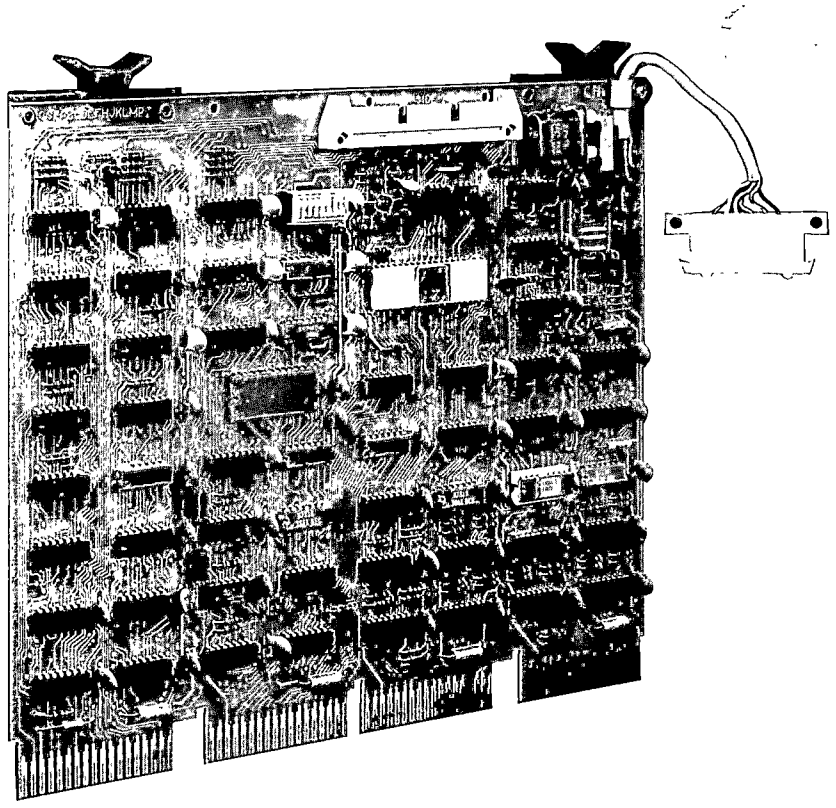
DIGITAL EQUIPMENT CORPORATION

MODULE DATA SHEET

LOGIC PRODUCTS

JULY 1974

M7341 Processor Module



DESCRIPTION

The M7341 Processor Module is a compact, low-cost, general-purpose microprocessor manufactured by DIGITAL to satisfy industrial, commercial, and laboratory applications. This module is one of the Microprocessor Series (MPS) modules, a group of four M Series modules designed to perform a range of industrial, laboratory and commercial control and decision-making activities not previously considered economic subjects for automation. These modules are:

- M7341 Processor Module
- M7344 Read/Write Memory Module
- M7345 Programmable Read-Only Memory Module
- M7346 External Event Detection Module

Used together, these modules can form low-cost digital control systems which exhibit characteristics normally atrributed to more costly minicomputer-based systems. MPS systems can serve as dedicated controllers, operate as a CPU in intelligent terminals, perform laboratory data acquisition and analysis tasks, and automate a host of industrial processes.

The M7341 Processor Module (PM) contains a single chip MOS/LSI microprocessor along with the integrated logic and control circuitry necessary to operate as a parallel 8-bit central processing unit. This microprocessor support logic consists of an adjustable 500 kHz two-phase symmetrical clock; a four-channel input multiplexer; data, memory, and address bus gating; I/O control logic; interrupt recognition logic; and a universal asynchronous receiver/transmitter driven by a 1.76 kHz clock.

The single-chip microprocessor contains a bidirectional data port, complete instruction decoding logic, an arithmetic unit, a state counter, an accumulator, an address stack, six general registers, and memory and I/O timing and control logic.

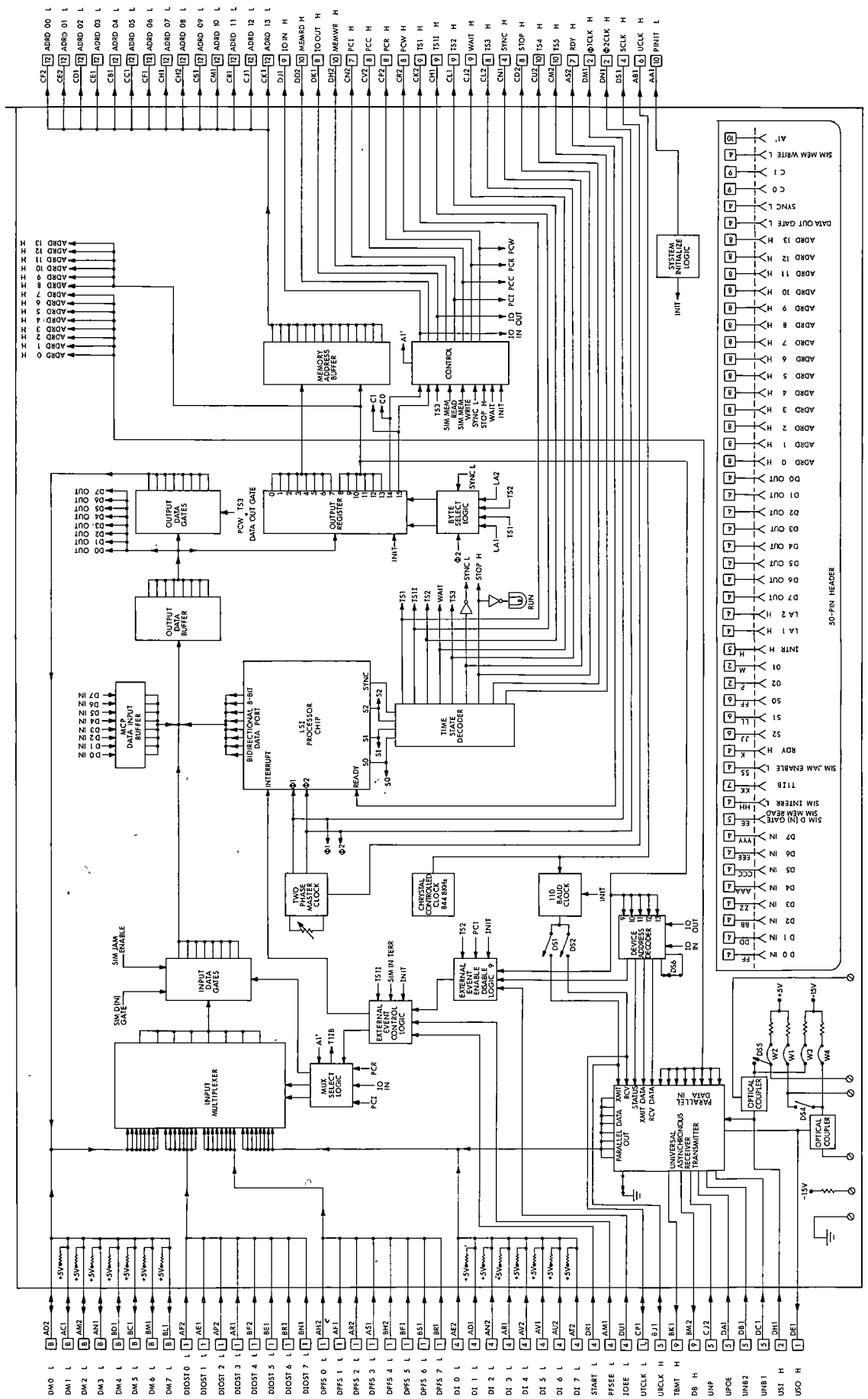


FIGURE 1. M7341 LOGIC BLOCK DIAGRAM

The Processor Module executes 48 basic instructions which can be functionally grouped into five categories:

- Register operations
- Accumulator operations
- Program counter and stack control operations
- Input/output operations
- Machine operations

To support the development of MPS user programs, DIGITAL furnishes the Microprocessor Series Software Tools (MPSST) package which includes the following routines:

Microprocessor Language Assembler (MLA)
Microprocessor Language Editor (MLE)
Microprocessor Read-Only Memory Programmer (MRP)
Microprocessor Host Loader (MHL)
Microprocessor Debugging Program (MDP)

FEATURES

- 8-bit parallel microprocessor completely contained on a single quad module.
- Uses proven commercially-available LSI microprocessor chip.
- Complete instruction decoding and control.
- Instruction execution time ranges from 12 to 44 μs.
- User program development supported by comprehensive software tools package.
- Contains full duplex universal asynchronous receiver/transmitter for serial communication with external data sources.
- Separate internal clock for 110-baud receiver/transmitter operation.
- Receiver/transmitter switchable to external clock for operation at up to 9600 baud.
- Serial data communication in either TTL or current-loop mode.
- Register complement includes 8-bit accumulator, memory address registers, and six 8-bit general purpose registers.
- Contains a stack to implement seven-level subroutine nesting.
- Executes 48 arithmetic, logical, register transfer, and memory reference instructions.
- Arithmetic unit capable of multiple precision arithmetic.
- Module can access an address space of up to 16K words.
- External event interrupts can be enabled and disabled under program control.
- Four multiplexed input ports for memory and I/O data, external event restart and power-fail addresses, and halt instructions.
- Module can address eight input peripheral devices and 24 output peripheral devices.
- TTL-compatible inputs and outputs.
- Module plugs into standard DEC H803 wire-wrap connector block backplanes.

