Build the Disk-80

Memory Expansion and Floppy-Disk Control

The term "memory expansion" no longer causes the same pained expression among computer owners as it did a few years ago. Back in the "Model T" days of personal computing, it was a major undertaking, often the largest expense of setting up a personal computer system. At that time, volatile memory integrated circuits contained only a fraction of the capacity of those available today, and mass storage often consisted of slow audio cassettes.

Back in 1975, if you were adding 32 K bytes of memory to your computer you would probably have used 256 type-2102A 1 K by 1-bit static memory chips. These cost between $2 and $5 each, and you probably would have needed a power supply larger and heavier than the computer. Believe it or not, the 2102A was a major improvement over the previous type-1101 memories (256 by 1-bit static devices). It would have taken 1024 (1 K) of the type-1101 components to make 32 K bytes.

Since that time, memory technology has progressed by leaps and bounds, and the cost per bit has dropped considerably. Many computer manufacturers now use dynamic rather than static memory. The result is much higher density, lower system cost, and easier after-market memory expansion on most computers.

The 64 K-byte personal computer system is more common than you would imagine. Adding another 16 or 32 K bytes of memory these days simply means plugging a few (8 or 16) integrated circuits into a memory-expansion unit or motherboard. Usually the standard power supply suffices.

Even with these advantages, using dynamic memory is not quite as simple as it sounds. There are considerable differences between static and dynamic memory. Most people know relatively little about designing a dynamic-memory system, and even professional designers are intimidated by having to deal with multiplexing addresses, selecting bus drivers, sequencing activation of power supplies, and decoupling and noise. I don't expect that reading this article will make you into an authority on dynamic memory, but perhaps you will at least have a better understanding of it.

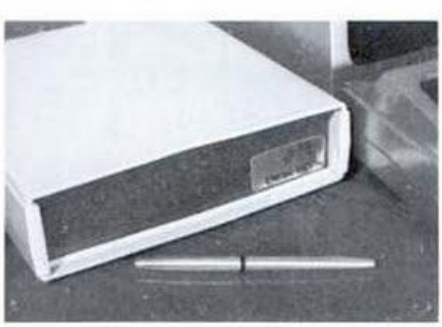


Photo 1: *The Disk-80 mounted in its enclosure and attached to the TRS-80 Model I computer. The Disk-80 is about half the size of the Radio Shack TRS-80 Expansion Interface.*

Mass-storage technology has also progressed during this same period. Displacing the audio cassette as the exclusive medium, the floppy disk and Winchester-technology hard disk have become the *de facto* storage standards.

For some time I have wanted to present articles on dynamic-memory and disk-controller integrated circuits. I have delayed chiefly because I generally prefer to present my articles as usable applications.

The three largest-selling personal computers, the Radio Shack TRS-80, the Apple II, and the Commodore

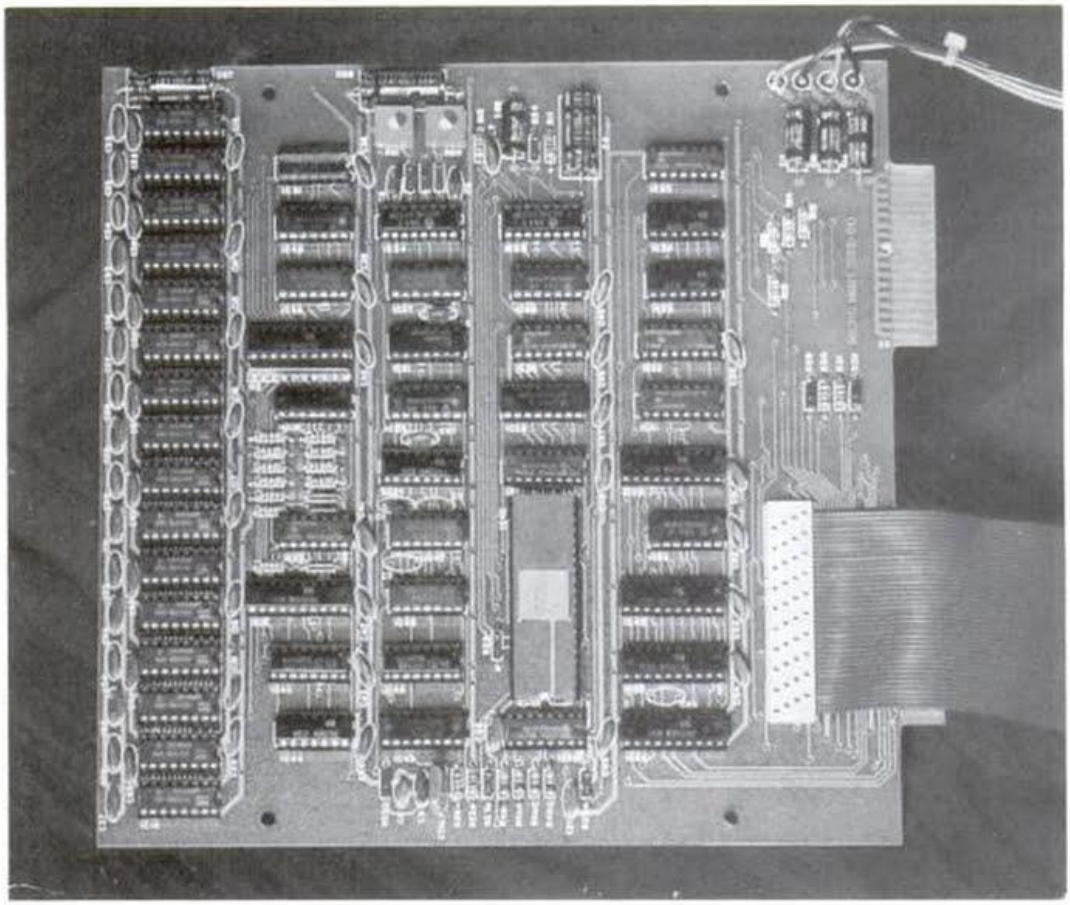


Photo 2: *The Disk-80 printed-circuit board (circuit of figure 2). The red cable connects to the expansion connector on the TRS-80 keyboard/processor module. The 34-pin edge connector above the red cable is a 5-inch floppy-disk drive interface connector. The sixteen integrated circuits lined against the left side of the board are 32 K bytes' worth of type-4116 16 K-bit dynamic memory devices.*

PET, use 16 K-bit dynamic-memory chips. Only the TRS-80 Model I requires the user to add extra memory via an external module, the TRS-80 Expansion Interface. (The TRS-80 Model III does not.)

This month's hardware project, the Disk-80, is an expansion interface for use with the TRS-80 Model I that expands the user memory and provides for the attachment and control of floppy-disk drives. Dynamic memory and a specialized floppy-disk-drive controller are used. Although this project was designed for use with the TRS-80 Model I, the elements of the systems and the principles involved are applicable to any personal computer.

What's Inside the Disk-80?

The Disk-80, shown in photo 1, is completely hardware- and software-compatible with the TRS-80 Model I and includes hardware enhancements for increased reliability. Readers familiar with the TRS-80 Expansion Interface will note that the Disk-80 is considerably smaller.

The keyboard/processor module of the TRS-80 Model I system is a single-board computer with memory-mapped video display and keyboard. The only provision for I/O (input/output) in the basic configuration is an I/O port for an audio-cassette recorder and a single-bit relay line for control of the recorder motor. Any user-memory expansion beyond 16 K bytes and any printer or disk I/O must be handled externally. The unit sold by Tandy/Radio Shack to perform these functions is called the TRS-80 Expansion Interface.

Figure 1 is a block diagram of the Disk-80. It attaches to the keyboard/processor unit through the 40-pin TRS-BUS connector and provides the following functions: 32 K-byte user-memory expansion, Centronics-compatible parallel printer port (full 8 bits), real-time clock, four-drive 5-inch floppy-disk controller, external data separator (used in reading floppy disks), buffered TRS-BUS connection to other peripheral devices, and power supply.

The three major functional sections are as follows:

- Memory-expansion section, which accommodates up to 32 K bytes of dynamic memory;
- Four-drive 5-inch floppy-disk controller; and
- Parallel printer port.

The activities of these sections are coordinated through a common address decoder.

The Disk-80 system is divided into two circuit boards. The main board, shown in photo 2 and outlined schematically in figures 2a, 2b, and 2c, contains everything except the power supply and the printer port. The other board, referred to as the power-supply/printer-interface board, is shown in photo 3 on page 104.

Disk-80 Addressing

The disk controller, printer, and real-time clock are addressed as memory-mapped parallel I/O ports through IC28, a 74LS155 decoder. Eight strobe signals are produced to decode memory addresses within the range of hexadecimal 37E0 to 37EC (only six of these are used in the Disk-80) to coordinate these peripherals. Their functions are shown in table 1 on page 98.

