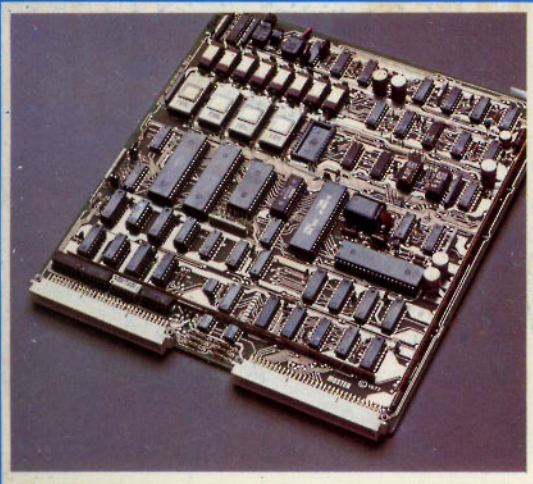
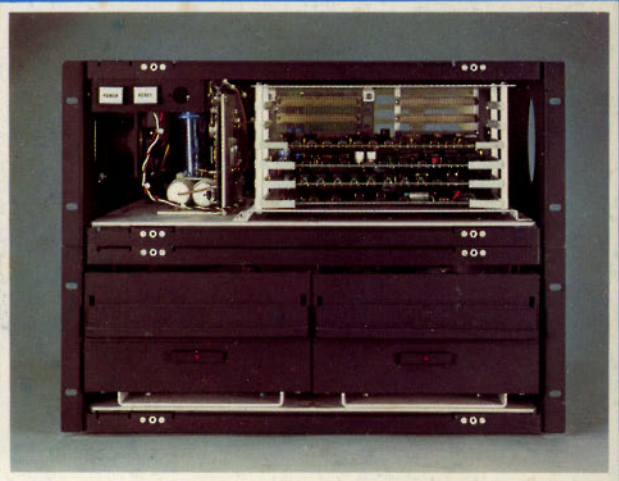
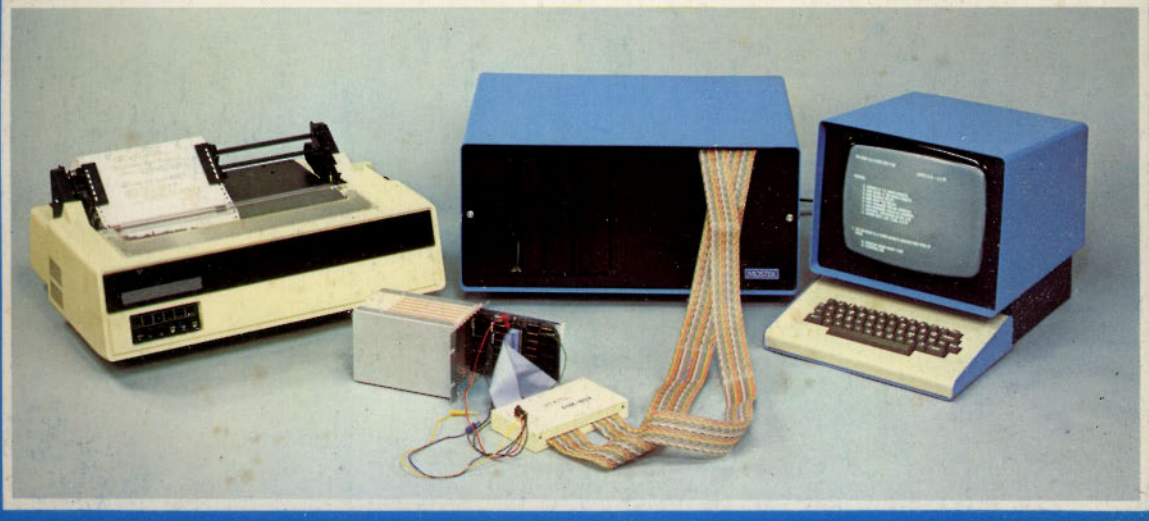
MOSTEK
MICROCOMPUTER SYSTEMS
DATA BOOK

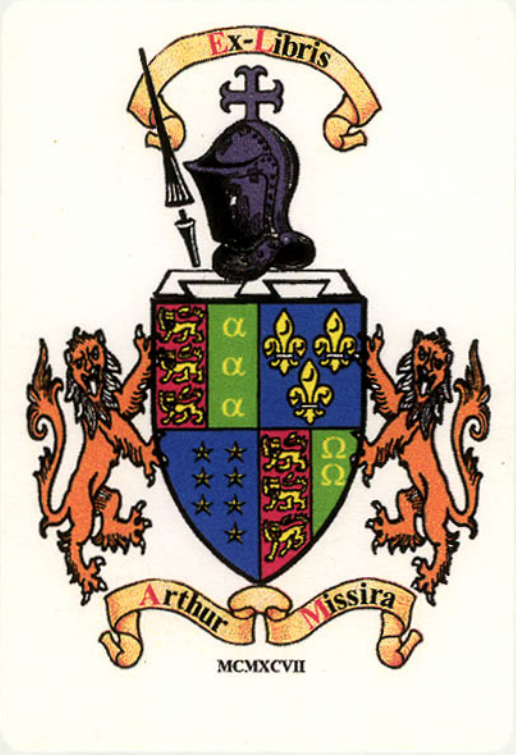






MOSTEK MICROCOMPUTER SYSTEMS DATA BOOK

MICROCOMPUTER SYSTEMS
DATA BOOK



PRINTED IN ENGLAND — SEPTEMBER 1979

Publication Number MK 79781

MICROCOMPUTER SYSTEMS DATA BOOK

Part I
MD (Micro Design) Series
Stand-alone and expandable
OEM Microcomputer Boards

Part II
SD (System Design) Series
Double Eurocard OEM Microcomputers

Part III
SYS-80 FT Floppy-disk based
Development System

Software

(See detailed index page 7.)

MOSTEK 1969-1979
TEN YEARS OF TECHNOLOGY LEADERSHIP

MOSTEK TECHNOLOGY. Technology links the past, present and future of Mostek. Innovations in both circuit and system design, and wafer processing have accounted for our rapid growth and for the strong acceptance of Mostek as a technology leader.

The proven process technology in the semiconductor industry is N-Channel silicon-gate MOS. Mostek is recognised as an important innovator in this process because of the continuing development of new techniques and enhancements which allow significant performance breakthroughs in our products. Competing technologies have not yet been able to approach either the performance or producibility of N-Channel MOS. Therefore, it appears that NMOS silicon-gate will continue to lead industry developments for several years to come.

MICROCOMPUTER COMPONENTS. Mostek's microcomputer products cover the full spectrum of microprocessor applications worldwide.

MOSTEK'S Z80 is the most powerful 8-bit microcomputer available. It is software compatible with the 8080A yet has some significant system advantages — an increased instruction set, reduced dynamic memory interfacing costs, reduced I/O costs and reduced support circuitry costs.

MOSTEK'S 3870 FAMILY of single-chip microcomputers allow system flexibility and expansion while retaining the design and economic advantages of single-chip construction. Software compatible with the F8, Mostek's 3870 family is the answer to a wide range of low-cost microcomputer applications.

MICROCOMPUTER SYSTEMS. Mostek's microcomputer line is supported by a wide array of development aids. These include software development boards that may be used as software development aids or as stand-alone microcomputers. Add-on memory boards, application interface modules, and emulators assist in system design, debugging and field testing. Now Mostek brings this experience to the Original Equipment Manufacturer.

Mostek's microcomputer line includes Mostek's MD Series™ of OEM microcomputer boards. The MD Series features both stand-alone boards (designated MD) and expandable boards (designated MDX) that are STD-Z80 BUS compatible. These powerful Z80-based boards are simple and economical to use.

The System Design Series (SD Series™) of OEM microcomputer boards offers powerful features and versatility to the OEM. Utilizing the Mostek Z80 and Mostek's industry-standard memories, the SD Series enables the user to construct high-performance, memory-intensive systems for a wide range of applications.

Also included in the SD Series is a broad line of peripherals, software and development system support to speed and ease the design process. Mostek's floppy disk development system is built using the same cards shown here; this means your application can be prototyped on the system and later transferred to your own custom environment without software integration problems. Also, our AIM-80 application interface module and DDT-80 debug ROM will plug into any SD or MD System for quick testing of system hardware or software.

The products in this book are only the beginning of a full family of OEM products. New and innovative products are in design that will give further flexibility to the system designer.

MOSTEK MICROCOMPUTER SYSTEM DATA BOOK

INDEX

MD Series

MOSTEK'S MD (MICRO DESIGN) SERIES Z80 MICROCOMPUTERS

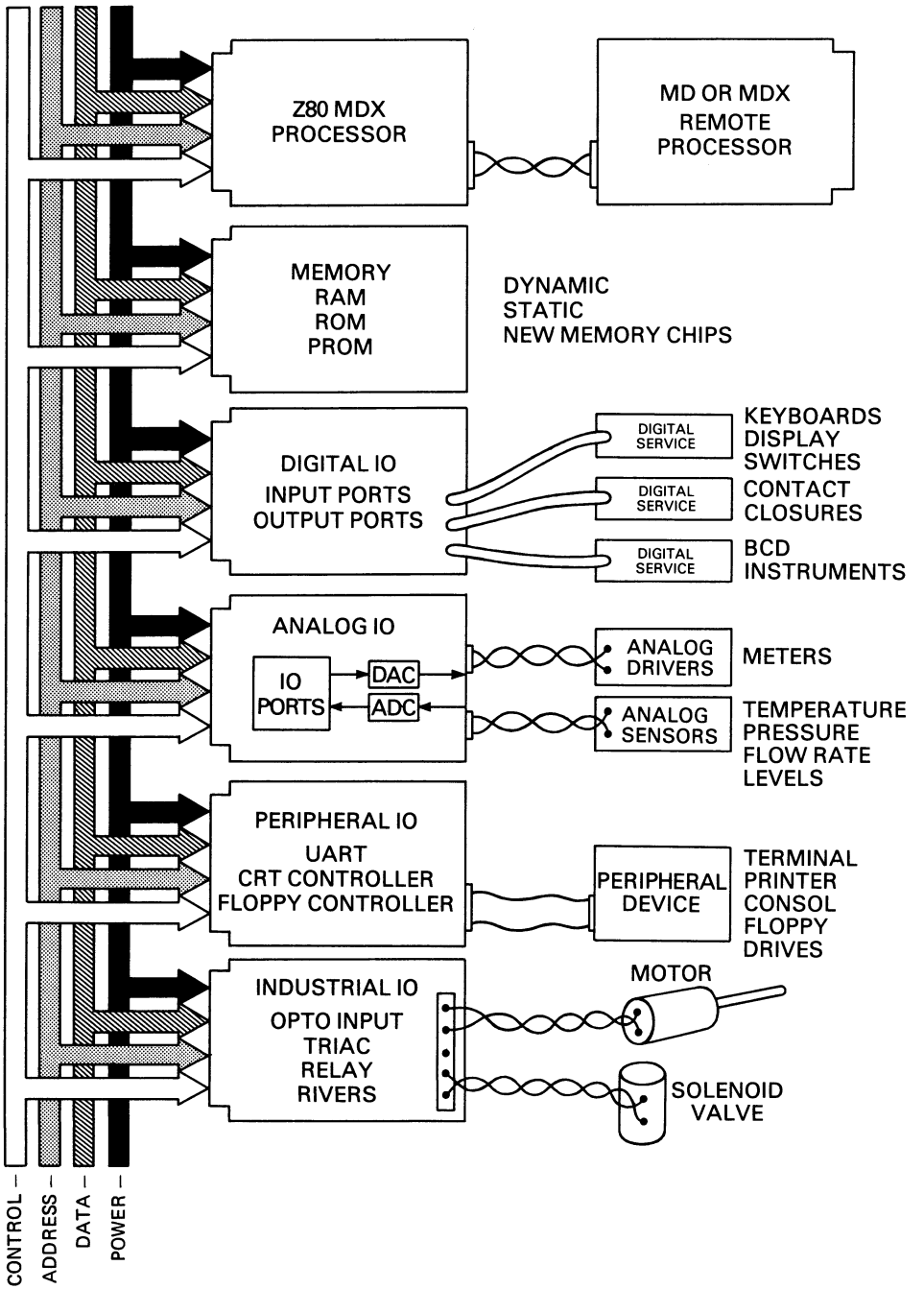
GENERAL DESCRIPTION

In many OEM applications, such as process control, test equipment, industrial automation, energy management systems, etc..., the existing Single Board Microcomputers are insufficiently modular for the system design requirements.
Having recognised this need for a more cost effective approach, Mostek introduces now the MD Series of OEM Microcomputer modules.
To the design engineer, the Mostek MD Series provides Z80 power, well designed OEM boards, sophisticated software and extensive field support at an attractive price.
Each module operates from a Standard (STD) position independent bus, this STD BUS concept (developed jointly by Prolog and Mostek and rapidly gaining wide acceptance) uses a motherboard interconnect system.
The MD Series modules are thus fully expandable and can be organised in unrestricted combinations (any card works in any slot) to form a particularly cost effective microcomputer system, closely matching the application requirements.
The MD Series comprises the following bus expandable modules (all in either 2.5 or 4.0 MHz versions) on 114 x 165 mm (4.5 x 6.5 in.) cards:

MDX-CPU1	— Z80 CPU, 4K x 8 PROM, 256 x 8 RAM
MDX-DRAM	— 8K, 16K or 32K x 8 add-on dynamic RAM module
MDX-EPROM/UART	— 10K x 8 universal PROM plus UART add-on module
MDX-PIO	— Fully buffered parallel I/O controller
MDX-SIO	— Multiprotocol asynchronous or synchronous Serial I/O interface
MDX-DEBUG	— 10K bytes of Firmware (DDT-80 ASMB-80) with Edit, Debug and Assemble capability to generate and Debug STD BUS programs
MDX-SST	— Single step Debug module
MDX-UMC	— Universal Memory Card
MDX-EPROM	— EPROM Memory Extension
MDX-SRAM	— 4K, 8K or 16K x 8 add-on static RAM Module
MDX-A/D	— 8 bit A/D Converter module

The MD Series also includes a fully functional, stand-alone microcomputer MD-SBC 1 on a single 114 x 165 mm (4.5 x 6.5 in) board.
Additional modules are being designed by Mostek to be made available shortly, see page 41.
Of particular interest to the OEM design engineer is the Mostek MDX-PROTO comprising the CPU, DRAM and Debug modules together with the 8-slot card cage, STD bus motherboard and all accessories to form a ready to use prototyping kit.

MD SERIES SYSTEMS CONCEPT



PRELIMINARY

MOSTEK®

MD SERIES MICROCOMPUTER MODULES

Z80 Single Board Computer (MD-SBC1)

FEATURES

- ☐ Z80 Microprocessor
- ☐ 2K byte RAM capacity with 1K included
- ☐ Sockets for 8K bytes 2716 EPROM
- ☐ Crystal Clock - 2.5 MHz
- ☐ Three TTL buffered 8-bit OUTPUT ports
- ☐ Two TTL buffered 8-bit INPUT ports
- ☐ Two Interrupt Inputs
- ☐ Single +5 volt power supply

DESCRIPTION

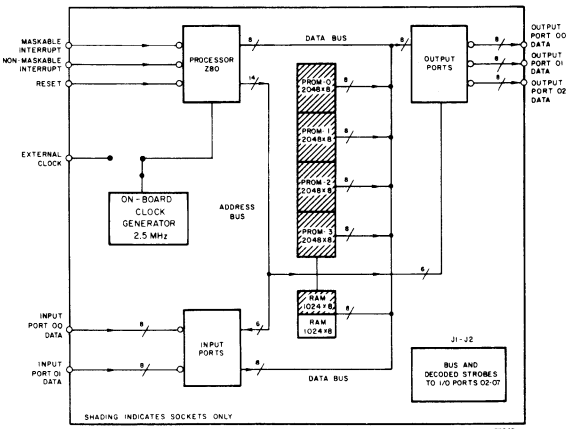
The MD-SBC1 is a complete Z80 based microcomputer on $4\frac{1}{2}$ in. by $6\frac{1}{2}$ in. circuit module. All I/O is fully TTL buffered and is brought to a 56 pin edge connector.

The smaller card size and the single power supply makes the MD-SBC1 easier to package and easier to use than most other modules. While the module size is small no compromises have been made in computing power due to increasing MOS-LSI densities and the use of the Z80 microcomputer. The 40 buffered TTL I/O lines and the 8K bytes of EPROM provide the capability to solve many control problems encountered by the OEM microcomputer user. The expandable MD Series (MDX) has the same form factor allowing easy expansion to a multi-board system with increased capability.

Figure 1 is a block diagram of the MD-SBC1. The basic module comes with 1K bytes of RAM expandable to 2K bytes by the addition of two 2114 type RAMs. Four 2716 sockets are provided for up to 8K bytes of EPROM, and are decoded in 2K blocks starting at address zero. The output ports are 74LS244 latches which are brought to the card cage connector. The input ports are 74LS240 Octal Buffers with 4.7K OHM pull-up resistors on the inputs. These input lines are also brought to the edge connector. The Z80-CPU is driven by a crystal clock at 2.5MHz (400nsec T-State).

Both the NMI and INT interrupt inputs to the Z80-CPU are terminated with 4.7K Ohm pull ups and brought to the card edge connector. An external clock can be used by changing strapping options on the board. Power on reset circuitry is included on the CPU's $\overline{\text{RESET}}$ input. Provision is made to expand the I/O capability through the use of on-board connectors.

MD-SBC1 BLOCK DIAGRAM
Figure 1



ELECTRICAL SPECIFICATIONS

WORD SIZE

INSTRUCTION 8, 16, 24 or 32 bits
DATA 8 bits

CYCLE TIME

T-STATE = 400nSec, fastest instruction is 1.6 microsecond.

MEMORY ADDRESSING

EPROM NUMBER	HEX ADDRESS
0	0000-07FF
1	0800-0FFF
2	1000-17FF
3	1800-1FFF

RAM NUMBER	HEX ADDRESS
STANDARD	2000-23FF
OPTIONAL	2400-27FF

MEMORY CAPACITY

8 K bytes of 2716 memory (none included)
2 K bytes of 2114 memory (1 K bytes included)

MEMORY SPEED REQUIRED

Memory	Access Time Required	Cycle Time Required
2716*	450nSec	450nSec
2114	450nSec	450nSec

* Single 5 volt type required

I/O ADDRESSING AND CAPACITY

PORT TYPE	HEX ADDRESS	DATA CAPACITY
Input	00 and 01	16 lines
Output	00, 01, 02	24 lines

INTERRUPTS

Two active low: $\overline{NMI}$ and $\overline{INT}$. See Z80-CPU (MK 3880) Technical Manual for a full description of Z80 interrupts.

I/O INTERFACES

Inputs - One 74LS load plus a 4.7K Ohm pull up resistor
Outputs - I_{OH} = 15mA at V_{OH} = 2.4 volts
I_{OL} = 12mA at V_{OL} = 0.4 volts

SYSTEM CLOCK

	MIN	MAX
MD-SBC1	250KHz	2.5MHz

POWER SUPPLY REQUIREMENTS

+5 volts ±5% at 1.2A max (fully loaded)
(100mA per RAM, 100mA per EPROM)

OPERATING TEMPERATURE RANGE

0 °C to +50 °C

MECHANICAL SPECIFICATIONS

CARD DIMENSIONS

4.5 in. (11.43cm) high by 6.50 in. (16.51cm) long
048 in. (1.22cm) maximum profile thickness
0.062 in. (0.16cm) printed circuit board thickness

CONNECTORS

FUNCTION	CONFIGURATION	MATING CONNECTOR
Paralleled I/O	56 pin (28 position) 0.125 in centers	Printed Circuit VIKING VH-28/1CK5 Wire Wrap VIKING 3VH-28/1CMD1 Solder Lug VIKING VH-28/1CL5

ORDERING INFORMATION

DESIGNATOR	DESCRIPTION	PART NO.
MD-SBC1	Complete Z80 Single Board Computer with Operation Manual less EPROMs and mating connector	MK77851
	MD-SBC1 Operation Manual only	MK79609
MDX-PROTO Data Sheet	MD Series prototyping package	MK79605
SYS-80F Data Sheet	Disk based development system	MK78575
AIM-80E Data Sheet	Z80 In-Circuit Emulation Module	MK78571

Z80 Central Processor Module (MDX-CPU1)

FEATURES

- ☐ Z80 CPU
- ☐ 4K x 8 EPROM (2-1716's)
- ☐ 256 x 8 Static RAM (compatible with DDT-80 debugger)
- ☐ Flexible Memory decoding for EPROM and RAM
- ☐ A80-CTC with four timer channels
- ☐ Restart to 0000H or E000H
- ☐ Debug compatible for single step in DDT-80
- ☐ 4MHz version available
- ☐ +5V only
- ☐ Fully buffered signals for system expandability
- ☐ STD BUS™ compatible

DESCRIPTION

The MOSTEK MDX-CPU1 is the heart of an MD Series Z80 system. Based on the powerful Z80 micro-processor, the MDX-CPU1 can be used with great versatility in an OEM microcomputer system application. This is done simply by inserting custom ROM or EPROM memories into the sockets provided on the board and configuring them virtually anywhere within the Z80 memory map.

On-board memory is provided in the form of 4K of EPROM (2-2716's) and 256 bytes of scratchpad RAM as pictured in the block diagram. In addition, an MK3882 Counter Timer Circuit is included on the MDX-CPU1 to provide counting and timing functions for the Z80. Either 2716 EPROM can be located at any 2K boundary within any given 16K block in the Z80 memory map via a jumper arrangement.

The MDX-CPU1 can be used in conjunction with the MDX-DEBUG and MDX-DRAM modules to utilize DDT-80 and ASMB-80 in system development. This is accomplished by strapping the scratchpad RAM to reside at location FF00 so that it will act as the Operating System RAM for DDT-80.

The MDX-CPU1 is also available in 4MHz version (MDX-CPU1-4). In this version, one wait cycle is automatically inserted each time on-board memory is accessed by a read or write cycle. This is necessary to make the access times of the 2716 PROMs and the 3539 scratchpad RAM compatible with the MK3880-4 MHz Z80-CPU.

WORD SIZE

Instruction: 8, 16, 24, or 32 bits
Data: 8 bits

CYCLE TIME

Clock period or T state = 0.4 microsecond @ 2.5MHz
0.25 microsecond @ 4.00 MHz
Instructions require from 4 to 23 T states.

MEMORY ADDRESSING

On-Board EPROM: jumper selectable for any 2K boundary within a 16K block of Z80 memory map.
On-Board RAM: FF00-FFFF.

MEMORY CAPACITY

On-Board EPROM - 4K bytes (sockets only).
On-Board RAM - 256 bytes.
Off-board Expansion - Up to 65,536 byte, with user-specified combinations of RAM, ROM, PROM.

MEMORY SPEED REQUIRED

MEMORY	ACCESS TIME	CYCLE TIME
2716*	450nS	450nS

* Single 5 volt type required

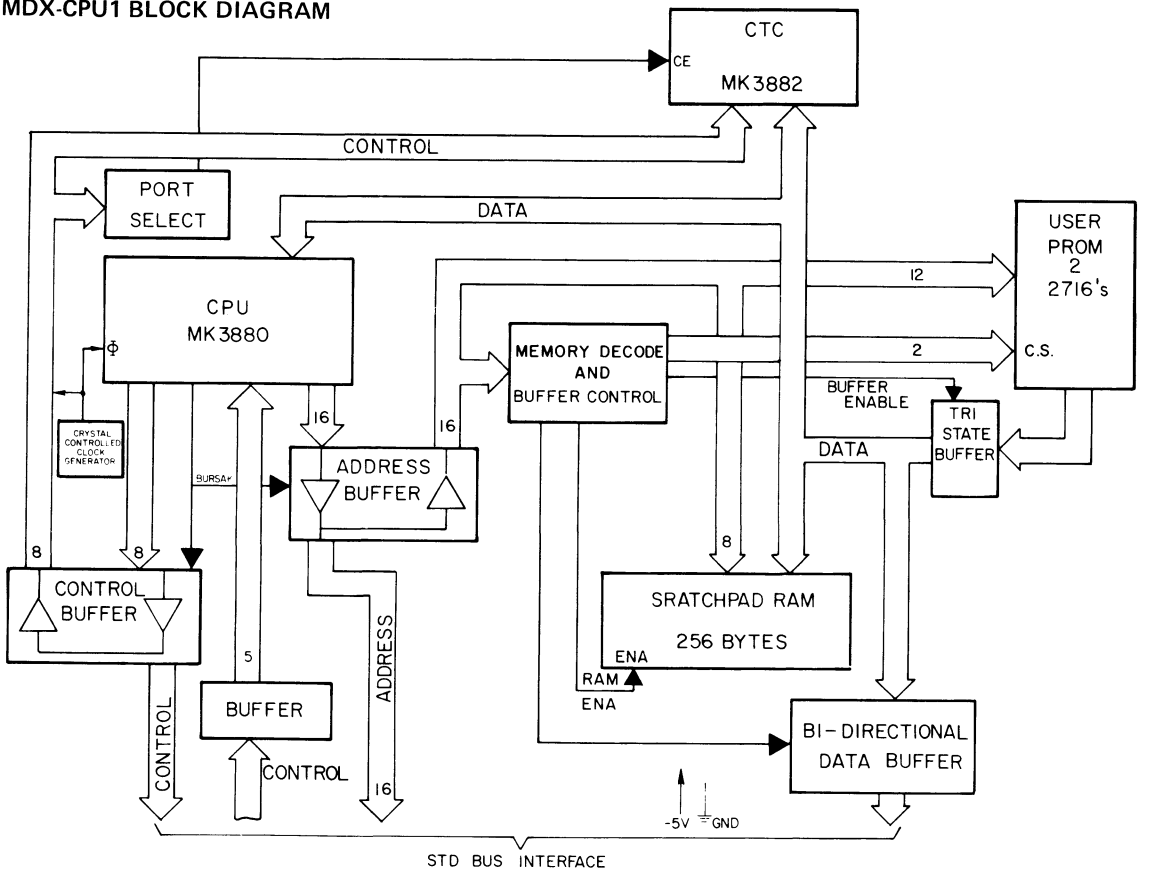
I/O ADDRESSING On-Board Programmable Timer

PORT ADDRESS (HEX)	MK 3882 CHANNEL
7C	0
7D	1
7E	2
7H	3

I/O CAPACITY

Up to 252 port addresses can be decoded off board. Four port addresses are on board. 252 + 4 = 256 total I/O ports.

MDX-CPU1 BLOCK DIAGRAM



INTERRUPTS

Multi-level with three vectoring mode (Mode 0, 1, 2). Interrupt requests may originate from user-specified I/O or from the on-board MK3882 CTC.

PARALLEL BUS INTERFACE - STD BUS COMPATIBLE

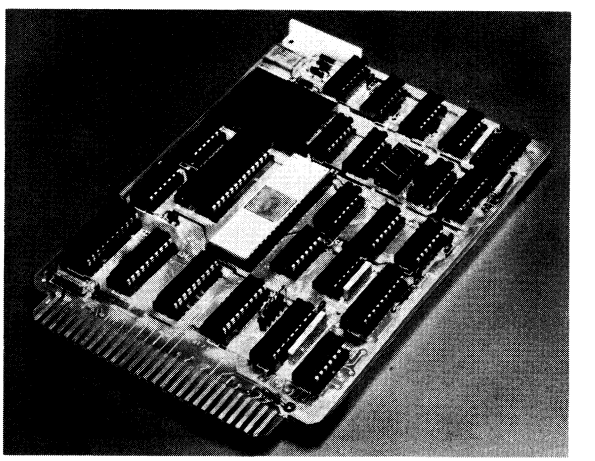
Inputs	One 74LS load max
Bus Outputs	I_{OH} = .3mA min at 2.4 volts
	I_{OL} = 12 mA min at 0.4 volts

SYSTEM CLOCK

	MIN	MAX
MDX-CPU1	500 KHz	2,500 MHz
MDX-CPU-4	500 KHz	4,000 MHz

POWER SUPPLY REQUIREMENTS

5V ± 5% at 1.1A maximum



Dynamic Ram Module (MDX-DRAM)

FEATURES

- ☐ Three memory sizes
  - 8K x 8 (MDX-DRAM8)
  - 16K x 8 (MDX-DRAM16)
  - 32K x 8 (MDX-DRAM32)
- ☐ Selectable addressing on 4K boundaries.
- ☐ 4MHz version available (MDX-DRAM-4)
- ☐ STD BUS compatible

DESCRIPTION

The MDX-DRAM is designed to be a RAM memory expansion board for the MOSTEK MD SERIES of Z80 based microcomputers. It is available in three memory capacities: 8K bytes (MDX-DRAM8), 16K bytes (MDX-DRAM16), and 32K bytes (MDX-DRAM32). Additionally, the MDX-DRAM16 and the MDX-DRAM32 are available in a 4MHz version. Thus, the designer can choose from the various options to tailor his add-on dynamic RAM directly to his system requirements.

The MDX-DRAM8 is designed using MOSTEK's MK4108 8, 192-bit dynamic RAM. The MDX-DRAM16 and MDX-DRAM32 utilize high-performance MK4116, 16,384-bit dynamic RAMs which allow 4MHz versions of these boards to be offered. No wait-state insertion circuitry is required on any of the RAM cards.

Address selection is provided on all MDX-DRAM cards for positioning the 8K, 16K, or 32K of memory to start on any 4K boundary.

ELECTRICAL SPECIFICATIONS

WORD SIZE

8 bits

MEMORY SIZE

MDX-DRAM8 - 8,192 bytes
MDX-DRAM16 - 16,384 bytes
MDX-DRAM32 - 32,768 bytes

ACCESS TIMES

SYSTEM CLOCK		MEMORY ACCESS TIMES	MEMORY CYCLE TIMES
MDX-DRAM	2.5 MHz	350ns max.	465ns min.
MDX-DRAM-4	4.0 MHz	200ns max.	325ns min.

ADDRESS SELECTION

Selection of 8K, 16K, or 32K contiguous memory blocks to reside at any 4K boundary

PARALLEL BUS INTERFACE-STD BUS COMPATIBLE

Inputs	One 74LS load max
Bus Outputs	I_{OH} = -3mA min. at 2.4 volts
	I_{OL} = 12mA min. at 0.4 volts

SYSTEM CLOCK

	Min	Max
MDX-DRAM	1.25MHz	2.5MHz
MDX-DRAM-4	1.25MHz	4.0MHz

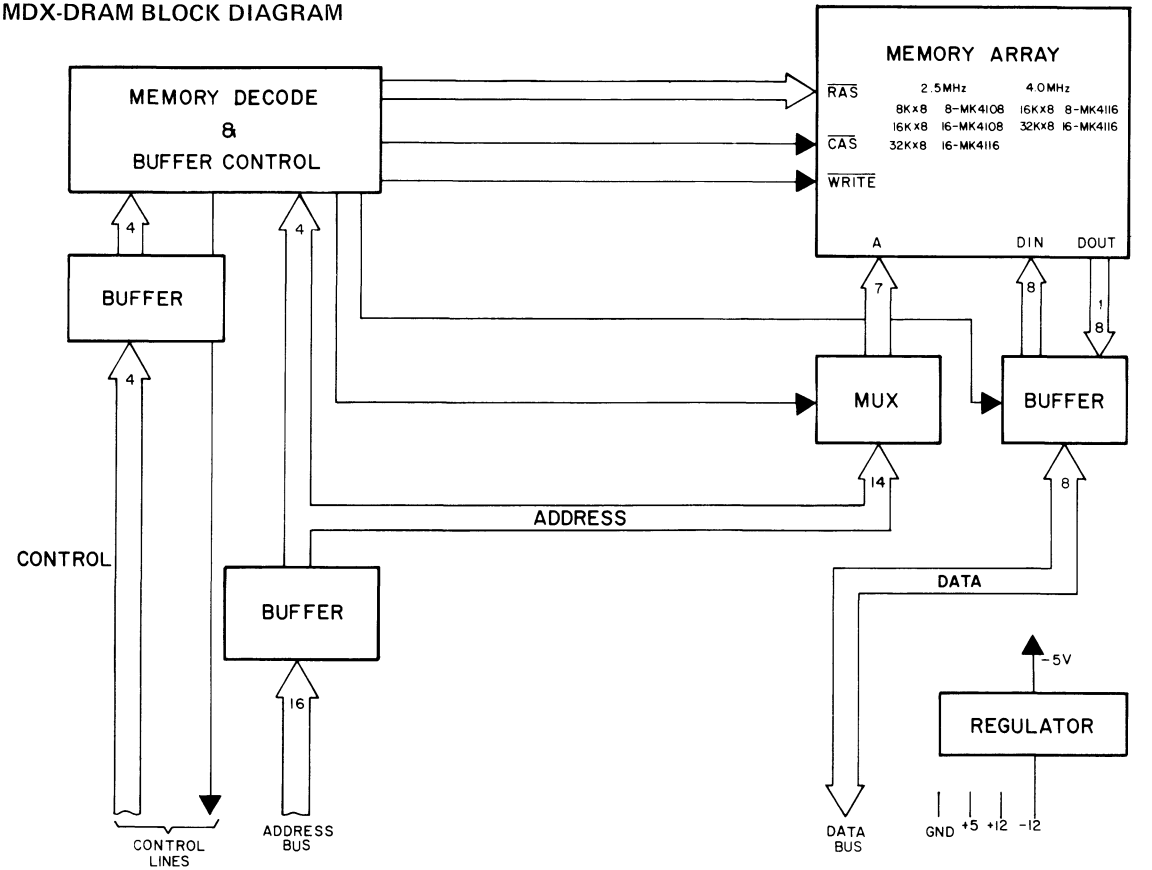
POWER SUPPLY REQUIREMENTS

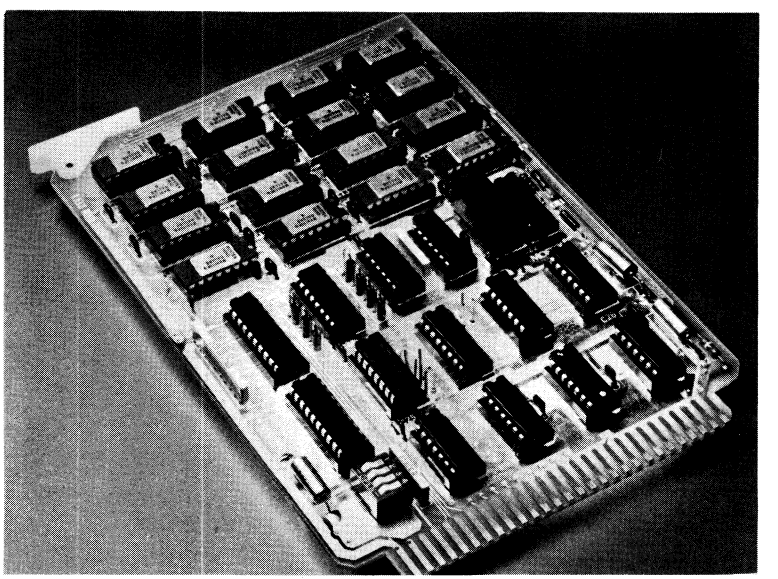
+5V ±5% at 0.6A max.
+12V ±5% at 0.25A max.
-12V ±5% at 0.03A max.

OPERATING TEMPERATURE

0 °C to 50 °C

MDX-DRAM BLOCK DIAGRAM





EPROM/UART Module (MDX-EPROM/UART)

FEATURES

- ☐ 10K x 8 EPROM/ROM (2716's not included)
- ☐ Serial I/O channel
  - RS - 232 and 20 mA interface
  - Reader step control for Teletypes
  - Baud rate generator 110-19200 Baud
- ☐ 4MHz version available (MDX-EPROM/UART-4)
- ☐ STD BUS compatible.

DESCRIPTION

The MDX-EPROM/UART is one of MOSTEK's complete line of STD BUS compatible Z80 microcomputer modules.

Designed as a universal EPROM add-on module for the STD BUS, the MDX-EPROM/UART provides the system designer with sockets to contain up to 10K x 8 of EPROM memory (5-2716's) as shown in the Block Diagram.

The EPROM memories can be positioned to start on any 2K boundary within a 16K block of memory via a strapping option provided on the MDX-EPROM/UART.

Included on-board the MDX-EPROM/UART is a fully buffered asynchronous I/O port with a Teletype reader step control. A full duplex UART is used to receive and transmit data at the serial port. Operation and UART options are under software control. Once the unit has been programmed, no further changes are necessary unless there is a modification of the serial data format. Features of the UART include:

- Full duplex operation
- Start bit verification
- Data word size variable from 5 to 8 bits
- One or two stop bit selection
- Odd, even, or no parity option
- One word buffering on both transmit and receive

The MDX-EPROM/UART is also available in a 4MHz version. Circuitry is provided to force one wait state each time on board EPROMs or the UART are accessed.

WORD SIZE

8 bits for PROM
5 to 8 bits for Serial I/O

MEMORY ADDRESSING

ROM/EPROM
2K blocks jumper selectable for any 2K boundary within a given 16K boundary of Z80 memory map.

MEMORY CAPACITY

10K bytes of 2716 memory.
(2716's not included)

MEMORY SPEED REQUIRED

MEMORY	ACCESS TIME	CYCLE TIME
2716*	450ns	450ns

* Single 5 Volt type required

I/O ADDRESSING

On-board Serial I/O Port
- Control Port DDH
- Data Port DCH

Modem and Reader Step Control DEH

I/O TRANSFER RATE

X16 BAUD RATE CLOCK	BAUD RATE (Hz)
1760	110
4800	300
9600	600
19200	1200
38400	2400
76800	4800
153600	9600
317200	19200

SERIAL COMMUNICATIONS CHARACTERISTICS

Asynchronous
- Full duplex operation
- Start bit verification
- Data word size variable from 5 to 8 bits.
- One or two stop bits
- Odd, even, or not parity
- One word buffering on both transmit and receive.

