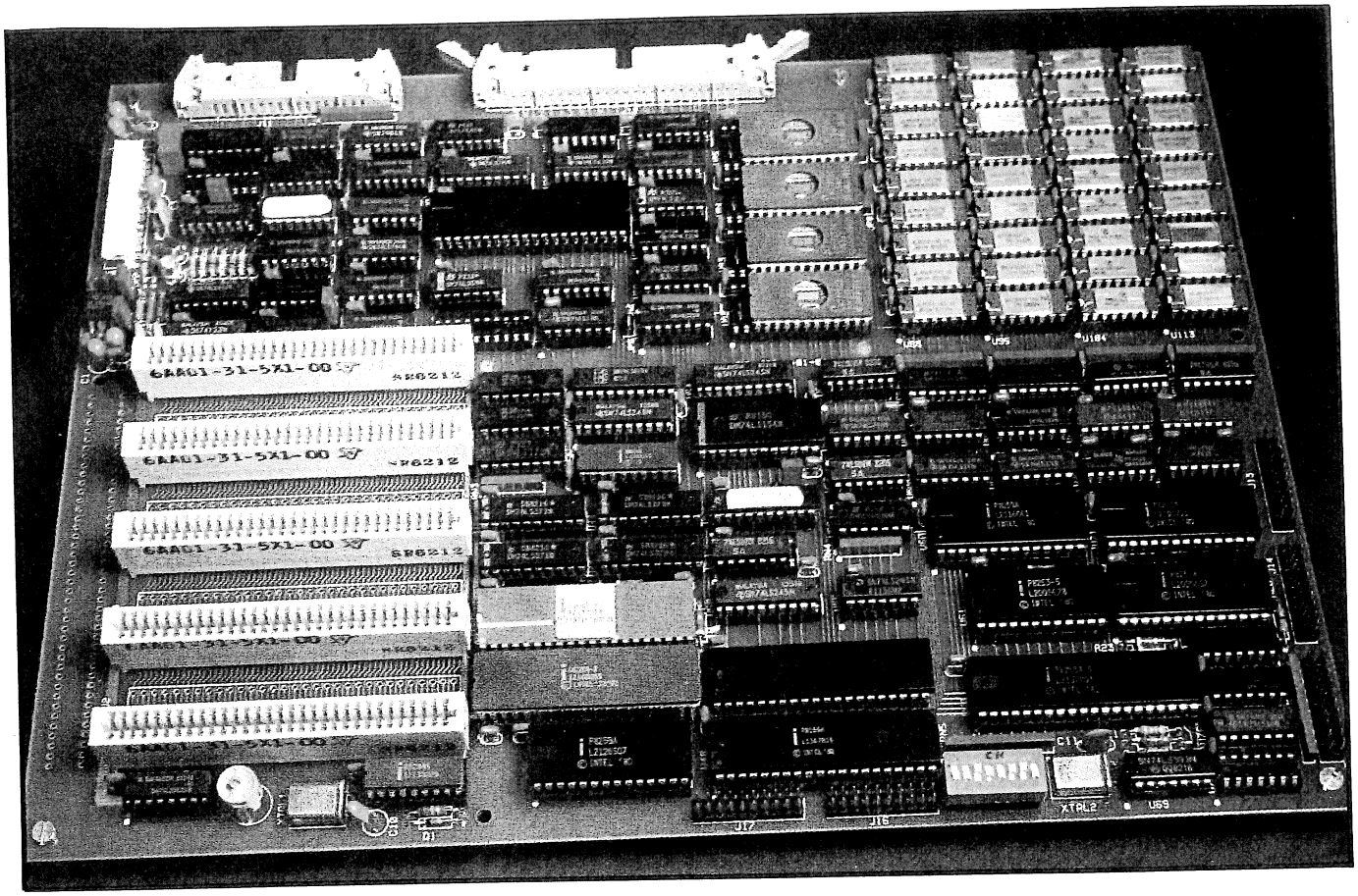


Photo 1: *The Circuit Cellar MPX-16 single-board computer system, which uses the latest technology to provide lots of low-cost computing power. The five-layer printed-circuit board contains 120 integrated circuits including most common peripheral-device interfaces; furthermore, any peripheral-device card intended for use with the IBM Personal Computer can be plugged into one of the I/O-expansion slots. There are nine slot positions, but only five sockets are installed initially.*

The capacity of the ROMs (or EPROMs) used on the system board must be compatible with the configuration of onboard jumpers JP1 through JP6 and with the program stored in the 32-word by 8-bit address-decoding PROM (programmable ROM) device IC45, an HM7603. The PROM program and jumper arrangements supplied with the system board are intended for type-2732 EPROMs. A different decoding PROM is needed for other memory-device types so that the four ROM sockets may be decoded into a contiguous address space in each case. (A PROM-programming table is included in the MPX-16 documentation.) The ROM-decoding logic and memory organization are respectively shown in sections 2 and 3 of the schematic diagrams.

The ROM-address-space-decoding logic for the system board is enabled whenever all three high-order system address bits, SYSA17 through SYSA19, are high, causing the output of a NAND gate (IC30) to go low. If five PROM-address bits SYSA11 through SYSA15 or SYSA12 through SYSA16 (depending on the jumper configuration) address one of the programmed locations, the selected ROM-chip-enable line (one of $\overline{\text{PROMSEL0}}$ through $\overline{\text{PROMSEL3}}$) is also driven low, selecting that memory device. The $\overline{\text{ROMSEL}}$ signal at IC28 pin 5 (a two-input OR gate in section 2) also enables a wait-state-generation circuit if jumper JP7 is connected. After one of the $\overline{\text{PROMSEL}}x$ lines has been driven active-low, a $\overline{\text{SYSMEMRD}}$ (system memory read, active-low) signal from the system bus master will initiate the memory-read cycle and generate a single wait state if JP7 is connected. Valid data from the ROMs is available on the data bus after $\overline{\text{SYSMEMRD}}$ goes low.

Memory chips are less glamorous than microprocessors, but just as necessary.

Normally, the MPX-16 requires ROM or EPROM devices with an access time of 350 ns (nanoseconds) or faster. The optional wait-state feature afforded through JP7 allows use of slower ROM devices with 450-ns access times. If faster devices are used, then JP7 should not be installed and the MPX-16 can operate with no wait states.

The EPROMs on the standard MPX-16 system board contain a power-on self-test routine and I/O drivers, including the CP/M-86 BIOS