APPLICATIONS

Since the M7341 Processor Module is a general-purpose parallel 8-bit programmable microprocessor, the scope of applications is essentially the same as that encompassed by conventional minicomputers. In conjunction with MPS memory, and external event detection modules as well as various M Series modules, the M7341 Processor Module can provide low-cost, highly efficient solutions to a wide range of industrial, commercial and laboratory applications.

Figure 2 depicts a typical example of how an MPS system can be structured for applications requiring a mass storage capability. The various application areas listed below further point out the versatility of potential MPS systems:

- Industrial Control
  - —Machine tool control
  - —Material flow
- Process Control
  - —Batch mixing
  - —Furnace monitoring
  - —Batch weighing
- Small Laboratory Automation
  - —Analog and digital instrument data acquisition
  - —Blood analyzers
- Data Communications
  - —Data concentrators
  - —Communications processors
  - —Minicomputer preprocessors
  - —Intelligent terminals
- Business Machines
  - —Optical character recognition
  - —Automatic banking
  - —"Smart" copying machines
- Health, Education, and Welfare
  - —Environmental control of large buildings
  - —Automatic teaching machines
  - —Remote pollution-monitoring systems
- Transportation
  - —Traffic signal controllers
  - —Vehicle recognition scanners
  - —Traffic flow monitoring

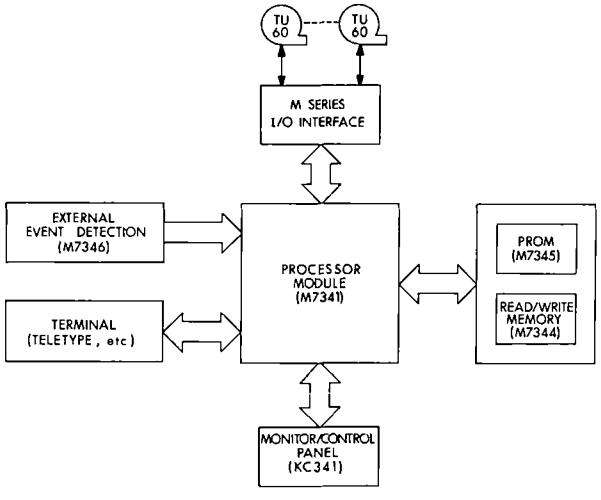


FIGURE 2. TYPICAL EXAMPLE OF AN MPS SYSTEM

SPECIFICATIONS SUMMARY

Performance Specifications

Operating Speed @ 500 kHz
Two-Phase Clock Period: 2μs
State Time: 4μs
Instruction Time: 12 to 44μs

Word Size
Data: 8-bit Word
Instruction: 1, 2, or 3 8-bit Words
Address: 14 bits

Input Data Ports
Memory Data: 8 bits
Peripheral Data: 8 bits
Power Fail/Stop: 8 bits Multiplexed
I/O External Event Interrupt/Start: 8 bits

Input/Output Lines
Memory Data: 8 bits bidirectional
Memory Address: 14 bits, output only
Peripheral Data: 8 bits input and output
Peripheral Address: 5 bits, output only
Communication Lines: 2, TTL or 20 mA current loop, active or passive.
Baud Rate
With Internal Clock: 110 baud (1.76 kHz)
With External Clock: 9600 baud (153.6 kHz), maximum (TTL)

Instruction Repertoire
Number of Basic Instructions: 48
Instruction Categories
Register Operation Instructions
Accumulator Operation Instructions
PC and Stack Control Instructions
I/O Instructions
Machine Instructions

MCP-PM Interface

Cable Length: Φ Frequency (kHz)	Recommended BC05W cable length	
	Feet	Meters
500	8, 6, 4 or 2	2.4, 1.8, 1.2 or .6
600	6, 4 or 2	1.8, 1.2 or .6
700	4 or 2	1.2 or .6

Connector/Plug Types: 50-pin PC board connector/header
Cable Type: 50-conductor, flat, shielded

Electrical Specifications

Input Logic Levels
Logical Low: 0.8 Vdc max.
Logical High: 2.0 Vdc min.
Output Logic Levels
Logical Low: 0.4 Vdc max.
Logical High: 2.4 Vdc min.
Power Consumption: 1.6 A @ +5 V, 150 mA @ −15 V; 10.25 W

Mechanical Specifications
Board Type: Quad-height, extended-length, single-width
Dimensions: 10.436 x 8.50 x 0.50 inches (26.5 x 21.6 x 1.27 cm)

Environmental Specifications
Ambient Temperature
Operating: 5 to 50°C; (41 to 122°F)
Nonoperating: −40 to 66°C; (−40 to 150°F)
Humidity: 10 to 95%, noncondensing
Altitude
Operating: 0 to 8,000 ft. (2.4 km)
Nonoperating: 0 to 30,000 ft. (9.1 km)

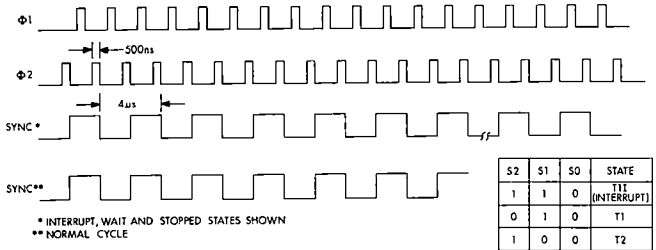
FUNCTIONS

Processor Module Timing

As shown in Figure 1, the basic timing signals for the Processor Module are produced by the two-phase clock. These signals, labeled Φ1 and Φ2, are symmetrical, nonoverlapping, positive-going clock pulses which drive the processor chip state counter.

This state counter controls all activity internal to the processor chip and produces the output signals S0, S1, S2, and SYNC. The timing signals available for external use are Φ1CLK H, Φ2CLK H, and SYNC H. SYNC H, along with S0, S1, and S2, serves to define processor module instruction execution states.

A typical PM machine cycle (see Figure 3) involves five sequential time states, TS1, TS2, TS3, TS4, and TS5. During time states TS1 and TS2, system memory is addressed by a lower and an upper address byte respectively to form a 14-bit address. During TS1, the program counter is incremented. In time state TS3, the instruction addressed during TS1, and TS2 is fetched, and during TS4 or TS4 and TS5 the fetched instruction is executed. The flow chart of time-state transitions shown in Figure 4 simplifies the progression through time states during a machine cycle.



S2	S1	S0	STATE
1	1	0	TTL (INTERRUPT)
0	1	0	T1
1	0	0	T2
0	0	0	WAIT
0	0	1	T3
0	1	1	STOPPED
1	1	1	T4
1	0	1	T5

FIGURE 3. INSTRUCTION EXECUTION STATES

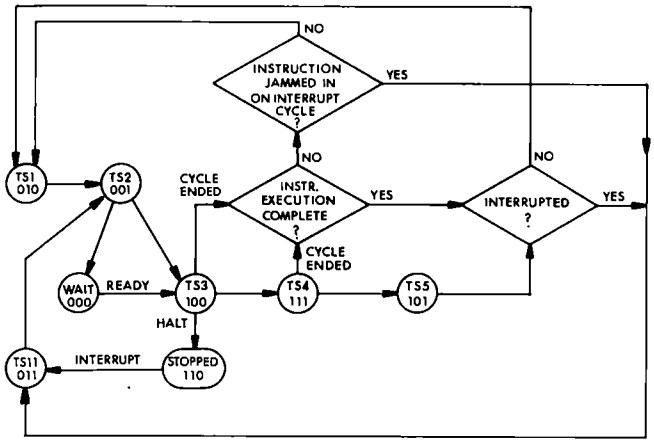


FIGURE 4. TIME-STATE TRANSITION FLOW DIAGRAM

If an interrupt is initiated by an external event, control does not return to TS1 upon completion of instruction execution but instead reverts to time state TS1I which replaces TS1. During TS1I, the external event is recognized, an interrupt is generated, and incrementing of the program counter (PC) is suppressed to permit execution of a one-byte instruction generated by the external event.

At the completion of time state TS2, the processor checks the state of the READY line. If this line is true (High), time state TS3 is entered; if not true (Low), the Wait state is entered. TS3 is entered from the Wait state when the READY line is next asserted. The state of the READY line is available for external use through the signal RDY H.

If the instruction fetched during time state TS3 is a Halt, the processor stops operation at the end of that time state. The processor remains halted until the START line is asserted, forcing entry into time state TS1I and execution of a jammed one-byte instruction. When the PM is operating (STOP H asserted), the RUN indicator is lit. When the Stop state is entered (STOP H not asserted), this indicator will be extinguished.