SERIAL COMMUNICATIONS INTERFACE

SIGNAL	BUFFERED FOR 20mA Current Loop	RS-232
Transmitted data	Output	Output
Received data	Input	Input
Reader Step Relay (RSR)	Output (20mA)	
Data Terminal Ready (DTR)		Input
Request to Send (RTS)		Input
Carrier Detect (CDET)		Output
Clear to Send (CTS)		Output
Data Set Ready (SDR))		Output

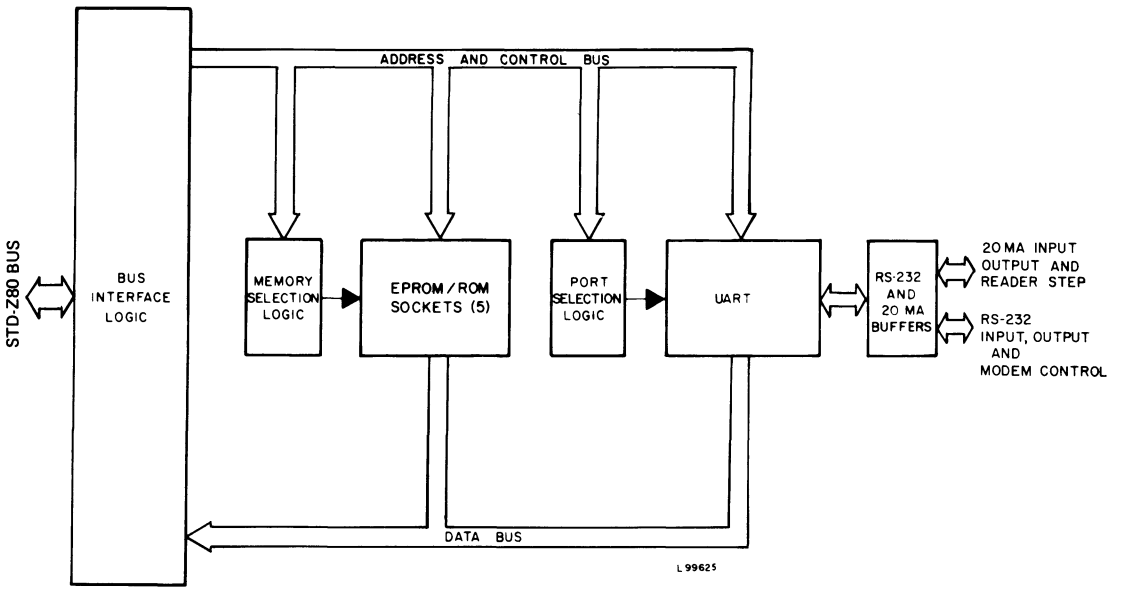
PARALLEL BUS INTERFACE - STD BUS COMPATIBLE

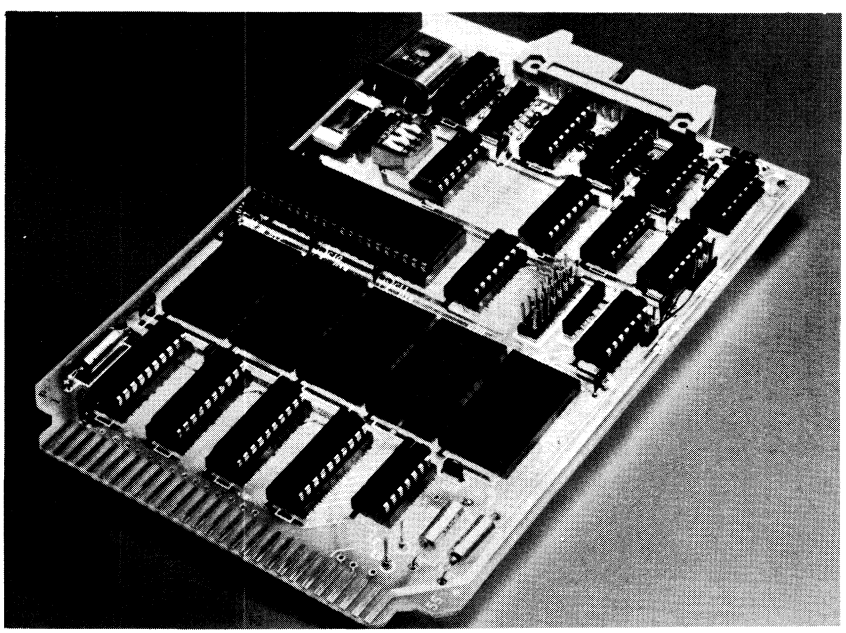
Inputs	One 74LS Load Max
Bus Outputs	IOH = -3mA min at 2.4 Volts
	IOL = 12mA min at 0.4 Volts

SYSTEM CLOCK

	MIN.	MAX.
MDX-EPROM/UART	250 KHz	2.5 MHz
MDX-EPROM/UART-4	250 KHz	4.0 MHz

MDX-EPROM/UART BLOCK DIAGRAM





Programmable Input/Output Unit (MDX-PIO)

FEATURES:

- ☐ Four 8-bit I/O ports with 2 handshake lines per port
- ☐ All I/O lines fully buffered
- ☐ I/O lines TTL compatible with provision for termination resistor networks
- ☐ Jumper options for inverted or non-inverted handshake
- ☐ Two 8-bit ports capable of true bidirectional I/O
- ☐ Programmable In only, Out only, or Bidirectional
- ☐ Output data buffers selectable to provide inverted or non-inverted drive capability
- ☐ Interrupt driven programmability
- ☐ Address strap selectable
- ☐ STD BUS Compatible
- ☐ 4 MHz Option
- ☐ Fully buffered for MD Series expandability

DESCRIPTION

The parallel I/O controller (MDX-PIO) ia a highly versatile unit designed to provide a variety of methods for inputting and outputting data from the MD series microcomputer system. The system is designed around two Mostek MK3881 Z80-PIO parallel I/O controllers which give four independent 8-bit I/O ports with two handshake (data transfer) control lines per port. All I/O lines are buffered and have provisions for termination resistors on board. All port lines are brought to two 26 pin connectors; two ports per connector. Logically, each port pair A & B (connector) look similar (depending upon jumper options) to the on-card PIO devices.

Figure 1 illustrates in block diagram form the major functional elements of port pair A and B of PIO 1. These elements can be defined as the resistor termination networks, data buffers, port configuration control, MK3881 PIO, and address decode and data bus buffers. Input and output from the ports are provided through J1, a 26 pin connector. This connector provides data paths for the two ports and their respective handshake signals.

One 14-pin socket is provided per port for resistor dual inline packages so that terminations may be placed on the data lines. A parallel termination is provided for each 8-bit port data line plus the input strobe (STB) handshake line. The MDX-PIO is normally shipped with four 1K pullup terminators. In addition to the parallel termination resistors, the ready (RDY) handshake output line is series terminated with a 47 Ω resistor. This is used to damp and reduce reflections on the output line.

Port A and B data bus lines are buffered using quadruple non-inverting transceivers. The buffers can be configured using port configuration jumpers to provide fixed Input, fixed Output or Bidirectional (Port A only) signals. Further the transceivers are configured such that port direction can be selected in 4-bit sections. The transceivers are mounted in sockets so that they can be easily replaced with their complements in order to achieve a polarity change if desired.

The handshake lines are also fully buffered. The port configuration control provides jumper options to independently control the polarity or "sense" of each handshake line so as to further ease the interfacing between the MDX-PIO and peripheral devices.

The MK3881, PIO parallel I/O controller is the heart of the module. This circuit is a fully programmable two port device which provides a wide range of configuration options. Any one of four distinct modes of operation can be selected for a port. They are byte output, byte input, byte bidirectional (Port A only) and bit control mode. The PIO also automatically generates all handshaking signals in all the above modes.

The PIO permits total interrupt control so that full usage of the MDX-CPU interrupt capabilities can be utilized during I/O transfers. Also the PIO can be programmed to interrupt the CPU on the occurrence of a specified status condition in a specific peripheral device. The PIO circuit will provide vectored interrupts and maintain the daisy chain priority interrupt logic compatible with the STD BUS.

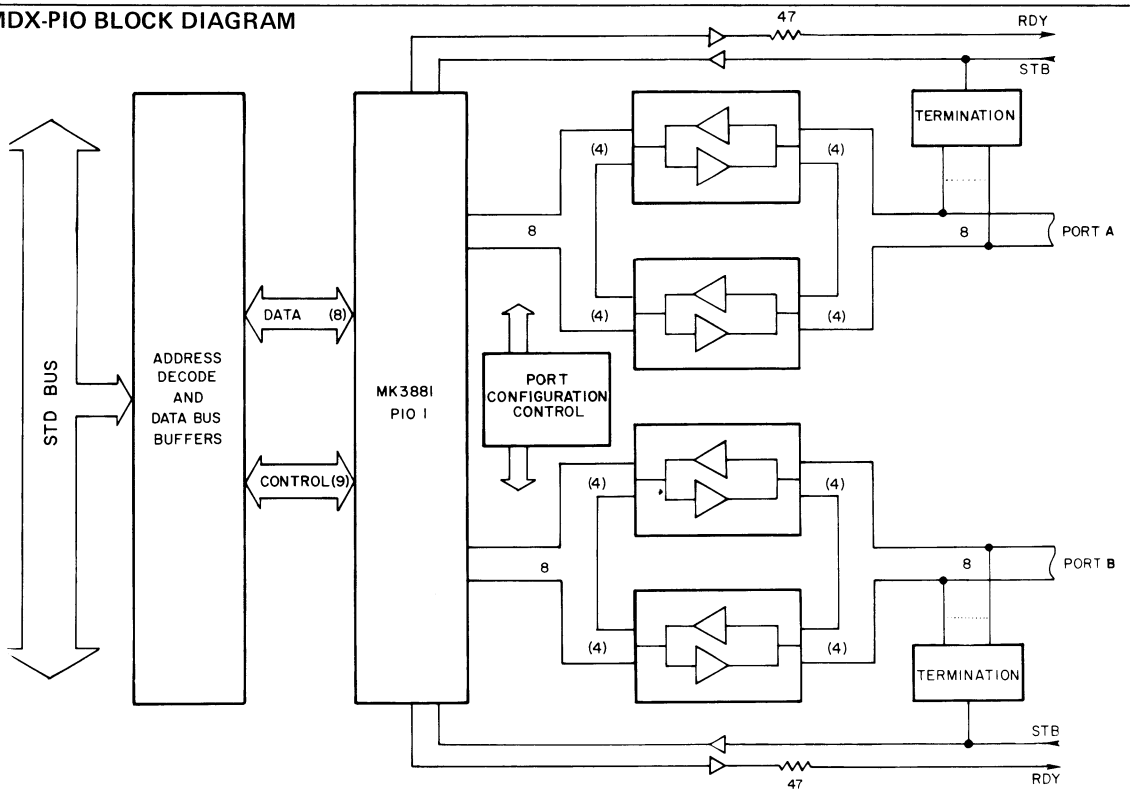
The address decoding, interface and bus management for the board are performed by the address decode and data bus circuit. Each MSX-PIO port has two addresses, one for control and one for Data. A total of eight addresses is utilized per board. These addresses are defined in the table below.

TABLE 1

	PIO 1		PIO 2	
	PORT A	PORT B	PORT A	PORT B
Data	$XX0_8$	$XX2_8$	$XX4_8$	$XX6_8$
Control	$XX1_8$	$XX3_8$	$XX5_8$	$XX7_8$

The XX symbols stand for the upper 5 bits of the I/O channel address. These bits are jumper selectable on the MDX-PIO board in order to provide address selectable fully decoded ports.

MDX-PIO BLOCK DIAGRAM



The circuitry for the other two ports provided by PIO #2 is identical to PIO #1. The port configuration logic, buffers, termination and pin out on connector J-2 is duplicated for PIO #s. These two ports share the address decode and data bus buffer circuitry with PIO #1. The only differences are in the address decoding as given in the port address table, and PIO #2 is lower priority in the daisy chain interrupt structure.

ELECTRICAL SPECIFICATIONS

WORD SIZE:

Data: 8-bits
I/O Addressing: 8-bits

I/O ADDRESSING:

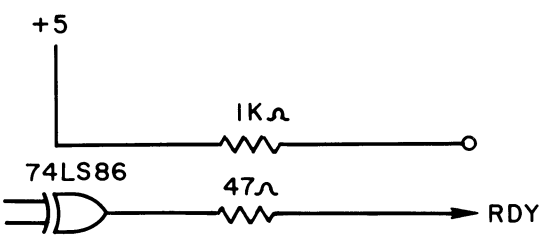
On-board programmable - See Table 1

I/O CAPACITY:

Parallel 4 8-bit ports. On board jumper, Port 1B and 2B only, (configurable in 4 bit bytes) as either In only, Out only or Bidirectional. Automatic handshake provided with each port.

TERMINATORS:

K ohm resistors on all I/O port lines.



PARALLEL BUS INTERFACE-STD BUS COMPATIBLE

Inputs	One 74LS Load Max.
Bus Outputs	I_{OH} = -3 mA min. at 2.4 volts
	I_{OL} = 12mA min. at 0.4 volts

INTERRUPTS

Vectored interrupts generated. Interrupt vector programmable upon initialization. Daisy chained interrupt priority. Selected bit channels can be masked out under program control.

SYSTEM CLOCK

	MIN	MAX
MDX-PIO	250KHz	2.5 MHz
MDX-PIO-4	250KHz	4.0 MHz

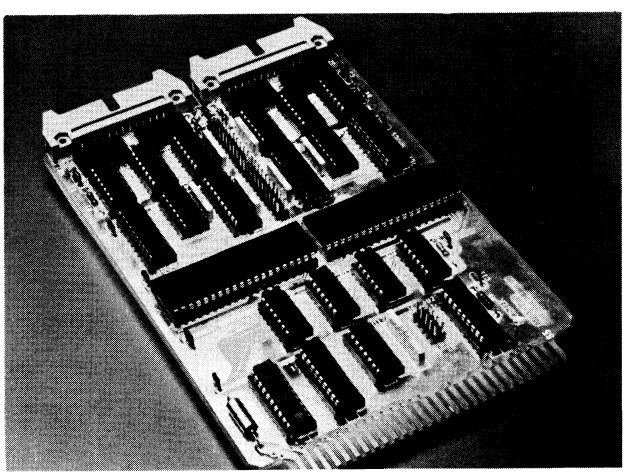
I/O DRIVERS

The following line drivers and terminations are all compatible with the I/O driver sockets on the MDX-PIO.

SIGNALS	TYPE	OUTPUT	SINK CURRENT (mA)
Address, Data Plus & Control	74LS245	NI Tri-State Bidirectional	24
I/O Ports 1A and 2A	*74LS244	NI Tri-State Bidirectional	24
	74LS241	I Tri-State Bidirectional	24
I/O Ports 1B and 2B	*74LS243	NI Tri-State Bidirectional	24
	74LS242	NI Tri-State Bidirectional	24
Handshake: RDY	7486	I/NI (strap selectable)	8

Note: I = inverting N.I. = non-inverting

These chips are normally supplied with the board. They may be exchanged with the other unit listed to provide the alternate signal polarity.



Serial Input/Output Module (MDX-SIO)

FEATURES

- ☐ Two independent full-duplex channels
- ☐ Independent programmable Baud rate clocks
- ☐ Asynchronous data rates - 110 to 19.2K bits per second
- ☐ Receiver data registers quadruply buffered
- ☐ Transmitter data registers double buffered
- ☐ Asynchronous operation
- ☐ Binary synchronous operation
- ☐ HDLC or IBM SDLC operation
- ☐ Both CRC-16 and CRC-CCITT (-0 and -1) hardware implemented
- ☐ Modem control
- ☐ Operates as DTE or DCE
- ☐ Serial input and output as either RS-232 or 20mA current loop
- ☐ Current loop optically isolated
- ☐ Current loop selectable for either active or passive mode
- ☐ Address programmable
- ☐ Compatible with STD BUS

DESCRIPTION

The Serial Input Output Module, MDX-SIO, is designed to be a multiprotocol asynchronous or synchronous I/O module for the STD BUS. The module is designed around the Mostek MK3884 Z80-SIO which provides two full duplex, serial data channels. Each channel has an independent programmable baud rate clock generator to increase module flexibility. Each channel is capable of handling asynchronous, synchronous, and synchronous bit oriented protocols such as IBM BiSync, IBM SDLC, HDLC and virtually any other serial protocol. It can generate CRC codes in any synchronous mode and can be programmed by the CPU for any traditional asynchronous format. The serial input and output data are fully buffered and is provided at the connector as either a 20mA current loop or RS-232-C levels. A modem control section is also provided for handshaking and status. The MDX-SIO module can be jumper configured as a data terminal (DTE) or as a modem (DCE) in order to facilitate a variety of interface configurations.

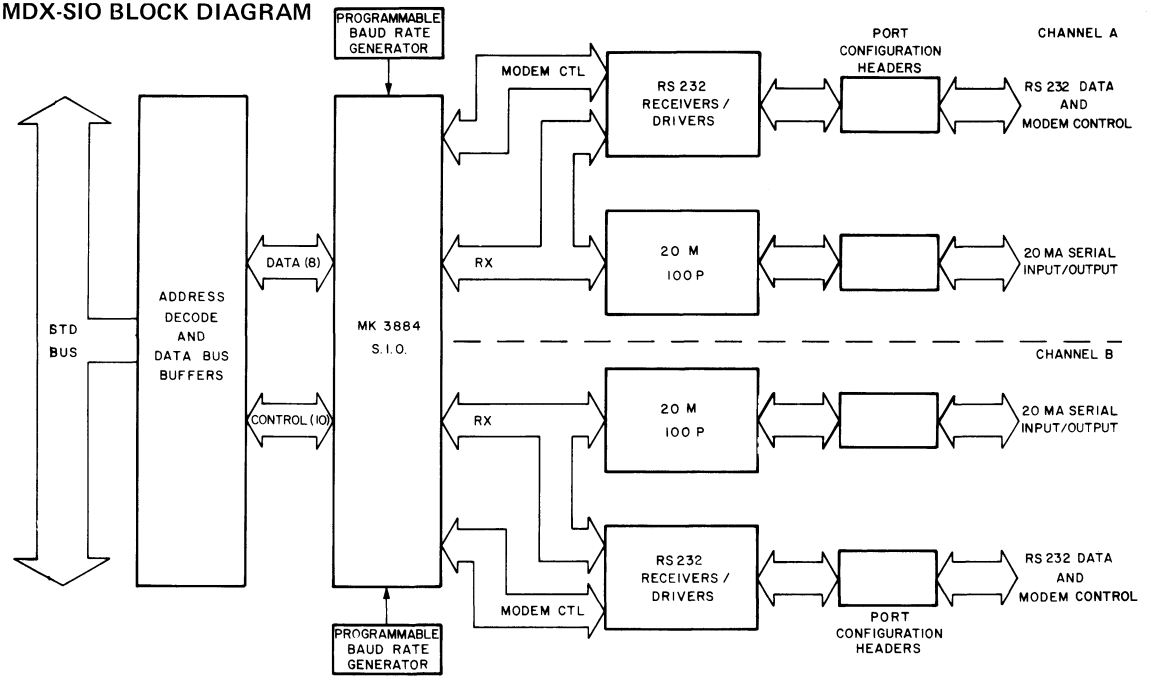
Figure 2 is a block diagram of the MDX-SIO module. It consists of five main elements. They are the channel configuration headers, line drivers and receivers, MK3884 Z80-SIO, programmable Baud rate generator, and address decode and data bus buffers. Input and output to the board is provided via two 26 pin connectors. One connector is dedicated for each channel. The configuration and pin out of each channel is identical.

Several features are available as options that are selected via the channel configuration header. The headers are used to select the orientation of the data communication interface and the mode of the 20mA current loop. The MDX-SIO can be selected to act as either a terminal or processor (Data Terminal Equipment DTE) or as the modem (Data Communications Equipment - DCE). The header allows reconfiguration of both data interchange and modem control signals. This allows increased flexibility necessary to link different hardware elements in OEM data link systems and networks. The module is shipped from the factory wired as a DTE interface.

The MDX-SIO has different selectable options for the 20mA current loop. The receiver and transmitter input and output lines can be reconfigured on the module to allow for reorientation of these signals. Also the receive and transmit circuits can be selected to function in either an active or passive mode. In the active mode, the MDX-SIO module provides the 20mA current source. In the passive mode, the module requires that the loop current be provided. The latter is the same mode as that of a Teletype.

An EIA and 20mA current loop interface circuit is used to provide the necessary level shifting and signal conditioning between the MK3884 Z80-SIO and the connector. These line drivers and receivers provide the correct electrical signal levels, slew rate and impedance for interfacing RS-232C and 20mA current loop peripherals. Additionally, optical isolation is provided for both transmit and receive circuits in the 20mA current loop mode.

MDX-SIO BLOCK DIAGRAM



The Mostek MK3884 Z80-SIO is the central element of this module. This device is a multifunction component designed to satisfy a wide variety of serial data communications requirements in microcomputer systems. Its basic role is this that of a serial to parallel, parallel to serial converter/controller but within that role it is configured by software programming so that its function can be optimized for a given serial data communications application. The MK3884 provides two independent full duplex channels; A and B. Each channel features the following:

- Asynchronous operation
  - 5, 6, 7, or 8 bits/character
  - 1, 1½ or 2 stop bits
  - Even, odd or no parity
  - x1, x16, x32 and x64 clock modes
  - Break generation and detection
  - Parity, Overrun and Framing error detection
- Binary Synchronous operation
  - One or two Sync characters in separate registers
  - Automatic Sync character insertion
  - CRC generation and checking
- HDLC or IBM SDLC operation
  - Automatic Zero insertion and deletion
  - Automatic Flag Insertion
  - Address field recognition
  - I-Field residue handling
  - Valid receive messages protected from overrun
  - CRC generation and checking

The MK3884 also provides modem control inputs and outputs as well as daisy chain priority interrupt logic. Eight different interrupt vectors are generated by the SIO in response to various conditions affecting the data communications channel transmission and reception.

Address decoding, STD BUS interface and bus management for the module are performed by the Address Decode and Data Bus circuit. The MDX-SIO contains command registers that are programmed to select the desired operational mode. The addressing scheme is as follows:

XXXXXX 00	Channel A Data
XXXXXX 01	Channel B Data Command/ Status
XXXXXX 10	Channel A Command/Status
XXXXXX 11	Channel B Command/Status

The X indicates the binary code necessary to represent which of the 64 port address is selected.

Each channel has an individual programmable baud rate generator. The X1 multiplier on the Z80-SIO must be used in the synchronous mode. The X16, X32, or X64 Z80-SIO clock rate can be specified for the asynchronous mode. Table 1 indicates the possible baud rates available for both operation modes with the Z80-SIO Data Rate multipliers.

Figure 1

BAUD RATE (H_Z)			
SYNCHRONOUS		ASYNCHRONOUS	
X1	X*16	X32	X64
800	50	25	12.5
1200	75	37.5	18.75
1760	110	55	27.50
2152	134.5	67.25	33.63
2400	150	75	37.50
4800	300	150	75
9600	600	300	150
19200	1200	600	300
28800	1800	900	450
32000	2000	1000	500
38400	2400	1200	600
57600	3600	1800	900
76800	4800	2400	1200
115200	7200	3600	1800
153600	9600	4800	2400
307200	19200	9600	4800

ELECTRICAL SPECIFICATIONS

WORD SIZE

Data: 8-bits
I/O addressing: 8-bits

I/O ADDRESSING

On board fully programmable

I/O CAPACITY

Serial - Two full duplex serial ports either synchronous of asynchronous. Special control registers and circuitry to permit implementation of SDLC, BiSync, Monosync, HDLC, and other formats can be programmed.

SERIAL BAUD RATES

See Table 1

INTERRUPTS

Generates vectored interrupts to 8 different locations corresponding to conditions within both channels. Interrupt vector location programmable. Daisy chained priority hardware interrupt circuitry.

SYSTEM CLOCK

	MIN	MAX
MDX-SIO	250KHz	2.5 MHz
MDX-SIO-4	250KHz	4.0 MHz

SERIAL COMMUNICATION INTERFACE

Two identical ports

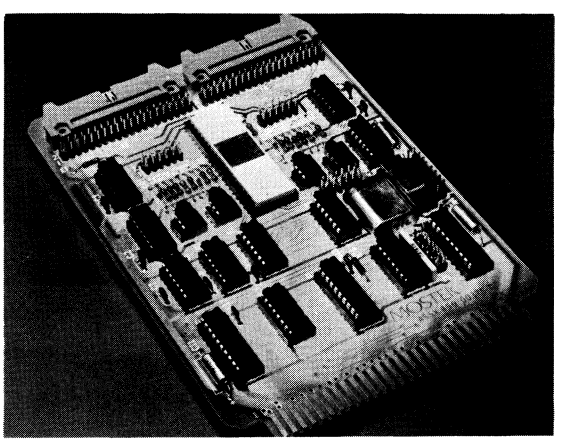
SIGNAL	BUFFERED FOR 20mA CURRENT LOOP	RS-232
Transmitted data	Output	Output
Received data	Input	Input
Data Terminal Ready (DTR)		Input/Output
Request to Send (RTS)		Input/Output
Clear to Send (CTS)		Output/Input
Carrier Detect (CDET)		Output/Input

PARALLEL BUS INTERFACE - STD COMPATIBLE

Inputs	One 74LS load max.
Bus Outputs	I_{OH} = 3mA min at 2.4 volts
	I_{OL} = 12mA min at 0.4 volts

POWER SUPPLY REQUIREMENTS

+12 volts ± 5% at 72 mA max.
-12 volts ± 5% at 46 mA max.
+5 volts ± 5% at 650 mA max.



MOSTEK®

MD SERIES MICROCOMPUTER MODULES

Z80 Microcomputer Debug Module (MDX-DEBUG)

HARDWARE FEATURES

- ☐ STD BUS compatible
- ☐ 4 MHz version available
- ☐ Serial I/O Channel
- ☐ 10K bytes of ROM contain the following firmware

DEBUGGER FEATURES

- ☐ Z80 Operating System with debug capability
- ☐ Channelized I/O for versatility
- ☐ I/O peripheral drivers supplied
- ☐ ROM based

TEXT EDITOR FEATURES

- ☐ Input and modification of ASCII Text
- ☐ Line and character editing
- ☐ Alternate command buffers for pseudo-macro command capability
- ☐ ROM based

ASSEMBLER FEATURES

- ☐ Assembles all Z80 mnemonics
- ☐ Object output in industry standard hexadecimal format extended for Relocatable and Linkable Programs
- ☐ Over fifteen pseudo-ops
- ☐ Two pass assembly
- ☐ ROM based

LINKING LOADER FEATURES

- ☐ Loads into memory both relocatable and non-relocatable object output of the assembler
- ☐ Loads Relocatable modules anywhere in memory
- ☐ Automatically provides linkage of global symbols
- ☐ Automatically provides linkage of global symbols between object modules as they are loaded
- ☐ Prints system load map
- ☐ ROM based

HARDWARE DESCRIPTION

The MDX-DEBUG Module has sockets for 10K bytes of masked ROM that are filled with a Z80 firmware package (DDT-80/ASMB-80). This module has a STD BUS interface and is available in both 2.5MHz and 4.0MHz versions. Included on-board is a fully buffered asynchronous I/O port capable of 110-19200 Baud Rates. Serial Data interfaces are available for 20mA current loop (with reader step control) and RS-232. The on-board Baud Rate Generator is selectable to all common Baud Rates from 110 to 19,200 Baud.

FIRMWARE DESCRIPTION

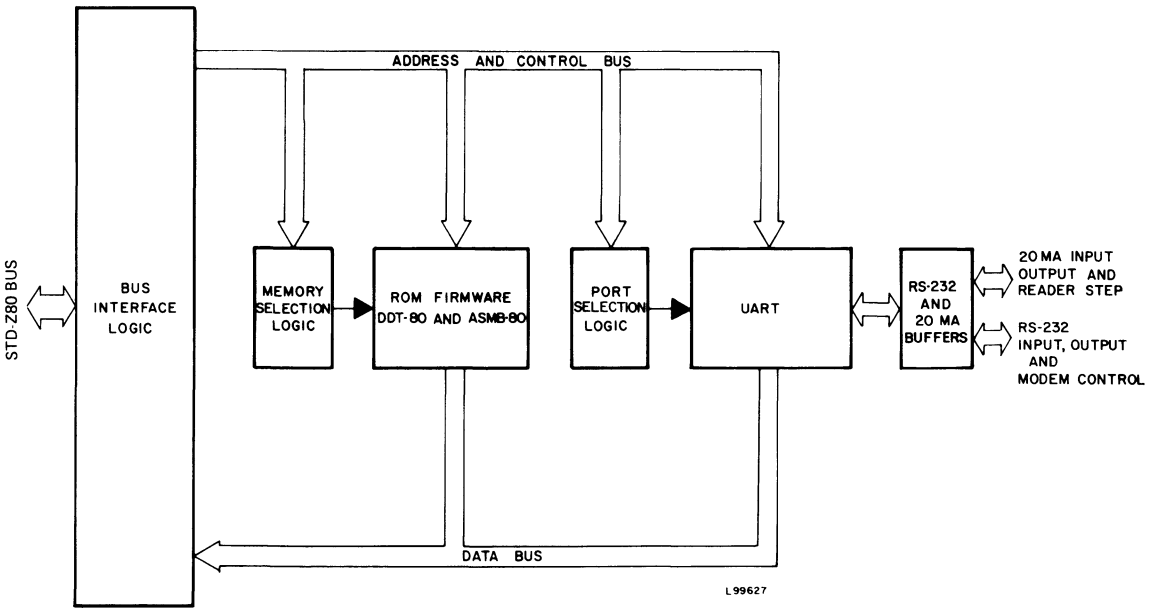
DEBUGGER DESCRIPTION

DDT-80 is the Operating System for the MDX-DEBUG Module. It resides in a 2K ROM (MK34000 series) resident on the MDX-DEBUG Module. It provides the necessary tools and techniques to operate the system, i.e., to efficiently and conveniently perform the tasks necessary to develop microcomputer software. DDT-80 is designed to support the user from initial design through production testing. It allows the user to display and update memory, registers, and ports, load and dump object files, set breakpoints, copy blocks of memory, and execute programs.

DDT-80 COMMAND SUMMARY

M s	- Display and/or update the contents of memory location s.
M s, f	- Tabulate the contents to memory locations s through f.
P s	- Display and/or update the content of I/O port s.
D s, f	- Dump the contents of memory locations s through f in a format suitable to be read by the L command.
L	- Load, into memory, data which is in the appropriate format.
E s	- Transfer control from DDT-80 to a user's program starting at location s.
H	- Perform 16 bit hexadecimal addition and/or subtraction.
C s, f, d	- Copy the contents of memory locations s through f to another location in memory starting at location d.

MDX-DEBUG BLOCK DIAGRAM



B s - Insert a breakpoint in the user's program (must be in RAM) at location s which transfers control back to DDT-80. This allows the user to intercept his program at a specific point (location s) and examine memory and CPU registers to determine if this program is working correctly.

R - Display the contents of the user registers.

The s, f, and d represent start, finish, and destination operands required for each command.

MEMORY- PORT AND REGISTER COMMANDS (M, P, R)

The M, P, and R commands provide the means for displaying the contents of specified memory locations, port addresses, or CPU registers. The M and P commands sequentially access memory locations or ports and display their contents. The user has the option of updating the content of the memory location or port. (Note some ports are output only and their contents cannot be displayed). The M command also gives the user access to the CPU registers through an area in RAM called the Register Map (discussed in the Execute, Breakpoint section below).

The M and R commands are used to tabulate blocks of memory locations (M) or the CPU registers (R). The M command will accept two operands, the starting and ending address of the memory block to be tabulated. The R command will accept either no operand or one. If no operand is specified, the CPU registers will be displayed without a heading. If an operand is specified then a heading which labels the registers contents will be displayed as well.

EXECUTE AND BREAKPOINT (E, B)

The E command is used to execute all programs, including aids such as the Assembler. The B command is used to set a breakpoint to exit from a program at some predetermined location for debugging purposes. At the instant of a breakpoint exit, the contents of all CPU register are saved in a designated area of MDX-DEBUG RAM called the Register Map. In the Register Map, the register contents may be examined or modified using the M command and a predefined mnemonic (or absolute address) of the storage location for that register (example: PC,:A, . . .,:SP). The Register Map is also used to initialize the CPU registers whenever execution is initiated or resumed. Thus the E and B commands can be used together to initiatize, execute, and examine the results of individual program segments.

The B command gives the user the option of having all CPU registers displayed when the breakpoint is encountered. This is done by entering a second operand to the B command. Otherwise DDT-80 defaults to displaying the PC and AF registers. When all CPU registers are displayed, the format is the same as for the R command previously discussed.

LOAD, DUMP, AND COPY (L, D, C)

The L and D commands load and dump object files through the object I/O channel in standard Intel Hex format. Checksums are used for error detection, and the addresses of questionable blocks are typed automatically while loading.

The C command will copy the contents of the memory block specified to another block of memory. There are no restrictions on the direction of the copy or on whether the blocks overlap.

HEXADECIMAL ARITHMETIC (H)

The H command is a dummy command used to allow hexadecimal addition and subtraction for expression evaluation without performing any other operation.

DDT-80 I/O CAPABILITIES

DDT-80 specifies I/O channels, designated 'Console', 'Object', and 'Source', to which any suitable devices may be assigned. The Channel Assignment Table is located in MDX RAM where is may be examined or modified using the M command. The table addresses correspond to the I/O channels and the table contents correspond to the addresses of the peripheral driver routines. A channel which has a device assignment may have that device assignment changed using the M command. This is accomplished by merely modifying the table contents of that channel's table address to correspond to the new peripheral driver routine. A set of peripheral driver routines is supplied and listed below. This scheme also allows the user to write a driver routine for his own peripheral, load it into memory,and easily configure that peripheral into the system.

DDT-80 1/0 PERIPHERAL DRIVERS

1. A serial input driver (usually a keyboard).
2. A serial output driver (usually a CRT or teletype typehead).
3. A serial input driver which sends out a reader step signal (usually a teletype reader).
4. A serial output driver which forces a delay after a carriage return (usually a Silent 700 typehead).
5. A parallel input driver (usually for high speed paper tape output).
6. A parallel output driver (usually for high speed paper tape output).
7. A parallel output driver (usually for a line printer).

TEXT EDITOR DESCRIPTION

The Text Editor permits random access editing of ASCII character strings. It can be used as a line or character oriented editor. Individual characters may be located by position or context. The Editor works on blocks of characters which are typically read into memory from magnetic tape or paper tape. Each edited block can be output to magnetic tape or paper tape after editing is completed. While the primary application for the Text Editor is in editing assembly language source statements, it may be applied to any ASCII text delimited by "carriage returns".

The Editor has a macro command processing option. Up to two sets of commands may be stored and processed at any time during the editing process.

All I/O is done via the DDT-80 channels. The Editor can be used with the MOSTEK ASMB-80 Assembler and Loader to edit, assemble, and load programs in memory without the need for external media for intermediate storage.

The following commands are recognized by the Text Editor:

An -	Advance record pointer n records
Bn -	Backup record pointer n records
Cn dS1dS2d -	Change string S1 to string S2 for n occurrences
Dn -	Delete next n records
E -	Exchange current record with records to be inserted
I -	Insert records
Ln -	Go to line number n
Mn -	Enter command buffers (pseudo-macro)
N -	Print top, bottom, and current line number
Pn -	Punch n records from buffer
R -	Read source records into buffer
Sn dS1d -	Search for nth occurrence of string S1

ASSEMBLER DESCRIPTION

The Assembler reads Z80 source mnemonics and pseudo-ops and outputs an assembly listing and object code. The assembly listing shows address, machine code, statement number, and source statement. The object code is in industry standard hexadecimal format modified for relocatable, linkable assemblies.

The Assembler supports conditional assemblies, global symbols, relocatable programs, and a printed symbol table. It can assemble any length program, limited only by a symbol table size which is user selectable. Expressions involving addition and subtraction are allowed. A global symbol is categorized as "internal" if it appears as a label in the program; otherwise it is an "external" symbol. The printed symbol table shows which symbols are internal and which are external. The Assembler allows the user to select relocatable or non-relocatable assembly via the "PSECT" pseudo-op. Relocation records are placed in the object output for relocatable assemblies (the MOSTEK

object format is defined below). The Assembler can be run as a single pass assembler or as a learning tool. (In this mode, global symbols and forward references are not allowed).

The following pseudo-ops are recognized by the Assembler:

ORG -	program origin
EQU -	equate label
DEFL -	define label
DEFM -	define message
DEFB -	define byte
DEFW -	define word
DEFS	define storage
END -	end statement
NAME -	program name definition
PSECT -	program section definition
GLOBAL -	global symbol definition supports the following assembler pseudo-ops
EJECT -	eject a page of listing
TITLE -	place heading at top of each page
LIST -	turn listing on
NLIST -	turn listing off

RELOCATING LINKING LOADER DESCRIPTION

The MOSTEK Relocating Linking Loader provides state-of-the-art capability for loading programs into memory by allowing loading and linking of any number of relocatable and non-relocatable object modules. Non-relocatable modules are always loaded at their starting address as defined by the ORG psuedo-op during assembly. Relocatable object modules can be positioned anywhere in memory at an offset address.

The Loader automatically links and relocates global symbols which are used to provide communication or linkage between program modules. As object programs are loaded, a table containing global symbol references and definitions is built up. At the end of each module, the loader resolves all references to global symbols which are defined by the current or a previously loaded module. It also prints on the console device the number of defined global symbols that have been referenced. The symbol table can be printed to list all global symbols and their load address. The number of object modules which can be loaded by the Loader is limited only by the amount of MDX-RAM available for the modules and the symbol table. Space for the symbol table is allocated dynamically downward in memory from either the top of memory or from a specified address entered as an operand of the load command.

All I/O is done via the DDT-80 channels. Assemblies can be done from source statements stored in memory (by the Editor). The object output can be directed to a memory buffer rather than to an external device. Thus, assembly and loading can be done without external storage media.

The Loader prints the beginning and ending address of each module as it is loaded. The transfer address as defined by the END pseudo-ops is printed for the first module loaded. The Loader execute command (E) can be used to automatically start execution at the transfer address.

The Loader Commands are the following:

L offset -	load object module at address "off-set" plus program origin address
E -	execute loaded program at transfer address of first module
T -	print global symbol table

MOSTEK OBJECT OUTPUT DEFINITION

Each record of an object module begins with a delimiter (colon or dollar sign) and ends with carriage return and line feed. A colon (:) is used for data records and the end-of-file record. A dollar sign ($) is used for records containing relocation information and linking information. All information is in ASCII. Each record is identified by "type". The type is determined by the 8th and 9th bytes of the record which can take the following values:

00 - data
01 - end-of-file
02 - internal symbol
03 - external symbol
04 - relocation information
05 - module definition

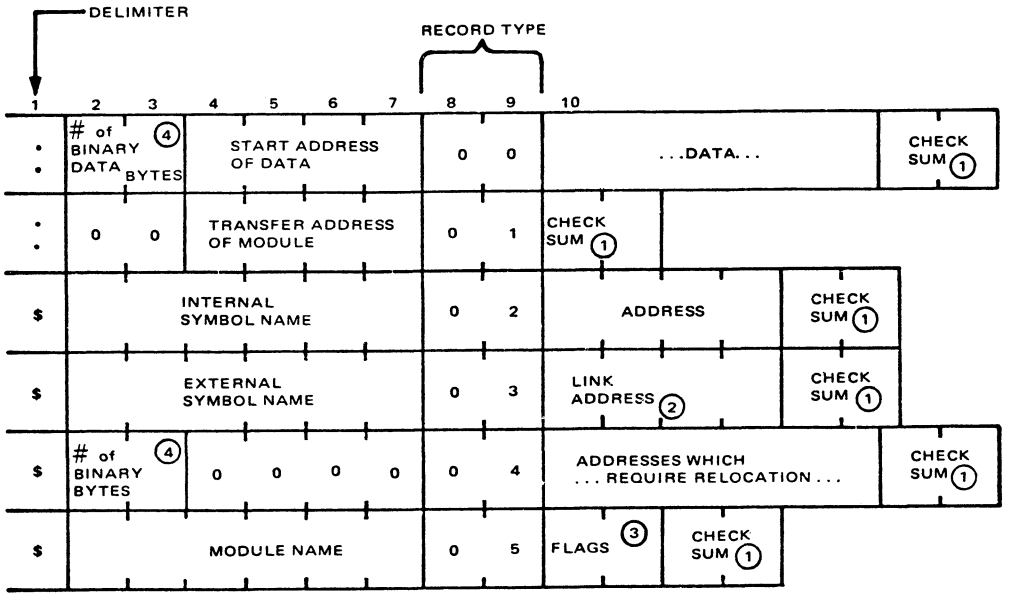
I/O TRANSFER RATE

X16 Baud Rate Clock	Baud Rate (Hz)
1,760	110
4,800	300
9,600	600
19,200	1,200
38,400	2,400
76,800	4,800
153,600	9,600
307,200	19,200

SERIAL COMMUNICATIONS CHARACTERISTICS

Asynchronous

- Full duplex operation
- Start bit verification
- Data word size variable from 5 to 8 bits
- One or two stop bits
- Odd, even, or no parity
- One word buffering on both transmit and on receive.



NOTES:

1. Check Sum is negative of the binary sum of all bytes except delimiter and carriage return/line feed.
2. Link Address points to last address in the data which uses the external symbol. This starts a backward link list through the data records for that external symbol. The list terminates at OFFFFH.
3. The flags are one binary byte. Bit 0 is defined as:
   0 - absolute module
   1 - relocatable module
4. Maximum of 64 ASCII bytes.

SERIAL COMMUNICATIONS INTERFACE

SIGNAL	BUFFERED FOR: 20mA Current Loop	RS-232
Transmitted data	Output	Output
Received data	Input	Input
Data Terminal Ready (DTR)		Input
Request to Send (RTS)		Input
Carrier Detect (CDET)		Output
Clear to Send (CTS)		Output
Data Set Ready (DSR)		Output
Reader Step relay (RS)	Output (20mA)	

PARALLEL BUS INTERFACE-STD BUS COMPATIBLE

Inputs	One 74 LS load Max
Bus Outputs	I_{OH}= -3mA min at 2.4 Volts
	I_{OL}=12mA min at 0.4 Volts

I/O ADDRESSING

On-board Serial I/O Port
Control Port DDH
Data Port DCH
Module and Reader Step Control Port DEH

SYSTEM CLOCK

MDX-DEBUG	1.25MHz	2.5MHz
MDX-DEBUG-4	1.25MHz	4.0MHz

POWER SUPPLY REQUIREMENT

+12 Volts ±5% at 50mA max.
−12 Volts ±5% at 35mA max.
+5 Volts ±5% at 1.2mA max.

