Zilog Data Book

Zilog

Introduction

Zilog's Component, Board and Development System Data Book provides design engineers with detailed specifications of the Zilog products they will use in constructing their own micro-computer systems. Included are each of the following:

- **Z-80 family of MOS LSI microcomputer components**
- **Zilog memory components**
- **MCB series of microcomputer board products**
- **ZDS-1 series of microcomputer development systems**

The Z-80 family of MOS LSI microcomputer components form the nucleus of this group of products. Together they provide the highest performance family of 8-bit components available, permitting system designers to enjoy unprecedented benefits at modest costs. Typically, designers will choose component level solutions to their system problems when the volume of production for a given system merits the investment in engineering (usually 500 or more systems).

In lower volume applications or in situations where problems other than microcomputer design require most of the engineer's time and talent, the MCB series of microcomputer board products permits the use of Z-80 technology without large investments in PC Board design. MCB boards can get microcomputer products in production faster and more reliably with lower front-end costs. Many designers of higher volume products also use MCB boards for early prototyping and pilot production—moving to their own PC board designs as the volume increases.

The ZDS-1 series of development systems provides hardware and software design and diagnostic tools for use in the development Z-80 based products at either the component or the board level. Hardware tools include Z-80 CPU hardware emulation, tracing bus cycles, stopping emulation on pre-defined bus conditions, verification of clock integrity, and mapping of development system memory into the space of the system under test. For software development, a complete disk operating system, text editor, macro assembler, and linker are provided together with comprehensive software debug tools. Optional programming languages such as PLZ, FORTRAN, and BASIC provide additional design capability as well as a means for using the ZDS as a general purpose laboratory computer.

Beyond the products in this Data Book, Zilog also manufactures a series of general purpose microcomputers for OEM or large end customer use. These products are currently described by individual specifications and brochures. These will be collated to form a Microcomputer System Data Book in the spring of 1979.

Table of Contents

Section 1
Components

Zilog

Section 1
Components

Zilog components fall into two categories: the Z-80 family of microprocessor circuits and the Z6100 family of read/write memories.

The Z-80 Microprocessor Family

The Z-80 CPU has rapidly established itself as the most powerful, most sophisticated and most versatile 8-bit microprocessor. Although source-code compatible with the popular 8080A microprocessor, the Z-80 CPU has many additional features that simplify hardware requirements and programming while increasing execution speed.

- Dual register set allows high-speed context switching and interrupt processing.
- Two index registers give additional memory addressing flexibility.
- On-chip refresh logic simplifies interfacing with dynamic RAMs.
- Powerful block move and string manipulation as well as bit manipulation instructions reduce programming effort, program size and execution time.

Four peripheral functions are generally required in microcomputer systems: parallel input/output, serial input/output, counting/timing, and direct memory access. These basic functions are provided by four Z-80 peripheral circuits that complement the Z-80 CPU:

- **The Z-80 PIO** (Parallel Input/Output) offers two 8-bit I/O ports with individual handshake and pattern-detection capability.
- **The Z-80 CTC** (Counter/Timer Circuit) offers four versatile counter/timers with individually programmable prescalers.
- **The Z-80 SIO** (Serial Input/Output) is a powerful dual-channel device that can operate as an asynchronous or synchronous receiver/transmitter. The Z-80 SIO supports all popular data communications protocols including Bisync, SDLC and HDLC. Additionally, the Z-80 SIO offers on-chip parity and CRC generation/checking, FIFO buffering, and flag and frame detection/generation logic.
- **The Z-80 DMA** (Direct Memory Access) is a sophisticated controller that transfers data directly between any two ports (I/O and memory, for example). The Z-80 DMA can search for data patterns even when transferring.

The normal clock rate for Z-80 components is 2.5 MHz, but all are available in a Z-80A version with a maximum clock rate of 4 MHz. A vectored priority interrupt structure common to all Z-80 components handles up to 128 vectored interrupts defined by loadable registers in both the peripherals and the CPU. No external interrupt controller is required.

Read/Write Memories

In addition to microprocessor components, Zilog offers three 4K static RAMs and a 16K dynamic RAM. All three 4K static RAMs contain on-chip address registers, activated by the Chip Enable input. Polysilicon load resistors used in the memory arrays of the static RAMs result in a very small die size.

- **The Z6104** is organized as 4096 addresses by one bit.
- **The Z6114** is organized as 1024 addresses by four bits, with four bidirectional I/O lines that make it ideal for small systems.
- **The Z6142** is identical to the Z6114, except that it offers an Output Enable input and an additional Chip Enable input for greater flexibility in small microprocessor-based systems.
- **The Z6116** is an industry-standard 16K by one bit dynamic RAM, available in a range of access times.

These Zilog components are manufactured with n-channel depletion-load silicon-gate technology using ion-implantation and shallow junctions for high performance and projection printing on four-inch wafers for high yield and low cost.

Zilog

Product Specification

Z-80® CPU
Z-80A CPU

The Zilog Z80 product line is a complete set of microcomputer components, development systems and support software. The Z80 microcomputer component set includes all of the circuits necessary to build high-performance microcomputer systems with virtually no other logic and a minimum number of low cost standard memory elements.

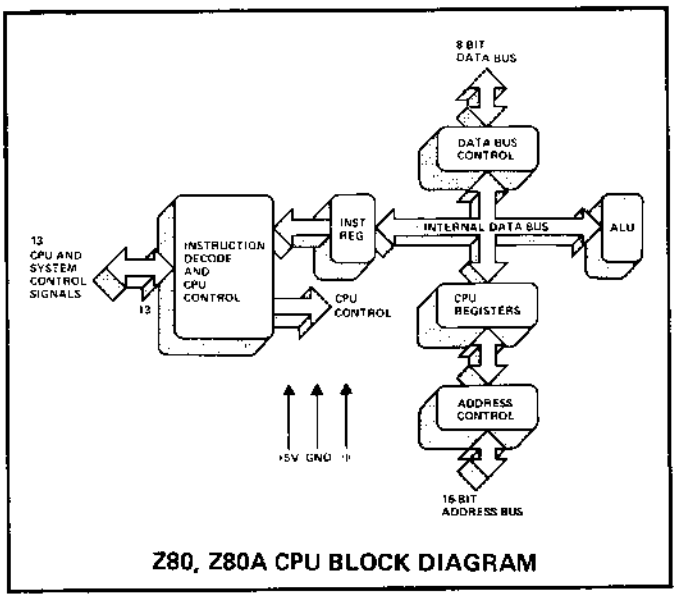
The Z80 and Z80A CPU's are third generation single chip microprocessors with unrivaled computational power. This increased computational power results in higher system through-put and more efficient memory utilization when compared to second generation microprocessors. In addition, the Z80 and Z80A CPU's are very easy to implement into a system because of their single voltage requirement plus all output signals are fully decoded and timed to control standard memory or peripheral circuits. The circuit is implemented using an N-channel, ion implanted, silicon gate MOS process.

Figure 1 is a block diagram of the CPU, Figure 2 details the internal register configuration which contains 208 bits of Read/Write memory that are accessible to the programmer. The registers include two sets of six general purpose registers that may be used individually as 8-bit registers or as 16-bit register pairs. There are also two sets of accumulator and flag registers. The programmer has access to either set of main or alternate registers through a group of exchange instructions. This alternate set allows foreground/background mode of operation or may be reserved for very fast Interrupt response. Each CPU also contains a 16-bit stack pointer which permits simple implementation of multiple level interrupts, unlimited subroutine nesting and simplification of many types of data handling.

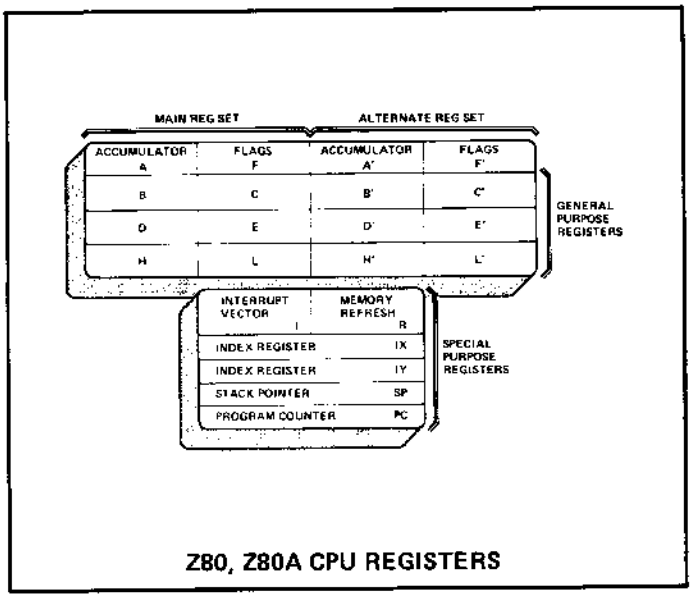
The two 16-bit index registers allow tabular data manipulation and easy implementation of relocatable code. The Refresh register provides for automatic, totally transparent refresh of external dynamic memories. The I register is used in a powerful interrupt response mode to form the upper 8 bits of a pointer to a interrupt service address table, while the interrupting device supplies the lower 8 bits of the pointer. An indirect call is then made to this service address.

FEATURES

- Single chip, N-channel Silicon Gate CPU.
- 158 instructions—includes all 78 of the 8080A instructions with total software compatibility. New instructions include 4-, 8- and 16-bit operations with more useful addressing modes such as indexed, bit and relative.
- 17 internal registers.
- Three modes of fast interrupt response plus a non-maskable interrupt.
- Directly interfaces standard speed static or dynamic memories with virtually no external logic.
- 1.0 μs instruction execution speed.
- Single 5 VDC supply and single-phase 5 volt Clock.
- Out-performs any other single chip microcomputer in 4-, 8-, or 16-bit applications.
- All pins TTL Compatible
- Built-in dynamic RAM refresh circuitry.



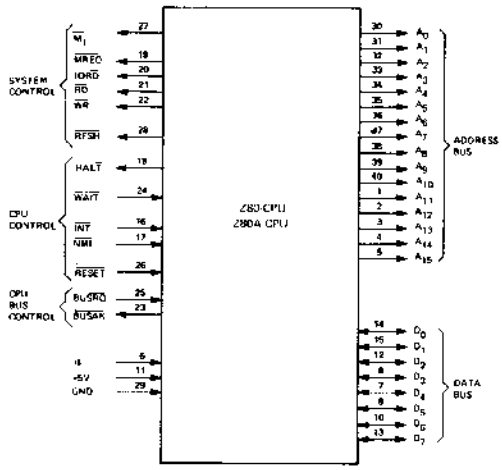
Z80, Z80A CPU BLOCK DIAGRAM



Z80, Z80A CPU REGISTERS

Zilog

Z-80, Z-80A CPU Pin Description



Z80, Z80A CPU PIN CONFIGURATION

A_0-A_{15} (Address Bus) Tri-state output, active high. A_0-A_{15} constitute a 16-bit address bus. The address bus provides the address for memory (up to 64K bytes) data exchanges and for I/O device data exchanges.

D_0-D_7 (Data Bus) Tri-state input/output, active high. D_0-D_7 constitute an 8-bit bidirectional data bus. The data bus is used for data exchanges with memory and I/O devices.

$\overline{M_1}$ (Machine Cycle one) Output, active low. $\overline{M_1}$ indicates that the current machine cycle is the OP code fetch cycle of an instruction execution.

$\overline{MREQ}$ (Memory Request) Tri-state output, active low. The memory request signal indicates that the address bus holds a valid address for a memory read or memory write operation.

$\overline{IORQ}$ (Input/ Output Request) Tri-state output, active low. The $\overline{IORQ}$ signal indicates that the lower half of the address bus holds a valid I/O address for a I/O read or write operation. An $\overline{IORQ}$ signal is also generated when an interrupt is being acknowledged to indicate that an interrupt response vector can be placed on the data bus.

$\overline{RD}$ (Memory Read) Tri-state output, active low. $\overline{RD}$ 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.

$\overline{WR}$ (Memory Write) Tri-state output, active low. $\overline{WR}$ indicates that the CPU data bus holds valid data to be stored in the addressed memory or I/O device.

$\overline{RFSH}$ (Refresh) Output, active low. $\overline{RFSH}$ indicates that the lower 7 bits of the address bus contain a refresh address for dynamic memories and the current $\overline{MREQ}$ signal should be used to do a refresh read to all dynamic memories.

$\overline{HALT}$ (Halt state) Output, active low. $\overline{HALT}$ indicates that the CPU has executed a HALT software instruction and is awaiting either a non-maskable or a maskable interrupt (with the mask enabled) before operation can resume. While halted, the CPU executes NOP's to maintain memory refresh activity.

$\overline{WAIT}$ (Wait) Input, active low. $\overline{WAIT}$ indicates to the Z-80 CPU that the addressed memory or I/O devices are not ready for a data transfer. The CPU continues to enter wait states for as long as this signal is active.

$\overline{INT}$ (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.

$\overline{NMI}$ (Non Maskable Interrupt) Input, active low. The non-maskable nterrupt request line has a higher priority than $\overline{INT}$ and is always recognized at the end of the current instruction, independent of the status of the interrupt enable flip-flop. $\overline{NMI}$ automatically forces the Z-80 CPU to restart to location 0066_H.

$\overline{RESET}$ Input, active low. $\overline{RESET}$ initializes the CPU as follows: reset interrupt enable flip-flop, clear PC and registers I and R and set interrupt to 8080A mode. During reset time, the address and data bus go to a high impedance state and all control output signals go to the inactive state.

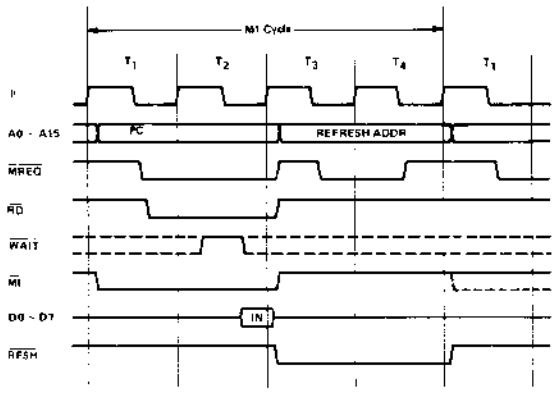
$\overline{BUSRQ}$ (Bus Request) Input, active low. The bus request signal has a higher priority than $\overline{NMI}$ and is always recognized at the end of the current machine cycle and is used to request the CPU address bus, data bus and tri-state output control signals to go to a high impedance state so that other devices can control these busses.

$\overline{BUSAK}$ (Bus Acknowledge) Output, active low. Bus acknowledge is used to indicate to the requesting device that the CPU address bus, data bus and tri-state control bus signals have been set to their high impedance state and the external device can now control these signals

Timing Waveforms

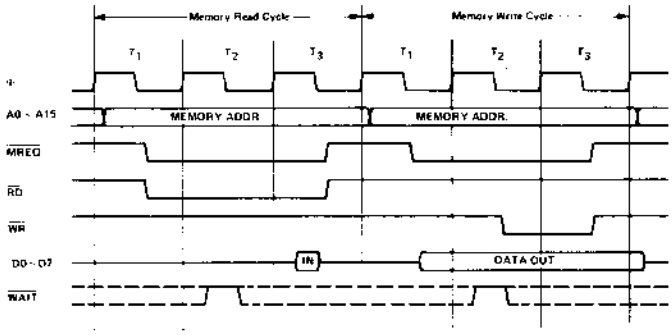
INSTRUCTION OP CODE FETCH

The program counter content (PC) is placed on the address bus immediately at the start of the cycle. One half clock time later $\overline{MREQ}$ goes active. The falling edge of $\overline{MREQ}$ can be used directly as a chip enable to dynamic memories. $\overline{RD}$ when active indicates that the memory data should be enabled onto the CPU data bus. The CPU samples data with the rising edge of the clock state T_3. Clock states T_3 and T_4 of a fetch cycle are used to refresh dynamic memories while the CPU is internally decoding and executing the instruction. The refresh control signal $\overline{RFSH}$ indicates that a refresh read of all dynamic memories should be accomplished.



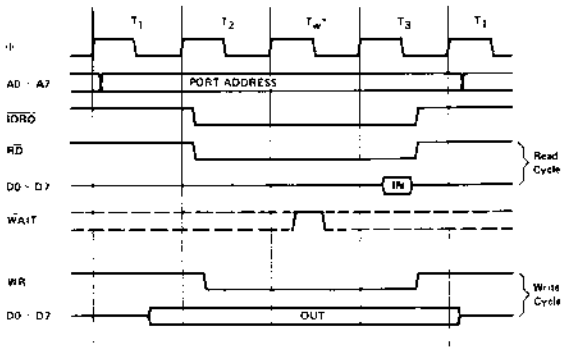
MEMORY READ OR WRITE CYCLES

Illustrated here is the timing of memory read or write cycles other than an OP code fetch (M_1 cycle). The $\overline{MREQ}$ and $\overline{RD}$ signals are used exactly as in the fetch cycle. In the case of a memory write cycle, the $\overline{MREQ}$ also becomes active when the address bus is stable so that it can be used directly as a chip enable for dynamic memories. The $\overline{WR}$ line is active when data on the data bus is stable so that it can be used directly as a R/W pulse to virtually any type of semiconductor memory.



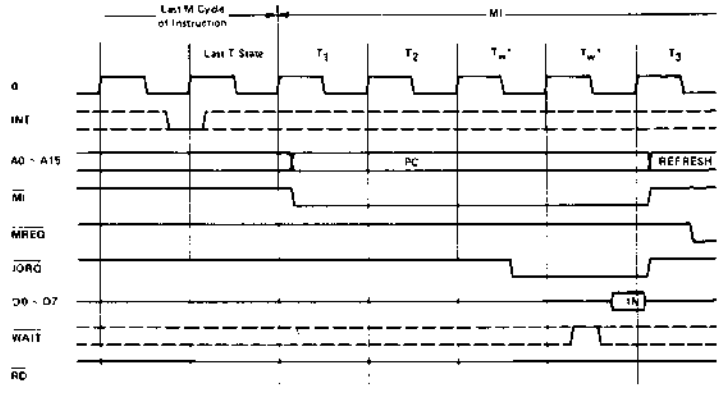
INPUT OR OUTPUT CYCLES

Illustrated here is the timing for an I/O read or I/O write operation. Notice that during I/O operations a single wait state is automatically inserted (Tw*). The reason for this is that during I/O operations this extra state allows sufficient time for an I/O port to decode its address and activate the $\overline{WAIT}$ line if a wait is required.



INTERRUPT REQUEST/ACKNOWLEDGE CYCLE

The interrupt signal is sampled by the CPU with the rising edge of the last clock at the end of any instruction. When an interrupt is accepted, a special M_1 cycle is generated. During this M_1 cycle, the $\overline{IORQ}$ signal becomes active (instead of $\overline{MREQ}$) to indicate that the interrupting device can place an 8-bit vector on the data bus. Two wait states (Tw*) are automatically added to this cycle so that a ripple priority interrupt scheme, such as the one used in the Z80 peripheral controllers, can be easily implemented.



Instruction Set

The following is a summary of the Z80, Z80A instruction set showing the assembly language mnemonic and the symbolic operation performed by the instruction. A more detailed listing appears in the Z80-CPU technical manual, and assembly language programming manual. The instructions are divided into the following categories:

- 8-bit loads
- 16-bit loads
- Exchanges
- Memory Block Moves
- Memory Block Searches
- 8-bit arithmetic and logic
- 16-bit arithmetic
- General purpose Accumulator & Flag Operations
- Miscellaneous Group
- Rotates and Shifts
- Bit Set, Reset and Test
- Input and Output
- Jumps
- Calls
- Restarts
- Returns

In the table the following terminology is used.

- b ≡ a bit number in any 8-bit register or memory location
- cc ≡ flag condition code
  - NZ ≡ non zero
  - Z ≡ zero
  - NC ≡ non carry
  - C ≡ carry
  - PO ≡ Parity odd or no over flow
  - PE ≡ Parity even or over flow
  - P ≡ Positive
  - M ≡ Negative (minus)
- d ≡ any 8-bit destination register or memory location
- dd ≡ any 16-bit destination register or memory location
- e ≡ 8-bit signed 2's complement displacement used in relative jumps and indexed addressing
- L ≡ 8 special call locations in page zero. In decimal notation these are 0, 8, 16, 24, 32, 40, 48 and 56
- n ≡ any 8-bit binary number
- nn ≡ any 16-bit binary number
- r ≡ any 8-bit general purpose register (A, B, C, D, E, H, or L)
- s ≡ any 8-bit source register or memory location
- s_b ≡ a bit in a specific 8-bit register or memory location
- ss ≡ any 16-bit source register or memory location
- subscript "L" ≡ the low order 8 bits of a 16-bit register
- subscript "H" ≡ the high order 8 bits of a 16-bit register
- ( ) ≡ the contents within the ( ) are to be used as a pointer to a memory location or I/O port number

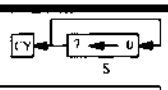
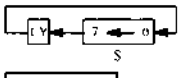
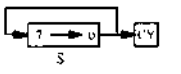
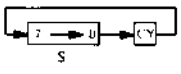
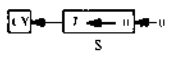
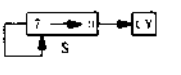
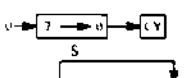
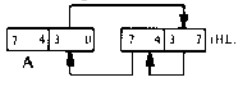
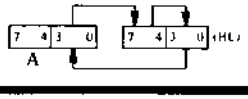
8-bit registers are A, B, C, D, E, H, L, I and R
16-bit register pairs are AF, BC, DE and HL
16-bit registers are SP, PC, IX and IY

Addressing Modes implemented include combinations of the following:

- Immediate
- Immediate extended
- Modified Page Zero
- Relative
- Extended
- Indexed
- Register
- Implied
- Register Indirect
- Bit

	Mnemonic	Symbolic Operation	Comments
8-BIT LOADS	LD r, s	r ← s	s ≡ r, n, (HL), (IX+e), (IY+e)
	LD d, r	d ← r	d ≡ (HL), r (IX+e), (IY+e)
	LD d, n	d ← n	d ≡ (HL), (IX+e), (IY+e)
	LD A, s	A ← s	s ≡ (BC), (DE), (nn), I, R
	LD d, A	d ← A	d ≡ (BC), (DE), (nn), I, R
16-BIT LOADS	LD dd, nn	dd ← nn	dd ≡ BC, DE, HL, SP, IX, IY
	LD dd, (nn)	dd ← (nn)	dd ≡ BC, DE, HL, SP, IX, IY
	LD (nn), ss	(nn) ← ss	ss ≡ BC, DE, HL, SP, IX, IY
	LD SP, ss	SP ← ss	ss = HL, IX, IY
	PUSH ss	$(SP-1) \leftarrow ss_H$; $(SP-2) \leftarrow ss_L$	ss = BC, DE, HL, AF, IX, IY
	POP dd	$dd_L \leftarrow (SP)$; $dd_H \leftarrow (SP+1)$	dd = BC, DE, HL, AF, IX, IY
EXCHANGES	EX DE, HL	DE ↔ HL	
	EX AF, AF'	AF ↔ AF'	
	EXX	$\begin{pmatrix} BC \\ DE \\ HL \end{pmatrix} \leftrightarrow \begin{pmatrix} BC' \\ DE' \\ HL' \end{pmatrix}$	
	EX (SP), ss	$(SP) \leftrightarrow ss_L$, $(SP+1) \leftrightarrow ss_H$	ss ≡ HL, IX, IY

	Mnemonic	Symbolic Operation	Comments
MEMORY BLOCK MOVES	LDI	(DE) ← (HL), DE ← DE+1 HL ← HL+1, BC ← BC-1	
	LDIR	(DE) ← (HL), DE ← DE+1 HL ← HL+1, BC ← BC-1 Repeat until BC = 0	
	LDD	(DE) ← (HL), DE ← DE-1 HL ← HL-1, BC ← BC-1	
	LDDR	(DE) ← (HL), DE ← DE-1 HL ← HL-1, BC ← BC-1 Repeat until BC = 0	
MEMORY BLOCK SEARCHES	CPI	A-(HL), HL ← HL+1 BC ← BC-1	
	CPIR	A-(HL), HL ← HL+1 BC ← BC-1, Repeat until BC = 0 or A = (HL)	A-(HL) sets the flags only. A is not affected
	CPD	A-(HL), HL ← HL-1 BC ← BC-1	
	CPDR	A-(HL), HL ← HL-1 BC ← BC-1, Repeat until BC= 0 or A = (HL)	
8-BIT ALU	ADD s	A ← A + s	
	ADC s	A ← A + s + CY	CY is the carry flag
	SUB s	A ← A - s	
	SBC s	A ← A - s - CY	s ≡ r, n, (HL) (IX+e), (IY+e)
	AND s	A ← A ∧ s	
	OR s	A ← A ∨ s	
	XOR s	A ← A ⊕ s	

	Mnemonic	Symbolic Operation	Comments
8-BIT ALU	CP s	A - s	s = r, n (HL) (IX+e), (IY+e)
	INC d	d ← d + 1	d = r, (HL) (IX+e), (IY+e)
	DEC d	d ← d - 1	
16-BIT ARITHMETIC	ADD HL, ss	HL ← HL + ss	ss ≡ BC, DE HL, SP
	ADC HL, ss	HL ← HL + ss + CY	
	SBC HL, ss	HL ← HL - ss - CY	
	ADD IX, ss	IX ← IX + ss	ss ≡ BC, DE, IX, SP
	ADD IY, ss	IY ← IY + ss	ss ≡ BC, DE, IY, SP
	INC dd	dd ← dd + 1	dd ≡ BC, DE, HL, SP, IX, IY
	DEC dd	dd ← dd - 1	dd ≡ BC, DE, HL, SP, IX, IY
GP ACC. & FLAG	DAA	Converts A contents into packed BCD following add or subtract.	Operands must be in packed BCD format
	CPL	$A \leftarrow \overline{A}$	
	NEG	A ← 00 - A	
	CCF	$CY \leftarrow \overline{CY}$	
	SCF	CY ← 1	
MISCELLANEOUS	NOP	No operation	
	HALT	Halt CPU	
	DI	Disable Interrupts	
	EI	Enable Interrupts	
	IM 0	Set interrupt mode 0	8080A mode
	IM 1	Set interrupt mode 1	Call to 0038_H
	IM 2	Set interrupt mode 2	Indirect Call
ROTATES AND SHIFTS	RLC s		s ≡ r, (HL) (IX+e), (IY+e)
	RL s		
	RRC s		
	RR s		
	SLA s		
	SRA s		
	SRL s		
	RLD		
	RRD		

	Mnemonic	Symbolic Operation	Comments
BIT S, R, & T	BIT b, s	$Z \leftarrow \overline{s_b}$	Z is zero flag
	SET b, s	$s_b \leftarrow 1$	s ≡ r, (HL) (IX+e), (IY+e)
	RES b, s	$s_b \leftarrow 0$	
INPUT AND OUTPUT	IN A, (n)	A ← (n)	
	IN r, (C)	r ← (C)	Set flags
	INI	(HL) ← (C), HL ← HL + 1 B ← B - 1	
	INIR	(HL) ← (C), HL ← HL + 1 B ← B - 1 Repeat until B = 0	
	IND	(HL) ← (C), HL ← HL - 1 B ← B - 1	
	INDR	(HL) ← (C), HL ← HL - 1 B ← B - 1 Repeat until B = 0	
	OUT(n), A	(n) ← A	
	OUT(C), r	(C) ← r	
	OUTI	(C) ← (HL), HL ← HL + 1 B ← B - 1	
	OTIR	(C) ← (HL), HL ← HL + 1 B ← B - 1 Repeat until B = 0	
	OUTD	(C) ← (HL), HL ← HL - 1 B ← B - 1	
	OTDR	(C) ← (HL), HL ← HL - 1 B ← B - 1 Repeat until B = 0	
JUMPS	JP nn	PC ← nn	
	JP cc, nn	If condition cc is true PC ← nn, else continue	cc ≡ NZ PO, Z PE, NC P, C M
	JR e	PC ← PC + e	
	JR kk, e	If condition kk is true PC ← PC + e, else continue	kk ≡ NZ NC, Z C
	JP (ss)	PC ← ss	ss = HL, IX, IY
	DJNZ e	B ← B - 1, if B = 0 continue, else PC ← PC + e	
CALLS	CALL nn	$(SP-1) \leftarrow PC_H$ $(SP-2) \leftarrow PC_L$, PC ← nn	
	CALL cc, nn	If condition cc is false continue, else same as CALL nn	cc ≡ NZ PO, Z PE, NC P, C M
RESTARTS	RST L	$(SP-1) \leftarrow PC_H$ $(SP-2) \leftarrow PC_L$, $PC_H \leftarrow 0$ $PC_L \leftarrow L$	
RETURNS	RET	$PC_L \leftarrow (SP)$, $PC_H \leftarrow (SP+1)$	
	RET cc	If condition cc is false continue, else same as RET	cc ≡ NZ PO, Z PE, NC P, C M
	RETI	Return from interrupt, same as RET	
	RETN	Return from non-maskable interrupt	

Zilog

Z-80 CPU A.C. Characteristics

T_A = 0°C to 70°C, Vcc = +5V ± 5%, Unless Otherwise Noted.