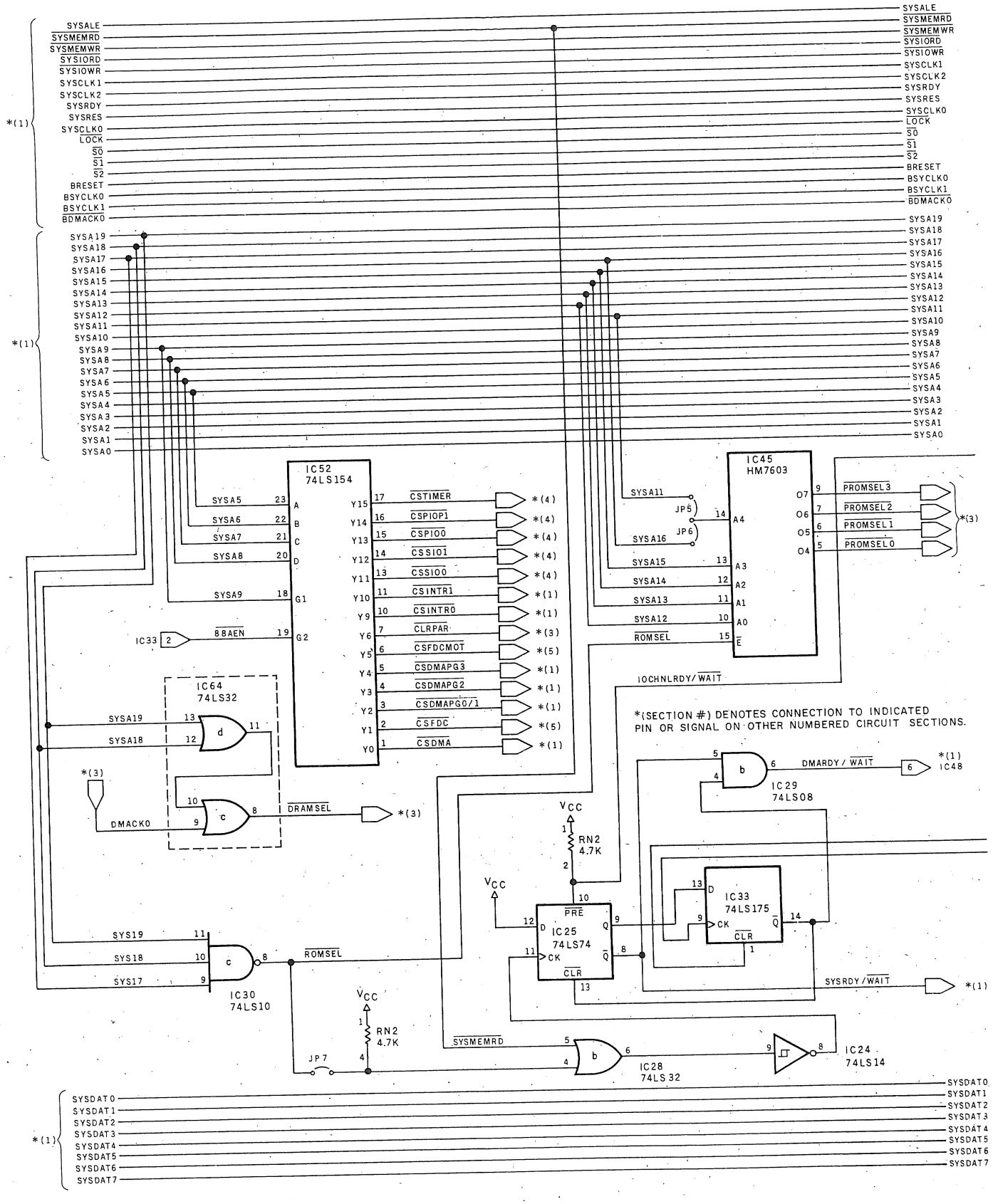
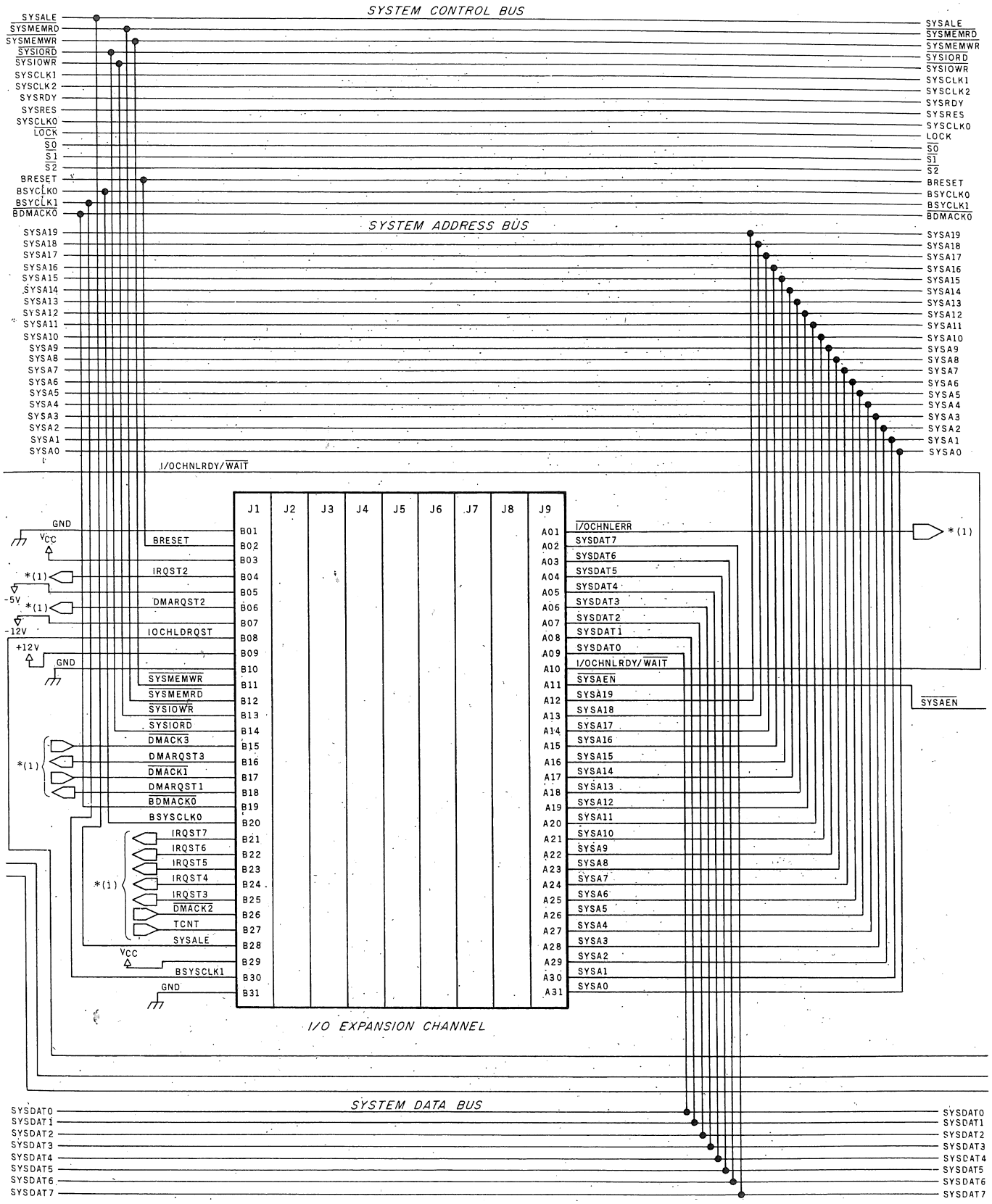


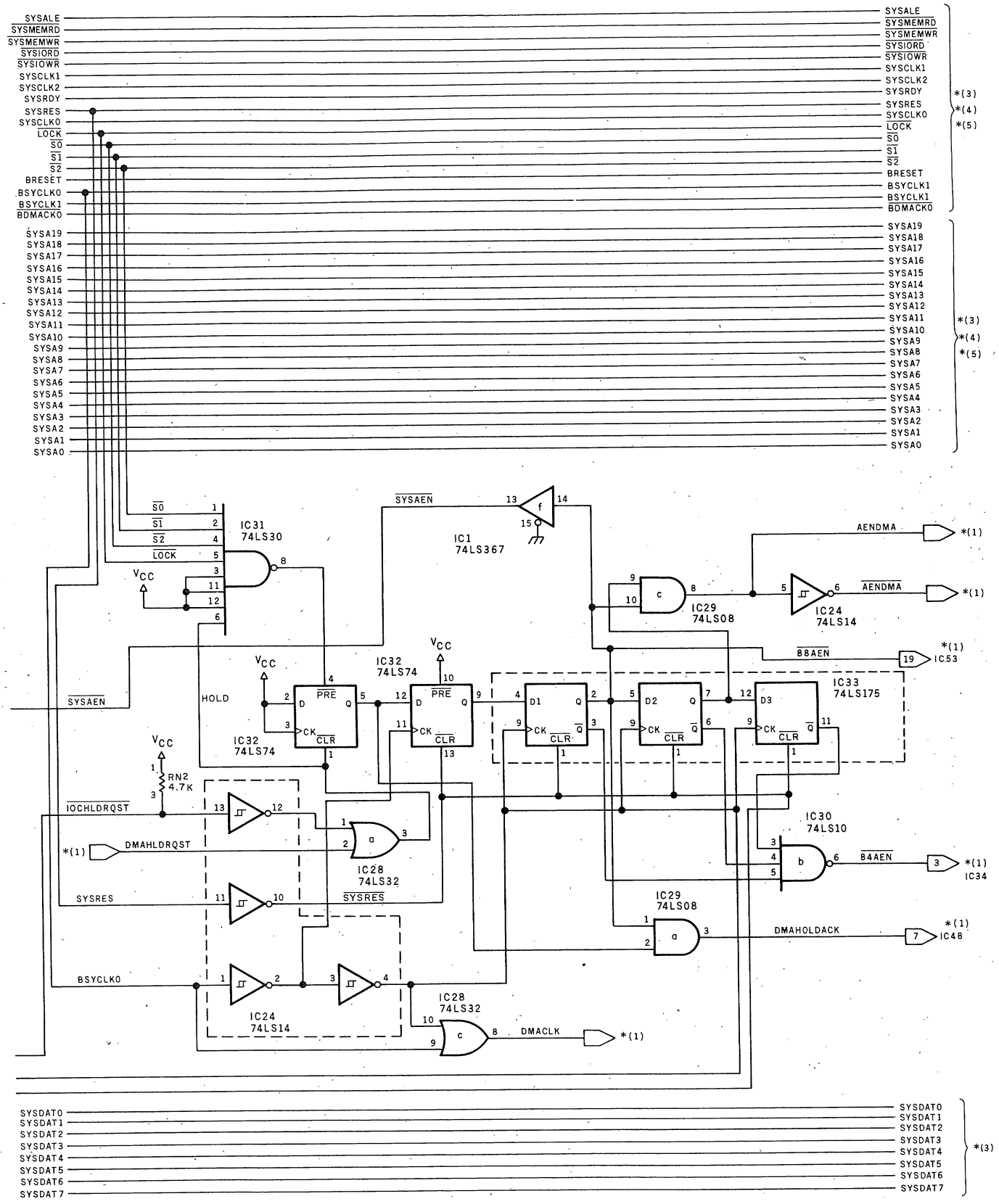
Figure 2: *Section 2 of the schematic diagram of the MPX-16 computer's main circuit board. (Section 1 was printed in last month's ar ticle; section 3 appears here in figure 3; sections 4 and 5 will appear in next month's article.) The notation *(n) indicates that a give signal line connects to a component or another line shown in schematic section n.*