MOSTEK®

MD SERIES MICROCOMPUTER MODULES

Z80 Single Step Module (MDX-SST)

FEATURES

- ☐ Hardware single-step capability
- ☐ Compatible with DDT-80 Operating System
- ☐ STD-Z80 BUS compatible

MDX-SST DESCRIPTION

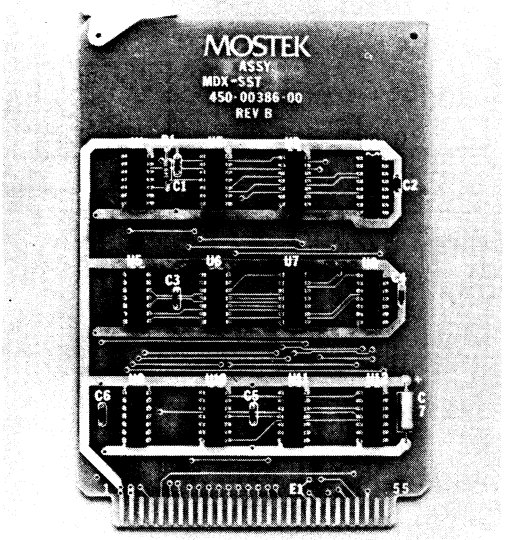
The MOSTEK MDX-SST was designed to enhance the hardware and software debug capability for MD Series systems. The use of the MDX-SST with the MDX-CPU1 and MDX-DEBUG boards allows the user to single-step instructions through RAM and/or EPROM/ROM with the capability of displaying all of the MDX-CPU1 registers on each instruction execution.

The MDX-SST board is implemented using the MDX-CPU1's nonmaskable interrupt and is controlled by firmware from the keyboard. When the command to single step an instruction is given, the sequence of events is the same as executing a program except that a "1" is output to the single step control port (DF_H) instead of a "0". The circuit decodes the double M1 instructions (CB_H, DD_H, ED_H, or FD_H) and M1 is used to clock a shift register circuit which (if a "1" is output to port DF_H) generates a nonmaskable interrupt at the start of the instruction to be single stepped. The nonmaskable interrupt saves the address of execution on the stack and causes the next instruction to be fetched from address $E066_H$. The shift register is clocked twice after the nonmaskable interrupt, causing the signal DEBUG to go low, forcing "E" on the most significant address lines, and causing the instruction to be fetched from the $E066_H$ in the operating system DDT-80. The operating system then jumps to $E069_H$, clears the debug flip-flop by reading PORT DF_H, saves the MDX-CPU1 registers in the MDX-CPU1 scratch RAM, and waits for the next command.

The single-step command is implemented in DDT-80 which resides on the MDX-DEBUG board and has the following format:

S COMMAND, Single-step

This command allows the user to start single-stepping from a given location for a given number of instructions and to display the CPU registers after each step.



Format:

.S aaaa,nn,b(cr)	start single-stepping at location aaaa for nn steps or instructions. If b=0, display only the PC and AF registers, if b≠0, display all the CPU registers.
.S aaaa,nn (cr)	the same as above with b = 0 assumed.
.S aaaa (cr)	the same as above with nn=1 and b=0 assumed.
.S (cr)	the same as above with nn=1 and b=0 assumed; aaaa is set equal to the contents of the user's PC.

The use of the MDX-SST board requires the MDX-CPU1 and the MDX-DEBUG.

SYSTEM CLOCK

MIN	MAX
500KHz	4.00MHz

PARALLEL BUS INTERFACE
STD-Z80 BUS COMPATIBLE

POWER SUPPLY REQUIREMENTS
+5Vdc @ 85mA

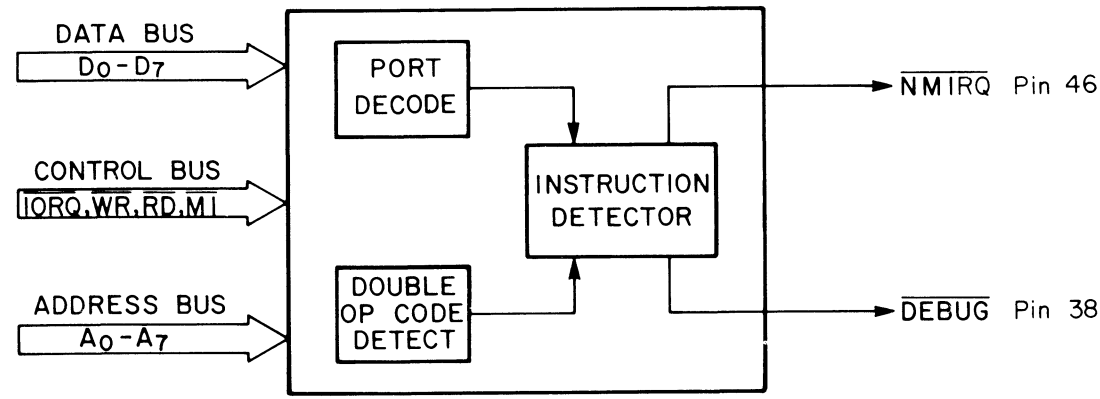
OPERATING TEMPERATURE

0°C to 50°C

ELECTRICAL SPECIFICATIONS

PORT ADDRESS (HEX)
DF

MDX-SST BLOCK DIAGRAM



Universal Memory Card (MDX-UMC)

FEATURES

- Can be strapped to accept the following industry-standard memory devices:

EPROM	STATIC RAM	ROM
2758 (1K x 8)	MK4118 (1K x 8)	
2716 (2K x 8)	MK4802 (2K x 8)	MK34000 (2K x 8)
2732 (4K x 8)		

- Memories can be mixed to form a combination memory board
- Wait state generator for 4MHz operation
- STD-Z80 BUS compatible
- +5 Volt only

MDX-UMC DESCRIPTION

The MDX-UMC is one of MOSTEK's complete line of STD-Z80 BUS compatible microcomputer modules.

Designed as a universal memory card for the STD-Z80 BUS, the MDX-UMC provides the user with the capability of configuring the board to meet the system requirement of ROM/EPROM and/or RAM. By the use of strapping options, the user is able to configure pairs of sockets for ROM/EPROM/RAM to form a combination memory board.

Other MDX-UMC features include 4K boundary addressing and an optional wait-state generator to accomodate slower memories for 4MHz operations.

ELECTRICAL SPECIFICATIONS

WORD SIZE
8 bits

MEMORY ADDRESSING
4K boundaries

MEMORY CAPACITY
8 sockets

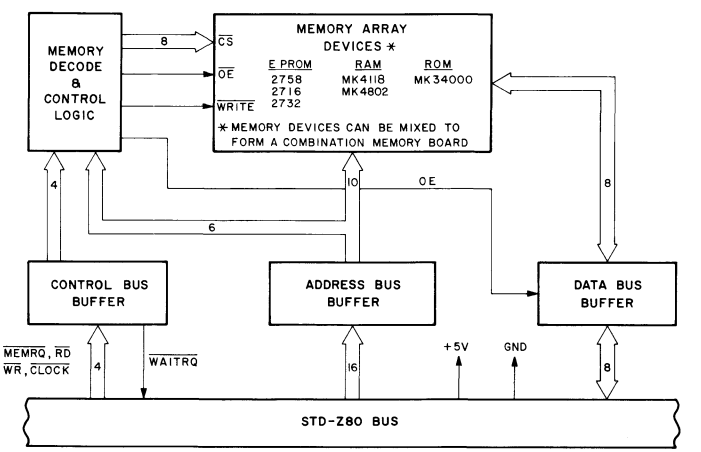
Sockets are strapped in pairs to accomodate the following memories:

EPROM	STATIC RAM	ROM
2758	MK4118	
2716	MK4802	MK34000
2732		

PARALLEL BUS INTERFACE - STD-Z80 BUS COMPATIBLE

Inputs: One 74LS load max.
BUS Outputs: I_{OH} = -15mA min at 2.4 Volts
I_{OL} = 24mA min at 0.5 Volts

MDX-UMC BLOCK DIAGRAM



EPROM Module (MDX-EPROM)

FEATURES

- ☐ Accepts the following industry standard EPROMS:
  2758 (1K x 8)
  2716 (2K x 8)
  2732 (4K x 8)
- ☐ Eight EPROM sockets for maximum storage of:
  8K x 8 using 2758's
  16K x 8 using 2716's
  32K x 8 using 2732's
- ☐ Wait state generator for 4MHz operation
- ☐ STD-Z80 BUS compatible
- ☐ +5 Volt only

MDX-EPROM DESCRIPTION

The MDX-EPROM is designed to be an EPROM memory expansion board for the MOSTEK MD SERIES™ of Z80-based microcomputers. The MDX-EPROM accepts the following EPROMS; 2758 (1K x 8), 2716 (2K x 8) and 2732 (4K x 8) which gives a maximum storage capacity of 8K, 16K, or 32K bytes respectively.

Starting address selection is provided for positioning the MDX-EPROM on any 4K boundary. A wait-state generator is also provided for optional 4MHz operation.

ELECTRICAL SPECIFICATIONS

WORD SIZE
8 bits

MEMORY CAPACITY
8K x 8 using eight 2758's
16K x 8 using eight 2716's*
32K x 8 using eight 2732's
*EPROMS included

REQUIRED ACCESS TIME

MEMORY TIME	MIN ACCESS TIME	CYCLE TIME
2758, 2716, 2732	450ns*	450ns

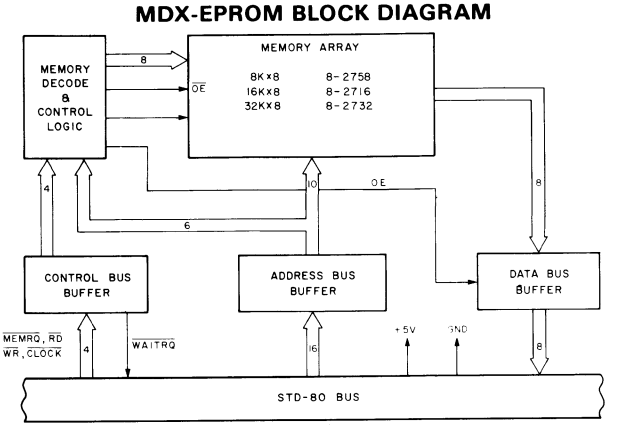
*One wait state must be added for 4MHz operation.

ADDRESS SELECTION
4K boundaries

BUS INTERFACE
STD-Z80 BUS compatible
Inputs: One 74LS load max.
Bus Outputs: I_{OH} = -15mA min at 2.4 Volts
I_{OH} = 24mA min at 0.5 Volts

POWER SUPPLY REQUIREMENTS*
+5 Volts ± 5% at 0.45A
*Does not include EPROMs. Add 100 mA for each EPROM.

MDX-EPROM BLOCK DIAGRAM



Static RAM Module (MDX-SRAM)

FEATURES

- ☐ Three memory sizes
  - 4K x 8 (MDX-SRAM4)
  - 8K x 8 (MDX-SRAM8)
  - 16K x 8 (MDX-SRAM16)
- ☐ Selectable starting adddress on 4K boundaries
- ☐ 2.5 MHz and 4.0 MHz compatible
- ☐ STD-Z80 BUS compatible
- ☐ +5 Volt only

MDX-SRAM DESCRIPTION

The MDX-SRAM is designed to be a static RAM Memory expansion board for the MOSTEK MD SERIES of Z80 based microcomputers. It is available in three memory capacities; 4K bytes (MDX-SRAM4), 8K bytes (MDX-SRAM8), and 16K bytes (MDX-SRAM16). Additionally, all MDX-SRAM boards are 2.5MHz and 4.0MHz compatible. Thus, the designer can choose from three options available and tailor the add-on static RAM directly to the system requirements.

The MDX-SRAM is designed using the state of the art MK4118 (1Kx8) static RAM and MK4802 (2Kx8) static

RAM memory devices. Because of the high speed of the MK4118 and MK4802, no wait states are necessary for operating the MDX-SRAM at 2.5MHz or 4.0MHz. Address selection is provided on all MDX-SRAM cards for positioning the 4K, 8K, or 16K of memory to start on any 4K boundary.

ADDRESS SELECTION

Selection of 4K, 8K, or 16K contiguous memory blocks to begin on any 4K boundary.

BUS INTERFACE

STD-Z80 BUS compatible
Inputs: One 74LS load max
Bus Outputs: I_{OH} = -15mA min at 2.4 Volts
I_{OL} = 24mA min at 0.5 Volts

POWER SUPPLY REQUIREMENTS

BOARDS	+5V ± 5%
MDX-SRAM4	0.8 A max
MDX-SRAM8	1.2 A max
MDX-SRAM16	1.2 A max

ELECTRICAL SPECIFICATIONS

WORD SIZE
8 bits

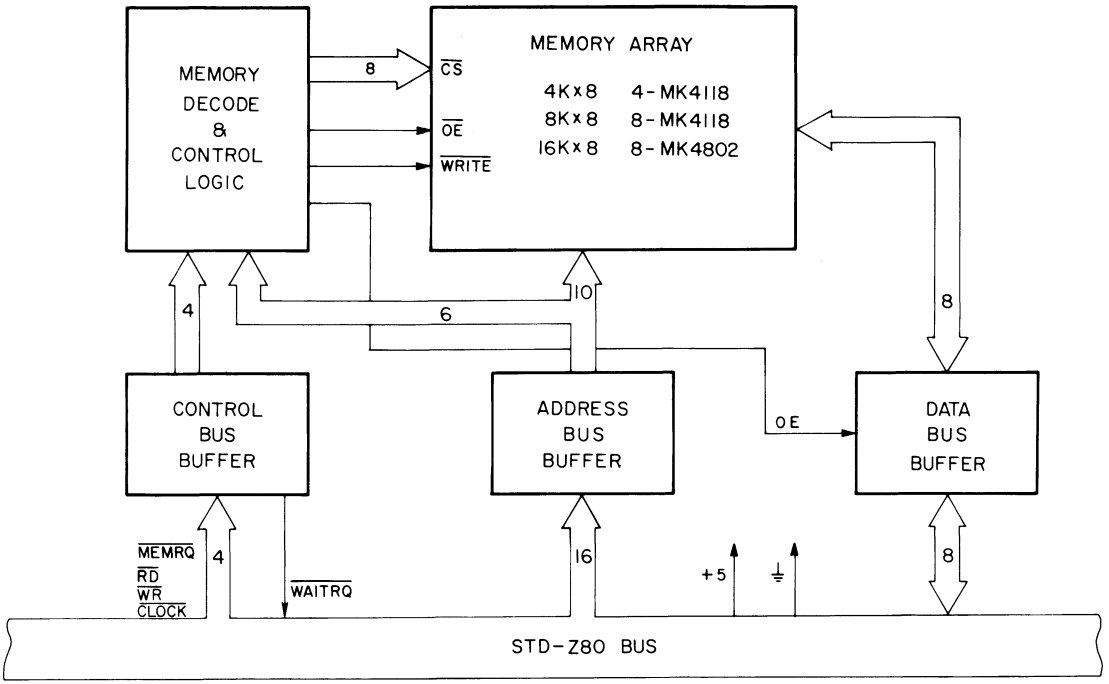
MEMORY SIZE

MDX-SRAM4 - 4,096 bytes
MDX-SRAM8 - 8,192 bytes
MDX-SRAM16 - 16,384 bytes

TIMING

	MEMORY ACCESS	MEMORY CYCLE
MDX-SRAM	250ns max.	250ns min.

MDX-SRAM BLOCK DIAGRAM



Analog to Digital Conversion Module (MDX-A/D)

FEATURES

- ☐ 8-Bit A/D converter with 16 single-ended analog inputs
- ☐ 3 full-scale input ranges
  - 0 to +1 Volts
  - 0 to +2 Volts
  - 0 to +5 Volts
- ☐ Total unadjusted error $< \pm$ ½ LSB
- ☐ Linearity error $< \pm$ ½ LSB
- ☐ No missing codes
- ☐ Guaranteed monotonicity
- ☐ No zero adjust required
- ☐ No full scale adjust required
- ☐ Provisions for additional channel expansion
- ☐ Optional sample and hold
- ☐ Address programmable
- ☐ 4MHz option
- ☐ Compatible with STD-Z80 BUS

DESCRIPTION

The MD Series and the STD-Z80 BUS were designed to satisfy the need for low-cost OEM microcomputer modules. The STD-Z80 BUS uses a motherboard interconnect system concept and is designed to handle any MD Series card type in any slot. The modules for the STD-Z80 BUS are a compact 4.5 x 6.5 inches which provides for system partitioning by function (RAM, EPROM, I/O). This smaller module size makes system packaging easier while increasing MOS-LSI densities providing high functionality per module.

The MD Series of OEM microcomputer boards and the STD-Z80 BUS offer the most cost effective system configuration available to the OEM system designer.

MDX-A/D DESCRIPTION

The Analog to Digital Converter Module, MDX-A/D, is designed to be a 16 channel single-ended A/D module for the STD-Z80 BUS. The module is designed around the MOSTEK MK5160 8-bit A/D converter/16 channel analog multiplexer. Additional provisions have been included to allow further analog expansion if desired. Also, an optional Sample and Hold module (AD582) may be added to increase system performance. Figure 1 is a block diagram of the MDX-A/D showing the major elements of the module.

The first element of this board is the multiplexer. This 16-channel multiplexer can directly access any one of 16 single-ended analog channels and provides logic for additional channel expansion. All analog input lines contain a diode/resistor protection circuit to reduce damage potential from overvoltage and transient inputs.

The output of the multiplexer can either drive the A/D converter directly or a Sample and Hold (S/H) module version is available. The board is shipped normally without a Sample and Hold.

If an S/H function is required, an Analog Devices AD582 needs to be inserted and one jumper removed. This circuitry allow sampling of signals up to 5KHz with a nominal 150nsec aperture time.

The other half of the MK5160 is the A/D converter. The 8-bit A/D consists of 256 series resistors with an analog switch tree, a chopper stabilized comparator and a sucessive approximation register. The series resistor approach guarantees monotonicity and no missing codes. The need for external zero and full-scale adjustments has been eliminated and an absolute accuracy of $\leq$ 1 LSB including quantizing error is provided. A start convert signal initiates the conversion process and can be jumper selected from either an external source or under program control. Upon completion, a DONE signal is generated to indicate end of conversion. This signal is used to flag the program as well as any external device.

The Data Bus Buffer and Interface Logic allows the MDX-A/D module to interface with the STD-Z80 BUS. It provides buffering for all signals as well as address decoding and A/D port control. A total of 4 port address locations are required and can start on any four-word boundary.

ELECTRICAL SPECIFICATIONS

WORD SIZE
Data: 8 bits
I/O Addressing: 8 bits

I/O ADDRESSING
On board programmable on 4-word boundaries
X X X X X X 0 0 A/D Port Configuration Data
X X X X X X 0 1 A/D Port Configuration Control
X X X X X X 1 0 A/D Data Input/Output Port
X X X X X X 1 1 Data Control Port

I/O CAPACITY

Eight bit analog to digital converter with up to sixteen single ended analog input channels. Channel expansion available. Start conversion and done handshake signals available at the edge connector.

Three full scale input ranges: 0 to +I Volt, 0 to +2 Volts and 0 to +5 Volts.

INTERRUPTS

Vectored interrupts generated. Interrupt vector programmable upon initialization. Daisy-chained interrupt priority. Interrupts are controlled by a MOSTEK MK3881 Parallel I/O controller chip.

SYSTEM CLOCK

	MIN	MAX
MDX-A/D	250 KHz	2.5 MHz
MDX-A/D-4	250 KHz	4.0 MHz

ELECTRICAL SPECIFICATIONS

POWER SUPPLY REQUIREMENTS
+12 Volts ±5% at 30 mA max
-12 Volts ±5% at 15 mA max
+5 Volts ± 5% at 0.6 A max

SAMPLE/HOLD OPTION DATA
DROOP RATE: 100mV at 25°C
APERTURE TIME: 150nsec
MAX INPUT FREQUENCY: 5KHz
APERTURE JITTER: 15 nsec

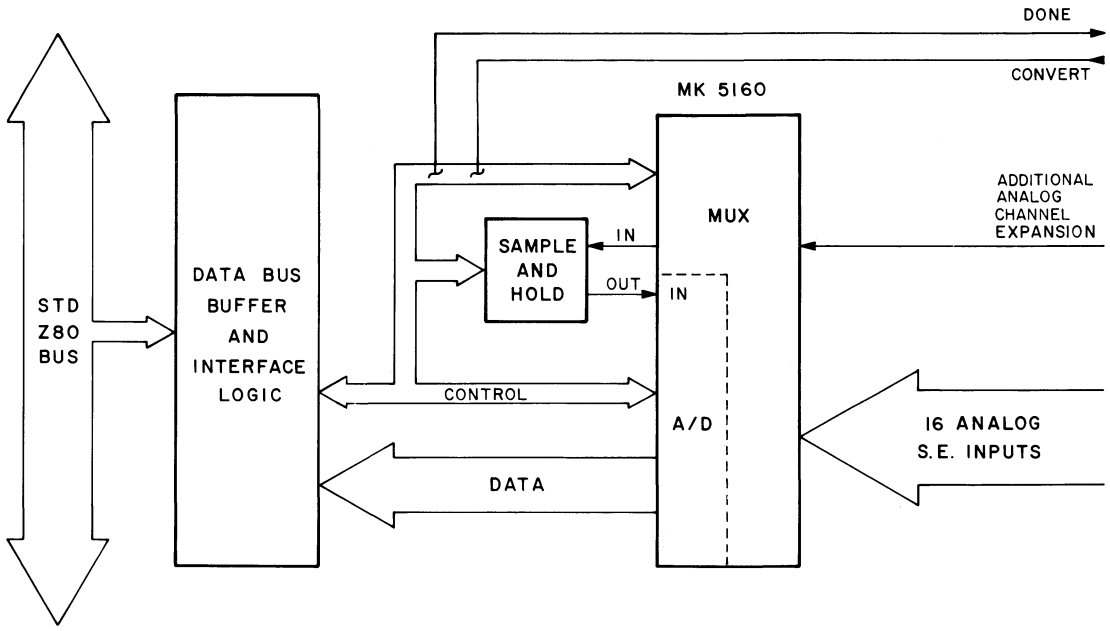
CONVERSION TIME
138 microseconds max

OPERATING TEMPERATURE RANGE
0° to +50°C

MECHANICAL SPECIFICATIONS

CARD DIMENSIONS
4.5 in. (11.43cm) high by 6.50 in. (16.51cm) long
0.48 in. (1.22cm) maximum profile thickness
0.062 in. (0.16cm) printed circuit board thickness

MDX-A/D BLOCK DIAGRAM
Figure 1



MD SERIES MICROCOMPUTER MODULES

ORDERING INFORMATION

DESIGNATOR	DESCRIPTION	PART NO.
MDX-CPU1	Module with Operational Manual less EPROMs and mating connectors. 2.5MHz version.	MK77850
MDX-CPU1-4	4.0 MHz version.	MK77850-4
MDX-DRAM8	Module with Operation Manual less mating connectors 8K Bytes 2.5MHz	MK77750
MDX-DRAM16	16K Bytes 2.5MHz	MK77751
MDX-DRAM32	32K Bytes 2.5MHz	MK77752
MDX-DRAM16-4	16K Bytes 4MHz	MK77754-4
MDX-DRAM32-4	32K Bytes 4MHz	MK77752-4
MDX-EPROM/UART	Module with Operation Manual less EPROMs and mating connectors. 2.5MHz version.	MK77753
MDX-EPROM/UART-4	Module with Operation Manual less EPROMs and mating connectors. 4.0MHz version.	MK77753-4
MDX-PIO	Module with Operation Manual less mating connectors. 2.5MHz version.	MK77650
MDX-PIO-4	Module with Operation Manual less mating connectors. 4.0MHz version.	MK77650-4
MDX-SIO	Dual channel, Full-Duplex Serial I/O Module less mating connectors (with Operation Manual) 2.5MHz version.	MK77651
MDX-SIO-4	Module with Operation Manual less mating connectors. 4.0MHz version.	MK77651-4
MDX-DEBUG	Module with 10K bytes of firmware and Operation Manual. No mating connectors. 2.5MHz version.	MK77950
MDX-DEBUG-4	Module with 10K bytes of firmware and Operation Manual. No mating connectors. 4.0MHz version. Program Source Listing of 10K byte firmware package (DDT/ASMB-80) including comments and flow charts. (Available free with purchase of either MDX-DEBUG Module).	MK77950-4

MD SERIES MICROCOMPUTER MODULES

ORDERING INFORMATION

DESIGNATOR	DESCRIPTION	PART NO.
MDX-SST	Module with Operation Manual	MK 77958
MDX-UMC	Module with Operation Manual	MK 77759
MDX-EPROM	Module with Operation Manual less EPROM Memories	MK 77758
MDX-SRAM 4	4K Bytes (4118) Module with Operation Manual	MK 77755
MDX-SRAM 8	8K Bytes (4118) version	MK 77756
MDX-SRAM 16	16K bytes (4802) version	MK 77757
MDX-A/D 8	Module with Operation Manual less mating connectors — 2.5 MHz version	MK 77653
MDX-A/D 8H	Above with sample hold.	MK 77653H
MDX-A/D 8-4 MDX-A/D 8H-4	4.0 MHz version	MK 77653-4 MK 77653H-4
MDX-PROTO	See page 47 for details	
MDX-PROTO 4	See page 47 for details	
MD-ACC	Accessories — See page 49 for details	

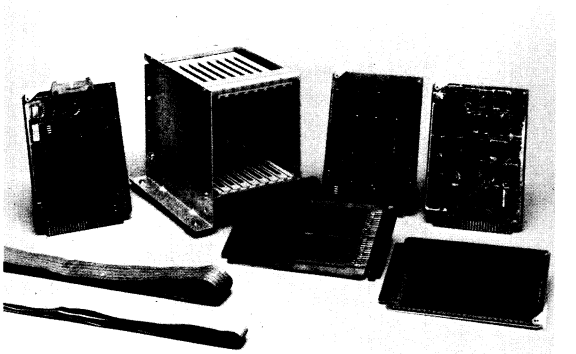
MOSTEK®

MD SERIES MICROCOMPUTER MODULES

Prototyping Package (MDX-PROTO)

FEATURES

- ☐ 8-slot card cage with mother board (MK77954)
- ☐ MDX-CPU1 module (MK77850)
- ☐ MDX-DRAM8 module (MK77750)
- ☐ MDX-DEBUG module (MK77950)
- ☐ MD-WW2 Wire wrap board (MK77952)
- ☐ MD-EXT Extender board (MK77593)
- ☐ Cables for RS232 device (MK77955) or TTY (MK77956)
- ☐ 4MHz option available (MDX-PROTO-4)
- ☐ STD BUS compatible



DESCRIPTION

The MD Series and the STD BUS were designed to satisfy the need for low-cost OEM microcomputer modules. The STD BUS uses a mother board interconnect system concept and is designed to handle any MD Series Card type in any slot. The modules for the STD BUS are a compact 4.5 x 6.5 inches which provide for system partitioning by function (RAM, EPROM, I/O). This smaller module size makes system packaging easier while increasing MOS-LSI densities provide high functionality per module.

The MD Series of OEM microcomputer boards and the STD BUS offer the most cost-effective system configuration available to the OEM system designer.

MDX-PROTO DESCRIPTION

HARDWARE DESCRIPTION

MDX-CPU1 DESCRIPTION

The MOSTEK MDX-CPU1 is the heart of an MD Series Z80 system. Based on the powerful Z80 microprocessor, the MDX-CPU1 can be used with great versatility in an OEM microcomputer system application. This is done simply by inserting custom ROM or EPROM memories into the sockets provided on the board and configuring them virtually anywhere within the Z80 memory map.

On-board memory is provided in the form of 4K of EPROM (2-2716's) and 256 bytes of scratchpad RAM as pictured in the block diagram. In addition, a MK3882 Counter Time Circuit is included on the MDX-CPU1 to provide counting and timing functions for the Z80. Either 2716 EPROM can be located at any 2K boundary within any given 16K block in the Z80 memory map via a jumper arrangement.

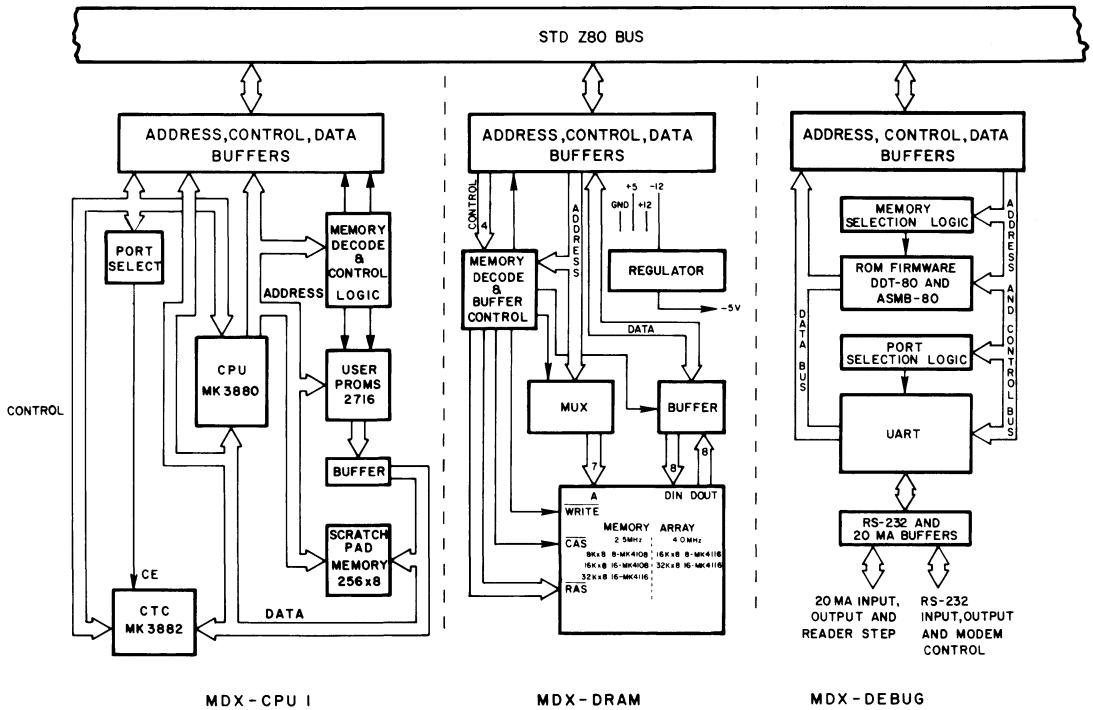
The MDX-CPU1 can be used in conjunction with the MDX-DEBUG and MDX-DRAM modules to utilize DDT-80 and ASMB-80 in system development. This is accomplished by strapping the scratchpad RAM to reside at location FF00 so that it will act as the Operating System RAM for DDT-80.

The MDX-CPU1 is also available in 4MHz version (MDX-CPU1-4). In this version, one wait cycle is automatically inserted each time on-board memory is accessed by a read or write cycle. This is necessary to make the access times of the 2716 PROMs and the 3539 scratchpad RAM compatible with MK3880-4 4MHz Z80-CPU.

MDX-DRAM DESCRIPTION

The MDX-DRAM is designed to be a RAM memory expansion board for the MOSTEK MD SERIES of Z80 based microcomputers. It is available in three memory capacities: 8K bytes (MDX-DRAM8), 16K bytes (MDX

BLOCK DIAGRAM MDX-PROTO



DRAM 16), and 32K bytes (MDX-DRAM32). Additionally, the MDX-DRAM16 and the MDX-DRAM32 are available in a 4MHz version. Thus, the designer can choose from the various options to tailor his add-on dynamic RAM directly to his system requirements.

The MDX-DRAM8 is designed using MOSTEK's MK4108 8,192-bit dynamic RAM. The MDX-DRAM32 utilizes high-performance MK4116, 16K-bit dynamic RAMs which allow 4MHz versions of these boards to be offered. No wait-state insertion circuitry is required on any of the RAM cards.

Address selection is provided on all MDX-DRAM cards for positioning the 8K, 16K, or 32K of memory to start on any 4K boundary.

MDX-DEBUG DESCRIPTION

The MDX-DEBUG Module has sockets for 10K bytes of masked ROM that are populated with a Z80 firmware package (DDT-80/ASMB-80). This module has a STD BUS interface and is available in both 2.5MHz and 4.0MHz versions. Included on board is a fully buffered asynchronous I/O port capable of 110-19200 Baud Rates. Serial Data interfaces are available for 20mA current loop (with reader step control) and RS-232. The on-board Baud Rate Generator is selectable to all common Baud Rates from 110 to 19,200 Baud.

FIRMWARE DESCRIPTION

DEBUGGER DESCRIPTION

DDT-80 is the Operating System for the MDX-DEBUG Module. It resides in a 2K ROM (MK34000 series) resident on the MDX-DEBUG Module. It provides the necessary tools and techniques to operate the system, i.e., to efficiency and conveniently develop microcomputer software. DDT-80 is designed to support the user from initial design through production testing. It allows the user to display and update memory, registers, and ports, load and dump object files, set breakpoints, copy blocks of memory, and execute programs.

DDT-80 COMMAND SUMMARY

M s	- Display and/or update the contents of memory location s.
M s, f	- Tabulate the contents of memory locations s through f.
P s	- Display and/or update the contents of I/O port s.
D s,f	- Dump the contents of memory locations s through f in a format suitable to read by the L command.
L	- Load, into memory, data which is in the appropriate format.
E s	- Transfer control from DDT-80 to a user's program starting at location s.

H - Perform 16-bit hexadecimal addition and/or subtraction.

C s,f,d - Copy the contents of memory locations s through f to another location in memory starting at location d.

B s - Insert a breakpoint in the user's program (must be in RAM) at location s which transfers control back to DDT-80. This allows the user to intercept his program at a specific point (location s) and examine memory and CPU registers to determine if this program is working correctly.

R - Display the contents of user registers.

The s,f, and d represent start, finish, and destination operands required for each command.

MEMORY, PORT AND REGISTER COMMANDS (M,P,R)

The M, P, and R commands provide the means for displaying the contents of specified memory location, port addresses, or CPU registers. The M and P commands sequentially access memory locations or ports and display their contents. The user has the option of updating the content of the memory location or port. (Note some ports are output only and their contents cannot be read or displayed). The M command also gives the user access to the CPU registers through an area in RAM called the Register Map (discussed in the Execute, Breakpoint section below).

The M and R commands are used to tabulate blocks of memory locations (M) or the CPU registers (R). The M command will accept two operands, the starting and ending addresses of the memory block to tabulated. The R command will accept either no operand or one. If no operand is specified, the CPU registers will be displayed without a heading. If an operand is specified then a heading which labels the register contents will be displayed as well.

EXECUTE AND BREAKPOINT (E,B).

The E command is used to execute all programs, including aids such as the Assembler. The B command is used to set a breakpoint to exit from a program at some predetermined location for debugging purposes. At the instant of a breakpoint exit, the contents of all CPU registers are saved in a designated area of MDX-DEBUG RAM called the Register Map. In the Register Map, the register contents may be examined or modified using the M command and a predefined mnemonic (or absolute address) of the storage location for that register (Example: PC, :A,...,:SP). The Register Map is also used to initialized the CPU registers whenever execution is initiated or resumed. Thus the E and B commands can be used together to initialize, execute, and examine the results of individual program segments.

The B command gives the user the option of having all CPU registers displayed when the breakpoint is encountered. This is done by entering a second operand to the B command. Otherwise DDT-80 defaults to displaying the PC and AF registers. When all CPU registers are displayed, the format is the same as for the R command previously discussed.

LOAD, DUMP, AND COPY, (L,D,C)

The L and D commands load and dump object files through the object I/O channel in standard Intel Hex format. Checksums are used for error detection, and the addresses of questionable blocks are typed automatically while loading.

The C command will copy the contents of the memory block specified to another block of memory. There are no restrictions on the direction of the copy or on whether the blocks overlap.

HEXADECIMAL ARITHMETIC (H)

The H command is a dummy command used to allow hexadecimal addition and subtraction for expression evaluation without performing any other operation.

DDT-80 I/O CAPABILITIES

DDT-80 specifies I/O channels, designated 'Console', 'Object', and 'Source', to which any suitable devices may be assigned. The Channel Assignment Table is located in MDX-RAM where it may be examined or modified using the M command. The table addresses correspond to the I/O channels and the table contents correspond to the addresses of the peripheral driver routines. A channel which has a device assignment may have that device assignment changed using the M command. This is accomplished by merely modifying the table contents of that channel's table address to correspond to the new peripheral driver routine. A set of peripheral driver routines is supplied and listed below. This scheme also allows the user to write a driver routine for his own peripheral, load it into memory, and easily configure that peripheral into the system.

DDT-80 I/O PERIPHERAL DRIVERS

1. A serial input driver (usually a keyboard).
2. A serial output driver (usually a CRT or teletype typehead).
3. A serial input driver which sends out a reader step signal (usually a teletype reader).
4. A serial output driver which forces a delay after a carriage return (usually a Silent 700 typehead).
5. A parallel input driver (usually for high-speed paper tape input).
6. A parallel output driver (usually for high-speed paper tape output).
7. A parallel output driver (usually for a line printer).

TEXT EDITOR DESCRIPTION

The Text Editor permits random access editing of ASCII character strings. It can be used as a line or character-oriented editor. Individual characters may be located by position or context. The Editor works on blocks of characters which are typically read into memory from magnetic tape or paper tape. Each edited block can be output to magnetic tape or paper tape after editing is completed. While the primary application for the Text

Editor is in editing assembly language source statements, it may be applied to any ASCII text delimited by "carriage returns".

The Editor has a macro command processing option. Up to two sets of commands may be stored and processed at any time during the editing process. All I/O is done via the DDT-80 channels. The Editor can be used with the MOSTEK ASMB-80 Assembler and Loader to edit, assemble, and load programs in memory without the need for external media for intermediate storage.

The following commands are recognized by the Text Editor:

An	- Advance record pointer n seconds
Bn	- Backup record pointer n seconds
Cn dS1dS2D	- Change string S1 to string S2 for n occurences
Dn	- Delete n records
E	- Exchange current record with records to be inserted
I	- Insert records
Ln	- Go to line number n
Mn	- Enter command buffers (pseudo-macro)
N	- Print top, bottom and current line number
Pn	- Punch n records from buffer
R	- Read source records into buffer
Sn dS1d	- Search for nth occurrence of signal S1

ASSEMBLER DESCRIPTION

The Assembler reads Z80 source mnemonics and pseudo-ops and outputs an assembly listing and object code. The assembly listing shows address, machine code, statement number, and source statement. The object code is in industry-standard hexadecimal format modified for relocatable, linkable assemblies.

The Assembler supports conditional assemblies, global symbols, relocatable programs and a printed symbol table. It can assemble any length program, limited only by a symbol table size which is user selectable. Expressions involving addition and subtraction are allowed. A global symbol is catagorized as "internal" if it appears as a label in the program; otherwise it is an "external" symbol. The printed symbol table shows which symbols are internal and which are external. The assembler allows the user to select relocatable or non-relocatable assembly via the "PSECT" pseudo-op. Relocation records are placed in the object output for relocatable assemblies. (The MOSTEK object format is defined below.) The Assembler can be run as a single-pass assembler or as a learning tool. (In this mode, global symbols and forward references are not allowed.) The following pseudo-ops are recognized by the Assembler:

EQU	- equate label
DEFL	- define label
DEFM	- define message
DEFB	- define byte
DEFW	- define word
DEFS	- define storage
END	- end statement
NAME	- program name definition
PSECT	- global symbol definition Supports the following assembler psuedo-ops
EJECT	- eject a page of listing
TITLE	- place heading at top of each page
LIST	- turn listing on
NLIST	- turn listing off

RELOCATING LINKING LOADER DESCRIPTION

The MOSTEK Relocating Linking Loader provides state-of-the-art capability for loading programs into memory by allowing loading and linking of any number of relocatable and non-relocatable object modules. Non-relocatable modules are always loaded at their starting address as defined by the ORG pseudo-op during assembly. Relocatable object modules can be positioned anywhere in memory at an offset address.