Signal	Symbol	Parameter	Min	Max	Unit	Test Condition
Φ	t_c	Clock Period	4	[12]	μsec	
	$t_{w(\Phi H)}$	Clock Pulse Width, Clock High	180	[E]	nsec	
	$t_{w(\Phi L)}$	Clock Pulse Width, Clock Low	180	2000	nsec	
	t_r, t_f	Clock Rise and Fall Time		30	nsec	
A_{0-15}	$t_{D(AD)}$	Address Output Delay		145	nsec	C_L = 50pF
	$t_{F(AD)}$	Delay to Float		110	nsec	
	t_{acm}	Address Stable Prior to $\overline{MREQ}$ (Memory Cycle)	[1]		nsec	
	t_{aci}	Address Stable Prior to $\overline{IORQ}$, $\overline{RD}$ or $\overline{WR}$ (I/O Cycle)	[2]		nsec	
	t_{ca}	Address Stable from $\overline{RD}$, $\overline{WR}$, $\overline{IORQ}$ or $\overline{MREQ}$	[3]		nsec	
	t_{caf}	Address Stable From $\overline{RD}$ or $\overline{WR}$ During Float	[4]		nsec	
D_{0-7}	$t_{D(D)}$	Data Output Delay		230	nsec	C_L = 50pF
	$t_{F(D)}$	Delay to Float During Write Cycle		90	nsec	
	$t_{S\Phi(D)}$	Data Setup Time to Rising Edge of Clock During M1 Cycle	50		nsec	
	$t_{S\overline{\Phi}(D)}$	Data Setup Time to Falling Edge of Clock During M2 to M5	60		nsec	
	t_{dcm}	Data Stable Prior to $\overline{WR}$ (Memory Cycle)	[5]		nsec	
	t_{dci}	Data Stable Prior to $\overline{WR}$ (I/O Cycle)	[6]		nsec	
	t_{cdf}	Data Stable From $\overline{WR}$	[7]			
	t_H	Any Hold Time for Setup Time	0		nsec	
$\overline{MREQ}$	$t_{DL\overline{\Phi}(MR)}$	$\overline{MREQ}$ Delay From Falling Edge of Clock, $\overline{MREQ}$ Low		100	nsec	C_L = 50pF
	$t_{DH\Phi(MR)}$	$\overline{MREQ}$ Delay From Rising Edge of Clock, $\overline{MREQ}$ High		100	nsec	
	$t_{DH\overline{\Phi}(MR)}$	$\overline{MREQ}$ Delay From Falling Edge of Clock, $\overline{MREQ}$ High		100	nsec	
	$t_{w(\overline{MRL})}$	Pulse Width, $\overline{MREQ}$ Low	[8]		nsec	
	$t_{w(\overline{MRH})}$	Pulse Width, $\overline{MREQ}$ High	[9]		nsec	
$\overline{IORQ}$	$t_{DL\Phi(IR)}$	$\overline{IORQ}$ Delay From Rising Edge of Clock, $\overline{IORQ}$ Low		90	nsec	C_L = 50pF
	$t_{DL\overline{\Phi}(IR)}$	$\overline{IORQ}$ Delay From Falling Edge of Clock, $\overline{IORQ}$ Low		110	nsec	
	$t_{DH\Phi(IR)}$	$\overline{IORQ}$ Delay From Rising Edge of Clock, $\overline{IORQ}$ High		100	nsec	
	$t_{DH\overline{\Phi}(IR)}$	$\overline{IORQ}$ Delay From Falling Edge of Clock, $\overline{IORQ}$ High		110	nsec	
$\overline{RD}$	$t_{DL\Phi(RD)}$	$\overline{RD}$ Delay From Rising Edge of Clock, $\overline{RD}$ Low		100	nsec	C_L = 50pF
	$t_{DL\overline{\Phi}(RD)}$	$\overline{RD}$ Delay From Falling Edge of Clock, $\overline{RD}$ Low		130	nsec	
	$t_{DH\Phi(RD)}$	$\overline{RD}$ Delay From Rising Edge of Clock, $\overline{RD}$ High		100	nsec	
	$t_{DH\overline{\Phi}(RD)}$	$\overline{RD}$ Delay From Falling Edge of Clock, $\overline{RD}$ High		110	nsec	
$\overline{WR}$	$t_{DL\Phi(WR)}$	$\overline{WR}$ Delay From Rising Edge of Clock, $\overline{WR}$ Low		80	nsec	C_L = 50pF
	$t_{DL\overline{\Phi}(WR)}$	$\overline{WR}$ Delay From Falling Edge of Clock, $\overline{WR}$ Low		90	nsec	
	$t_{DH\overline{\Phi}(WR)}$	$\overline{WR}$ Delay From Falling Edge of Clock, $\overline{WR}$ High		100	nsec	
	$t_{w(\overline{WRL})}$	Pulse Width, $\overline{WR}$ Low	[10]		nsec	
$\overline{M1}$	$t_{DL(M1)}$	$\overline{M1}$ Delay From Rising Edge of Clock, $\overline{M1}$ Low		130	nsec	C_L = 50pF
	$t_{DH(M1)}$	$\overline{M1}$ Delay From Rising Edge of Clock, $\overline{M1}$ High		130	nsec	
$\overline{RFSH}$	$t_{DL(RF)}$	$\overline{RFSH}$ Delay From Rising Edge of Clock, $\overline{RFSH}$ Low		180	nsec	C_L = 50pF
	$t_{DH(RF)}$	$\overline{RFSH}$ Delay From Rising Edge of Clock, $\overline{RFSH}$ High		150	nsec	
$\overline{WAIT}$	$t_{s(WT)}$	$\overline{WAIT}$ Setup Time to Falling Edge of Clock	70		nsec	
$\overline{HALT}$	$t_{D(HT)}$	$\overline{HALT}$ Delay Time From Falling Edge of Clock		300	nsec	C_L = 50pF
$\overline{INT}$	$t_{s(IT)}$	$\overline{INT}$ Setup Time to Rising Edge of Clock	80		nsec	
$\overline{NMI}$	$t_{w(\overline{NML})}$	Pulse Width, $\overline{NMI}$ Low	80		nsec	
$\overline{BUSRQ}$	$t_{s(BQ)}$	$\overline{BUSRQ}$ Setup Time to Rising Edge of Clock	80		nsec	
$\overline{BUSAK}$	$t_{DL(BA)}$	$\overline{BUSAK}$ Delay From Rising Edge of Clock, $\overline{BUSAK}$ Low		120	nsec	C_L = 50pF
	$t_{DH(BA)}$	$\overline{BUSAK}$ Delay From Falling Edge of Clock, $\overline{BUSAK}$ High		110	nsec	
$\overline{RESET}$	$t_{s(RS)}$	$\overline{RESET}$ Setup Time to Rising Edge of Clock	90		nsec	
	$t_{F(C)}$	Delay to Float ($\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$ and $\overline{WR}$)		100	nsec	
	t_{mr}	$\overline{M1}$ Stable Prior to $\overline{IORQ}$ (Interrupt Ack.)	[11]		nsec	

[1] $t_{acm} = t_{w(\Phi H)} + t_f - 75$

[2] $t_{aci} = t_c - 80$

[3] $t_{ca} = t_{w(\Phi L)} + t_r - 40$

[4] $t_{caf} = t_{w(\Phi L)} + t_r - 60$

[5] $t_{dcm} = t_c - 210$

[6] $t_{dci} = t_{w(\Phi L)} + t_r - 210$

[7] $t_{cdf} = t_{w(\Phi L)} + t_r - 80$

[8] $t_{w(MRL)} = t_c - 40$

[9] $t_{w(MRH)} = t_{w(\Phi H)} + t_r - 30$

[10] $t_{w(\overline{WRL})} = t_c - 40$

[11] $t_{mr} = 2t_c + t_{w(\Phi H)} + t_f - 80$

[12] $t_c = t_{w(\Phi H)} + t_{w(\Phi L)} + t_r + t_f$

NOTES:

A Data should be enabled onto the CPU data bus when $\overline{RD}$ is active. During interrupt acknowledge data should be enabled when $\overline{M1}$ and $\overline{IORQ}$ are both active.

B All control signals are internally synchronized, so they may be totally asynchronous with respect to the clock.

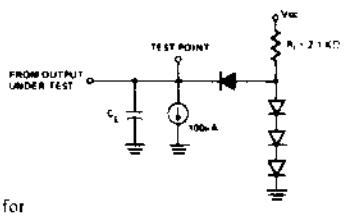
C The $\overline{RESET}$ signal must be active for a minimum of 3 clock cycles.

D Output Delay vs. Loaded Capacitance

TA = 70°C Vcc = +5V ±5%

Add 10nsec delay for each 50pf increase in load up to a maximum of 200pf for the data bus & 100pf for address & control lines.

E Although static by design, testing guarantees $t_{w(\Phi H)}$ of 200 μsec maximum.



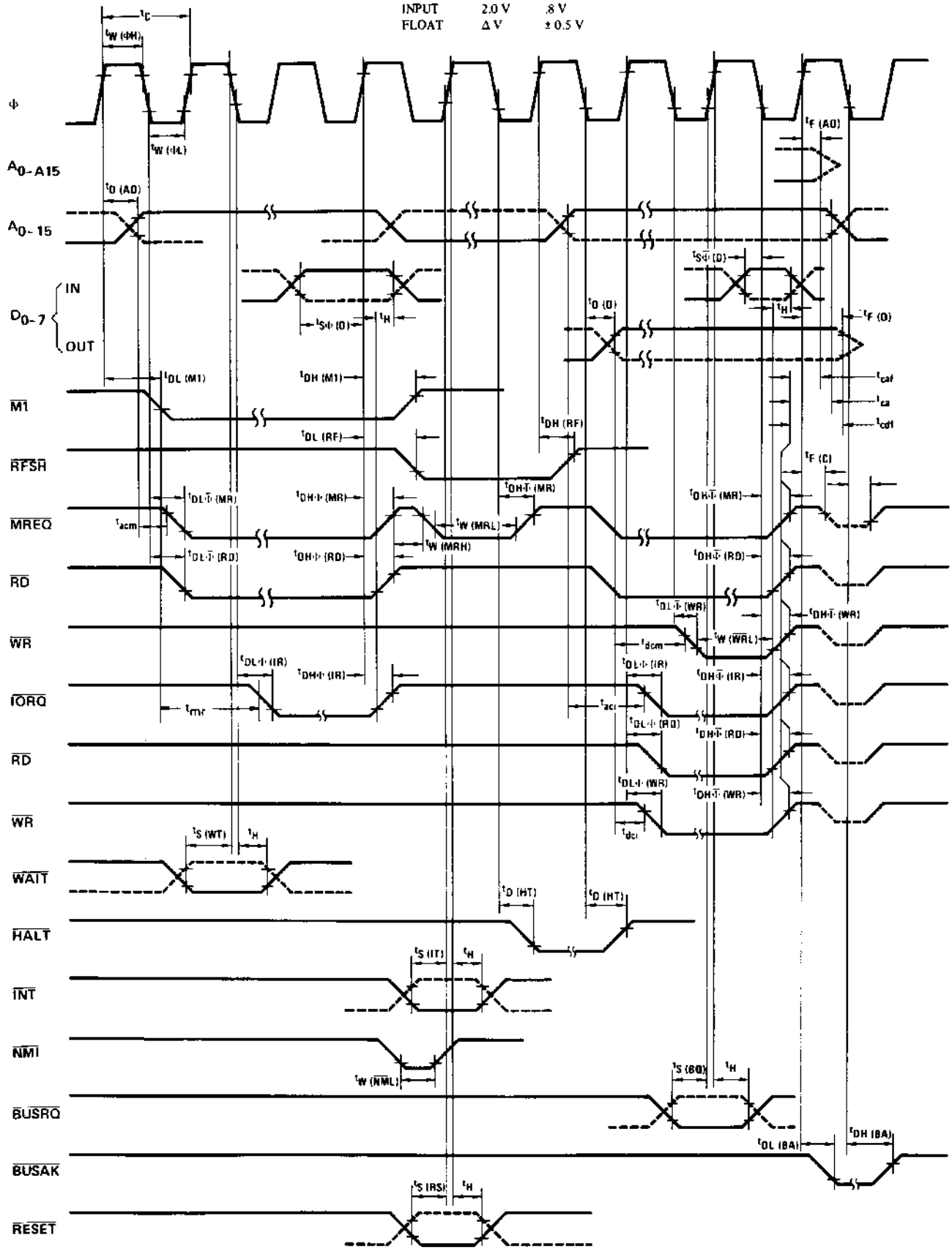
Load circuit for Output

Zilog

A.C. Timing Diagram

Timing measurements are made at the following voltages, unless otherwise specified:

	"1"	"0"
CLOCK	V_{cc} -.6V	.45V
OUTPUT	2.0 V	.8 V
INPUT	2.0 V	.8 V
FLOAT	Δ V	± 0.5 V



Zilog

Z-80A CPU A.C. Characteristics

T_A = 0°C to 70°C, V_{CC} = +5V ± 5%, Unless Otherwise Noted.

Signal	Symbol	Parameter	Min	Max	Unit	Test Condition
Φ	t_c	Clock Period	250	[12]	nsec	
	$t_{w(\Phi H)}$	Clock Pulse Width, Clock High	110	[E]	nsec	
	$t_{w(\Phi L)}$	Clock Pulse Width, Clock Low	110	2000	nsec	
	$t_{r,f}$	Clock Rise and Fall Time		30	nsec	
A_{0-15}	$t_{D(AD)}$	Address Output Delay		110	nsec	C_L = 50pF
	$t_{F(AD)}$	Delay to Float		90	nsec	
	t_{acm}	Address Stable Prior to $\overline{MREQ}$ (Memory Cycle)	[1]		nsec	
	t_{aci}	Address Stable Prior to $\overline{IORQ}$, $\overline{RD}$ or $\overline{WR}$ (I/O Cycle)	[2]		nsec	
	t_{ca}	Address Stable from $\overline{RD}$, $\overline{WR}$, $\overline{IORQ}$ or $\overline{MREQ}$	[3]		nsec	
	t_{caf}	Address Stable From $\overline{RD}$ or $\overline{WR}$ During Float	[4]		nsec	
D_{0-7}	$t_{D(D)}$	Data Output Delay		150	nsec	C_L = 50pF
	$t_{F(D)}$	Delay to Float During Write Cycle		90	nsec	
	$t_{S\Phi(D)}$	Data Setup Time to Rising Edge of Clock During M1 Cycle	35		nsec	
	$t_{S\overline{\Phi}(D)}$	Data Setup Time to Falling Edge of Clock During M2 to M5	50		nsec	
	t_{dcm}	Data Stable Prior to $\overline{WR}$ (Memory Cycle)	[5]		nsec	
	t_{dci}	Data Stable Prior to $\overline{WR}$ (I/O Cycle)	[6]		nsec	
	t_{cdf}	Data Stable From $\overline{WR}$	[7]			
	t_H	Any Hold Time for Setup Time		0	nsec	
$\overline{MREQ}$	$t_{DL\overline{\Phi}(MR)}$	$\overline{MREQ}$ Delay From Falling Edge of Clock, $\overline{MREQ}$ Low		85	nsec	C_L = 50pF
	$t_{DH\Phi(MR)}$	$\overline{MREQ}$ Delay From Rising Edge of Clock, $\overline{MREQ}$ High		85	nsec	
	$t_{DH\overline{\Phi}(MR)}$	$\overline{MREQ}$ Delay From Falling Edge of Clock, $\overline{MREQ}$ High		85	nsec	
	$t_{w(\overline{MRL})}$	Pulse Width, $\overline{MREQ}$ Low	[8]		nsec	
	$t_{w(\overline{MRH})}$	Pulse Width, $\overline{MREQ}$ High	[9]		nsec	
$\overline{IORQ}$	$t_{DL\Phi(IR)}$	$\overline{IORQ}$ Delay From Rising Edge of Clock, $\overline{IORQ}$ Low		75	nsec	C_L = 50pF
	$t_{DL\overline{\Phi}(IR)}$	$\overline{IORQ}$ Delay From Falling Edge of Clock, $\overline{IORQ}$ Low		85	nsec	
	$t_{DH\Phi(IR)}$	$\overline{IORQ}$ Delay From Rising Edge of Clock, $\overline{IORQ}$ High		85	nsec	
	$t_{DH\overline{\Phi}(IR)}$	$\overline{IORQ}$ Delay From Falling Edge of Clock, $\overline{IORQ}$ High		85	nsec	
$\overline{RD}$	$t_{DL\Phi(RD)}$	$\overline{RD}$ Delay From Rising Edge of Clock, $\overline{RD}$ Low		85	nsec	C_L = 50pF
	$t_{DL\overline{\Phi}(RD)}$	$\overline{RD}$ Delay From Falling Edge of Clock, $\overline{RD}$ Low		95	nsec	
	$t_{DH\Phi(RD)}$	$\overline{RD}$ Delay From Rising Edge of Clock, $\overline{RD}$ High		85	nsec	
	$t_{DH\overline{\Phi}(RD)}$	$\overline{RD}$ Delay From Falling Edge of Clock, $\overline{RD}$ High		85	nsec	
$\overline{WR}$	$t_{DL\Phi(WR)}$	$\overline{WR}$ Delay From Rising Edge of Clock, $\overline{WR}$ Low		65	nsec	C_L = 50pF
	$t_{DL\overline{\Phi}(WR)}$	$\overline{WR}$ Delay From Falling Edge of Clock, $\overline{WR}$ Low		80	nsec	
	$t_{DH\overline{\Phi}(WR)}$	$\overline{WR}$ Delay From Falling Edge of Clock, $\overline{WR}$ High		80	nsec	
	$t_{w(\overline{WRL})}$	Pulse Width, $\overline{WR}$ Low	[10]		nsec	
$\overline{M1}$	$t_{DL(M1)}$	$\overline{M1}$ Delay From Rising Edge of Clock, $\overline{M1}$ Low		100	nsec	C_L = 50pF
	$t_{DH(M1)}$	$\overline{M1}$ Delay From Rising Edge of Clock, $\overline{M1}$ High		100	nsec	
$\overline{RFSH}$	$t_{DL(RF)}$	$\overline{RFSH}$ Delay From Rising Edge of Clock, $\overline{RFSH}$ Low		130	nsec	C_L = 50pF
	$t_{DH(RF)}$	$\overline{RFSH}$ Delay From Rising Edge of Clock, $\overline{RFSH}$ High		120	nsec	
$\overline{WAIT}$	$t_{s(WT)}$	$\overline{WAIT}$ Setup Time to Falling Edge of Clock	70		nsec	
$\overline{HALT}$	$t_{D(HT)}$	$\overline{HALT}$ Delay Time From Falling Edge of Clock		300	nsec	C_L = 50pF
$\overline{INT}$	$t_{s(IT)}$	$\overline{INT}$ Setup Time to Rising Edge of Clock	80		nsec	
$\overline{NMI}$	$t_{w(\overline{NML})}$	Pulse Width, $\overline{NMI}$ Low	80		nsec	
$\overline{BUSRQ}$	$t_{s(BQ)}$	$\overline{BUSRQ}$ Setup Time to Rising Edge of Clock	50		nsec	
$\overline{BUSAK}$	$t_{DL(BA)}$	$\overline{BUSAK}$ Delay From Rising Edge of Clock, $\overline{BUSAK}$ Low		100	nsec	C_L = 50pF
	$t_{DH(BA)}$	$\overline{BUSAK}$ Delay From Falling Edge of Clock, $\overline{BUSAK}$ High		100	nsec	
$\overline{RESET}$	$t_{s(RS)}$	$\overline{RESET}$ Setup Time to Rising Edge of Clock	60		nsec	
	$t_{F(C)}$	Delay to Float ($\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$ and $\overline{WR}$)		80	nsec	
	t_{mr}	M1 Stable Prior to IORQ (Interrupt Ack.)	[11]		nsec	

[1] $t_{acm} = t_{w(\Phi H)} + t_f - 65$

[2] $t_{aci} = t_c - 70$

[3] $t_{ca} = t_{w(\Phi L)} + t_r - 50$

[4] $t_{caf} = t_{w(\Phi L)} + t_r - 45$

[5] $t_{dcm} = t_c - 170$

[6] $t_{dci} = t_{w(\Phi L)} + t_r - 170$

[7] $t_{cdf} = t_{w(\Phi L)} + t_r - 70$

[8] $t_{w(\overline{MRL})} = t_c - 30$

[9] $t_{w(\overline{MRH})} = t_{w(\Phi H)} + t_f - 20$

[10] $t_{w(\overline{WRL})} = t_c - 30$

[11] $t_{mr} = 2t_c + t_{w(\Phi H)} + t_f - 65$

[12] $t_c = t_{w(\Phi H)} + t_{w(\Phi L)} + t_r + t_f$

NOTES:

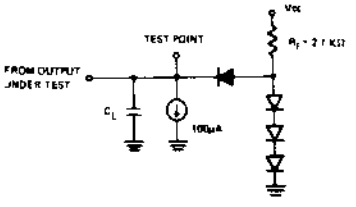
A. Data should be enabled onto the CPU data bus when $\overline{RD}$ is active. During interrupt acknowledge data should be enabled when $\overline{M1}$ and $\overline{IORQ}$ are both active.

B. All control signals are internally synchronized, so they may be totally asynchronous with respect to the clock.

C. The $\overline{RESET}$ signal must be active for a minimum of 3 clock cycles.

D. Output Delay vs. Loaded Capacitance
TA = 70°C Vcc = +5V ±5%
Add 10nsec delay for each 50pf increase in load up to maximum of 200pf for data bus and 100pf for address & control lines.

E. Although static by design, testing guarantees $t_{w(\Phi H)}$ of 200 μsec maximum



Load circuit for Output

Zilog

Absolute Maximum Ratings

Temperature Under Bias	Specified operating range.
Storage Temperature	-65°C to +150°C
Voltage On Any Pin with Respect to Ground	-0.3V to +7V
Power Dissipation	1.5W

*Comment

Stresses above those listed under "Absolute Maximum Rating" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other condition above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Note: For Z80-CPU all AC and DC characteristics remain the same for the military grade parts except I_{cc}.

I_{cc} = 200 mA

Z-80 CPU D.C. Characteristics

T_A = 0°C to 70°C, V_{cc} = 5V ± 5% unless otherwise specified

Symbol	Parameter	Min.	Typ.	Max.	Unit	Test Condition
V_{ILC}	Clock Input Low Voltage	-0.3		0.45	V	
V_{IHC}	Clock Input High Voltage	V_{cc}-.6		V_{cc}+.3	V	
V_{IL}	Input Low Voltage	-0.3		0.8	V	
V_{IH}	Input High Voltage	2.0		V_{cc}	V	
V_{OL}	Output Low Voltage			0.4	V	I_{OL}=1.8mA
V_{OH}	Output High Voltage	2.4			V	I_{OH}=-250μA
I_{CC}	Power Supply Current			150	mA	
I_{LI}	Input Leakage Current			10	μA	V_{IN}=0 to V_{cc}
I_{LOH}	Tri-State Output Leakage Current in Float			10	μA	V_{OUT}=2.4 to V_{cc}
I_{LOL}	Tri-State Output Leakage Current in Float			-10	μA	V_{OUT}=0.4V
I_{LD}	Data Bus Leakage Current in Input Mode			±10	μA	0≤V_{IN}≤V_{cc}

Capacitance

T_A = 25°C, f = 1 MHz,
unmeasured pins returned to ground

Symbol	Parameter	Max.	Unit
C_Φ	Clock Capacitance	35	pF
C_{IN}	Input Capacitance	5	pF
C_{OUT}	Output Capacitance	10	pF

Z-80 CPU Ordering Information

C – Ceramic
P – Plastic
S – Standard 5V ±5% 0° to 70°C
E – Extended 5V ±5% -40° to 85°C
M – Military 5V ±10% -55° to 125°C

Z-80A CPU D.C. Characteristics

T_A =0°C to 70°C, V_{cc} =5V±5% unless otherwise specified

Symbol	Parameter	Min.	Typ.	Max.	Unit	Test Condition
V_{ILC}	Clock Input Low Voltage	-0.3		0.45	V	
V_{IHC}	Clock Input High Voltage	V_{cc}-.6		V_{cc}+.3	V	
V_{IL}	Input Low Voltage	-0.3		0.8	V	
V_{IH}	Input High Voltage	2.0		V_{cc}	V	
V_{OL}	Output Low Voltage			0.4	V	I_{OL}=1.8mA
V_{OH}	Output High Voltage	2.4			V	I_{OH}=-250μA
I_{CC}	Power Supply Current		90	200	mA	
I_{LI}	Input Leakage Current			10	μA	V_{IN}=0 to V_{cc}
I_{LOH}	Tri-State Output Leakage Current in Float			10	μA	V_{OUT}=2.4 to V_{cc}
I_{LOL}	Tri-State Output Leakage Current in Float			-10	μA	V_{OUT}=0.4V
I_{LD}	Data Bus Leakage Current in Input Mode			±10	μA	0≤V_{IN}≤V_{cc}

Capacitance

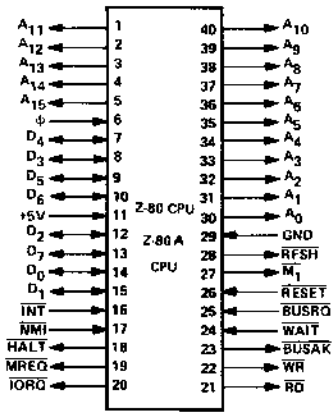
T_A = 25°C, f = 1 MHz,
unmeasured pins returned to ground

Symbol	Parameter	Max.	Unit
C_Φ	Clock Capacitance	35	pF
C_{IN}	Input Capacitance	5	pF
C_{OUT}	Output Capacitance	10	pF

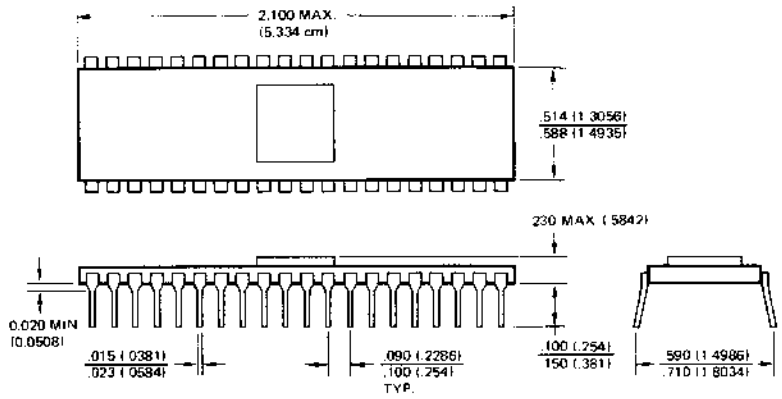
Z-80A CPU Ordering Information

C – Ceramic
P – Plastic
S Standard 5V ±5% 0° to 70°C

Package Configuration



Package Outline



*Dimensions for metric system are in parentheses

Zilog

Product Specification

Z-80® PIO
Z-80A PIO

The Zilog Z-80 product line is a complete set of microcomputer components, development systems and support software. The Z-80 microcomputer component set includes all of the circuits necessary to build high-performance microcomputer systems with virtually no other logic and a minimum number of low cost standard memory elements.

The Z-80 Parallel I/O (PIO) Interface Controller is a programmable, two port device which provides TTL compatible interfacing between peripheral devices and the Z80-CPU. The Z80-CPU configures the Z80-PIO to interface with standard peripheral devices such as tape punches, printers, keyboards, etc.

Structure

- N-Channel Silicon Gate Depletion Load technology
- 40 Pin DIP
- Single 5 volt supply
- Single phase 5 volt clock
- Two independent 8-bit bidirectional peripheral interface ports with "handshake" data transfer control

Features

- Interrupt driven "handshake" for fast response
- Any one of the following modes of operation may be selected for either port:
  - Byte output
  - Byte input
  - Byte bidirectional bus (available on Port A only)
  - Bit Mode
- Programmable interrupts on peripheral status conditions.
- Daisy chain priority interrupt logic included to provide for automatic interrupt vectoring without external logic.
- Eight outputs are capable of driving Darlington transistors.
- All inputs and outputs fully TTL compatible.

PIO Architecture

A block diagram of the Z80-PIO is shown in figure 1. The internal structure of the Z80-PIO consists of a Z80-CPU bus interface, internal control logic, Port A I/O logic, Port B I/O logic, and interrupt control logic. A typical application might use Port A as the data transfer channel and Port B for the status and control monitoring.

The Port I/O logic is composed of 6 registers with "handshake" control logic as shown in figure 2. The registers include: an 8-bit input register, an 8-bit output register, a 2-bit mode control register, an 8-bit mask register, an 8-bit input/output select register, and a 2-bit mask control register. The last three registers are used only when the port has been programmed to operate in the bit mode.