A table of all the MPX-16's integrated circuits appears as table 2 on pages 56 and 60, giving each device's number, type, se



tion/figure location, and power connections. Connections to the I/O-expansion-channel slots are of course made to each individual slot. Possible substitutes for the HM7603 are the 74S288, the 82S123, and the AM27S09, although it's best to use the HM7603. (The diagram is continued on page 50.)

Figure 2: *Continued from page 49.*



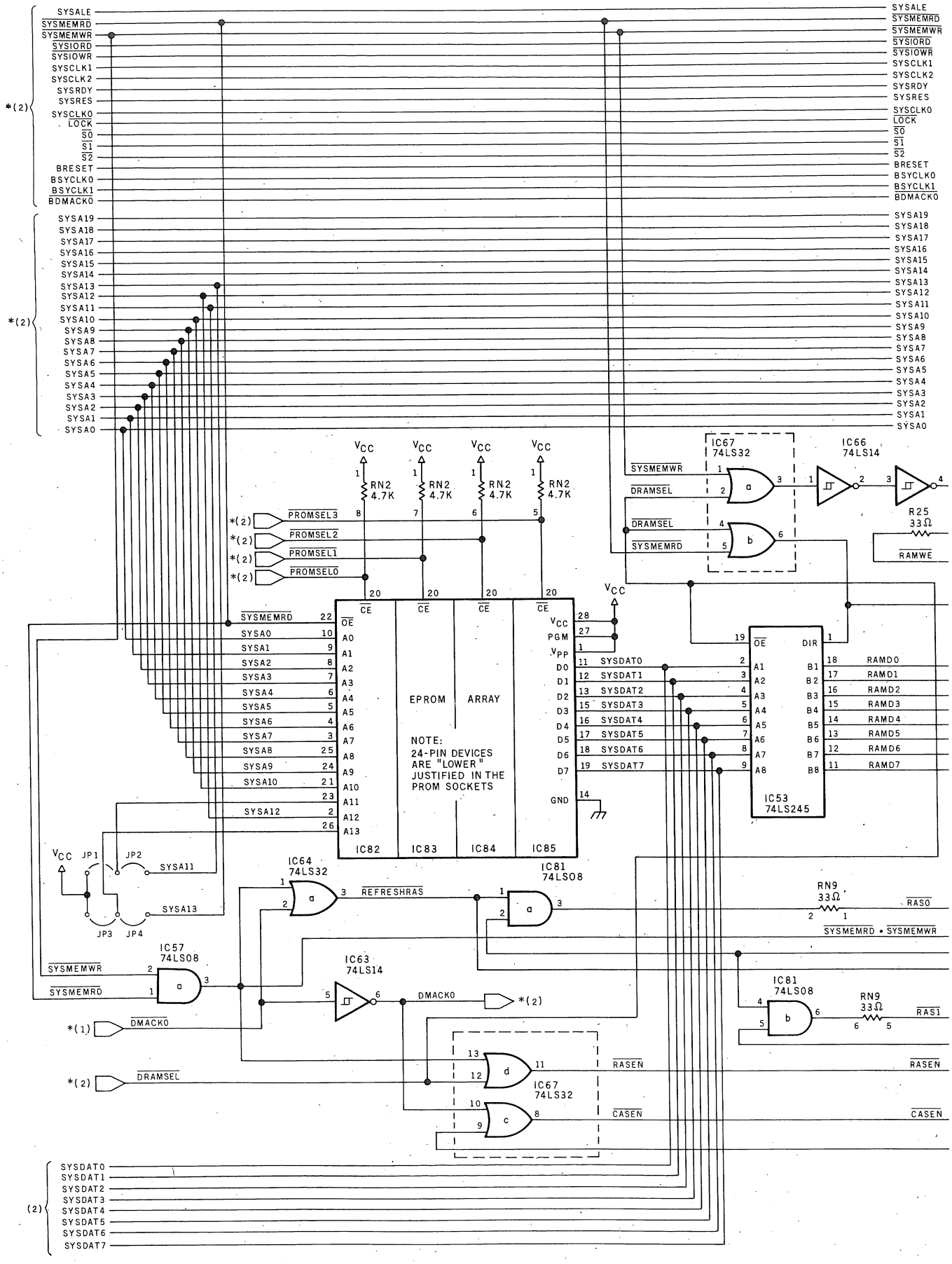
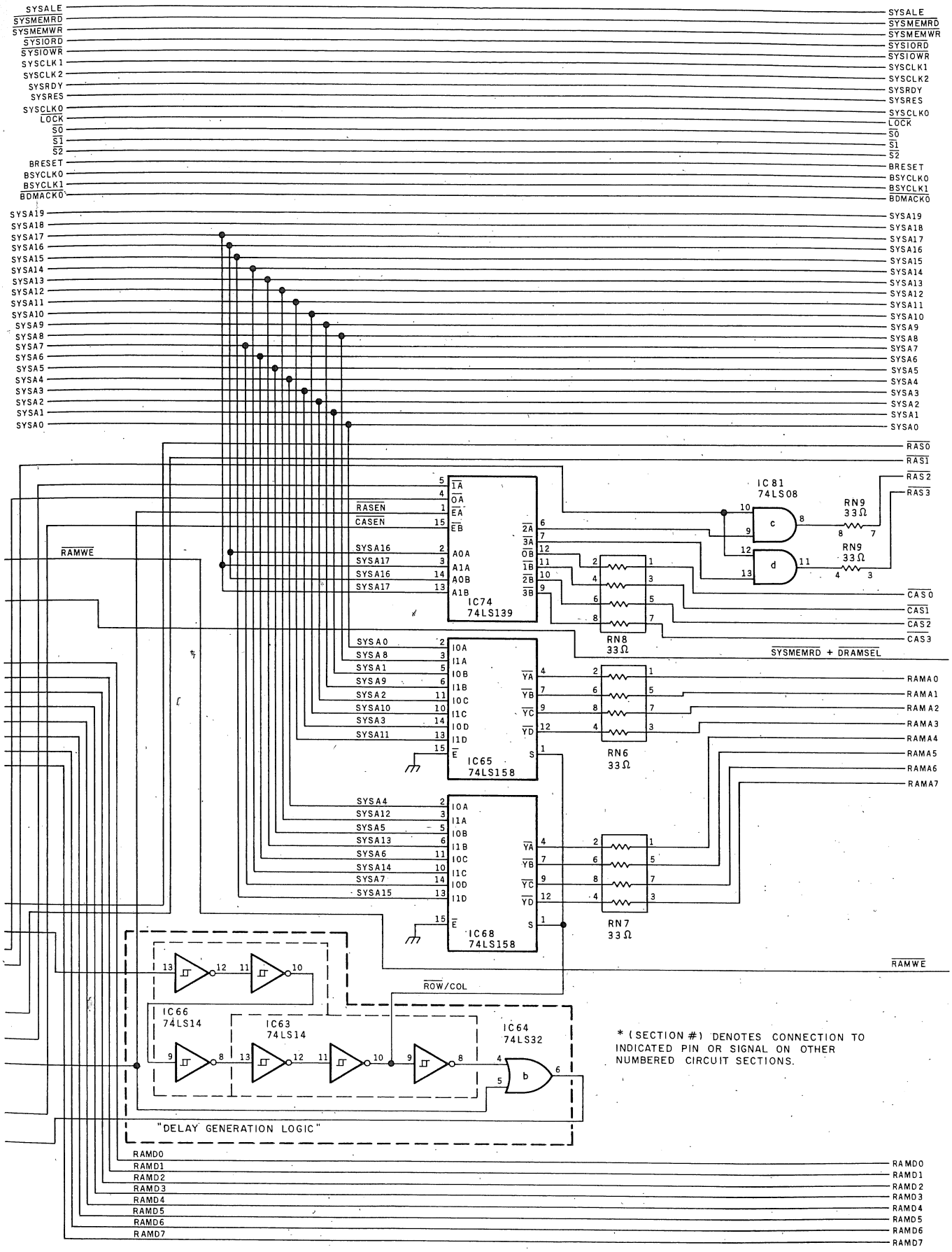