The Loader automatically links and relocates global symbols which are used to provide communication or linkage betweeen program modules. As object programs are loaded a table containing global symbol references and definitions is built up. At the end of each module, the loader resolves all references to global symbols which are defined by the current or a previously loaded module. It also prints on the console device the number of defined global symbols that have been referenced. The symbol table can be printed in order to list all global symbols and their load address. The number of object modules which can be loaded by the Loader is limited only by the amount of MDX-RAM available for the modules and the symbol table. Space for the symbol table is allocated dynamically downward in memory from either the top of memory or from a specified address entered as an operand of the load command.

All I/O is done via the DDT-80 channels. Assemblies can be done from source statements stored in memory (by the Editor). The object output can be directed to a memory buffer rather than to an external device. Thus, assembly and loading can be done without external storage media.

The Loader prints the beginning and ending address of each module as it is loaded. The transfer address as defined by the END pseudo-op is printed for the first module loaded. The Loader execute command (E) can be used to automatically start execution at the transfer address.

The Loader Commands are the following:

L offset	- load object module at address "offset" plus program origin address
E	- execute loaded program at transfer address of first module
T	- print global symbol table

MOSTEK OBJECT OUTPUT DEFINITION

Each record of an object module begins with a delimiter (colon or dollar sign) and ends with carriage return and line feed. A colon (:) is used for data records and end-of-file record. A dollar sign ($) is used for records containing relocation information and linking

ELECTRICAL SPECIFICATIONS

MDX-CPU1

WORD SIZE

Instruction: 8, 16, 24, or 32 bits
Data: 8 bits

CYCLE TIME

Clock period or T state = 0.4 microsecond @ 2.5MHz
0.25 microsecond @ 4.00 MHz
Instructions require from 4 to 23 T states

MEMORY ADDRESSING

On-Board EPROM: jumper selectable for any 2K boundary within a 16K block of Z80 memory map.
On-Board RAM: FF00-FFFF

I/O ADDRESSING

On-Board Serial I/O Port
Control Port DDH
Data Port DCH
Module and Reader Step Control Port DEH

SYSTEM CLOCK

MDX-DEBUG	1.25MHz	2.5MHz
MDX-DEBUG-4	1.25MHz	4.0MHz

POWER SUPPLY REQUIREMENT

+12 Volts ± 5% at 50 mA max.
-12 Volts ± 5% at 35 mA max.
+5 Volts ± 5% at 1.2 mA max.

MEMORY CAPACITY

On-Board EPROM - 4K bytes (sockets only)
On-Board RAM-256 bytes
Off-board Expansion - Up to 65,536 bytes with user-specified combinations of RAM, ROM, PROM.

MEMORY SPEED REQUIRED

MEMORY	ACCESS TIME	CYCLE TIME
2716*	450ns	450ns

*Single 5 volt type required

I/O ADDRESSING

On-Board Programmable Timer

CONNECTORS

FUNCTION	CONFIGURATION	MATING CONNECTOR
STD BUS	56 pin dual	Printed Circuit Viking 3VH28/1CE5
	0.125 in. centers	Wire Wrap Viking 3VH28/1CND5
		Solder Lug Viking 3VH28/1CN5

MD-CC8 STD BUSSED 1/4 rack (MK77954) bussed motherboard with eight connectors on 0.5 in. centers.

STD BUS Organization

RS232 Cable MD-RS232 — 26 pin socket connector ANSLEY #609-2061M 5 feet of 26 wire flatcable ANSLEY #171-26 25-pin standard EIA ANSLEY #609-25S

TTY Cable MD-TTY — 26 pin socket connector ANSLEY #609-2061M 5 feet of 26 wire flatcable ANSLEY #171-26 TTY connector Molex 15 Pin Molex #03-09-2151

ORDERING INFORMATION

DESIGNATOR	DESCRIPTION	PART NO.
MDX-PROTO	2.5 MHz Prototyping kit with Manuals (Includes the 8K bytes MDX-DRAM 8)	MK 77951
MDX-PROTO 4	4.0 MHz prototyping kit with manuals (Includes the 16K bytes MDX-DRAM 16-4)	MK 77951-4

NEW MD SERIES PRODUCTS

		AVAILABLE
MDX-FLP	Floppy Disk Controller card for 8 or 5 inch floppy disk drives, single or double sided.	October
MD DOS	Minimal Controller Software for the MDX-FLP.	October
FLP-80DOS/MD	Complete Disk Operating System for supporting BASIC, etc...	October
MDX-MATH	Industry standard Math Processor chip on the Industry standard STD BUS.	October
MDX-SC/D	LED 7-segment Display and thumbwheel switch on the outside card edge allows application parameters to be field programmed and status to be displayed.	October
MDX-80/P	19-inch Rack Card Cage and power supply; designed to further simplify construction of MD board system.	November
MDX-A/D 10 MDX-A/D 12 MDX-D/A 12	Family of high precision Analog I/O cards. Up to 16 channels on one board or 64 with expander card.	November
MITE-80 for the Z80	Real Time Multi-Tasking Executive for the Z80. Supports multiple activities on one Z80 CPU.	December
MDX-CPU2	New Z80 CPU card using the new Mostek 1K x 8 RAM, 4K RAM/PROM Sockets and CTC on board.	January 1980
MDX-INT	Interrupt-Expander: contains 2 CTC's and WAIT state generator	January
MEDECS	Diagnostic Package for on-site testing of MD Series cards	January

MD-ACC

The following items are available as accessories to support design, development, and production of products designed around the MOSTEK MD Series Z80 microcomputer modules:

- WW1 wire wrap card with bussed power and ground
- WW2 wire wrap card without bussed power and ground
- MD-CC8 8-slot card cage
- MD-CC14 14-slot card cage
- MD-CC28 28-slot card cage
- MD-EXT Extender card.

Description

The STD BUS concept is a joint design between Mostek and Pro-Log to satisfy the need for cost-effective OEM Microcomputer Systems. The definition of the STD BUS and the MD Series of OEM microcomputer modules are a result of years of microcomputer component and module manufacturing experience. The STD BUS uses a motherboard interconnect system concept and is designed to handle any MD Series card in any card slot. Modules for the STD BUS range from CPU, RAM and EPROM Modules to Input, Output, A/D, and TRIAC control modules. A ROM-based DEBUG module provides users of the STD BUS with Edit, Assembly, and Debug capability using only an ASCII terminal.

Printed circuit modules for the STD BUS are a compact 4.5 x 6.5 inches providing for system partitioning by function (RAM, PROM, I/O). This smaller module size makes system packaging easier while increasing MOS-LSI densities provide high functionality per module.

MECHANICAL SPECIFICATIONS

CARD DIMENSIONS

4.5 in (11.43cm) high by 6.50 in. (16.51cm) long
0.48 in. (1.22cm) maximum profile thickness
0.062 in. (0.16cm) printed circuit board thickness

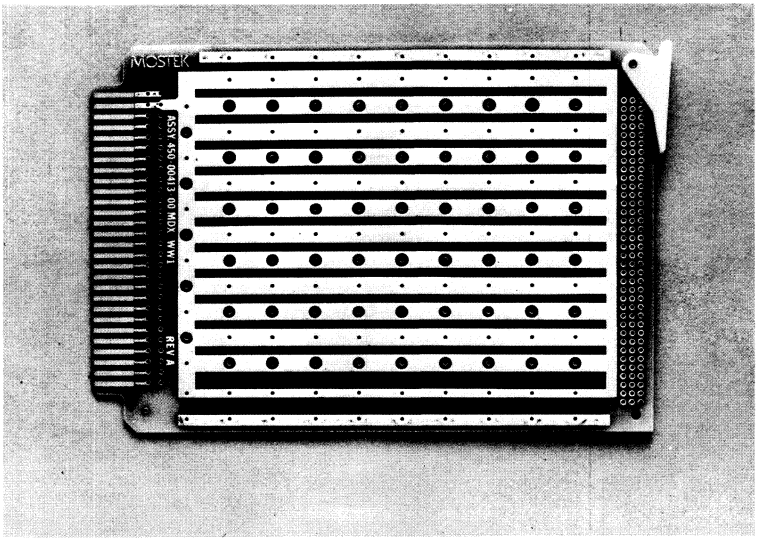
CONNECTORS

FUNCTION	CONFIGURATION	MATING CONNECTOR
STD-Z80 BUS	56 pin dual read out 0.125 in. centers	Printed Circuit Viking 3VH28/ICE5 Wire Wrap Viking 3VH28/1CND5 Solder Lug Viking 3VH28/1CN5

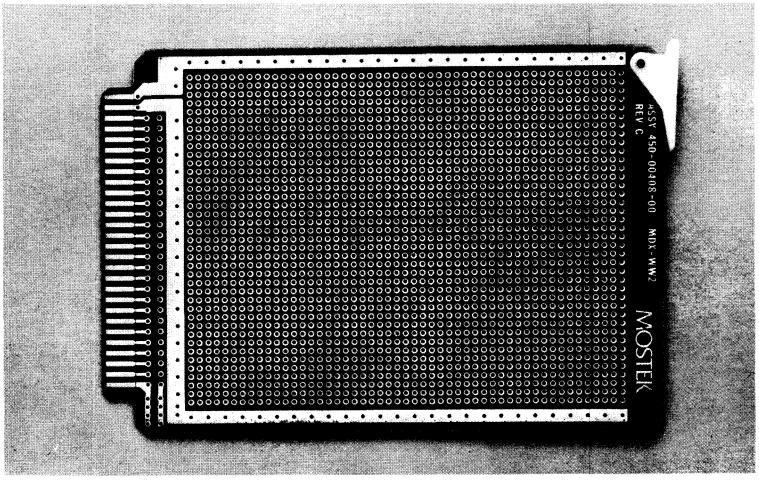
ORDER INFORMATION

DESIGNATOR	DESCRIPTION	PART NO.
MD-WW1	MD Series wire wrap card with bussed power and ground	MK77959
MD-WW2	MD Series wire wrap card without bussed power and ground	MK77952
MD-EXT	MD Series extender card	MK77953
MD-CC8	MD Series 8-slot card cage with STD BUS motherboard	MK77954
MD-CC14	MD Series 14-slot card cage with STD BUS motherboard.	MK77960
MD-CC28	MD Series 28-slot card cage with STD BUS motherboard	MK77961

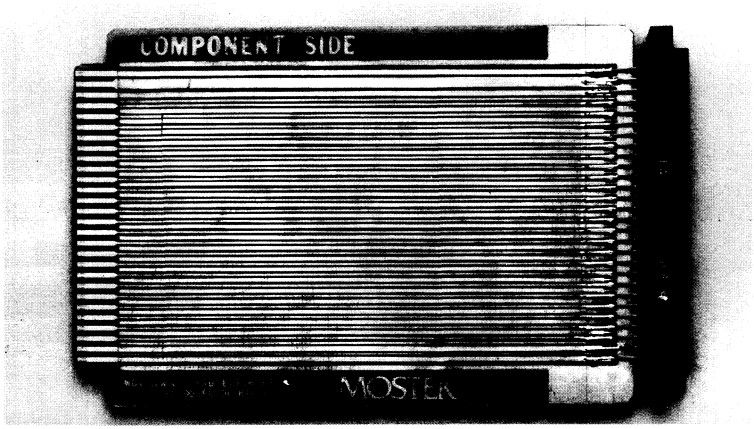
WW1 PHOTO MK77959



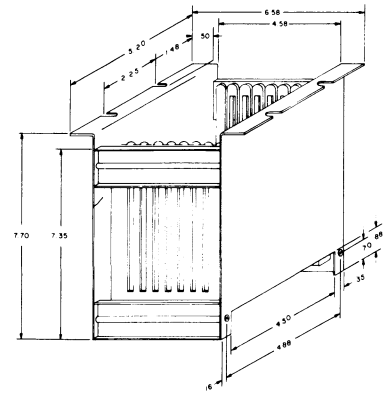
WW2 PHOTO MK77952



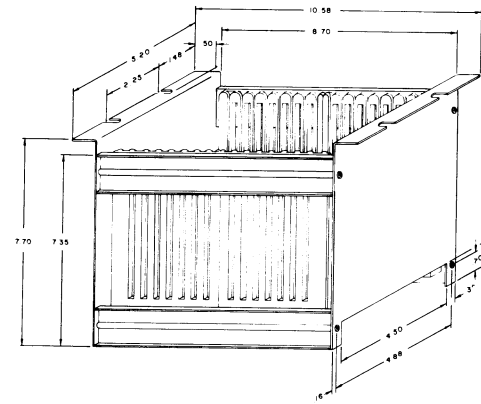
MD - EXT PHOTO MK77953



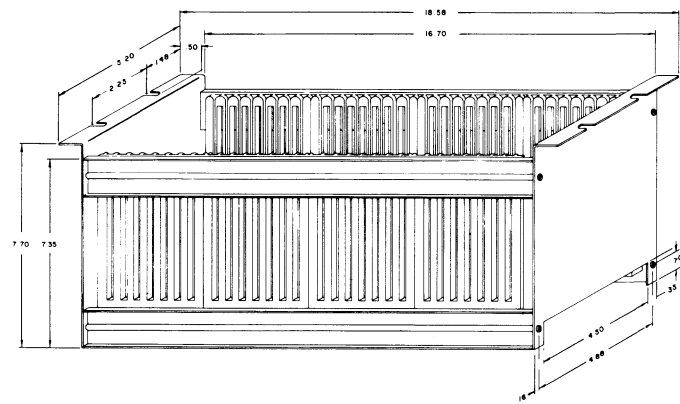
MD-CC 8 Drawing with Dimensions



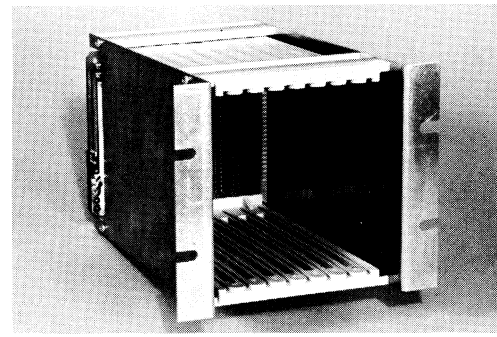
MD-CC 14 Drawing with Dimensions



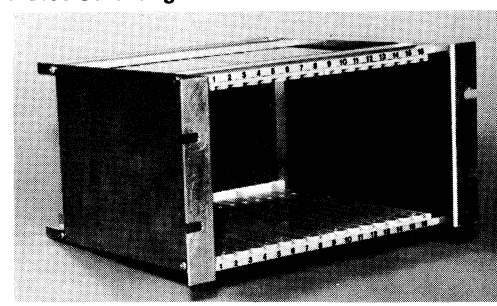
MD-CC 28 Drawing with Dimensions



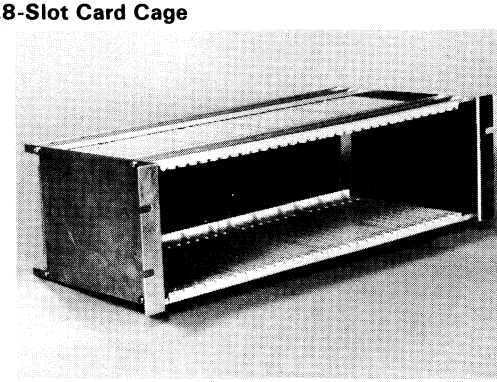
MD-CC8 8-Slot Card Cage



MD-CC14 14-Slot Card Cage



MD-CC28 28-Slot Card Cage



STD-Z80 BUS DESCRIPTION AND ELECTRICAL SPECIFICATIONS

Application Note

DESCRIPTION

The purpose of this application note is to provide the O.E.M. system designer with more information about the STD-Z80 BUS. The information presented is a bus description of the STD-Z80 BUS, the pin out, and the recommended BUS loading specifications.

In April of 1978, several meetings were held between MOSTEK CORP. and PROLOG CORP. to discuss the possibility of defining a new O.E.M. microcomputer board BUS. The goals for the new BUS were that it be simple to interface to, be well defined, and be able to use a standard 56 pin edge card connector. The results of these meetings were successful, and the STD BUS was defined.

The STD BUS was defined as a general purpose microprocessor bus which is capable of supporting the following processors: Z80, 8080, 8085, 6800, and 6809. It is possible to design simple function cards which will work with each of the processors, however it may be difficult or impossible to design an add on card which used one of the many peripheral chips and then have the card work with all of the STD BUS processors. It was for this reason that MOSTEK defined the STD-Z80 BUS. The STD-Z80 is a subset of the general purpose STD BUS and is defined exclusively for the Z80. By specifying the STD-Z80 bus, exact functional pin descriptions and bus timing can be given. Therefore, a STD-Z80 system will be guaranteed to work with all STD-Z80 designed boards.

The STD-Z80 backplane pin assignments are listed and described in Table 1. A table showing the BUS pins versus BUS signals is shown in Table 2.

STD-Z80 BUS DESCRIPTION

Table 1

BUS PIN	MNEMONIC	DESCRIPTION
1	+5V	+5Vdc system power
2	+5V	+5Vdc system power
3	GND	Ground-System signal ground and DC return
4	GND	Ground-System signal ground and DC return
5	-5V	-5Vdc system power
6	-5V	-5Vdc system power
7	D3	Data Bus (Tri—state, input/output, active high). D0-D7 constitute an 8-bit bi-directional data bus. The data bus is used for data exchange with memory and I/O devices.
8	D7	
9	D2	
10	D6	
11	D1	
12	D5	
13	D0	
14	D4	
15	A7	Address Bus (Tri-state, output, active high). A0-A15 make up a 16-bit address bus. The address bus provides the address for memory (up to 65K bytes) data exchanges and for I/O device data exchanges. I/O addressing uses the lower 8 address bits to allow the user to directly select up to 256 input or 256 output ports. A0 is the least significant address bit. During refresh time, the lower 7 bits contain a valid refresh refresh address for dynamic memories.
16	A15	
17	A6	
18	A14	
19	A5	
20	A13	
21	A4	
22	A12	
23	A3	
24	A11	
25	A2	
26	A10	
27	A1	
28	A9	
29	A0	
30	A8	
31	/WR	Write (Tri-state, output, active low) /WR indicates that the CPU data bus holds valid data to be stored in the addressed memory or I/O device.
32	/RD	Read (Tri-state, output, active low). RD indicates that the CPU wants to read data from memory or an I/O device. The addressed I/O device or memory should use this signal to gate data onto the CPU data bus.
33	/IORQ	Input/Output Request (Tri-state, output, active low). The /IORQ signal indicates that the lower half of the address bus holds a valid I/O address for an I/O read or write operation. An /IORQ signal is also generated with a /M1 signal when an interrupt is being acknowledged to indicate that an interrupt

		response vector can be placed on the data bus. Interrupt Acknowledge operations occur during /M1 time, while I/O operations never occur during /M1 time.
34	/MEMRQ	Memory Request (Tri-state, output, active low). The /MEMRQ signal indicates that the address bus holds a valid address for a memory read or memory write operation.
35	/IOEXP	I/O Expansion, not used on MDX cards. (Normally strapped to ground on the MOSTEK motherboard)
36	/MEMEX	Memory Expansion, not used on Mostek MDX cards.(Normally strapped to ground on the MOSTEK motherboard)
37	/REFRESH	REFRESH (Tri-state, output, active low). /REFRESH indicates that the lower 7 bits of the address bus contain a refresh address for dynamic memories and the /MEMRQ signal should be used to perform a refresh cycle for all dynamic RAMs in the system. During the refresh cycle A7 is a logic 0 and the upper 8 bits of the address bus contains the I register.
38	/MCSYNC	Not generated on the MOSTEK MDX-CPU1. Can be generated by gating the following signals:/RD+ /WR + /INTAK. By connecting a jumper on the MDX-CPU1, this line becomes /DEBUG (Input). /DEBUG is used in conjunction with the DDT-80 operating system on the MDX-DEBUG card, and the MDX-SST card for implementing a hardware single step function. When pulled low, the /DEBUG line will set an address modification latch which will force the upper three address lines A15, A14, and A13 to a logic 1. These address lines will remain at a logic 1 until reset by performing any I/O operation.
39	/STATUS 1	Machine Cycle One (Tri-state, output, active low). /M1 indicates that the current machine cycle is in the op code fetch cycle of an instruction. Note that during the execution of two-byte op-codes /M1 will be generated as each op-code is fetched. These two-byte op-codes always begin with a CBh, DDh, EDh, or FDh. /M1 also occurs with IORQ to indicate an interrupt acknowledge cycle.
40	/STATUS 0	Not used on Mostek MDX cards.
41	/BUSAK	Bus Acknowledge (Output, active low). Bus acknowledge is used to indicate to the requesting device that the CPU address bus, data bus, and control bus signals have been set to their high impedance state and the external device can now control the bus.
42	/BUSRQ	Bus Request (Input, active low). The /BUSRQ signal is used to request the CPU address bus, data bus, and control signal bus to go to a high impedance state so that other devices can control those buses. When /BUSRQ is activated, the CPU will set these buses to a high impedance state as soon as the current CPU machine cycle is terminated and the /BUSAK signal is activated.
43	/INTAK	Interrupt Acknowledge (Tri-state, output, active low). The /INTAK signal indicates that an interrupt acknowledge cycle is in progress, and the interrupting device should place its response vector on the data bus. The /INTAK signal is equivalent to an IORQ during an /M1.
44	/INTRQ	Interrupt Request (Input, active low). The Interrupt Request signal is generated by I/O devices. A request will be honored at the end of the current instruction if the internal software controlled interrupt enable flip flop (IFF)

		is enabled and if the BUSRQ signal is not active. When the CPU accepts the interrupt, an interrupt acknowledge signal /INTAK (IORQ during an M1) is sent out at the beginning of the next instruction.
45	/WAITRQ	Wait Request (Input, active low). Wait Request indicates to the CPU that the addressed memory or I/O device is not ready for a data transfer. The CPU continues to enter wait states for as long as this signal is active. This signal allows memory or I/O devices of any speed to be synchronized to the CPU. Use of this signal postpones refresh as long as it held active.
46	/NMIRQ	Non-Maskable Interrupt Request (Input, negative edge triggered). The Non-Maskable Interrupt Request line has a higher priority than the /INTRQ line and is always recognized at the end of the current instruction, independent of the status of the interrupt enable flip-flop. /NMIRQ automatically forces the CPU to restart to location 0066h. The program counter is automatically saved in the external stack so that the user can return to the program that was interrupted. Note that continuous WAIT cycles can prevent the current instruction from ending and that a /BUSRQ will override a /NMIRQ.
47	/SYSRESET	System Reset (Output, active low). The System Reset line indicates that a reset has been generated either from an external reset or the power on reset circuit. The system reset will occur only once per reset and will be approximately 2 microseconds in duration. A system reset will also force the CPU program counter to zero, disable interrupts, set the I register to 00h, set the R register to 00h, and set Interrupt Mode 0.
48	/PBRESET	Push Button Reset (Input, active low). The Push Button Reset will generate a debounced system reset.
49	/CLOCK	Processor Clock (Output, active low). Single phase system clock.
50	/CNTRL	Not used on MOSTEK MDX cards.
*51	PCO	Priority Chain Output (Output, active high). The signal is used to form a priority-interrupt daisy chain when more than one interrupt-driven device is being used. A high level on this pin indicates that no other devices of higher priority are being serviced by a CPU interrupt service routine.
*52	PCI	Priority Chain In (Input, active high). This signal is used to form a priority-interrupt daisy chain when more than one interrupt-driven device is being used. A high level on this pin indicates that no other devices of higher priority are being serviced by a CPU interrupt service routine.
53	AUX GND	Auxiliary Ground (Bussed)
54	AUX GND	Auxiliary Ground (Bussed)
55	+12V	+12Vdc system power
56	-12V	-12Vdc system power

NOTES:

1. Input/Output references of each signal are made with respect to MDX-CPU1 module.
2. The following signals have pull-up resistors: /WR, /RD, /IORQ, /MEMRQ, /REFRESH, /DEBUG, /M1, /BUSRQ, /INTAK, /INTRQ, /WAITRQ, /NMIRQ, /SYSRESET, /PBRESET, and /CLOCK. The value of the pull-up resistors are 1K except for /WAITRQ which is 500 ohms and /PBRESET which is 10K ohms. These resistors are located on the MDX-CPU1 module.

*The Mostek card cage is prioritized from left to right as viewed from the top with component side of boards to the left.

STD—Z80 ELECTRICAL BUS SPECIFICATIONS

BUS RECEIVERS

Logical Low: 0.8V maximum at -0.36mA
Logical High: 2.0V minimum at 20μA

BUS DRIVERS

Logical Low: 0.5V maximum at 24mA
Logical High: 2.4V minimum at -15mA
Off State Output Current (tri-state): $\pm$100μA

RECOMMENDED BUS DRIVERS AND RECEIVERS

Bus Drivers: 74LS240, 74LS241, 74LS373, 74LS374, and 74LS244.
Bus Receivers: 74LS240, 74LS241, and 74LS244.
Bus Transceivers: 74LS245, 74LS242, and 74LS243.

STD BUS PIN-OUT

	COMPONENT SIDE				CIRCUIT SIDE			
	PIN	MNEMONIC	SIGNAL FLOW	DESCRIPTION	PIN	MNEMONIC	SIGNAL FLOW	DESCRIPTION
LOGIC POWER BUS	1	+5V	In	+5 Volts DC (Bussed)	2	+5V	In	+5 Volts DC (Bussed)
	3	GND	In	Digital Ground (Bussed)	4	GND	In	Digital Ground (Bussed)
	5	−5V	In	−5 Volts DC	6	−5V	In	−5 Volts DC
DATA BUS	7	D3	In/Out	Low Order Data Bus	8	D7	In/Out	High Order Data Bus
	9	D2	In/Out	Low Order Data Bus	10	D6	In/Out	High Order Data Bus
	11	D1	In/Out	Low Order Data Bus	12	D5	In/Out	High Order Data Bus
	13	D0	In/Out	Low Order Data Bus	14	D4	In/Out	High Order Data Bus
ADDRESS BUS	15	A7	Out	Low Order Address Bus	16	A15	Out	High Order Address Bus
	17	A6	Out	Low Order Address Bus	18	A14	Out	High Order Address Bus
	19	A5	Out	Low Order Address Bus	20	A13	Out	High Order Address Bus
	21	A4	Out	Low Order Address Bus	22	A12	Out	High Order Address Bus
	23	A3	Out	Low Order Address Bus	24	A11	Out	High Order Address Bus
	25	A2	Out	Low Order Address Bus	26	A10	Out	High Order Address Bus
	27	A1	Out	Low Order Address Bus	28	A9	Out	High Order Address Bus
	29	A0	Out	Low Order Address Bus	30	A8	Out	High Order Address Bus
CONTROL BUS	31	WR*	Out	Write to Memory or I/O	32	RD*	Out	Read to Memory or I/O
	33	IORQ*	Out	I/O Address Select	34	MEMRQ*	Out	Memory Address Select
	35	IOEXP*	In/Out	I/O Expansion	36	MEMEX*	In/Out	Memory Expansion
	37	REFRESH*	Out	Refresh Timing	38	MCSYNC*	Out	CPU Machine Cycle Sync
	39	STATUS 1*	Out	CPU Status	40	STATUS 0*	Out	CPU Status
	41	BUSAK*	Out	Bus Acknowledge	42	BUSRQ*	In	Bus Request
	43	INTAK*	Out	Interrupt Acknowledge	44	INTRQ*	In	Interupt Request
	45	WAITRQ*	In	Wait Request	46	NMIRQ*	In	Non-Maskable Interrupt
	47	SYSRESET*	Out	System Reset	48	PBRESET*	In	Push Button Reset
	49	CLOCK*	Out	Clock from Processor	50	CNTRL*	In	AUX Timing
	51	PCO	Out	Priority Chain Out	52	PCI	In	Priority Chain In
POWER BUS	53	AUX GND	In	AUX Ground (Bussed)	54	AUX GND	In	AUX Ground (Bussed)
	55	AUX+V	In	AUX Positive (+12 Volts DC)	56	AUX−V	In	AUX Negative (−12 Volts DC)

*Low Level Active Indicator

MD SERIES

MECHANICAL SPECIFICATIONS

Card Dimensions

11.43 cm wide by 16.51 cm long (4½ by 6½ in)
1.22 cm maximum profile thickness (0.48 in)
0.16 cm printed circuit board thickness (0.062 in)

Operating Temperature

From 0° C to + 50° C

Connectors Table

FUNCTION	CONFIGURATION	MATING CONNECTOR
STD BUS	56 pin dual 0.125 in. centers	Printed Circuit Viking 3 VH28/1 CE5 Wire Wrap Viking 3 VH28/1CND5 Solder Lug Viking 3 VH28/1CN5
Parallel I/O and Serial I/O	26 pin dual 0.100 in. grid	Flat Ribbon Ansley 609-2600 M Discrete Wires Winchester PGB26A (housing) Winchester 100-70020S (contacts)
Analog I/O	40 pin dual 0.100 centers	Ansley 609-4000

SD Series

Z80 Software Development Board (SDB-80E)

HARDWARE FEATURES

- ☐ Available as board or complete system
- ☐ 4K bytes of RAM, expandale on board to 16K Bytes
- ☐ Four 8-bit I/O ports with handshake lines
- ☐ Serial ASCII interface (110-9600 BAUD)
- ☐ Fully buffered for system expandability
- ☐ Four counter/timer channels
- ☐ On board capacity from 5K bytes of PROM to 20K bytes of ROM
- ☐ Double euro-card format

SOFTWARE FEATURES

- ☐ 2K x 8 Operating System in ROM (DDT-80)
- ☐ 8K x 8 assembler/editor in ROM (ASMB-80)
- ☐ Channeled I/O for user convenience
- ☐ Double euro-card format

GENERAL DESCRIPTION

The SDB-80 is a stand-alone microcomputer designed by MOSTEK around the advanced Z80 microprocessor family. It contains more on-board firmware and RAM memory than any previously offered single board microcomputer, plus all the features of the industries most sophisticated microprocessor. This board represents the very latest in state-of-the-art technology by utilizing MOSTEK's new 16K Dynamic RAM memories. The SDB-80 also is the first single board microcomputer to offer a complete package of software development aids in ROM. This 10K byte firmware package is included with the SDB-80 and provides the ability to generate, edit, assemble, load, execute, and debug Z80 programs for all types of applications.

USING THE SDB-80

In addition to functioning as a stand-alone development aid, the SDB-80 is fully expandable through the addition of optional add-on circuit boards.

SYSTEM FIRMWARE

A standard feature of the SDB-80 is a complete package of development software aids which are resident in the five MK 34000, 2k x 8 ROM memories located on the board. This firmware includes a sophisticated operating system, debug package, assembler, and text editor. The presence of this software in ROM provides instant access to these development aids, eliminating the time-consuming requirement of loading the software from some peripheral device into RAM.
Another key feature of having the development aid software in ROM is that entire RAM space is available for the user's programs.

Debug (DDT-80) includes:

- ☐ object program Load/Dump
- ☐ Memory or Port Examine/Change
- ☐ Breakpoint/Execute
- ☐ Logical/Physical I/O mapping (with user expandable drivers)
- ☐ Drivers for Standard Peripherals

The Assembler (ASMB-80) includes:

- ☐ 1, 2 or 3 pass operation
- ☐ conditional Assembly
- ☐ Relocatable object module generation
- ☐ Relocatable linking loader
- ☐ Drivers for Silent 700 Cassette

The Text Editor (EDIT-80) includes:

- ☐ Line or character operation
- ☐ Macro commands

ELECTRICAL SPECIFICATIONS

Operating Temperature Range . . . 0°C to 50°C

Power Supply requirements (Typical)

+12V ± 5%	175 mA
+ 5V ± 5%	1.5A
−12V ± 5%	100 mA

Interface Levels . . . TTL Compatible

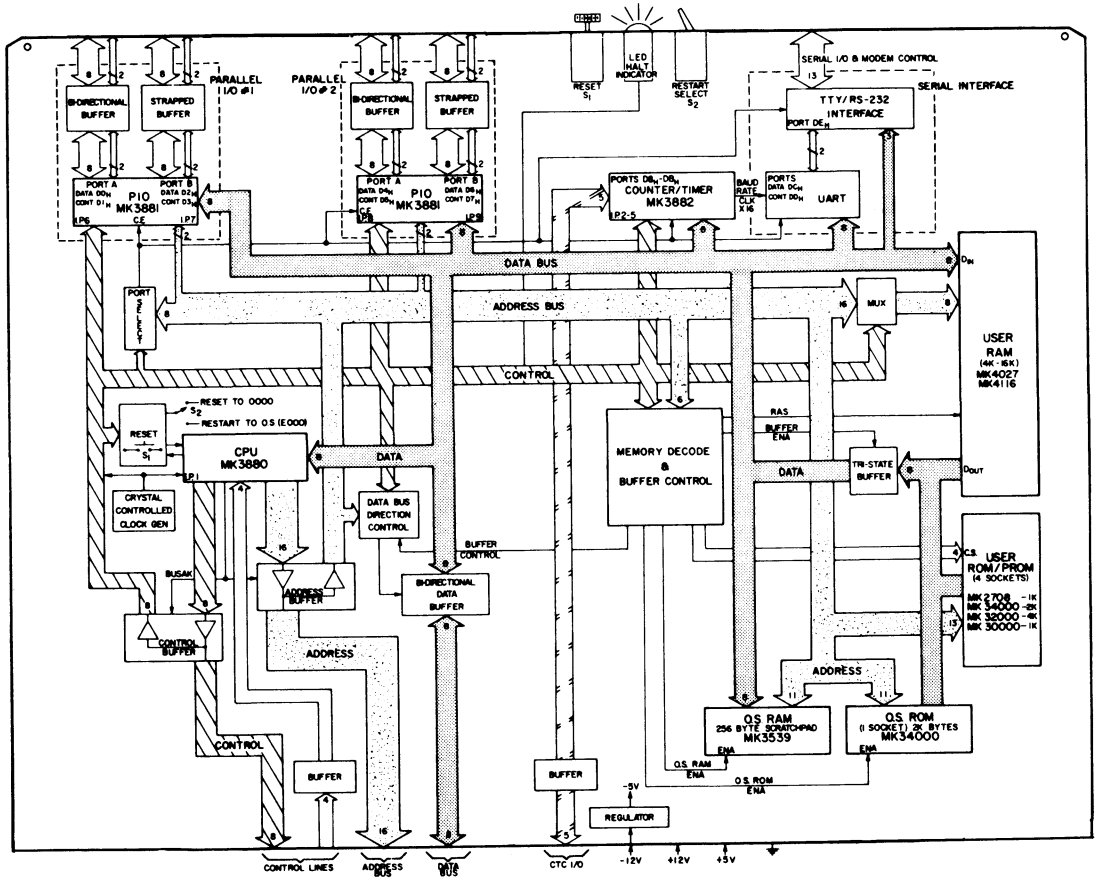
MECHANICAL SPECIFICATIONS

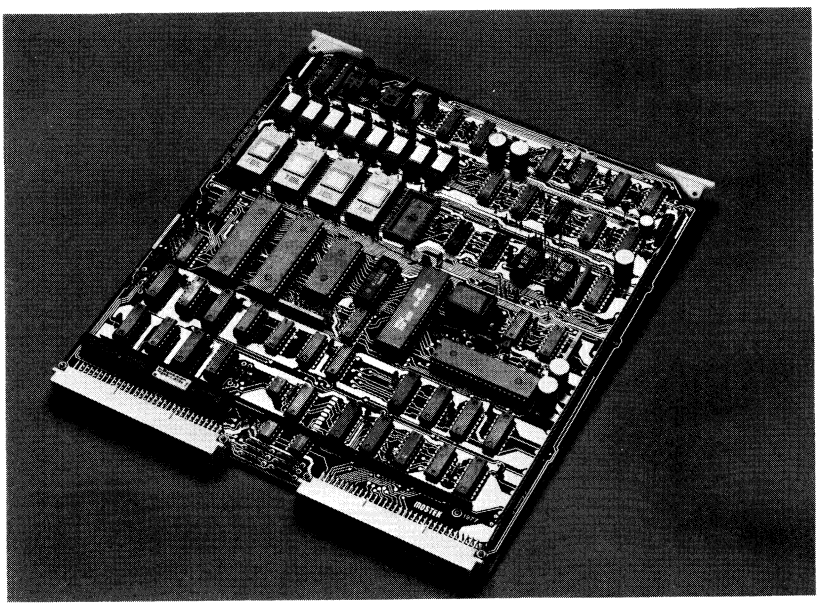
Extended double Eurocard

Board Size: 250 mm x 233.4 mm x 18 mm

Connector: Dual 64 pin Eurocard Connector
DIN 41612 form D; A and C pinned.

SDB-80 FUNCTIONAL BLOCK DIAGRAM





Z80 Single Board Computer (OEM-80)

HARDWARE FEATURES

- ☐ Z80 CPU
- ☐ 20K x 8 EPROM 2708, 2716 or 2532
- ☐ 256 x 8 static scratchpad RAM for debug
- ☐ 16K (4K) x 8 dynamic RAM
- ☐ Z80 CTC — 4 counter/timer channels
- ☐ Restart to 0000H or E000H (switch option)
- ☐ On-board serial I/O port
- ☐ Software programmable baud rate
- ☐ Current loop or V24 interface
- ☐ 4 8-bit parallel ports with handshake
- ☐ Ports buffered with TTL, socket programmed Direction in 4-bit blocks
- ☐ Programmable polarity on strobe lines
- ☐ Halt lamp
- ☐ Fully debounced reset switch
- ☐ Power on reset logic
- ☐ On-board regulator for -5 volt PROMS/RAMS
- ☐ Fully hysteresis buffered SD bus
- ☐ Noise reducing bus control technique
- ☐ Double eurocard format
- ☐ High reliability DIN 41612 connectors
- ☐ All signals on one card edge
- ☐ Separate CPU bus and I/O Connectors

OEM-80 DESCRIPTION

The OEM-80 is the start of an SD board system. The card has sufficient on-board I/O and memory to be used in a stand-alone mode in many applications, but it is fully expandable to support more memory and I/O in applications requiring it. The on-board 5 EPROM sockets can be strapped to use a number of standard 24 pin ROM/EPROM products including PROMs and ROMs with capacities to 4K bytes each. The 8 RAM sockets can be strapped for 4K or 16K RAMs, giving up to 16K bytes on board. The Z80 built-in refresh logic reduces the area taken by the dynamic RAMs to that required by other manufacturers 1K byte static RAMs, the cost per bit being of course significantly reduced. Also on board are two memory decoding bipolar PROMs. These allow a wide range of RAM/PROM/ROM combinations to be selected by the user, if the exact combination required is not already supported, new PROMs can be easily blown.

The user switch selectable restart address allows the DDT-80 debug program to be resident in the system without conflict with the users own PROM based software. If a problem develops the user can switch from address 0 reset to address E000 (where DDT-80 resides) and use the powerful commands of the 2K byte DDT-80 to localize the problem.

The double eurocard format and DIN connectors allow quick integration into users system hardware. Putting all connectors on one card edge, a unique feature in microcomputer modules, is standard design practice for many large systems builders. The simplifed maintenance and clean cabling made possible by this technique should be appreciated by all experienced users.

MEMORY ADDRESSING AND CAPACITY

The recommended memory map is shown below:

0000-3FFF	PROM (1 TO 16K)
4000-7FFF	RAM (4 TO 16K)
8000-DFFF	EXTERNAL MEMORY
E000-E7FF	DDT-80 (2K)
E800-FEFF	EXTERNAL MEMORY
FF00-FFFF	SCRATCHPAD RAM (256) only needed if DDT-80 is used

Memory cycles times required for the PROMs is 450 ns.