IC26, an eight-input NAND gate, and IC27, a 74LS139 decoder, function as memory-bank decoders. They

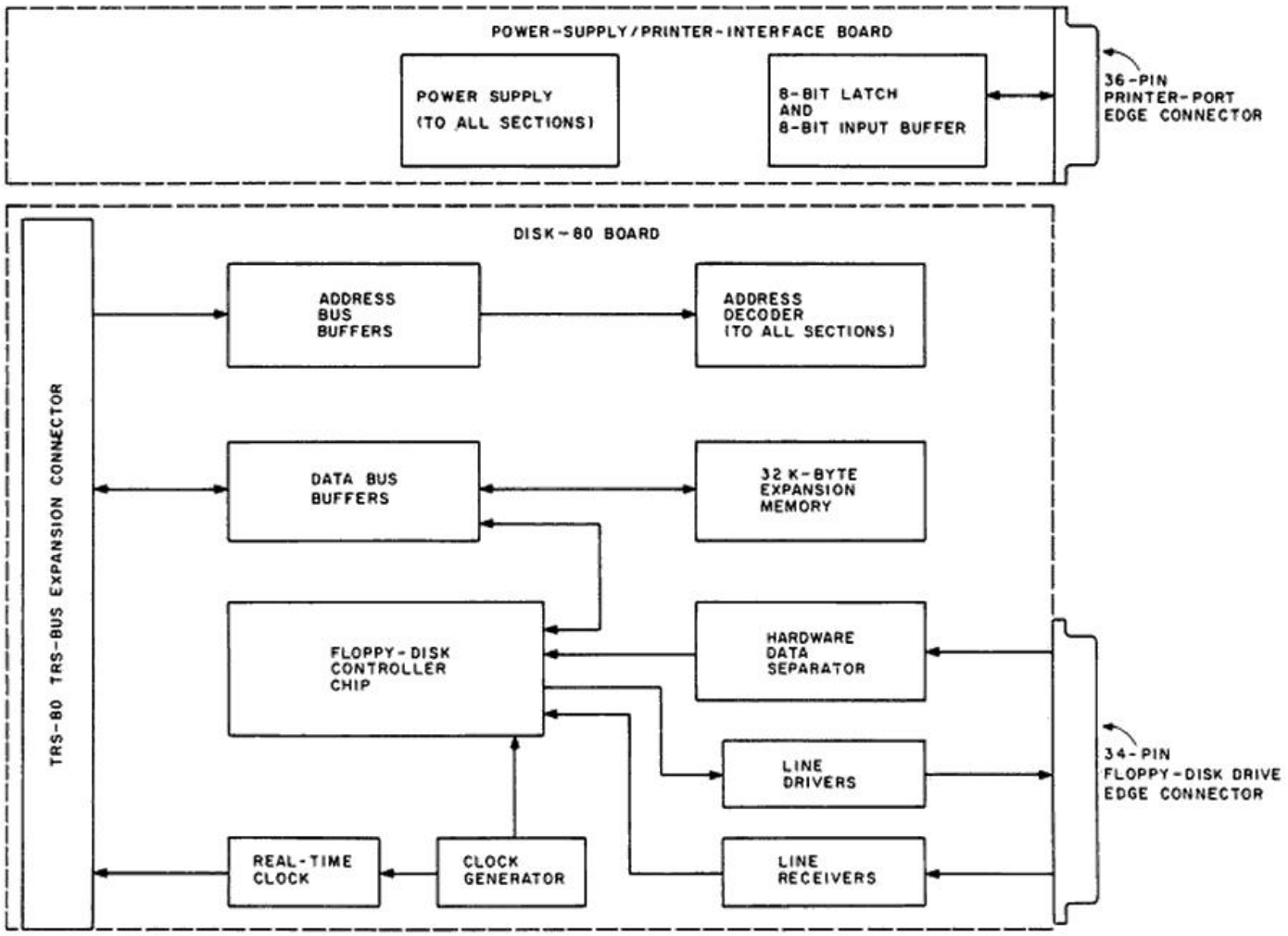


Figure 1: *Block diagram of the Disk-80 expansion interface for the Radio Shack TRS-80 Model I.*

Elements of the systems presented and the principles involved are applicable to any personal computer.

produce two strobe signals. One, designated $\overline{32\ K}$ (or informally, 32 K enable), is the active-low enable strobe for expansion memory between hexadecimal addresses 8000 and BFFF. The second strobe, $\overline{48\ K}$ (or 48 K enable), controls the bank of memory between C000 and FFFF.

Operation of Dynamic Memory

When designing memory systems, it is necessary to understand both the components and the computer system. In the past, the most popular dynamic memory components were MK4096- and MK4027-type 4 K by 1-bit devices. Today the standard memory component in personal computers is the MK4116. (More recently, maximum density has increased to 64 K bits per chip. Unfortunately, these parts are expensive, about $70 each, and are not yet generally used in personal computers.)

The 4116 is a 16 K-bit dynamic memory device. The 14 address bits required to specify one of the 16,384 cell locations that each store a single bit of data are multiplexed into seven shared pins. The timing of the signals presented to these pins is shown in figure 3 on page 104.

During execution of a Z80 memory-read or memory-write instruction, a 16-bit address is present on the processor's address bus. If the memory address is between hexadecimal 8000 and FFFF, the processor will try to find the addressed memory devices in the Disk-80's memory-expansion section. Decoding address lines A14 and A15 determines whether the location is in the 32 K or the 48 K memory range and enables the appropriate bank. The remaining 14 bits are multiplexed directly into the eight 4116s (one 4116 is used for each bit of the addressed byte location).

IC20 and IC21, 74LS157 quad 2-to-1-line multiplexers, apply the first 7 row-address bits to each 4116 when the MUX (multiplex) and $\overline{RAS}$ (row-address strobe) signals are low. This latches the row address into the 4116. Next, the MUX signal goes high, applying the 7 column-address bits to the 4116, and $\overline{CAS}$ (column-address strobe) goes low. At the conclusion of this sequence, data is either written into or read from the 4116

depending upon the polarity of the Write Enable input. In turn, the Read Enable line controls the direction of data flow through the memory data buffers, IC17 and IC18, 74LS244 non-inverting octal buffers.

The MUX, CAS, and RAS signals are generated within the TRS-80 keyboard/processor module in a 3-bit shift register. Figure 4 on page 105 illustrates, in simplified logic, the derivation of these signals.

At the beginning of each read or write cycle, the Z80 microprocessor's MREQ (memory request) line is pulled low. The MREQ signal is also used by the TRS-80 as the RAS signal. The RD and WR (negative-logic read-enable and write-enable) lines are logically ORed to feed the CLR (clear) inputs and the D input of FF1, the first flip-flop in the 3-bit shift register. When either RD or WR goes low, a logic 1 is loaded into FF1 at the occurrence of the rising edge of the 10.6445 MHz master clock pulse. On the next clock pulse, the logic 1 is shifted into FF2, the second flip-flop, of which the Q output controls the MUX signal. The next clock pulse shifts the logic 1 into flip-flop FF3. The inverted Q output of FF3 is the CAS signal to memory. When the RD or WR line goes high again, the three flip-flops are cleared and the address multiplexers are reset.

It is easy to see that multiplexing the addresses is fairly simple, especially when the signals needed are available on the 40-pin TRS-BUS connector.

Interestingly enough, Radio Shack did not use these signals in late-production TRS-80 Expansion Interfaces. Because some of the signal pulses are very short in duration (about 200 ns) and susceptible to noise, the early-production Expansion Interfaces had to have a buffered cable to eliminate memory errors. Eventually, this arrangement was

Notes

1. *On IC1 thru IC16 (the 4116 components) the +5 V lead on each IC should have one decoupling capacitor. One decoupling capacitor should be on every other chip for the +12 V and −5 V leads, for a total of thirty-two decoupling capacitors. Careful placement of decoupling capacitors is* absolutely critical *to proper operation.*

2. *All other places where decoupling capacitors are required are denoted by an asterisk (*) on the diagram.*

3. *All capacitors are 12 V ceramic disk type unless otherwise noted.*

4. *All resistors are 1/4 W 5% tolerance carbon-film type unless otherwise noted.*

Write Strobes
- 37E0 — disk-drive select (1 of 4)
- 37E4 — not used
- 37E8 — printer data out
- 37EC — set disk-controller registers

Read Strobes
- 37E0 — read real-time clock/reset interrupt
- 37E4 — not used
- 37E8 — read printer status
- 37EC — read disk-controller registers

Table 1: *Hexadecimal memory-mapped addresses of registers used by the Disk-80 to coordinate the disk-drive controller, the printer interface, and the real-time clock.*

IC Number	Type	+5V	GND	+12V	−5V
1 thru 16	4116 (200 ns)	9	16	8	1
17	74LS244	20	10		
18	74LS244	20	10		
19	74LS32	14	7		
20	74LS157	16	8		
21	74LS157	16	8		
22	74LS00	14	7		
23	74LS14	14	7		
24	74LS244	20	10		
25	74LS244	20	10		
26	74LS30	14	7		
27	74LS139	16	8		
28	74LS155	16	8		
29	7416	14	7		
30	7416	14	7		
31	74LS20	14	7		
32	74LS175	16	8		
33	74LS123	16	8		
34	74LS123	16	8		
35	74LS00	14	7		
36	74LS04	14	7		
37	74LS74	14	7		
38	74LS74	14	7		
39	74LS240	20	10		
40	74LS240	20	10		
41	74LS367	16	8		
42	INS1771D-1	21	20	40	1
43	CD4049	1	8		
44	74LS90	5	10		
45	CD4518	16	8		
46	CD4518	16	8		
47	74LS74	14	7		
48	74LS74	14	7		
49	74LS00	14	7		
50	74LS14	14	7		
51	*				