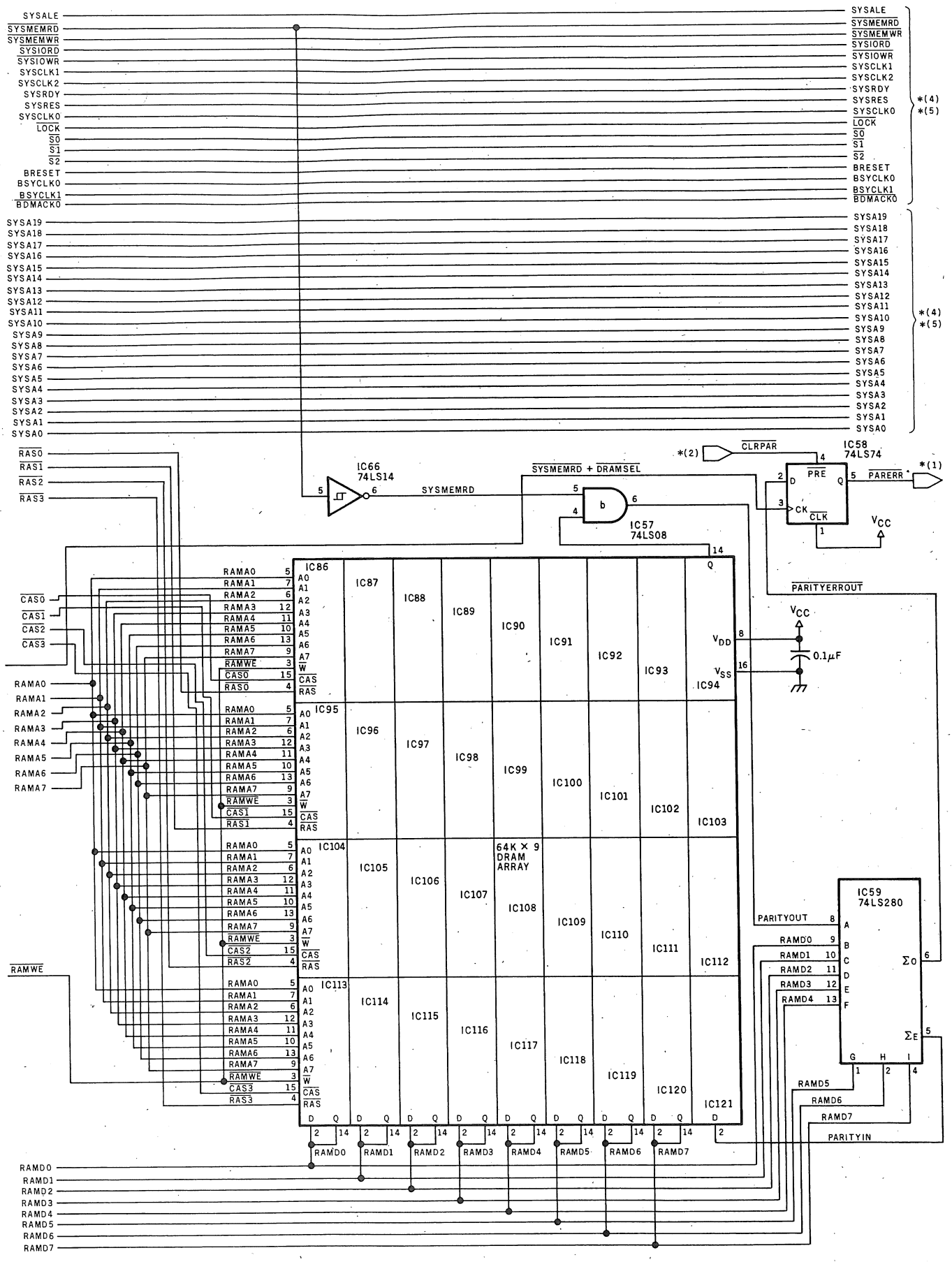
Figure 3: *Section 3 of the schematic diagram of the MPX-16 computer's main circuit board. The notation *(n) indicates that a give signal line connects to a component or another line shown in schematic section n.*

Connections shown on the edges of the dynamic-memory array on page 54 are of course made to each individual chip. Bypa



capacitors, not shown, should be installed adjacent to most integrated circuits between +5 V and ground. A table of all the MPX-16's integrated circuits appears as table 2 on pages 56 and 60, giving each device's number, type, section/figure location, and power connections. (The diagram is continued on page 54.)

Figure 3: *Continued from page 53.*



IC Number	Type	Schematic Section	+5 V	GND	+12 V	-12 V
VR1	LM7905	5 (3-2)	(voltage regulator)			3
IC1	74LS367	1 (1-3b), 2 (2-2)	16	8		
IC2	74LS123	5 (3-2)	16	8		
IC3	74LS157	5 (3-2)	16	8		
IC4	74LS124	5 (3-2)	16	8		
IC5	74LS175	5 (3-2)	16	8		
IC6	74LS173	5 (3-2)	16	8		
IC7	74LS393	5 (3-2)	14	7		
IC8	74LS10	5 (3-2)	14	7		
IC9	74LS74	5 (3-2)	14	7		
IC10	M1116-8M	5 (3-2)	14	7		
IC11	74LS153	5 (3-2)	16	8		
IC12	74LS14	5 (3-2)	14	7		
IC13	74LS74	5 (3-2)	14	7		
IC14	74LS74	5 (3-2)	14	7		
IC15	74LS74	5 (3-2)	14	7		
IC16	74LS74	5 (3-2)	14	7		
IC17	74LS175	5 (3-2)	16	8		
IC18	7406	5 (3-2)	14	7		
IC19	spare socket					
IC20	74LS04	1 (1-3ab), 5 (3-2)	14	7		
IC21	8272	5 (3-2)	40	20		
IC22	74LS240	5 (3-2)	20	10		
IC23	7407	5 (3-2)	14	7		
IC24	74LS14	2 (2-2)	14	7		
IC25	74LS74	2 (2-2)	14	7		
IC26	74LS139	5 (3-2)	16	8		
IC27	7407	5 (3-2)	14	7		
IC28	74LS32	2 (2-2), 4 (3-1)	14	7		
IC29	74LS08	2 (2-2)	14	7		
IC30	74LS10	1 (1-3a), 2 (2-2)	14	7		
IC31	74LS30	2 (2-2)	14	7		
IC32	74LS74	2 (2-2)	14	7		
IC33	74LS175	2 (2-2)	16	8		
IC34	8284A	1 (1-3a)	18	9		
IC35	8259A	1 (1-3a)	28	14		
IC36	8088	1 (1-3a)	40	1,20		
IC37	8087 (option)	1 (1-3a)	40	1,20		
IC38	74LS373	1 (1-3a)	20	10		
IC39	74LS373	1 (1-3b)	20	10		
IC40	74LS173	1 (1-3b)	16	8		
IC41	74LS173	1 (1-3b)	16	8		
IC42	74LS173	1 (1-3b)	16	8		
IC43	74LS245	1 (1-3a)	20	10		
IC44	74LS373	1 (1-3a)	20	10		
IC45	HM7603-5	2 (2-2)	16	8		
IC46	74LS245	1 (1-3a)	20	10		
IC47	8155H-2	4 (3-1)	40	20		