BUS INTERFACE

All Z80 signals are buffered before leaving the OEM-80. This protects the MOS components from static charge during handling and bus transients which could otherwise destroy these devices. The bus supports DMA transfers and the daisy chained multi-level interrupt structure of the Z80. The bus uses exclusively hysteresis input receivers and current limited bus drivers to improve the noise margin of the bus. Switching the bus drivers on only when data is needed and stable further reduces the noise on the bus. This is not done on many other microcomputer busses.

INTERRUPTS

The OEM-80 has 9 on board interrupts, they are:

Z80 CPU	1-NMI (non maskable interrupt)
Z80 PIO(2)	4
Z80 CTC	4

More interrupt devices (up to 128 total) can be added to the SD bus.

COUNTER/TIMER CHANNELS

Four counter timer channels are provided on the card in a Z80-CTC chip. One channel is used for the baud rate generator of the serial I/O port. The other 3 channels are available to the user. These may be programmed as delay generators, event counters or simply discrete interrupt inputs with programmable edge trigger. The device can generate four interrupts.

SERIAL I/O PORTS

A UART with 20mA. current loop and V24 buffer/drivers provides a serial communication channel for interfacing to TTY or CRT terminals or serial printers. The baud rate is software programmable over the range of 50 to 9600 baud.

PARALLEL I/O PORTS

The 4 parallel ports are designed to allow maximum flexibility in matching the MOS I/O ports to the real world of long lines or high voltages. Two ports support bidirectional TTL I/O with hysteresis inputs. They can also be programmed for input or output only. The other ports are supplied with sockets which support a number of standard TTL devices for buffering.

TYPE	USE
7400	16MA TTL inverting output
7402	TTL inverting input
7408	16MA TTL non inverting output
7426	16MA high voltage inverting open collector output
7437	48MA TTL inverting output
7438	48MA TTL inverting open collector output

POWER REQUIREMENTS

5V	5%	at	1.5 A
+12V	5%	at	.175 A
−12V	5%	at	.1 A

Temperature range: 0 to 50 deg C.

MECHANICAL SPECIFICATIONS

Board size: 233.4 × 250 mm.
Connectors: 2-64 pin DIN 41612 (a-c) indirect, male

PRELIMINARY

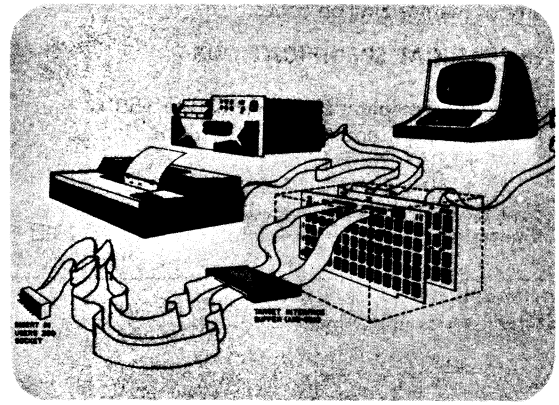
MOSTEK®

Z80 MICROCOMPUTER SYSTEMS

Random Access Memory Board (RAM-80E)

FEATURES

- Memory Capacity
  - RAM-80AE – 16,384 (16K) bytes using MK4027 RAM's
  - RAM-80BE –16,384 (16K) bytes expandable to 65,536 (65K) bytes using MK4116 RAM's
  - RAM-80BE under page mode operation – up to 1 megabyte of memory
- I/O Capacity (RAM-80BE only)
  Four 8-bit ports with handshake lines
- Memory Access Time – 345ns (maximum)
- Memory Cycle Time – 450ns (minimum)



GENERAL DESCRIPTION

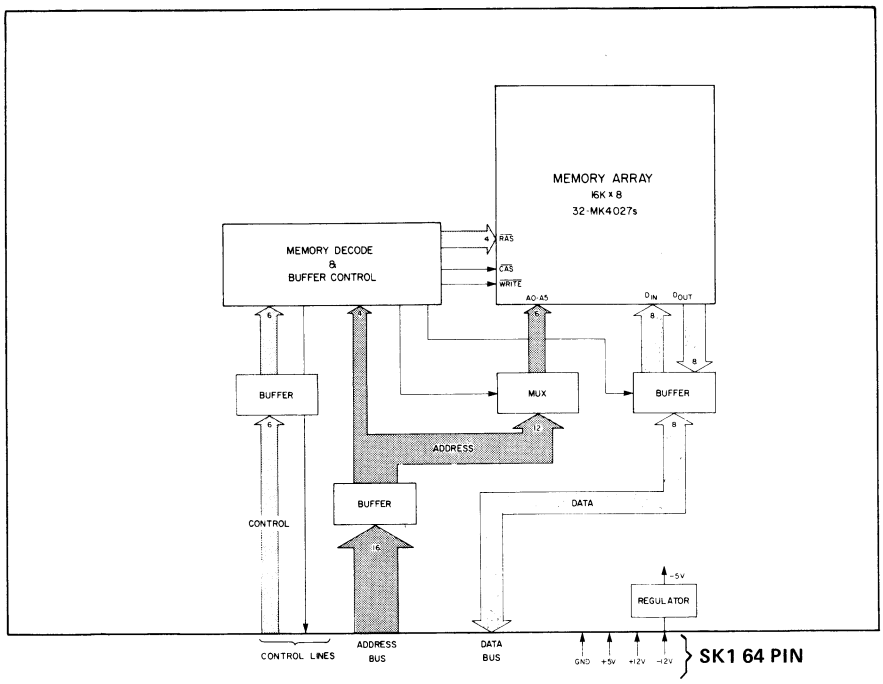
The RAM-80E is designed to provide RAM expansion capability for the Z80 based SDB-80E Microcomputer. For user flexibility, it is offered in two basic configurations designated RAM-80AE and RAM-80BE.

The RAM-80AE is the basic 16K byte RAM board for users requiring the most economical means for adding RAM to an SDB-80E Microcomputer. It is designed using the high performance MK4027-4, 4096 x 1 bit dynamic RAM, and includes address strapping options for positioning the decoded memory space to start on any 4K incremental address boundary.

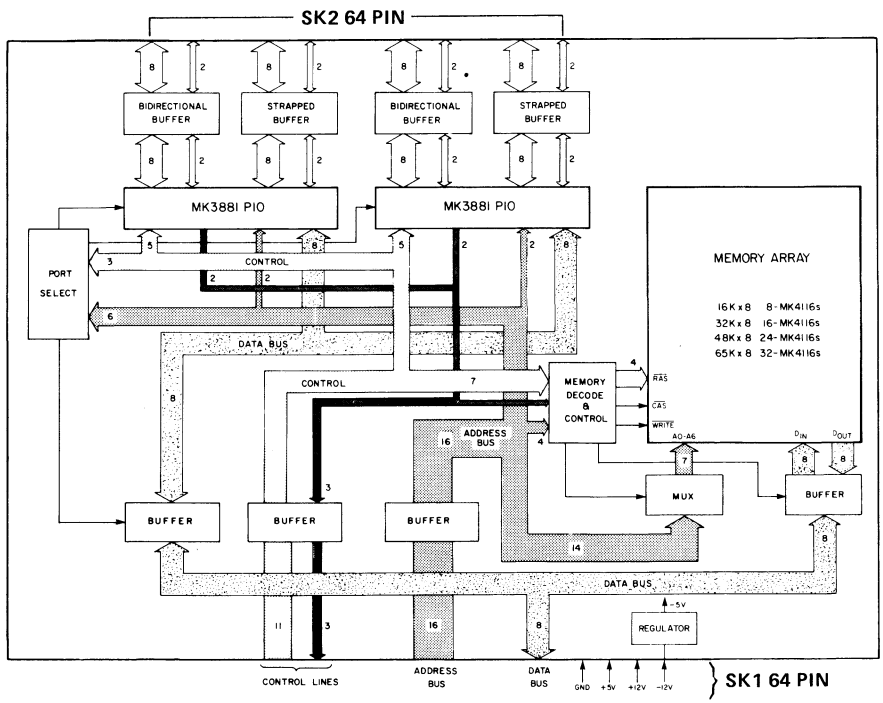
The RAM-80BE is a combination memory and I/O expansion board. The memory may be configured to have a memory capacity of 16K, 32K, 48K, or 65K bytes of RAM. This on-board memory expandability is made possible by population options of either eight, sixteen, twenty-four or thirty-two MK4116-4 (16,384 x 1 MOS dynamic RAM) memories. The RAM-80BE provides strapping options for positioning the decoded memory space to start on any 16K address boundary. In addition to the add-on memory, the RAM-80BE provides four 8-bit I/O ports from the two on-board MK3881 Z80 PIO circuits. Each I/O port is fully TTL buffered and has two handshake lines per I/O port. The RAM-80BE also includes logic for a "Page Mode Operation" which permits up to 1 megabyte of memory (sixteen 65K x 8 RAM-80B's) to be used in a single SDB-80E system.

A complete set of documentation for each RAM-80E board is available to ensure easy utilization.

RAM-80AE FUNCTIONAL DIAGRAM



RAM-80BE FUNCTIONAL DIAGRAM



ELECTRICAL SPECIFICATIONS

Memory Access Time—345 ns (maximum)
Memory Cycle Time—450 ns (minimum)
Operating Temperature: 0°C to 50°C
Power Supply Requirements

Voltage	RAM-80AE	RAM-80BE
+12V±5%	200 mA typ.	200 mA typ.
	575 mA max	575 mA max
−12V±5%	25 mA typ	25 mA typ
	30 mA max.	30 mA max.
+5V±5%	370 mA typ.	1.1A typ.
	550 mA max.	1.5A max.

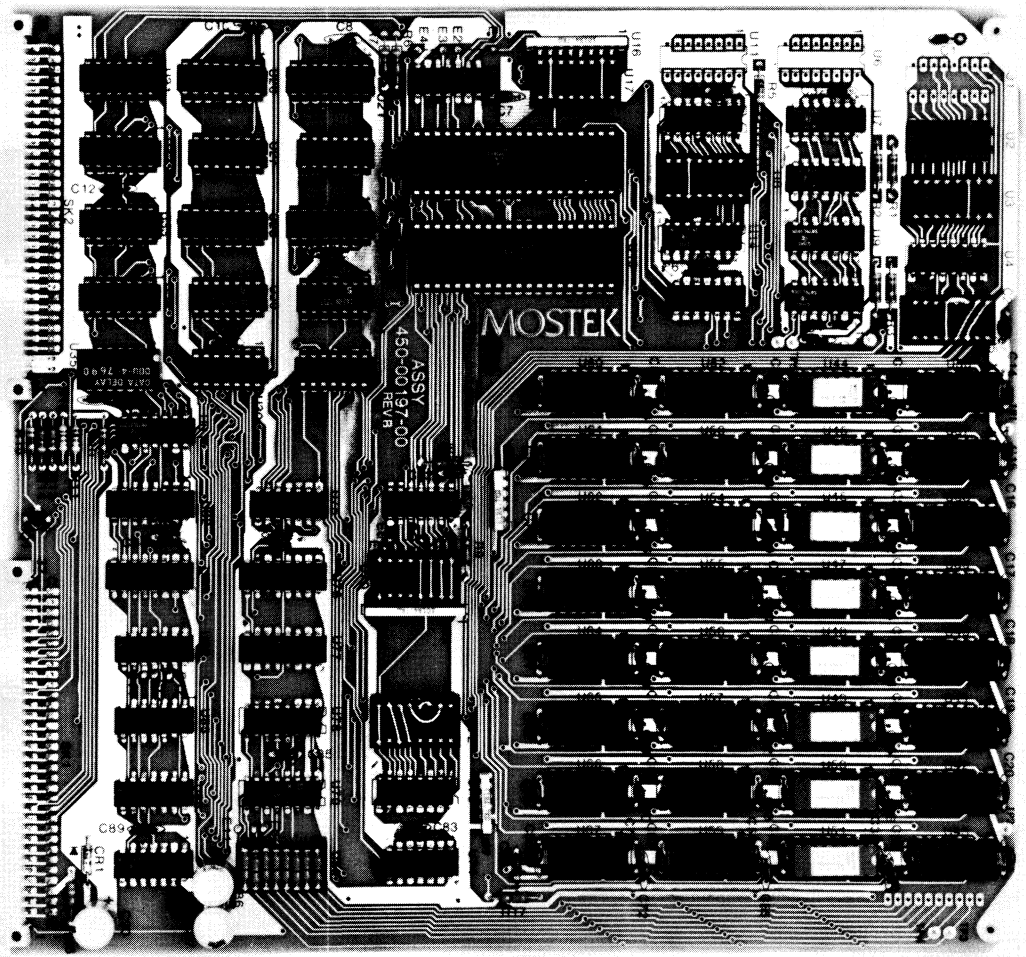
MECHANICAL SPECIFICATIONS

U.S. Version

Board Size: 8.5" x 12.0" x 0.65"
Bottom Connector: 100 pin, 125 mil centers
Top Parallel Connectors (RAM-80B): 50 pin, 100 mil centers

Double Eurocard Version

Board Size: 250 mm x 233.4 mm x 18 mm
Connector: Dual 64 pin Eurocard Connector



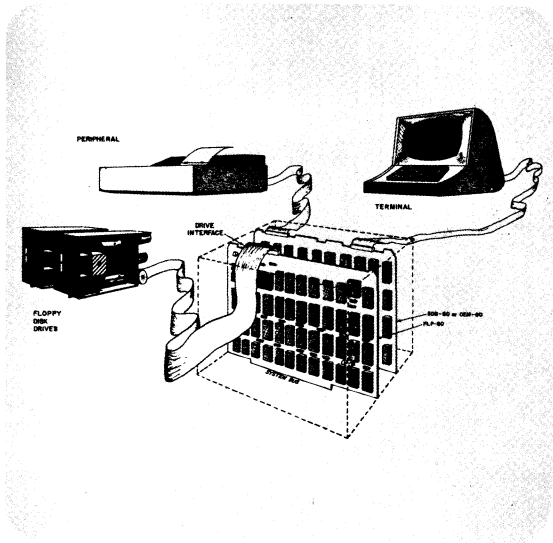
MOSTEK®

Z80 MICROCOMPUTER SYSTEMS

Flexible Disk Drive Controller (FLP-80)

HARDWARE FEATURES

- ☐ Soft sector format compatible with IBM 3740 data entry system format.
- ☐ Capable of controlling up to four flexible disk drives per subsystem.
- ☐ Double sided drive capability.
- ☐ Full disk initialization (Formatting).
- ☐ Full sector (128 bytes) FIFO buffering for data.
- ☐ Double buffering for control and status.
- ☐ Automatic track seek with verification.
- ☐ Completely interruptable for real time systems.



APPLICATIONS

- ☐ Flexible disk drive interface for use with MOSTEK's Software Development Board (SDB-80) in a disk based Z80 Development System SYS-80FT
- ☐ Single or multiple flexible disk drive controller/formatter for disk based OEM systems using the OEM-80 Single Board Computer.

GENERAL DESCRIPTION

The FLP-80 is an add-on flexible disk controller used to interface up to four flexible disk drives to the MOSTEK Software Development Board (SDB-80).

The FLP-80 provides the necessary electronics to accomplish track selection, head loading, data transfer, error detection, flexible drive interface, status reporting and format generation/recognition. The FLP-80 is designed to operate with either Shugart SA-800 Single Sided or SA-850 Double Sided Flexible Disk Drives. In addition to functioning as an add-on card to the SDB-80 system, the FLP-80 may be utilized directly in OEM applications to control/format up to four flexible disk drives of either single or dual sided type in 8080A or Z80 systems.

AVAILABLE SOFTWARE

Software for the FLP-80 Disk controller is the MOSTEK Disk Operating System (FLP-80DOS). A user can easily design his own OEM software package around 20 powerful disk operating system commands permitting complex record insertion, deletion, and position manipulation. Other software includes application packages such as an advanced monitor and debugger, disk-based Text Editor, Z80 Assembler, Relocating Linking Loader, Peripheral Interchange Program, and a channelized I/O system for each peripheral interface. These programs provide state-of-the-art software for developing Z80 programs as well as establishing a firm basis for OEM products. Further Information is provided on Page 83 and following pages.

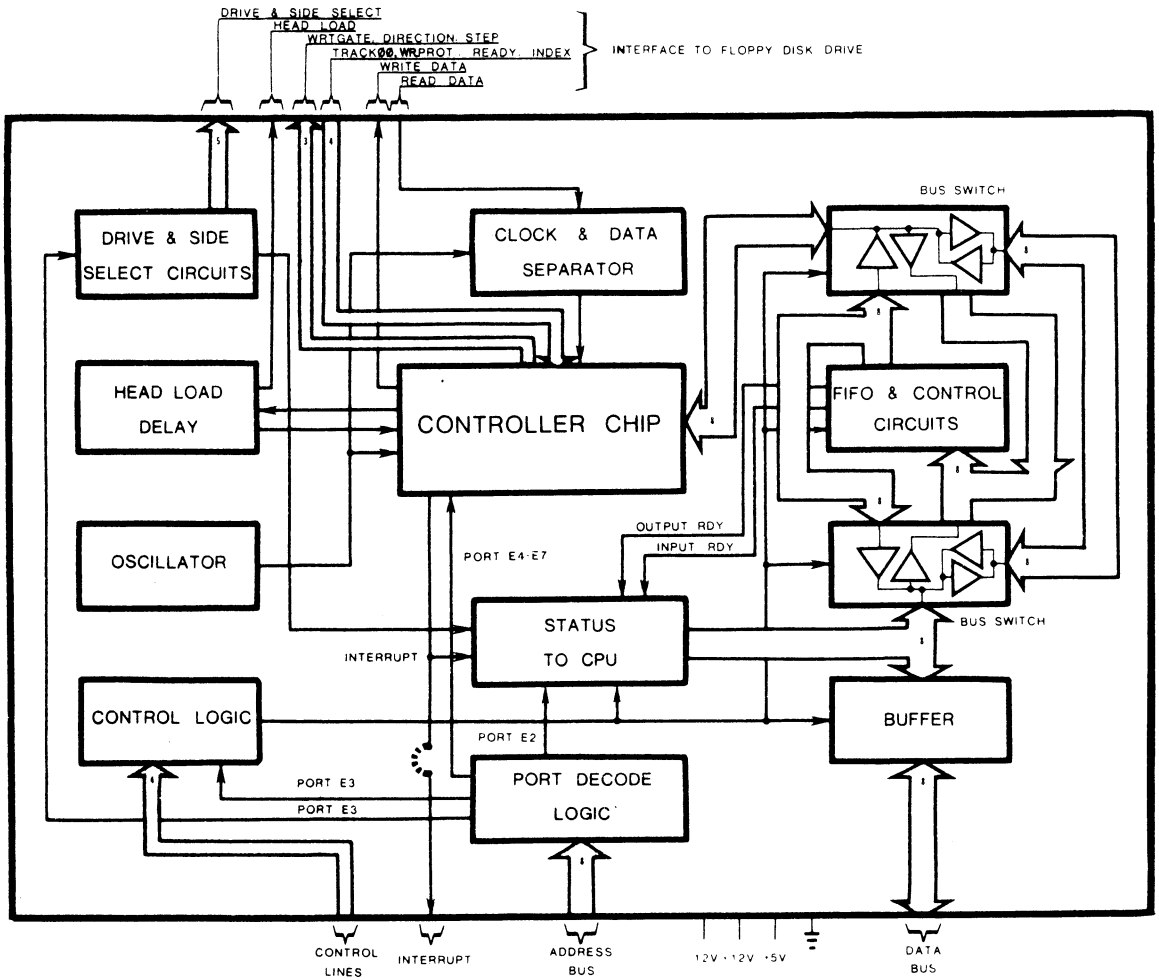
ELECTRICAL SPECIFICATIONS

Operating Temperature Range - 0°C to 50°C
Power Supply Requirements (Typical)
+12V ± 5% @ .006A
+5V ± 5% @ 1.1A
-12V ± 5% @ .03A
Interface Levels - TTL Compatible

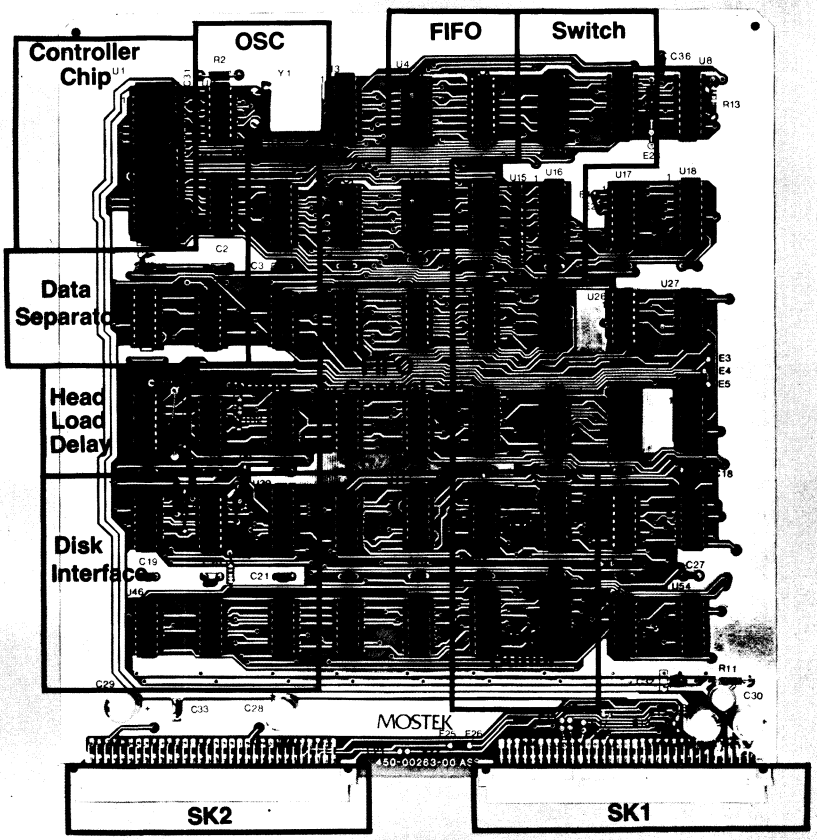
MECHANICAL EUROCARD

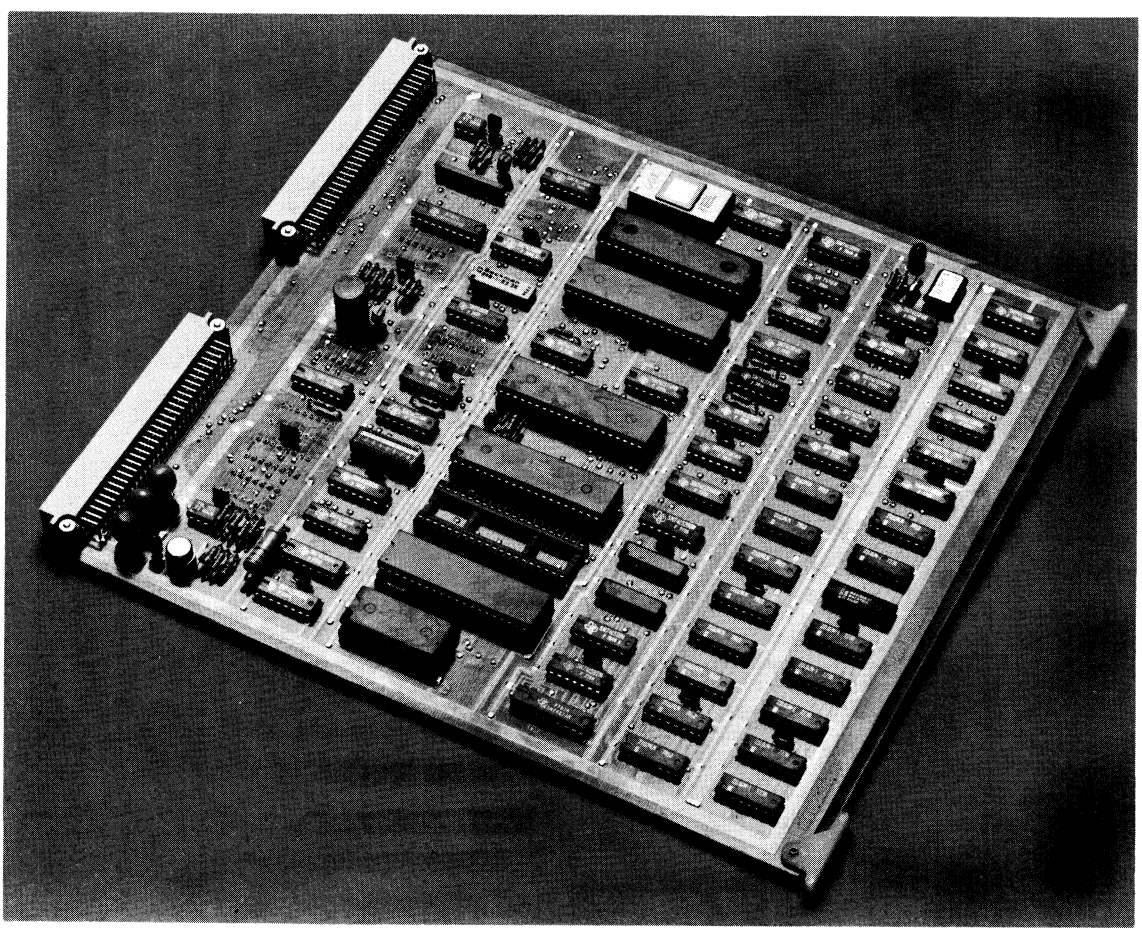
Board Size:250mm x 233.4mm x 18mm
Bottom Connector: Dual 64 pin Eurocard Connector DIN 41612 form D; A and C pinned.

FLP-80E BLOCK DIAGRAM



FLP-80E BOARD





Video Display Interface

FEATURES

- ☐ Complete video/keyboard interface
- ☐ 24 line x 80 character display
- ☐ Inverse video by character programmable
- ☐ 5 x 8 dot matrix
- ☐ Serial UART port : 110-9600 baud
- ☐ Parallel port : up to 3200 characters/second
- ☐ Upper and lower case display : 96 character ASCII +32 special characters
- ☐ Auto repeat on keyboard interface
- ☐ Direct cursor addressing and bidirectional scrolling
- ☐ 48 character FIFO
- ☐ 50/60 Hz operation

DESCRIPTION

The VDI card is a self-contained interface for interconnection of a video monitor and an ASCII keyboard to a computer over a serial link. Both 20 mA. current loop and RS232 (V24) compatible voltage loop interconnection techniques are supported.

The card can be used as a teletype replacement or as an intelligent terminal in many applications. The interface is constructed on a standard width double Eurocard. The power connections (and bus) are directly compatible to the MOSTEK standard SD series bus.

The present cursor position is indicated by a blinking white rectangle which inverts any data that may be covered by the cursor. Besides the normal up/down/left/right cursor motions, the cursor may be directly addressed and positioned anywhere on the screen. The terminal may also be used in local mode; screen data can be read back by a computer.

The card supports up to 80 user-defined tab locations. The 48 character FIFO allows continuous high speed data transmission even when time-consuming commands are given; for example, clear screen.

SPECIFICATIONS

Operating temperature 0 to 50°C

Power supply requirements
+ 5V @ 2.0 A
+12V @ 0.1 A
-5V (for use in optional RS232 or 2708 PROM)

Board format	:	233.4 x 250 mm (9.19 x 9.84 inches)
	:	DIN 41612 connectors
	:	extended double Eurocard
Video output	:	RS170 1V into 75 Ohm
	:	525 line 60Hz or 625 line 50Hz
Keyboard inputs	:	standard TTL, ASCII encoded
	:	active high or low strobe 300 microseconds minimum pulse.
	:	active high data
Serial link	:	110,300,600,1200,2400, 4800,9600 baud generator on board
	:	external baud rate may be used (max 19200 baud)
	:	programmable 1/2 stops bits, even/odd/no parity
	:	completely opto isolated current loop (20 mA.)
Parallel interface	:	SD bus compatable with on board Z80 PIO
	:	3200 char per second transfer rate
	:	2 interrupts
	:	64 possible port addresses
Character generator	:	MK34073 2K x 8 ROM
	:	customer defined 2708/58 PROM can also be used
Processing time	:	typical (character data) 320 microseconds.
	:	worst case (clear display) 9.33 ms.
Bell output	:	direct drive to 50 Ohm speaker

POWER ON CONDITION

At power on, the following conditions are set: home, clear screen, line, all tabs clear and normal video.

COMMAND SET

Shift means the shift key must also be depressed on most keyboards. Esc means that the command must be preceded by the escape character

HEX	CONTROL	FUNCTION	COMMENT
07	G	BELL	300 ms. 700Hz tone
08	H	BACKSPACE	Cursor moves left once unless at left margin
09	I	TAB	Cursor moves to next tab location
0A	J	LINE FEED	Cursor moves down one line. If at bottom then display scrolls up one line
0B	K	VT	Cursor moves up one line, if at top, then display scrolls down one line
0C	L	FF	Cursor moves right once unless at right margin
0D	M	RETURN	Cursor moves to left margin.
0E	N	RESET TAB	Tab is cleared from this position
0F	O	SET TAB	Tab is set at this position
10	P	DOWNSHIFT	Following character is displayed as special non ASCII character
1B	SHIFT-K	ESCAPE	Enables escape sequence, following character is the command.
3D	ESC =	MOVE CURSOR	Command to move cursor. Next two characters are binary y and x addresses of new cursor position. Lower left is 0 : 0
2B	ESC +	MOVE CURSOR	Command to move cursor. Next two characters are binary y and x two's complement offsets from current position.
3F	ESC ?	READ CURSOR	Read back cursor address. VDI sends STX then y,x address of cursor. 30 hex is added to avoid control character conflicts.
7D	ESC †	LOCAL	Sets VDI to local mode. Keyboard data is sent directly to the display without being output to the serial/parallel port.
29	ESC)	READ ONE	Read back to the port the contents of the cursor postion. Cursor does FF.
3C	ESC <	READ LINE	Line from cursor to last space is sent to the port. Data is preceded by a STX and followed by a RETURN and ETX. Cursor does not move.
3E	ESC >	READ PAGE	Entire page from cursor to end is sent to the port. Data is preceded by a STX, trailing spaces are deleted, and lines are separated by returns. At the end an ETX is sent.
17	W	INVERT	All following characters will be written in inverted video. (black on white)
19	Y	NORMAL	All following characters will be written in normal video. (white on black)
1C	®	LINE	Set VDI to line mode, send ACK to port. All following keyboard inputs go directly to the port.
1D	]	CLEAR SCREEN	Clear display from cursor to end
1E	©	HOME	Move cursor to upper left corner
1F	—	CLEAR LINE	Clear line from cursor to end
7F		DEL	Nop

OPTIONS

All options including port address, baud rate, parity, stop bits, 50/60 Hz and keyboard strobe active edge are programmable on connector #2.

SOFTWARE SUPPORT

A screen editor for the Z80 which uses the VDI card as a "window" into a Z80 system memory is also available from MOSTEK.

Analog/Digital Converter (A/D-80)

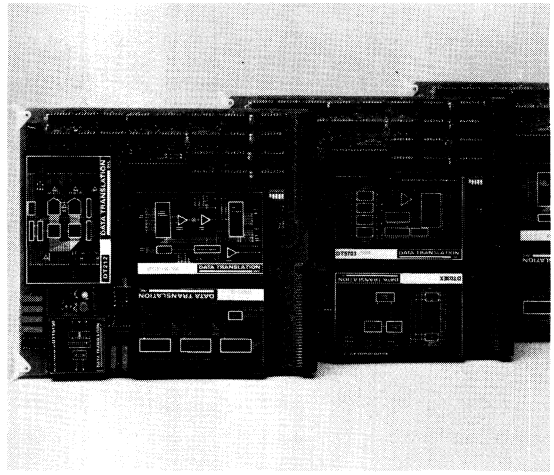
FEATURES

- ☐ Complete line of Analog I/O Systems for Mostek's SD Series
- ☐ Three Analog input versions available:
  - A/D-80/1792 - High-level (±5V, ±10V, 0-10V, 0-5V)
    - Choice of 16 SE, 8 DI, 64 SE, or 32 DI channels
    - 12 bit A/D converter
    - Programmable gain option
    - Current loop input option
  - A/D-80/1794 - Low-level, wide range (±10mV to ± 10V)
    - Choice of 16SE, 8DI, 64SE, or 32DI channels
    - 12 bit A/D converter
  - A/D-80/1798 - Wide-range, isolated inputs (± 10mV to ±10V)
    - 4 DI or 12 DI channels
    - Programmable gain option
    - Current loop input option.
- ☐ Three Analog I/O versions available:
  - A/D-80/1791 - Same analog input features as 1792
    - 2 D/A output channels with scope control
  - A/D-80/1795 - Same analog input features as 1794
    - 2 D/A output channels with scope control
  - A/D-80/1799 - Same analog input features as 1798
    - 2 D/A output channels with scope control
- ☐ One Analog Output Version available
  - A/D-80/1796 - 2 D/A output channels with scope control.

GENERAL DESCRIPTION

The System Design Series (SD Series™) of OEM microcomputer boards offers powerful features and versatility to the OEM. Utilizing the MOSTEK Z80 and MOSTEK's industry-standard memories, the SD Series enables the user to construct high-performance, memory-intensive systems for a wide range of application.

The A/D-80 line of analog I/O systems is offered as a part of the SD Series. Available in seven different versions with various user-specified options, the A/D-80 can be configured with the right combination of analog I/O to meet the system designer's needs.



MOSTEK offers three types of low-cost analog input systems for three different input voltage ranges. The 1792, 1794, and 1798 handle the input voltage ranges as shown under "SPECIFICATIONS". The user can order up to 64 single-ended analog input channels on either the A/D-80/1792 or the A/D-80/1798. In addition, a programmable gain option and current loop input option is available on the A/D-80/1792 and A/D-80/1798.

The A/D-80/1791, A/D-80/1795, and A/D-80/1798, have the same A/D input features as the A/D-80/1792, A/D-80/1794, and A/D-80/1798, respectively, with the addition of two D/A output channels with scope control. These three boards are complete analog I/O systems.

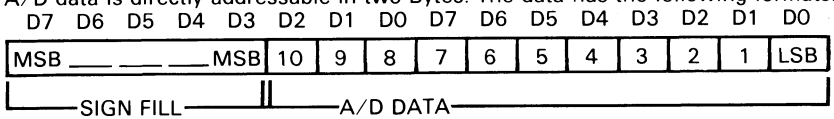
The A/D-80/1796 is an analog output-only system featuring two D/A output channels with scope control.

The interface to the A/D-80 contains all the control circuitry for Z axis and scope control mode bits.

The A/D-80 line of analog I/O boards has been designed using DATAX-II™ data acquisition modules manufactured by Data Translation, Inc. Each DATAX module is a complete self-contained unit with multiple shielding for operation in a microprocessor system. This eliminates ground-loop and noise problems inherent in interconnection of separate modules.

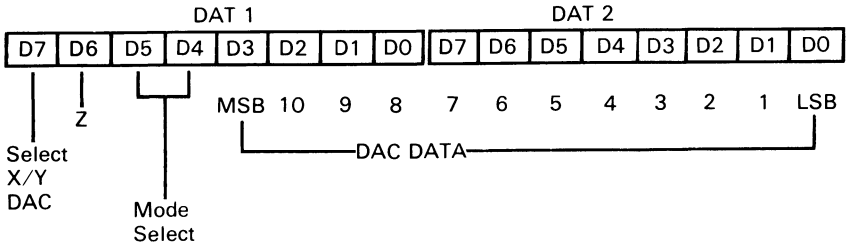
A/D DATA

A/D data is directly addressable in two Bytes. The data has the following formats:



D/A CONVERTER DATA FORMAT

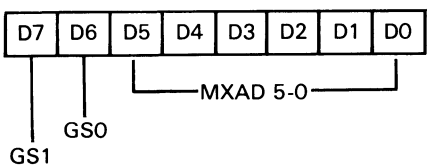
D/A data can be loaded directly in two bytes. DAT 1 must be loaded prior to DAT 2 so that the D/A will not glitch on the output. Double buffering is provided on the DAT 1 data.



The following is an explanation of the various D/A control BITS.

BIT	DESCRIPTION
X/Y select	When set = Y DAC When reset = X DAC
Z	When set will generate a Z output upon loading DAC DATA BYTE 2
MODE	Two Mode Select Bits provide 4 Mode outputs

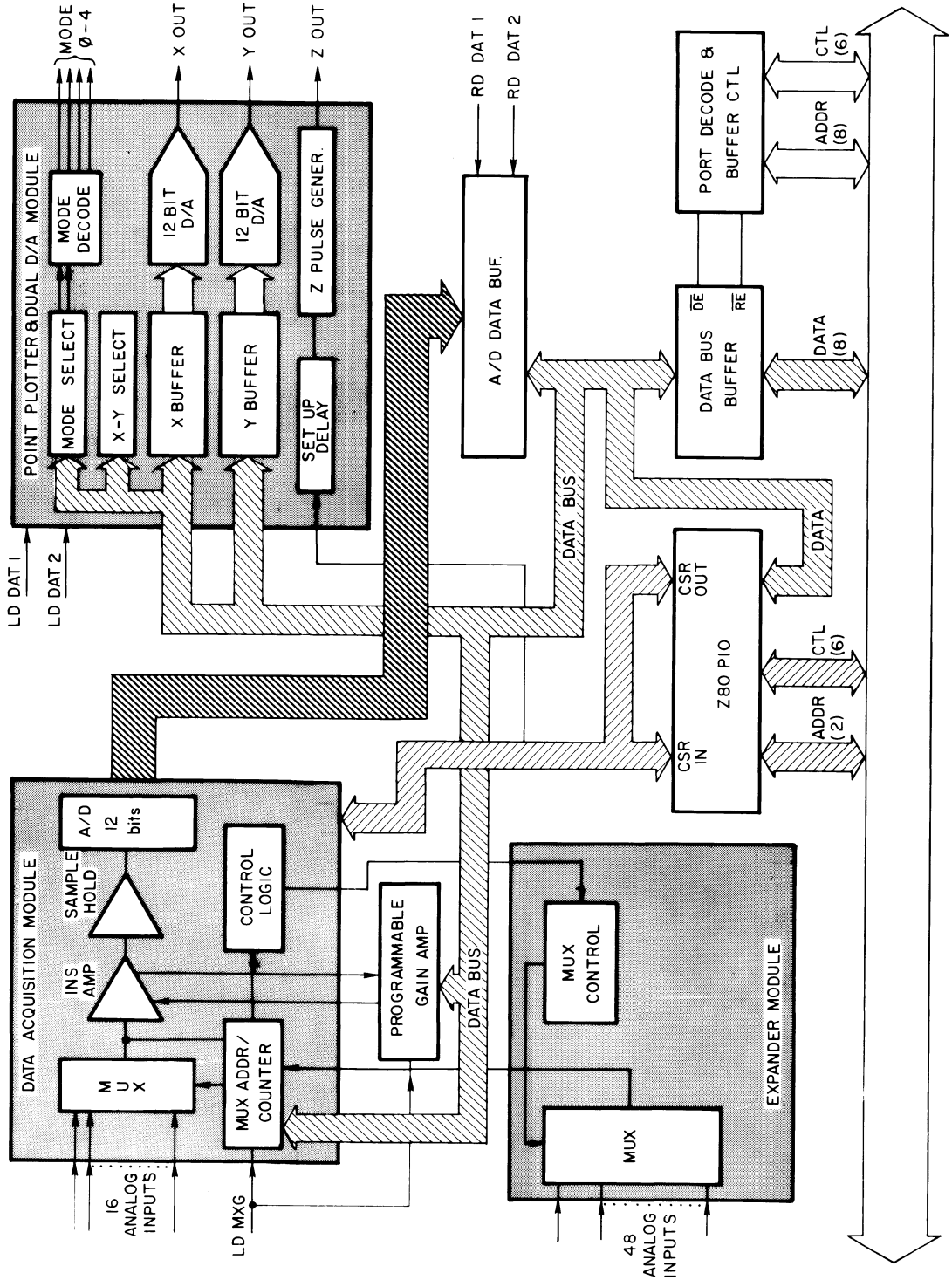
MULTIPLEXER/GAIN REGISTER
MXR (Multiplexer & Gain Register)



The MUX-GAIN Register is directly addressable and contains this information as follows:

BIT	NAME	DESCRIPTION
7-6	GAIN SELECT	SET/RESET via program control BIT 7 / BIT 6 / GAIN 1791 / GAIN 1798 0 / 0 / 1 / 1 0 / 1 / 2 / 10 1 / 0 / 4 / 100 1 / 1 / 8 / 500
5-0	MUX ADDRESS	Select 1 of 64 multiplier address SET/RESET by program control.