Table 2: *List of integrated circuits and power-wiring requirements for the Disk-80, excluding those integrated circuits found on the optional power-supply/printer-interface circuit board. The entity marked IC51 on the schematic diagram is really a connector for the 14-conductor ribbon cable running between the power-supply/printer-interface board and the disk-controller board.*

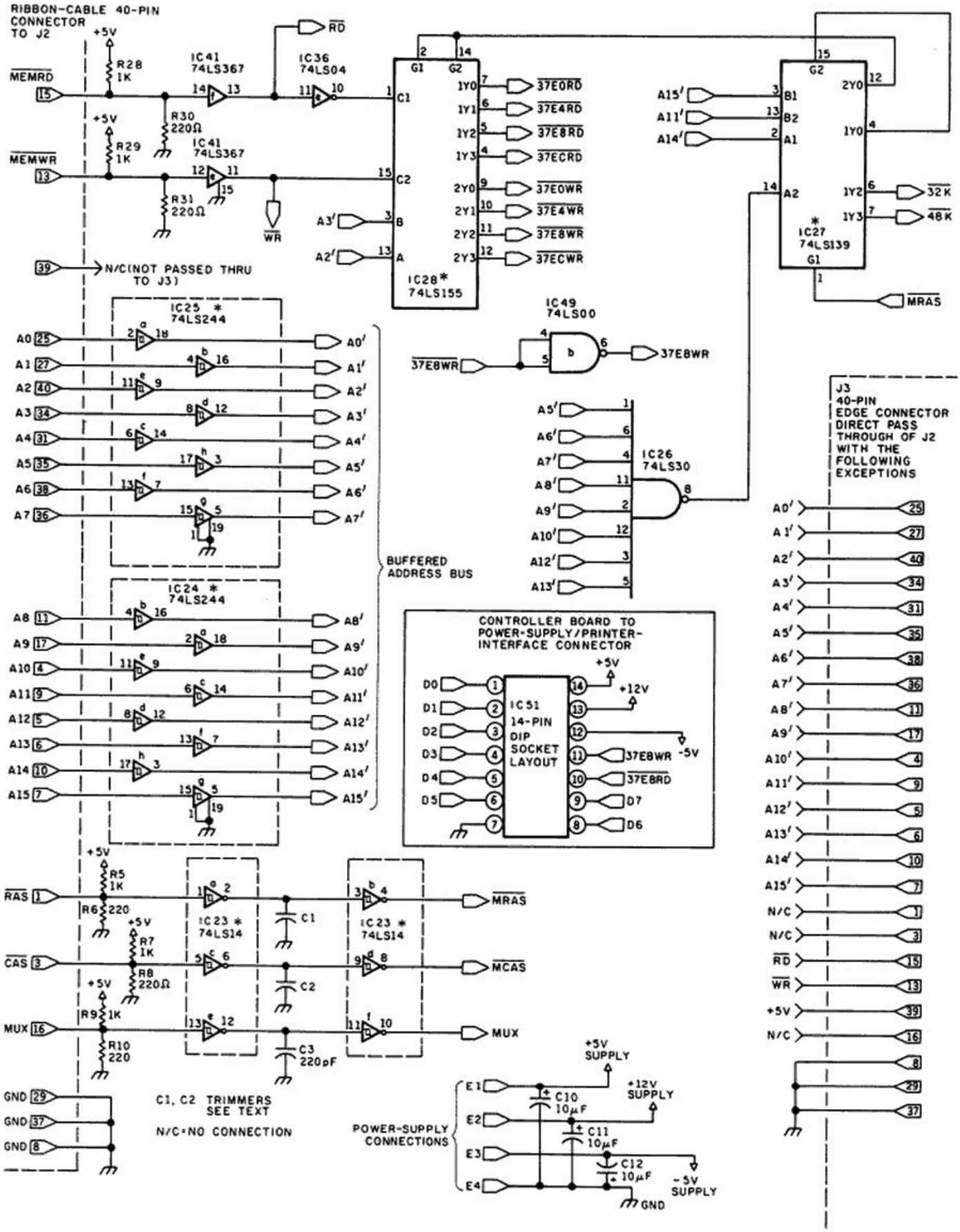


Figure 2a: *Schematic diagram of the address-decoding and buffering section of the Disk-80. Figure notes are found in the text box on page 98.*

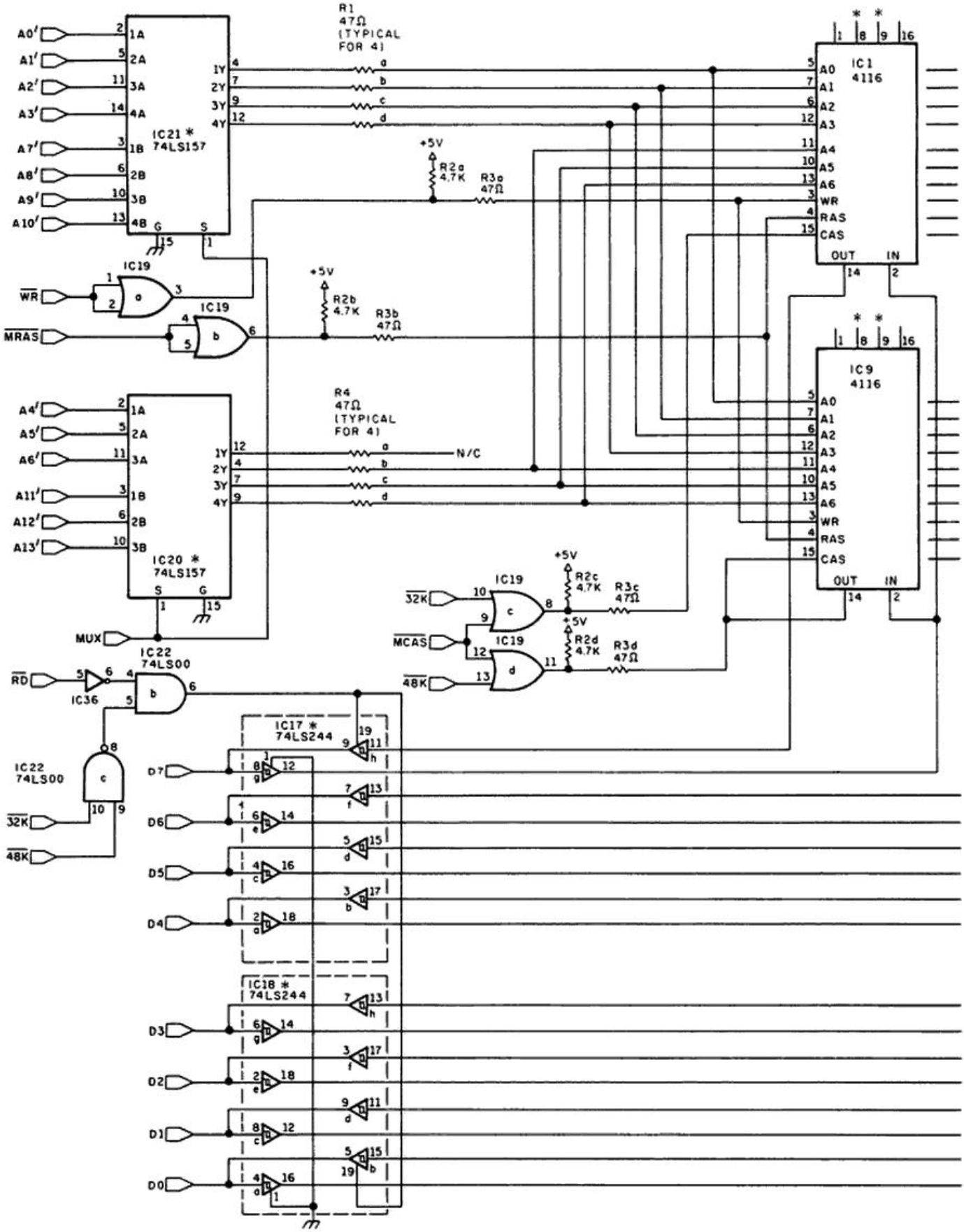
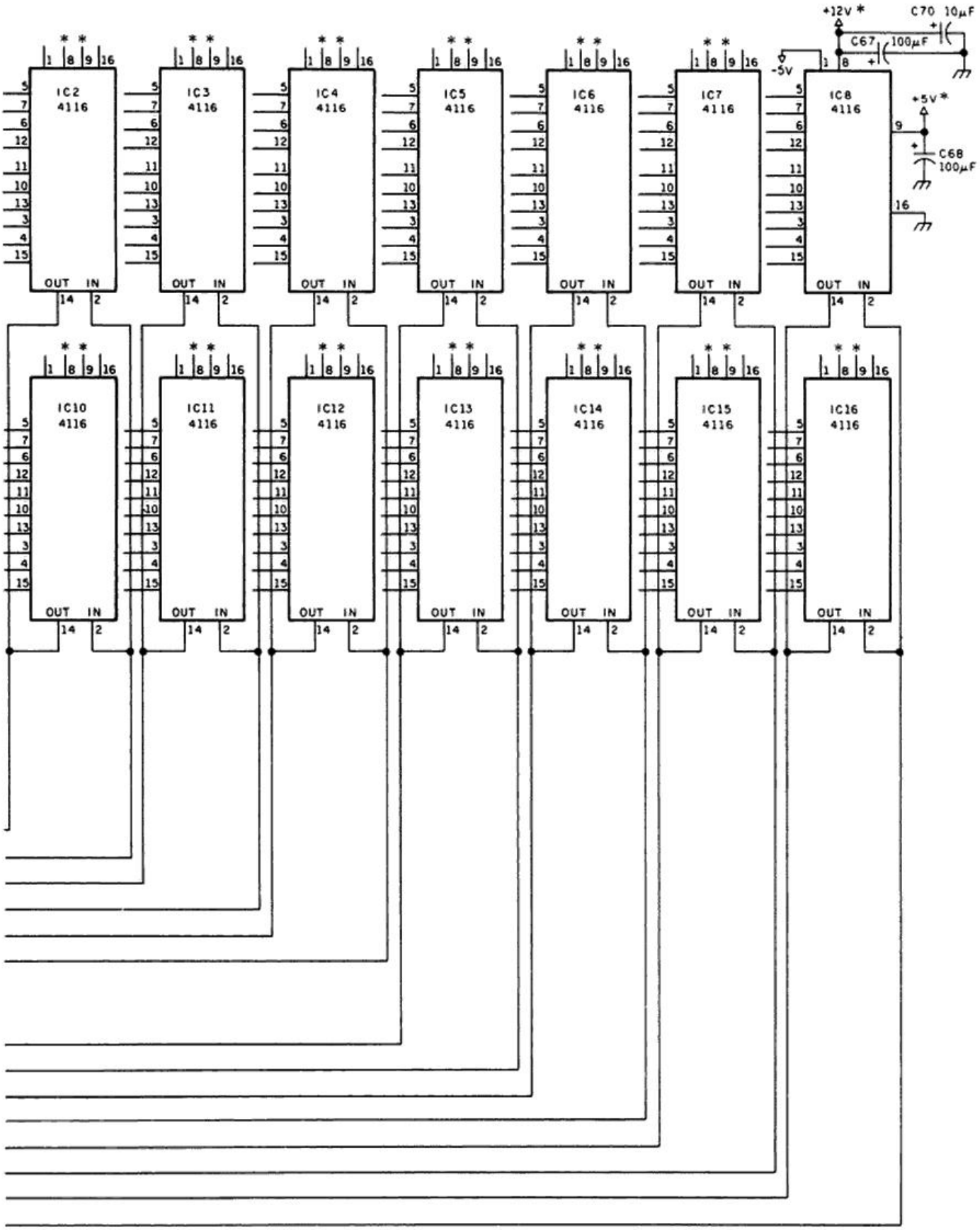