Processor Module Instruction Cycle

Figure 4 shows that a machine cycle can be completed at the end of time states TS3, TS4, or TS5. The instruction cycle for Processor Module instructions is variable depending on the class and function of the specific instruction executed and can consist of one, two, or three machine cycles.

The completion point within a machine cycle is also instruction-dependent so that the number of state times encompassed by these instructions can range from a minimum of three to a maximum of eleven.

The processor module executes four types of machine cycles which are listed and defined below:

1. PCI Cycle—This cycle is always the first cycle of every PM instruction and initiates an instruction fetch. The two bytes which address memory during this cycle are always taken from the PC.
2. PCR Cycle—This cycle inititates the addressing of memory by the incremented PC to retrieve a subsequent byte of a two- or three-byte instruction, or to retrieve a data byte addressed by the contents of registers H and L.
3. PCC Cycle—This cycle initiates the set-up and execution of I/O instructions by placing the address of the peripheral device to be accessed and the content of the accumulator onto the memory address bus and retrieving and/or storing the data at the pertinent peripheral.
4. PCW Cycle—This cycle initiates the addressing of memory by the content of the H and L registers and implementing the writing of data into that location.

At time state TS2, the specific signal corresponding to the machine cycle being executed is asserted and latched for external use. These signals are derived from the states of the two high-order bits of the high address byte and are decoded and gated out for external use during time state TS3.

Input Data Paths

The processor chip is equipped with a single time-shared 8-bit bidirectional data port to permit memory addressing, instruction fetching, and data input and output. This port connects to a bidirectional data bus on the Processor Module, and input data is gated onto this bidirectional bus from a unidirectional input data bus. Data is multiplexed and gated onto the input data bus from four input ports which are selected as a function of the machine cycle currently being executed.

When a PCI or PCR machine cycle is in process, time state TS3 selects the bidirectional memory port DM0 L to DM7 L to fetch from memory the instruction or data word addressed during TS1 and TS2 of that cycle. During a PCC cycle, the signal I/O IN asserted by the I/O control logic selects the peripheral data-in port DI0 L to DI7 L to retrieve data from the addressed peripheral device as specified by the I/O read instruction being executed.

At start-up or restart time, or in response to an external event, one of the signals START L or IOEE L is asserted to select the I/O start port DIOST0 L to DIOST7 L for external instruction input. Figure 3 shows that when an interrupt occurs in response to an external event, time state TS1I is entered so that normal incrementing of the program counter is inhibited.

As a consequence of selecting the I/O start port, an externally supplied one-byte instruction is automatically fetched. This instruction is executed during time states TS4 and TS5.

Note that when IOEE L is asserted, the external event recognition logic must be enabled under program control in order for the I/O start port to be selected as a response to an external event. (See PROGRAMMING.) The signal START L, when asserted, bypasses the event recognition to select the I/O start port irrespective of program-enabling action.

The signal PFSEE L can be asserted by a power-fail sensing circuit, by a system start-stop switch, or by some other external logic. A power-fail circuit or external logic connected to the power-fail/stop port DPFS0 L to DPFS7 L can jam a one-byte Restart instruction into this port upon detection of a power failure or in reaction to some external event. This instruction would then be executed to initiate a service routine or sequence.

Activating an external system stop switch would also select the power-fail/stop port. However, in this case, all the data lines into this port would be normally High which is equivalent to a Halt instruction. When the signal PFSEE L is asserted, the power-fail/stop port is selected at time state TS3 following entry into time state TS1I which, as shown in Figure 4, occurs in the same manner as with a normal interrupt. During time state TS3, the instruction at this port is fetched by the processor for execution.

Output Data Paths

Processor Module output can be in the form of memory addresses, memory and I/O control information, I/O device addresses, data output to memory, and data output to peripherals. With the exception of data output to memory, all of these addresses and data are stored for output in the 16-bit multipurpose output register. Data words are latched into the register by the data selection logic during time states TS1 and TS2 (see Figure 1).

Memory addresses are issued as two separate words to form a 14-bit address word during machine cycles PCI, PCR, and PCW, the lower word at time state TS1 and the upper word at TS2 of these cycles.

During each one of the time states, the corresponding memory address word is loaded into the 16-bit output register by the address selection logic. At the end of time state TS2, the current memory address is present as output from the output register. These output lines are buffered to drive the address/data lines ADRD00 L to ADRD13 L which can be bused out to MPS ROMs, RAMs, and I/O devices.

Data to be written in read/write memory is gated onto the bidirectional memory bus DM0 L to DM7 L during time state TS1 and TS2 of a PCW cycle. Data must be accepted by the memory during time state TS3 of that cycle.

During a PCC cycle, the content of the accumulator and the instruction register are stored in output register bits 0 through 13 for use as peripheral device output data and address. At time state TS1 of a PCC cycle, the content of the accumulator, which is the data to be output to the addressed peripheral device, is placed in the output register bit positions 0 to 7.

At time state TS2, the content of the instruction register is transferred to output register bit positions 8 through 15, with bits 9 through 13 containing the address of the peripheral device being accessed during the PCC cycle. Once stored in the output register, these address and data fields are available to external peripheral devices over the lines ADRD00 L to ADRD13 L. This device address field permits the addressing of eight input devices and 22 output devices.

Control Logic

The control logic (see Figure 1) provides the various control signals necessary to memory accessing and to the performance of input/output operations with associated peripherals. Input to the control logic is the state of bits 14 and 15 of the output register.

During time state TS3 of every machine cycle, the I/O control signals pertinent to the machine cycle currently being executed are asserted (see Figure 5). For example, during PCI and PCR cycles where memory is addressed to fetch an instruction or data, the signal MEM RD H is asserted at the end of time state TS2 and throughout TS3. This signal controls the reading of data from that memory location addressed during the pertinent machine cycle.

During a PCC cycle, the signals I/O IN H or I/O OUT H are asserted during time state TS3 to control the storage and retrieval of data at external peripheral devices. When data is to be written into an addressed memory location during a PCW cycle, the signal MEM WR H is asserted by the control during time state TS3 of that machine cycle.

In addition to these signals, the control logic also asserts one of the signals PCI L, PCC L, PCR L, or PCW L during time state TS3 of the corresponding machine cycle.

Asynchronous Communications Receiver/Transmitter Logic

The PM is equipped with a full duplex communication receiver/transmitter implemented by a Universal Asynchronous Receiver/Transmitter (UAR/T). A 1.76-kHz clock driven by an 844.8-kHz crystal-controlled clock integral to the PM is input to this device to produce a 110-baud data transfer rate. The receiver/transmitter is addressed for transmission or receipt of data or for status information during time state TS3 by bits 9 to 13 of the output register. Data is transmitted and received at the module over 20-mA current loop or TTL-compatible lines (USI H and USO H) to interface with Teletype-like lines or to telephone lines through a modem.

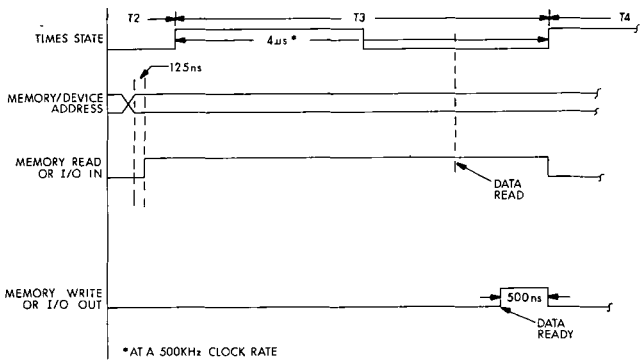


FIGURE 5. INPUT-OUTPUT TIMING

Higher baud rates can be obtained by using an external clock input (URCLK H or UTCLK H) which can be derived from external logic which divides the basic crystal clock frequency (UCLK H). The maximum data transfer rate is 9600 baud for TTL lines and 4800 baud for current loop lines (at limited lengths).

Internal switches on the PM permit selection of operation under external clock control; also, the number of stop bits used (one or two) can be selected through switch action. Active or passive operation of current loop lines is jumper-selectable.

Odd parity, even parity, or no parity is selectable by asserting the corresponding input signals. Also, the number of data bits in a word can be selected to vary from five to eight bits by encoding the input signals UNB1, UNB2 as shown below.

Word Length	UNB2	UNB1
5 bits	LOW	LOW
6 bits	LOW	HIGH
7 bits	HIGH	LOW
8 bits	HIGH	HIGH

Both TBMT H (Transmitter Buffer Empty) and DA H (Data Available) are available for external interrupt drive capability.

When an input instruction is being executed, the signal I/O IN H is asserted by the control logic. This signal and the receiver/transmitter device-receive address are decoded to gate data from the device onto peripheral data-in port DI0 L to DI7 L.

Data for transmission is written from the output register into the receiver/transmitter transmission buffer during time state TS3 of a PCC cycle when the signal I/O OUT H is asserted. This signal and the transmit data address are decoded to strobe data from the PM output register into the transmission buffer.