A/D BLOCK DIAGRAM



A/D-80
ANALOG INPUT
SPECIFICATION

	A/D-80/1791 & 1792	A/D-80/1794 & 1795	A/D-80/1798 & 1799
Number of channels	Up to 64 single ended or 32 differential	*	4 differential or 12 differential
Input Impedance	100 megOhm	*	10 megOhm
Input Overvoltage	±35V non-destructive	±15V non-destructive	15V DC max.
Input Range	0-5V, ±5V, 0-10V, ±10V All jumper selectable	0-10mV, 0-10V, ±10mV, ±10V selectable via a single register	0-10V unipolar, ±10V bipolar
Optional programmable gain amplifier	gains: 1, 2, 4, 8	Not Available	gains: 1, 10, 100, 500
Conversion resolution	12 bits	*	*
Linearity	±½LSB	±½LSB	±½LSB
Inherent quantizing error	±½LSB	*	*
Stability Tempco	± 25ppm/°C, F.S.R.	± 30ppm/°C	Zero - ± 20 microVolt/°C Full Scale - ± 30ppm/°C
Throughput	35KHz stand. 100KHz optional	31KHz	Random mode: 20 conversions/sec Sequential mode: 40 conversions/sec
Power Requirements	+5V @ 2.0 A Max	*	*
Mechanical printed circuit board	233.40mm x 257.62mm x 18mm	*	*
Temperature	0° - 50°C	*	*
Implementation	Programmed I/O and Interrupt Functions	*	*
Device address	Selectable via jumper	*	*
			* - Same as 1791 ** - Same as 1795

Analog Output Specifications

Resolution - 12 Bits

Linearity - ±1/2 LSB

Range - ±10V, 0 - 10 V; @ 25 mA minimum current output, all jumper selectable.

Relative Accuracy - ±0.025%

Full Scale Settling - 0.1% - 1 microsecond, 0.01% -3 microsecond into 50ft, coaxial cable terminated with 470 Ohm

Temperature Coefficient - 25 ppm/°C

Z Axis Control - The Interface contains all the control circuitry of Z axis and scope control mode bits.

Z Output (Intensity) - LO (0.8V) to HI (2.4V) TTL compatible into 50 Ohm termination

Z Risetime - 100 nsec into 50 ft. of terminated COAX

Z Pulse Width - Jumper Selectable
a. 0.5 microsecond
b. 5 microsecond
c. external RC 1 microsecond to 0.5 msec

A/D-80 INTERFACE

The Z80-PIO chip and some external logic are utilized to provide the interface for the A/D-80. In this manner, the Z80-PIO chip is used to provide all the interrupt circuitry for Z80- Mode 2 operation, i.e., a vectored daisy chain priority interrupt structure.

The I/O addresses are jumper selectable in groups of 8 addresses anywhere in the I/O address space of the Z80. Once bits 7 thru 3 are assigned the low order bits 2,1, and 0 have the following assignments.

Analog I/O address assignments

7	6	5	4	3	2	1	0

Assigned by jumper selection

Z 1 0

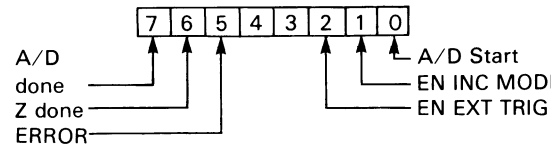
0 0 0 - PIO CHIP PORT A CSR IN
0 0 1 - PIO CHIP PORT B CSR OUT
0 1 0 - PIO CHIP PORT A CONTROL
0 1 1 - PIO CHIP PORT B CONTROL
1 0 0 - DAT 1 I/O (A/D INPUT DATA - D/A OUTPUT DATA)
1 0 1 - DAT 2 I/O (A/D INPUT DATA - D/A OUTPUT DATA)
1 1 0 - MXG (MUX - GAIN REGISTER)
1 1 1 - Not used

*CSR = Control and Status Register

PROGRAMMING

The following is a description of the I/O ports used in programming the A/D-80.

CONTROL & STATUS REGISTER (CSR)



Port A and Port B of the PIO chip are utilized to implement the CSR function. Port A of the PIO chip in Mode 3 operation is utilized for the input of the CSR and Port B of the PIO chip in Mode 0 is utilized for the output function to the CSR.

The control and status register provides all the control of the interface as follows:

BIT	NAME	DESCRIPTION
7	A/D Done	Set by A/D data ready, reset by read A/D Data Byte 2. This bit is READ only, and is reset by initialization.
6	Z Done	Set by the trailing edge of the Z output pulse, reset by program control.
5	ERROR	This bit is used when external triggers are utilized to start the A/D conversion. It will be set with the following conditions, if EXT start is received. 1. During MUX settling time 2. During A/D conversion 3. Before A/D DATA has been READ. The bit can be reset by program control and is reset by initialize.
4-3	UNUSED	
2	EN EXT TRIG	When set, this bit will enable the user of the external trigger inputs to start A/D conversions. Reset by initialize, controlled via program.
1	EN INC MD	When set via program control this bit allows the A/D multiplex to run in increment mode as follows: 1. When A/D start is set (Bit O CSR) the A/D will increment to the next sequential channel and start a conversion on that channel. 2. If external trigger enable is set, when any external trigger is received the channel will be incremented and a conversion started on that channel. Reset by program control and initialize.
0	A/D START	When set by program control this bit will start the A/D converter. It is WRITE only, and always reads as 0.

A/D PART NUMBERS, OPTIONS, & PRICES

ORDER INFORMATION			
PART NO.	OPTIONS	DESIGNATION	DESCRIPTION
78172-xx	A B C D	A/D-80/1791	High level analog I/O board
78173-xx	A B C D	A/D-80/1792	High level analog aboard
78174-xx	A X X D	A/D-80/1794	Low level, nonisolated analog input board
78175-xx	A X X D	A/D-80/1795	Low level, non-isolated analog I/O board
78176-xx	X X X X	D/A-80/1796	Analog output board
78177-xx	*A B C D	A/D-80/1798	Wide range, isolated analog board
78178-xx	*A B C D	A/D-80/1799	Wide range, isolated analog I/O board

*Available with 4 or 12 differential input only

OPTION — EXPLANATION

A
- 40 = 16 single ended inputs
- 60 = 64 single ended inputs
- 10 = **4 differential inputs
- 30 = 8 differential inputs
- 30 = **12 differential inputs
- 50 = 32 differential inputs

**Available on 78177 and 78178 only

B
- 0 = No programmable gain
- 2 = Programmable gain

OPTION — EXPLANATION

C***
- 0 = No current loop input
- 1 = Current loop input

D
- 0 = Standard 35KHz throughput
- 4 = Optional 100KHz throughput

X = Not available option

***Supplied in differential configuration only

-xx = A + B + C + D

SAMPLE

MK 78173 - 53 = 32 differential channels, programmable gain, current loop, 35KHz throughput

MOSTEK®

Z80 MICROCOMPUTER SYSTEMS

SYS-80F Flexible Disk Operating System (FLP-80DOS)

INTRODUCTION

The MOSTEK FLP-80DOS software package is designed for the MOSTEK dual floppy disk Z80 Development System (SYS-80F). Further information on this system can be found in the SYS-80F Data Sheet, MK78575. FLP-80 DOS is a software package that consists of two collections of programs: DSS-80, the Development System Software, and DOPS-80, the Disk Operating Software. FLP-80DOS includes:

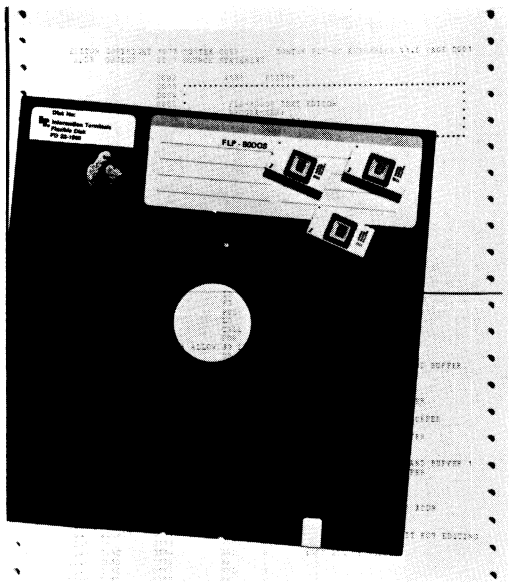
- ☐ Monitor
- ☐ Debugger
- ☐ Text Editor
- ☐ Z80 Assembler
- ☐ Relocating Linking Loader
- ☐ Peripheral Interchange Program
- ☐ Linker
- ☐ A Generalized I/O System For Peripherals

These programs provide state-of-the-art software for developing Z80 programs as well as establishing a firm basis for OEM products.

DEVELOPMENT SYSTEM SOFTWARE-DSS-80

Monitor

The Monitor provides user interface from the console to the rest of the software. The user can load and run system programs, such as the Assembler, using one simple command. Programs in object and binary format can be loaded into and dumped from RAM. All I/O is done via channels which are identified by Logical Unit Numbers. The Monitor allows any software device handler to be assigned to any Logical Unit Number. Thus, the software provides complete flexibility in configuring the system with different peripherals. The Monitor also allows two character mnemonics to represent 16-bit address values. Using mnemonics simplifies the command language. Certain mnemonics are reserved for I/O device handlers such as 'DK' for the flexible disk handler. The user can create and assign his own mnemonics at any time from the console, thus simplifying the command language for his own use. The Monitor also allows "batch mode operation" from any input device or disk file.



The Monitor commands are:

$ASSIGN - assign a Logical Unit Number to a device.
$CLEAR - remove the assignment of a Logical Unit Number to a device.
$RTABLE - print a list of current Logical Unit Number to device assignments.
$DTABLE - print default Logical Unit Number to device assignments.
$LOAD - load object modules into RAM.
$GTABLE - print a listing of global symbol table.
$GINIT - initialize global symbol table.
$DUMP - dump RAM to a device in object format.
$GET - load a binary file into RAM from disk.
$SAVE - save a binary file on disk.
$BEGIN - start execution of a loaded program.
$INIT - initialize disk handler.
$DDT - enter DDT debug environment.
IMPLIED RUN COMMAND - get and start execution of a binary file.

Designer's Development Tool - DDT

The DDT debugger program is supplied in a combination of PROM and on the FLP-80DOS diskette. It provides a complete facility for interactively de-

bugging relative and absolute Z80 programs. Standard commands allow displaying and modifying memory and CPU registers, setting breakpoints, and executing programs. Additional commands allow use of the MOSTEK AIM-80 to interactively debug a target system. Mnemonics are used to represent Z80 registers, thus simplifying the command language.

The allowed commands are:

B - Insert a breakpoint in user's program.
C - Copy contents of a block of memory to another location in memory.
E - Execute a program.
F - Fill an area of RAM with a constant.
H - 16-bit hexadecimal arithmetic.
L - Locate and print every occurrence of an 8-bit pattern.
M - Display, update, or tabulate the contents of memory.
P - Display or update the contents of a port.
R - Display the contents of the user's registers.
S - Hardware single step - requires MOSTEK's AIM-80 board.
W - Software single step.
V - Verify memory (compare two blocks and print differences).

Text Editor - EDIT

The FLP-80DOS Editor permits random access editing of ASCII character strings. The Editor works on blocks of characters which are rolled in from disk. It can be used as a line or character oriented editor. Individual characters may be located by positon or context. Each edited block is automatically rolled out to disk after editing. Although the Editor is used primarily for creating and modifying Z80 assembly language source statements, it may be applied to any ASCII text delimited by "carriage returns".

The Editor has a pseudo-macro command processing option. Up to two sets of commands may be stored and processed at any time during the editing process. The Editor allows the following commands:

An - Advance record pointer n records.
Bn - Backup record pointer n records.
Cn dS1dS2d - Change string S1 to string S2 for n occurrences.
Dn - Delete the next n records.
En - Exchange current records with records to be inserted.
Fn - If n = 0, reduce printout to console device (for TTY and slow consoles).
G - Get a file and insert it after the current line.
I - Insert records.
Ln - Go to line number n.
Mn - Enter commands into one of two alternate command buffers (pseudo-macro).
Pn - Put n records out to another file.
Q - Quit - Return to Monitor.
Sn dS1d - Search for nth occurrence of string S1.
T - Insert records at top of file before first record.
Vn - Output n records to console device.
Wn - Output n records to Logical Unit Number five (LUN 5) with line numbers.
Xn - Execute alternate command buffer n.

Z80 Assembler - ASM

The FLP-80DOS Assembler reads standard Z80 source mnemonics and pseudo-ops and outputs an assembly listing and object code. The assembly listing shows address, machine code, statement number, and source statement. The code is in industry standard hexadecimal format modified for relocatable, linkable assemblies.

The Assembler supports conditional assemblies, global symbols, relocatable programs, and a printed symbol table. It can assemble any length program, limited only by a symbol table size of over 400 symbols. Expressions involving arithmetic and logical operations are allowed. Although normally used as a two pass assembler, the Assembler can also be run as a single pass assembler or as a learning tool. The following pseudo-ops are supported:

DEFB - define byte.
DEFL - set label.
DEFM - define message (ASCII).
DEFS - define storage.
DEFW - define word.
END - end statement.
ENDIF - end of conditional assembly.
EQU - equate label.
GLOBAL - global symbol definition.
IF - conditional assembly.
INCLUDE - include another file within an assembly.
NAME - program name definition.
ORG - program origin.
PSECT - program section definition.
EJECT - eject a page of listing.
TITLE - place heading at top of each page of listing.
LIST - turn listing on.
NLIST - turn listing off.

Relocating Linking Loader - RLL

The MOSTEK FLP-80DOS Relocating Linking Loader provides state-of-the-art capability for loading programs into memory. Loading and linking of any number of relocatable or nonrelocatable object modules is done in one pass. A non-relocatable module is always loaded at its starting address as defined by the ORG pseudo-op during assembly. A relocatable object module can be positioned anywhere in memory at an offset address.

The Loader automatically links and relocates global symbols which are used to provide communication or linkage between program modules. As object modules are loaded, a table containing global symbol references and definitions is built up. The symbol table can be printed to list all global symbols and their load addresses. The number of object modules which can be loaded by the Loader is limited only by the amount of RAM available for the modules and the symbol table.

The Loader also loads industry standard non-relocatable, non-linkable object modules.

Linker - LINK

The Linker provides capability for linking object modules together and creating a binary (RAM image) file on disk. A binary file can be loaded using the Monitor GET or IMPLIED RUN command. Modules are linked together using global symbols for communication between modules. The linker produces a global symbol table and a global cross reference table which may be listed on any output device.

The Linker also provides a library search option for all global symbols undefined after the specified object modules are processed. If a symbol is undefined the Linker searches the disk for an object file having the file-name of the symbol. If the file is found, it is linked with the main module in an attempt to resolve the undefined symbol.

Peripheral Interchange Program - PIP

The Peripheral Interchange Program provides complete file maintenance facilities for the system. In addition, it can be used to copy information from any device or file to any other device or file. The command language is easy to use and resembles that used on DEC minicomputers. The following commands are supported:

COMMAND	FUNCTION
APPEND	Append files.
COPY	Copy files from any device to another device or file.
DIRECT	List directory of specified disk unit.
ERASE	Delete a file.
FORMAT	Format a disk.
INIT	Initialize the disk handler.
RENAME	Rename a file.
STATUS	List number of used and available sectors on specified disk unit.
QUIT	Return to Monitor.

The first letter only of each command may be used.

DISK OPERATING SOFTWARE - DOPS-80

The disk software, as well as being the heart of the AID-80F development system, can be used directly in OEM applications. The software consists of two programs which provide a complete disk handling facility.

Input/Output Control System - IOCS

The first package is called the I/O Control System (IOCS). This is a generalized blocker/deblocker which can interface to any device handler. Input and output can be done via the IOCS in any of four modes:

1. single byte transfer.
2. line at a time, where the end of a line is defined by carriage return.
3. multibyte transfers, where the number of bytes to be transferred is defined as the logical record length.
4. continuous transfer to end-of-file, which is used for binary (RAM-image)files.

The IOCS provides easy application of I/O oriented packages to any device. There is one entry point, and all parameters are passed via a vector defined by the calling program. Any given handler defines the physical attributes of its device which are, in turn, used by the IOCS to perform blocking and deblocking.

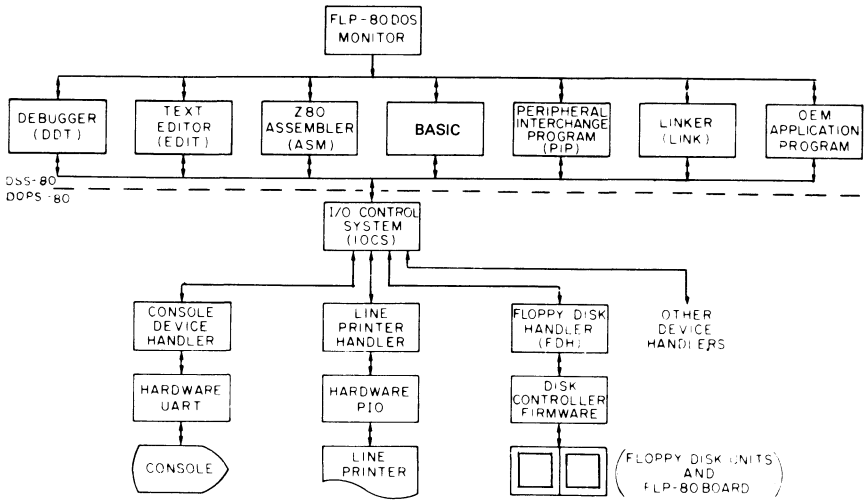
Floppy Disk Handler - FDH

The Floppy Disk Handler (FDH) interfaces from the IOCS to a firmware controller for up to 4 floppy disk units. The FDH provides a sophisticated command structure to handle advanced OEM products. The firmware controller interfaces to MOSTEK's FLP-80E Controller Board. The disk format is IBM 3740 soft sectored. The software can be easily adapted to double-sided disks. The Floppy Disk Handler commands include:

- erase file
- create file
- open file
- close file
- rename file
- rewind file
- read next n sectors
- reread current sector
- read previous sector
- skip forward n sectors
- skip backward n sectors
- replace (rewrite) current sector
- delete n sectors

The FDH has advanced error recovery capability. It supports a bad sector map and an extensive directory which allows multiple users. The file structure is doubly linked to increase data integrity on the disk and a bad file can be recovered from either its start or end.

FLP-80DOS FLOW CHART



MOSTEK®

Z80 MICROCOMPUTER SOFTWARE SUPPORT

Operating System (DDT-80)

FEATURES

- ☐ Program debug capability
- ☐ Channeled I/O for user convenience
- ☐ A set of I/O peripheral drivers is supplied
- ☐ Interactive hexadecimal addition and subtraction is in force when entering commands
- ☐ User expandable operating system

DESCRIPTION

DDT-80 is the Operating System for the Z80 Software Development Board (SDB-80). It resides in a 2K ROM (MK34000 series) resident on the SDB-80. It provides the necessary tools and techniques to operate the system, i.e., to efficiently and conveniently perform the tasks necessary to develop microcomputer software. DDT-80 is designed to support the user from initial design through production testing. It allows the user to display and update memory, registers, and ports, load and dump object files, set breakpoints, copy blocks of memory, and execute programs.

DDT-80 COMMAND SUMMARY

M s — Display and/or update the content of memory location s.

M s,f — Tabulate the contents to memory locations s through f.

P s — Display and/or update the content of I/O port s.

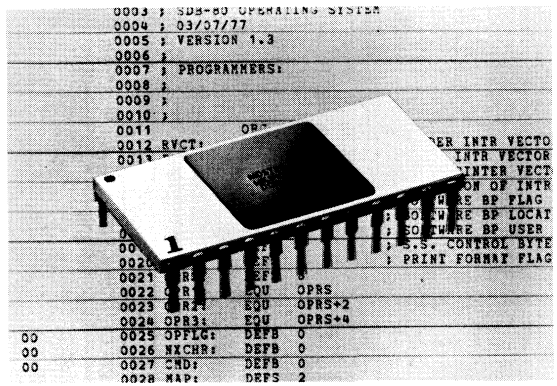
D s,f — Dump the contents of memory locations s through f in a format suitable to be read by the L command.

L — Load, into memory, data which is in the appropriate format.

E s — Transfer control from DDT-80 to a user's program starting at location s.

H — Perform 16 bit hexadecimal addition and/or subtraction.

C s,f,d — Copy the contents of memory locations s through f to another location in memory starting at location d.



B s — Insert a breakpoint in the user's program (must be in RAM) at location s which transfers control back to DDT-80. This allows the user to intercept his program at a specific point (location s) and examine memory and CPU registers to determine if his program is working correctly.

R — Display the contents of the user registers.

The s, f, and d represent start, finish, and destinations operands required for each command.

MEMORY, PORT AND REGISTER COMMANDS (M, P, R)

The M, P, and R commands provide the means for displaying the contents of specified memory locations, port addresses, or CPU registers. The M and P commands sequentially access memory locations or ports and display their contents. The user has the option of updating the content of the memory location or port. (Note some ports are output only and their contents cannot be displayed). The M command also gives the user access to the CPU registers through an area in RAM called the Register Map (discussed in the Execute, Breakpoint section below).

The M and R commands are used to tabulate blocks of memory locations (M) or the CPU registers (R). The M command will accept two operands, the starting and ending address of the memory block to be tabulated. The R command will accept either no operand or one. If no operand is specified, the CPU registers will be displayed without a heading. If an operand is specified then a heading which labels the register contents will be displayed as well.

EXECUTE AND BREAKPOINT (E, B)

The E command is used to execute all programs, including aids such as the Assembler. The B command is used to set a breakpoint to exit from a program at some predetermined location for debugging purposes. At the instant of a breakpoint exit, the contents of all CPU registers are saved in a designated area of SDB-80 RAM called the Register Map. In the Register Map, the register contents may be examined or modified using the M command and a predefined mnemonic (or absolute address) of the storage location for that register (example :PC, :A, . . . , :SP). The Register Map is also used to initialize the CPU registers whenever execution is initiated or resumed. Thus the E and B commands can be used together to initialize, execute, and examine the results of individual program segments.

The B command gives the user the option of having all CPU registers displayed when the breakpoint is encountered. This is done by entering a second operand to the B command. Otherwise DDT-80 defaults to displaying the PC and AF registers. When all CPU registers are displayed, the format is the same as for the R command previously discussed.

LOAD, DUMP, AND COPY (L, D, C)

The L and D commands load and dump object files through the object I/O channel in standard Intel Hex format. Checksums are used for error detection, and the addresses of questionable blocks are typed automatically while loading.

The C command will copy the contents of the memory block specified to another block of memory. There are no restrictions on the direction of the copy or on whether the blocks overlap.

HEXADECIMAL ARITHMETIC (H)

The H command is a dummy command used to allow hexadecimal addition and subtraction for expression evaluation without performing any other operation.

DDT-80 I/O CAPABILITIES

DDT-80 specifies three I/O channels, designated 'Console', 'Object', and 'Source', to which any suitable devices may be assigned. The Channel Assignment Table is located in RAM where it may be examined or modified using the M command. The table addresses correspond to the I/O channels and the table contents correspond to the addresses of the peripheral driver routines. A channel which has a device assignment may have that device assignment changed using the M command. This is accomplished by merely modifying the table contents of that channel's table address to correspond to the address of the new peripheral driver routine. A set of peripheral driver routines is supplied and listed below. This scheme also allows the user to write a driver routine for his own peripheral, load it into memory, and easily configure that peripheral into the system.

DDT-80 I/O PERIPHERAL DRIVERS

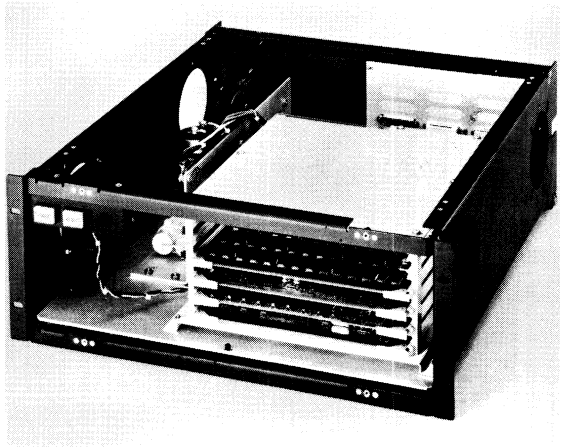
1. A serial input driver (usually a keyboard).
2. A serial output driver (usually a CRT or teletype typehead).
3. A serial input driver which sends out a reader step signal (usually a teletype reader).
4. A serial output driver which forces a delay after a carriage return (usually a Silent 700 typehead).
5. A parallel input driver (usually for high speed paper tape input).
6. A parallel output driver (usually for high speed paper tape output).
7. A parallel output driver (usually for a line printer).

DDT-80 USER EXPANDABILITY

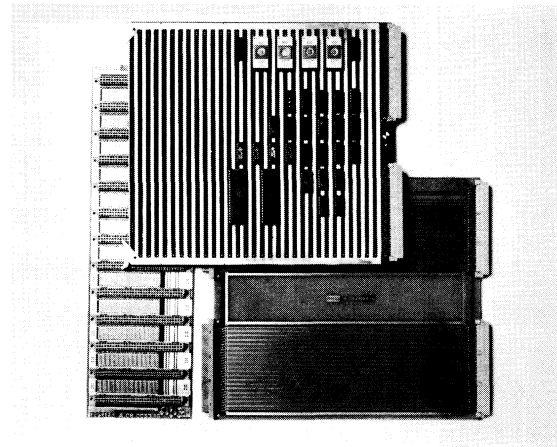
In its operation, DDT-80 will perform a jump indirect to itself using the contents of 2 designated RAM locations as the address jumped to. Usually this jump will be to a location in DDT-80 and on power-up and reset the 2 RAM locations are loaded with the correct address in DDT-80 for the jump. However, using the M command, the 2 RAM locations may be modified to correspond to a different address. DDT-80 will collect the command (single letter) and save it and will also scan for operands (up to 3), evaluating expressions to 4 hex digits. It is at this point that DDT-80 will perform the indirect jump to the address specified in the 2 RAM locations. Therefore, the user can supply an additional set of commands to enhance the operating system if desired.

SD SERIES ACCESSORIES

Mostek also has available SD Series compatible racks, extender cards, wire wrap cards, backplanes and cables to simplify system construction. The SYS-80 F can be used as a convenient starting point for many applications. Its 7 slots of which 3 are free for memory and I/O expansion and its heavy duty power supplies provide a solid base for system design.



Wire Wrap card : MK 79063



Extender card : MK 79062

SD-BP7 : MK 78099

7-slot prewired printed circuit back plane for the SD microcomputer board family. This card greatly simplifies system construction.

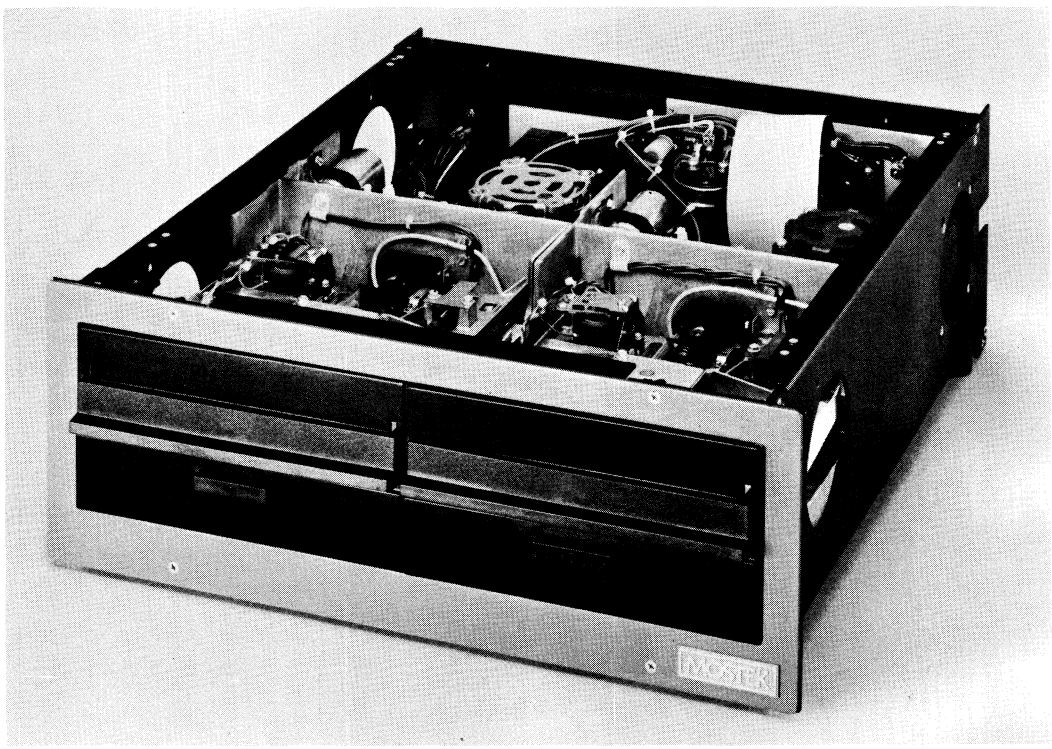
OEM RACK ASSEMBLIES

SD-RMS Rack Mounted System

- ☐ 6 slots for SD Series (double Eurocard)
- ☐ Saturated power supply + 5v at 12 amps
  + 12v at 1.5 amps
  − 12v at 1.5 amps
- ☐ PC Motherboard designed for minimum cross-talk
- ☐ Simple cabling to external peripherals
- ☐ 19" rack; 7" panel height
- ☐ Fan cooled
- ☐ 50/60 Hz — 220/110 v operation
- ☐ Table-top version available
- ☐ For stand alone or expanded operation (with SD-DFE — see page 90)

SD — DFE DUAL FLOPPY DISK ENCLOSURE

- ☐ Horizontally mounts two standard 8-inch floppy disk units
- ☐ Integral power supply : + 24 v at 3 Amps
  + 5 V at 3 Amps
  — 5 V at 1 Amps
- ☐ 50/60 Hz — 220/110 V operation
- ☐ Compatible with SD-RMS (see page 89)
- ☐ 19" Rack; 7" standard height
- ☐ Available with or without floppy disk driver



SD Series — COMING PRODUCTS

DCC-80

- ☐ 4 channel full duplex ADCCP serial interface card
- ☐ Supports asynchronous, monosynchronous, bisynchronous, SDLC and HDLC interchange protocoles
- ☐ Includes on board clock recovery logic and NRZ Encoding/Decoding logic
- ☐ Software programmable baud rates — from 75 to 19200 baud

HDC-80

- ☐ Hard disk controller, double Eurocard format
- ☐ Microprocessor controlled
- ☐ 9.2 MBit transfer rate
- ☐ SMD or Winchester compatible
- ☐ 12 to 96 megabyte disk on-line for applications requiring high throughput or large amounts of data
- ☐ Supports (amongst others) : Ampex, Shugart, CDC, Calcomp
- ☐ OEM-80 BUS compatible
- ☐ Complete disk handler operates under FLP-80DOS

Memory Mapping Proms for the OEM-80

The OEM-80 offers a very general mapping of the RAM and PROM memory spaces on the card. The user can program a pair of bipolar proms to configure the exact memory mapping he needs. Many applications can use a standard memory map and for these customers MOSTEK has a set of standard bipolar proms.

The memory mapping technique incorporates two proms, one selects the areas in 2K byte sections for RAM/PROM selection. The other selects 1K byte areas for the individual proms used in the prom area.

RAM/PROM SELECTION

For the 2K byte blocks (IC U31), a 32 x 8 prom selects what type of memory is in each block. The 8 outputs are defined as:

1 RAM area a
2 RAM area b
3 Scratchpad RAM
4 PROM socket 5
5 PROM area a
6 PROM area b
7 PROM area c
8 PROM area d

The OEM-80 logic allows the prom areas a to d to be or'd together. RAM area a or b must be exclusively selected. In the MK6260, the LSB select line 'J' is tied either high or low to limit selection to 4K blocks but allow two patterns to be stored in the prom. The MK6266 uses this line tied to A11. For 2K Byte block sizes. A 0 in the table indicates the block is selected.

PROM TYPE SELECTION

The second prom selects the EPROM type to be used. One preprogrammed prom is presently available. The MK6267 supports the mappings shown in the MK6267 table. The table assumes that the same type of prom is used in all sockets.

Address	MK6260 j=0								MK6260 j=1								MK6266							
	9	7	6	5	4	3	2	1	9	7	6	5	4	3	2	1	9	7	6	5	4	3	2	1
0000							0	0				0							0				0	
0800							0	0				0							0				0	
1000							0				0							0					0	
1800							0				0							0					0	
2000							0			0								0					0	
2800							0			0								0					0	
3000							0		0									0					0	
3800							0		0									0					0	
4000															0	0								
4800															0	0								
5000															0									
5800															0									
6000															0									
6800															0									
7000															0									
7800															0									
8000																								
8800																								
9000																								
9800																								
A000					0																			
A800					0																			
B000				0																				
B800				0																				
C000			0																					0
C800			0																					0
D000	0																							0
D800	0																							0
E000					0								0				0							
E800					0								0				0							
F000						0								0							0			
F800						0								0								0		0

MK6267		
Starting address	PROM type	K bytes/prom
0000/4000/8000/C000	2708/2758	1k
2000/6000/A000/E000	2708/2758	1k
0000/4000/8000/C000	2716	2k
0000/4000/8000/C000	2532	4k

SD (SYSTEM DESIGN) SERIES MICROCOMPUTER BOARDS

ORDERING INFORMATION

DESIGNATOR	DESCRIPTION	MOSTEK ORDER No.
OEM-80/4	4K RAM, 5 sockets for EPROM or ROM, 2-PIO with sockets for TTL buffering logic, CTC, UART, socket for 256x8 scratch pad RAM and sockets for memory mapping PROMs. Manual MK78548	MK78159
OEM-80/16	16K RAM, otherwise same as OEM-80/4	MK78160
FLP-80	Floppy disk controller board with 128 byte FIFO, WD 1771 controller and complete FLP-80 DOS software on four 2708 PROMs and 1 diskette; includes manual MK78561	
FLP-80 OEM	Floppy disk controller board with FIFO and WD 1771; no software, only available to licensed customers.	MK78146
DOPS-80[1]	DOPS-80 Disk Operating Software is supplied on Mostek diskette as source, relocatable object and binary; and in the form of complete listings. FLP-80DOS Operations Manual MK78557	MK78136
RAM-80A	16K bytes RAM non expandable Manual MK78573	MK78162
RAM-80B	16K bytes RAM plus 2 PIO. Sockets for TTL I/O buffering. RAM expandable to 64K. Manual MK78555	MK78161
RAM-80C	16K bytes RAM, expandable to 64K. Manual MK78573	MK78163
VDI-S	80x24 display controller for standard video monitors. Serial interface over UART. Requires ASCII encoded keyboard.	MK78033a
VDI-P VDI S/P	Same as VDI-S but uses Z80 PIO for interface with OEM-80 bus. Manual MK78586	MK78035a

ORDERING INFORMATION

DESIGNATOR	DESCRIPTION	PART NO.
A/D-80E	Analog to digital board. See page 82 for optional configurations available Manual MK79660	See page 82
DDT-80[1]	Debug ROM, 2K byte preprogrammed mask ROM. Manual MK78522	MK78118
EXTENDER CARD	3-layer extender card for testing OEM-80 systems.	MK79062
WIRE WRAP	Blank card with space for 120 20-Pin ICs. Used for custom OEM-80 interfaces.	MK79063
BACKPLANE MODULE	4-layer, 7-slot PC backplane for OEM-80 systems, PC and power connectors are mounted on the card.	MK78099
SD RMS	6-slot OEM rack (19")	MK78182
SD DFE	Dual disk drive rack	MK78185

[1]Delivery subject to licensing agreement procedure.

Microcomputer Systems & Software

Microcomputer Development System (SYS-80FT)

INTRODUCTION

The Mostek SYS-80FT is a complete state of the art disk based computer. Not only does it provide all the necessary tools for software development but it provides complete hardware/software debug through Mostek 's AIM series of in-circuit emulation cards for the Z80, the 3870 family and future Mostek microprocessors.

The disk based system eliminates the need for other mass storage media but provides ease of interface to any peripheral normally used with computers. The file oriented floppy disk structure for data storage and retrieval provides a reliable, portable media to speed and facilitate software development.

The FLP-80DOS Disk operating system is designed for maximum flexibility both in use and expansion to meet a multitude of end user or OEM needs.

Because FLP-80DOS is supported by Mostek's SD and MD Series of OEM boards, software designed on the SYS-80FT can be directly used in OEM board applications.

DEVELOPMENT SYSTEM FEATURES

The SYS-80FT is an excellent integration of both hardware and software development tools for use throughout the complete system design and development phase. The software development is begun by using the combination of Mostek's Text Editor with « roll in - roll out » virtual memory operation and the Mostek relocating assembler. Debug can then proceed with the SYS-80FT using its resources as if they were in the final system. Using combinations of the Monitor, Designer's Debugging Tool, execution time breakpoints, and single step/multistep operation along with a formatted memory dump provides control for attacking those tough problems. The use of the Mostek AIM option provides extended debug with versatile hardware breakpoints on memory or port locations, a buffered in-circuit emulation cable for extending the software debug into its own natural hardware environment, as well as a history memory to capture bus transactions in real time for later examination and external probes for analysis of signals not directly related to the CPU bus. The relocatable and linking feature of the assembler enables the use of contemporary modular design techniques whereby major system alterations can be made in small tractable modules. Using the linker, the small modules can be combined to form a run time module without major reassembly of the entire program.

PACKAGED SYSTEM FEATURES

From a system standpoint, the SYS-80FT has been designed to be the basis of an end product small business/industrial computer. The flexibility provided in the FLP-80DOS operating system permits application programs to be as diverse as a high level language compiler or a supervisory control system in the industrial environment. Other hardware options are available, with even more to be added.

Expansion of the disk drive units to a total of four single sided or double sided units provide up to two megabytes of storage. This computer uses the third generation Z80 processor supported with the power of a complete family of peripheral chips. Through use of its 158 instructions (including : 16-bits arithmetic, bit manipulation, advanced block moves and interrupt handling), almost any application from communication concentrators to general purpose accounting systems is made easy.

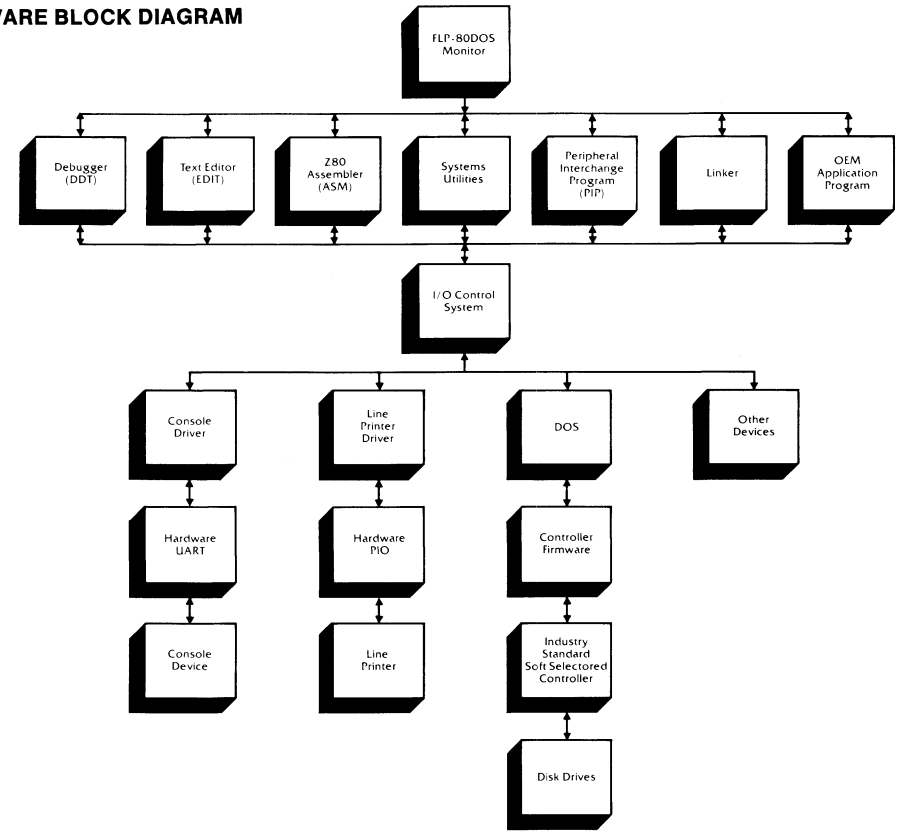
OEM FEATURES

The hardware and software basis for the SYS-80FT is also available separately to the OEM purchaser. Through a software licensing agreement, all Mostek Software can be utilized on these OEM series of cards. A growing line of support cards and card cages, permits the user to configure a multitude of different systems.