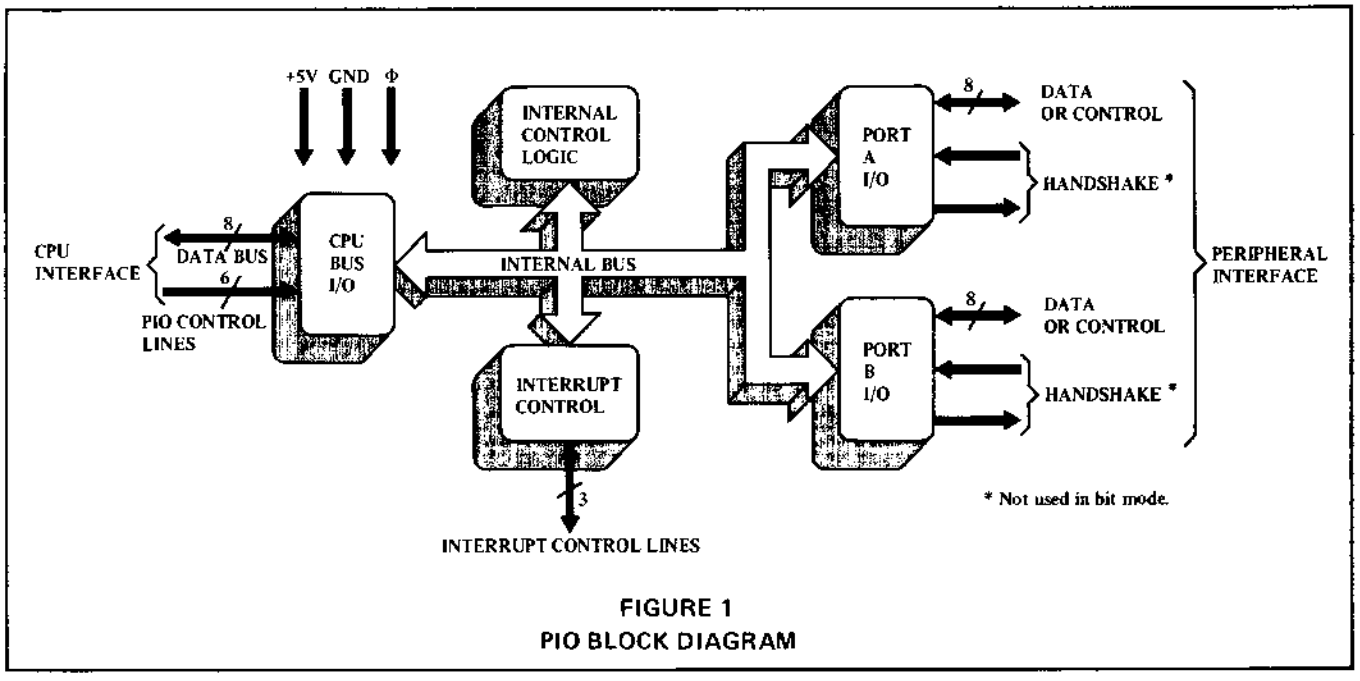


FIGURE 1
PIO BLOCK DIAGRAM

Register Description

Mode Control Register–2 bits, loaded by CPU to select the operating mode: byte output, byte input, byte bidirectional bus or bit mode.

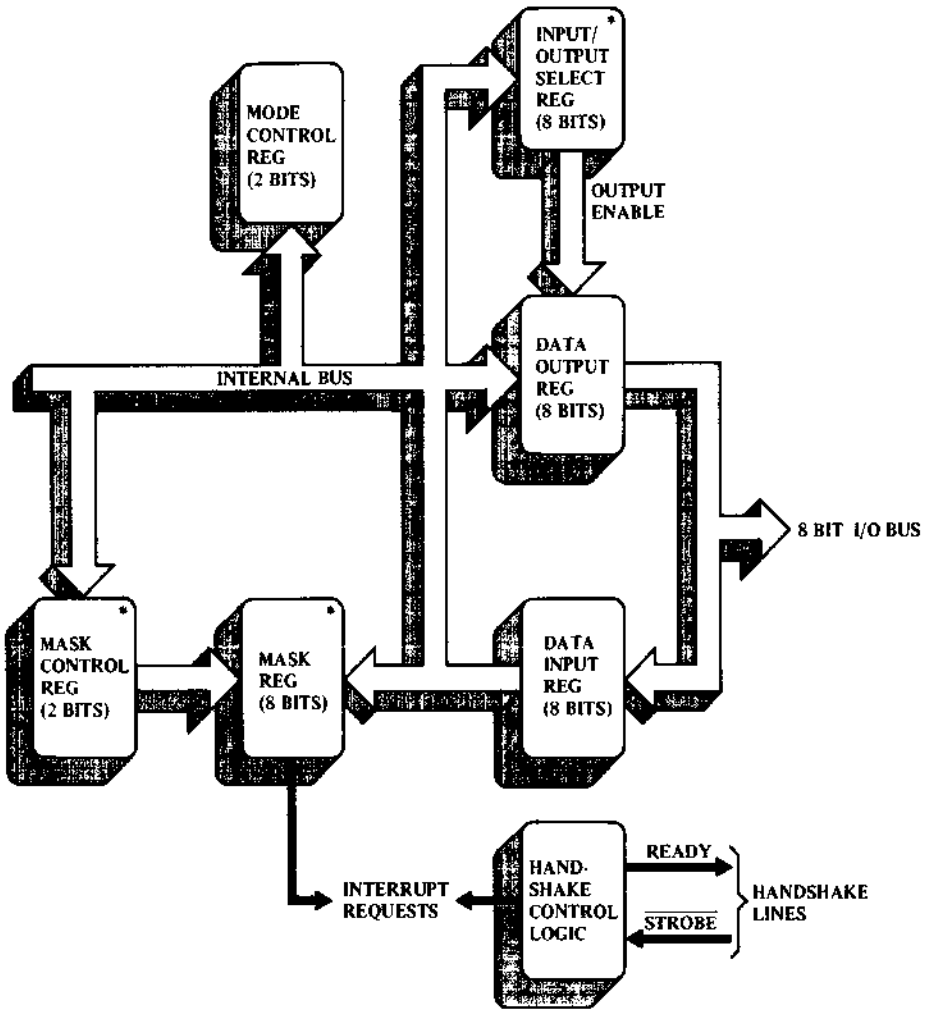
Data Output Register–8 bits, permits data to be transferred from the CPU to the peripheral.

Data Input Register–8 bits, accepts data from the peripheral for transfer to the CPU.

Mask Control Register–2 bits, loaded by the CPU to specify the active state (high or low) of any peripheral device interface pins that are to be monitored and, if an interrupt should be generated when all unmasked pins are active (AND condition) or, when any unmasked pin is active (OR condition).

Mask Register–8 bits, loaded by the CPU to determine which peripheral device interface pins are to be monitored for the specified status condition.

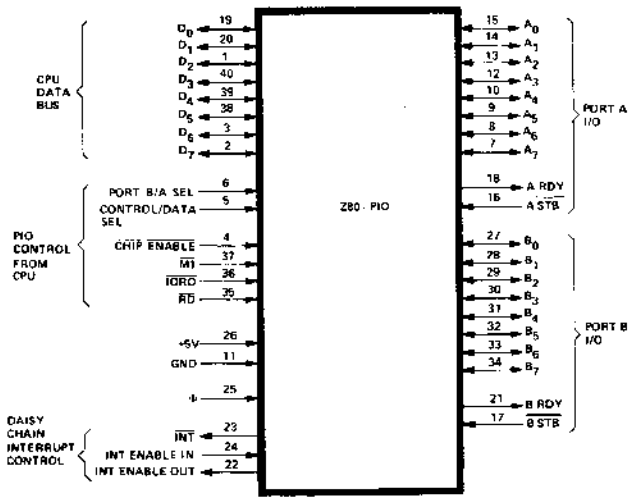
Input/Output Select Register–8 bits, loaded by the CPU to allow any pin to be an output or an input during bit mode operation.



* Used in the bit mode only to allow generation of an interrupt if the peripheral I/O pins go to the specified state.

FIGURE 2
A TYPICAL PORT I/O BLOCK DIAGRAM

Z-80 PIO Pin Description

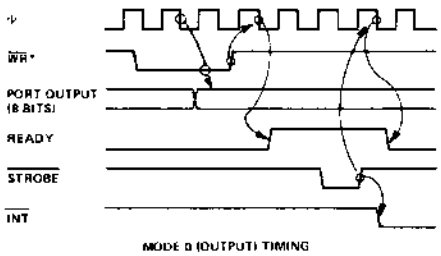


D_7-D_0	Z80-CPU Data Bus (bidirectional, tristate)
B/A Sel	Port B or A Select (input, active high)
C/D Sel	Control or Data Select (input, active high)
$\overline{CE}$	Chip Enable (input, active low)
Φ	System Clock (input)
$\overline{M1}$	Machine Cycle One Signal from CPU (input, active low)
$\overline{IORQ}$	Input/Output Request from Z80-CPU (input, active low)
$\overline{RD}$	Read Cycle Status from the Z80-CPU (input, active low)
IEI	Interrupt Enable In (input, active high)
IEO	Interrupt Enable Out (output, active high). IEI and IEO form a daisy chain connection for priority interrupt control.
$\overline{INT}$	Interrupt Request (output, open drain, active low)
A_0-A_7	Port A Bus (bidirectional, tristate)
$\overline{A\ STB}$	Port A Strobe Pulse from Peripheral Device (input, active low)
A RDY	Register A Ready (output, active high)
B_0-B_7	Port B Bus (bidirectional, tristate)
$\overline{B\ STB}$	Port B Strobe Pulse from Peripheral Device (input, active low)
B RDY	Register B Ready (output, active high)

Timing Waveforms

OUTPUT MODE

An output cycle is always started by the execution of an output instruction by the CPU. The $\overline{WR}$ pulse from the CPU latches the data from the CPU data bus into the selected port's output register. The write pulse sets the ready flag after a low going edge of Φ, indicating data is available. Ready stays active until the positive edge of the strobe line is received indicating that data was taken by the peripheral. The positive edge of the strobe pulse generates an $\overline{INT}$ if the interrupt enable flip flop has been set and if this device has the highest priority.

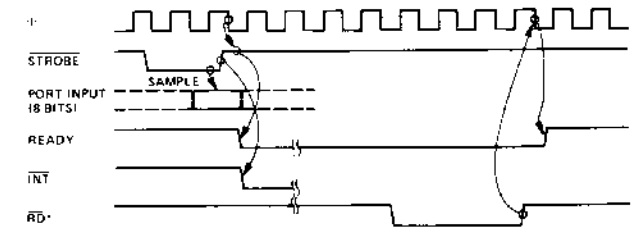


MODE 0 (OUTPUT) TIMING

$WR^* = \overline{RD} \cdot CE \cdot \overline{C/D} \cdot IORQ$

INPUT MODE

When $\overline{STROBE}$ goes low data is loaded into the selected port input register. The next rising edge of strobe activates $\overline{INT}$ if interrupt enable is set and this is the highest priority requesting device. The following falling edge of Φ resets Ready to an inactive state, indicating that the input register is full and cannot accept any more data until the CPU completes a read. When a read is complete the positive edge of $\overline{RD}$ will set Ready at the next low going transition of Φ. At this time new data can be loaded into the PIO.



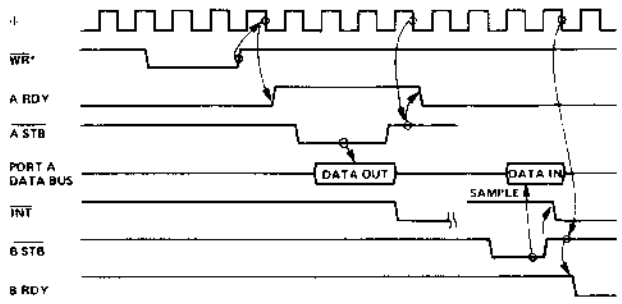
$RD^* = RD \cdot CE \cdot \overline{C/D} \cdot IORQ$ MODE 1 (INPUT) TIMING

Zilog

Timing Waveforms

BIDIRECTIONAL MODE

This is a combination of modes 0 and 1 using all four handshake lines and the 8 Port A I/O lines. Port B must be set to the Bit Mode. The Port A handshake lines are used for output control and the Port B lines are used for input control. Data is allowed out onto the Port A bus only when $\overline{A\ STB}$ is low. The rising edge of this strobe can be used to latch the data into the peripheral.

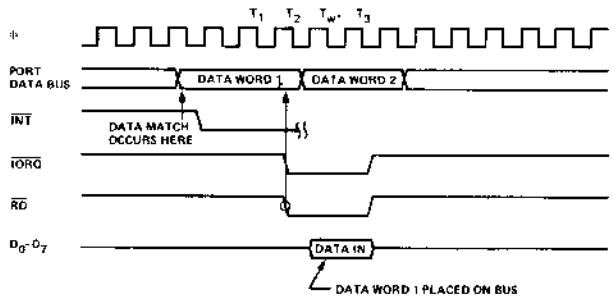


$WR^* = \overline{RD} \cdot CE \cdot \overline{C/D} \cdot IORQ$

BIT MODE

The bit mode does not utilize the handshake signals and a normal port write or port read can be executed at any time. When writing, the data will be latched into the output registers with the same timing as the output mode.

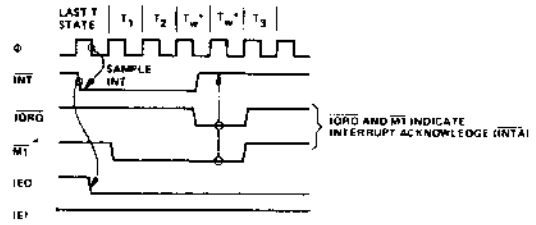
When reading the PIO, the data returned to the CPU will be composed of output register data from those port data lines assigned as outputs and input register data from those port data lines assigned as inputs. The input register will contain data which was present immediately prior to the falling edge of $\overline{RD}$. An interrupt will be generated if interrupts from the port are enabled and the data on the port data lines satisfy the logical equation defined by the 8-bit mask and 2-bit mask control registers.



* Timing Diagram Refers to Bit Mode Read.

INTERRUPT ACKNOWLEDGE

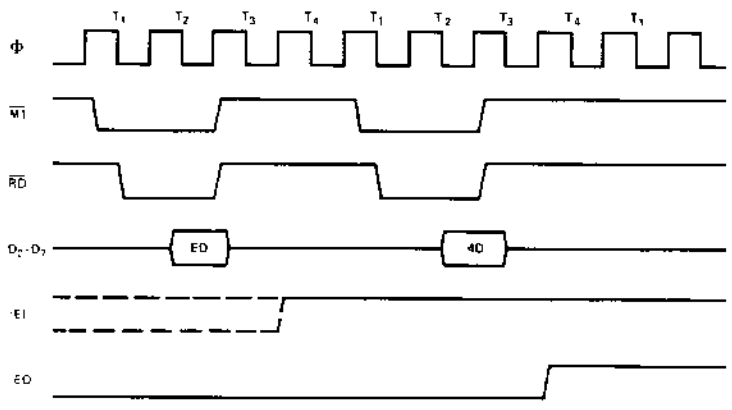
During $\overline{M1}$ time, peripheral controllers are inhibited from changing their interrupt enable status, permitting the $\overline{INT}$ Enable signal to ripple through the daisy chain. The peripheral with IEI high and IEO low during $\overline{INTA}$ will place a preprogrammed 8-bit interrupt vector on the data bus at this time. IEO is held low until a return from interrupt (RETI) instruction is executed by the CPU while IEI is high. The 2-byte RETI instruction is decoded internally by the PIO for this purpose.



RETURN FROM INTERRUPT CYCLE

If a Z80 peripheral device has no interrupt pending and is not under service, then its IEO=IEI. If it has an interrupt under service (i.e., it has already interrupted and received an interrupt acknowledge) then its IEO is always low, inhibiting lower priority chips from interrupting. If it has an interrupt pending which has not yet been acknowledged, IEO will be low unless an "ED" is decoded as the first byte of a two byte opcode. In this case, IEO will go high until the next opcode byte is decoded, whereupon it will again go low. If the second byte of the opcode was a "4D" then the opcode was an RETI instruction.

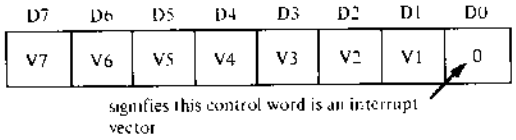
After an "ED" opcode is decoded, only the peripheral device which has interrupted and is currently under service will have its IEI high and its IEO low. This device is the highest priority device in the daisy chain which has received an interrupt acknowledge. All other peripherals have IEI=IEO. If the next opcode byte decoded is "4D", this peripheral device will reset its "interrupt under service" condition.



PIO Programming

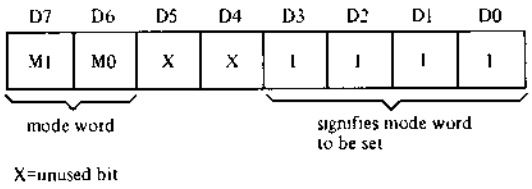
LOAD INTERRUPT VECTOR

The Z80-CPU requires an 8-bit interrupt vector be supplied by the interrupting device. The CPU forms the address for the interrupt service routine of the port using this vector. During an interrupt acknowledge cycle the vector is placed on the Z-80 data bus by the highest priority device requesting service at that time. The desired interrupt vector is loaded into the PIO by writing a control word to the desired port of the PIO with the following format.



SELECTING AN OPERATING MODE

When selecting an operating mode, the 2-bit mode control register is set to one of four values. These two bits are the most significant bits of the register, bits 7 and 6; bits 5 and 4 are not used while bits 3 through 0 are all set to 1111 to indicate "set mode."



Mode	M_1	M_0
Output	0	0
Input	0	1
Bidirectional	1	0
Bit	1	1

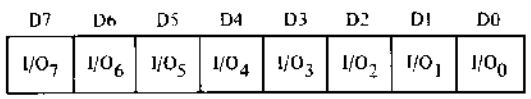
MODE 0 active indicates that data is to be written from the CPU to the peripheral.

MODE 1 active indicates that data is to be read from the peripheral to the CPU.

MODE 2 allows data to be written to or read from the peripheral device.

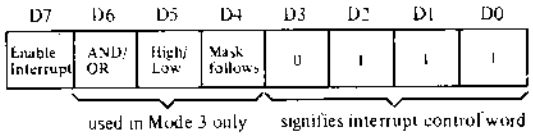
MODE 3 is intended for status and control applications. When selected, the next control word must set the I/O Register to indicate which lines are to be input and which lines are to be output.

I/O = 1 sets bit to input.
I/O = 0 sets bit to output.

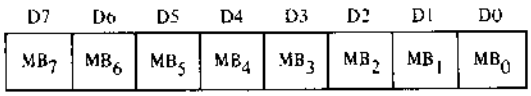


INTERRUPT CONTROL

Bit 7 = 1	interrupt enable is set--allowing interrupt to be generated.
Bit 7 = 0	indicates the enable flag is reset and interrupts may not be generated.
Bits 6,5,4	are used in the bit mode interrupt operations; otherwise they are disregarded.
Bits 3,2,1,0	signify that this command word is an interrupt control word.

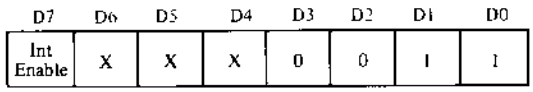


If the "mask follows" bit is high (D4 = 1), the next control word written to the port must be the mask.



Only those port lines whose mask bit is a 0 will be monitored for generating an interrupt.

The interrupt enable flip-flop of a port may be set or reset without modifying the rest of the interrupt control word by the following command.



Z-80 PIO A.C. Characteristics

TA = 0° C to 70° C, Vcc = +5 V ± 5%, unless otherwise noted

SIGNAL	SYMBOL	PARAMETER	MIN	MAX	UNIT	COMMENTS
Φ	t_c	Clock Period	400	[1]	nsec	
	$t_{W(\Phi H)}$	Clock Pulse Width, Clock High	170	2000	nsec	
	$t_{W(\Phi L)}$	Clock Pulse Width, Clock Low	170	2000	nsec	
	t_r, t_f	Clock Rise and Fall Times		30	nsec	
	t_h	Any Hold Time for Specified Set-Up Time	0		nsec	
CS, $\overline{CE}$ ETC.	$t_{S\Phi(CS)}$	Control Signal Set-Up Time to Rising Edge of Φ During Read or Write Cycle	280		nsec	
D_0-D_7	$t_{DR(D)}$	Data Output Delay from Falling Edge of $\overline{RD}$		430	nsec	[2]
	$t_{S\Phi(D)}$	Data Set-Up Time to Rising Edge of ΦDuring Write or $\overline{M1}$ Cycle	50		nsec	C_L = 50 pf
	$t_{DI(D)}$	Data Output Delay from Falling Edge of $\overline{IORQ}$ During INTA Cycle.		340	nsec	[3]
	$t_{F(D)}$	Delay to Floating Bus (Output Buffer Disable Time)		160	nsec	
IEI	$t_{S(IEI)}$	IEI Set-Up Time to Falling Edge of $\overline{IORQ}$ During INTA Cycle	140		nsec	
IEO	$t_{DH(IO)}$	IEO Delay Time from Rising Edge of IEI		210	nsec	[5]
	$t_{DL(IO)}$	IEO Delay Time from Falling Edge of IEI		190	nsec	[5] C_L = 50 pf
	$t_{DM(IO)}$	IEO Delay from Falling Edge of $\overline{M1}$ (Interrupt Occurring Just Prior to $\overline{M1}$) See Note A.		300	nsec	[5]
$\overline{IORQ}$	$t_{S\Phi(IR)}$	$\overline{IORQ}$ Set-Up Time to Rising Edge of Φ During Read or Write Cycle	250		nsec	
$\overline{M1}$	$t_{S\Phi(M1)}$	$\overline{M1}$ Set-Up Time to Rising Edge of Φ During INTA or $\overline{M1}$ Cycle. See Note B.	210		nsec	
$\overline{RD}$	$t_{S\Phi(RD)}$	$\overline{RD}$ Set-Up Time to Rising Edge of Φ During Read or $\overline{M1}$ Cycle	240		nsec	
A_0-A_7, B_0-B_7	$t_{S(PD)}$	Port Data Set-Up Time to Rising Edge of $\overline{STROBE}$ (Mode 1)	260		nsec	
	$t_{DS(PD)}$	Port Data Output Delay from Falling Edge of $\overline{STROBE}$ (Mode 2)		230	nsec	[5]
	$t_{F(PD)}$	Delay to Floating Port Data Bus from Rising Edge of $\overline{STROBE}$ (Mode 2)		200	nsec	C_L = 50 pf
	$t_{DI(PD)}$	Port Data Stable from Rising Edge of $\overline{IORQ}$ During WR Cycle (Mode 0)		200	nsec	[5]
$\overline{ASTB}$, $\overline{BSTB}$	$t_{W(ST)}$	Pulse Width, $\overline{STROBE}$	150		nsec	
			[4]		nsec	
$\overline{INT}$	$t_{D(IT)}$	$\overline{INT}$ Delay Time from Rising Edge of $\overline{STROBE}$		490	nsec	
	$t_{D(IT3)}$	$\overline{INT}$ Delay Time from Data Match During Mode 3 Operation		420	nsec	
ARDY, BRDY	$t_{DH(RY)}$	Ready Response Time from Rising Edge of $\overline{IORQ}$		t_c+ 460	nsec	[5] C_L = 50 pf
	$t_{DL(RY)}$	Ready Response Time from Rising Edge of $\overline{STROBE}$		t_c+ 400	nsec	[5]

A. $2.5\ t_c > (N\ 2)\ t_{DL(IO)} + t_{DM(IO)} - t_{S(IEI)}$ + TTL Buffer Delay, if any

B. $\overline{M1}$ must be active for a minimum of 2 clock periods to reset the PIO.

[1] $t_c = t_{W(\Phi H)} + t_{W(\Phi L)} + t_r + t_f$

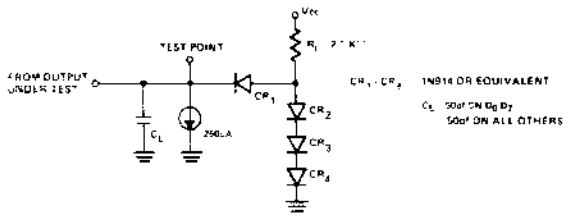
[2] Increase $t_{DR(D)}$ by 10 nsec for each 50 pf increase in loading up to 200 pf max.

[3] Increase $t_{DI(D)}$ by 10 nsec for each 50 pf increase in loading up to 200 pf max.

[4] For Mode 2: $t_{W(ST)} > t_{S(PD)}$

[5] Increase these values by 2 nsec for each 10 pf increase in loading up to 100 pf max

Output load circuit.



Capacitance

TA = 25° C, f = 1 MHz

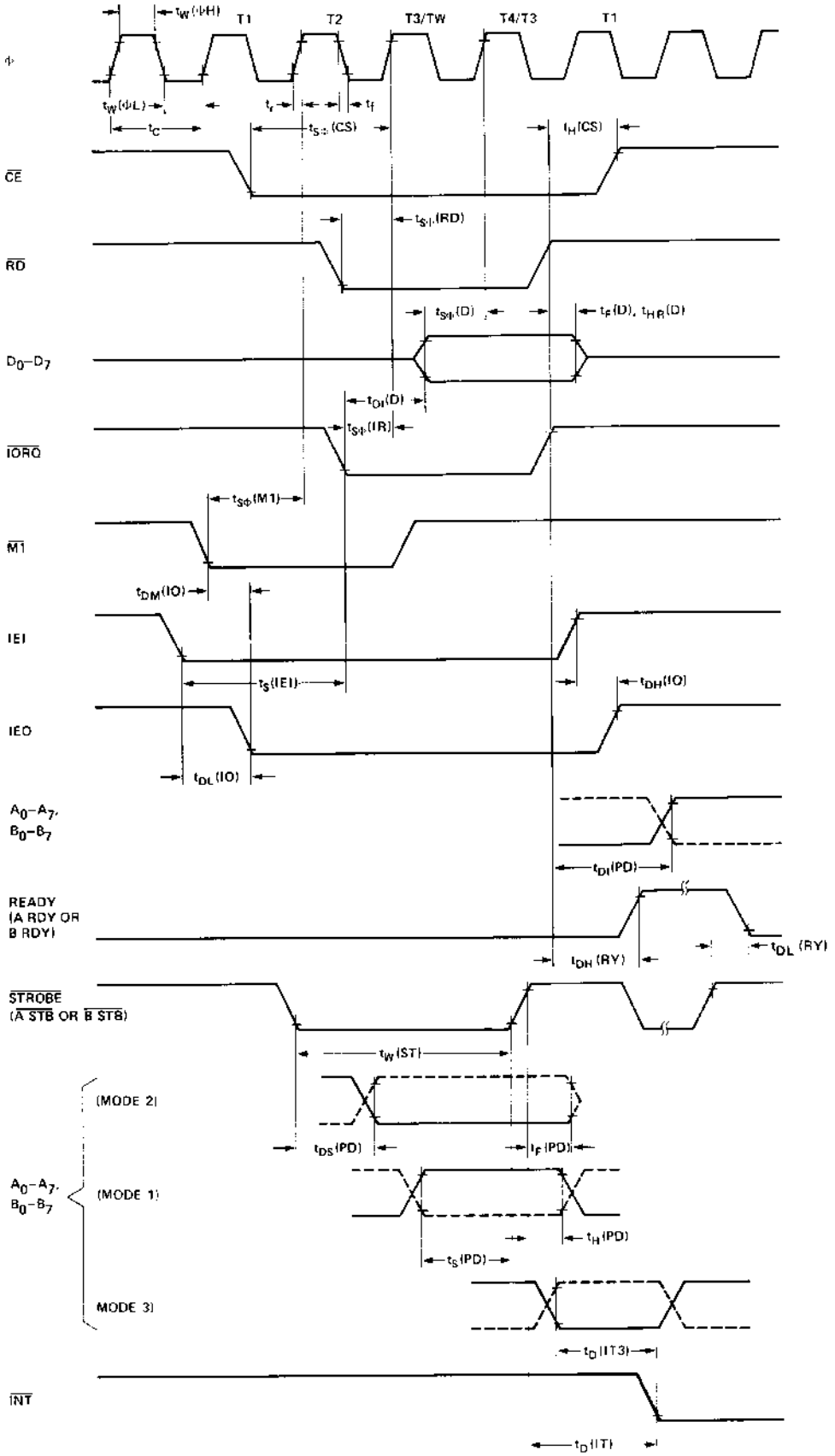
Symbol	Parameter	Max.	Unit	Test Condition
C_Φ	Clock Capacitance	10	pF	Unmeasured Pins
C_{IN}	Input Capacitance	5	pF	Returned to Ground
C_{OUT}	Output Capacitance	10	pF	

Zilog

A.C. Timing Diagram

Timing measurements are made at the following voltages, unless otherwise specified:

	"1"	"0"
CLOCK	4.2V	0.8V
OUTPUT	2.0V	0.8V
INPUT	2.0V	0.8V
FLOAT	ΔV = +0.5V	



Zilog

Z-80A PIO A.C. Characteristics

TA = 0° C to 70° C, Vcc = +5 V ± 5%, unless otherwise noted

SIGNAL	SYMBOL	PARAMETER	MIN	MAX	UNIT	COMMENTS
Φ	t_c	Clock Period	250	[1]	nsec	
	$t_{W(\Phi H)}$	Clock Pulse Width, Clock High	105	2000	nsec	
	$t_{W(\Phi L)}$	Clock Pulse Width, Clock Low	105	2000	nsec	
	t_r, t_f	Clock Rise and Fall Times		30	nsec	
	t_h	Any Hold Time for Specified Set-Up Time	0		nsec	
CS, $\overline{CE}$ ETC.	$t_{S\Phi(CS)}$	Control Signal Set-Up Time to Rising Edge of Φ During Read or Write Cycle	145		nsec	
D_0-D_7	$t_{DR(D)}$	Data Output Delay From Falling Edge of $\overline{RD}$		380	nsec	[2]
	$t_{S\Phi(D)}$	Data Set-Up Time to Rising Edge of Φ During Write or $\overline{M1}$ Cycle	50		nsec	C_L = 50 pf
	$t_{DI(D)}$	Data Output Delay from Falling Edge of $\overline{IORQ}$ During INTA Cycle		250	nsec	[3]
	$t_{F(D)}$	Delay to Floating Bus (Output Buffer Disable Time)		110	nsec	
IEI	$t_{S(IEI)}$	IEI Set-Up Time to Falling edge of $\overline{IORQ}$ During INTA Cycle	140		nsec	
IEO	$t_{DH(IO)}$	IEO Delay Time from Rising Edge of IEI		160	nsec	[5]
	$t_{DL(IO)}$	IEO Delay Time from Falling Edge of IEI		130	nsec	[5] C_L = 50 pf
	$t_{DM(IO)}$	IEO Delay from Falling Edge of $\overline{M1}$ (Interrupt Occurring Just Prior to $\overline{M1}$) See Note A.		190	nsec	[5]
$\overline{IORQ}$	$t_{S\Phi(IR)}$	$\overline{IORQ}$ Set-Up Time to Rising Edge of Φ During Read or Write Cycle.	115		nsec	
$\overline{M1}$	$t_{S\Phi(M1)}$	$\overline{M1}$ Set-Up Time to Rising Edge of Φ During INTA or $\overline{M1}$ Cycle See Note B	90		nsec	
$\overline{RD}$	$t_{S\Phi(RD)}$	$\overline{RD}$ Set-Up Time to Rising Edge of Φ During Read or $\overline{M1}$ Cycle	115		nsec	
A_0-A_7, B_0-B_7	$t_{S(PD)}$	Port Data Set-Up Time to Rising Edge of $\overline{STROBE}$ (Mode 1)	230		nsec	
	$t_{DS(PD)}$	Port Data Output Delay from Falling Edge of $\overline{STROBE}$ (Mode 2)		210	nsec	[5]
	$t_{F(PD)}$	Delay to Floating Port Data Bus from Rising Edge of $\overline{STROBE}$ (Mode 2)		180	nsec	C_L = 50 pf
	$t_{DI(PD)}$	Port Data Stable from Rising Edge of $\overline{IORQ}$ During WR Cycle (Mode 0)		180	nsec	[5]
$\overline{ASTB}$, $\overline{BSTB}$	$t_{W(ST)}$	Pulse Width, $\overline{STROBE}$	150 [4]		nsec nsec	
$\overline{INT}$	$t_{D(IT)}$	$\overline{INT}$ Delay time from Rising Edge of $\overline{STROBE}$		440	nsec	
	$t_{D(IT3)}$	$\overline{INT}$ Delay Time from Data Match During Mode 3 Operation		380	nsec	
ARDY, BRDY	$t_{DH(RY)}$	Ready Response Time from Rising Edge of $\overline{IORQ}$		t_c+ 410	nsec	[5] C_L = 50 pf
	$t_{DL(RY)}$	Ready Response Time from Rising Edge of $\overline{STROBE}$		t_c+ 360	nsec	[5]

A. $2.5\,t_c > (N-2)\,t_{DL(IO)} + t_{DM(IO)} + t_{S(IEI)}$ + TTL Buffer Delay, if any

B. $\overline{M1}$ must be active for a minimum of 2 clock periods to reset the PIO.

[1] $t_c = t_{W(\Phi H)} + t_{W(\Phi L)} + t_r + t_f$

[2] Increase $t_{DR(D)}$ by 10 nsec for each 50 pf increase in loading up to 200 pf max

[3] Increase $t_{DI(D)}$ by 10 nsec for each 50 pf increase in loading up to 200 pf max.