Table 2: *Integrated circuits in the MPX-16. Here are shown each device's number, type, section/figure location, and power connections.*

The location of each chip in the five-part schematic diagram is listed by schematic section; the characters in parentheses show in which article the section appeared and which figure the device appears in. Some integrated circuits containing multiple gates appear in more than one schematic section. *(The table is continued on page 60.)*

IC Number	Type	Schematic Section	+5 V	GND	+12 V	-12 V
IC48	8237A-5	1 (1-3b)	31	20		
IC49	74LS245	1 (1-3a)	20	10		
IC50	74LS373	1 (1-3a)	20	10		
IC51	8288	1 (1-3a)	20	10		
IC52	74LS154	2 (2-2)	24	12		
IC53	74LS245	3 (2-3)	20	10		
IC54	74LS243	1 (1-3b)	14	7		
IC55	74LS08	1 (1-3ab), 5 (3-2)	14	7		
IC56	74LS32	1 (1-3a)	14	7		
IC57	74LS08	3 (2-3)	14	7		
IC58	74LS74	3 (2-3), 4 (3-1)	14	7		
IC59	74LS280	3 (2-3)	14	7		
IC60	8255A-5	4 (3-1)	26	7		
IC61	8253-5	4 (3-1)	24	12		
IC62	8259A	1 (1-3a)	28	14		
IC63	74LS14	1 (1-3a), 3 (2-3)	14	7		
IC64	74LS32	2 (2-2), 3 (2-3)	14	7		
IC65	74LS158	3 (2-3)	16	8		
IC66	74LS14	3 (2-3)	14	7		
IC67	74LS32	3 (2-3)	14	7		
IC68	74LS158	3 (2-3)	16	8		
IC69	74LS393	4 (3-1)	14	7		
IC70	8251A	4 (3-1)	26	4		
IC71	8251A	4 (3-1)	26	4		
IC72	1489	4 (3-1)	14	7		
IC73	1489	4 (3-1)	14	7		
IC74	74LS139	3 (2-3)	16	8		
IC75	74LS00	1 (1-3b)	14	7		
IC76	74LS14	4 (3-1)	14	7		
IC77	7407	4 (3-1)	14	7		
IC78	7407	4 (3-1)	14	7		
IC79	1488	4 (3-1)		7	14	1
IC80	1488	4 (3-1)		7	14	1
IC81	74LS08	3 (2-3)	14	7		
IC82	EPROM	3 (2-3)	28,1*	14		
IC83	EPROM	3 (2-3)	28,1*	14		
IC84	EPROM	3 (2-3)	28,1*	14		
IC85	EPROM	3 (2-3)	28,1*	14		
IC86	4164	3 (2-3)	8	16		
↓	↓	↓	↓	↓		
IC121	4164	3 (2-3)	8	16		

* depends on type of EPROM used

Table 2: *Continued from page 56.*

and a floppy-disk bootstrap-loader routine.

RAM Configuration

The onboard user-programmable memory of the MPX-16 consists of one to four 64K-byte banks of nine type-4164 64K-bit dynamic RAM devices. Within the 8088 processor's 1-megabyte address space, the MPX-16 must have at least the lowest 64K-byte bank of RAM (bank 0) installed from hexadecimal addresses 00000 to 0FFFF so that interrupt-routine pointers can reside in the locations from hexadecimal 00000 to 003FF. The RAM chips are required to have an access time of no more than 200 ns and a cycle time of 335 ns. Single-bit parity generation and

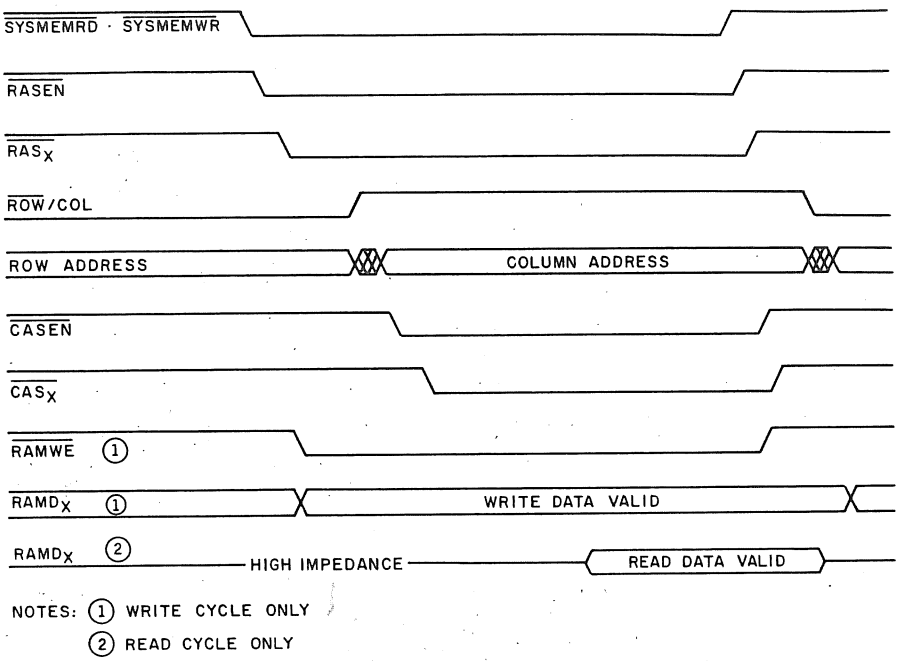


Figure 4: *Timing diagram for the memory operation of the MPX-16.*

error detection are provided for all of the 256K-byte onboard memory.

The RAM address-decoding logic is shown in section 2 of the schematic diagram (figure 2), and the read/write control logic, address multiplexers, RAM array organization, and parity-generation/error-detection logic are shown in section 3 (figure 3). The onboard RAM address space is selected when two conditions are met: the two high-order address bits SYSA18 and SYSA19 are both low and a memory-refresh cycle is not in progress (shown by DMACK0, the DMA-channel-0-acknowledge signal, being low). Because of this decoding scheme and the fact that the MPX-16 power-on self-test routine automatically clears memory and determines its size, the full 256K bytes of onboard RAM should be installed before you put in additional RAM in the I/O-expansion slots.

Dynamic Memory Refresh

Because dynamic RAM devices are used for the MPX-16's programmable memory, a memory-refresh circuit is necessary to prevent data stored in them from being lost. The 64K-bit dynamic RAMs require that all 256 rows be addressed every 4 ms (milliseconds) to maintain the integrity of the data (the columns need not be individually addressed); one row must be addressed for refreshing approximately every 15 μs (microseconds). To eliminate having a separate bus-arbitration circuit for this purpose, memory refresh is carried out by executing a DMA (direct memory access) read cycle in a "RAS-only" manner—that is, using only the row-address-strobe inputs of the memory chips. Because refresh is controlled by the DMA circuit, there can never be a conflict between the refresh operation and the processor's memory references.

The $\overline{\text{DMACK0}}$ signal goes active-low to indicate to the rest of the system that a refresh cycle is in progress. This signal disables the RAM-decoding circuitry, prevents the generation of a CAS (column-address strobe) signal, and enables the $\overline{\text{REFRESHRAS}}$ input at IC64 pin 2 (in section 3, figure 3). When the system bus master, the 8237A DMA controller (IC48 in section 1, printed last month), drives the $\overline{\text{SYSMEMRD}}$ or $\overline{\text{SYSMEMWR}}$ (system memory write) line low, the output at IC 64 pin 3 also goes low. This causes the outputs of the four two-input gates (sections of IC81: positive AND gates used as negative ORs), whose other input comes from IC74, to go low. These outputs form the RAS inputs for each of the four RAM banks. (The 33-ohm series resistors in the RAS control lines are there to reduce ringing on the lines, which might latch a new row address during the middle of the memory cycle.) The DMA controller is set up by the system-initialization software to automatically increment the address counter after each refresh-memory cycle.