Status information, which includes receiver/transmitter error conditions and transmit and receive buffer status, is retrieved through execution of an input instruction with the assigned status device address. As with reading data from the receiver/transmitter, the signal I/O IN H is asserted as a result of a PCC cycle execution. This signal and the device address are decoded to gate device status into the peripheral data-in port DI0 L through DI7 L.

Switch DS6 (see Figure 1) can be used to disable the PM receiver/transmitter to gain the use of two additional input and one additional output device addresses.

Interrupt Control

The interrupt control logic drives the input data multiplexer to select one of two input ports (see Figure 1). If the interrupt results from a power failure or a stop command, the power-fail/stop port DPFS0 L to DPFS7 L is selected.

Similarly, if the PFSEE L line or the system START L line is asserted, the I/O start port DIOST0 L to DIOST7 L is selected. The instruction jammed into these ports as a consequence of a power failure or an I/O or restart interrupt, is furnished by external logic as determined by the specific application.

I/O interrupts can be enabled or disabled under program control by the PM external event enable/disable logic. Interrupts are disabled by executing the instruction IOF and enabled by executing ION. (See PROGRAMMING.) Since interrupts will be enabled or disabled one instruction time after execution of an ION or IOF, one instruction can be executed after ION or IOF before interrupts are actually enabled or disabled. The external event recognition logic is automatically disabled after every interrupt.

SIGNAL SPECIFICATIONS

Input Signals

Symbol	*Description*	*Pin*	*Function*
DIOST0 L	I/O Start Input 0	AF2	These eight parallel input lines comprise the external event port to the input multiplexer. These lines are multiplexed in response to assertion of the signal START L or IOEE L.
DIOST1 L	I/O Start Input 1	AE1	
DIOST2 L	I/O Start Input 2	AP2	
DIOST3 L	I/O Start Input 3	AR1	
DIOST4 L	I/O Start Input 4	BF2	
DIOST5 L	I/O Start Input 5	BE1	
DIOST6 L	I/O Start Input 6	BP1	
DIOST7 L	I/O Start Input 7	BN1	

Input Signals

Symbol	*Description*	*Pin*	*Function*
START L	Start Processor Module	DR1	This pulse, having a period >50 ns, is asserted to place the Processor Module in the run state by multiplexing an appropriate instruction from the DIOST lines into the module for fetching.
IOEE L	External Event Line	DU1	This line is asserted by external event detection logic. When asserted, this signal multiplexes the instruction on the DIOST lines into the input multiplexer for fetching. This line can be enabled or disabled under program control.
DI0 L	I/O Device Input 0	AE2	These eight prallel input lines comprise the peripheral device data input port to the input multiplexer. Data must appear on these lines as a consequence of I/O device addressing resulting from execution of an INP instruction (see PROGRAMMING).
DI1 L	I/O Device Input 1	AD1	
DI2L	I/O Device Input 2	AN2	
DI3 L	I/O Device Input 3	AP1	
DI4 L	I/O Device Input 4	AV2	
DI5 L	I/O Device Input 5	AV1	
DI6 L	I/O Device Input 6	AU2	
DI7 L	I/O Device Input 7	AT2	
DPFS0 L	Power/Fail Stop Input 0	AH2	These eight parallel input lines comprise the power/fail stop port to the input multiplexer. An appropriate instruction must be multiplexed into the Processor Module from this port for fetching by the processor module by the assertion of the signal DFSEE L.
DPFS1 L	Power/Fail Stop Input 1	AF1	
DPFS2 L	Power/Fail Stop Input 2	AR2	
DPFS3 L	Power/Fail Stop Input 3	AS1	
DPFS4 L	Power/Fail Stop Input 4	BH2	
DPFS5 L	Power/Fail Stop Input 5	BF1	
DPFS6 L	Power/Fail Stop Input 6	BS1	
DPFS7 L	Power/Fail Stop Input 7	BR1	
PFSEE L	Power/Fail Stop External Event Line	AM1	This pulse, having a period >50 ns, is asserted in response to a power failure or some external stop switch. When asserted, this signal causes the appropriate instruction on the DPSF lines to be multiplexed into the processor module for fetching.
RDY H	READY	AS2	When true (High), this signal causes the Processor Module to be ready for operation which is the normal state of this signal. When false (Low), the Processor Module enters the Wait state and remains in this state until Ready is again asserted.
UPOE	Even-Odd Parity Select	DA1	When this input line to the Processor Module receiver/transmitter is asserted High, even parity is selected, and when asserted Low, odd parity is selected.

Symbol	Description	Pin	Function
UNP	Inhibit Parity Checking	CJ2	When this input line to the Processor Module receiver/transmitter is true (High), the appending and checking of a parity bit is inhibited. When false (Low), parity is enabled.
UNB1	Data Word-Size	DC1	These signals are used to select the number of bits comprising the data word handled by the receiver/transmitter accordingly:
UNB2	Selection Bits	DB1	
USI H	TTL Serial Input	DH1	This input signal line provides for the transfer of serial TTL data into the receiver/transmitter. In order to use this line, the 20-mA current loop must be disconnected. (See Special Wiring/Adjustments.)
	Test Initialize	DL1	Test point; can be used to simulate the power-up initialize function.

UNB2	UNB1	Bits/Character
LOW	LOW	5
LOW	HIGH	6
HIGH	LOW	7
HIGH	HIGH	8

Output Signals

Symbol	Description	Pin	Function
ADRD00 L	Address Line 0	CF2	These 14 parallel output address lines are multifunction. These lines carry the 14-bit memory address for accessing instructions and data; address lines 0 to 7 convey data to an addressed peripheral device; address lines 9 through 13 serve to address a peripheral device for writing or reading.
ADRD01 L	Address Line 1	CE2	
ADRD02 L	Address Line 2	CD1	
ADRD03 L	Address Line 3	CE1	
ADRD04 L	Address Line 4	CB1	
ADRD05 L	Address Line 5	CC1	
ADRD06 L	Address Line 6	CF1	
ADRD07 L	Address Line 7	CH1	
ADRD08 L	Address Line 8	CH2	
ADRD09 L	Address Line 9	CS1	
ADRD10 L	Address Line 10	CM1	
ADRD11 L	Address Line 11	CR1	
ADRD12 L	Address Line 12	CJ1	
ADRD13 L	Address Line 13	CK1	

Output Signals

Symbol	*Description*	*Pin*	*Function*
MEM RD H	Memory Read Control Signal	DD2	When asserted, this signal designates that contents of the memory location currently addressed by ADRD00 L to ADRD13 L are at the bidirectional memory bus DM0 L to DM7 L are stable and ready for input multiplexing.
MEM WR H	Memory Write Control Signal	DH2	When asserted, this signal designates that the word to be written into the memory location currently addressed by ADRD00 L to ADRD13 L is on the bidirectional memory bus DM0 L to DM7 L and stable.
IO IN H	Peripheral Device Read Control	DJ1	When asserted, this signal strobes the data word from the peripheral device addressed by ADRD09 to ADRD13 onto lines DI0 to DI7 for input multiplexing.
IO OUT H	Peripheral Device Write Control	DK1	When asserted, this signal designates that the data on output lines ADRD00 to ADRD07 to be written to the peripheral device addressed by output lines ADRD09 to ADRD13 is stable and ready to be written into the device.
SCLK H	System Clock	DS1	This signal is the basic system clock frequency. The Processor Module is normally shipped with this clock frequency set to 1 MHz.
UCLK H	Crystal Controlled Clock	AB1	This output signal is a crystal-controlled 844.8-kHz clock frequency which can be the base frequency for deriving various operating frequencies up to 9600 baud.
Φ1CLK H Φ2CLK H	LSI Processor Clock	DM1 DN1	These output signals comprise the two-phase nonoverlapping clock frequency derived from SCLKH and serve to time the processor chip.
TS1 H TS1 IH TS2 H TS3 H TS4 H TS5 H STOP H WAIT H	Time States	CK2 CA1 CL1 CL2 CU2 CM2 CD2 CT2	These eight output signals provide an external reference to Processor Module time states. (See Processor Module Timing under FUNCTIONS.)
PCI L PCR L PCW L PCC L	Machine Cycles	CN2 CV2 CP2 CR2	These four output signals provide an external reference to Processor Module machine cycles. (See Processor Module Instruction Cycles under FUNCTIONS.)
SYNC H	Basic Timing Signal	CN1	This output signal is Φ2CLK H÷2 and provides the basic timing relationship within a given time state.
PINIT L	System Initialize Signal	AA1	This output signal supplies a power-up initialize pulse of approximately 10 ms in duration.
USO H	TTL Serial Output	DE1	This output signal line provides for the transfer of serial TTL data from the asynchronous communications controller.
TBMT H	Transmitter Buffer Empty	BK1	When asserted, this signal designates that the receiver/transmitter transmitter buffer is empty and therefore is ready to accept a word for transmission.
DA H	Data Available	BM2	When asserted, this signal designates that a word is ready for reading from the receiver/transmitter receive buffer.