[4] For Mode 2: $t_{W(ST)} > t_{S(PD)}$

[5] Increase these values by 2 nsec for each 10 pf increase in loading up to 100 pf max.

Absolute Maximum Ratings

Temperature Under Bias	Specified operating range.
Storage Temperature	-65° C to +150° C
Voltage On Any Pin With Respect To Ground	-0.3 V to +7 V
Power Dissipation	6 W

*Comment

Stresses above those listed under "Absolute Maximum Rating" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other condition above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Note: All AC and DC characteristics remain the same for the military grade parts except I_{CC}.

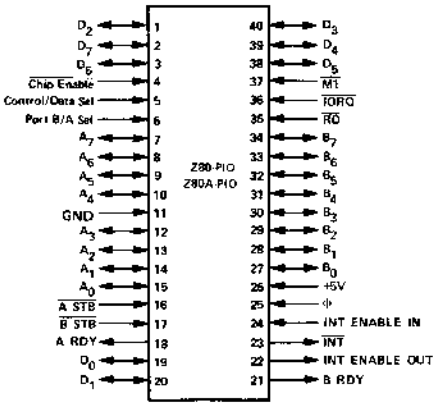
I_{CC} = 130 mA.

Z-80 PIO and Z-80A PIO D.C. Characteristics

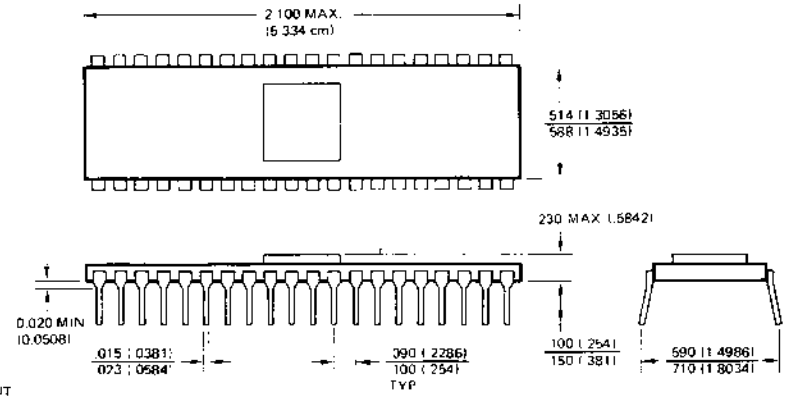
TA = 0° C to 70° C, Vcc = 5 V ± 5% unless otherwise specified

Symbol	Parameter	Min.	Max.	Unit	Test Condition
V_{ILC}	Clock Input Low Voltage	-0.3	.45	V	
V_{IHC}	Clock Input High Voltage	Vcc-.6	Vcc+.3	V	
V_{IL}	Input Low Voltage	-0.3	0.8	V	
V_{IH}	Input High Voltage	2.0	Vcc	V	
V_{OL}	Output Low Voltage		0.4	V	I_{OL} = 2.0 mA
V_{OH}	Output High Voltage	2.4		V	I_{OH} - 250 μA
I_{CC}	Power Supply Current		70	mA	
I_{LI}	Input Leakage Current		10	μA	V_{IN} = 0 to Vcc
I_{LOH}	Tri-State Output Leakage Current in Float		10	μA	V_{OUT} = 2.4 to Vcc
I_{LOL}	Tri-State Output Leakage Current in Float		-10	μA	V_{OUT} = 0.4 V
I_{LD}	Data Bus Leakage Current in Input Mode		+10	μA	$0 \leqslant V_{IN} \leqslant$ Vcc
I_{OHD}	Darlington Drive Current	-1.5		mA	V_{OH} = 1.5 V Port B Only

Package Configuration



Package Outline



*Dimensions for metric system are in parentheses

Ordering Information

C — Ceramic
P — Plastic
S — Standard 5V ± 5% 0° to 70° C
E — Extended 5V ± 5% —40° to 85°C
M — Military 5V ± 10% —55° to 125°C

Example:

Z80-PIO CS (Ceramic—Standard range)

Zilog

Product Specification

Z-80® CTC
Z-80®A CTC

The Zilog Z80 product line is a complete set of microcomputer components, development systems and support software. The Z80 microcomputer component set includes all of the circuits necessary to build high-performance microcomputer systems with virtually no other logic and a minimum number of low cost standard memory elements.

The Z80-Counter Timer Circuit (CTC) is a programmable, four channel device that provides counting and timing functions for the Z80-CPU. The Z80-CPU configures the Z80-CTC's four independent channels to operate under various modes and conditions as required.

Structure

- N-Channel Silicon Gate Depletion Load Technology
- 28 Pin DIP
- Single 5 volt supply
- Single phase 5 volt clock
- Four independent programmable 8-bit counter/16-bit timer channels

Features

- Each channel may be selected to operate in either a counter mode or timer mode.
- Programmable interrupts on counter or timer states.
- A time constant register automatically reloads the down counter at zero and the cycle is repeated.
- Readable down counter indicates number of counts-to-go until zero.
- Selectable 16 or 256 clock prescaler for each timer channel.
- Selectable positive or negative trigger may initiate timer operation.
- Three channels have zero count/timeout outputs capable of driving Darlington transistors.
- Daisy chain priority interrupt logic included to provide for automatic interrupt vectoring without external logic.
- All inputs and outputs fully TTL compatible.
- Outputs directly compatible with Z80-SIO.

CTC Architecture

A block diagram of the Z80-CTC is shown in figure 1. The internal structure of the Z80-CTC consists of a Z80-CPU bus interface, internal control logic, four counter channels, and interrupt control logic. Each channel has an interrupt vector for automatic interrupt vectoring, and interrupt priority is determined by channel number with channel Ø having the highest priority.

The channel logic is composed of 2 registers, 2 counters and control logic as shown in figure 2. The registers include an 8-bit time constant register and an 8-bit channel control register. The counters include an 8-bit readable down counter and an 8-bit prescaler. The prescaler may be programmed to divide the system clock by either 16 or 256.

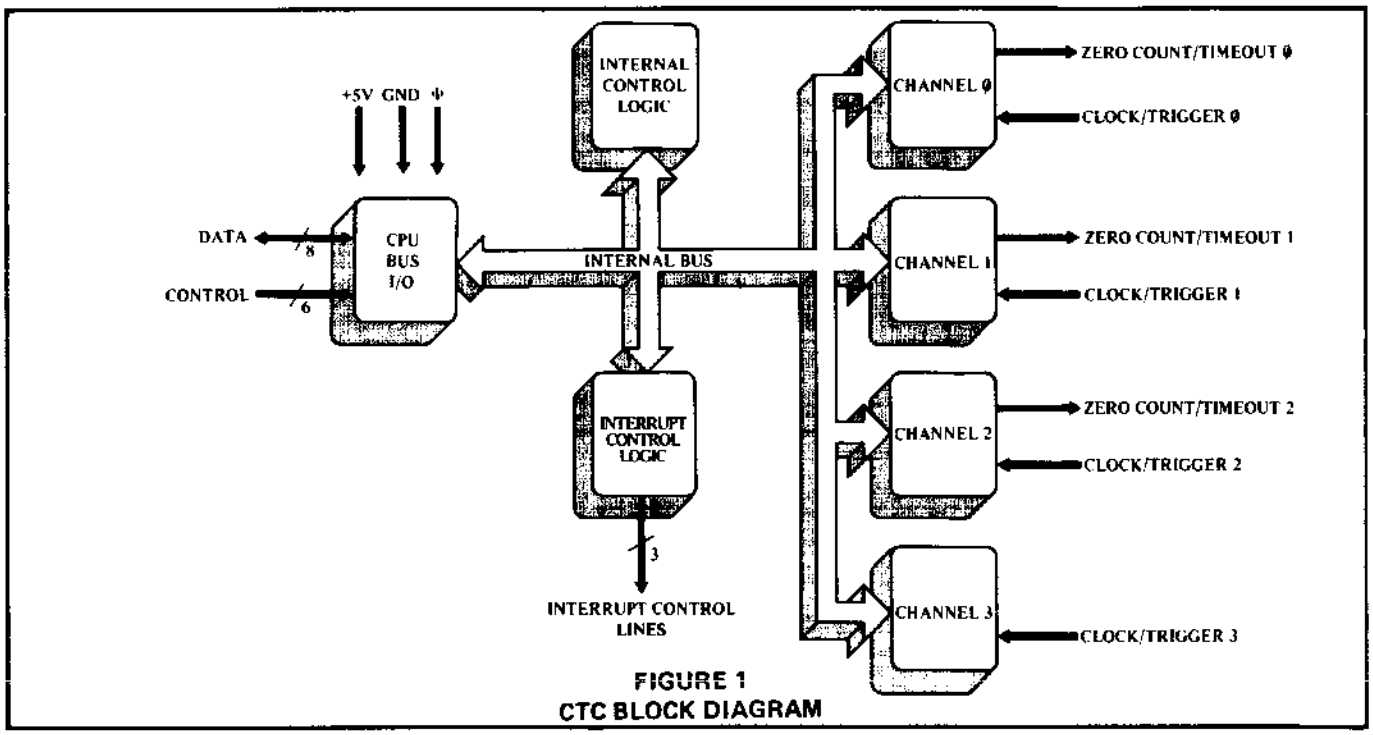


FIGURE 1
CTC BLOCK DIAGRAM

Channel Counter and Register Description

Time Constant Register – 8 bits, loaded by the CPU to initialize and re-load Down Counter at a count of zero.

Channel Control Register – 8 bits, loaded by the CPU to select the mode and conditions of channel operation.

Down Counter – 8 bits, loaded by the Time Constant Register under program control and automatically at a count of zero. At any time, the CPU can read the number of counts-to-go until a zero count. This counter is decremented by the prescaler in timer mode and CLK/TRIG in counter mode.

Prescaler – 8 bit counter, divides system clock by 16 or 256 for decrementing Down Counter. It is used in timer mode only.

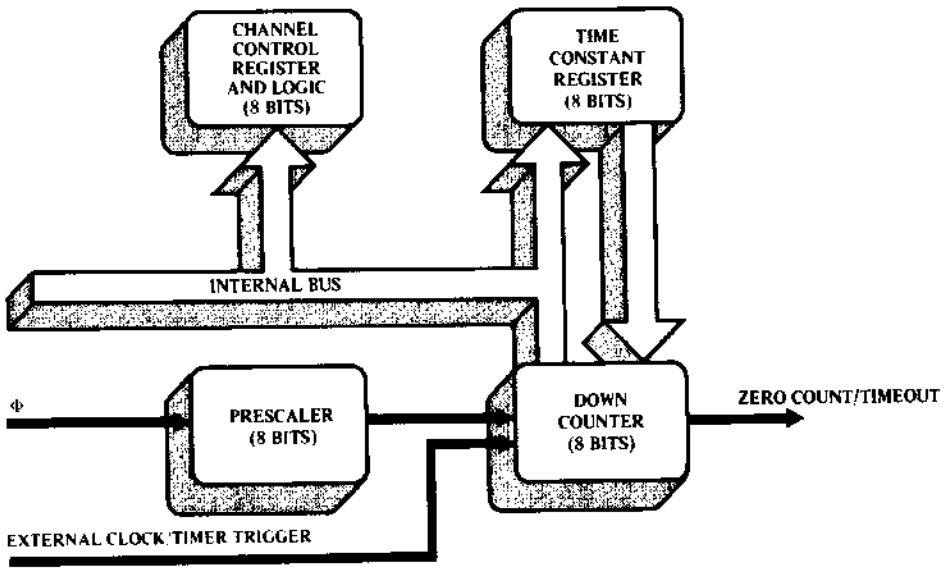
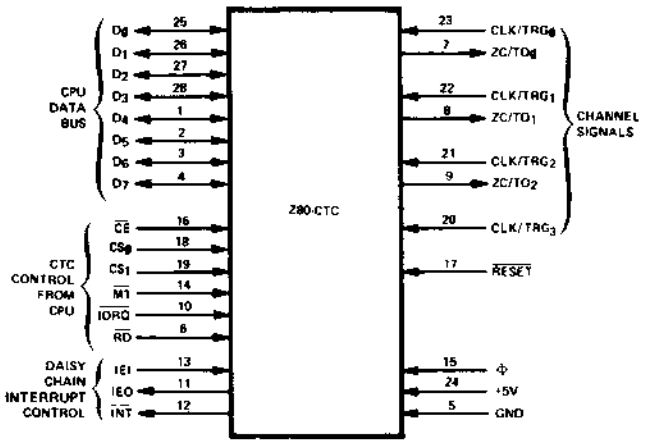


FIGURE 2
CHANNEL BLOCK DIAGRAM

Z-80 CTC Pin Description



$CLK/TRG_{\emptyset}$	Channel Ø External Clock or Timer Trigger (Input)
CLK/TRG_1	Channel 1 External Clock or Timer Trigger (Input)
CLK/TRG_2	Channel 2 External Clock or Timer Trigger (Input)
CLK/TRG_3	Channel 3 External Clock or Timer Trigger (Input)
$ZC/TO_{\emptyset}$	Channel Ø Zero Count or Timeout (output, active high)

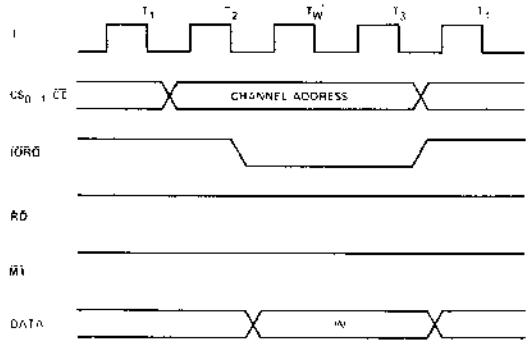
Z-80 CTC Pin Description (continued)

ZC/TO_1	Channel 1 Zero Count or Timeout (output, active high)
ZC/TO_2	Channel 2 Zero Count or Timeout (output, active high)
CS_1 - $CS_\emptyset$	Channel Select (input, active high). These form a 2-bit binary address of the channel to be accessed.
$D7 - D\emptyset$	Z80-CPU Data Bus (bidirectional, tristate)
$\overline{CE}$	Chip Enable (input, active low)
Φ	System Clock (input)
$\overline{M1}$	Machine Cycle One Signal from Z80-CPU (input, active low)
$\overline{IORQ}$	Input/Output Request from Z80-CPU (input, active low)
$\overline{RD}$	Read Cycle Status from the Z80-CPU (input, active low)
IEI	Interrupt Enable In (input, active high)
IEO	Interrupt Enable Out (output, active high). IEI and IEO form a daisy chain connection for priority interrupt control
$\overline{INT}$	Interrupt Request (output, open drain, active low)
$\overline{RESET}$	RESET stops all channels from counting and resets channel interrupt enable bits in all control registers. During reset time $ZC/TO\emptyset$-2 and $\overline{INT}$ go to the inactive states, IEO reflects the state of IEI, and the data bus output drivers go to the high impedance state (input, active low)

Timing Waveforms

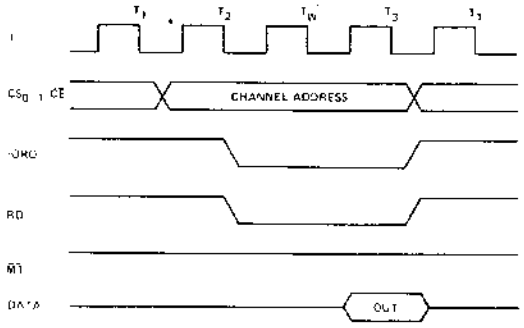
CTC WRITE CYCLE

Illustrated here is the timing for loading a channel control word, time constant and interrupt vector. No wait states are allowed for writing to the CTC other than the automatically inserted (T_W*). Since the CTC does not receive a specific write signal, it internally generates its own from the lack of an $\overline{RD}$ signal.



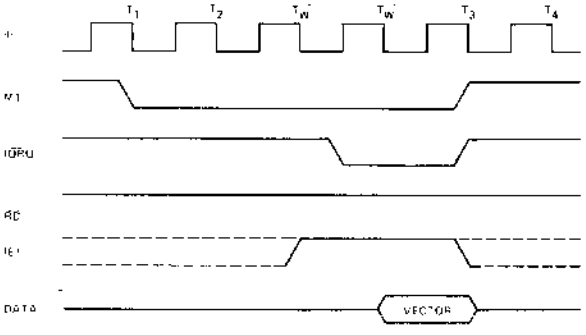
CTC READ CYCLE

Illustrated here is the timing for reading a channel's Down Counter when in Counter Mode. The value read onto the data bus reflects the number of external clock's rising edges prior to the rising edge of cycle (T_2). No wait states are allowed for reading the CTC other than the automatically inserted (T_W*).



INTERRUPT ACKNOWLEDGE CYCLE

Some time after an interrupt is requested by the CTC, the CPU will send out an interrupt acknowledge ($\overline{M1}$ and $\overline{IORQ}$). During this time the interrupt logic of the CTC will determine the highest priority channel which is requesting an interrupt. To insure that the daisy chain enable lines stabilize, channels are inhibited from changing their interrupt request status when $\overline{M1}$ is active. If the CTC Interrupt Enable Input (IEI) is active, then the highest priority interrupting channel places the contents of its interrupt vector register onto the Data Bus when $\overline{IORQ}$ goes active. Additional wait cycles are allowed.



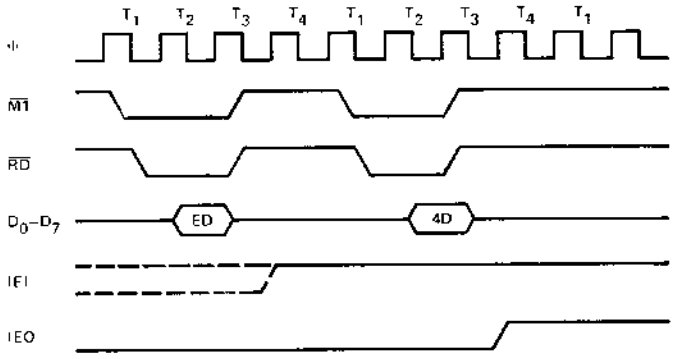
Timing Waveforms (continued)

RETURN FROM INTERRUPT CYCLE

If a Z80 peripheral device has no interrupt pending and is not under service, then its IEO = IEI. If it has an interrupt under service (i.e. it has already interrupted and received an interrupt acknowledge) then its IEO is always low, inhibiting lower priority chips from interrupting. If it has an interrupt pending which has not yet been acknowledged, IEO will be low unless an "ED" is decoded as the first byte of a two byte opcode. In this case, IEO will go high until the next opcode byte is decoded, whereupon it will again go low. If the second byte of the opcode was a "4D" then the opcode was an RETI instruction.

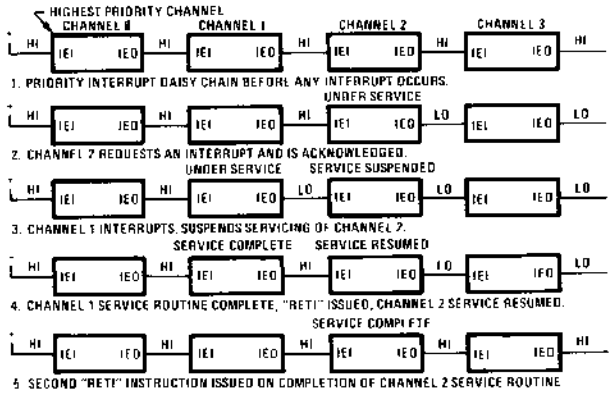
After an "ED" opcode is decoded, only the peripheral device which has interrupted and is currently under service will have its IEI high and its IEO low. This device is the highest priority device in the daisy chain which has received an interrupt acknowledge. All other peripherals have IEI = IEO. If the next opcode byte decoded is "4D", this peripheral device will reset its "interrupt under service" condition.

Wait cycles are allowed in the $\overline{M1}$ cycles.



DAISY CHAIN INTERRUPT SERVICING

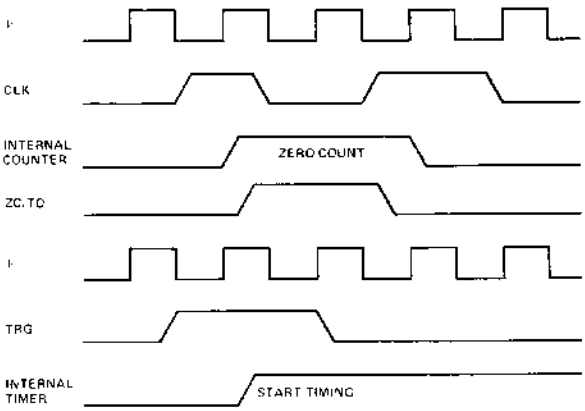
Illustrated at right is a typical nested interrupt sequence which may occur in the CTC. In this sequence channel 2 interrupts and is granted service. While this channel is being serviced, higher priority channel 1 interrupts and is granted service. The service routine for the higher priority channel is completed and a RETI instruction is executed to indicate to the channel that its routine is complete. At this time the service routine of lower priority channel 2 is completed.



CTC COUNTING AND TIMING

In the counter mode the rising or falling edge of the CLK input causes the counter to be decremented. The edge is detected totally asynchronously and must have a minimum CLK pulse width. However, the counter is synchronous with Φ therefore a setup time must be met when it is desired to have the counter decremented by the next rising edge of Φ.

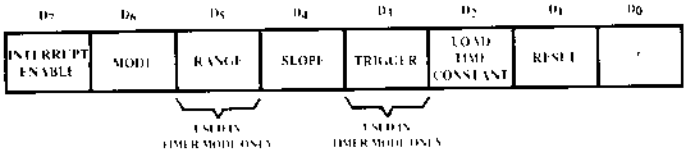
In the timer mode the prescaler may be enabled by a rising or falling edge on the TRG input. As in the counter mode, the edge is detected totally asynchronously and must have a minimum TRG pulse width. However, when timing is to start with respect to the next rising edge of Φ a setup time must be met. The prescaler counts rising edges of Φ.



CTC Programming

SELECTING AN OPERATING MODE

When selecting a channel's operating mode, bit Ø is set to 1 to indicate this word is to be stored in the channel control register.



Bit 7 = Ø — Channel interrupts disabled.

Bit 7 = 1 — Channel interrupts enabled to occur every time Down Counter reaches a count of zero. Setting Bit 7 does not let a preceding count of zero cause an interrupt.

Bit 6 = Ø — Timer Mode – Down counter is clocked by the prescaler. The period of the counter is:
$t_C \bullet P \bullet TC$
t_C = system clock period
P = prescale of 16 or 256
TC = 8 bit binary programmable time constant (256 max)

Bit 6 = 1 — Counter Mode – Down Counter is clocked by external clock. The prescaler is not used.

Bit 5 = Ø — Timer Mode Only–System clock Φ is divided by 16 in prescaler.

Bit 5 = 1 — Timer Mode Only–System clock Φ is divided by 256 in prescaler.

Bit 4 = Ø — Timer Mode – negative edge trigger starts timer operation.
Counter Mode – negative edge decrements the down counter.

Bit 4 = 1 — Timer Mode – positive edge trigger starts timer operation.
Counter Mode – positive edge decrements the down counter.

Bit 3 = Ø — Timer Mode Only – Timer begins operation on the rising edge of T_2 of the machine cycle following the one that loads the time constant.

Bit 3 = 1 — Timer Mode Only – External trigger is valid for starting timer operation after rising edge of T_2 of the machine cycle following the one that loads the time constant. The Prescaler is decremented 2 clock cycles later if the setup time is met, otherwise 3 clock cycles.

Bit 2 = Ø — No time constant will follow the channel control word. One time constant must be written to the channel to initiate operation.

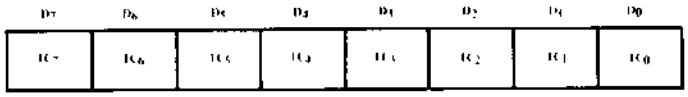
Bit 2 = 1 — The time constant for the Down Counter will be the next word written to the selected channel. If a time constant is loaded while a channel is counting, the present count will be completed before the new time constant is loaded into the Down Counter.

Bit 1 = Ø — Channel continues counting.

Bit 1 = 1 — Stop operation. If Bit 2 = 1 channel will resume operation after loading a time constant, otherwise a new control word must be loaded.

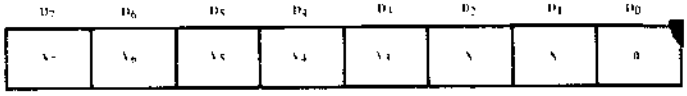
LOADING A TIME CONSTANT

An 8-bit time constant is loaded into the Time Constant register following a channel control word with bit 2 set. All zeros indicate a time constant of 256.



LOADING AN INTERRUPT VECTOR

The Z80-CPU requires that an 8-bit interrupt vector be supplied by the interrupting channel. The CPU forms the address for the interrupt service routine of the channel using this vector. During an interrupt acknowledge cycle the vector is placed on the Z80 Data Bus by the highest priority channel requesting service at that time. The desired interrupt vector is loaded into the CTC by writing into channel Ø with a zero in DØ. D_7-D_3 contain the stored interrupt vector, D_2 and D_1 are not used in loading the vector. When the CTC responds to an interrupt acknowledge, these two bits contain the binary code of the highest priority channel which requested the interrupt and DØ contains a zero since the address of the interrupt service routine starts at an even byte. Channel Ø is the highest priority channel.