Memory Operation

A diagram of typical timing cycles for normal memory-read and write operations is shown in figure 4. For either type of memory cycle, the read/write-control logic is enabled when the $\overline{\text{DRAMSEL}}$ signal is low, indicating that two conditions have both been fulfilled: a valid address (lower than hexadecimal C0000) has been latched on the system bus and the DMACK0 signal (from IC63, pin 6) is low (indicating that a refresh cycle is not in progress).

A memory cycle is initiated when the output of an AND gate (IC57 pin 3 in section 3) goes low, indicating that either the $\overline{\text{SYSMEMRD}}$ or the $\overline{\text{SYSMEMWR}}$ control signal has been driven low by the system bus master. The $\overline{\text{RASEN}}$ (RAS enable) signal at IC67 pin 11, produced from the output of IC57 ORed with $\overline{\text{DRAMSEL}}$, enables the 1-of-4 (2- to 4-line) decoder IC74 to select one of the four lines $\overline{\text{RAS0}}$, $\overline{\text{RAS1}}$, $\overline{\text{RAS2}}$, or $\overline{\text{RAS3}}$ (row-address-input enable for each of the four banks—which one is selected depends on the logic levels of the SYSA16 and SYSA17 address lines) and sets up the row address on the multiplexed memory-address lines RAMA0 through RAMA7. A chain of Schmitt-trigger inverter sections, IC63 and IC66, delays the active-low output from IC57 pin 3 by five gate-delay periods, holding the row-address condition until the type-4164 memory chips have had sufficient time to latch the address bits.

When the $\overline{\text{ROW/COL}}$ signal goes high (column addressing active), the

multiplexers change the contents of RAMA0 through RAMA7 to the column address derived from the system-address-bus lines SYSA8 through SYSA15. The CASEN signal enables the B outputs of the 1-of-4 decoder IC74, which drives the CAS-control line for one of the memory banks.

The data-input and data-output lines of each RAM chip are tied together onto a common bidirectional memory-data line. The entire RAM array is isolated from the system data bus by bus transceiver IC53, which is enabled by the DRAMSEL signal during nonrefresh memory cycles, allowing data to pass between the RAM array and the system data bus.

The direction of data flow is controlled by the output of IC67 pin 6, a logical OR of DRAMSEL and SYSMEMRD. During memory-read cycles, this signal is low, causing the data on the memory data bus to be transferred to the system data bus. During memory-write cycles, the direction signal is high and the data flow is from the system data bus to the memory data bus.

Parity Checking

Until the introduction of the IBM Personal Computer, memory with parity checking was rare in personal computers but had been used for years in larger computers. IBM did well to copy this feature of larger machines, since the constant decreases in memory prices have made it more and more cost-effective. The MPX-16 also incorporates parity memory for increased system reliability and user confidence. Parity generation and checking in the MPX-16 are provided by a 74LS280 parity generator (IC59) and a type-D flip-flop (IC58), shown in figure 3 on page 54.

During a memory-write cycle, the PARITYOUT signal presented to pin 8 of IC59 is low, because the output of IC57 (an AND gate) is disabled by the low state of the active-high SYSMEMRD signal. The parity bit computed by IC59 from the eight RAM data lines is written into the parity-bit memory chip (the ninth one of each bank) for the bank being addressed.

When a memory-read cycle occurs, the output of IC57 is enabled, and the parity bit that was previously written for each byte is routed to IC59 and used to check for an error in the parity value. When the rising edge of the signal from IC67 pin 6 (DRAMSEL OR SYSMEMRD) is detected by the flip-flop IC58, it latches parity value.

When no parity error is present, the odd-parity output (Σ O) of the 74LS280 will be a logic high state. When an error does occur, the odd-parity output will be low. The PARERR signal from IC58 is sent to the NMI (nonmaskable interrupt) logic and will remain set until the next memory-read cycle for which no parity error occurs, or until the flip-flop is preset by a low state on the CLRPAR (clear parity) line, IC58

pin 4. Software called through the interrupt vector then notifies the user of a memory error.

Interrupt Advantages

The versatility of any computer system is enhanced if its processing can be interrupted by outside events so that it doesn't have to continually keep track of what is going on in the outside world. The MPX-16 supports an interrupt system with 16 levels of interrupt priority, for a high degree of versatility in dealing with the external environment.

Perhaps the major advantage to using interrupts is the increase in throughput resulting from their use in handling the system I/O functions. Instead of the processor's spending a great deal of time checking to see if I/O devices are ready to transfer data or waiting for them to be ready, in an interrupt-driven system the processor can continue executing its application program, only suspending execution to attend to an I/O device when the device signals that it is actually ready for data transfer.

Although it can be tougher to debug, interrupt-driven software is generally more compact and efficient than that which must explicitly check I/O devices by polling or waiting. But we don't have space here to discuss the software aspects at length.

MPX-16 Interrupt Logic

The interrupt structure of an 8088-based system revolves around an interrupt-vector lookup table located low in system memory from location hexadecimal 00000 through 003FF. Each interrupt vector in the table consists of 4 bytes that point to the address of an interrupt-service routine. Up to 256 interrupt vectors, numbered from decimal 0 to 255, can be used to specify starting addresses of interrupt routines anywhere in the 8088's 1-megabyte address space. Each of the interrupt vectors is assigned an interrupt-type number that points to its location in the lookup table. The type number multiplied by 4 equals the offset of the vector from location 00000.

The highest priority interrupt is the

nonmaskable-interrupt (NMI) input at pin 17 of the 8088 microprocessor, IC36. This signal is an internally synchronized edge-triggered input which causes a predefined "type-2" interrupt that "vectors" (passes control) to the location identified by the eighth position in the table. Although the 8088's NMI input is not directly maskable by software, the MPX-16 contains extra hardware that can mask the interrupt signal before it gets to the 8088, given proper setup by the software. The NMI input is used to report system memory-parity errors and errors from the I/O-channel expansion slots.

The next 15 levels of interrupts are implemented by two Intel 8259A programmable interrupt controllers (PICs), IC35 and IC62 in section 1 of the schematic diagram, which was printed in last month's article. One of the programmable interrupt controllers, IC35, serves as the master and resides on the multiplexed local bus shared with the processors. The other, IC62, is a slave device to IC35 and resides on the system bus. The master/slave configuration is set up during the initialization process by software.

All of the peripheral devices residing on the system board, such as the serial and parallel I/O-port controllers, are supported by interrupt-request lines on the 8259A PICs. Interrupt requests from the PICs drive the INTR input of the 8088 (pin 18). This signal is a level-triggered input that can be internally masked by a software instruction. Interrupts requested by the INTR input do not have predefined vector types as does the nonmaskable interrupt. In the case of the 8259A PIC, a consecutive block of eight interrupt types, one for each of the eight interrupt-request input pins, is programmed into the device by the system software as part of the initialization process when the power is turned on.

Handling Interrupts

When an interrupt signal is received on the 8088's INTR pin, the processor enters an interrupt-acknowledge cycle that is used to determine the interrupt type. First the processor preserves what it was doing when interrupted: the state of the machine is saved by pushing the contents of the flag register, code-segment register, and instruction pointer onto the stack. In addition, the interrupt flag is cleared, disabling further interrupts from occurring until the processor is ready for them. (If nested interrupts are desired, the interrupt-service routine must reenable the processor to receive interrupts, while ensuring that the most crucial tasks are not delayed until too late. The programming is not easy.)