Input/Output Signals

Symbol	*Description*	*Pin*	*Function*
DM0 L	Bidirectional Memory Data Line 0	AD2	
DM1 L	Bidirectional Memory Data Line 1	AC1	
DM2 L	Bidirectional Memory Data Line 2	AM2	
DM3 L	Bidirectional Memory Data Line 3	AN1	These eight parallel bidirectional lines comprise the input/output memory data bus for the Processor Module. Incoming memory data is received by the input multiplexer and outgoing memory data is gated onto these lines by the output data gates. Each line comprising this parallel bus will drive ten unit loads, and sink four unit loads.
DM4 L	Bidirectional Memory Data Line 4	BD1	
DM5 L	Bidirectional Memory Data Line 5	BC1	
DM6 L	Bidirectional Memory Data Line 6	BM1	
DM7 L	Bidirectional Memory Data Line 7	BL1	
UTCLK H	Internal/External Clock Signal for Data Transmission	CP1	As an input, this clock signal permits transmission of data from the receiver/transmitter at baud rates other than the standard 110-baud rate. When used in this manner, switch DS1 must be open. When switch DS1 is closed, this signal is the output of the 110-baud clock (1.76 kHz).
URCLK H	Internal/External Clock Signal for Data Receipt	BJ1	As an input, this clock signal permits the receipt of data by the receiver/transmitter at baud rates other than the standard 110-baud rate. When used in this manner, switch DS2 must be open. When switch DS2 is closed, this signal is the output of the 110-baud clock.

PROGRAMMING

Processor Module Instruction Set

The processor module instruction set is presented as five instruction categories with the criteria for such categorizing being the nature of the function performed by instruction execution. The functions performed by these categories are:

1. Register operations
2. Accumulator operations
3. Program counter and stack control operations
4. Input/output operations
5. Machine operations

The notational standards used in this presentation are listed in Table 1.

Register Operation Instructions

Register operation instructions implement four modes of register and memory data manipulation. These modes are:

1. Load data into registers and memory
2. Load constants into registers or memory
3. Increment a register
4. Decrement a register

The registers manipulated by these instructions include the accumulator (A register), the six scratch pad registers, (B, C, D, E, H, and L), and any addressable read-write and/or read-only memory location.

TABLE 1
Processor Module Instruction Set Notational Standards

Symbol	*Definition*
<B2>	Second byte of an instruction
<B3>	Third byte of an instruction
r, r1, r2	Any of the 8-bit general-purpose registers A, B, C, D, E, H, L
A	A general-purpose register used as the accumulator
B, C, D, E	General-purpose registers
H, L	Two general-purpose registers used as memory address registers
C_3, C_4	Condition flip-flops: Designation — Code (C_4 C_3) — Truth Condition Carry (C) — 0 0 — Overflow, underflow Zero (Z) — 0 1 — Result is ZERO Sign (S) — 1 0 — MSB of result is ZERO Parity (P) — 1 1 — Parity of result is even
C	One of the condition flip-flops—C, Z, S, or P
M	The memory location addressed by the content of registers H and L. The source/destination code for memory is 111.
( )	Content of a memory location, a register, or a condition flip-flop
∧	Logical AND
∨	Inclusive OR
⊻	Exclusive OR
Am	Bit m of the Accumulator register
STACK	A group of eight 14-bit registers with seven organized as a LIFO pushdown stack to store nested subroutine addresses with the eighth storing the current program counter (P)
P	A 14-bit register which contains the address of the next instruction to be fetched
←	Becomes
XXX	Don't care about the content of this field
SSS	Source register r code field
DDD	Destination register r code field: Source/Destination register codes: Register — Code A — 000 B — 001 C — 010 D — 011 E — 100 H — 101 L — 110

Load Data into Registers and Memory

Instructions implementing this mode of data manipulation serve to transfer data between any of the registers r (A, B, C, D, E, H, L) into any of these registers from any memory location addressed by H, L; and into any memory location addressed by H, L from any of these registers. These instructions have a one-byte format.

Load r1 with Content of r2

MNEMONIC Lr1r2

7	6	5			3	2		0
1	1	D	D	D		S	S	S

EXECUTION TIME/ MACHINE CYCLES EXECUTED
20 μs, PCI

FUNCTION
(r1)←(r2)

Load r with Content of Memory Location H, L

MNEMONIC LrM

7	6	5			3	2		0
1	1	D	D	D		1	1	1

EXECUTION TIME/ MACHINE CYCLES EXECUTED
32 μs, PCI PCR

FUNCTION
(r)←(M)

MNEMONIC LMr

7					3	2		0
1	1	1	1	1		S	S	S

EXECUTION TIME/ MACHINE CYCLES EXECUTED
28 μs, PCI PCW

FUNCTION
(M)←(r)

Load Constants into a Register or Memory

In this mode of data manipulation, a data byte can be immediately loaded into any of the registers A, B, C, D, E, H, L and into any memory location addressed by registers H, L. These instructions have a two-byte format.

Load a Data Byte into Register r

MNEMONIC LrI

7	6	5			3	2		0
0	0	D	D	D		1	1	0

7								0
<B2>								

EXECUTION TIME/ MACHINE CYCLES EXECUTED
32 μs, PCI PCR

FUNCTION
(r)←<B2>

Load a Data Byte into Memory Location M

MNEMONIC LMI

7	6	5			3	2		0
0	0	1	1	1		1	1	0

7								0
<B2>								

EXECUTION TIME/ MACHINE CYCLES EXECUTED
36 μs, PCI PCR PCW

FUNCTION
(M)←<B2>

Increment Register r

MNEMONIC INr

7	6	5			3	2		0
0	0	D	D	D		0	0	0

EXECUTION TIME/ MACHINE CYCLES EXECUTED
20 μs, PCI

FUNCTION
(r)←(r)+1 WHERE r≠A or M

Decrement Register r

MNEMONIC DCr

7	6	5			3	2		0
0	0	D	D	D		0	0	1

EXECUTION TIME/ MACHINE CYCLES EXECUTED
20 μs, PCI

FUNCTION
(r)←(r)−1 WHERE r≠A or M

Accumulator Operation Instructions

The instructions comprising this class are grouped for discussion into four categories:

1. Arithmetic/logical operations with registers r
2. Arithmetic/logical operations with memory
3. Immediate arithmetic/logical operations
4. Rotate the accumulator operations

When executed, instructions in this class (unless otherwise specified) have the following general effects on the four condition flip-flops. If a carry or borrow is generated, the carry flip-flop is set to ONE. If the result of a comparison with the accumulator is zero, the zero flip-flop is set to ONE. When bit 7 of a result is one, the sign flip-flop is set to ONE. When a result contains an even number of ones, the parity flip-flop is set to ONE. In each of these cases, the opposite circumstance results in the pertinent condition flip-flop being reset to ZERO. Depending on the specific instruction in this class being executed; one or more of the condition flip-flops can be set as a consequence of execution.

Arithmetic/Logical Operations with Registers r

The eight instructions implementing arithmetic/logical operations are performed between the accumulator and one of the general-purpose registers r. The content of a register r is unchanged when operated on by one of these instructions except when r is the accumulator. In this case, the prior content of the accumulator is replaced by the result. All instructions in this category have a one-byte format.

The result of an arithmetic or logical operation can affect the four condition flip-flops. Multiple precision binary arithmetic is performed through use of the carry flip-flop, and logical operations always reset the carry flip-flop to ZERO. Rotate instructions can only affect the carry flip-flop, with all other condition flip-flops remaining unchanged by these operations.

Add r to Accumulator
MNEMONIC ADr

7	6	5	4	3	2	1	0
1	0	0	0	0	S	S	S

EXECUTION TIMES/ MACHINE CYCLES EXECUTED
20 μs, PCI

FUNCTION
(A)←(A)+(r)

Add r to Accumulator and Carry Flip-Flop
MNEMONIC ACr

7	6	5	4	3	2	1	0
1	0	0	0	1	S	S	S

EXECUTION TIME/ MACHINE CYCLES EXECUTED
20 μs, PCI

FUNCTION
(A)←(A)+(CARRY)+(r)

Subtract r from Accumulator
MNEMONIC SUr

7	6	5	4	3	2	1	0
1	0	0	1	0	S	S	S

EXECUTION TIMES/ MACHINE CYCLES EXECUTED
20 μs, PCI

FUNCTION
(A)←(A)−(r)

Subtract r and Carry Flip-Flop from Accumulator
MNEMONIC SBr

7	6	5	4	3	2	1	0
1	0	0	1	1	S	S	S

EXECUTION TIME/ MACHINE CYCLES EXECUTED
20μs, PCI

FUNCTION
(A)←(A)−(r)−(BORROW)

Logically AND r and Accumulator
MNEMONIC NDr

7	6	5	4	3	2	1	0
1	0	1	0	0	S	S	S

EXECUTION TIME/ MACHINE CYCLES EXECUTED
20 μs, PCI

FUNCTION
(A)←(A) ∧ (r)

Logically OR r and Accumulator
MNEMONIC ORr

7	6	5	4	3	2	1	0
1	0	1	1	0	S	S	S

EXECUTION TIME/ MACHINE CYCLES EXECUTED
20 μs, PCI

FUNCTION
(A)←(A) V (r)

Exclusive OR r and Accumulator
MNEMONIC XRr

7	6	5	4	3	2	1	0
1	0	1	0	1	S	S	S

EXECUTION TIME/ MACHINE CYCLES EXECUTED
20 μs, PCI

FUNCTION
(A)←(A)⊻(r)

Compare r with Accumulator
MNEMONIC CPr

7	6	5	4	3	2	1	0
1	0	1	1	1	S	S	S

EXECUTION TIME/ MACHINE CYCLES EXECUTED
20μs, PCI

FUNCTION
(A)−(r)

Arithmetic/Logical Operations with Memory
Arithmetic/logical operations involving the accumulator and a memory location addressed by registers H, L are implemented by the eight instructions described below. Content of the memory location addressed by registers H, L is unchanged by instruction execution. All instructions in this category have a one-byte format.