Figure 2b: *The memory-expansion section of the Disk-80 module. See notes on page 98.*


+12V *
C70 10µF
C67 100µF
-5V
+5V *
C68 100µF
IC2 4116
IC3 4116
IC4 4116
IC5 4116
IC6 4116
IC7 4116
IC8 4116
IC10 4116
IC11 4116
IC12 4116
IC13 4116
IC14 4116
IC15 4116
IC16 4116
OUT
IN

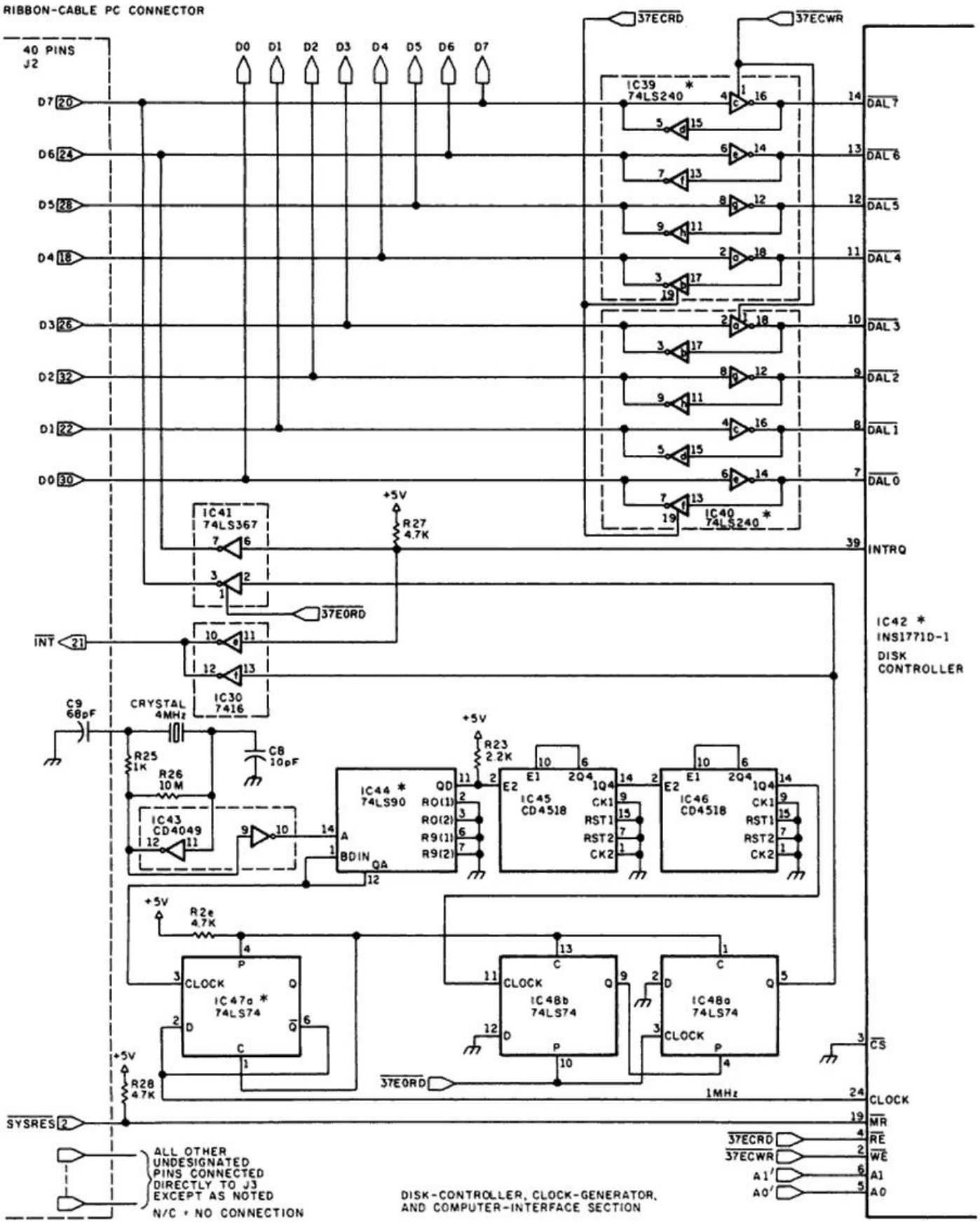
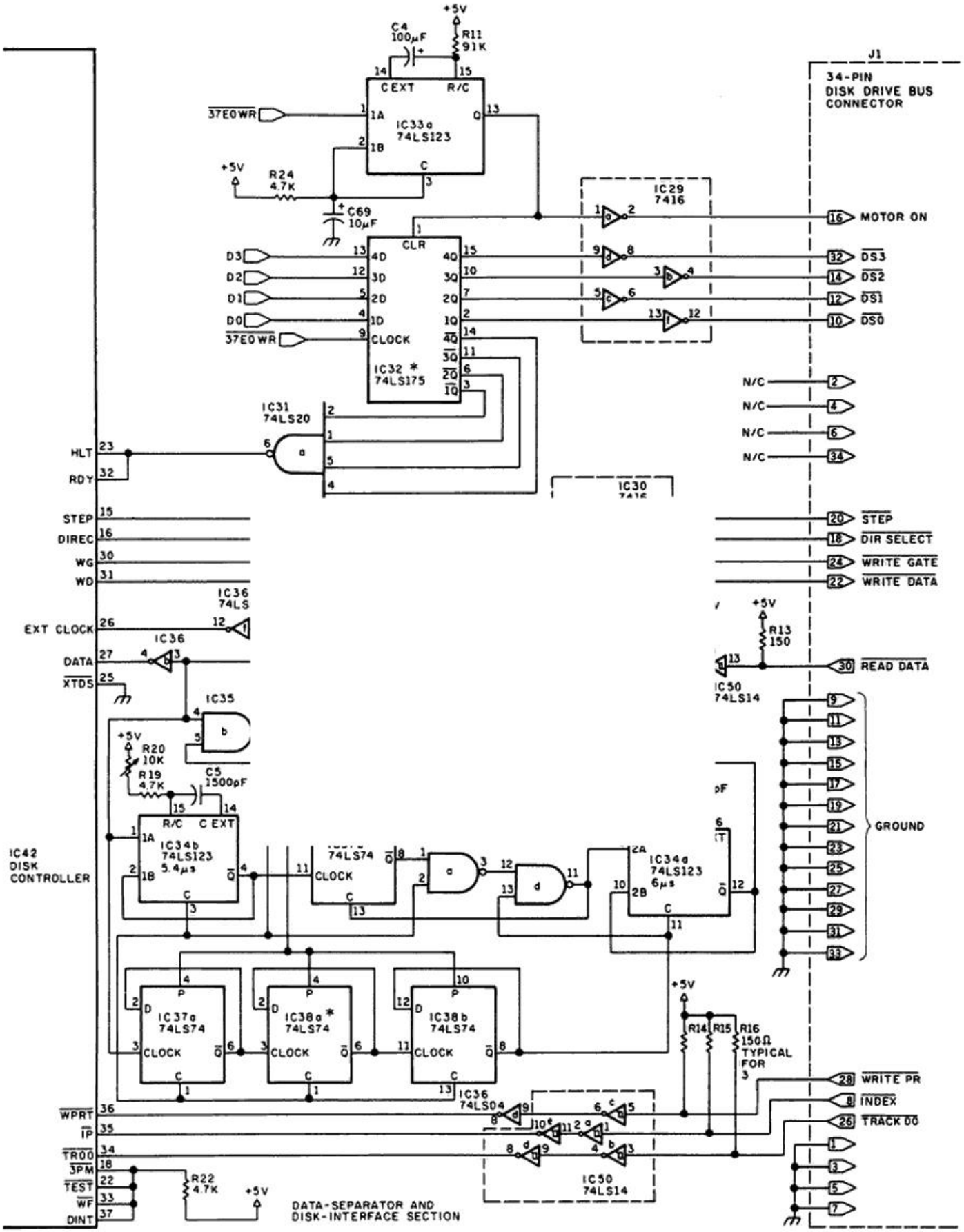