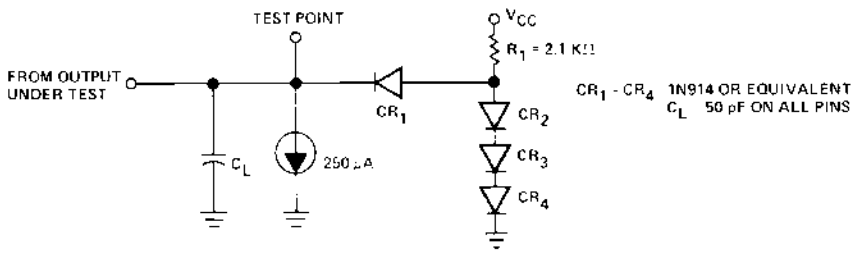
Z-80 CTC A.C. Characteristics

TA = 0° C to 70° C, Vcc = +5 V ± 5%, unless otherwise noted

Signal	Symbol	Parameter	Min	Max	Unit	Comments
Φ	t_C	Clock Period	400	[1]	ns	
	$t_W(\Phi H)$	Clock Pulse Width, Clock High	170	2000	ns	
	$t_W(\Phi L)$	Clock Pulse Width, Clock Low	170	2000	ns	
	t_r, t_f	Clock Rise and Fall Times		30	ns	
	t_H	Any Hold Time for Specified Setup Time	0		ns	
CS, $\overline{CE}$, etc.	$t_{S\Phi}(CS)$	Control Signal Setup Time to Rising Edge of Φ During Read or Write Cycle	160		ns	
D_0–D_7	$t_{DR}(D)$	Data Output Delay from Rising Edge of $\overline{RD}$ During Read Cycle		480	ns	[2]
	$t_{S\Phi}(D)$	Data Setup Time to Rising Edge of Φ During Write or M1 Cycle	60		ns	
	$t_{DI}(D)$	Data Output Delay from Falling Edge of IORQ During INTA Cycle		340	ns	[2]
	$t_F(D)$	Delay to Floating Bus (Output Buffer Disable Time)		230	ns	
IEI	$t_S(IEI)$	IEI Setup Time to Falling Edge of $\overline{IORQ}$ During INTA Cycle	200		ns	
IEO	$t_{DH}(IO)$	IEO Delay Time from Rising Edge of IEI		220	ns	[3]
	$t_{DL}(IO)$	IEO Delay Time from Falling Edge of IEI		190	ns	[3]
	$t_{DM}(IO)$	IEO Delay from Falling Edge of $\overline{M1}$ (Interrupt Occurring just Prior to $\overline{M1}$)		300	ns	[3]
$\overline{IORQ}$	$t_{S\Phi}(IR)$	$\overline{IORQ}$ Setup Time to Rising Edge of Φ During Read or Write Cycle	250		ns	
$\overline{M1}$	$t_{S\Phi}(M1)$	$\overline{M1}$ Setup Time to Rising Edge of Φ During INTA or M1 Cycle	210		ns	
$\overline{RD}$	$t_{S\Phi}(RD)$	$\overline{RD}$ Setup Time to Rising Edge of Φ During Read or M1 Cycle	240		ns	
$\overline{INT}$	$t_{DCK}(IT)$	$\overline{INT}$ Delay Time from Rising Edge of CLK/TRG		$2t_C(\Phi) + 200$		Counter Mode
	$t_{D\Phi}(IT)$	$\overline{INT}$ Delay Time from Rising Edge of Φ		$t_C(\Phi) + 200$		Timer Mode
CLK/TRG_{0-3}	$t_C(CK)$	Clock Period	$2t_C(\Phi)$			Counter Mode
	t_r, t_f	Clock and Trigger Rise and Fall Times		50		
	$t_S(CK)$	Clock Setup Time to Rising Edge of Φ for Immediate Count	210			Counter Mode
	$t_S(TR)$	Trigger Setup Time to Rising Edge of Φ for Enabling of Prescaler on Following Rising Edge of Φ	210			Timer Mode
	$t_W(CTH)$	Clock and Trigger High Pulse Width	200			Counter and Timer Modes
	$t_W(CTL)$	Clock and Trigger Low Pulse Width	200			Counter and Timer Modes
ZC/TO_{0-2}	$t_{DH}(ZC)$	ZC/TO Delay Time from Rising Edge of Φ, ZC/TO High		190		Counter and Timer Modes
	$t_{DL}(ZC)$	ZC/TO Delay Time from Falling Edge of Φ, ZC/TO Low		190		Counter and Timer Modes

Notes: [1] $t_C = t_W(\Phi H) + t_W(\Phi L) + t_r + t_f$.
[2] Increase delay by 10 nsec for each 50 pF increase in loading, 200 pF maximum for data lines and 100 pF for control lines.
[3] Increase delay by 2 nsec for each 10 pF increase in loading, 100 pF maximum
[4] $\overline{RESET}$ must be active for a minimum of 3 clock cycles.

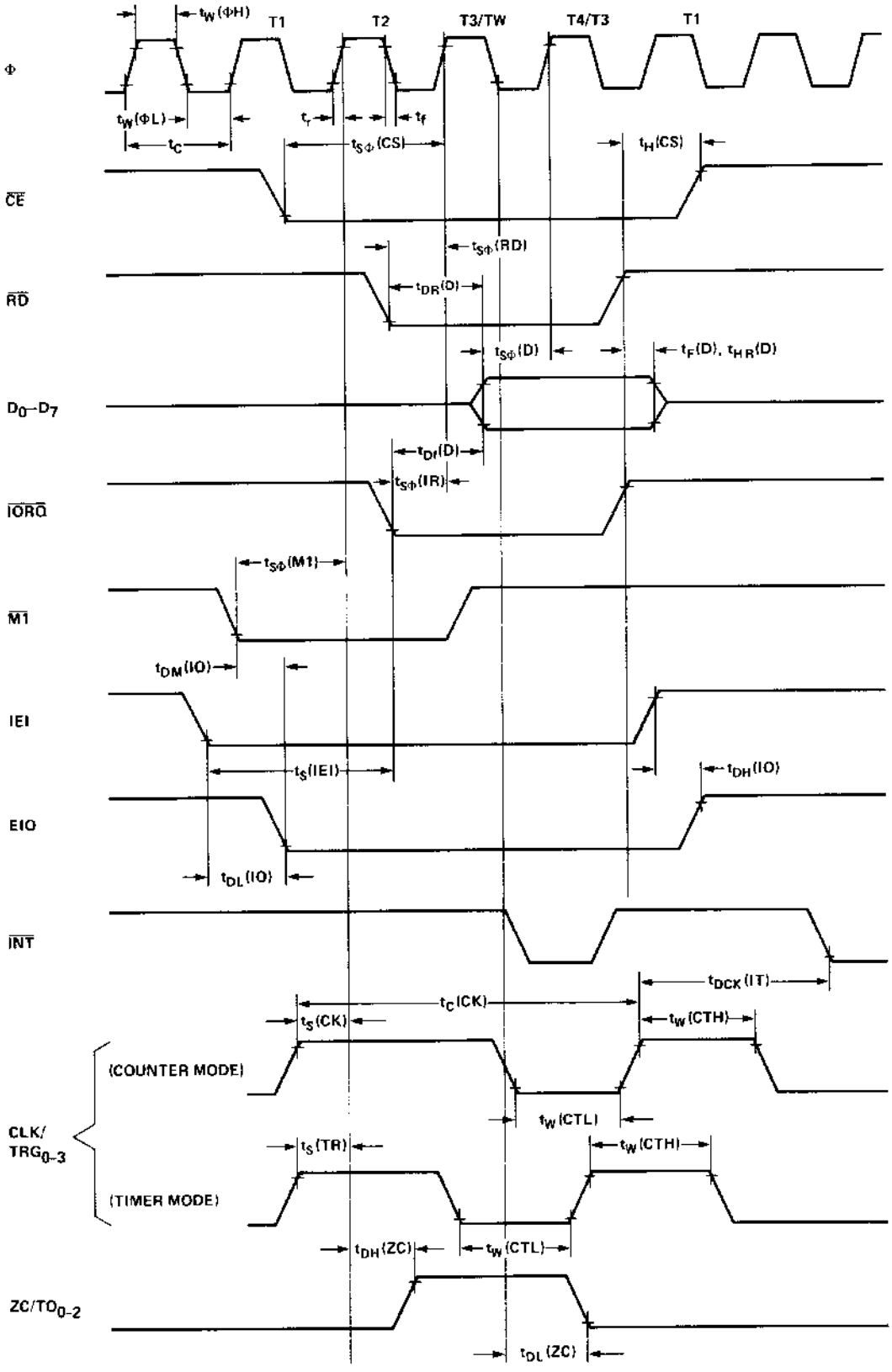
OUTPUT LOAD CIRCUIT



A.C. Timing Diagram

Timing measurements are made at the following voltages, unless otherwise specified:

	"1"	"0"
CLOCK	V_{CC} - .6V	.45V
OUTPUT	2.0V	.8V
INPUT	2.0V	.8V
FLOAT	ΔV	±0.5V



Zilog

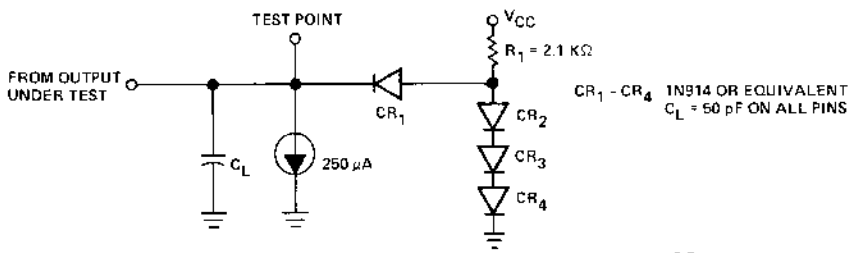
Z-80A CTC A.C. Characteristics

TA = 0° C to 70° C, Vcc = +5 V ± 5%, unless otherwise noted

Signal	Symbol	Parameter	Min	Max	Unit	Comments
Φ	t_C	Clock Period	250	[1]	ns	
	$t_W(\Phi H)$	Clock Pulse Width, Clock High	105	2000	ns	
	$t_W(\Phi L)$	Clock Pulse Width, Clock Low	105	2000	ns	
	t_r, t_f	Clock Rise and Fall Times		30	ns	
	t_H	Any Hold Time for Specified Setup Time	0		ns	
CS, $\overline{CE}$, etc	$t_{S\Phi}(CS)$	Control Signal Setup Time to Rising Edge of Φ During Read or Write Cycle	60		ns	
D_0–D_7	$t_{DR}(D)$	Data Output Delay from Falling Edge of $\overline{RD}$ During Read Cycle		380	ns	[2]
	$t_{S\Phi}(D)$	Data Setup Time to Rising Edge of Φ During Write or M1 Cycle	50		ns	
	$t_{DI}(D)$	Data Output Delay from Falling Edge of IORG During INTA Cycle		160	ns	[2]
	$t_F(D)$	Delay to Floating Bus (Output Buffer Disable Time)		110	ns	
IEI	$t_S(IEI)$	IEI Setup Time to Falling Edge of $\overline{IORQ}$ During INTA Cycle	140		ns	
IEO	$t_{DH}(IO)$	IEO Delay Time from Rising Edge of IEI		160	ns	[3]
	$t_{DL}(IO)$	IEO Delay Time from Falling Edge of IEI		130	ns	[3]
	$t_{DM}(IO)$	IEO Delay from Falling Edge of $\overline{M1}$ (Interrupt Occurring just Prior to $\overline{M1}$)		190	ns	[3]
$\overline{IORQ}$	$t_{S\Phi}(IR)$	$\overline{IORQ}$ Setup Time to Rising Edge of Φ During Read or Write Cycle	115		ns	
$\overline{M1}$	$t_{S\Phi}(M1)$	$\overline{M1}$ Setup Time to Rising Edge of Φ During INTA or M1 Cycle	90		ns	
$\overline{RD}$	$t_{S\Phi}(RD)$	$\overline{RD}$ Setup Time to Rising Edge of Φ During Read or M1 Cycle	115		ns	
$\overline{INT}$	$t_{DCK}(IT)$	$\overline{INT}$ Delay Time from Rising Edge of CLK/TRG		$2t_C(\Phi) + 140$		Counter Mode
	$t_{D\Phi}(IT)$	$\overline{INT}$ Delay Time from Rising Edge of Φ		$t_C(\Phi) + 140$		Timer Mode
CLK/TRG_{0-3}	$t_C(CK)$	Clock Period	$2t_C(\Phi)$			Counter Mode
	t_r, t_f	Clock and Trigger Rise and Fall Times		50		
	$t_S(CK)$	Clock Setup Time to Rising Edge of Φ for Immediate Count	210			Counter Mode
	$t_S(TR)$	Trigger Setup Time to Rising Edge of Φ for enabling of Prescaler on Following Rising Edge of Φ	210			Timer Mode
	$t_W(CTH)$	Clock and Trigger High Pulse Width	200			Counter and Timer Modes
	$t_W(CTL)$	Clock and Trigger Low Pulse Width	200			Counter and Timer Modes
ZC/TO_{0-2}	$t_{DH}(ZC)$	ZC/TO Delay Time from Rising Edge of Φ, ZC/TO High		190		Counter and Timer Modes
	$t_{DL}(ZC)$	ZC/TO Delay Time from Falling Edge of Φ, ZC/TO Low		190		Counter and Timer Modes

Notes: [1] $t_C = t_W(\Phi H) + t_W(\Phi L) + t_r + t_f$.
[2] Increase delay by 10 nsec for each 50 pF increase in loading, 200 pF maximum for data lines and 100 pF for control lines.
[3] Increase delay by 2 nsec for each 10 pF increase in loading, 100 pF maximum.
[4] $\overline{RESET}$ must be active for a minimum of 3 clock cycles.

OUTPUT LOAD CIRCUIT



Absolute Maximum Ratings

Temperature Under Bias	0° C to 70° C
Storage Temperature	-65° C to +150° C
Voltage On Any Pin With Respect To Ground	-0.3 V to +7 V
Power Dissipation	0.8W

*Comment

Stresses above those listed under "Absolute Maximum Rating" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other condition above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

D.C. Characteristics

TA = 0° C to 70° C, Vcc = 5 V ± 5% unless otherwise specified

Z-80 CTC

Symbol	Parameter	Min	Max	Unit	Test Condition
V_{ILC}	Clock Input Low Voltage	-0.3	.45	V	
V_{IHC}	Clock Input High Voltage [1]	V_{CC} - .6	V_{CC} + .3	V	
V_{IL}	Input Low Voltage	-0.3	0.8	V	
V_{IH}	Input High Voltage	2.0	V_{CC}	V	
V_{OL}	Output Low Voltage		0.4	V	I_{OL} = 2 mA
V_{OH}	Output High Voltage	2.4		V	I_{OH} = -250 μA
I_{CC}	Power Supply Current		120	mA	T_C = 400 nsec
I_{LI}	Input Leakage Current		10	μA	V_{IN} = 0 to V_{CC}
I_{LOH}	Tri-State Output Leakage Current in Float		10	μA	V_{OUT} = 2.4 to V_{CC}
I_{LOL}	Tri-State Output Leakage Current in Float		-10	μA	V_{OUT} = 0.4V
I_{OHD}	Darlington Drive Current	-1.5		mA	V_{EXT} = 1.5V R_{EXT} = 390Ω

Z-80A CTC

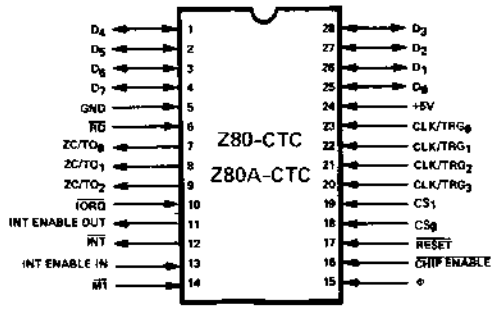
Symbol	Parameter	Min	Max	Unit	Test Condition
V_{ILC}	Clock Input Low Voltage	-0.3	.45	V	
V_{IHC}	Clock Input High Voltage [1]	V_{CC} - .6	V_{CC} + .3	V	
V_{IL}	Input Low Voltage	-0.3	0.8	V	
V_{IH}	Input High Voltage	2.0	V_{CC}	V	
V_{OL}	Output Low Voltage		0.4	V	I_{OL} = 2 mA
V_{OH}	Output High Voltage	2.4		V	I_{OH} = -250 μA
I_{CC}	Power Supply Current		120	mA	T_C = 250 nsec
I_{LI}	Input Leakage Current		10	μA	V_{IN} = 0 to V_{CC}
I_{LOH}	Tri-State Output Leakage Current in Float		10	μA	V_{OUT} = 2.4 to V_{CC}
I_{LOL}	Tri-State Output Leakage Current in Float		-10	μA	V_{OUT} = 0.4V
I_{OHD}	Darlington Drive Current	-1.5		mA	V_{EXT} = 1.5V R_{EXT} = 390Ω

Capacitance

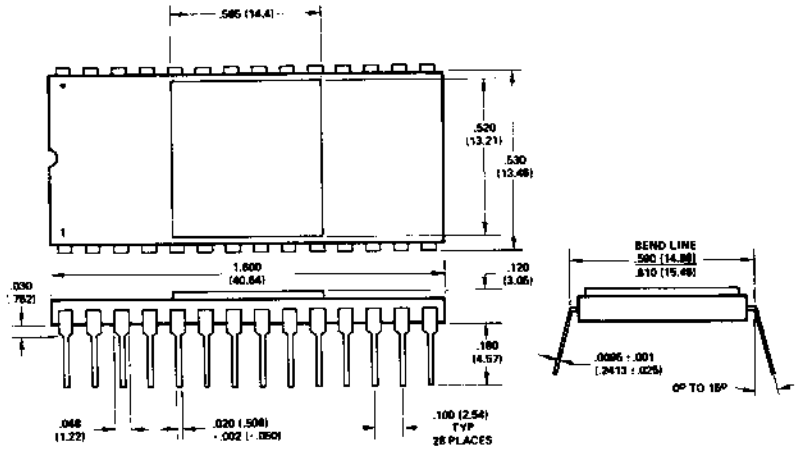
TA = 25° C, f = 1 MHz

Symbol	Parameter	Max.	Unit	Test Condition
C_Φ	Clock Capacitance	20	pF	Unmeasured Pins Returned to Ground
C_{IN}	Input Capacitance	5	pF	
C_{OUT}	Output Capacitance	10	pF	

Package Configuration



Package Outline



* DIMENSIONS FOR METRIC SYSTEM IN PARENTHESES (mm)

Zilog

Product Specification

Z-80® DMA
Z-80A DMA

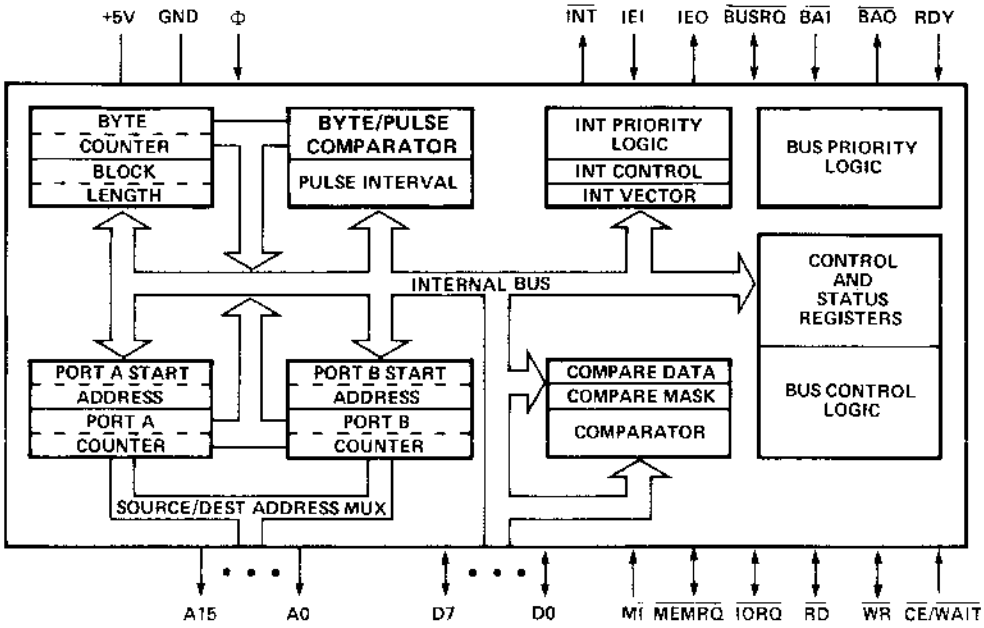
The Z-80 DMA (Direct Memory Access) circuit is a programmable single-channel device which provides all address, timing and control signals to effect the transfer of blocks of data between two ports within most microprocessor-based systems. These ports may be either system main memory or any system peripheral I/O device. The DMA can also search a block of data for a particular byte (bit maskable), with or without a simultaneous transfer.

Structure

- N-channel Silicon Gate Depletion Load Technology
- 40 Pin DIP
- Single 5 volt supply
- Single phase 5 volt clock
- Single channel, two port

Features

- Three classes of operation:
  - –Transfer Only
  - –Search Only
  - –Search-Transfer
- Address and Block Length Registers fully buffered. Values for next operation may be loaded without disturbing current values.
- Dual addresses generated during a transfer (one for read port and one for write port).
- Programmable data transfers and searches, automatically incrementing or decrementing the port addresses from programmed starting addresses (they can also remain fixed).
- Three modes of operation:
  - –Byte-at-a-time: One byte transferred per request
  - –Burst: Continues as long as ports are ready
  - –Continuous: Locks out CPU until operation complete
- Timing may be programmed to match the speed of any port.
- Interrupts on Match Found, End of Block, or Ready, may be programmed.
- An entire previous operation may be repeated automatically or on command. (Auto restart or Load)
- The DMA can signal when a specified number of bytes has been transferred, without halting transfer.
- Multiple DMA's easily configured for rotating priority.
- The channel may be enabled, disabled or reset under software control.
- Complete channel status upon program (CPU) request.
- Up to 1.25 megabyte/second Search.
- Daisy-chain priority interrupt and bus acknowledge included to provide automatic interrupt vectoring and bus request control, without need for additional external logic.
- TTL compatible inputs and outputs
- The CPU can read current Port counters, Byte counter, or Status Register. A mask byte can be set which defines which registers can be accessed during read operations.



DMA Internal Block Diagram

Fig. 1

DMA Architecture

A block diagram of the Z80 DMA is shown in Figure 1. The internal structure consists of the following circuitry:

- *Bus Interface*: provides driver and receiver circuitry to interface to the Z80-CPU Bus.
- *Control Logic and Registers*: set the class, mode and other basic control parameters of the DMA.
- *Address, Byte Count and Pulse Circuitry*: generates the proper port addresses for the read and write operations, with provisions for incrementing or decrementing the address. When zero bytes remain to be handled, the byte count circuitry sets a flag in the status register. Pulse circuitry generates a pulse each time the byte counter lower 8-bits equal the pulse register.
- *Timing Circuitry:* allows the user to completely specify the read/write timing for each port.
- *Match Circuitry*: holds the match byte and a mask byte which allows for the comparison of only certain bits within the byte. If a match is encountered during a Search or Transfer, this circuitry sets a flag in the status register.
- *$\overline{INT}$ and $\overline{BUSRQ}$ Circuitry:* includes a control register which specifies the conditions under which the DMA can generate an interrupt; priority encoding logic to select between the generation of an $\overline{INT}$ or $\overline{BUSRQ}$ output under these conditions; and an interrupt vector register for automatic vectoring to the interrupt service routine.
- *Status Register*: holds current status of DMA.

Register Description

The following DMA-internal registers are available to the programmer:

Control Registers: Write only; 8 bits. Hold DMA control information: such as, when to initiate an interrupt or pulse, what mode or class of operation to perform, etc.

Timing Registers: Write only; 8 bits. Hold read/write timing parameters for the two ports.

Interrupt Vector Register: Read/write; 8 bits. Holds the 8-bit vector that the DMA will put onto the data bus after receiving an $\overline{IORQ}$ during an interrupt acknowledge sequence if it is the highest priority device requesting an interrupt. (This register is readable only during interrupt acknowledge cycles.)

Block Length Register: Write only; 16 bits. Contains total block length of data to be searched and/or transferred.

Byte Counter: Read only; 16 bits. Counts number of bytes transferred (or searched). On a Load or Continue the Byte Counter is reset to zero. Thereafter, each byte transfer operation increments it until it matches the contents of the Block Length Register, at which time End of Block is set in the status register and operation is suspended if programmed. Also if so programmed the DMA will generate an interrupt.

Match Register: Write only; 8 bits. Holds the byte for which a match is being sought in Search operations.

Mask Register: Write only; 8 bits. Holds the 8-bit mask to determine which bits in the match register are to be examined for a match.

Starting Address Registers (Port A and Port B): Write only; 16 bits each. Hold the starting addresses (upper and lower 8 bits) for the two ports involved in Transfer operations. In Search Only operations, only one port address would have to be specified. Only memory starting addresses require both upper and lower 8 bits; I/O ports are generally addressed with only the lower 8 bits, and in this case the address contained in the register is a generally fixed address.

Address Counters (Port A and Port B): Read only; 16 bits each. These counters are loaded with the contents of the corresponding Starting Address Registers whenever Searches or Transfers are initiated with a Load or Continue. They are incremented, decremented or remain fixed, as programmed.

Pulse Control Register: Write only; 8 bits. The content of this register is continuously compared with the lower eight bits of the byte counter. When they become equal, the $\overline{INT}$ output is activated. Since this occurs while $\overline{BUSRQ}$ and $\overline{BUSAK}$ are both active, the CPU does not interpret this as an interrupt request. Instead, the signal is used to communicate with a peripheral I/O device. When the Pulse Control Register contains a value n, the first pulse is generated after n + 1 bytes of search or transfer. The next and all subsequent pulses occur at 256-byte intervals.

Status Register: Read only; 8 bits. Match, End of Block, Ready Active, Interrupt Pending, and DMA Cycle Occurred bits indicate these functions when set.

Modes of Operation

The DMA may be programmed for one of four modes of operation. (See Command Register 2B.)

- *Byte at a time*: control is returned to the CPU after each one-byte cycle.
- *Burst*: operation continues as long as the DMA's RDY input is active, indicating that the relevant port is ready. Control returns to the CPU when RDY is inactive or at end of block or a match if so programmed.
- *Continuous*: the entire Search and/or Transfer of a block of data is completed before control is returned to CPU.

Classes of Operation

The DMA has three classes of operation: Transfer only, Search Only and a combined Search-Transfer. (See Command Register 1A.)

During a Transfer, data is first read from one port and then written to the other port, byte by byte. (The DMA's two ports are termed Port A and Port B.) The ports may be programmed to be either system main memory or peripheral I/O devices. Thus, a block of data might be written from a peripheral to another; or it might be written from one area in main memory to another; or from a peripheral to main memory.

During a Search, data is read only, and compared byte by byte against two DMA-internal registers, one of which contains a match byte and the other an optional mask byte which allows only certain bits to be compared. If any byte of searched data matches, a DMA-internal status bit is set; if programmed to do so, the DMA will then suspend operation and/or generate an interrupt.

The third class of operation is a combined Search-Transfer. In such an operation a block of data is transferred as described above until a match is found; then, as in a Search Only operation, the transfer may be suspended and/or an interrupt generated.

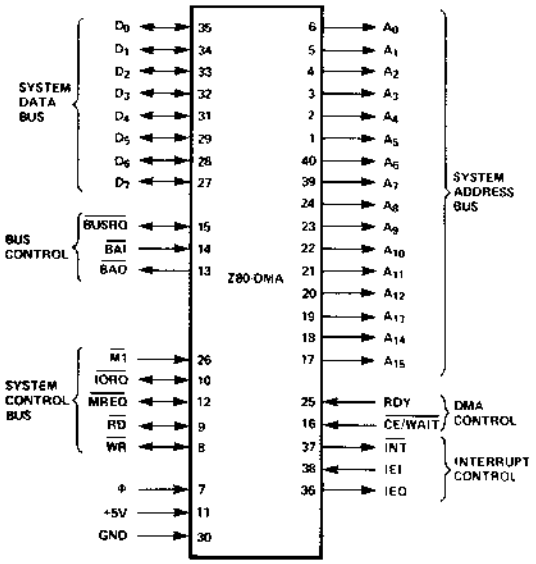
Addressing

The DMA's addressing of ports is either fixed or sequential, incrementing or decrementing from a starting address. The length of the operation (number of bytes) is specified by the programmed contents of a block length register. The DMA can address block lengths of up to 64K bytes. During a transfer two separate port addresses are generated, one during the Read cycle and one during the Write cycle.

Operating Sequence

Once the DMA has been programmed it may be "Enabled" (Command Register 2A or 2D). In the enabled condition when Ready goes active the DMA will request the bus by bringing $\overline{BUSRQ}$ low. The CPU will acknowledge this with a $\overline{BUSACK}$ which will normally be attached to $\overline{BAI}$. When the DMA receives $\overline{BAI}$ it will start its programmed operation releasing $\overline{BUSRQ}$ to a "high" state when it is through.