Add M to Accumulator
MNEMONIC ADM

7	6	5	4	3	2	1	0
1	0	0	0	0	1	1	1

EXECUTION TIME/ MACHINE CYCLES EXECUTED
32 μs, PCI PCR

FUNCTION
(A)←(A)+(M)

Add M to Carry Flip-Flop and Accumulator
MNEMONIC ACM

7	6	5	4	3	2	1	0
1	0	0	0	1	1	1	1

EXECUTION TIME/ MACHINE CYCLES EXECUTED
32 μs, PCI PCR

FUNCTION
(A)←(A)+(CARRY)+(M)

Subtract M from Accumulator
MNEMONIC SUM

7	6	5	4	3	2	1	0
1	0	0	1	0	1	1	1

EXECUTION TIME/ MACHINE CYCLES EXECUTED
32 μs, PCI PCR

FUNCTION
(A)←(A)−(M)

Subtract M and Carry Flip-Flop from Accumulator
MNEMONIC SBM

7	6	5	4	3	2	1	0
1	0	0	1	1	1	1	1

EXECUTION TIME/ MACHINE CYCLES EXECUTED
32 μs, PCI PCR

FUNCTION
(A)←(A)−(M)−(BORROW)

Logically AND M and Accumulator
MNEMONIC NDM

7	6	5	4	3	2	1	0
1	0	1	0	0	1	1	1

EXECUTION TIME/ MACHINE CYCLES EXECUTED
32 μs, PCI PCR

FUNCTION
(A)←(A) ∧ (M)

Logically OR M and Accumulator
MNEMONIC ORM

7	6	5	4	3	2	1	0
1	0	1	1	0	1	1	1

EXECUTION TIME/ MACHINE CYCLES EXECUTED
32 μs, PCI PCR

FUNCTION
(A)←(A) V (M)

Exclusive OR M and Accumulator
MNEMONIC XRM

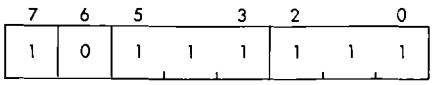
7	6	5	4	3	2	1	0
1	0	1	0	1	1	1	1

EXECUTION TIME/ MACHINE CYCLES EXECUTED
32 μs, PCI PCR

FUNCTION
(A)←(A)⊻(M)

Compare M with Accumulator

MNEMONIC CPM



EXECUTION TIME/ MACHINE CYCLES EXECUTED
32 μs, PCI PCR

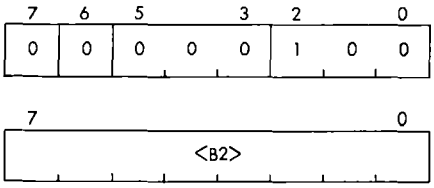
FUNCTION
(A)−(M)

Immediate Arithmetic/Logical Operations

The eight instructions performing immediate arithmetic/ logical operations are accomplished in an immediate mode using the accumulator and a data byte accompanying each instruction as arguments. All instructions in this category have a two-byte format.

Add <B2> to Accumulator

MNEMONIC ADI

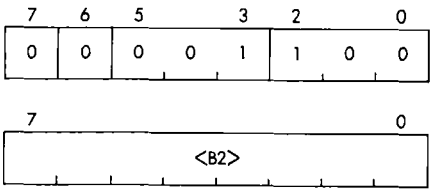


EXECUTION TIME/ MACHINE CYCLES EXECUTED
32 μs, PCI PCR

FUNCTION
(A)←(A)+<B2>

Add <B2> to Accumulator and Carry Flip-Flop

MNEMONIC ACI

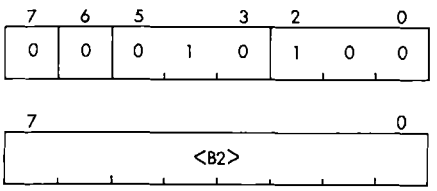


EXECUTION TIME/ MACHINE CYCLES EXECUTED
32 μs, PCI PCR

FUNCTION
(A)←(A)+(CARRY)+<B2>

Subtract <B2> from Accumulator

MNEMONIC SUI



EXECUTION TIME/ MACHINE CYCLES EXECUTED
32 μs, PCI PCR

FUNCTION
(A)←(A)−<B2>

Subtract <B2> and Carry Flip-Flop From Accumulator

MNEMONIC SBI

EXECUTION TIME/ MACHINE CYCLES EXECUTED
32 μs, PCI PCR

FUNCTION
(A)←(A)−<B2>−(BORROW)

Logically AND <B2> and Accumulator

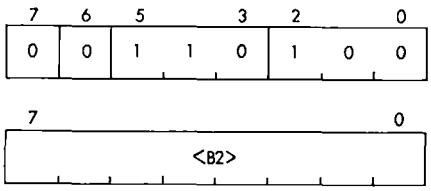
MNEMONIC NDI

EXECUTION TIME/ MACHINE CYCLES EXECUTED
32 μs, PCI PCR

FUNCTION
(A)←(A) ∧ <B2>

Logically OR <B2> and Accumulator

MNEMONIC ORI



EXECUTION TIME/ MACHINE CYCLES EXECUTED
32 μs, PCI PCR

FUNCTION
(A)←(A) ∨ <B2>

Exclusive OR <B2> and Accumulator

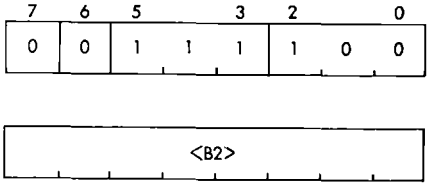
MNEMONIC XRI

EXECUTION TIME/ MACHINE CYCLES EXECUTED
32 μs, PCI PCR

FUNCTION
(A)←(A) ∀ <B2>

Compare <B2> with Accumulator

MNEMONIC CPI



EXECUTION TIME/ MACHINE CYCLES EXECUTED
32 μs, PCI PCR

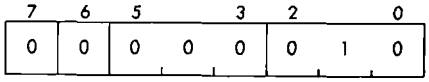
FUNCTION
(A)−<B2>

Rotate the Accumulator Operations

The four instructions in this category permit the content of the accumulator to be rotated, one bit per instruction execution, right or left, and into, or through the carry flip-flop. Only the carry flip-flop is affected by execution of these instructions; all other condition flip-flops remain unaffected. All instructions in this category have a one-byte format.

Rotate Accumulator Left and Into Carry Flip-Flop

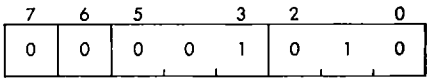
MNEMONIC RLC



EXECUTION TIME/ MACHINE CYCLES EXECUTED
20 μs, PCI

FUNCTION
Am+1←Am, A0 ←A7, (CARRY)←A7

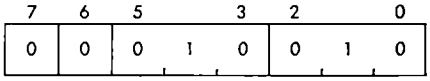
Rotate Accumulator Right and Into Carry Flip-Flop
MNEMONIC RRC



EXECUTION TIME/
MACHINE CYCLES
EXECUTED
20 μs, PCI

FUNCTION
Am←Am+1, A7←A0, (CARRY)←A0

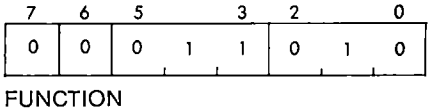
Rotate Accumulator Left Through Carry Flip-Flop
MNEMONIC RAL



EXECUTION TIME/
MACHINE CYCLES
EXECUTED
20 μs, PCI

FUNCTION
Am+1←Am, A0←(CARRY), (CARRY)←A7

Rotate Accumulator Right Through Carry Flip-Flop
MNEMONIC RAR



EXECUTION TIME/
MACHINE CYCLES
EXECUTED
20 μs, PCI

FUNCTION
Am←Am+1, A7←(CARRY), CARRY)←A0

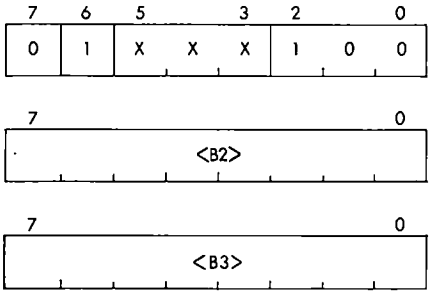
Program Counter and Stack Control Operations
Instructions in this class perform three basic programming functions, namely: jump to nonsequential memory locations, subroutine entry, and return from subroutine. The commonality which exists between the instructions performing these three functions is found in the use of the program counter.