Figure 2c: *Schematic diagram of the disk-controller, clock-generator, keyboard/processor-module-interface, external-data-separator, and disk-drive-interface sections of the Disk-80 expansion module.* *See figure notes on page 98.*


+5V
C4 100µF
R11 91K
CEXT
R/C
37E0WR
IC33a 74LS123
R24 4.7K
C69 10µF
CLR
D3
D2
D1
D0
37E0WR
IC32 * 74LS175
IC29 7416
IC31 74LS20
HLT
RDY
IC30
J1
34-PIN DISK DRIVE BUS CONNECTOR
MOTOR ON
DS3
DS2
DS1
DS0
N/C
N/C
N/C
N/C
STEP
DIREC
WG
WD
STEP
DIR SELECT
WRITE GATE
WRITE DATA
IC36 74LS
EXT CLOCK
IC36
DATA
XTDS
IC35
+5V
R13 150
READ DATA
IC50 74LS14
R20 10K
R19 4.7K
C5 1500pF
IC42 DISK CONTROLLER
IC34b 74LS123 5.4µs
IC34a 74LS123 6µs
CLOCK
GROUND
IC37a 74LS74
IC38a * 74LS74
IC38b 74LS74
R14
R15
R16 150Ω TYPICAL FOR 3
WRITE PR
INDEX
TRACK 00
WPRT
IP
TR00
3PM
TEST
WF
DINT
IC36 74LS04
IC50 74LS14
R22 4.7K
+5V
DATA-SEPARATOR AND DISK-INTERFACE SECTION

replaced with a circuit in the Expansion Interface that derives the MUX and $\overline{CAS}$ signals by sending $\overline{RAS}$ through a delay line. Of the three original signals, only the $\overline{RAS}$ signal, which has the longest pulse duration, is used.

If cable lengths are kept to a minimum and proper signal termination is employed, there is no good reason why any signal available from the keyboard/processor module should not be used. The Disk-80 uses a combination of active termination and Schmitt-trigger inputs to guarantee reception of all available signals.

Memory Refreshing

So far you have heard only the good things about dynamic memory. One of the less desirable characteristics is called memory refreshing. Unlike static memory, which stores data in active bistable circuits composed of three transistors, the dynamic 4116 stores its 1s and 0s in single-transistor cells that simulate capacitors. As from a capacitor, the electrical charge that represents a bit slowly drains off unless it is "refreshed." Refreshing is accomplished by *addressing* all memory cells (or a required minimum of them) on a regular basis.

The 4116 is a $\overline{RAS}$-only-refresh device. Instead of addressing all 16,384 bit-cell locations, only the 128 rows are cycled. This type of refreshing uses only the $\overline{RAS}$ signal and is achieved in less time than methods that use both row and column addressing. Because the MUX and $\overline{CAS}$ pulses are not used, the memory is not enabled, and the refreshing does not interfere with other system operations. However, all 128 rows must be addressed at least every 2 ms to avoid loss of data.

Refresh circuits are generally binary counters that generate sequential addresses which are applied to the memory chips. The Z80 processor includes a built-in 8-bit $\overline{RAS}$-only refresh register. During the decoding and execution of an instruction op code, the 7 bits of the refresh register contents are placed on the low-order lines of the address bus, and the $\overline{MREQ}$ line is strobed. In effect, the Z80 accomplishes "hidden refresh" as it executes its normal program. For more information on this capability, I refer you to the Zilog *Z80-CPU Technical Manual*.

Sequencing the Power Supply and Decoupling

Unfortunately, in addition to refreshing dynamic memory, a designer has to be concerned about sequencing the turning on of the power supplies. While some brands of type-4116 memory devices are more tolerant than others, the following rule must be applied: *the −5 V supply (V_{BB}) must be applied to the 4116 before the +12 V supply (V_{DD}), and the −5 V supply must remain on until the +12 V supply has been removed.* The +5 V (V_{CC}) supply is less critical, but it is best to turn it on and off synchronously with the +12 V supply. Many dynamic-memory components have been destroyed by designers not adhering to these rules.

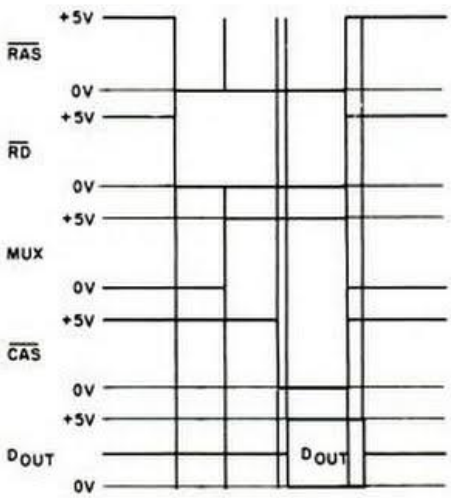


Figure 3: *$\overline{RAS}$, $\overline{CAS}$, and MUX timing diagram for 4116-type 16-pin dynamic-memory integrated circuit. A 14-bit address (16,384 by 1) is multiplexed into seven address pins. When MUX and RAS are low, the row-address bits are read into the 4116. Later, when MUX is high and CAS goes low, the column bits are read into the 4116, activating the data output for that memory cell.*

Supplying Power

Power-supply sequencing is important because many power supplies overshoot their rated voltages when they are turned on. If V_{BB} (−5 V) is not turned on and V_{DD} (+12 V) overshoots to more than +15 V, the chip will blow. Applying V_{BB} first provides an extra margin to prevent device destruction. Also, V_{BB} must never go positive with respect to any other input.

The Disk-80 power supply, shown in photo 3 with the optional printer interface, meets these requirements. It is designed such that the time constants of the various sections produce a phased start-up and shutdown. This sequential operation is primarily achieved by use of filter components

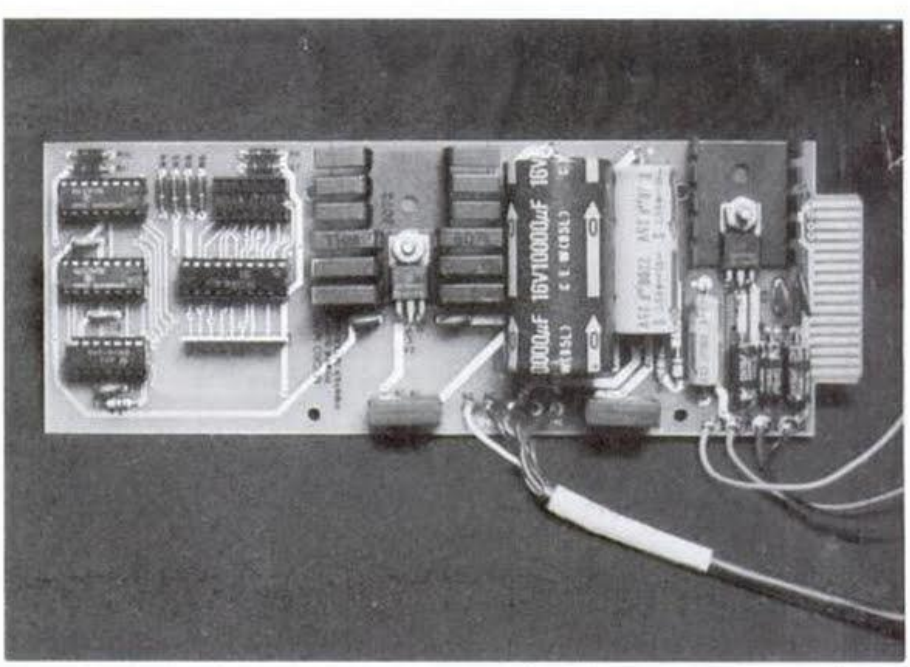


Photo 3: *The Disk-80 power supply with the parallel printer interface. The power supply provides 1 A at +5 V, 400 mA at +12 V, and 50 mA at −5 V, and is designed for use with dynamic memories such as the 4116 that require sequenced application of power.*