Z-80 DMA Pin Description



A_0-A_{15} System Address Bus. All sixteen of these pins are used by the DMA to address system main memory or an I/O port (output)

D_0-D_7 System Data Bus. Commands from the CPU, DMA status and data from memory or peripherals are transferred on these tristate pins (input/output)

+5V Power

GND Ground

Φ System clock (input)

$\overline{M1}$ Machine cycle One signal from CPU (input)

$\overline{IORQ}$ Input/Output Request to and from the System Bus (input/output)

$\overline{MREQ}$ Memory REQuest to the System Bus (input/output)

$\overline{RD}$ ReaD to and from the System Bus (input/output)

$\overline{WR}$ WRite to and from the System Bus (input/output)

$\overline{CE/WAIT}$ Chip Enable; may also be programmed to be $\overline{WAIT}$ during time when $\overline{BAI}$ is low (input)

$\overline{BUSRQ}$ BUS ReQuest. Requests control of the CPU Address Bus, Data Bus and Status/Control Bus (input/output, open drain)

$\overline{BAI}$ Bus Acknowledge In. Signals that the system buses have been released for DMA control (input)

$\overline{BAO}$ Bus Acknowledge Out. $\overline{BAI}$ and $\overline{BAO}$ form a daisy-chain connection for system-wide priority bus control (output)

$\overline{INT}$ INTerrupt request (output, open drain)

IEI Interrupt Enable In (input)

IEO Interrupt Enable Out. IEI and IEO form a daisy-chain connection for system-wide priority interrupt control (output)

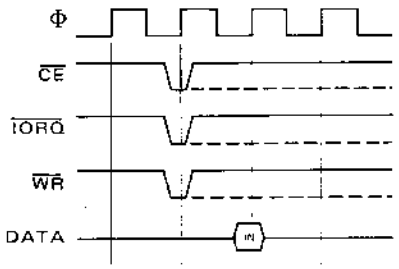
RDY ReaDY is monitored by the DMA to determine when a peripheral device associated with a DMA port is ready for a read or write operation (input, programmable as active high or low)

Zilog

DMA Timing Waveforms

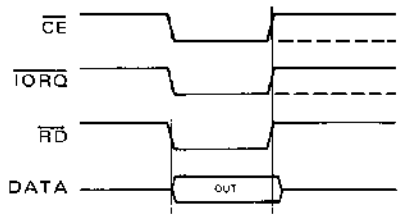
DMA Command Write Cycle

Illustrated here is the timing associated with a command byte or control byte being written to the DMA which is to be loaded into internal registers. Z80 Output instructions satisfy this timing.



DMA Register Read Cycle

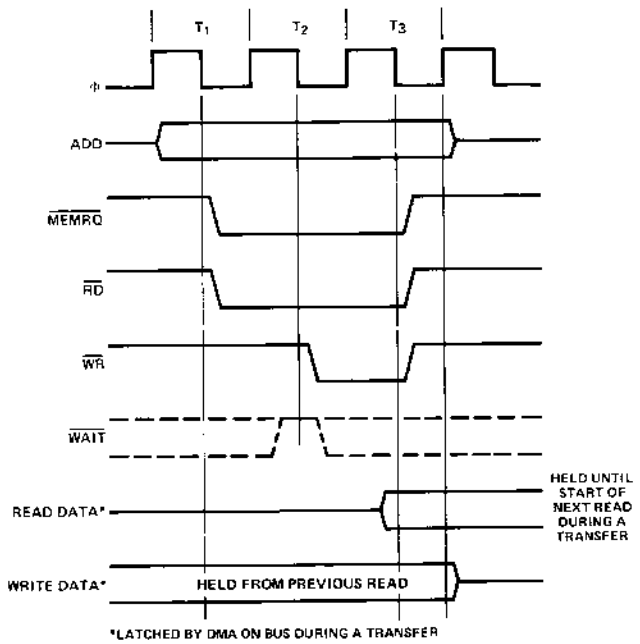
This timing is used when a read operation is performed on the DMA to access the contents of the Status Register, Address Counter or other readable registers. Z80 Input instructions satisfy this timing.



STD Memory Timing

This timing is exactly the same as used by the Z80-CPU to access system main memory, either in a Read or Write operation. The DMA will default to this timing after a power-on reset, or when a Reset or Reset Timing command is written to it; and unless otherwise programmed, will use this timing during all Transfer or Search operations involving system main memory. During the memory Read portion of a transfer cycle, data is latched in the DMA on the negative edge of Φ during T_3 and held into the following Write cycle. During the memory Write portion of a transfer cycle, data is held from the previous Read cycle and released at the end of the present cycle.

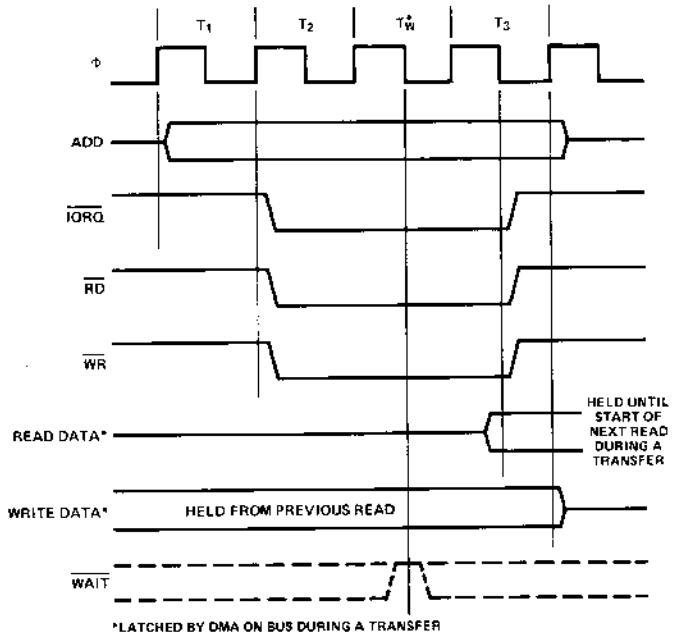
NOTE: The DMA is normally programmed for a 3 T-cycle duration in memory transactions. But $\overline{WAIT}$ is sampled during the negative transition of T_2, and if it is low, T_2 will be extended another T-cycle, during which $\overline{WAIT}$ will again be sampled. The duration of a memory transaction cycle may thus be indefinitely extended.



STD Peripheral Timing

This timing is identical to the Z80-CPU's Read/Write timing to I/O peripheral devices. The DMA will default to this timing after a power-on reset, or when a Reset or Reset Timing command is written to it; and unless otherwise programmed, will use this timing during all Transfer or Search operations involving I/O peripherals. During the I/O Read of a transfer cycle, data is latched on the negative edge of Φ during T_3 and is then held into the Write cycle. During an I/O Write, data is held from the previous Read cycle until the end of the Write cycle.

NOTE: If $\overline{WAIT}$ is low during the negative transition of T_W*, then T_W* will be extended another T-cycle and $\overline{WAIT}$ will again be sampled. The duration of a peripheral transaction cycle may thus be indefinitely extended.

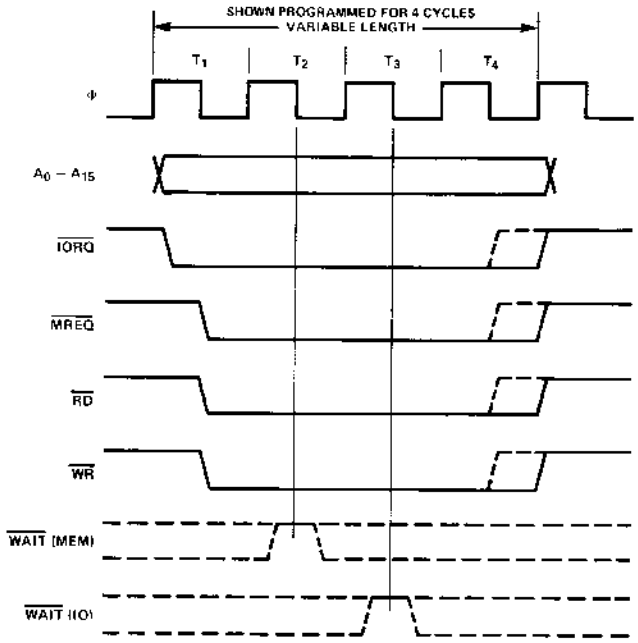


DMA Timing Waveforms (Continued)

Variable Cycle

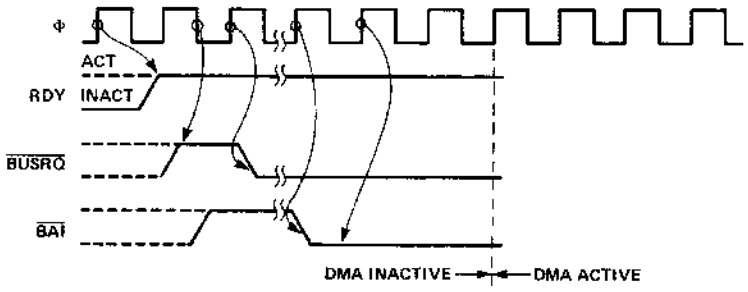
The Variable feature of the DMA allows the user to program the DMA's memory or peripheral transaction timing to values different than given above in the standard default diagrams. This permits the designer to tailor his timing to the particular requirements of his system components, and maximizes the data transfer rate while eliminating external signal conditioning logic. Cycle length can be two to four T-cycles (more if $\overline{WAIT}$ is used). Signal timing can be varied as shown. During a transfer, data will be latched by the DMA on the clock edge causing the rising edge of $\overline{RD}$ and will be held on the data lines until the end of the following Write cycle.

(See Timing Control Byte, page 7.)



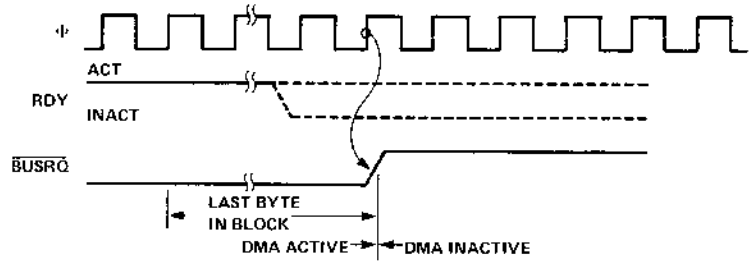
DMA Bus Request and Acceptance for Byte-at-a-Time, Burst, and Continuous Mode

Ready is sampled on every rising edge of Φ. When it is found to be active, the following rising edge of Φ generates $\overline{BUSRQ}$. After receiving $\overline{BUSRQ}$ the CPU will grant a $\overline{BUSAK}$ which will be connected to $\overline{BAI}$ either directly or through the Bus Acknowledge Daisy Chain. When a low is detected on $\overline{BAI}$ for two consecutive edges of Φ, the next rising edge of Φ will start an active DMA cycle.



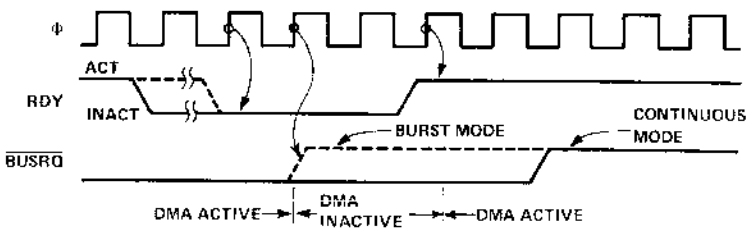
DMA Bus Release at End of Block for Burst or Continuous Mode

Timing for End of Block and DMA not programmed for Auto-restart.



DMA Bus Release with 'Ready' for Burst and Continuous Mode

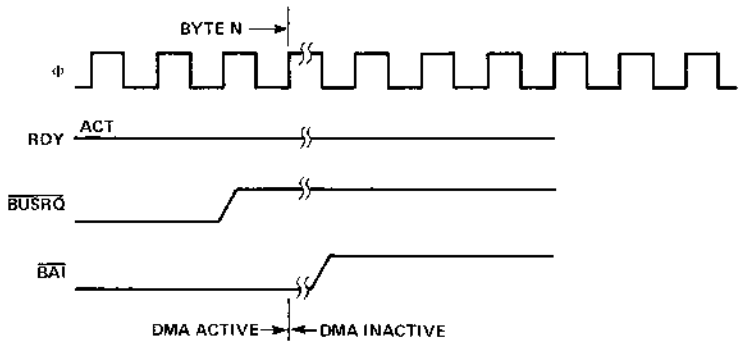
The DMA will relinquish the bus after RDY has gone inactive (Burst mode) or after an End of Block or a Match is found (Continuous mode). With RDY inactive, the DMA in Continuous mode is inactive but maintains control of the bus ($\overline{BUSRQ}$ low) until the cycle is resumed when RDY goes active.



DMA Timing Waveforms (Continued)

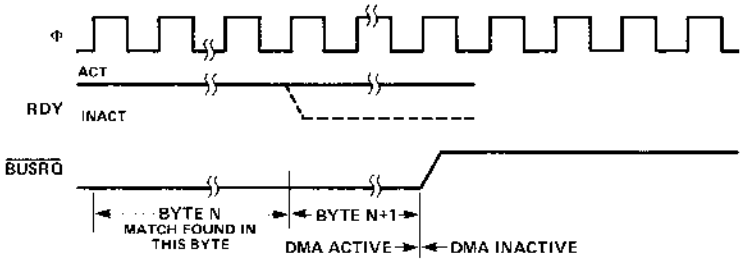
DMA Bus Release for Byte-at-a-Time Mode

In the Byte mode the DMA will release $\overline{BUSRQ}$ on the rising edge of Φ prior to the end of each Read cycle in Search Only or each Write cycle in a Transfer, regardless of the state of RDY. The next bus request will come after both $\overline{BUSRQ}$ and $\overline{BAI}$ have returned high.



DMA Bus Release with Match for Burst or Continuous Modes

When a Match is found and the DMA is programmed to stop on Compare, the DMA performs an operation on the next byte and then releases bus.



Reading the DMA Internal Registers

The CPU can read seven internal DMA registers, always in the following order: Status, lower byte of the Block Length register, upper byte of the Block Length register, lower byte of the Port A Address, upper byte of the Port A Address, lower byte of the Port B Address and the upper byte of the Port B Address.

The Read Mask register must be programmed to either include or exclude any of these seven registers by programming a 1 (include) or 0 (exclude) in the appropriate positions of the Read Mask register. After a Reset or Load, the read sequence must be initiated through an Initiate Read Sequence command (Command Byte 2D). The sequence of reading all registers that are not excluded by the Read Mask register must be completed before a new Initiate Read Sequence or RD Status command.

Programming the DMA

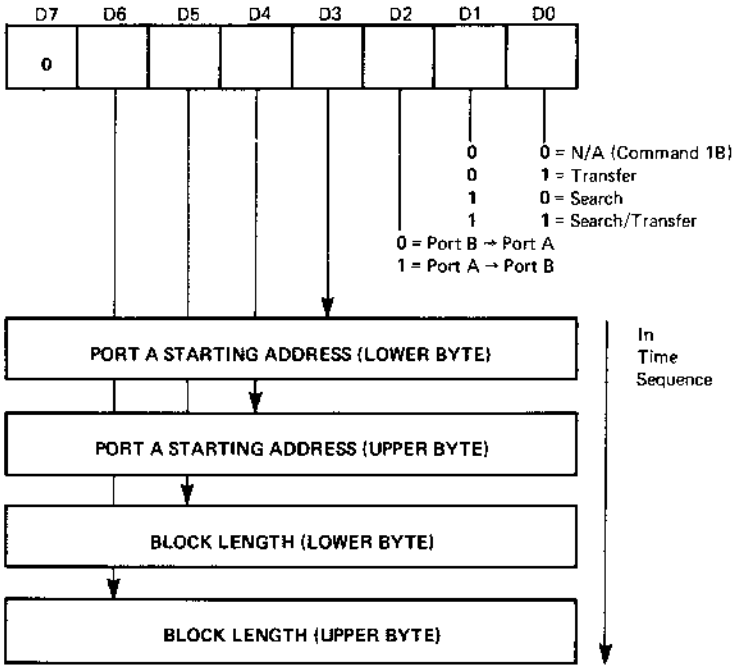
Previous sections of this specification have indicated the various functions and modes of the DMA. The diagrams and charts below show how the DMA is programmed to select among these functions and modes and to adapt itself to the requirements of the user system.

The Z80-DMA chip may be in an "enable" state, in which it can gain control of the system buses and direct the transfer of data between its ports, or in a "disable" state, when it cannot gain control of the bus. Program commands can be written to it in either state, but writing a command to it automatically puts it in the disable state, which is maintained until an enable command is issued to the DMA. The CPU must program it in advance of any data search or transfer by addressing it as an I/O port and sending it a sequence of command bytes via the system data bus using Output instructions. When the DMA is powered up or reset by any means, the DMA is automatically placed into a disable state, in which it can initiate neither bus requests nor data transfers nor interrupts.

The command bytes contain information to be loaded into the DMA's control and other registers and/or information to alter the state of the chip, such as an Enable Interrupt command. The command structure is designed so that certain bits in some commands can be set to alert the DMA to expect the next byte written to it to be for a particular internal register.

The following diagrams and charts give the function of each bit in the six different command bytes. Two of these are defined as being from Group 1, and are termed command bytes 1A and 1B. These Group 1 commands contain the most basic DMA set-up information. The other four are categorized as Group 2, and are termed commands 2A, 2B, 2C and 2D. Group 2 words specify more detailed set-up information.

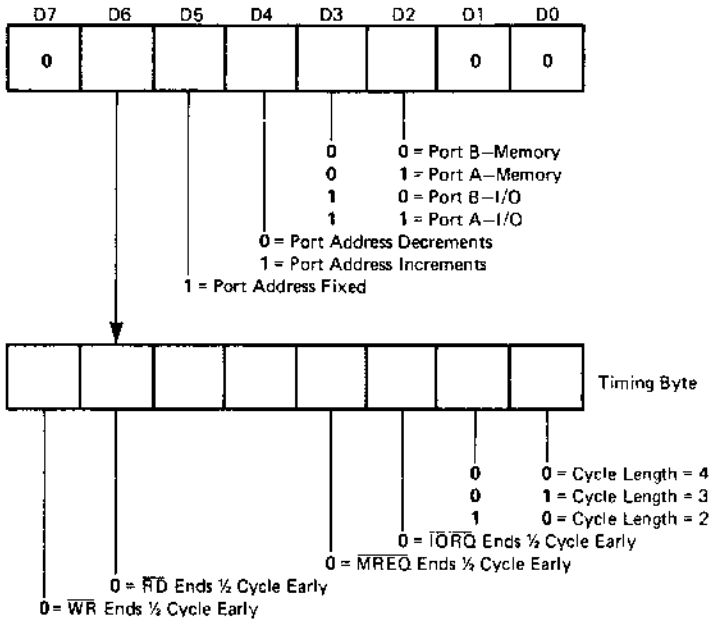
Command Register 1A



A "1" in positions D_3 through D_6 means that the indicated byte will follow. Note that the sequence of bytes is absolutely rigid.

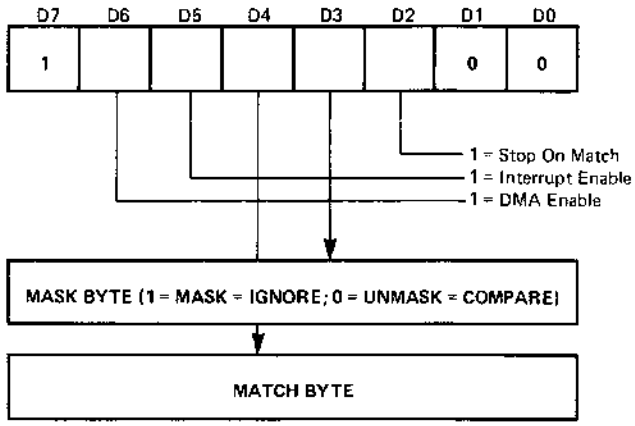
The DMA always transfers or searches one byte more than the number written into the Block Length registers. A "0" in the block length register results in the transfer or search of 2^{16} + 1 bytes. The shortest programmable block length is therefore two bytes long, programmed by writing a 1 into the Block Length register.

Command Register 1B

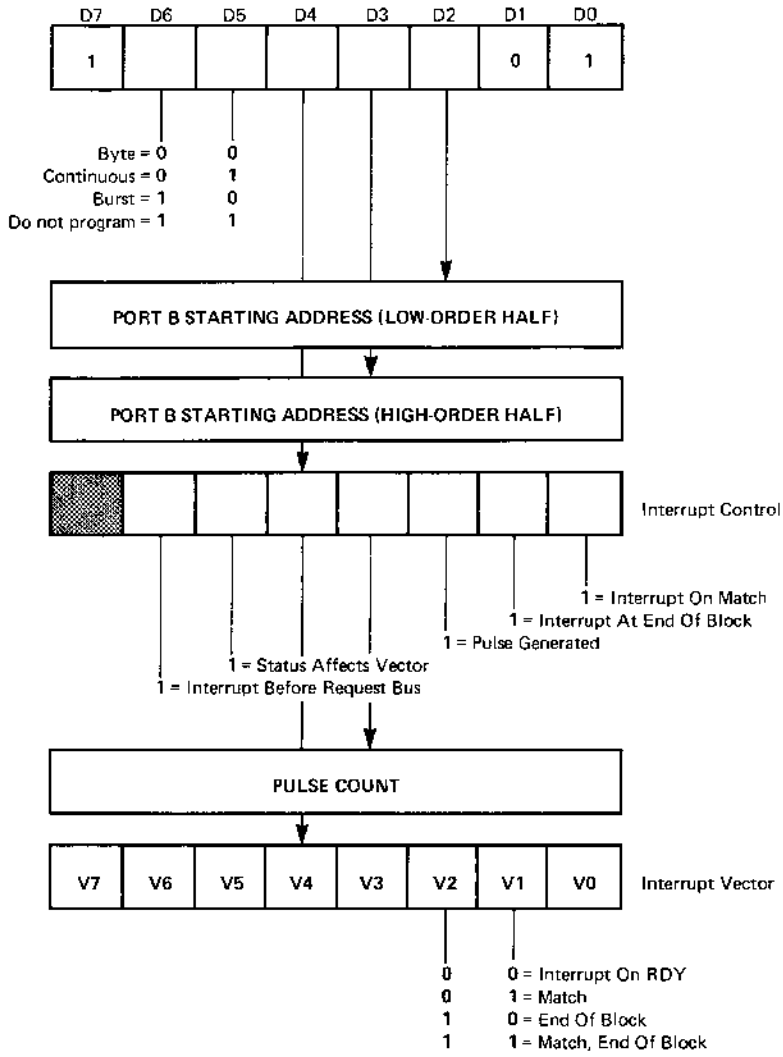


For transfers, this byte is normally written twice, once for Port A and again for Port B.

Command Register 2A



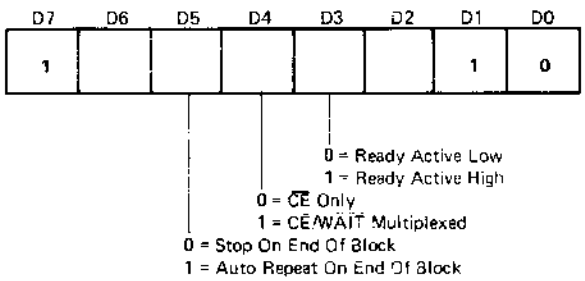
Command Register 2B



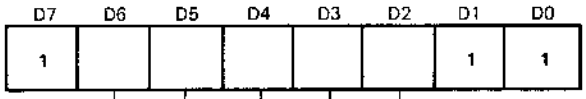
[1] If "Interrupt Before Requesting Bus" is selected (by a 1 in bit 6 of the Interrupt Control byte), the Z-80 DMA does not request the bus until the following set of instructions has been received by the Z-80 DMA:

- Enable after RETI command (B7 in Command byte 2D)
- Enable DMA command (87 in Command byte 2D)
- A RETI instruction that resets the IUS (Interrupt Under Service latch) in the Z-80 DMA

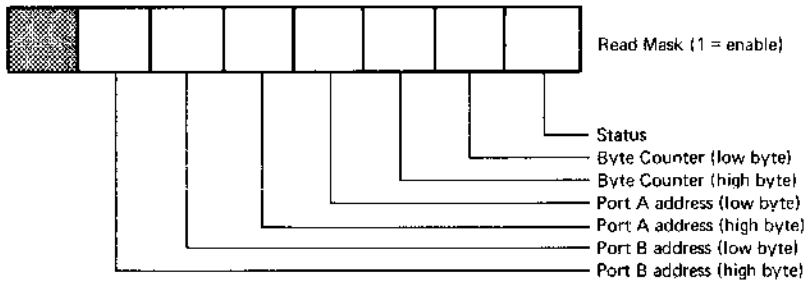
Command Register 2C



Command Register 2D



HEX					
C3	1	0	0	0	0 = **Reset** Interrupt circuitry, disable interrupt and bus request logic, unforce internal ready condition, disable "MUXCE" and stop auto repeat.
C7	1	0	0	0	1 = **Reset Port A Timing** to standard Z-80 CPU timing.
CB	1	0	0	1	0 = **Reset Port B Timing** to standard Z-80 CPU timing.
CF	1	0	0	1	1 = **Load** starting address for both ports, clear byte counter.*
D3	1	0	1	0	0 = Addresses **continue** from present locations, clear byte counter.
AB	0	1	0	1	0 = **Enable interrupts**
AF	0	1	0	1	1 = **Disable interrupts**
A3	0	1	0	0	0 = **Reset and disable interrupt** circuits (like RETI) and unforce the internal ready condition
87	0	0	0	0	1 = **Enable DMA** ⎫ Both affect all operations except interrupts, but do not
83	0	0	0	0	0 = **Disable DMA** ⎭ reset any functions.
A7	0	1	0	0	1 = **Initiate read** sequence to the first register designated as readable by the Read Mask register.
BF	0	1	1	1	1 = **Set read status** so next read is from status register.
B3	0	1	1	0	0 = **Force** an internal **ready** condition independent of the RDY input. Used for memory-to-memory operations where no RDY signal is needed. This command does not function in the "byte-at-a-time" mode.
8B	0	0	0	1	0 = **Clear** Match and End of Block **status** bits.
B7	0	1	1	0	1 = **Enable after RETI** so DMA will request bus only after receiving a RETI. Must be followed by an Enable DMA command.
BB	0	1	1	1	0 = **Read mask** is the following byte.



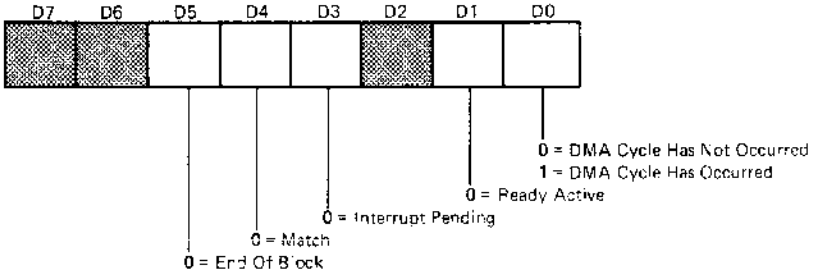
* Loading Port Addresses. The "Load" command (CF in Command Register 2D) loads a fixed address only into a port selected as the source, not into a port selected as the destination. Therefore, the destination address must be loaded by temporarily mislabeling the destination as the source.

The following example is a set-up procedure for a transfer from Port A to Port B:

1. Command byte 1A with B as source port
2. Command byte 2D with CF = load
3. Command byte 1A with A as source port
4. Command byte 2D with CF = load
5. Command byte 2D with 87 = Enable DMA

This manipulation is required only when the destination has a fixed address.

Status Register



The Sample DMA Program shows how the DMA may be programmed to transfer data from memory (Port A) to a peripheral device (Port B). In this example, the Port A memory starting address is 1050_H and the Port B peripheral fixed address is 05_H. Note that the data flow is 1001_H bytes–one more than specified by the block length. The table of DMA commands may be stored in consecutive memory locations and transferred to the DMA with an output instruction such as OTIR.

Figure 1: Sample DMA Program

	D7	D6	D5	D4	D3	D2	D1	D0	HEX
1) Command Register 1A sets DMA to receive block length, Port A starting address and temporarily sets Port B as source.	0 Group One	1 Block Length Upper Follows	1 Block Length Lower Follows	1 Port A Upper Addr Follows	1 Port A Lower Addr Follows	0 B→A Temporary For Loading B Address	0	1 Command Byte 1A Transfer, No Search	79
2) Port A address (lower)	0	1	0	1	0	0	0	0	50
3) Port A address (upper)	0	0	0	1	0	0	0	0	10
4) Block length (lower)	0	0	0	0	0	0	0	0	00
5) Block length (upper)	0	0	0	1	0	0	0	0	10
6) Command Register 1B defines Port A as memory with incrementing address.	0 Group One	0 No Timing Follows	0 Address Changes	1 Address Increments	0 Port Is Memory	1 This Is Port A	0	0 Byte 1B	14
7) Command Register 1B defines Port B as peripheral with fixed address.	0 Group One	0 No Timing Follows	1 Fixed Address	0 Not Used	1 Port Is I/O	0 This Is Port B	0	0 Byte 1B	28
8) Command Register 2B sets mode to Burst, sets DMA to expect Port B address.	1 Group Two	1	0 Burst Mode	0 No Interrupt Control Byte Follows	0 No Upper Address	1 Port B Lower Addr Follows	0	1 Byte 2B	C5
9) Port B address (lower)	0	0	0	0	0	1	0	1	05
10) Command Register 2C sets Ready active High.	1 Group Two	0 Not Used	0 No Auto Restart	0 No Wait States	1 RDY Active HIGH	0 Not Used	1	0 Byte 2C	8A
11) Command Register 2D loads Port B address and resets block counter.	1 Group Two	1	0	0 Load	1	1	1	1 Byte 2D	CF
12) Command Register 1A sets Port A as source. *	0 Group One	0	0	0 No Addr Or Block Length Bytes	0	1 A→B	0	1 Byte 1A, Transfer No Search	03
13) Command Register 2D loads Port A address and resets block counter. *	1 Group Two	1	0	0 Load	1	1	1	1 Byte 2D	CF
14) Command byte 2D enables DMA to start operation.	1 Group Two	0	0	0 Enable DMA		1	1	1 Byte 2D	87

NOTE: The actual number of bytes transferred is one more than specified by the block length.
* These commands are necessary only in the case of a fixed destination address.