The four condition flip-flops (carry, zero, sign, and parity) can be used by conditional jump, subroutine call, and subroutine return instructions.

All jump and subroutine call instructions are accompanied by the two 8-bit bytes <B2> and <B3> which are used during instruction execution to modify the program counter. Byte <B2> always contains the eight low-order bits of the memory address, and <B3> contains the six high-order bits. Since all memory addresses are 14 bits in length, bits 6 and 7 of <B3> are not used.

Jump Instructions
Jump instructions implement the transfer of program control to some nonsequential memory address either conditionally or unconditionally based on the state of a condition flip-flop as determined by prior processing. All jump instructions have a three-byte format.

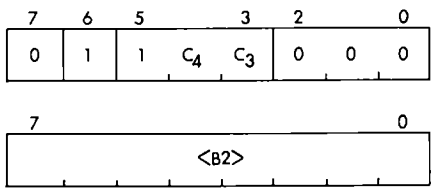
Jump Unconditionally
MNEMONIC JMP



EXECUTION TIME/
MACHINE CYCLES
EXECUTED
44 μs, PCI PCR PCR

FUNCTION
(P)←<B3><B2>

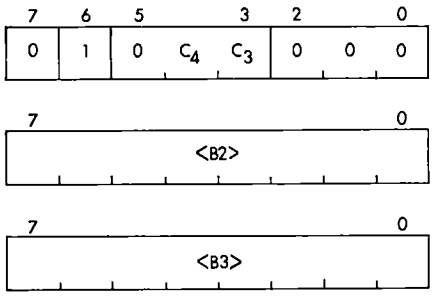
Jump if Condition True
MNEMONIC JTc



EXECUTION TIME/
MACHINE CYCLES
EXECUTED
44 μs, PCI PCR PCR
36 μs, PCI PCR PCR

FUNCTION
(P)←<B3><B2>IF C=1
(P)←(P)+3 IF C=0

Jump if Condition False
MNEMONIC JFc

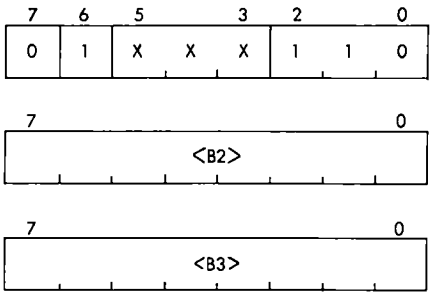


EXECUTION TIME/
MACHINE CYCLES
EXECUTED
44 μs, PCI PCR PCR
36 μs, PCI PCR PCR

FUNCTION
(P)←<B3><B2>IF C=0
(P)←(P)+3 IF C=1

Subroutine Call Instructions
Call instructions initiate entry into a subroutine either unconditionally or conditionally based on the state of a designated condition flip-flop as determined by prior processing. When a call instruction is executed, and an attending condition is satisfied, the content of the program counter is automatically stored in the STACK to implement return to the program line when subroutine function is complete. The prior content of the program counter is then replaced by <B3> <B2> which is the starting address of the subroutine being called. All subroutine call instructions have a three-byte format. The RST instruction, which is a special one-byte call instruction used to perform calls to restart routines, is for purposes of functional presentation described under "Machine Instructions."

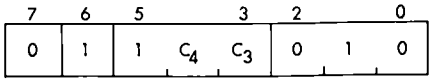
Call Subroutine Unconditionally
MNEMONIC CAL

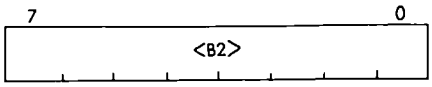


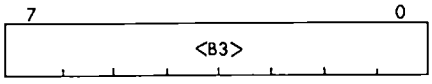
EXECUTION TIME/
MACHINE CYCLES
EXECUTED
44 μs, PCI PCR PCR

FUNCTION
(STACK)←(P), (P)←<B3><B2>

Call Subroutine if Condition True
MNEMONIC CTc



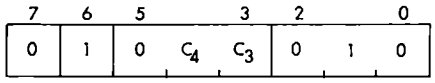


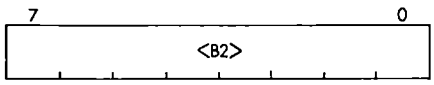


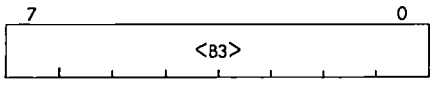
EXECUTION TIME/
MACHINE CYCLES
EXECUTED

FUNCTION
(STACK)←(P), (P)←<B3> <B2> IF C=1 44 μs, PCI PCR PCR
(P)←(P)+3 IF C=0 36 μs, PCI PCR PCR

Call Subroutine if Condition False
MNEMONIC CFc







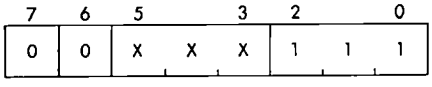
EXECUTION TIME/
MACHINE CYCLES
EXECUTED

FUNCTION
(STACK)←(P), (P)←<B3> <B2> IF C=0 44 μs, PCI PCR PCR
(P)←(P)+3 IF C=1 36 μs, PCI PCR PCR

Subroutine Return Instructions

Return instructions provide the mechanism for returning to the next instruction in the program line following the call instruction, after completion of the subroutine function. With these instructions, a return can be unconditional or conditional based on the state of a designated condition flip-flop as determined by prior processing. When executed, and any conditions are satisfied, these instructions automatically pop the STACK to retrieve the last state of the program counter prior to subroutine entry. This 14-bit address is then jammed into the program counter to point to the memory address from which the next instruction in the program line is to be fetched. All subroutine return instructions have a one-byte format.

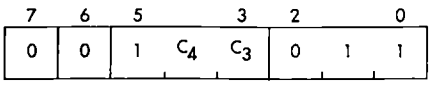
Return Unconditionally
MNEMONIC RET



EXECUTION TIME/
MACHINE CYCLES
EXECUTED
20μs, PCI

FUNCTION
(P)←(STACK)

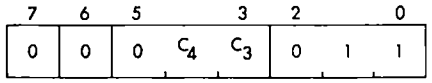
Return if Condition True
MNEMONIC RTc



EXECUTION TIME/
MACHINE CYCLES
EXECUTED
20μs, PCI

FUNCTION
(P)←(STACK) IF C=1,
(P)←(P)+1 IF C=0 12 μs, PCI

Return if Condition False
MNEMONIC RFc



EXECUTION TIME/
MACHINE CYCLES
EXECUTED

FUNCTION
(P)←(STACK) IF C=0 20 μs, PCI
(P)←(P)+1 IF C=1 12 μs, PCI

Input/Output Operations

Instructions in this category implement the transfer of data between a Microprocessor Series system and associated peripheral devices. Data is transferred between the accumulator and an addressed peripheral device in the form of 8-bit bytes at the rate of one byte per I/O instruction executed. Both instructions in this category have a one-byte format. The address field in the input instruction permits accessing of up to eight input-only peripheral devices, and the output instruction address field allows accessing of up to 24 output devices. Together, these fields permit the addressing of eight input devices and 24 output devices.

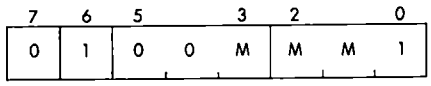
Three input/output instructions are reserved for special use by the processor module. This special use category includes the following instructions:

1. INPO—receive data from the asynchronous communications controller.
2. OUTO—send data to the asynchronous communications controller for transmission.
3. INP1—read asynchronous communications controller status.

The octal form for each of these instructions is listed below:

1. INP0=101_8 2. OUT0=121_8 3. INP1=103_8

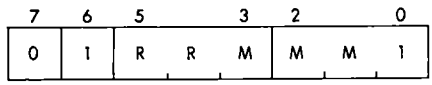
Input Data
MNEMONIC INP



EXECUTION TIME/
MACHINE CYCLES
EXECUTED
32 μs, PCI PCC

FUNCTION
(A)←(INPUT BUS)

Output Data
MNEMONIC OUT



EXECUTION TIME/
MACHINE CYCLES
EXECUTED
24 μs, PCI PCC

FUNCTION
(OUTPUT BUS)←(A)

Machine Operations

The four instructions in this category perform the basic machine control functions halt, restart, and interrupt enable and disable. When a halt instruction is executed, the machine enters the STOPPED state after completion of time state TS3. The interrupt enable/disable instructions ION and IOF are specialized input/output instructions which are defined as machine instructions because of the interrupt control functions performed. The octal form of these two instructions is listed below:

1. ION=123_8
2. IOF=125_8

The restart instruction is a one-byte unconditional call on any one of eight reserved locations within the first 64 words of memory. These eight locations are addressed directly by the restart instruction at octave intervals within these 64 memory words. Each of the eight locations can be the starting location of eight word subroutines which are entered under application-defined conditions. This instruction stores the content of the program counter in the STACK to permit a return to the program line.