In the next step, the 8288 bus controller (IC51) issues two interrupt-acknowledge pulses on the $\overline{\text{INTA}}$ line. The first pulse signals the 8259A PICs that the interrupt request is being granted. When the second $\overline{\text{INTA}}$ pulse is issued, the 8-bit code for the interrupt type is placed onto the data bus. The value of the interrupt type is multiplied by 4 (simply by being shifted left 2 places) to determine the

Priority Level	Source	Signal Name	Description
0	NMI	PARERR or IOCHNLERR	memory-parity or I/O-channel errors
1	master	TIMEINTR	real-time clock
2	slave	SIO0RXRDY	serial-channel-A receive ready
3	slave	SIO1RXRDY	serial-channel-B receive ready
4	slave	SIO0TXRDY	serial-channel-A transmit ready
5	slave	SIO1TXRDY	serial-channel-B transmit ready
6	slave	PRINTRDY	printer-port ready
7	slave	FDCINT	floppy-disk-controller interrupt
8	slave	NPXINT	numeric-processor-extension (8087) interrupt
9	slave	PIOINT	parallel-I/O-port interrupt
10	master	IRQST2	I/O-channel interrupt
11	master	IRQST3	I/O-channel interrupt
12	master	IRQST4	I/O-channel interrupt
13	master	IRQST5	I/O-channel interrupt
14	master	IRQST6	I/O-channel interrupt
15	master	IRQST7	I/O-channel interrupt

Table 3: *Interrupt signals in the MPX-16, listed in order of priority. Priority-0 errors go through the 8088's NMI input, while the rest go through either the master or the slave 8259A interrupt controller.*

address of the interrupt vector. Program control is then transferred to the address contained in the 4 bytes of the interrupt vector. Note that the first 2 bytes are used as the new instruction pointer (lower 16 bits of the address) and the second 2 bytes are used to form the new code-segment register (upper 16 bits). When the interrupt-service routine has completed execution, control is returned to the main program via an IRET instruction, which pops the original flag and address information off the stack into the active registers. The main program then resumes execution where it left off, with the interrupts reenabled.

Interrupt Priorities

The organization of the system-board interrupt-priority scheme is shown in table 3. The highest priority hardware interrupt, as we've seen, is the NMI, which is caused by memory-parity or I/O-channel errors. The highest priority maskable interrupt is from the IR0 input of the master 8259A PIC, which is generated by the real-time clock. The next eight interrupts in priority come from peripheral devices attached to the slave 8259A PIC, which is in turn attached to the IR1 input of the master 8259A. The last six interrupts come from the I/O-expansion-channel connectors. These interrupts drive the IR2 through IR7 inputs of the master 8259A.

Two other points concerning the 8259A PICs should be noted. Although a priority has been assigned to each interrupt-request input of the 8259A PICs, these can be changed by the system software. In addition, the 8259A PICs can even be used to implement a polled I/O system. (These devices provide considerable flexibility for handling I/O servicing at a relatively low hardware cost.) And finally, all of the interrupt-service routines in the MPX-16 system can be be invoked via a software-interrupt instruction that specifies the interrupt type. This can be useful in starting an I/O device and in debugging the interrupt routines.

I/O-Expansion Channels

The MPX-16 system board supports an I/O-expansion channel that represents an extension of the system bus. Peripheral devices are connected through several 62-pin card-edge connectors like those used by peripherals designed for the IBM Personal Computer. The MPX-16 computer system

Pin	Signal Name		Pin	Signal Name	
A01	I/OCHNLERR		B01	GND	
A02	SYSDAT7		B02	BRESET	
A03	SYSDAT6		B03	V_{cc}	
A04	SYSDAT5		B04	IRQST2	
A05	SYSDAT4		B05	−5 V DC	
A06	SYSDAT3	system	B06	DMARQST2	
A07	SYSDAT2	data	B07	−12 V DC	
A08	SYSDAT1	bus	B08	IOCHLDRQST	
A09	SYSDAT0		B09	+12 V DC	
A10	IOCHNLRDY/WAIT		B10	GND	
A11	SYSAEN		B11	SYSMEMWR	system-status control
A12	SYSA19		B12	SYSMEMRD	
A13	SYSA18		B13	SYSIOWR	
A14	SYSA17		B14	SYSIORD	
A15	SYSA16		B15	DMACK3	
A16	SYSA15		B16	DMARQST3	DMA control
A17	SYSA14		B17	DMACK1	
A18	SYSA13		B18	DMARQST1	
A19	SYSA12	system	B19	BDMACK0	
A20	SYSA11	address	B20	BSYSCLK0	
A21	SYSA10	bus	B21	IRQST7	
A22	SYSA9		B22	IRQST6	
A23	SYSA8		B23	IRQST5	
A24	SYSA7		B24	IRQST4	
A25	SYSA6		B25	IRQST3	
A26	SYSA5		B26	DMACK2	
A27	SYSA4		B27	TCNT	
A28	SYSA3		B28	SYSALE	
A29	SYSA2		B29	V_{cc}	
A30	SYSA1		B30	BSYSCLK1	
A31	SYSA0		B31	GND	

Table 4: *Pin/signal relationships in the I/O-expansion connectors. These assignments are compatible with those in the expansion slots of the IBM Personal Computer. Many of the system control signals are buffered before being fed to these connectors.*

can potentially contain 1 megabyte of memory and still have spare expansion slots for special-purpose I/O modules, which might include videotex decoders, process-control or data-acquisition interfaces, or local-network interfaces.

The standard MPX-16 system board has five expansion connectors installed in alternating positions, effectively located on 1-inch center-to-center spacings. An additional four connectors can be installed between them, if needed; the resulting nine connectors will be on half-inch center-to-center spacing. Spacing on 1-inch centers is usually required for disk controllers and I/O boards. Memory boards, on the other hand, will generally fit in half-inch spacing.

The I/O-expansion channel has been designed to be pin-for-pin hardware-compatible with the IBM Personal Computer (model 5150). The IBM PC bus was chosen, as I explained last month, to take advantage of the expected proliferation of IBM-PC-compatible peripheral-adapter modules and expansion memories. However, because the MPX-16 system board already supports most of the peripheral I/O functions that would ordinarily be added to the IBM computer, the I/O-expansion slots are available for new uses.

Table 4 lists the signal connections to the pins of the I/O-expansion connectors. All signal lines in the I/O channel are compatible with LS-TTL (low-power Schottky-diode-clamped transistor-transistor logic) signals. Brief descriptions of each group of lines follow.

Oscillator Clock (BSYSCLK1): This is a buffered version of the main system timing clock. It runs at a frequency of either 14.31818 MHz or 15.0 MHz, depending on which crystal is installed. It has a 50 percent duty cycle.

System Clock (BSYSCLK0): This is a buffered version of the system processor clock. It runs at a frequency that is one-third that of SYSCLK1. It has a 33 percent duty cycle (high for one-third of the cycle, low for two-thirds).

System Reset (BRESET): This is a buffered version of SYSRES, which is active on power-up. It is synchronized to the falling edge of the SYSCLK0 waveform and is used for initialization of all system hardware components.

Address Latch Enable (SYSALE): This signal is used to indicate the presence of a valid address on the system bus. The falling edge of SYSALE is normally used to latch the address. This signal is generated by the 8288 bus controller during bus cycles initiated by one of the local-bus masters. The system-address enable signal, SYSAEN, should be used to enable this signal in the I/O channel.

System Address Enable (SYSAEN): This line, when active-low, indicates that one of the system coprocessors (either the 8088 or the 8087) has control of the system bus. When SYSAEN is high, the 8237A-5 DMA controller has control of the system bus and drives the system address, system memory, and I/O-read/write lines.

I/O Channel Ready (IOCHNLRDY/WAIT): This line is normally high. When a slow I/O device or expansion memory board decodes a valid address, this line should be driven low, causing the flip-flops IC25 and IC33 to insert wait states into the bus cycle until the slow device has completed its cycle. (To avoid conflict with memory refresh, this line should never be held low for more than 1 or 2 μs.)

System Memory Read (SYSMEMRD): This control line is used to gate the memory-device data buffers onto the system data bus during memory-read cycles initiated by either the processor or DMA controller.

System Board Peripheral Device	Base Address (hexadecimal)
8237A-5 DMA controller	000
8272 floppy-disk controller	020
DMA page registers 0 and 1	040
DMA page register 2	060
DMA page register 3	080
floppy-disk-drive motor-on register	0A0
parity-error flip-flop clear	0C0
spare (reserved)	0E0
spare (reserved)	100
8259A interrupt controller—slave	120
8259A interrupt controller—master	140
console serial I/O port	160
auxiliary serial I/O port	180
8255A-5 parallel I/O	1A0
8155H-2 parallel I/O and timer	1C0
8253-5 counter-timers	1E0

Table 5: *Base addresses of the I/O-device-control registers.*

System Memory Write (SYSMEMWR): This control is used to store the data present on the system data bus into the selected memory location during memory-write cycles initiated by either the processor or DMA controller.

System I/O Read (SYSIORD): This control line is used to gate the selected I/O device to accept the data present on the system data bus during I/O-read cycles initiated by either the processor or the DMA controller.

System I/O Write (SYSIOWR): This control line tells the selected I/O device to accept the data present on the system data bus. It is active in I/O-write cycles initiated by either the processor or DMA controller.