Absolute Maximum Ratings

Temperature Under Bias	0°C to +70°C
Storage Temperature	−65°C to +150°C
Voltage On Any Pin with Respect to Ground	−0.3V to +7V
Power Dissipation	1.5W

*Comment

Stresses above those listed under "Absolute Maximum Rating" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other condition above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

D.C. Characteristics

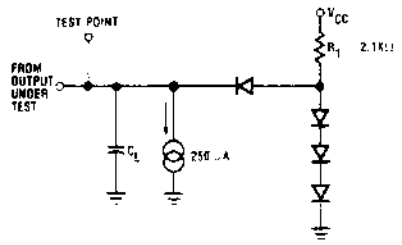
T_A = 0°C to 70°C, V_{CC} = 5V ±5% unless otherwise specified.

Symbol	Parameter	Min.	Max.	Unit	Test Condition
V_{ILC}	Clock Input Low Voltage	−0.3	0.45	V	
V_{IHC}	Clock Input High Voltage	V_{CC}−.6	5.5	V	
V_{IL}	Input Low Voltage	−0.3	0.8	V	
V_{IH}	Input High Voltage	2.0	5.5	V	
V_{OL}	Output Low Voltage		0.4	V	I_{OL} = 3.2 mA for $\overline{BUSRQ}$ I_{OL} = 2.0 mA for all others
V_{OH}	Output High Voltage	2.4		V	I_{OH} = −250 μA
I_{CC}	Power Supply Current Z-80 DMA Z-80A DMA		150 200	mA mA	t_C = 400 ns t_C = 250 ns
I_{LI}	Input Leakage Current		10	μA	V_{IN} = 0 to V_{CC}
I_{LOH}	Tri-State Output Leakage Current in Float		10	μA	V_{OUT} = 2.4 to V_{CC}
I_{LOL}	Tri-State Output Leakage Current in Float		−10	μA	V_{OUT} = 0.4V
I_{LD}	Data Bus Leakage Current in Input Mode		±10	μA	$0 \leqslant V_{IN} \leqslant V_{CC}$

Capacitance

T_A = 25°C, f = 1 MHz

Symbol	Parameter	Max.	Unit	Test Condition
C_Φ	Clock Capacitance	35	pF	Unmeasured Pins
C_{IN}	Input Capacitance	5	pF	Returned to Ground
C_{OUT}	Output Capacitance	10	pF	



C_L = 50 pF. Increase delay by 10 ns for each 50 pF increase in C_L, up to 200 pF maximum.

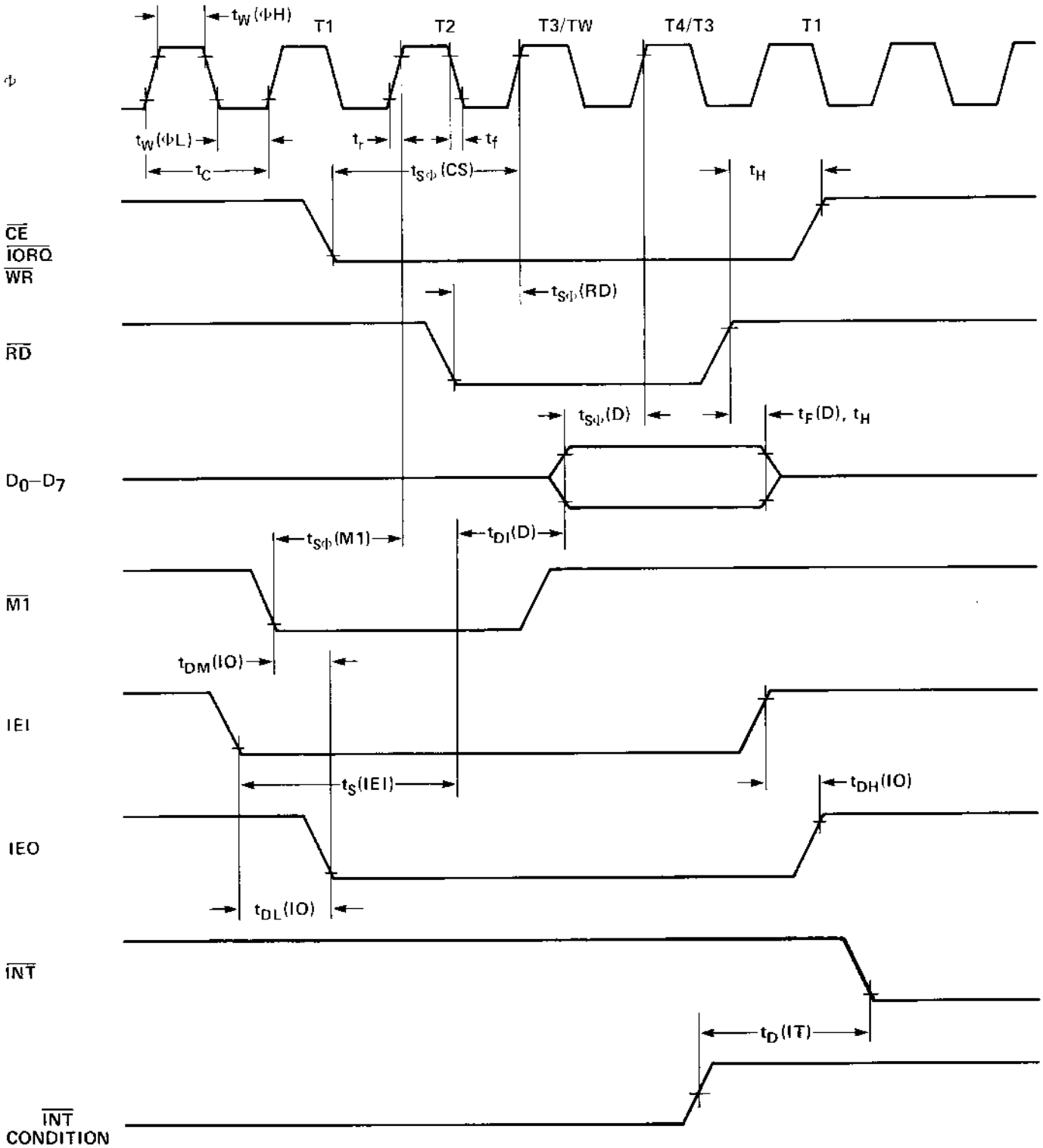
Zilog

A.C. Timing Diagrams

Z80 and Z80A as a Peripheral Device (Inactive State)

Timing measurements are made at the following voltages, unless otherwise specified:

	"1"	"0"
CLOCK	4.2V	0.8V
OUTPUT	2.0V	0.8V
INPUT	2.0V	0.8V
FLOAT	ΔV	= +0.5V



NOTE: This diagram does not show an actual timing sequence. Refer to this diagram only for the detailed timing relationships of individual edges. Use the illustrations in the "DMA Timing Waveforms" section as an explanation of the various timing sequences.

Z80-DMA as a Peripheral Device (Inactive State).
$T_A = 0°C$ to $70°C$, Vcc = +5V±5%, Unless Otherwise Noted

SIGNAL	SYMBOL	PARAMETER	Z-80 DMA		Z-80A DMA*		UNIT
			MIN	MAX	MIN	MAX	
Φ	t_c	Clock Period	400	[1]	250	[1]	nsec
	$t_{w(\Phi H)}$	Clock Pulse Width, Clock High	170	2000	105	2000	nsec
	$t_{w(\Phi L)}$	Clock Pulse Width, Clock Low	170	2000	105	2000	nsec
	$t_{r,f}$	Clock Rise and Fall Times		30		30	nsec
	t_H	Any Hold Time for Specified Setup Time	0		0		nsec
$\overline{CE}$, $\overline{WR}$ $\overline{IORQ}$	$t_{S\Phi(CS)}$	Control Signal Setup Time to Rising Edge of Φ during Write Cycle ($\overline{IORQ}$, $\overline{WR}$, $\overline{CE}$)	280		145		nsec
D_{0-7}	$t_{DR(D)}$	Data Output Delay from Falling Edge of $\overline{RD}$		500		380	nsec
	$t_{S\Phi(D)}$	Data Setup Time to Rising Edge of Φ during Write or $\overline{M1}$ Cycle	50		50		nsec
	$t_{DI(D)}$	Data Output Delay from Falling Edge of $\overline{IORQ}$ during INTA Cycle		340		250	nsec
	$t_{F(D)}$	Delay to Floating Bus (Output Buffer Disable Time)		160		110	nsec
IEI	$t_{S(IEI)}$	IEI Setup Time to Falling Edge of $\overline{IORQ}$ during INTA Cycle	140		140		nsec
IEO	$t_{DH(IE)}$	IEO Delay Time from Rising Edge of IEI		210		160	nsec
	$t_{DL(IO)}$	IEO Delay Time from Falling Edge of IEI		190		130	nsec
	$t_{DM(IO)}$	IEO Delay from Falling Edge of $\overline{M1}$ (Interrupt Occurring Just Prior to $\overline{M1}$) See Note A		300		190	nsec
$\overline{M1}$	$t_{S\Phi(M1)}$	$\overline{M1}$ Setup Time to Rising Edge of Φ during INTA or $\overline{M1}$ Cycle. See Note B.	210		90		nsec
$\overline{RD}$	$t_{S\Phi(RD)}$	$\overline{RD}$ Setup Time to Rising Edge of Φ during $\overline{M1}$ Cycle.	240		115		nsec
$\overline{INT}$	$t_{D(IT)}$	$\overline{INT}$ Delay Time from Condition Causing $\overline{INT}$. $\overline{INT}$ generated only when DMA is inactive.		500		500	nsec
$\overline{BAO}$	$t_{DH(BO)}$	BAO Delay from Rising Edge of BAI	150	200	150	200	nsec
	$t_{DL(BO)}$	BAO Delay from Falling Edge of BAI	150	200	150	200	nsec

* Z-80A DMA Timing Specifications are preliminary.

Zilog

A.C. Characteristics

Z80-DMA as a Bus Controller (Active State)
T_A = 0°C to 70°C, Vcc = +5V±5%, Unless Otherwise Noted.

SIGNAL	SYMBOL	PARAMETER	Z-80 DMA		Z-80A DMA		UNIT
			MIN	MAX	MIN	MAX	
Φ	t_C	Clock Period	400		250		nsec
	$t_{W(\Phi H)}$	Clock Pulse Width, Clock High	180	2000	110	2000	nsec
	$t_{W(\Phi L)}$	Clock Pulse Width, Clock Low	180	2000	110	2000	nsec
	$t_{r,f}$	Clock Rise and Fall Time		30		30	nsec
A_0–A_{15}	$t_{D(AD)}$	Address Output Delay		145		110	nsec
	$t_{F(AD)}$	Delay to Float		110		90	nsec
	t_{acm}	Address Stable Prior to $\overline{MREQ}$ (Memory Cycle)	$t_{W(\Phi H)} + t_f - 75$		$t_{W(\Phi H)} + t_f - 60$		nsec
	t_{aci}	Address Stable Prior to $\overline{IORQ}$, $\overline{RD}$ or $\overline{WR}$ (I/O Cycle)	$t_C - 80$		$t_C - 70$		nsec
	t_{ca}	Address Stable from $\overline{RD}$ or $\overline{WR}$	$t_{W(\Phi L)} + t_r - 40$		$t_{W(\Phi L)} + t_r - 30$		nsec
	t_{caf}	Address Stable from $\overline{RD}$ or $\overline{WR}$ during Float	$t_{W(\Phi L)} + t_r - 60$		$t_{W(\Phi L)} + t_r - 50$		nsec
D_0–D_7	$t_{D(D)}$	Data Output Delay		230		150	nsec
	$t_{F(D)}$	Delay to Float during Write Cycle		90		90	nsec
	$t_{S\Phi(D)}$	Data Setup Time to Rising Edge of Clock during Read when Falling Edge Ends $\overline{RD}$	50				nsec
	$t_{S\overline{\Phi}(D)}$	Data Setup Time to Falling Edge of Clock during Read when Falling Edge Ends $\overline{RD}$	60				nsec
	t_{dcm}	Data Stable Prior to $\overline{WR}$ (Memory Cycle)	$t_C - 210$		$t_C - 170$		nsec
	t_{dci}	Data Stable Prior to $\overline{WR}$ (I/O Cycle)	$t_{W(\Phi L)} + t_r - 210$		$t_{W(\Phi L)} + t_r - 170$		nsec
	t_{cdf}	Data Stable from $\overline{WR}$	$t_{W(\Phi L)} + t_r - 80$		$t_{W(\Phi L)} + t_r - 70$		nsec
	t_H	Any Hold Time where Setup Time is Specified	0			0	nsec
$\overline{MREQ}$	$t_{DL\Phi(MR)}$	$\overline{MREQ}$ Delay from Rising Edge of Clock, $\overline{MREQ}$ Low		100		85	nsec
	$t_{DH\overline{\Phi}(MR)}$	$\overline{MREQ}$ Delay from Falling Edge of Clock, $\overline{MREQ}$ Low		100		85	nsec
	$t_{DH\Phi(MR)}$	$\overline{MREQ}$ Delay from Rising Edge of Clock, $\overline{MREQ}$ High		100		85	nsec
	$t_{DH\overline{\Phi}(MR)}$	$\overline{MREQ}$ Delay from Falling Edge of Clock, $\overline{MREQ}$ High		100		85	nsec
	$t_{W(\overline{MRL})}$	Pulse Width, $\overline{MREQ}$ Low	$t_C - 40$		$t_C - 30$		nsec
	$t_{W(\overline{MRH})}$	Pulse Width, $\overline{MREQ}$ High	$t_{W(\Phi H)} + t_f - 30$		$t_{W(\Phi H)} + t_f - 20$		nsec
$\overline{IORQ}$	$t_{DL\Phi(IR)}$	$\overline{IORQ}$ Delay from Rising Edge of Clock, $\overline{IORQ}$ Low		90		75	nsec
	$t_{DL\overline{\Phi}(IR)}$	$\overline{IORQ}$ Delay from Falling Edge of Clock, $\overline{IORQ}$ Low		110		85	nsec
	$t_{DH\overline{\Phi}(IR)}$	IORQ Delay from Rising Edge of Clock, $\overline{IORQ}$ High		100		85	nsec
	$t_{DH\Phi(IR)}$	$\overline{IORQ}$ Delay from Falling Edge of Clock, $\overline{IORQ}$ Low		110		85	nsec
$\overline{RD}$	$t_{DL\Phi(RD)}$	$\overline{RD}$ Delay from Rising Edge of Clock, $\overline{RD}$ Low		100		85	nsec
	$t_{DL\overline{\Phi}(RD)}$	$\overline{RD}$ Delay from Falling Edge of Clock, $\overline{RD}$ Low		130		95	nsec
	$t_{DH\Phi(RD)}$	$\overline{RD}$ Delay from Rising Edge of Clock, $\overline{RD}$ High		100		85	nsec
	$t_{DH\overline{\Phi}(RD)}$	$\overline{RD}$ Delay from Falling Edge of Clock, $\overline{RD}$ High		110		85	nsec
$\overline{WR}$	$t_{DL\Phi(WR)}$	$\overline{WR}$ Delay from Rising Edge of Clock, $\overline{WR}$ Low		80		65	nsec
	$t_{DL\overline{\Phi}(WR)}$	$\overline{WR}$ Delay from Falling Edge of Clock, $\overline{WR}$ Low		90		80	nsec
	$t_{DH\Phi(WR)}$	$\overline{WR}$ Delay from Rising Edge of Clock, $\overline{WR}$ High		100		80	nsec
	$t_{DH\overline{\Phi}(WR)}$	$\overline{WR}$ Delay from Falling Edge of Clock, $\overline{WR}$ High		100		80	nsec
	$t_{W(\overline{WRL})}$	Pulse Width, $\overline{WR}$ Low	$t_C - 40$		$t_C - 30$		nsec
$\overline{WAIT}$	$t_{S(WT)}$	$\overline{WAIT}$ Setup Time to Falling Edge of Clock	70		70		nsec
$\overline{BUSRQ}$	$t_{D(BQ)}$	$\overline{BUSRQ}$ Delay Time from Rising Edge of Clock		100		100	nsec
	$t_{F(C)}$	Delay to Float ($\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$ and $\overline{WR}$)		100		80	nsec

NOTES:

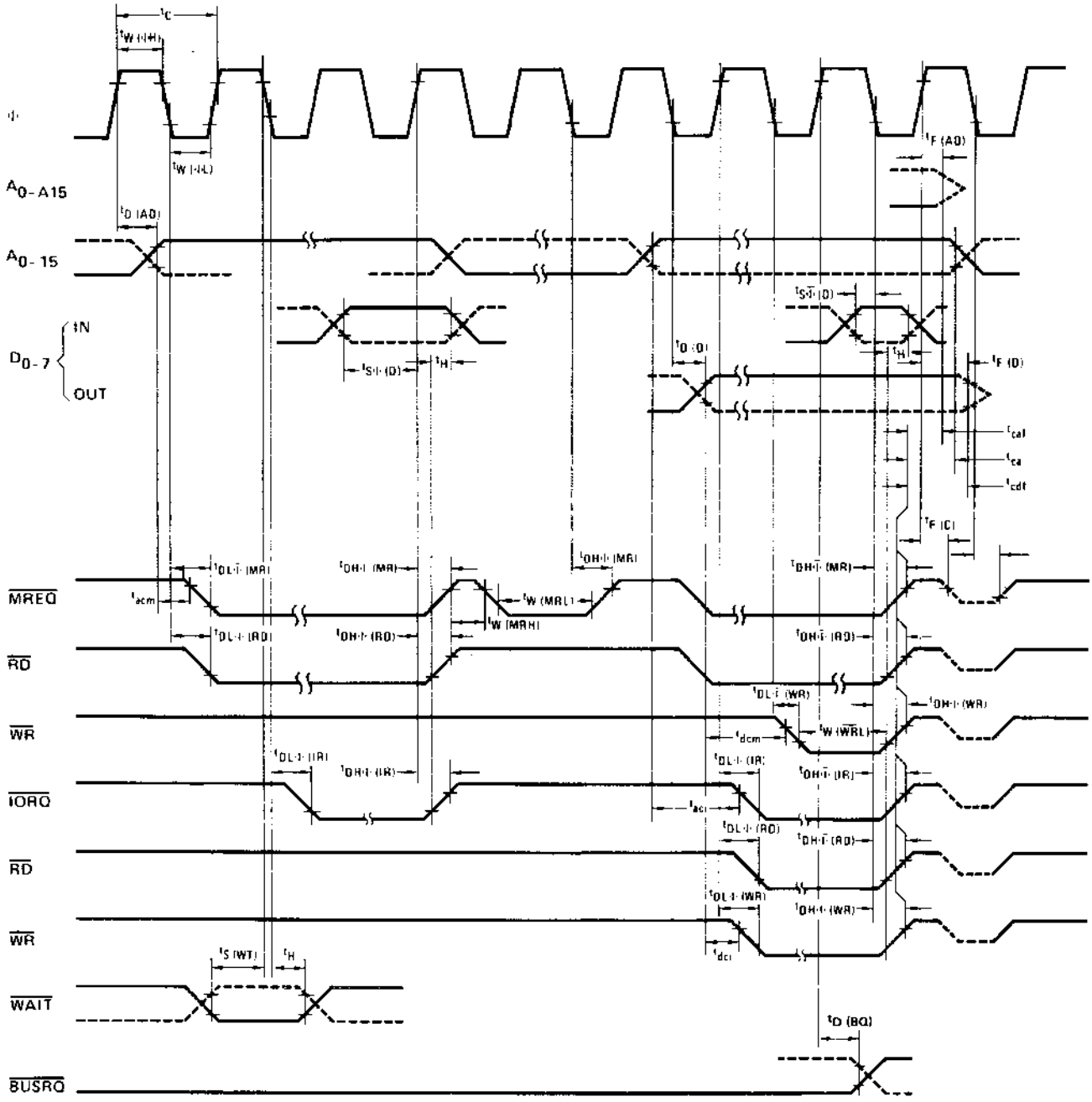
- Data must be enabled onto the DMA data bus when $\overline{RD}$ is active.
- All equations imply standard Z-80 CPU and Z-80A CPU timing.
- Z-80A DMA timing specifications are preliminary.

A.C. Timing Diagrams

Z80 and Z80A as a Bus Controller (Active State)

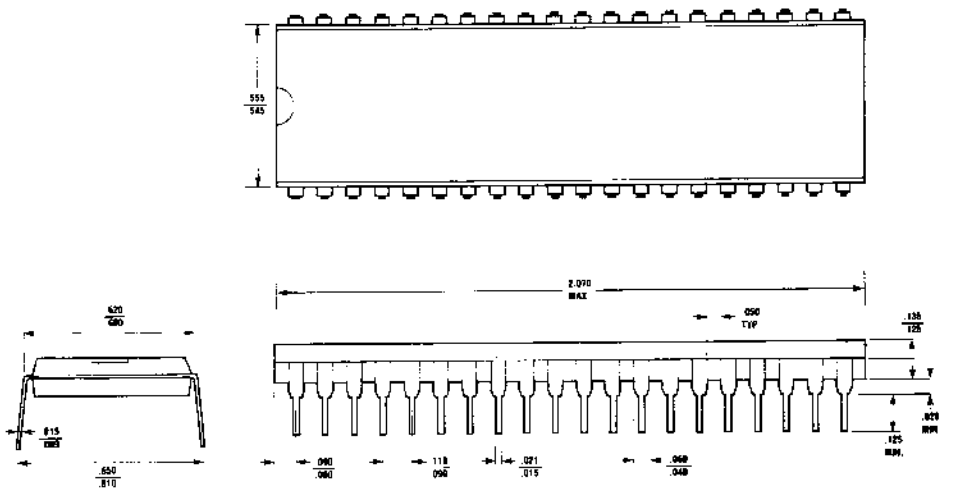
Timing measurements are made at the following voltages, unless otherwise specified:

	"1"	"0"
CLOCK	4.2V	0.8V
OUTPUT	2.0V	0.8V
INPUT	2.0V	0.8V
FLOAT	ΔV	= +0.5V

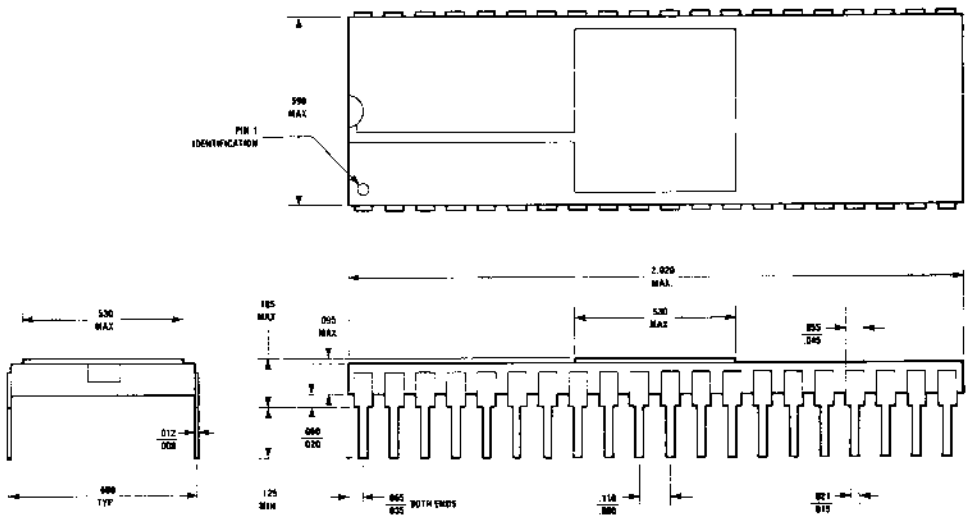


This diagram does not show an actual timing sequence. Refer to this diagram only for the detailing timing relationships of individual edges.

Package Outline



40-Pin Plastic



40-Pin Ceramic

Ordering Information

Examples:

Z-80 DMA CS (ceramic package, standard range)
Z-80 DMA PS (plastic package, standard range)

C — Ceramic Package
P — Plastic Package

S — Standard Range (5V, ±5%, 0°C to 70°C)
E — Extended Range (5V, ±5%, −40°C to 85°C)
M — Military Range (5V, ±10%, −55°C to 125°C)

Zilog

Product Specification

Z-80® SIO
Z-80A SIO

General Description

The Z80-SIO (Serial Input/Output) is a dual-channel multi-function peripheral component designed to satisfy a wide variety of serial data communications requirements in microcomputer systems. Its basic function is a serial-to-parallel, parallel-to-serial converter/controller, but—within that role—it is configurable by systems software so its "personality" can be optimized for a given serial data communications application.

The Z80-SIO is capable of handling asynchronous formats, synchronous byte-oriented protocols such as IBM Bisync, and synchronous bit-oriented protocols such as HDLC and SDLC. This versatile device can also be used to support virtually any other serial protocol for applications other than data communications (cassette or floppy disk interfaces, for example).

The Z80-SIO can generate and check CRC codes in any synchronous mode and can be programmed to check data integrity in various modes. The device also has facilities for modem controls in both channels. In applications where these controls are not needed, the modem controls can be used for general-purpose I/O.

Structure

- N-channel silicon-gate depletion-load technology
- 40-pin DIP
- Single 5 V power supply
- Single-phase 5 V clock
- All inputs and outputs TTL compatible

Features

- Two independent full-duplex channels
- Data rates in synchronous or isosynchronous modes:
  - 0–500K bits/second with 2.5 MHz system clock rate
  - 0–800K bits/second with 4.0 MHz system clock rate
- Receiver data registers quadruply buffered; transmitter doubly buffered.
- Asynchronous features:
  - 5, 6, 7 or 8 bits/character

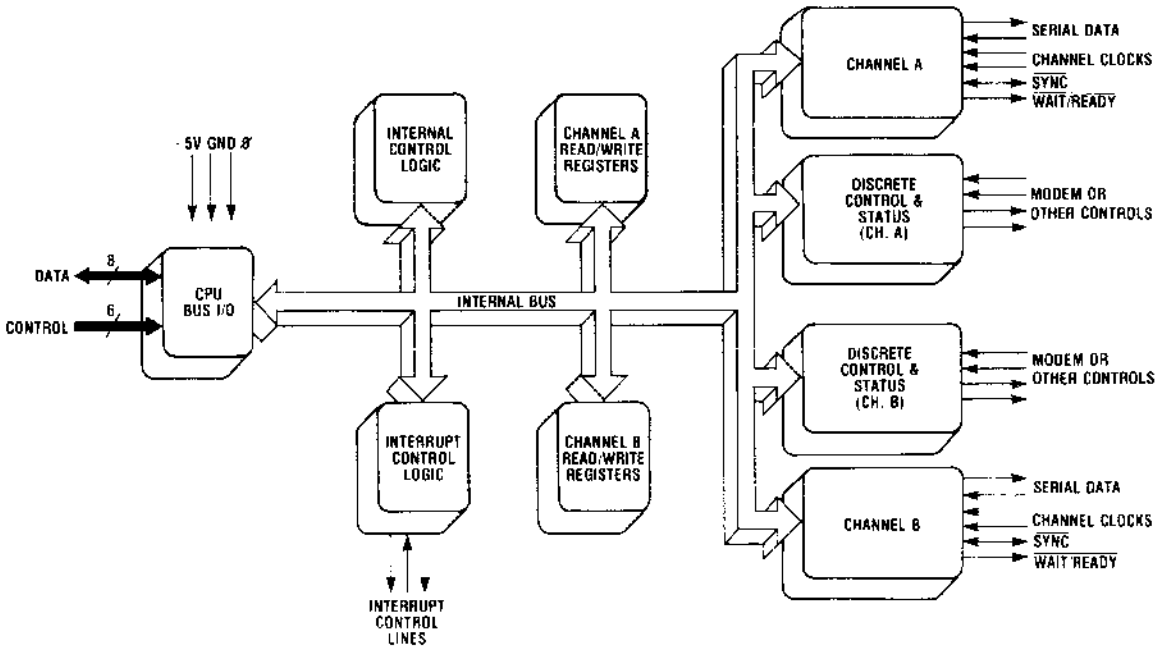


Figure 1. Z80-SIO Block Diagram

Z-80 SIO Pin Description

- 1, 1½ or 2 stop bits
- Even, odd or no parity
- ×1, ×16, ×32 and ×64 clock modes
- Break generation and detection
- Parity, overrun and framing error detection

■ Binary synchronous features:
- Internal or external character synchronization
- One or two sync characters in separate registers
- Automatic sync character insertion/deletion
- CRC generation and checking

■ HDLC and SDLC features:
- Abort sequence generation and detection
- Automatic zero insertion and deletion
- Automatic flag insertion between messages
- Address field recognition
- Support for one to eight bits/character
- Valid receive messages protected from overrun
- CRC generation and checking

■ Interrupt features:
- Daisy-chain interrupt logic provides automatic interrupt vectoring with no external logic
- Programmable interrupt vector
- Status Affects Interrupt Vector mode for fast interrupt processing

■ CRC-16 or CRC-CCITT block frame check

■ Separate modem control inputs and outputs for both channels

■ Modem status can be monitored

Pin Description

D_0–D_7. *System Data Bus* (bidirectional, 3-state). The system data bus transfers data and commands between the CPU and the Z80-SIO. D_0 is the least significant bit.