All four instructions in this category have a one-byte format.

Halt

MNEMONIC HLT

7							1	0
0	0	0	0	0	0	0	X	

OR

7							0
1	1	1	1	1	1	1	1

EXECUTION TIME/ MACHINE CYCLES EXECUTED
16 μs, PCI

FUNCTION
HALT PROCESSOR

Enable Interrupts

MNEMONIC ION

7	6	5	4	3	2	1	0
0	1	0	1	0	0	1	1

EXECUTION TIME/ MACHINE CYCLES EXECUTED
32 μs, PCI PCC

FUNCTION
ENABLE EXTERNAL EVENT INTERRUPT

Disable Interrupts

MNEMONIC IOF

7	6	5	4	3	2	1	0
0	1	0	1	0	1	0	1

EXECUTION TIME/ MACHINE CYCLES EXECUTED
24 μs, PCI PCC

FUNCTION
DISABLE EXTERNAL EVENT INTERRUPT

Restart

MNEMONIC RST

7	6	5			2	1	0
0	0	A	A	A	1	0	1

EXECUTION TIME/ MACHINE CYCLES EXECUTED
20 μs, PCI

FUNCTION
(STACK)←(P), (P)←(00000000AAA000)

HARDWARE/ACCESSORIES

In order to be operational, a processor module must be equipped with a memory of sufficient size to contain the operating program. Such a memory can be configured from M7344 Read/Write Memory Modules and/or M7345 Read-Only Memory Modules. Programs can be bootstrapped into a Processor Module memory through use of the KC341-B Monitor/Control Panel (MCP).

The KC341-B serves as an address and data input station for operating Processor Modules and provides a visual display of addresses and data as well as a display of Processor Module machine states and operating status. The MCP also contains a 32-word random access scratchpad memory and a 256-word programmable read-only memory.

In a standard KC341-B Monitor/Control Panel, this programmable read-only memory contains a bootstrap loader program which occupies approximately 128 words of that memory. Figure 1 presents the signal names and shows the signal flow for the 50-pin header and cable assembly which connects the KC341-B Monitor/Control Panel with the processor module.

SPECIAL WIRING

The M7341 processor module is supplied with a set of special switches and factory-installed jumpers. Each of these switches and jumpers is depicted in the block diagram shown in Figure 1. The table presented summarizes the functions implemented by each switch and jumper.

INPUT/OUTPUT SYMBOLOGY

The direction of signal flow is indicated by arrows on the signal lines. Arrows toward the module indicate input signals; arrows away from the module indicate output signals (see Figure 1 and Figure 6).

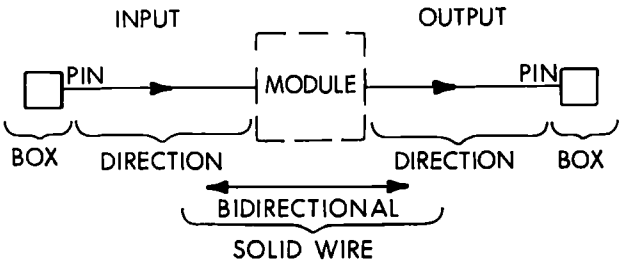


FIGURE 6. SIGNAL FLOW DIRECTION

Fan-in and fan-out (in TTL unit loads) are indicated by the number contained in the box associated with each pin. A box containing the letter B designates that the associated signal line is connected to a bus.

INPUT—The fan-in designation box always precedes the pin designation.

OUTPUT—The fan-out designation box always follows the pin designation.

SUMMARY OF M7341
SWITCH and JUMPER FUNCTIONS

Switch	Jumper	Open	Closed
DS1		Permits use of external receive clock through input signal URCLK H.	Internal 110-baud clock drives receiver/transmitter receiver logic. The signal URCLK H is the internal 110-baud rate output.
DS2		Permits use of external transmit clock through input signal UTCLK H.	Internal 110-baud clock drives receiver/transmitter transmit logic. The signal UTCLK H is the internal 110-baud rate output.
DS3		Two stop bits are selected to be appended to each data word during transmission.	One stop bit is selected to be appended to each data word during transmission.
DS4		The receiver/transmitter is configured to receive data in the active mode.	The receiver/transmitter is configured to receive data in the passive mode. Note that when operating in the passive mode, jumpers W3 and W4 must be cut.
DS5		The receiver/transmitter is configured to transmit data in the active mode.	The receiver/transmitter is configured to transmit data in the passive mode. Note that when operating in this mode, jumpers W1 and W2 must be cut.
	W1 W2 W3 W4	These machine-installed jumpers, when all cut and with switches DS4 and DS5 closed, implement transmission and receipt of data by the receiver/transmitter in the passive mode.	When all jumpers are in place and switches DS4 and DS5 open, data is transmitted and received by the receiver/transmitter in the active mode.
DS6		Receiver/transmitter device selection logic disabled.	Receiver/transmitter device selection logic enabled.
DS7		Switch should be open for test purposes only.	Switch closed for normal operation.

POWER REQUIREMENTS

	Power Consumption		Voltage Regulation	Ripple Regulation		Pins
Volts	mA	Watts			Power	Ground
+ 5	1620	8.00	±5%	50 mV	AA2, BA2, CA2, DA2	AC2, AT1, BC2, BT1, CC2, CT1,
−15	150	2.25	±3%	50 mV	AB2	DC2, DT1

digital

DIGITAL EQUIPMENT CORPORATION

COMPONENTS GROUP HEADQUARTERS

ONE IRON WAY, MARLBOROUGH, MASSACHUSETTS 01752

(617) 481-7400 TWX: 710-347-0348

For detailed information about products and policies, call 800-225-9480 toll-free (USA only); Massachusetts residents call (617) 481-7400.

DIGITAL EQUIPMENT CORPORATION, Maynard, Massachusetts, Telephone: (617) 897-5111 • ARIZONA, Phoenix • CALIFORNIA, Sunnyvale, Santa Ana, Los Angeles, Oakland, San Diego and San Francisco (Mountain View) • COLORADO, Englewood • CONNECTICUT, Meriden, Fairfield • DISTRICT OF COLUMBIA, Washington (Lanham, Md.) • FLORIDA, Orlando • GEORGIA, Atlanta • ILLINOIS, Northbrook • INDIANA, Indianapolis • LOUISIANA, Metairie • MASSACHUSETTS, Marlborough and Waltham • MICHIGAN, Ann Arbor and Detroit (Southfield) • MINNESOTA, Minneapolis • MISSOURI, Kansas City and Maryland Heights • NEW JERSEY, Fairfield, Metuchen and Princeton • NEW MEXICO, Albuquerque • NEW YORK, Huntington Station, Manhattan, New York, Syracuse and Rochester • NORTH CAROLINA, Durham/Chapel Hill • OHIO, Cleveland, Dayton and Euclid • OKLAHOMA, Tulsa • OREGON, Portland • PENNSYLVANIA, Bluebell and Pittsburgh • TENNESSEE, Knoxville • TEXAS, Dallas and Houston • UTAH, Salt Lake City • WASHINGTON, Bellevue • WISCONSIN, Milwaukee • ARGENTINA, Buenos Aires • AUSTRALIA, Adelaide, Brisbane, Canberra, Melbourne, Perth and Sydney • AUSTRIA, Vienna • BELGIUM, Brussels • BRAZIL, Rio de Janeiro, Sao Paulo and Porto Alegre • CANADA, Calgary, Montreal, Ottawa, Toronto and Vancouver • CHILE, Santiago • DENMARK, Copenhagen • FINLAND, Helsinki • FRANCE, Grenoble and Paris • GERMANY, Berlin, Cologne, Hannover, Frankfurt, Munich and Stuttgart • INDIA, Bombay • ISRAEL, Tel Aviv • ITALY, Milan and Turin • JAPAN, Tokyo • MEXICO, Mexico City • NETHERLANDS, The Hague • NEW ZEALAND, Auckland • NORWAY, Oslo • PHILIPPINES, Manila • PUERTO RICO, Santurce • SPAIN, Barcelona and Madrid • SWEDEN, Stockholm • SWITZERLAND, Geneva and Zurich • UNITED KINGDOM, Birmingham, Bristol, Edinburgh, London, Manchester and Reading • VENEZUELA, Caracas

PRINTED IN U.S.A. 5804 00874 3985/M 10 10