I/O-Channel (Parity) Error (I/OCHNLERR): This signal, when enabled by the system software, will cause an interrupt via the NMI input of the 8088 processor. It is normally used to alert the processor to a parity error in memory devices residing in the I/O channel.

System Address Bus (SYSA0 through SYSA19): These lines form a 20-bit system address bus, which can address up to 1 megabyte of memory. SYSA0 represents the least significant address bit (LSB), and SYSA19 represents the most significant address bit (MSB). These lines can be driven either from the processor or from the DMA controller and are considered to be active-high.

The MPX-16 computer system can potentially contain 1 megabyte of memory and still have spare expansion slots.

System Data Bus (SYSDAT0 through SYSDAT7): These lines form the 8-bit system data bus and can be driven by the processor, memory devices, or I/O devices. They are bi-directional and are considered to be active-high. SYSDAT0 is the LSB, SYSDAT7 the MSB.

I/O Channel Interrupt Requests (IRQST2 through IRQST7): These lines are prioritized interrupt-request lines, with IRQST2 having the highest priority and IRQST7 the lowest priority. The lines are edge-triggered and active-high; however, the request signal must be maintained in the high state until the interrupt request has been acknowledged. The interrupt-service routine written for each particular device in use must usually do this.

DMA Requests (DMARQST1 through DMARQST3): These lines are prioritized DMA-request lines, with DMARQST1 having the highest priority and DMARQST3 the lowest priority. The lines are active-high and must be held high until the corresponding $\overline{\text{DMACK}x}$ line goes active-low. DMARQST2 is used by the system-board floppy-disk controller and is included in the I/O channel only for compatibility with the IBM Personal Computer. These lines are typically used by peripheral devices such as disk controllers to request DMA service.

DMA Acknowledge Lines ($\overline{\text{DMACK1}}$ through $\overline{\text{DMACK3}}$): These lines are used to acknowledge DMA requests generated by the DMARQSTx lines.

DMA Acknowledge 0 ($\overline{\text{BDMACK0}}$): This is a buffered $\overline{\text{DMACK0}}$ line and signifies that a DMA-controlled dynamic-memory-refresh cycle is in progress.

DMA Terminal Count (TCNT): This signal is active-high when any of the four DMA channels reaches a terminal count. The corresponding DMA-acknowledge line should be used in conjunction with the TCNT signal.

Peripheral Power: +5 volts (V) DC ±5%, logic ground, +12 V DC ±5%, −12 V DC ±10%, and −5 V DC ±10% power connections are all provided in each expansion connector.

I/O-Decoder Logic

The MPX-16 computer system contains a variety of onboard, high-performance peripheral devices: direct support for all of the major I/O functions needed to form a complete microcomputer system, as listed in table 1 on page 44.

All of the system-board I/O peripherals are addressed or selected by the 4-to-16 decoder IC52 (shown in

section 2, figure 2 on page 48). This decoding logic maintains addressing compatibility with IBM Personal Computer peripherals by using the system-address-bus line SYSA9 to determine whether the peripheral device being selected is on the main circuit board or off it. A low state on the SYSA9 line enables one of the strobe inputs of the decoder; the other strobed input is enabled if one of the local bus masters has control of the system bus, indicated by a low state on $\overline{88AEN}$. When an I/O-device interface chip is selected by this decoded address and either the $\overline{SYSIORD}$ or $\overline{SYSIOWR}$ line is active, an I/O bus cycle is performed. During DMA cycles the I/O decoder is disabled.

The base address for each of the system-board I/O devices is shown in table 5 on page 76. The total number of address-space locations used by each peripheral device varies; this will be discussed in more detail next month in part 3.

Next Month:

If you've followed everything in this second installment on the Circuit Cellar MPX-16 computer system, you're doing well. In the January article I'll fill you in on the serial and parallel I/O ports, counters, floppy-disk controller, and operating-system BIOS, among other topics.■

Acknowledgments

Thanks to Jim Norris, George Martin, and Linda Spencer of Owl Electronic Laboratories for their contributions to this project.

Thanks to Mark Dahmke and Gordon Heins for their help with the documentation.

Thanks to Bill Morello and his staff at Techart Associates for their careful work in drawing the schematic diagrams.

Editor's Note: *Steve often refers to previous Circuit Cellar articles as reference material for each month's current article. Most of these past articles are available in reprint books from BYTE Books, McGraw-Hill Book Company, POB 400, Hightstown, NJ 08520.*

Ciarcia's Circuit Cellar, Volume I, *covers articles that appeared in BYTE from September 1977 through November 1978.* Ciarcia's Circuit Cellar, Volume II, *contains articles from December 1978 through June 1980.* Ciarcia's Circuit Cellar, Volume III, *contains the articles that were published from July 1980 through December 1981.*

To receive a complete list of Ciarcia's Circuit Cellar project kits available from the Micromint, circle 100 on the reader service inquiry card at the back of the magazine.

References

1. Cantrell, Thomas. "An 8088 Processor fo the S-100 Bus." Part 1, September 198(BYTE, page 46. Part 2, October 198(BYTE, page 62. Part 3, November 198(BYTE, page 340.
2. Ciarcia, Steve. "Build the Circuit Cella MPX-16 Computer System, Part 1." November 1982 BYTE, page 78.
3. Ciarcia, Steve. "Ease into 16-Bit Com puting." Part 1, March 1980 BYTE, pag 17. Part 2, April 1980 BYTE, page 40. Re printed in *Ciarcia's Circuit Cellar, Volum II.* Peterborough, NH: BYTE Books, 198 page 171.
4. Ciarcia, Steve. "The Intel 8086. November 1979 BYTE, page 14. Re printed in *Ciarcia's Circuit Cellar, Volum II.* Peterborough, NH: BYTE Books, 198 page 120.
5. *Component Data Catalog.* Santa Clar CA: Intel Corporation, 1981.
6. *iAPX Book.* Santa Clara, CA: Intel Corpor tion, 1981.
7. *iAPX 86,88 User's Manual.* Santa Clar CA: Intel Corporation, 1981.
8. Morgan, Christopher L. and Mitche Waite. *8086/8088 16-bit Microprocesso Primer.* Peterborough, NH: BYTE Book 1982.

The following items are available from:

The Micromint Inc.
561 Willow Ave.
Cedarhurst, NY 11516
(516) 374-6793
(for technical information)
(800) 645-3479
(for orders only)

1. MPX-16 single-board computer system: assembled, tested, and burned-in. Includes 64K bytes of RAM, Digital Research CP/M-86 operating system on 8-inch or 5¼-inch floppy disk, CP/M-86 BIOS in EPROM, MPX-16 Technical Reference and User's Manual. *Requires power supply and floppy-disk drive.*
Single-quantity price.....$1895

2. MPX-16 single-board computer system, as above, but with 256K bytes of RAM installed.
Single-quantity price.....$2135

3. MPX-16 single-board computer system, with 64K bytes of RAM.
In OEM quantities of 100 $1200 each

4. Complete MPX-16 disk-based system: includes MPX-16 single-board computer, assembled, tested, and burned-in, with 256K bytes of RAM installed, CP/M-86 operating system on 5¼-inch floppy disk, CP/M-86 BIOS in EPROM, power supply, one 5¼-inch single-sided floppy-disk drive, connecting cables, MPX-16 Technical Reference and User's Manual. *Enclosure sold separately.*
Single-quantity price.....$2895

5. Unpopulated (blank) printed-circuit board for the MPX-16 computer system: five-layer, screened, and solder-masked. Includes CP/M-86 BIOS in EPROM, MPX-16 Technical Reference and User's Manual.
Single-quantity price.....$300

6. Digital Research CP/M-86 User's Manual *(three-volume set), sold separately.....$40*

7. MPX-16 Technical Reference and User's Manual, *sold separately....$50*

8. Enclosure for MPX-16 circuit board.....call for price

* * *

When it becomes available for the MPX-16, Microsoft's MS-DOS operating system may be optionally substituted for CP/M-86.

The MPX-16 is available to OEMs in large quantities either as a circuit board or as a complete system with floppy-disk drives and enclosure. Call the Micromint for prices and delivery information.

For orders within the continental United States, please include $10 for shipping; overseas orders please include $30. Residents of New York please include 7 percent sales tax.