STEK
• INDUSTRIAL
nd Systems
OSTEK
ATION • INDUSTRIAL
its and Systems
STEK
• INDUSTRIAL
d Systems


MOSTEK
TARGET
MOSTEK
AIM-80X
AIM-80

SOFTWARE BLOCK DIAGRAM



SOFTWARE

A microcomputer system is only as good as its software. Mostek has a wide range of software products that can turn the powerful Z80 based hardware into a system of mini-computer capabilities.

SYS-80FT Resident Software (FLP-80 DOS)

A totally integrated package of resident software is offered in conjunction with the SYS-80FT consisting of:

Monitor
DDT-80 with extended debug through AIM-80
Text Editor
Z80 Relocating Assembler
Peripheral Interchange Program
Linker
I/O Control System
Floppy disk handler
Device driver library
Batch mode operation

Optional Software includes:

Macro Assemblers Z80 + 3870
Basic
Fortran
Pascal
Micro Cobol
Utility package

Monitor

The FLP-80DOS Monitor is the environment from which all activity in the system initiates. From the Monitor, any system routine such as PIP or a user generated program is begun by simply entering the program name. FLP-80DOS I/O is done in terms of logical unit numbers, as is commonly done in FORTRAN. A set of logical units are preassigned to default I/O drivers upon power or reset. From the console the user can reassign any logical unit to any new I/O device and can also display logical unit assignments. Executable file creation can be done by the Save command, and Hex object files can be produced using the Dump command. User generated binary files can be loaded with the GET command or can be directly executed by entering the program name.

Text Editors

The Text Editors permit editing/creating of any source file independent of the language being written.
Two are provided. The Editor for TTY type I/O.
The Video Editor for CRT I/O.
The Editor is both line and string oriented to give maximum utility and user flexibility.
The Editor through its virtual memory "roll in roll out" technique can edit a file whose length is limited only by maximum diskette storage.
Included in the repertoire of 16 commands are macro commands to save time when encountering redundant editing tasks. The Editor is also capable of performing in one operation all the commands which will fit into an 80 column command buffer.

Summary of Editor commands

Advance N	— Advance line pointer N line
Backup N	— Backs up N lines
Change N/S1/S2	— Change N occurrences of string 1 to string 2
Delete N	— Delete current line plus next N-1 lines of text
Exchange N	— Exchanges current line plus next N-1 lines with lines to be inserted while in insert mode.
Get	— Inserts a file from the disk into the text.
Insert	— Place Editor in insert mode. Text will be inserted after present line.
Line N	— Place line pointer on Line N.
Macro 1 or Macro 2	— Defines Macro 1 or Macro 2 by the following string of text Editor commands.
Put	— Writes a section of code to the disk.
Quit	— Stores off file under editing process and returns to Monitor environment.
Search N/S 1	— Search from existing pointer location until Nth occurrence of string S1 is located and print it.
Top	— Inserts records at top of file before first line.
Verify N	— Print current record to console plus next n-1 records while advancing pointer N records ahead.
Window N	— Prints current records plus next N-1 records to source output device while advancing pointer N records.
eXecute N	— Executes Macro 1 or Macro 2 as defined by Macro command.

The video Editor displays the text to be edited directly on the CRT screen as if it were a window into memory.
The window and cursor need only be positioned over the character to be changed. Then the new text can be added or old deleted with the changed data dispayed immediatly on the screen. The video Editor allows programs (or any text) to be entered or corrected much quicker and easier than possible with TTY oriented Editors.
The video Editor provides all the commands of the Editor except E,I,M,V,W,X.

Z80 Assembler

The Z80 Resident Assembler generates relocatable or absolute object code from source files independent of source medium. The assembler recognizes all 158 Z80 instructions as well as 20 powerful pseudo operators. The object code generated is industry standard absolute or relocatable format. With the relocating feature, large programs can easily be developed in smaller sections and linked using the system Relocating Linking Loader or Linker. Because the assembler utilizes the I/O Control System, object modules or list modules can be directed to disk files, paper tape, console, or line printer. Portability of output media eliminates the requirement for a complete set of peripherals at every software/hardware development system.

The assembler run time options include sorted symbol table generation, no list, no object, pass 2 only, quit, cross reference table and reset symbol table. The assembler is capable of handling 14 expression operators including logical, shift, multiplication, division, addition and subtraction operations.

These permit complex expressions to be resolved at assembly time by the assembler rather than manually by the programmer. Comments can be placed anywhere but must be preceded by a semicolon. Error messages are integrated with listing file but can be directed to console device. Additions to standard assembler pseudo operators are :

GLOBAL	— For global definition.
PSECT operator	— To generate relocatable or absolute modules.
IF expression	— Conditional assembly IF expression is true.
INCLUDE dataset	— To include other datasets (files) as in-line code anywhere in source file.

Peripheral Interchange Program

PIP provides complete file maintenance activity for operations such as copy file from disk to disk, disk to peripheral, or any peripheral to any other peripheral supporting both file structured and character oriented devices. Key operations such as renaming, appending, and erasing files also exist along with status commands for diskette ID and vital statistics. PIP can search the diskette directories for any file or a specific name, extension, and user number. The PIP operations are :

Append	— Appends file 1 to file 2 without changing file 1.
Copy	— Copies input files or data from an input device to an output file or device. The Copy command can be used for a variety of purposes such as listing files, concatenating individual files, or copying all the files or a single file from one disk unit (e.g. DK0) to a second disk unit (e.g. DK1).

Directory — Lists the directory of a specified disk unit (DK0, DK1, etc...). The file name, extension, and user number is listed for each file in the directory. The user can also request listing only files of a specified name, only files of a specified extension or only files of a specified user number. The list device can be any device supported by the system as well as a file.

Erase — Erases a single file or files from a diskette in a specified disk unit. The user has the option to erase all files, only files of a specified file name, only files of a specified extension or only files of a specified user number.

Format — Takes completely unformatted soft sectored diskettes. Formats to IBM 3740, and prepares to be system diskette. Operation is performed on diskette unit 1 and a unique 11 character name is assigned to that diskette.

Init — Initializes maps in disk handler when a new diskette has been changed while in the PIP environment.

Rename — Renames a file, its extension, and its user number to a file of name X, extension Y, and user Z.

Status — Lists all vital statistics of a disk unit to any list device. These include number of allocated records, number of used records, and number of bad records.

Quit — Returns to Monitor Environment.

DOS/Disk Handler

The heart of the FLP-80DOS software package is the Disk Operating System. Capable of supporting 4 double sided units, the system provides a file structured orientation timed and optimized for rapid storage and retrieval. Thorough error reporting exists from the DOS to enable an application programmer to quickly debug his program as wel as extensive error recovery and bad sector allocation which insures data and file integrity. The DOS not only provides file reading and writing capability but special pointer manipulation, record deletions, record insertions, skip records both forward and backward as well as directory manipulation such as file creation, renaming,and erasure. The DOS is initiated by a calling vector which is a subset of the I/O control system vector or through the standard IOCS calling sequence to elect buffer allocation, blocking and deblocking of data to a user selectable logical record type.

A unique dynamic allocation algorithm makes optimal use of disk storage space. Run time (Binary Files) are given first priority to large blocks of free space to eliminate any such overhead in operation system and overlay programs. The algorithm marks storage fragments as low priority and uses them only when diskette is nearing maximum capacity. The DOS permits 7 files to be active at any one time, thus permitting complex application programs as well as multiuser operation of the DOS.

I/O Control system

The I/O control system provides a central facility from which all calls to I/O can be structured. This permits a system applications program to dissolve any device dependence by utilizing the logical unit approach of large main frame computers. For example, a programmer may want to structure his utility to use logical unit no 5 as his list device which normally in his system defaults to the line printer. He may, however, assign at run time a different device for logical unit No 5. His application program remains unchanged.

Interface by a user to IOCS is done simply be entering a device mnemonic in a table and observing the calling sequence format. IOCS supplies a physical buffer of desired length, handles buffer allocation, blocking, deblocking and provides a logical record structure as specified by the user.

DDT

The Designer's Debugging Tool consists of commands for facilitating an otherwise difficult debugging process. The SYS-80FT support of fast program modifications through editing and re-assemblies, followed by DDT operations close the loop on the debug cycle. The DDT commands include :

Memory — Display, update, or tabulate memory.

Port — Display, update or tabulate I/O ports.

Execute — Performs 16 bit add/sub.

Hexadecimal — Execute user's program.

Copy — Copy one block to another.

Breakpoint — Set software trap in user code for interrupting execution in order to examine CPU registers.

Register — Display contents of user's registers.

Offset — Enter address adder for debug of relocatable modules.

Fill — Fill specified portion of memory with 8 bit byte.

Verify — Compare two blocks of memory.

Walk — Software single step/multistep.

Quit — Return to Monitor.

Linker

The Linker program provides the capability of linking assembler generated absolute or relocatable object modules together to create a binary or run time file. The linking process includes the library search option that if elected, will link in standard library object files (device drivers, math pack functions IOCS features) on disk to resolve undefined global symbols. By selecting an option a complete cross reference table will be generated and stored in a separate file, a list of undefined global symbols will be printed, and/or the global symbol table will be generated and stored in the same file as the cross reference symbol table.

Batch Mode Operation

In Batch Mode Operation, a command file is built on disk or assigned to a peripheral input device such

as a card reader. The console input normally taken from the keyboard is taken from this batch device or batch file.

While operating under direction from a batch file, either the console device prompts the user or the prompting can be directed to any other output device. The Batch file definable operation is especially useful to execute rendundant procedures not requiring constant attention of the operator and to allow several programmers to use one system.

SYSTEM SPECIFICATIONS

- ☐ Z80 CPU.
- ☐ 4K byte PROM bootstrap and Z80 debugger.
- ☐ 58K bytes RAM.
- ☐ 4K bytes PROM bootstrap
- ☐ 8 x 8 bit I/O ports (4 x PIO) with user definable drivers/receivers.
- ☐ Serial port. RS 232 and current loop.
- ☐ 4 channel counter/timer (CTC).
- ☐ 2 disk drives 250K bytes per floppy disk.
- ☐ 80 x 24 display terminal with Cursor Addressing and Inverse Video. Full ASCII keyboard.
- ☐ 3 free positions for AIM modules, A/D cards, Modem interfaces, etc.
- ☐ Cross Assembler and debug ability for 3870 and future Mostek microprocessors.
- ☐ Device drivers for paper tape readers, punches, card readers, line printers, Silent 700's, teletypes and CRT's are included.
  Others can be added.
- ☐ PROM programer I/O port. Programer itself is optional.
- ☐ Bus compatible with Mostek SD series of OEM boards.

HARDWARE DESCRIPTION

CPU Module (OEM-80E)

The OEM-80 provides the essential CPU power of the system. While using the Z80 as the central processing unit, the OEM-80 is provided with other Z80 family peripheral chip support. Two Z80 PIO's give 4 completely programmable 8 bit parallel I/O ports with handshake from which the standard system peripherals are interfaced. Also on the card is the Z80-CTS counter time circuit which as 3 free flexible channels to perform critical counting and event counter timing functions. Along with 16K of RAM, the OEM-80 provides 5 ROM/PROM sockets which can be utilized for 10/20K of ROM or 5/10K PROM. Four sockets contain the firmware portion of FLP-80DOS. The remaining socket can be strapped for other ROM/PROM elements.

RAM-80BE

The RAM 80B adds additional memory with Mostek's MK 4116 16K dynamic memory along with more I/O. The four fully programmable 8 bit I/O ports with handshake provide the additional I/O ports in the system.

FLP-80E

Integral to the SYS-80FT system is the floppy controller. The FLP-80 is a complete IBM 3740 single density/double sided controller for up to 4 drives. The controller has 128 bytes of FIFO buffer resulting in a completely interruptable disk system.

AIM80E + AIM-80/4

The AIM-80 module provides extended debug for the SYS-80FT in Z80 development, real time in-circuit emulation permits debug of the hardware and the software at the most intimate level. Hardware single step/multistep with register trace, execution intercept on memory access, port access, or external trigger provides the absolute control over any system no matter how complex. The "pushbutton intercept" enables the programmer to perform a controlled recovery for those extremely difficult to trace processor lock out loops. With the memory clock selectable history module, any past 256 events of data, address, or control bus operation are captured in real time and displayable.

The AIM-80 includes 8K bytes of ROM firmware introducing unique software including a mnemonic disassembler for inverse assembly of history module contents or single step/multistep operations. "In line" code disassembled to language mnemonics provides insight into execution results as if examining an assembler generated listing. Extra added capability is the ROM resident self test of OEM-80 or target RAM.

AIM-80/4 provides upgraded operation to 4.0 MHz.

AIM - 72E + AIM - 73

The AIM-72 module provides debug and in-circuit emulation capabilities for the 3870 series microcomputers (3870, 3872 and 3876) on the SYS-80FT. Multiple breakpoint capability and single step operation allows the designer complete control over the execution of the 3870 Series microcomputer.

Register and Port display and modification capability provides information needed to find system "bugs" A11 I/O is in the user's system connected to AIM-72 by a 40-pin interface cable.

The debugging operation is controlled by ZAIM-72 a mnemonic debugger which controls the interaction between the Z80 host computer and the 3870 slave. For 3873 debugging, the AIM-73 can be used. It includes a history module for the last 1024 CPU cycles and also supports all 3870 family circuits.

Assembly and linking is done using the Macro-70 Assembler and the standard FLP-80DOS linker.

MECHANICAL SPECIFICATIONS

SYS-80FT Enclosure

Overall Dimensions : (in cm) 60 w x 42 d x 29 h.
Weight : 40 kg.
Fan capacity : 110 CFM.
Card Cage Capacity : 7 Double Eurocards.
Card Connectors : Din 41612.
Operating Temperature Range : + 10° C to + 35° C.
Power Supply.
Input : 230 V ± 10 % - AC - 50/60 Hz.
Outputs : + 5 VDC at 15 Amps Max.
+ 12 VDC at 1.2 Amps Max.
— 12 VDC at 1.2 Amps Max.
+ 24 VDC at 3 Amps Max.

Z80 Macro Assembler (MACRO-80)

FEATURES

- ☐ Assembles standard Z80 instruction set to produce relocatable, linkable, object modules.

- ☐ Provides nested conditional assembly, an extensive expression evaluation capability, and an extended set of assembler pseudo-ops:

ORG	- origin
EQU	- equate
DEFL	- set/define macro label
DEFM	- define message
DEFB	- define byte
DEFW	- define word
DEFS	- define storage
END	- end of program
GLOBAL	- global symbol definition
NAME	- module name definition
PSECT	- program section definition
IF/ENDIF	- conditional assembly
INCLUDE	- include another file in source module
LIST/NLIST	- list on/off
CLIST	- code listing only of macro expansions
ELIST	- list/no list of macro expansions
EJECT	- eject a page of listing
TITLE	- place title on listing

- ☐ Provides options for obtaining a printed cross-reference listing, terminating after pass 1 if errors are encountered, redefining standard Z80 opcodes via macros, and obtaining an unused symbol reference table.

- ☐ Provides the most advanced macro handling capability in the microcomputer market which includes:
  - optional arguments
  - default arguments
  - looping capability
  - global/local macro labels
  - nested/recursive expansions
  - integer/boolean variables
  - string manipulation
  - conditional expansion based on symbol definition
  - call by value facility
  - expansion of code-producing statements only
  - expansion of macro call statement only

- ☐ Listing and object modules can be output on disk files or any device.

- ☐ Compatible with other Mostek Z80 assemblers and FLP-80DOS Version 2.0 or higher. Requires 32K or more of system RAM.

DESCRIPTION

MACRO-80 is an advanced upgrade from the FLP-80DOS Assembler (ASM). In addition to its macro capabilities, it provides for nested conditional assembly and allows symbol lengths of any number of characters. It supports global symbols, relocatable programs, a symbol cross-reference listing, and an unused symbol reference table. MACRO-80 is upward compatible with all other MOSTEK Z80 assemblers.

The Mostek Z80 Macro Assembler (MACRO-80) is designed to run on the Mostek Dual Disk Development System with 32K or more of RAM. It requires FLP-80DOS, Version 2.0 or higher. Macro pseudo-ops include the following:

MACRO/MEND	- define a macro
MNEXT	- step to next argument
MIF	- evaluate expression and branch to local macro label if true
MGOTO	- branch to local macro label
MEXIT	- terminate macro expansion
MERROR	- print error message in listing
MLOCAL	- define local macro label

Predefined macro-related parameters include the following:

%NEXP	- current number of this expansion
%NARC	- number of arguments passed to expansion
#PRM	- expand last used argument
%NPRM	- number of last used argument
%NCHAR	- number of characters in argument

The operations manual describes in detail all facilities available in MACRO-80 and provides a host of examples and sample print-outs.

3870/F8 Macro Cross Assembler (MACRO-70)

FEATURES

- ☐ Assembles standard 3870/F8 instruction set to produce relocatable, linkable object modules.

- ☐ Provides nested conditional assembly, an extensive expression evaluation capability, and an extended set of assembler pseudo-ops:

ORG	- origin
EQU	- equate
DC	- define constant
DEFL	- set/define macro label
DEFM	- define message
DEFB	- define byte
DEFW	- define word
DEFS	- define storage
END	- end of program
GLOBAL	- global symbol definition
NAME	- module name definition
PSECT	- program section definition
IF/ENDIF	- conditional assembly
INCLUDE	- include another file in source module
LIST/NLIST	- list on/off
CLIST	- code listing only of macro expansions
ELIST	- list/no list of macro expansions
EJECT	- eject a page of listing
TITLE	- place title on listing

- ☐ Provides options for obtaining a printed cross-reference listing, terminating after pass 1 if errors are encountered, redefining standard 3870 opcodes via macros, and obtaining an unused symbol reference table.

- ☐ Provides the most advanced macro handling capability on the microcomputer market which includes:
  - optional arguments
  - default arguments
  - looping capability
  - global/local macro labels
  - nested/recursive expansions
  - integer/boolean variables
  - string manipulation
  - conditional expansion based on symbol definition
  - call by value facility
  - expansion of code-producing statements only
  - expansion of macro call statements only

- ☐ An extended instruction set for the 3870 is defined via a macro definiton file and is shipped with the MACRO-70 diskette.

- ☐ Listing and object modules can be output on disk files or any device.

- ☐ Compatible with other Mostek 3870/F8 assemblers and FLP-80DOS Version 2.0 or higher. Requires 32K or more of system RAM.

DESCRIPTION

MACRO-70 is an advanced upgrade from the 3870/F8 Cross Assembler (FZCASM). In addition to its macro capabilities, it provides for nested conditional assembly and allows symbol lengths of any number of characters. It supports global symbols, relocatable programs, a symbol cross-reference listing, and an unused symbol reference table. MACRO-70 is upward compatible with all other MOSTEK 3870/F8 Assemblers.

The Mostek 3870/F8 Macro Assembler (MACRO-70) is designed to run the on Mostek Dual Disk Development System with 32K or more of RAM. It requires FLP-80DOS, Version 2.0 or higher. Macro pseudo-ops include the following:

MACRO/MEND	- define a macro
MNEXT	- step to next argument
MIF	- evaluate expression and branch to local macro label if true
MGOTO	- branch to local macro label
MEXIT	- terminate macro expansion
MERROR	- print error message in listing
MLOCAL	- define local macro label

Predefined macro-related parameters include the following:

%NEXP	- current number of this expansion
%NARC	- number of arguments passed to expansion
#PRM	- expand last used argument
%NPRM	- number of last used argument
%NCHAR	- number of characters in argument

MOSTEK®

Z80 MICROCOMPUTER SYSTEMS

FORTRAN IV Compiler

FEATURES

- ☐ All of ANSI standard FORTRAN IV (X3.9-1966) except complex data type.
- ☐ Generates relocatable linkable object code.
- ☐ Subroutines may be compiled separately and stored in a system library.
- ☐ Compiles several hundred statements per minute in a single pass.
- ☐ Enhancements include
  1. LOGICAL variables which can be used as integer quantities
  2. LOGICAL DO loops for tighter, faster execution of small valued integer loops.
  3. Mixed mode arithmetic.
  4. Hexadecimal constants.
  5. Literals and Holleriths allowed in expressions.
  6. Logical operations on integer data. .AND., .OR., .NOT. and .XOR. can be used for 16-bit or 8-bit Boolean operations.
  7. READ/WRITE End-of-File or Error Condition transfer. END=n and ERR=n (where n is the statement number) can be included in READ or WRITE statements to transfer control to the specified statement on detection of an error or end-of-file condition.
  8. ENCODE/DECODE for FORMAT operations to memory.
- ☐ Long descriptive error messages.
- ☐ Extended optimizations
- ☐ Z80 assembly language subprograms may be called from FORTRAN programs

DESCRIPTION

Mostek's FORTRAN IV Compiler package provides new capabilities for users of Z80-based microcomputer systems. Mostek FORTRAN is comparable to FORTRAN compilers on large mainframes and minicomputers. All of ANSI Standard FORTRAN X3.9-1966 is included except the COMPLEX data type. Therefore, users may take advantage of the many applications programs already written in FORTRAN.

Mostek FORTRAN IV is unique in that it provides a microprocessor FORTRAN development package that generates relocatable object modules. This means that

only the subroutines and system routines required to run FORTRAN programs are loaded before execution. Subroutines can be placed in a system library so that users can develop a common set of subroutines that are used in their programs. Also, if only one module of a program is changed, it is necessary to re-compile only that module.

The standard library of subroutines supplied with FORTRAN includes:

ABS	IABS	DABS	AINT
INT	IDINT	AMOD	MOD
AMAXO	AMAX1	MAXO	MAX1
DMAX1	AMINO	AMIN1	MINO
MIN1	DMIN1	FLOAT	IFIX
SIGN	ISIGN	DSIGN	DIM
IDIM	SNGL	DBLE	EXP
DEXP	ALOG	DLOG	ALOG10
DLOG10	SIN	DSIN	COS
DCOS	TANH	SQRT	DSQRT
ATAN	DATAN	ATAN2	DATAN2
DMOD	PEEK	POKE	INP
OUT			

The library also contains routines for 32-bit and 64-bit floating point additon, subtraction, multiplication, division, etc. These routines are among the fastest available for performing these functions on the Z80.

A minimum system size of 48K bytes (including FLP-80DOS) is required to provide efficient optimization. The Mostek FORTRAN compiler optimizes the generated object code in several ways:

1. Common subexpression elimination. Common subexpressions are evaluated once, and the value is substituted in later occurrences of the subexpression.

2. Peephole Optimization. Small sections of code are replaced by more compact, faster code in special cases.

3. Constant folding. Integer constant expressions are evaluated at compile time.

4. Branch Optimizations. The number of conditional jumps in arithmetic and logical IFs is minimized.

Long descriptive error messages are another feature of the compiler. For instance:

?Statement unrecognizable

is printed if the compiler scans a statement that is not an assignment or other FORTRAN statement. The last twenty characters scanned before the detected error are also printed.

As an option, the compiler generates a fully symbolic listing of the machine language to be generated. At the end of the listing, the compiler produces an error summary and tables showing the addresses assigned to labels, variables and constants.

LINKER

A relocating linking loader (LINK-80) and a library manager (LIB-80) are included in the Mostek FORTRAN package.

LINK-80 resolves internal and external references between the object modules loaded and also performs library searches for system subroutines and generates a load map of memory showing the locations of the main program, subroutines and common areas.

LIBRARY MANAGER

LIB-80 allows users to customize libraries of object modules. LIB-80 can be used to insert, replace or delete object modules within a library, or create a new library from scratch. Library modules and the symbol definitions they contain may also be listed.

CP/M UTILITY

A utility program (XCPM) is included which allows the user to copy FORTRAN source programs from CP/M diskettes to FLP-80DOS diskettes. At this point the programs can be compiled using the Mostek FORTRAN compiler.

FTRANS UTILITY

FTRANS allows the user to convert object programs produced by the Mostek Z80 assembler to a form that is linkable to FORTRAN programs.

MOSTEK®

Z80 MICROCOMPUTER SYSTEMS

BASIC Software Interpreter

FEATURES

- ☐ Direct access to CPU I/O Ports
- ☐ Ability to read or write any memory location (PEEK, POKE)
- ☐ Arrays with up to 255 dimensions
- ☐ Dynamic allocation and de-allocation of arrays
- ☐ IF...THEN...ELSE and nested IF...THEN...ELSE
- ☐ Direct (immediate) execution of statements
- ☐ Error trapping, with error messages in English
- ☐ Four variable types: Integer, string, real and double precision real.
- ☐ Full PRINT USING capabilities for formatted output
- ☐ Extensive program editing facilities
- ☐ Trace facilities
- ☐ Can call up to 10 assembly language subroutines
- ☐ Boolean (logical) operations
- ☐ Supports up to 6 sequential and random access files on floppy disk.
- ☐ Complete set of file manipulation statements
- ☐ Occupies only 19K bytes, not including operating system
- ☐ Supports console and line printer I/O
- ☐ Allows console output to be redirected to the line printer

DESCRIPTION

MOSTEK BASIC is an extensive implementation of Microsoft BASIC for the Z80 microprocessor. Its features are comparable to those BASICs found on minicomputers and large mainframes. Mostek BASIC is among the fastest microprocessor BASICs available.

Designed to operate on the Mostek Dual Disk Development System with FLP-80DOS and 32K bytes or more memory. Mostek BASIC provides a sophisticated software development tool.

Mostek BASIC is implemented as an interpreter and is highly suitable for user interactive processing. In a 32K byte system, about 7K bytes of free storage area are available to the user. Programs and data are stored in a compressed internal format to maximize memory utilization. By adding more memory to the system, the user's program and data storage area may be increased to as much as 31K bytes.

Unique features include long variable names, substring assignments and hexadecimal and octal constants. Many other features ease the task of programming complex functions. The Programmer is seldom limited by array size (up to 255 dimensions, with run time allocation and deallocation) or I/O restrictions. Full PRINT USING capabilities allow formatted output, while both input and output may be performed with multiple sequential and random files on floppy disk as well as with the CPU I/O ports. Editing, error trapping, and trace facilities greatly simplify program debugging.

Commands:

AUTO	CLEAR	CONTST	DELETE	EDIT
FILES	LIST	LLIST	LOAD	MERGE
NEW	NULL	RENUM	RESET	RUN
SAVE	SYSTEM	TRON	TROFF	WIDTH

Program Statement:

DEFNx	DEFDBL	DEFINT	DEFSNG	DRFSTR
DIM	END	ERASE	ERROR	FOR
GOSUB	GOTO	IF...THEN(ELSE)	LET	NEXT
ON...ERROR	ON...GO SUB	ON...GOTO	OUT	POKE
REM	RESUME	RETURN	STOP	SWAP
WAIT				

Input/Output Statements:

CLOSE	DATA	FIELD	GET	INPUT
KILL	LINEINPUT	LSET	NAME	OPEN
PRINT	PUT	READ	RESTORE	RESET

Operators

=	-	+	*	/
↑	\	MOD	NOT	AND
OR	XOR	IMP	EQV	<
>	<=	>=	<>	

Arithmetic Functions

ABS	ATN	CDBL	CINT	COS
CSNG	ERL	ERR	EXP	FRE
INP	INT	LOG	LPOS	PEEK
POS	RND	SGN	SIN	SPC
SQR	TAB	USRn	VARPTR	

String Functions

ASC	CHR$	FRE	HEX$	INSTR
LEFT$	LEN	MID$	OCT$	RIGHT$
SPACE$	STRING$	STR$	VAL	

Input/Output Functions

CVD	CVI	CVS	EOF	LOC
LOF	MKD$	MKI$	MKS$	

In order to receive Mostek BASIC, the attached Mostek BASIC Non-disclosure agreement should be signed and returned with each purchase order.

MOSTEK BASIC NON-DISCLOSURE AGREEMENT

The party below agrees that it is receiving a copy of Mostek BASIC for use on a single computer only, as designated on its registration card. The party agrees to fill out and mail in the registration card before making use of Mostek BASIC. The party agrees that all copies will be strictly safeguarded against disclosure to or use by persons not authorized by Mostek to use Mostek BASIC, and that the location of all copies will be reported to Mostek at Mostek's request. The party agrees that this agreement shall insure to the benefit of any third party holding any right, title or interest in the Mostek BASIC or any software from which it was derived.

"Party"_____ (Date)_____
Company: _____
Address: _____
Phone: _____

Return to:

Mostek Corp.
Microcomputer Dept.
Software Librarian
MS 501
1215 W. Crosby Road
Carrollton, Texas 75006

UCSD Pascal

FEATURES

- ☐ Implementation of UCSD PASCAL
- ☐ Page text editor
- ☐ File maintenance utility
- ☐ Z80 Assembler
- ☐ PASCAL Interpreter
- ☐ Linker
- ☐ Library Manager
- ☐ Calculator Utility
- ☐ Setup Utility
- ☐ Structured programming language
- ☐ Four basic data types: Boolean, integer, real, and character
- ☐ Scalar type definition: TYPE statement
- ☐ Structured data type definition: SET and RECORD statements
- ☐ Multi-dimensional array types
- ☐ Blocked structured: BEGIN...END
- ☐ Two types of subroutine definition: PROCEDURE and FUNCTION
- ☐ Nested IF...THEN...ELSE
- ☐ Multiple branch statement: CASE
- ☐ Repetitive statements: REPEAT...UNTIL, WHILE.. DO, FOR...DO
- ☐ Input/Output statements: READ, WRITE, READLN, and WRITELN
- ☐ Random file I/O statements: RESET, REWRITE, SEEK, GET, PUT, CLOSE

DESCRIPTION

Mostek UCSD PASCAL is an installation of UCSD PASCAL on Mostek's dual disk Microcomputer Development System. Its features are comparable to those PASCALS found on microcomputers, minicomputers, and large mainframes. Mostek PASCAL is supplied as a Level 2 software product. Level 2 designation implies that Mostek offers the product but Mostek will not provide technical support and Mostek will not automatically provide product updates.

Mostek PASCAL requires the following hardware configuration:

Mostek Dual Disk Development System with FLP-80DOS V2.0 and 64K bytes of RAM

Console device (i.e. Mostek CRT)

Line printer (i.e. Mostek Centronics LP)

Mostek PASCAL generates code for a pseudo-machine which is emulated at run time by an interpreter written in Z80 assembly language.

The standard Mostek PASCAL system intrinsics include:

BLOCKREAD	BLOCKWRITE	CLOSE	CONCAT
DELETE	DRAWLINE	DRAWBLOCK	EXIT
GOTOXY	FILLCHAR	HALT	IDSEARCH
INSERT	IORESULT	LENGTH	MARK
MOVELEFT	MOVERIGHT	REWRITE	RESET
POS	PWROFTEN	RELEASE	SEEK
SIZEOF	STR	TIME	UNITBUSY
UNITCLEAR	UNITREAD	UNITWAIT	UNITWRITE

EDITOR

The page text EDITOR is specifically designed for use with video display terminals. Upon invocation of the editor, the text file is displayed on the screen. By using editor commands and cursor movements, the file can be edited efficiently. For teletype terminals, a line-oriented text editor is included.

ASSEMBLER

The Assembler provides the capability to write and assemble a Z80 code which can be linked to PASCAL modules using the Linker. Thus, time-critical or memory restricted modules can be optimized by the user.

LINKER

The Linker provides the capability to combine pre-compiled files, which may have been written in PASCAL Z80 assembly language, or contained in a library.

LIBRARY UTILITY

The library utility provides the capability to generate a file of separately compiled or assembled modules. Modules which comprise a system can be grouped together in one library file. Library files can be used as input to the Linker, which will automatically link modules from the library that satisfy external references in the modules being linked.

CALCULATOR UTILITY

This utility allows the user to make quick calculations directly on the console device. Expressions of up to 25 different variables are possible. Also, functions such as MOD, SQRT, and trignometric expressions are available.

SETUP UTILITY

This utility allows the user to define a new console device to the Mostek PASCAL system. This task is accomplished via modification of a setup table which defines the various key codes and hardware-related parameters of the new console device.

Microcobol

FEATURES

LANGUAGE

- ☐ English-like syntax
- ☐ Data structures
- ☐ File structures
  RSAM — relative sequential access method
  ISAM — indexed sequential access method
- ☐ Arithmetic operations accurate to 18 decimal digit
- ☐ Interactive operation with display/keyboard terminal
- ☐ System routines : File conversion
  File copy
  Catalogue
  Date
  Sort
- ☐ Printer support
- ☐ Data management
- ☐ Storage protection
- ☐ Error handling and debugging
- ☐ Standard application packages available

DESCRIPTION

MICROCOBOL is a proven language and business operating system (BOS) for developing programs to run interactively in the small machine environment.
It gives the programmer : structured programming
extended subrouting
interactive debugging

MICROCOBOL enables you to fully benefit from all the superior features of the SYS-80 FT used as a management computer. The utility programs handle all the standard operations of business data processing and allow for many operations without having to write special programs.
Several standard application packages are available.
They include : Auto Index — a generalized data base system.
Auto Clerk — data file manager and report generator
Sales Ledger
Purchase Ledger
etc.

MOSTEK®

Z80 MICROCOMPUTER SYSTEMS

Application Interface Module (AIM-80E)

HARDWARE FEATURES

- ☐ Direct Interface with SDB-80E
- ☐ Single step/multistep with register trace
- ☐ Execution intercept (breakpoint) intercepts on memory access, port access, external trigger, event counter, or delay counter
- ☐ Push button execution intercept
- ☐ 256x32 history memory which samples Data Bus, Address Bus, $\overline{M1}$, $\overline{MREQ}$, $\overline{RD}$, $\overline{IORQ}$, and four external probes
- ☐ History memory clock selectable from $\overline{M1}$, $\overline{MREQ}$, $\overline{IORQ}$, or INTERRUPT ACKNOWLEDGE
- ☐ Selectable history memory clock conditions: read only, write only, DMA only, or external probe only (high or low)
- ☐ 8Kx8 ROM memory (firmware)

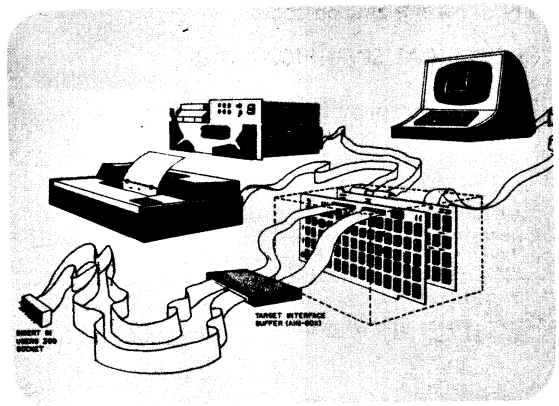
SOFTWARE FEATURES

- ☐ ROM resident mnemonic dis-assembler
- ☐ ROM resident RAM test for SDB or target RAM.

GENERAL DESCRIPTION

AIM-80E provides Z80 system debug assistance for both software and hardware via in-circuit emulation. (See Block Diagram)

Single step/multistep allows the programmer to trace through a program and display the CPU registers after each instruction. The execution intercept feature allows suspending program execution on the nth occurrence of an address or other specified condition. If the program has begun an unknown sequence, the intercept pushbutton will return the system to the single step mode. Single step and execution intercept (breakpoint) operate in RAM or ROM/PROM.



Hardware debugging is aided greatly by use of the 256x32 history memory which monitors bus transactions for a specified period. This information may then be displayed on the console. The data bus, address bus, $\overline{M1}$, $\overline{MREQ}$, $\overline{RD}$, $\overline{IORQ}$, and inputs from four probes are sampled and stored in the history memory upon every occurrance of the user specified clock ($\overline{M1}$, $\overline{MREQ}$, $\overline{IORQ}$, or interrupt acknowledge) qualified by the user specified conditions (read only, write only, DMA only, probe High only, or probe Low only). Upon the occurrence of the selected intercept, AIM-80E returns control to the system debug (DDT-80). The history memory may then be displayed (See Figure 1) with or without mnemonic dis-assembly.

USING THE AIM-80E

AIM-80E may be added directly to any SDB-80E system. All system bus signals are wired one to one between SDB-80E and AIM-80E. Voltage requirements for the AIM-80E are the same as for the SDB-80E. Programs may be debugged in SDB-80E memory space or with the target interface buffer box (AIM-80X) may be debugged in the target environment. Dynamic memory mapping allows target memory to be simulated using SDB-80E system RAM. All peripheral devices of the SDB-80E are still functional with the AIM-80E.

* Trademark of Mostek Corporation

SYSTEM FIRMWARE

To minimize the impact of the AIM-80E firmware is resident in one MK36000, 8Kx8 ROM. This firmware is completely compatible with DDT-80 firmware and includes five new commands for control of the AIM-80E. The interactive nature of the commands makes operation simple and avoids operator errors. The ROM resident dis-assembler makes correlation with the user's source listing easier and reduces the necessity of memorizing op codes.

ELECTRICAL SPECIFICATIONS

Operating Temperature Range 0°C to +50°C
Power Supply Requirements (Typical)
+12V + 5% @ 12mA
+ 5V ± 5% @ 1.0 Amp
Interface - SDB-80E compatible
Operating Frequency - 1-2.5 MHz (with SDB-80E)

MECHANICAL SPECIFICATIONS

Board Size: 250mm x 233.4mm x 18mm
Bottom Connector: Dual 64 pin Eurocard Connector DIN 41612 form D; A and C Pinned
Top Connectors: One 40 pin 3M ribbon
One 50 pin 3M ribbon

AIM-80E PRINT OUT EXAMPLE
(User entries underlined)

```
.I 2,T 2

TRIG ON (MREQ/IORQ/+/-) M
EVENT CNT (1-FF) 2
DELAY CNT (0-FF) 3
CLOCK ON (M1,MREQ,IORQ,INTA) MR IO
ONLY IF (RD/WR/DMA/H/L)
.E 0 2
0005 2421
```

Set intercept at address 0002H with trigger option

Trigger on $\overline{\text{MREQ}}$
After 2 occurrences
Delay 1 clock after trigger
Clock history (sample) memory on $\overline{\text{MREQ}}$ or $\overline{\text{IORQ}}$
No qualifying conditions selected
Begin execution at address 0002H
Intercept occurs at second occurrence of address 2 with a delay of one.

At this point the history memory contains the bus transactions which occurred before the intercept.