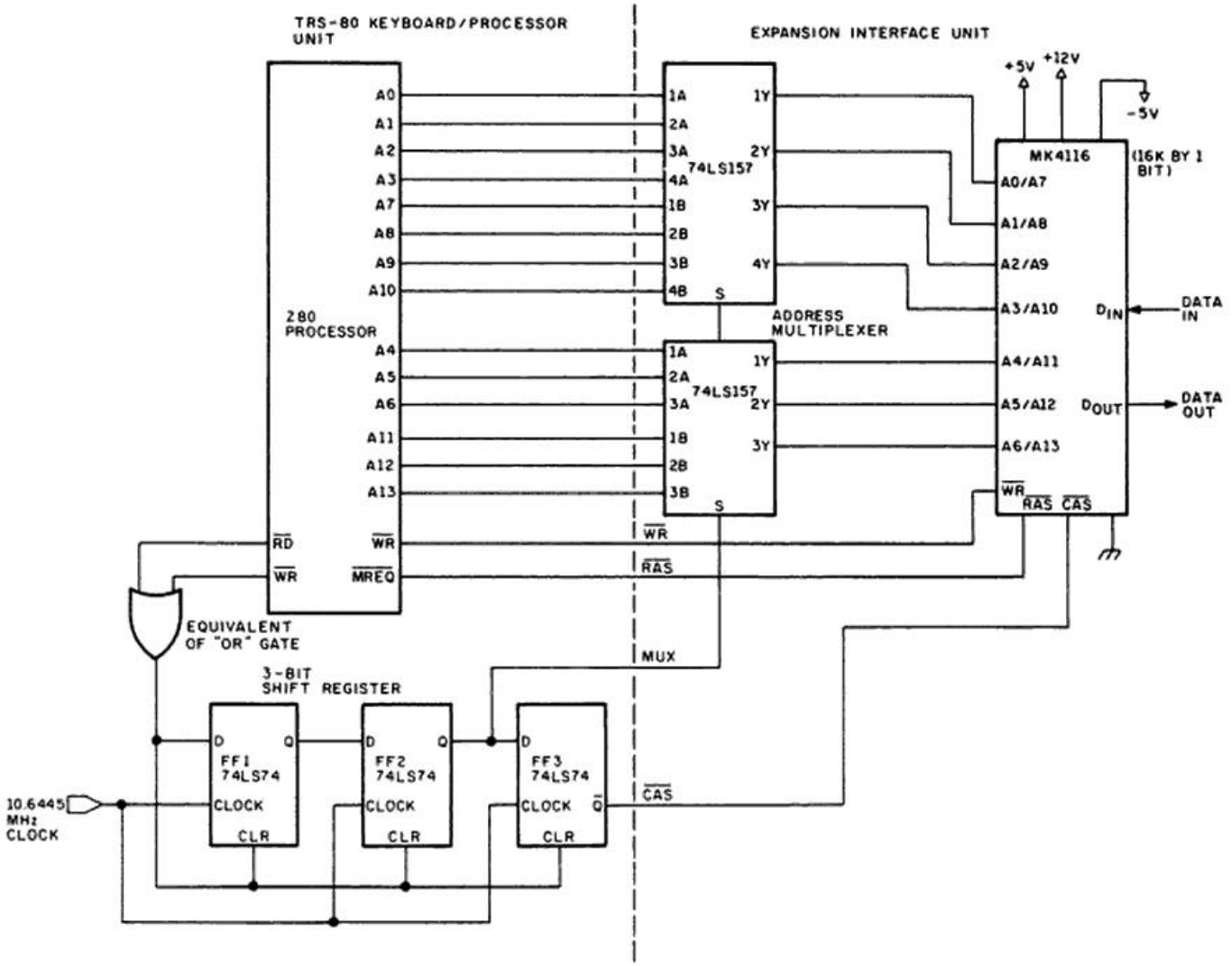


Figure 4: *Simplified schematic diagram of the internal circuitry of the TRS-80 Model I showing the derivation of the memory-refresh logic.*

that are matched to the transformer impedance. Also, because I have designed it around a transformer with specific secondary voltages, the Disk-80 power supply is very efficient and produces relatively little heat. It is designed as a separate circuit board, allowing it to be used with any project requiring power for dynamic memory. It easily powers the full fifty-four-chip Disk-80, including 32 K bytes of memory.

A Centronics-compatible parallel printer port can be optionally built on the power-supply board. A schematic diagram of this port was printed in my previous article, "I/O Expansion for the TRS-80, Part 2: Serial Ports," BYTE, June 1980, page 42.

Finally, techniques of properly distributing power and decoupling transient noise voltages must be addressed. Correct layout of the components in the Disk-80 is critical. The 4116s can generate high-current transients when in operation. Resulting voltage spikes can cause data loss unless the voltage transients are minimized by properly placed decoupling capacitors (a capacitor, usually a ceramic disk type with a value of 0.01 μF to 0.1 μF attached between power and ground).

Some suggestions that are of particular concern in the memory area of the circuit board:

- Decouple the V_{BB} and V_{DD} supply lines on every other chip.
- Distribute larger capacitors around the board to reduce supply-voltage droop.
- Decouple V_{CC} every few chips.
- Keep signal lines short.

Real-Time Clock

To be compatible with TRS-80 hardware, the Disk-80 contains a real-time clock. It provides an interrupt to the Z80 at a rate of 40 times a second (every 25 ms). When the NMI (nonmaskable interrupt) is enabled, the clock-produced interrupts cause the Z80 to transfer control to a specific ROM (read-only memory) address (the interrupt vector). Unless there is a user-supplied routine to be executed, the Z80 simply returns from the interrupt sequence and continues where it left off. Various disk operating systems for the TRS-80 use an interrupt-servicing routine called in this manner to increment a time-of-day clock or event timer.

Five-Inch Floppy-Disk Controller

The Disk-80 uses an LSI (large-

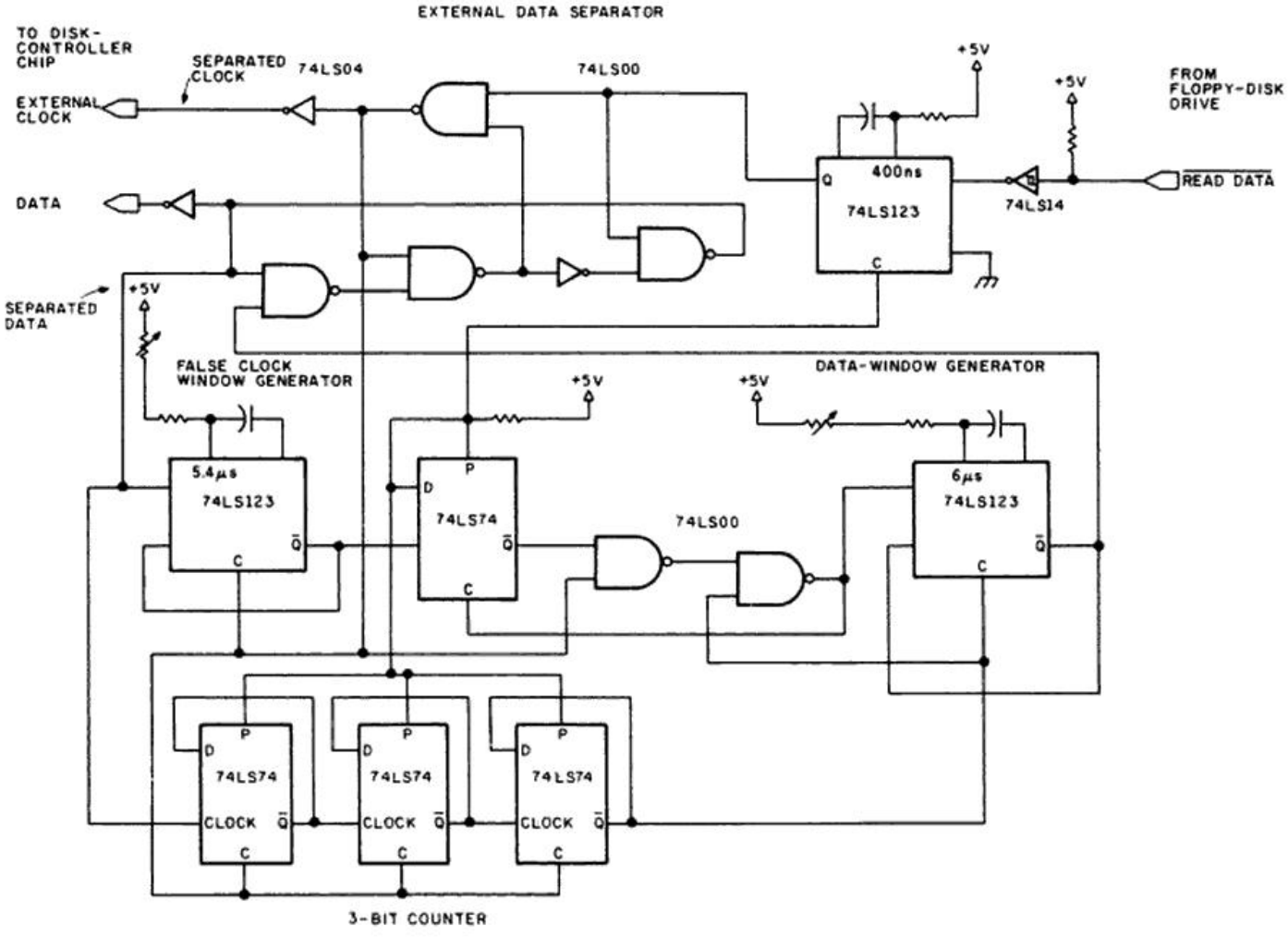


Figure 5: *Simplified schematic diagram of a 5-inch floppy-disk external data separator. The internal data separator of the FD1771 is not recommended for use in such applications. This circuit can be added to any existing TRS-80 Expansion Interface (which does not have an external data separator) to improve performance.*

scale integration) floppy-disk-controller integrated circuit. This one component performs the following functions: encoding, decoding, pattern recognition, serial-to-parallel and parallel-to-serial conversion, CRC- (cyclic redundancy check) character generation, and control of the disk-drive mechanism.

Floppy-disk controllers are available from a number of manufacturers in both single- and double-density versions. Since practically all TRS-80 Model I disk software is stored in single density, the Disk-80 uses a Western Digital FD1771-B01 single-density disk-controller integrated circuit. This component is second-sourced by National Semiconductor as the INS1771D-1.

The standard single-density 5-inch floppy-disk drive stores 110 K unformatted bytes per disk distributed on thirty-five tracks (some drives can use forty or more tracks). Using a soft-sectored format like that used in the IBM 3740 Data-Entry System, each track is divided into 16 sectors storing 128 bytes each. The total amount of data that can be stored on a disk is a function of the disk operating system and the number of tracks per disk supported by the drive itself.

The 5-inch floppy disk is rotated by a DC motor at a speed of 300 rpm. An 8-inch floppy-disk drive contains an AC synchronous motor, which spins the disk at 360 rpm. The bit density of the data is the same, but, due to the differences in rotational speed and disk diameter, the 5-inch drive transfers data at 125 kbps (thousand bits per second) as compared to the 8-inch drive's rate of 250 kbps. The 5-inch drive's lower data rate makes programmed I/O a practical transfer method. Programmed data transfer through specific registers requires less complex hardware than DMA (direct memory access) transfer.

Drive selection is handled by IC32, a 74LS175 4-bit register, and IC33a, a 74LS123 one-shot (monostable multivibrator). Only one drive is selected at a time, and the drive motors are turned off between disk accesses. To address a particular drive, a one-of-four drive code is loaded into IC32 through the memory-mapped register at hexadecimal address 37E0. This action starts a 5-second "motor-on" timer, which is activated whenever a drive is selected. It also activates the Head Load Time (HLT) control line on the FD1771. The software takes into account the 1 second required for the motor to come up to speed and the 80 ms required for head loading. Unless another access is made to this same drive, the motor will shut off

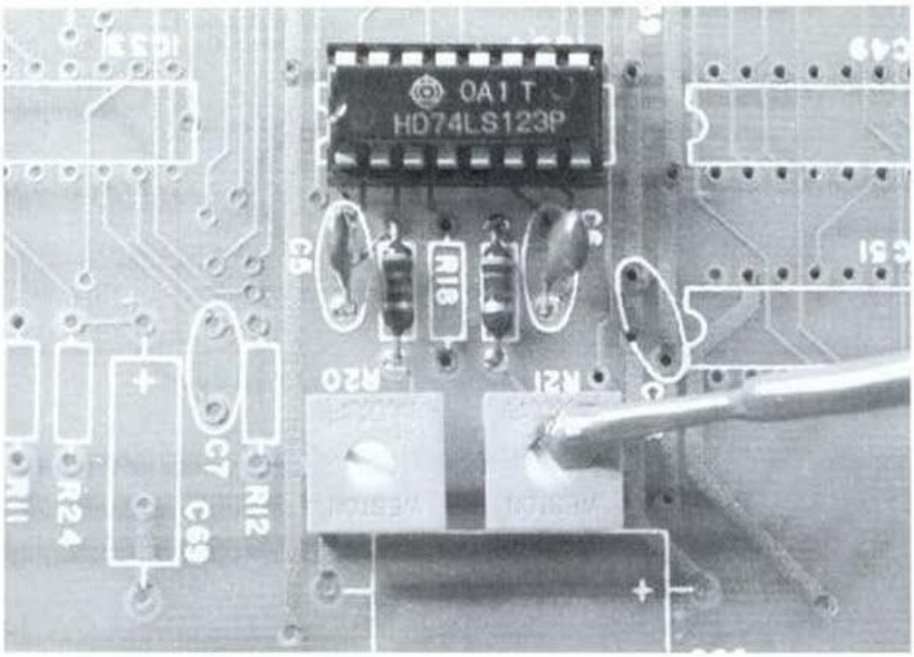


Photo 4: *External-data-separator section of the Disk-80 board. An external data separator is recommended when using the FD1771 disk controller with the 5-inch floppy disks. This circuit (shown in figure 5) can also be added to the Radio Shack TRS-80 Expansion Interface to improve performance. After setting the adjustment potentiometers, use nail polish as shown to lock their positions.*

The following items are available postpaid in the US from:

The MicroMint Inc
917 Midway
Woodmere NY 11598
(516) 374-6793

Item	Ordering Description	Price
Disk-80 blank circuit board (containing no components)	*DSK04*	*$48*
Power-supply/printer-interface blank board	*DSK05*	*$16*
INS1771D-1 disk controller chip with manual	*DSK06*	*$24*
Power supply complete kit: PC board and parts	*DSK07*	*$53*
Disk-80 complete kit: case, power supply, printer port, cable and 32K memory	*DSK03*	*$275*
Disk-80 expansion interface: completely assembled and tested with 32K bytes of memory and printer port	*DSK02*	*$379.95*

All printed-circuit boards are solder-masked and silk-screened and come with assembly instructions.

New York residents please add 7% sales tax.

after 5 seconds.

The Z80 bus structure makes it relatively easy to use a floppy-disk system. All data, commands, and control for the FD1771 are handled through conventional memory-reference instructions. Eight memory-mapped ports (four in and four out) handle all the communication between the Z80 and the FD1771. The range of addresses is hexadecimal 37EC to 37EF.

The Z80 controls the FD1771 through eleven commands, which are divided into four groups:

Type I —Commands that move the read/write head: Restore, Seek, Step, Step-in, Step-out.
Type II —Commands that read and write data: Read sector, Write sector.
Type III—Commands that perform status checking and formatting: Read address, Read track, Write track.
Type IV—Force-interrupt command.

An address map of Type I, II, and III FD1771 register-access functions is shown in table 3 on page 108. The commands and data are communicated to the FD1771 by setting the appropriate logic levels on address lines A0 and A1 (pins 5 and 6 on the FD1771) and strobing either the $\overline{RE}$ (read-enable) or $\overline{WE}$ (write-enable) inputs (pins 4 and 2).

Many disk-control commands require a parameter such as a track or sector address. This data must first be loaded into the appropriate register in the FD1771. To send a track address, for example, the 8-bit track address is loaded into the Z80's accumulator, and a store-accumulator [LD (HL), A] instruction to the track-register port at address 37ED is executed.

Of the FD1771's sixteen control

lines, seven interface directly to the disk drive through drivers and receivers (type-7416 and 74LS14 components). The Write Data line transmits the digitized serial composite data to be written on the floppy disk. The Write Gate line enables the actual writing process. The Index input transmits the pulse from the index-hole photodetector that indicates the beginning of a track, and the Write Protect line tells the controller when a write-protected disk has been inserted into the drive. The Track 00 line is activated when the read/write head is positioned over track 00 (the outermost track) of the disk's surface. The Direction Select line defines the direction in which the head will move when the Step line is pulsed. Each pulse moves the head one track.