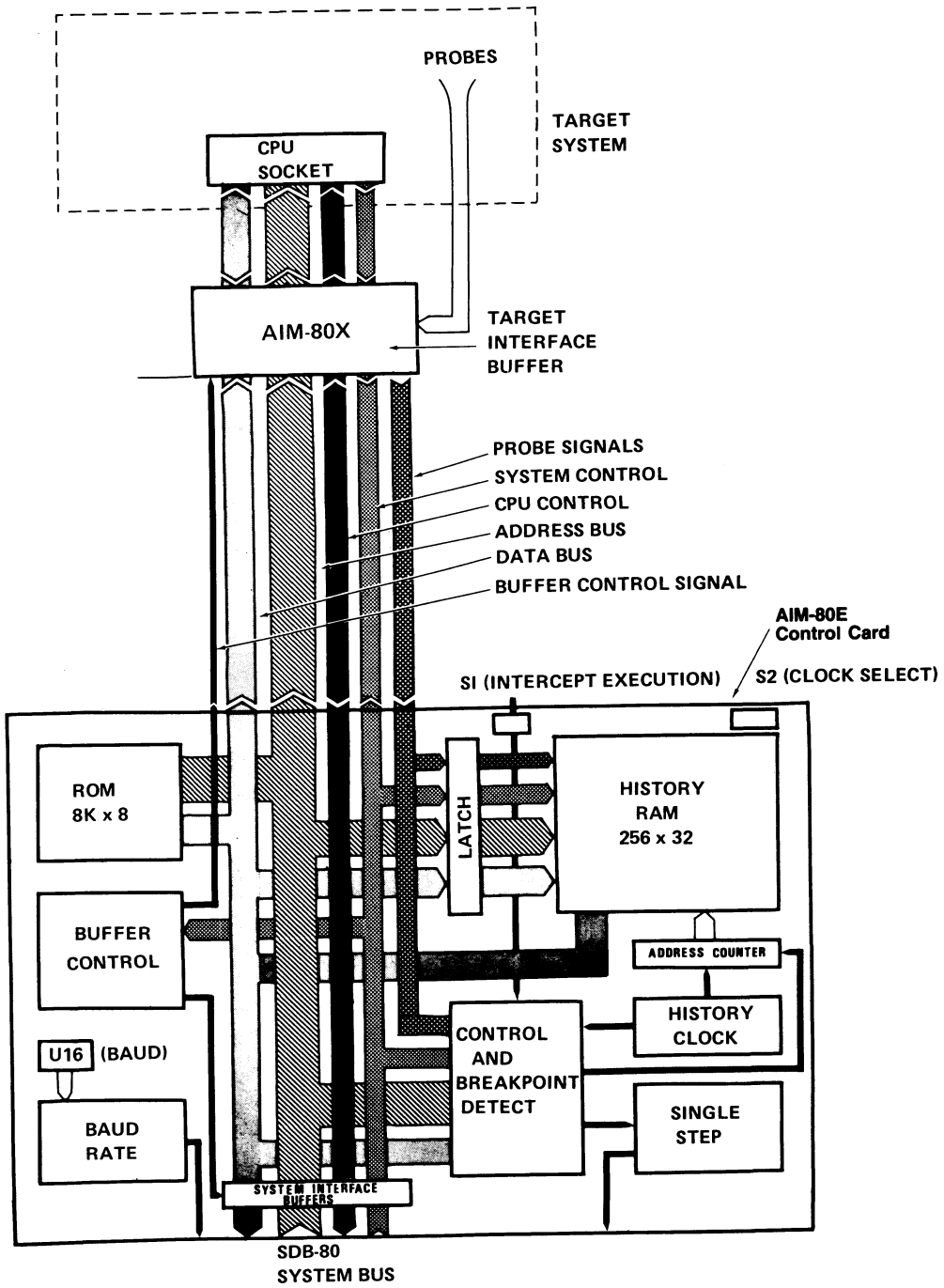
```
.T F,-B 2
```

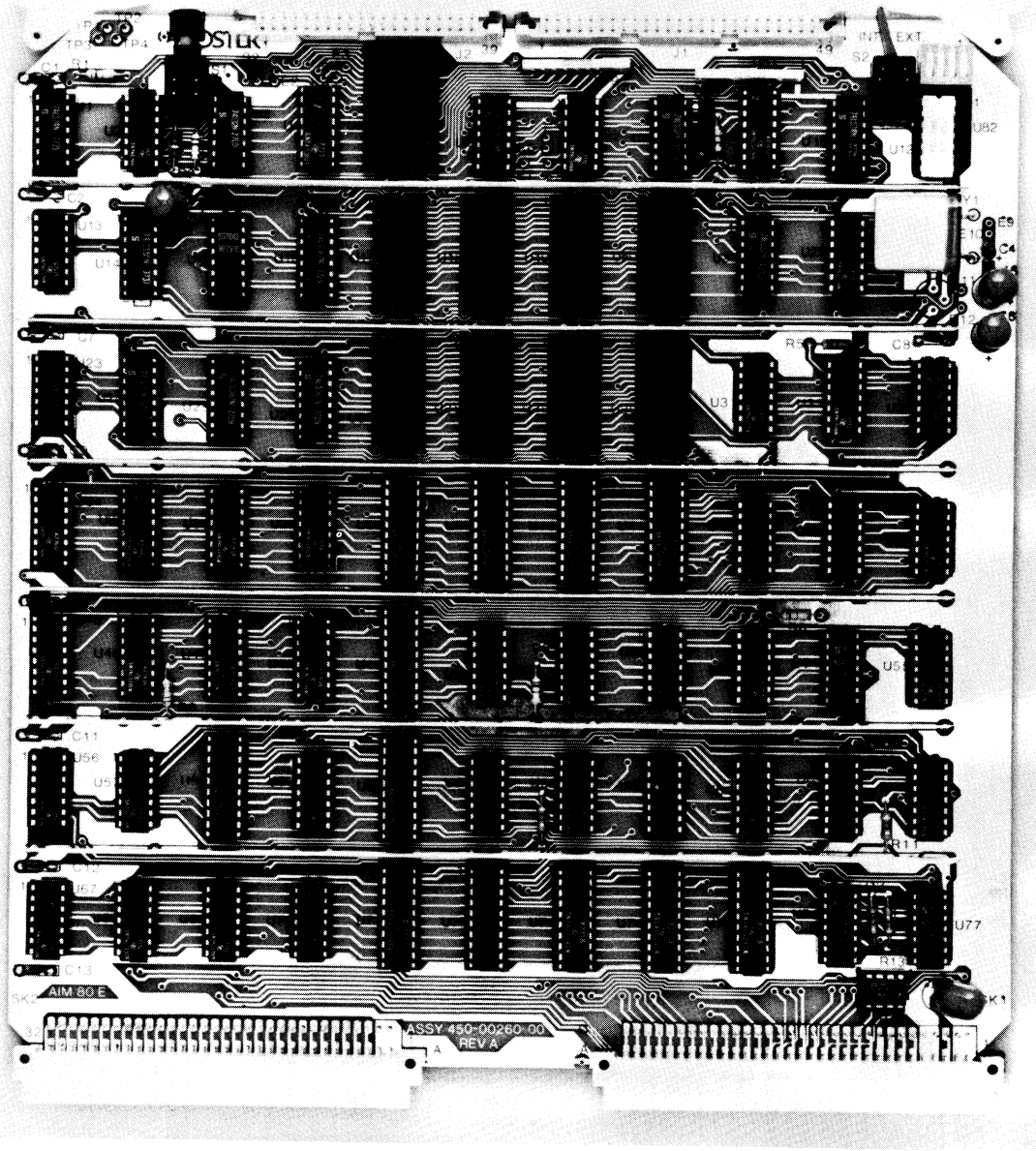
Trace for 15 lines - starting at offset -11 before trigger.
Offset from trigger
Memory Address Bus contents
Data Bus contents
Dis-assembled instructions
Control Bus
Probes (red, blue, green, yellow)

```
 OS  ADDR  DB      /M1 /MR /RD /IO  R  B  G  Y
-0B  0000  3E  LD   A,20H   0   0   0   1   1  1  1  1
-0A  0001  20              1   0   0   1   1  1  1  1
-09  0002  D3  OUT  (0),A   0   0   0   1   1  1  1  1   <- First occurrence of address 0002H
-08  0003  00              1   0   0   1   1  1  1  1
-07  3000  20              1   1   1   0   1  1  1  1
-06  0004  3C  INC  A       0   0   0   1   1  1  1  1
-05  0005  3C  INC  A       0   0   0   1   1  1  1  1
-04  0006  3C  INC  A       0   0   0   1   1  1  1  1
-03  0007  C3  JP   0002H   0   0   0   1   1  1  1  1
-02  0008  02              1   0   0   1   1  1  1  1
-01  0009  00              1   0   0   1   1  1  1  1
+00  0002  D3  OUT  (0),A   0   0   0   1   1  1  1  1   <- Trigger (OS = 0) (2nd occurrence
+01  0003  00              1   0   0   1   1  1  1  1       of address 0)
+02  2300  23              1   1   1   0   1  1  1  1
+03  0004  3C  INC  A       0   0   0   1   1  1  1  1   <- Delay 1 count after trigger

-FC ?.   <- No more data in history module
```

BLOCK DIAGRAM




AIM 80 E
ASSY 450-00260-00
REV A

MOSTEK®

3870 MICROCOMPUTER SYSTEMS

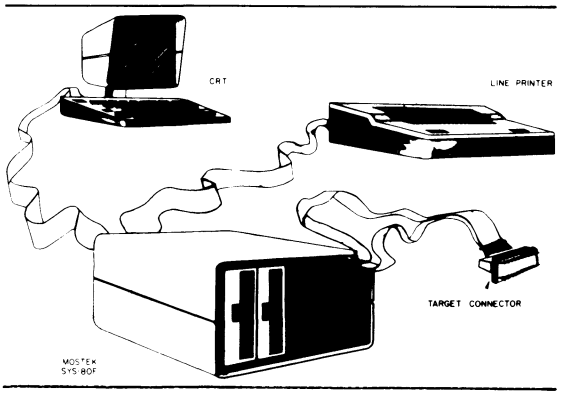
Application Interface Module (AIM-72E)

FEATURES

- ☐ Real time in-circuit emulation of Mostek's 3870 family of microcomputers, including MK3870, MK3872, and MK3876,
- ☐ Direct interface to Mostek's SYS-80F Dual Floppy Disk Microcomputer with ZAIM-72 software supplied on floppy disk.
- ☐ Standard Features Include:
  - Breakpoint insertion, memory display and modification, port display and modification, and single step.
  - Execution intercept from user keyboard with the ESCAPE key.
  - Debugging or emulation mode is selectable from the user's console.
  - Debugging of 3870 and F8 programs up to 8K long.

DESCRIPTION

AIM-72E (Application Interface Module) is a unique development aid for debugging MK3870 Series Microcomputer applications in the actual hardware and software configuration of the user's final system (referred to as the 'Target'.) To accomplish this, it is first necessary to emulate the Target ROM with RAM. This RAM must appear as ROM to the application while retaining the ability to be loaded, debugged, and modified using peripherals independent of the Target. It is the purpose of AIM, used in conjunction with the SYS-80F Disk Based Microcomputer to provide these capabilities. With AIM-72E, all of the peripheral and debugging capabilities of the user's development system may be applied directly to either the prototype or final production configuration of any MK3870, MK3872, or MK3876 application; no modifications to the user's hardware, software, or mechanical package are required.



USING AIM

The pictorial diagram above shows how AIM-72E would typically be used during system development. Because the AIM-72E is an exact functional emulation of the MK3870 family, it may be directly inserted into the 3870, 3872, or 3876 socket in the target system. Also, since the Target can be a production version of the user's application, product revisions and enhancements may be easily implemented. As shown in the diagram, the AIM Board is mounted in the card cage of the user's development system. It is the purpose of the SYS-80F to provide the user with the means for accessing and controlling the target system (via the AIM Board) during the program development phase. This provides access to all the debugging software and peripherals of the development system without having to introduce any perturbations to the Target system environment. AIM-72E does not affect the peripheral expansion capabilities of the development system.

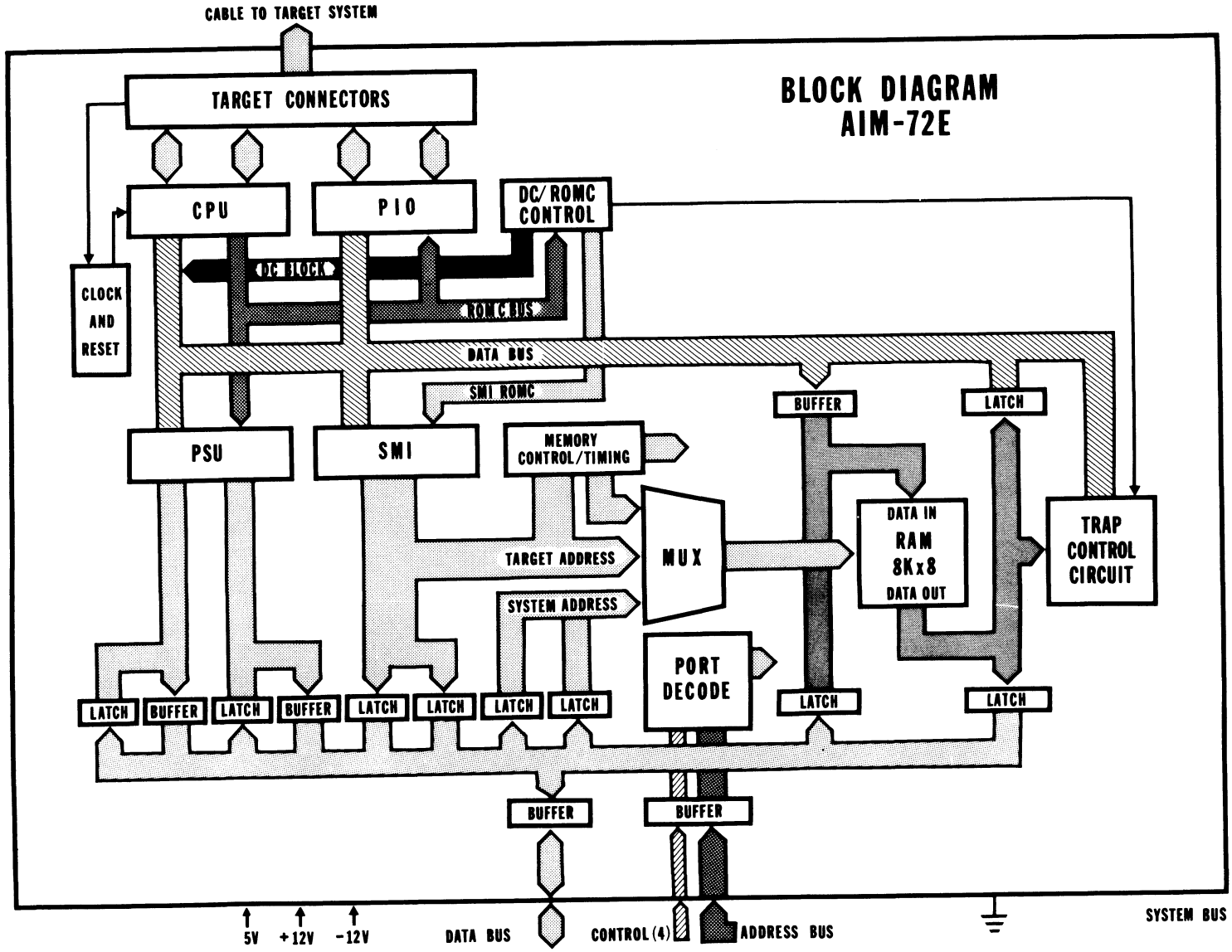
SPECIFICATIONS

Operating Temperature Range. . . . . 0°C to 50°C

Power Supply Requirements
+5V ± 5% @ 1.5A max.
+12V ± 5% @ 100mA max.
-12V ± 5% @ 30mA max.

Board Size. . . . . .233.4 mm x 257.62 x 25 mm

Connectors/Cables:
40-Pin Ribbon Cable (24" long)


CABLE TO TARGET SYSTEM
BLOCK DIAGRAM
AIM-72E
TARGET CONNECTORS
CPU
PIO
DC/ROMC CONTROL
CLOCK AND RESET
DC BUS
ROMC BUS
DATA BUS
SMI ROMC
PSU
SMI
MEMORY CONTROL/TIMING
BUFFER
LATCH
TARGET ADDRESS
MUX
DATA IN
RAM
8K x 8
DATA OUT
TRAP CONTROL CIRCUIT
SYSTEM ADDRESS
PORT DECODE
LATCH
BUFFER
BUFFER
5V
+12V
-12V
DATA BUS
CONTROL (4)
ADDRESS BUS
SYSTEM BUS

BLOCK DIAGRAM DESCRIPTION

As shown in the block diagram, the AIM-72E contains all the functional elements necessary to emulate MK3870 Series Microcomputers. Target Ports are emulated with the CPU and PIO Ports. Target ROM and RAM are emulated with the 8K x 8 RAM which can also be accessed directly by the control system via the system bus connector. System memory accesses are transparent to the Target system execution. Thus, there is no impact on Target execution timing. The Target memory map can be controlled from the system allowing 2K, 4K or 8K Bytes of memory to be available in the Target System. Debug firmware in a PSU on the AIM-72E interfaces with the system to implement the breakpoint, single step and other functions. Trap control circuitry allows the use of a single byte breakpoint, providing complete flexibility when using break points in tight programming loops. Execution is at full speed, determined only by the user's crystal frequency - no speed reduction is introduced by the AIM's operating system. The AIM-72E clock may be emulated for the Target system from an onboard crystal oscillator or from the SYS-80F clock.

MULTI 3870 SERIES APPLICATIONS

Up to eight AIM-72E boards may be installed in one control system with each AIM-72E used to emulate a different MK3870 Series Microcomputer. The debug functions on each AIM-72E may be enabled one at a time and each program developed until all Target programs are functional. Only one AIM-72E may be in the active debug mode at a time; other AIM-72E's will be in the Emulator mode.

ZAIM-72 SOFTWARE DESCRIPTION

ZAIM-72 is the software designed to operate the AIM-72E board on Mostek's SYS-80F Dual Floppy Disk Microcomputer. It is supplied on a standard FLP-80DOS diskette. The software has the same command structure as other Mostek debuggers. The commands available with ZAIM-72 are summarized below. Designations s, f, and d stand for operands.

,A s, f	Assign data byte f to target memory location s.
,B s	Set a breakpoint at target memory location s. Up to 8 breakpoints can be set at once.
,C s, f, d	Copy the target memory block s to f to target memory starting at d.
,E s	Execute target program at location s.
,F s, f, d	Fill target memory locations s through f with data d.
,G s	Get binary file s and load it into Target memory.
,H	Hexadecimal arithmetic.
,I	Reinitialize target system.
,L s, f, d	Locate data d in target memory at location f. .
,M s	Display and update target memory at location s.
,M s, f, d	Tabulate target memory locations s through f. Option d specifies additional printout of ASCII characters or disassembly.
,O s	Set relative offset equal to s for all address operands.
,P s	Display and update target port number s.
,Q	Quit and return to FLP-80DOS Monitor.
,R s, f	Display target registers, Option s allows a heading to be printed and option f specifies the number of scratchpad registers to be displayed.
,S s, f	Single step starting at target location s for f number of steps.
,V s, f, d	Verify target memory block s through f against target memory block starting at location d.

Target system programs are developed using the Mostek SYS-80F Cross Assembler for 3870/F8 Microcomputers (FZCASM-MK79075). Then ZAIM-72 is used to debug the completed program on the user's Target system. The software features multiple breakpoints, single step, and in-line disassembly. Target system memory, ports, and registers may be displayed and updated.

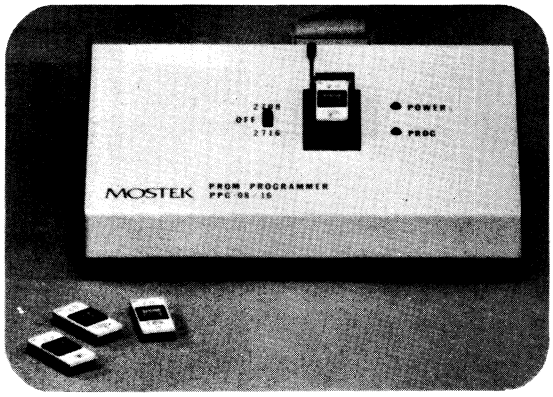
MOSTEK®

MICROCOMPUTER SYSTEM

Prom Programmer (PPG-8/16)

FEATURES

- ☐ Programs, reads, and verifies 2708, 2758, and 2716 type PROMs
- ☐ Directly interfaces to SDB 50/70, SDB-80 and SYS-80F
- ☐ Driver software included
- ☐ Zero insertion force socket
- ☐ Power and programming indicators



DESCRIPTION

The PPG-8/16 PROM Programmer is a peripheral which provides a low-cost means of programming 2708, 2758, or 2716 PROMs. It is directly compatible with MOSTEK's F8 Software Development Board (SDB 50/70), Z80 Software Development Board (SDB-80), MD Series and Z80 Microcomputer Development System (SYS-80FT). The PPG-8/16 has a generalized computer interface (two 8-bit I/O ports) allowing it to also be controlled by other types of host computers with user-generated driver software. A complete set of documentation is provided with the PPG-8/16 which describes the internal operation and details user's operating procedures. See also block diagram.

The PPG-8/16 is available in a metal enclosure for use with SYS-80FT or any Mostek system. Cable is included.

SOFTWARE DESCRIPTION

The driver software available for the SDB-80, SDB-50/70, and SYS-80FT accomplishes four basic operations. These are: (1) loading data (object tapes for SDB-80 and SDB-50/70, or binary files for the SYS-80FT into computer memory, (2) reading the contents of a PROM into computer memory, (3) programming a PROM from the contents of the computer memory, and (4) verifying the contents of a PROM with the contents of the computer memory.

The driver software is provided in the form of paper-tape for both the SDB-50/70 and the SDB-80. An optional accessory is a TI Silent 700 compatible cassette object tape containing control software for the SDB-50/70 and SDB-80. Users of MOSTEK's SYS-80FT who wish to upgrade their systems with a PPG-8/16 will find the driver software on their system diskette (version 2.0 or later). The user documentation provided with the PPG-8/16 fully explains the programming procedures.

SPECIFICATIONS

INTERFACE

SYS-80FT : cable supplied
Other : see ordering information

POWER REQUIREMENTS

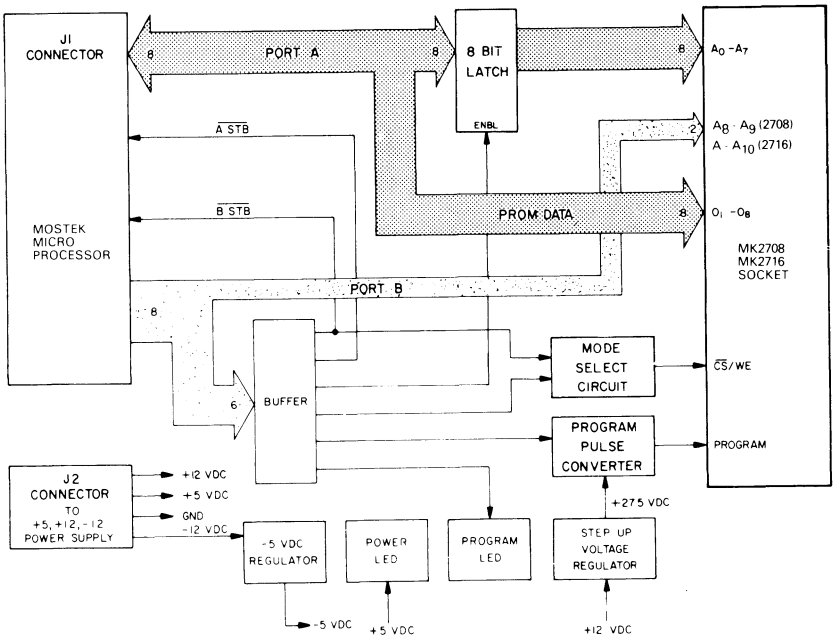
+ 12 VDC at 250 mA typical
+ 5 VDC at 100 mA typical
-12 VDC at 50 mA typical

OPERATING TEMPERATURE ..... 0°C - 50°C

PROGRAMMING TIME

2708 ..... 150 seconds
2758 ..... 55 seconds
2716 ..... 110 seconds

PPG 8/16 BLOCK DIAGRAM



ORDERING INFORMATION

DESIGNATOR	DESCRIPTION	PART NO.
PPG-8/16	PROM Programmer for 2708/2758/2716 PROMs with Operations Manual and paper tape drivers for SDB-50/70 and SDB-80. Includes cables for SYS-80FT, SYS-50/70, ZDS-80.	MK79082
	PPG 8/16 Interface Cable for SDB-50/70	MK79046
	PPG-8/16 Interface Cable for MDX-PIO	MK77957
SWD-2	PPG-8/16 Object programs for SDB-50/70 & SDB-80 on Silent 700 cassette tape	MK79084
	PPG-8/16 Operations Manual	MK79603

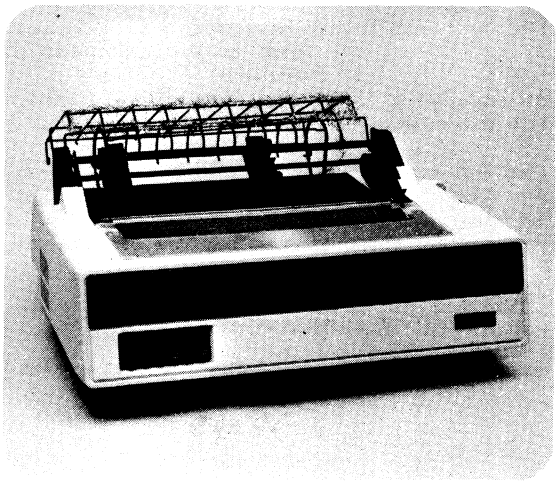
MOSTEK®

MICROCOMPUTER SUPPORT

Line Printer

FEATURES

- ☐ Interfaces directly to SYS-80F
- ☐ Prints 120 characters per second
- ☐ Up to 132 characters per line
- ☐ Prints original plus five copies
- ☐ Character elongation
- ☐ 8 inches per second paper slew rate
- ☐ Ribbon cartridge
- ☐ 7x7 dot matrix, 64 character ASCII
- ☐ Tractor feed/Pin feed platen
- ☐ Parallel interface



DESCRIPTION

The MOSTEK line printer is a state-of-the-art microprocessor controlled, dot-matrix line printer that prints at the rate of 120 characters per second. The printer has a maximum print width of 132 characters with a horizontal format of 10 characters per inch and 6 lines per inch vertical. Elogated (double width) characters are software selectable.

The Mostek line printer interfaces directly to the AID-80F or SYS-80F Microcomputer Systems and can be interfaced easily to other computer systems supporting parallel I/O.

SPECIFICATIONS

Print Performance - Minimum Throughout

Printer Model	Print Speed (cps)	Max. Print Width	10Char/ Line (lpm)	80Char/ Line (lpm)	132Char/ Line (lpm)
702	120	13.2 in. (335mm)	260	74	47

Characters
7x7 dot matrix, 64 character U.S. ASCII

Format
10 Characters per inch horizontal
6 Lines per inch vertical
Elongated (double width) characters software selectable

Forms Handling
Tractor feed, for rear or bottom feed forms

8 ips slew rate
Usable paper 4 in. (102mm) to 17.3 in. (439mm) width
Paper tension adjustment

Ribbon System
Ribbon cartridge
Continuous ribbon 9/16 in. (14mm) wide, 20 yards (18.3 meters) long.
Mobius loop allows printing on upper and lower portion on alternate passes.

Panel Indicators
Power On: Indicates AC power is applied to printer.
Select: Indicates printer can receive data.
Alert: Indicates operator-correctable error condition.

Operator Controls
Select/deselect
Forms thickness

Top of form
Horizontal forms positioning
Vertical forms positioning
Power ON/OFF
Single line feed
Paper empty override
Self-test

Internal Controls

Auto motor control: turns stepping motors off when no data is received.
Electronic top of form: allows paper to space to top of form when command is received.
Preset for 11 in. (279mm) or 12 in. (305mm) forms
Opt. VFU must be used for other form lengths.

Data Input

7 or 8 bit ASCII parallel; microprocessor electronics; TTL levels with strobe.
Acknowledge pulse indicates that data was received.

Electrical Requirements

50/60 Hz, 115/230 VAC;+10%/-15% of Nominal Tappable Transformer (100, 110, 115, 120, 200, 220, 230, 240 VAC).

Physical Dimensions

Model 702

Weight: 60 lbs. (27 Kg)
Width: 24.5 in. (622mm)
Height: 8 in. (203mm)
Depth: 18 in. (457mm)

Temperature

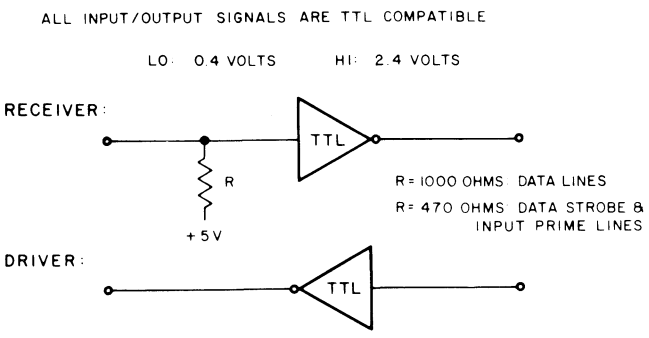
Operating: 40° to 100°F (4.4° to 37.7°C)
Storage: -40° to 160°F (-40° to 71.1°C)

Humidity

Operating: 20% to 90% (No condensation)
Storage: 5% to 95% (No condensation)

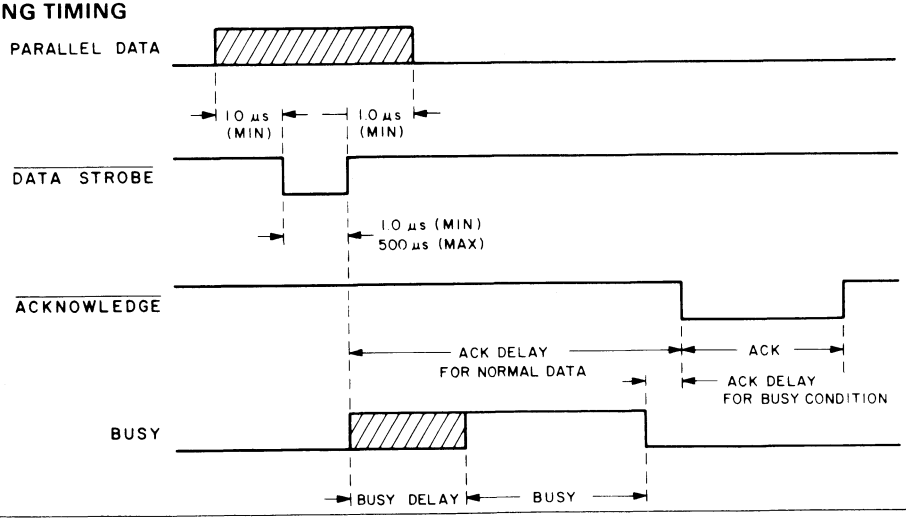
INTERFACING

INTERFACE DRIVERS AND RECEIVERS



CONNECTOR: AMPHENOL 57 40360 SERIES, 36-PIN
(CENTRONICS 31310019)

INTERFACING TIMING



INTERFACE TIMING CONT'D

NORMAL DATA INPUT TIMING	ACK DELAY ACK	2 - 6 μsec. 4 μsec.
BUSY CONDITION TIMING	BUSY DELAY ACK DELAY ACK BUSY DURATION Line Feed Vertical Tab (1-in.) Form Feed (11-in.) Delete Bell Select* Deselect Printer	0 - 1.5 μsec. 1 - 6 μsec. 4 μsec. 350 - 500 μsec. 135 - 145 msec. 1.48 - 1.50 sec. 160 - 400 μsec. 0 0 - 1.5 μsec. Until Printer is selected 8.33 msec./char., plus 148 msec. non-printing time/line.

*No busy if inhibit prime on select option is used

DISK BASED DEVELOPMENT SYSTEM/OEM MICROCOMPUTER AND SOFTWARE PACKAGES

ORDERING INFORMATION

DESIGNATOR	DESCRIPTION	MOSTEK ORDER No.
SYS-80FT	Complete floppy disk system with 58K RAM, 8-8 bit I/O ports, video display and keyboard, 2 floppy disks, FLP-80 DOS and development system software[1]	MK78042
MACRO-80[1]	Powerful macro assembler for the Z80.	MK78165
MACRO-70[1]	Powerful macro assembler for the 3870/F8.	MK79085
FORTRAN[1]	Fortran IV high level language compiler for FLP-80 DOS. Requires 48K bytes of RAM. Includes Operation Manual. MK79643	MK78158
BASIC[1]	17K disk Basic interpreter high level language for FLP-80 DOS. Requires 32K bytes of RAM. Includes Operation Manual. MK79623	MK78157
PASCAL[1]	High level language for FLP-80 DOS. Requires 64K bytes of RAM. Includes documentation. UCSD Pascal Operation Manual MK79761	MK78179
MICROCOBOL	High level general use computer programming language. Includes Manuals.	
AIM-80E	Application Interface Module for 3880 CPU emulation; includes 8K byte firmware package. Installation guide MK78559	MK78106
AIM-72E	Application Interface Module for 3870 family devices. Includes the Operation Manual MK79579 and ZAIM-72 (on diskette) software.	MK79077
PPG-8/16	PROM programmer for 2708, 2758, 2716; with Installation Guide MK79603 includes cable for interface with SYS-80FT	MK79082
LINE PRINTER	120 CPS matrix printer and cable for connection to SYS-80F.	MK78150C

NOTE
Delivery subject to licensing agreement procedure (see page 133).
License must be submitted with order.

STANDARD SOFTWARE LICENSING AGREEMENT

All Mostek Corporation products are protected by copyright and are sold on condition that the Purchaser agrees to the following terms :

1. The Purchaser agrees not to sell, provide, give away, or otherwise make available to any unauthorized persons, all or any part of, the Mostek software products listed below; including, but not restricted to : object code, source code and program listings.
2. The Purchaser may at any time demonstrate the normal operation of the Mostek software product to any person.
3. The Purchaser may not incorporate Mostek software products or any portions thereof into his/her own products without written permission of Mostek.
4. Purchaser shall be notified by Mostek of all updates and modifications made by Mostek for a one-year period after purchase of said Mostek software product. Updated and/or modified software and manuals will be supplied at the current cataloged prices.
5. In no event will Mostek Corporation be held liable for consequential damage resulting from the use or misuse of any Mostek software product.
   Due to the expense of generating software, Mostek must have a copy of this agreement, signed by the customer, before shipping certain software products to the customer. Additionally, the above rules apply to all Mostek software, regardless whether or not a signed license has been received. Signing the license confirms the customer's understanding, and allows Mostek to comply with point 4.

Order Number Description

1) _____

2) _____

3) _____

4) _____

5) _____

Agreed to :

PURCHASER	MOSTEK CORPORATION
By : _____	By : _____
Title : _____	Title : _____
Date : _____	Date : _____
Address : _____	

MOSTEK MARKETING OFFICES

BELGIUM	:	MOSTEK INTERNATIONAL 150, Chaussée de La Hulpe, B-1170 Brussels	Tel: 02/660.69.24 Tlx: 620.11
FRANCE	:	MOSTEK FRANCE S.A.R.L. 30, Rue du Morvan, Silic 505, F-94623 Rungis Cedex	Tel: (1)687.34.14 Tlx: 20.40.49
GERMANY	:	MOSTEK GMBH, (Stuttgart) Talstraße 172, D-7024 Filderstadt 1 MOSTEK GMBH, (Hamburg) Friedlandstraße 1, D-2085 Quickborn MOSTEK GMBH, (München) Zaunkönigstraße 18, D-8012 Ottobrunn	Tel: 0711/70.10.45 Tlx: 7.255.792 Tel: 04106/2077-2078 Tlx: 21.36.85 Tel: 089/609.10.17
ITALY	:	MOSTEK ITALIA S.P.A. 10, Via G. da Procida, 1-20149 Milano	Tel: 02/349.26.96-318.53.37 Tlx: 33.36.01
SCANDINAVIA	:	MOSTEK SCANDINAVIA AB Magnusvägen 1, S-17531 Järfälla	Tel: 0758/343.38-343.48 Tlx: 12997
UNITED KINGDOM	:	MOSTEK UK LTD. Masons House, 1-3 Valley Drive, Kingsbury Rd, London NW9	Tel.01/204.9322 Tlx: 259.40

EUROPEAN SALES REPRESENTATIVES AND DISTRIBUTORS

AUSTRIA
Transistor-Vertriebs GmbH
Auhofstrasse 41A
A-1130 Vienna
0222-829.45.12
Telex: 13738

BELGIUM
Sotronic
14 Rue Père De Deken
B-1040 Brussels
02-736.10.07
Telex: 25141

DENMARK
Semicap APS
Gammel Kongevej 148.5
DK-1850 Copenhagen
01-22.15.10
Telex: 15987

FINLAND
S. W. Instruments
Karstulantie 4B
SF-00550 Helsinki 55
8-0-73.82.65
Telex: 122411

FRANCE
Société COPEL
Rue Fourny, Z.I.
B.P. 22, F-78 530 BUC
(1) 956 1018
Telex: 696379

P.E.P.
4 Rue Barthélemy
F-92120 Montrouge
(1)-735.33.20
Telex: 204534

Scaib
80 Rue d'Arcueil
SILIC 137
F-94150 Rungis Cedex
(1) 687.23.13
Telex: 240674

Sorhodis
150-152 Rue A. France
F69100 Villeurbanne
(78) 850044
Telex: 380181

GERMANY
Neye Enatechnik GmbH
Schillerstrasse 14
D-2085 Quickborn
0 4106-61.22.95
Telex: 213.590

Dr Dohrenberg
Bayreuther Strasse 3
D-1000 Berlin 30
030-213.80.43
Telex: 184860

Raffel-Electronic GmbH
Lochnerstrasse 1
D-4030 Ratingen 1
0 2102-280.24
Telex: 8585180

Siegfried Ecker
Königsberger Strasse 2
D-6120 Michelstadt
0 6061-2233
Telex: 4191630

Matronic GmbH
Lichtenberger Weg 3
D-7400 Tubingen
07071-24331
Telex: 7262879

Dema-Electronic GmbH
Blütenstrasse 21
D-8000 München 40
0 8928 8018
Telex: 29345

ITALY
Comprel s.r.l.
Viale Romagna 1
I-20092 Cinisello Balsamo
02 928 0809/928 03.45
Telex: 332484

Branch offices in
Bologna, Firenze,
Lavagna, Loreto,
Padova, Roma, Torino

THE NETHERLANDS
Nijkerk Elektronika BV
Drentestraat 7
1083 HK Amsterdam
020.428.933
Telex: 11625

NORWAY
Hefro Teknisk A/S
Postboks 6596
Rodelokka
Oslo 5
02-38.02 86
Telex: 16205

SWEDEN
Interelko AB
Strandbergsg. 47
S-11251 Stockholm
08-13.21.60
Telex: 10689

SPAIN
Comelta S.A.
CiaElectronica Tecnicas
Aplicadas
Consejo de Ciento 204
Entlo 3A
Barcelona 11
03-254.66.07/08
Telex: 51934

Comelta SA
Orense 9-50 fZQ
Madrid 20
34-3 254 98.31

SWITZERLAND
Memotec AG
CH-4932 Lotzwil
063-28.11.22
Telex: 68636

UNITED KINGDOM
Celdis Limited
37-39 Loverock Road
Reading
Berks. RG 31 ED
0734-58.51.71
Telex: 848370

Distronic Limited
50-51 Burnt Mill
Elizabeth Way
Harlow
Essex CM 202 HU
0279-32.497/39.701
Telex: 81387

A. M. Lock Co. Ltd.
Neville Street
Chadderton
Oldham
Lancashire
061-652.04.31
Telex: 669971

Pronto Electronic Systems Ltd.
645 High Road
Seven Kings
Ilford
Essex IG3 8RA
01-599.30.41
Telex: 24507

YUGOSLAVIA
Chemcolor
Inozemma Zastupstva
Proleterskih brigada 37-a
41001 Zagreb
041-513.911
Telex: 21236
Branch office in Beograd

For all other countries :
MOSTEK INTERNATIONAL — Tel: 02/660.69.24
150, Chaussée de la Hulpe, 1170 Brussels / BELGIUM — Tlx: 62011
or
MOSTEK CORPORATION — Tel: 214/242-0444
International Dept. — Tlx: 730423
1215 West Crosby Road, Carrollton, Texas 75006, USA

U.S. $3.00

MOSTEK®

PRINTED IN ENGLAND — SEPTEMBER 1979 — MK 79781

MOSTEK MICROCOMPUTER SYSTEMS DATA BOOK