Hexadecimal Memory-Mapped	A1	A0	37EC Read Enable	37EC Write Enable
37EC	0	0	Status Register	Command Register
37ED	0	1	Track Register	Track Register
37EF	1	0	Sector Register	Sector Register
37EF	1	1	Data Register	Data Register

Table 3: *Memory-mapped addresses used by the Disk-80 to communicate with the FD1771 or INS1771 floppy-disk-controller integrated circuit. The FD1771 interacts with the Z80 processor by memory-reference instructions, not by DMA. The FD1771 can execute eleven high-level function commands.*

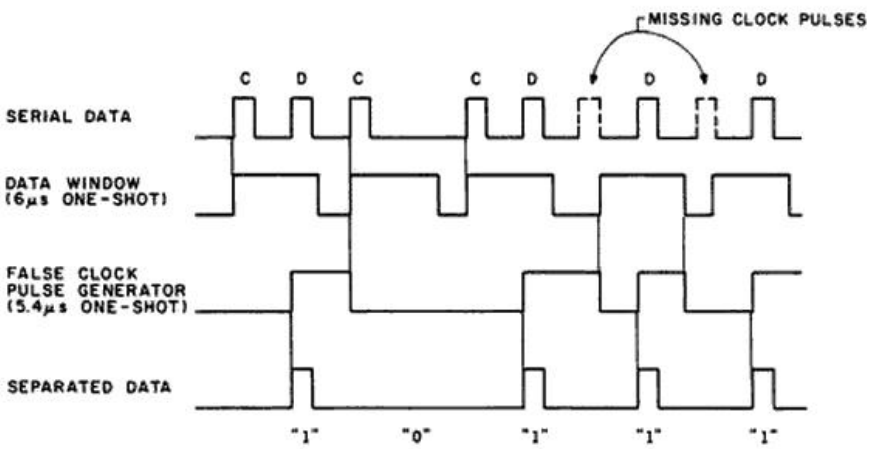


Figure 6: *Timing diagram illustrating the operation of the external data separator shown in figure 5 on page 106. Clock pulses are denoted by the letter C, data pulses by the letter D.*

An External Data Separator Is the Best Insurance

As previously mentioned, the 5-inch floppy-disk drive transfers data at 125 kbps, while an 8-inch drive transfers at 250 kbps. The difference in data rates affects the data separator's timing values as well as the clock rate used by the controller chip. The 5-inch drive requires a 1 MHz clock, while a standard 8-inch drive uses a 2 MHz clock.

Data received from the drive's electronic circuitry is a multiplexed combination of data and clock pulses. The FD1771's internal data separator can separate the data and clock bits, but use of the FD1771's *internal* data separator is *not recommended where high reliability is required*. An *external* data separator must be added to maintain a soft-error rate better than 1 in 10^8.

The internal separator operates from the 1 MHz system clock, which is not synchronous with the clock pulses of the disk data. Due to mechanical variations and other factors, sometimes a bit of data can arrive at the FD1771 at a point in time "outside the data window," that is, when the controller is not expecting it.

[**Editor's Note:** *For a more detailed explanation of the importance of the data window, see "Interface a Floppy-Disk Drive to an 8080A-Based Computer" by John Hoeppner in the May 1980 issue of BYTE, page 72....***RSS**] The nonsynchronous data window's 1 μs (microsecond) resolution can move with respect to a data bit's arrival by enough that the data bit can actually fall outside the data window. This would be interpreted as an error.

To help eliminate what has been a major problem for TRS-80 Model I disk users, *the Disk-80 includes an external data separator*. Neither the Radio Shack TRS-80 Expansion Interface nor the LNW Research System Expansion Board has an external data separator. Figures 5 and 6 illustrate the circuitry and function of the Disk-80's external data separator.

The external data separator places a 400 ns (nanosecond) one-shot on the Read Data line from the drive. This arrangement reduces the Read Data input pulse width from 1.2 μs to 400 ns. When configured for external clock and data separation, the FD1771 requires pulse widths between 300 and 700 ns. The narrower the pulse width, the better the data separator's resolution.

To produce the separator's data window, a 6 μs one-shot is triggered by the leading edge of the clock pulse. Since the time between clock pulses is 8 μs, a data bit is expected within 4 μs after the clock pulse's leading edge. The extra 2 μs allow for shifts in the phase of the data or clock bits. This is all that is required to satisfy any potential timing problems. However, since we also have to be IBM 3740 compatible, more is required.

The IBM 3740 format creates a unique addressing mark by dropping three clock pulses during the address-mark clock pattern. To produce data windows during missing clock-pulse intervals, a false clock pulse is generated with a 5.4 μs one-shot. If the 5.4 μs one-shot times out past the expected instant of the next clock pulse, its own pulse's trailing edge triggers the other (6 μs) one-shot, generating a data window.

A 3-bit counter distinguishes between missing clock pulses and address marks. If the data separator is already in phase, it is constantly reset by the separated clock output. When the separator encounters the address mark, the counter is incremented by a pulse on the separated data line. On the occurrence of the fourth missing

clock pulse, the data window is reset. The separator becomes in-phase again on the next true clock pulse.

Photo 4 shows the location of the false-clock and data-window adjustment potentiometers, R20 and R21, on the Disk-80 circuit board. These are the only user adjustments in the unit. The best method for setting them is to use an oscilloscope and a pulse source. With only IC34 inserted in the board and the +5 V supply on, apply a 50 to 100 kHz clock pulse first to IC34's pin 1. With a scope probe on pin 4, adjust R20 until the one-shot period is 5.4 μs. A similar clock signal is applied to pin 9 of IC34: that section should be set for a period of 6.0 μs by turning R21.

In Conclusion

The TRS-80 Model I may no longer be on the minds of the marketing moguls at Tandy Corporation, but the hundreds of thousands of Model I owners will want to keep using it. Now that you know what is inside an expansion interface, you could build one, if necessary.

Correctly assembling an expansion interface from the circuit diagrams of figure 2 is more than just making all the right connections, however. Layout, decoupling, and power distribution are probably the most critical factors to be considered.

I had to be aware of these same considerations while I was designing the Disk-80, and I had a dilemma when it came time to build the prototype I do for every Circuit Cellar article. The Disk-80 uses fifty-four integrated circuits and 120 resistors and capacitors. The *placement* of these components is as important, in many cases, as the *inclusion* of the component.

To eliminate major trouble-shooting headaches and make it easier for others to construct this interface, I went straight from my schematic diagram to a printed-circuit board, without breadboarding or wire-wrapping. Besides making it easier for me, the result is an elimination of the concern that experimenters would have about the placement of components and decoupling capacitors and the routing of signal lines. The printed-circuit boards are available from The MicroMint, at the address given in the text box on page 107. The schematic diagram of the power supply is not provided here because the correct sequential application of the voltages depends on the use of the exact transformer and components I specified; the circuit may not work with substitute components. If you really want a schematic diagram and a parts list for the power supply, send a stamped, self-addressed envelope to The MicroMint.

I hope that many of you will take this opportunity to build your own expansion interfaces. ■

References

1. *Z80-CPU Technical Manual*. Zilog Inc, 10460 Bubb Rd, Cupertino CA 95014, 1977.
2. Hoeppner, John. "Interface a Floppy-Disk Drive to an 8080A-Based Computer." BYTE, May 1980, page 72.

4116 Pointers

Dear Steve,

Being an ardent do-it-yourselfer, I'm currently in the process of designing a home-brew computer system. Since the 16 K-bit type-4116 dynamic memories are cheap, compact, and use little power, I have decided to use them as my main memory components. Designing the interface is no problem, but I'm all too aware of the 4116's cantankerous nature with respect to circuit-board layout, power-supply fluctuations, and so forth. What should I know about these devices?

Ken McDonald
Yellowknife, NWT, Canada

The most important thing to remember when designing any computer that uses 4116s is that the power-supply voltages have to be turned on and off in sequence. To keep from blowing the 4116 on power-up, the −5 V supply must be turned on before the +5 V and +12 V supplies. On power-down, the −5 V has to remain on while the +5 and +12 are removed.

In lower-current power supplies (such as you will probably use), the sequencing can be accomplished through the time constants of the power supply itself; this technique is used in the TRS-80. By giving the −5 V section a very fast time constant compared to the other two supplies, it appears to come on first. On power-down, the sequence is reversed. Because the −5 V has such a low-current draw on it, it will stay up long after the other voltages have dropped.

*Other than that, use prime components and stay away from surplus devices. For more information on refresh timing signals of dynamic memories, refer to my article in the March 1981 BYTE "Build the Disk-80: Memory Expansion and Floppy-Disk Control," on page 36....***Steve**

Upgrading Kits

Dear Steve,

I would like to increase my TRS-80's memory capacity without spending any more money than necessary, and I don't want to blow it up in the process.

I have a Model III with 16 K bytes of memory, which isn't enough for some of my programming applications. It also limits the length of my Scripsit documents. I would like to add the maximum memory the Model III can hold (48 K bytes). Radio Shack sells 16 K-byte memory kits for $119 plus installation, while various mail-order suppliers advertising in BYTE list similar upgrade kits for around $29.

What is the difference between these memory upgrade kits? Is the installation difficult or within the capabilities of someone who is not a computer technician—like me?

Ralph W Karcher Jr
Broadalbin NY

Theoretically, any 4116-type memory rated for 200 ns access time should work in your TRS-80 Model III. If you carefully disassemble your Model III, you should be able to add them yourself. The sockets are already provided, and no jumpers are required.