B/$\overline{A}$. *Channel A Or B Select* (input, High selects Channel B). This input defines which channel is accessed during a data transfer between the CPU and the Z80-SIO. Address bit A_0 from the CPU is often used for the selection function.

C/$\overline{D}$. *Control Or Data Select* (input, High selects Control). This input defines the type of information transfer performed between the CPU and the Z80-SIO. A High at this input during a CPU write to the Z80-SIO causes the information on the data bus to be interpreted as a command for the channel selected by B/$\overline{A}$. A Low at C/$\overline{D}$ means that the information on the data bus is data. Address bit A_1 is often used for this function.

$\overline{CE}$. *Chip Enable* (input, active Low). A Low level at this input enables the Z80-SIO to accept command or data input from the CPU during a write cycle, or to transmit data to the CPU during a read cycle.

ϕ. *System Clock* (input). The Z80-SIO uses the standard Z80 System Clock to synchronize internal signals. This is a single-phase clock.

$\overline{M1}$. *Machine Cycle One* (input from Z80-CPU, active Low). When $\overline{M1}$ is active and $\overline{RD}$ is also active, the Z80-CPU is fetching an instruction from memory; when $\overline{M1}$ is active while $\overline{IORQ}$ is active, the Z80-SIO accepts $\overline{M1}$

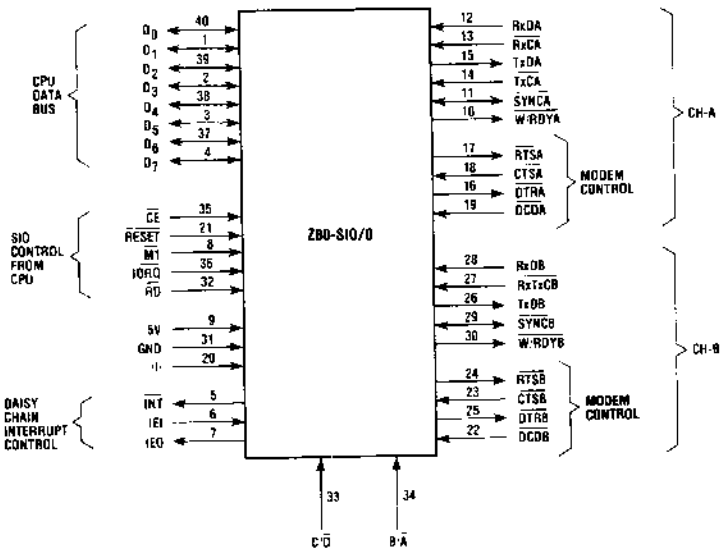


Figure 2. Z80-SIO/0 Pin Configuration

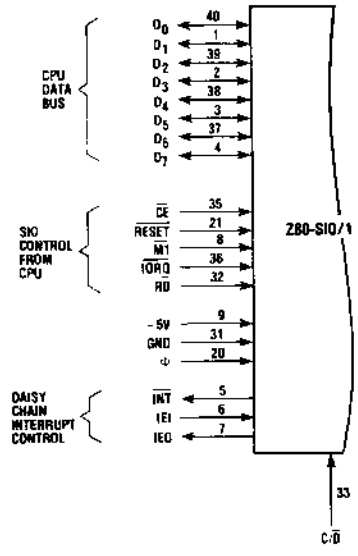


Figure 3.

Z-80 SIO Pin Description

and $\overline{IORQ}$ as an interrupt acknowledge if the Z80-SIO is the highest priority device that has interrupted the Z80-CPU.

$\overline{IORQ}$. *Input/Output Request* (input from CPU, active Low). $\overline{IORQ}$ is used in conjunction with B/$\overline{A}$, C/$\overline{D}$, $\overline{CE}$ and $\overline{RD}$ to transfer commands and data between the CPU and the Z80-SIO. When $\overline{CE}$, $\overline{RD}$ and $\overline{IORQ}$ are all active, the channel selected by B/$\overline{A}$ transfers data to the CPU (a read operation). When $\overline{CE}$ and $\overline{IORQ}$ are active, but $\overline{RD}$ is inactive, the channel selected by B/$\overline{A}$ is written to by the CPU with either data or control information as specified by C/$\overline{D}$. As mentioned previously, if $\overline{IORQ}$ and $\overline{M1}$ are active simultaneously, the CPU is acknowledging an interrupt and the Z80-SIO automatically places its interrupt vector on the CPU data bus if it is the highest priority device requesting an interrupt.

$\overline{RD}$. *Read Cycle Status.* (input from CPU, active Low). If $\overline{RD}$ is active, a memory or I/O read operation is in progress. $\overline{RD}$ is used with B/$\overline{A}$, $\overline{CE}$ and $\overline{IORQ}$ to transfer data from the Z80-SIO to the CPU.

$\overline{RESET}$. *Reset* (input, active Low). A Low $\overline{RESET}$ disables both receivers and transmitters, forces TxDA and TxDB marking, forces the modem controls High and disables all interrupts. The control registers must be rewritten after the Z80-SIO is reset and before data is transmitted or received.

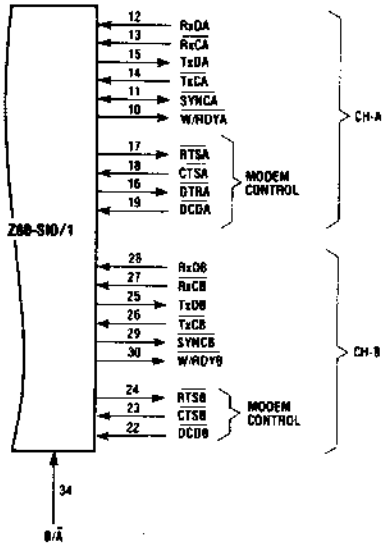
IEI. *Interrupt Enable In* (input, active High). This signal is used with IEO to form a priority daisy chain when there is more than one interrupt-driven device. A High on this line indicates that no other device of higher priority is being serviced by a CPU interrupt service routine.

IEO. *Interrupt Enable Out* (output, active High). IEO is High only if IEI is High and the CPU is not servicing an interrupt from this Z80-SIO. Thus, this signal blocks lower priority devices from interrupting while a higher priority device is being serviced by its CPU interrupt service routine.

$\overline{INT}$. *Interrupt Request* (output, open drain, active Low). When the Z80-SIO is requesting an interrupt, it pulls $\overline{INT}$ Low.

$\overline{W/RDYA}$, $\overline{W/RDYB}$. *Wait/Ready A, Wait/Ready B* (outputs, open drain when programmed for Wait function, driven High and Low when programmed for Ready function). These dual-purpose outputs may be programmed as Ready lines for a DMA controller or as Wait lines that synchronize the CPU to the Z80-SIO data rate. The reset state is open drain.

$\overline{CTSA}$, $\overline{CTSB}$. *Clear To Send* (inputs, active Low). When programmed as Auto Enables, a Low on these inputs enables the respective transmitter. If not programmed as Auto Enables, these inputs may be programmed as general-purpose inputs. Both inputs are Schmitt-trigger buffered to accommodate slow-risetime signals. The Z80-SIO detects pulses on these inputs and interrupts the CPU on both logic level transitions. The Schmitt-trigger buffering does not guarantee a specified noise-level margin.



Z80-SIO/1 Pin Configuration

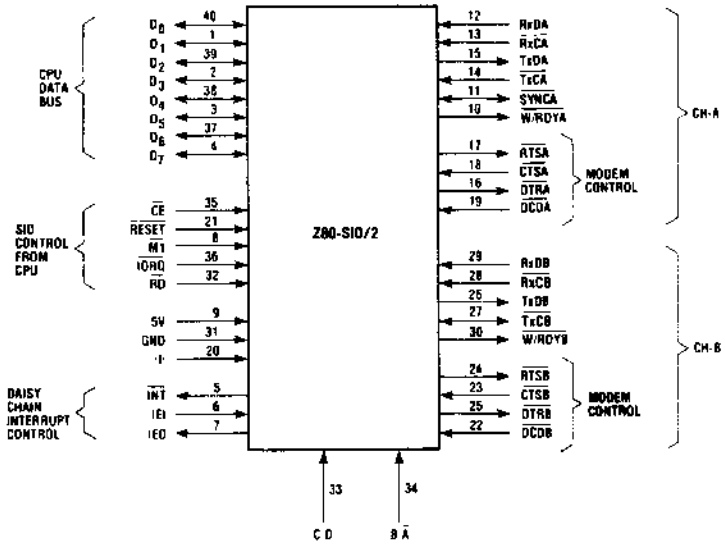


Figure 4. Z80-SIO/2 Pin Configuration

Z-80 SIO Bonding Options

$\overline{\text{DCDA}}$, $\overline{\text{DCDB}}$. *Data Carrier Detect* (inputs, active Low). These pins function as receiver enables if the Z80-SIO is programmed for Auto Enables; otherwise they may be used as general-purpose input pins. Both pins are Schmitt-trigger buffered to accommodate slow-risetime signals. The Z80-SIO detects pulses on these pins and interrupts the CPU on both logic level transitions. Schmitt-trigger buffering does not guarantee a specific noise level margin.

RxDA, RxDB. *Receive Data* (inputs, active High).

TxDA, TxDB. *Transmit Data* (outputs, active High).

$\overline{\text{RxCA}}$, $\overline{\text{RxCB}}$. *Receiver Clocks* (inputs). Receive data is sampled on the rising edge of RxC. The Receive Clocks may be 1, 16, 32 or 64 times the data rate in Asynchronous modes. These clocks may be driven by the Z80-CTC Counter Timer Circuit for programmable baud rate generation. Both inputs are Schmitt-trigger buffered (no noise level margin is specified). See the following section for bonding options.

$\overline{\text{TxCA}}$, $\overline{\text{TxCB}}$. *Transmitter Clocks* (inputs). TxD changes on the falling edge of $\overline{\text{TxC}}$. In Asynchronous modes, the Transmitter Clocks may be 1, 16, 32 or 64 times the data rate; however, the clock multiplier for the transmitter and the receiver must be the same. The Transmit Clock inputs are Schmitt-trigger buffered for relaxed rise- and fall-time requirements (no noise level margin is specified). Transmitter Clocks may be driven by the Z80-CTC Counter Timer Circuit for programmable baud rate generation. See the following section for bonding options.

$\overline{\text{RTSA}}$, $\overline{\text{RTSB}}$. *Request To Send* (outputs, active Low). When the RTS bit is set, the $\overline{\text{RTS}}$ output goes Low. When the RTS bit is reset in the Asynchronous mode, the output goes High after the transmitter is empty. In Synchronous modes, the $\overline{\text{RTS}}$ pin strictly follows the state of the RTS bit. Both pins can be used as general-purpose outputs.

$\overline{\text{DTRA}}$, $\overline{\text{DTRB}}$. *Data Terminal Ready* (outputs, active Low). See note on bonding options. These outputs follow the state programmed into the DTR bit. They can also be programmed as general-purpose outputs.

$\overline{\text{SYNC}}$ A, $\overline{\text{SYNC}}$ B. *Synchronization* (inputs/outputs, active Low). These pins can act either as inputs or outputs. In the Asynchronous Receive mode, they are inputs similar to $\overline{\text{CTS}}$ and $\overline{\text{DCD}}$. In this mode, the transitions on these lines affect the state of the Sync/Hunt status bits in RR0. In the External Sync mode, these lines also act as inputs. When external synchronization is achieved, $\overline{\text{SYNC}}$ must be driven Low on the second rising edge of $\overline{\text{RxC}}$ after that rising edge of $\overline{\text{RxC}}$ on which the last bit of the sync character was received. In other words, after the sync pattern is detected, the external logic must wait for two full Receive Clock cycles to activate the $\overline{\text{SYNC}}$ input. Once $\overline{\text{SYNC}}$ is forced Low, it is wise to keep it Low until the CPU informs the external sync logic that synchronization has been lost or a new message is about to start. Character assembly begins on the rising edge of $\overline{\text{RxC}}$ that immediately precedes the falling edge of $\overline{\text{SYNC}}$ in the External Sync mode.

In the Internal Synchronization mode (Monosync and Bisync), these pins act as outputs that are active during the part of the receive clock ($\overline{\text{RxC}}$) cycle in which sync characters are recognized. The sync condition is not latched, so these outputs are active each time a sync pattern is recognized, regardless of character boundaries.

Bonding Options

The constraints of a 40-pin package make it impossible to bring out the Receive Clock, Transmit Clock, Data Terminal Ready and Sync signals for both channels. Therefore, Channel B must sacrifice a signal or have two signals bonded together. Since user requirements vary, three bonding options are offered:

- Z80-SIO/0 has all four signals, but $\overline{\text{TxCB}}$ and $\overline{\text{RxCB}}$ are bonded together (Fig. 2).
- Z80-SIO/1 sacrifices $\overline{\text{DTRB}}$ and keeps $\overline{\text{TxCB}}$, $\overline{\text{RxCB}}$ and $\overline{\text{SYNCB}}$ (Fig. 3).
- Z80-SIO/2 sacrifices $\overline{\text{SYNCB}}$ and keeps $\overline{\text{TxCB}}$, $\overline{\text{RxCB}}$ and $\overline{\text{DTRB}}$ (Fig. 4).

Architecture

The device internal structure includes a Z80-CPU interface, internal control and interrupt logic, and two full-duplex channels. Each channel contains read and write registers, and discrete control and status logic that provides the interface to modems or other external devices.

The read and write register group includes five 8-bit control registers, two sync-character registers and two status registers. The interrupt vector is written into an additional 8-bit register (Write Register 2) in Channel B that may be read through Read Register 2 in Channel B. The registers for both channels are designated in the text as follows:

WR0–WR7 — Write Registers 0 through 7
RR0–RR2 — Read Registers 0 through 2

The bit assignment and functional grouping of each register is configured to simplify and organize the programming process. Table 1 lists the functions assigned to each read or write register.

RR0	Transmit/Receive buffer status, interrupt status and external status
RR1	Special Receive Condition status
RR2	Modified interrupt vector (Channel B only)

Read Register Functions

Z-80 SIO Architecture

WR0	Register pointers, CRC initialize, initialization commands for the various modes, etc.
WR1	Transmit/Receive interrupt and data transfer mode definition.
WR2	Interrupt vector (Channel B only)
WR3	Receive parameters and control
WR4	Transmit/Receive miscellaneous parameters and modes
WR5	Transmit parameters and controls
WR6	Sync character or SDLC address field
WR7	Sync character or SDLC flag

Write Register Functions

Table 1. Functional Assignments of Read and Write Registers

The logic for both channels provides formats, synchronization and validation for data transferred to and from the channel interface. The modem control inputs Clear to Send ($\overline{CTS}$) and Data Carrier Detect ($\overline{DCD}$) are monitored by the discrete control logic under program control. All the modem control signals are general purpose in nature and can be used for functions other than modem control.

For automatic interrupt vectoring, the interrupt control logic determines which channel and which device within the channel has the highest priority. Priority is fixed with Channel A assigned a higher priority than Channel B; Receive, Transmit and External/Status interrupts are prioritized in that order within each channel.

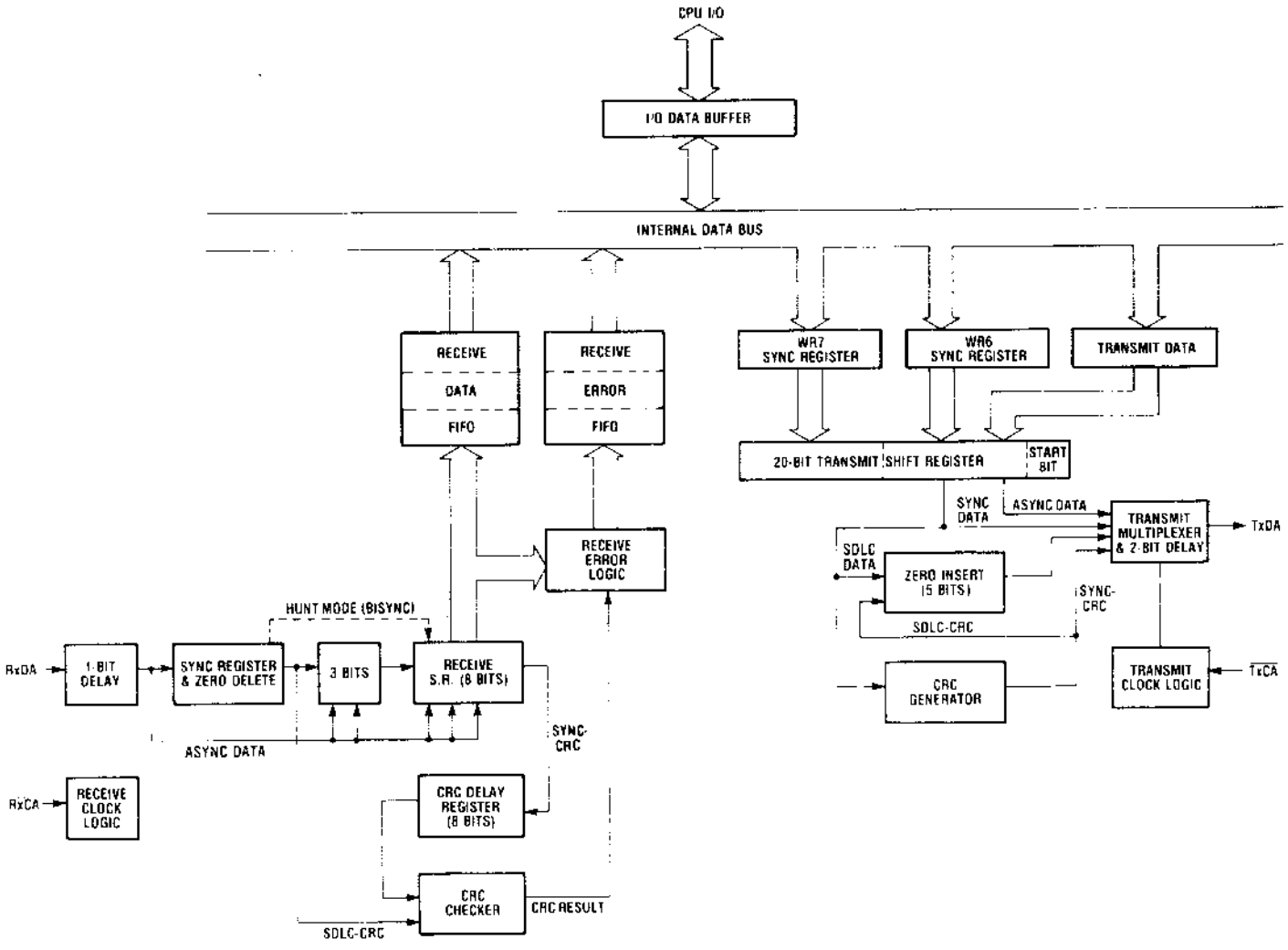


Figure 5. Transmit and Receive Data Path

Z-80 SIO Architecture

Data Path

The transmit and receive data path illustrated for Channel A in Figure 5 is identical for both channels. The receiver has three 8-bit buffer registers in a FIFO arrangement in addition to the 8-bit receive shift register. This scheme creates additional time for the CPU to service an interrupt at the beginning of a block of high-speed data. Incoming data is routed through one of several paths (data or CRC) depending on the selected mode and—in Asynchronous modes—the character length.

The transmitter has an 8-bit transmit data register that is loaded from the internal data bus, and a 20-bit transmit shift register that can be loaded from the sync character buffers (WR6 and WR7) or from the transmit data register. Depending on the operational mode, outgoing data is routed through one of four main paths before it is transmitted from the Transmit Data Output (TxD).

Functional Description

The functional capabilities of the Z80-SIO can be described from two different points of view: as a data communications device, it transmits and receives serial data, and meets the requirements of various data communications protocols; as a Z80 family peripheral, it interacts with the Z80-CPU and other Z80 peripheral circuits, and shares the data, address and control busses, as well as being a part of the Z80 interrupt structure. As a peripheral to other microprocessors, the Z80-SIO offers valuable features such as non-vectored interrupts, polling and simple handshake capability.

The first part of the following functional description describes the interaction between the CPU and Z80-SIO; the second part introduces its data communications capabilities.

I/O Interface Capabilities

The Z80-SIO offers the choice of Polling, Interrupt (vectored or non-vectored) and Block Transfer modes to transfer data, status and control information to and from the CPU. The Block Transfer mode can be implemented under CPU or DMA control.

Polling. There are no interrupts in the Polled mode. Status registers RR0 and RR1 are updated at appropriate times for each function being performed (for example, CRC Error status valid at the end of the message). All the interrupt modes of the Z80-SIO must be disabled to operate the device in a polled environment.

While in its Polling sequence, the CPU examines the status contained in RR0 for each channel; the RR0 status bits serve as an acknowledge to the Poll inquiry. The two RR0 status bits D_0 and D_2 indicate that a data transfer is needed. The status also indicates Error or other special status conditions (see "Z80-SIO Programming"). The Special Receive Condition status contained in RR1 does not have to be read in a Polling sequence because the status bits in RR1 must be accompanied by a Receive Character Available status in RR0.

Interrupts. The Z80-SIO offers an elaborate interrupt scheme to provide fast interrupt response in real-time applications. Channel B registers WR2 and RR2 contain the interrupt vector that points to an interrupt service routine in the memory. To service operations in both channels and to eliminate the necessity of writing a status analysis routine, the Z80-SIO can modify the interrupt vector in RR2 so it points directly to one of eight interrupt service routines. This is done under program control by setting a program bit (WR1, D_2) in Channel B called "Status Affects Vector." When this bit is set, the interrupt vector in WR2 is modified according to the assigned priority of the various interrupting conditions. The table in the Write Register 1 description (Z80-SIO Programming section) shows the modification details.

Transmit interrupts, Receive interrupts and External/Status interrupts are the main sources of interrupts. Each interrupt source is enabled under program control with Channel A having a higher priority than Channel B, and with Receiver, Transmit and External/Status interrupts prioritized in that order within each channel. When the Transmit interrupt is enabled, the CPU is interrupted by the transmit buffer *becoming* empty. (This implies that the transmitter must have had a data character written into it so it can become empty.) When enabled, the receiver can interrupt the CPU in one of three ways:

- Interrupt on the first received character
- Interrupt on all received characters
- Interrupt on a Special Receive condition

Interrupt On First Character is typically used with the Block Transfer mode. Interrupt On All Receive Characters has the option of modifying the interrupt vector in the event of a parity error. The Special Receive Condition interrupt can occur on a character or message basis (End Of Frame interrupt in SDLC, for example). The Special Receive condition can cause an interrupt only if the Interrupt On First Receive Character or Interrupt On All Receive Characters mode is selected. In Interrupt On First Receive Character, an interrupt can occur from Special Receive conditions (except Parity Error) after the first receive character interrupt (example: Receive Overrun interrupt).

The main function of the External/Status interrupt is to monitor the signal transitions of the $\overline{CTS}$, $\overline{DCD}$ and $\overline{SYNC}$ pins; however, an External/Status interrupt is also caused by a Transmit Underrun condition or by the detection of a Break (Asynchronous mode) or Abort (SDLC mode) sequence in the data stream. The interrupt caused by the Break/Abort sequence has a special feature that allows the Z80-SIO to interrupt when the Break/Abort sequence is detected or terminated. This feature facilitates the proper termination of the current message, correct initialization of the next message, and

Functional Description

the accurate timing of the Break/Abort condition in external logic.

CPU/DMA Block Transfer. The Z80-SIO provides a Block Transfer mode to accommodate CPU block transfer functions and DMA controllers (Z80-DMA or other designs). The Block Transfer mode uses the $\overline{\text{WAIT/READY}}$ output in conjunction with the Wait/Ready bits of Write Register 1. The $\overline{\text{WAIT/READY}}$ output can be defined under software control as a $\overline{\text{WAIT}}$ line in the CPU Block Transfer mode or as a $\overline{\text{READY}}$ line in the DMA Block Transfer mode.

To a DMA controller, the Z80-SIO READY output indicates that the Z80-SIO is ready to transfer data to or from memory. To the CPU, the WAIT output indicates that the Z80-SIO is not ready to transfer data, thereby requesting the CPU to extend the I/O cycle. The programming of bits 5, 6 and 7 of Write Register 1 and the logic states of the $\overline{\text{WAIT/READY}}$ line are defined in the Write Register 1 description (Z80-SIO Programming section).

Data Communications Capabilities

In addition to the I/O capabilities previously discussed, the Z80-SIO provides two independent full-duplex channels that can be programmed for use in Asynchronous, Synchronous and SDLC (HDLC) modes. These different modes are provided to facilitate the implementation of commonly used data communications protocols. The following is a short description of the data communications protocols supported by the Z80-SIO. A more detailed explanation of these modes can be found in the *Z80-SIO Technical Manual*.

Asynchronous Modes. The Z80-SIO offers transmission and reception of five to eight bits per character, plus optional even or odd parity. The transmitter can supply one, one and a half or two stop bits per character and can provide a break output at any time. The receiver break detection logic interrupts the CPU only at the start and end of a received break. Reception is protected from spikes by a transient spike rejection mechanism that checks the signal one-half a bit time after a Low level is detected on the Receive Data input. If the Low does not persist—as in the case of a transient—the character assembly process is not started.

Framing errors and overrun errors are detected and buffered together with the partial character on which they occurred. Vectored interrupts allow fast servicing of error conditions using dedicated routines. Furthermore, a built-in checking process avoids interpreting a framing error as a new start bit: a framing error results in the addition of one-half a bit time to the point at which the search for the next start bit is begun.

The Z80-SIO does not require symmetric Transmit and Receive Clock signals—a feature that allows it to be used with a Z80-CTC or any other clock source. The transmitter and receiver can handle data at a rate of 1, 1/16, 1/32 or 1/64 of the clock rate supplied to the Receive and Transmit Clock inputs.

In Asynchronous modes, the $\overline{\text{SYNC}}$ pin may be programmed for an input that can be used for functions such as monitoring a ring indicator.

Synchronous Modes. The Z80-SIO supports both byte-oriented and bit-oriented synchronous communication. Synchronous byte-oriented protocols can be handled in several modes that allow character synchronization with an 8-bit sync character (Monosync), any 16-bit sync pattern (Bisync), or with an external sync signal. Leading sync characters can be removed without interrupting the CPU. CRC checking for synchronous byte-oriented modes is delayed by one character time so the CPU may disable CRC checking on specific characters. This permits implementation of protocols such as IBM Bisync.

Both CRC-16 ($X^{16}+X^{15}+X^2+1$) and CCITT ($X^{16}+X^{12}+X^5+1$) error checking polynomials are supported. In all non-SDLC modes, the CRC generator is initialized to 0's; in SDLC modes, it is initialized to 1's. (This means that the Z80-SIO cannot generate or check CRC for IBM-compatible soft-sectored disks.) The Z80-SIO also provides a feature that automatically transmits CRC data when no other data is available for transmission. This allows very high-speed transmissions under DMA control with no need for CPU intervention at the end of a message. When there is no data or CRC to send in Synchronous modes, the transmitter inserts 8- or 16-bit sync characters regardless of the programmed character length. Since the CPU can read status information from the Z80-SIO, it can determine the type of transmission (data, CRC or sync characters) that is taking place at any time.

The Z80-SIO supports synchronous bit-oriented protocols such as SDLC and HDLC by performing automatic flag sending, zero insertion and CRC generation. A special command can be used to abort a frame in transmission. The Z80-SIO automatically transmits the CRC and trailing flag when the transmit buffer becomes empty. An interrupt warns the CPU of this status change so an abort may be issued if a transmitter underrun has occurred. One to eight bits per character can be sent, which allows transmission of a message exactly as received with no prior information about the character structure in the information field of a frame.

The receiver automatically synchronizes on the leading flag of a frame and provides a synchronization signal that can be programmed to interrupt. In addition, an interrupt on the first received character or on every character can be selected. The receiver automatically deletes all zeroes inserted by the transmitter during character assembly. It also calculates and automatically checks the CRC to validate frame transmission. At the end of transmission, the status of a received frame is available in the status registers. The receiver can be programmed to search for frames addressed to only a specified user-selectable address or to a global broadcast address. In this mode, frames that do not match the user-

selected or broadcast address are ignored. The Address Search mode provides for a single-byte address recognizable by the hardware. The number of address bytes can be extended under software control.

The Z80-SIO can be conveniently used under DMA control to provide high-speed reception. The Z80-SIO can interrupt the CPU when the first character of a message is received. The CPU then enables the DMA to transfer the message to memory. The Z80-SIO then issues an End Of Frame interrupt and the CPU checks the status of the received message. Thus, the CPU is freed for other service while the message is being received. A similar scheme allows message transmission under DMA control.

Z80-SIO Programming

To program the Z80-SIO, the system program first issues a series of commands that initialize the basic mode of operation and then other commands that qualify conditions within the selected mode. For example, the Asynchronous mode, character length, clock rate, number of stop bits, even or odd parity are first set, then the interrupt mode and, finally, receiver or transmitter enable. The WR4 parameters must be issued before any other parameters are issued in the initialization routine.

Both channels contain command registers that must be programmed via the system program prior to operation. The Channel Select input ($B/\overline{A}$) and the Control/Data input ($C/\overline{D}$) are the command structure addressing controls, and are normally controlled by the CPU address bus. Figure 8 illustrates the timing relationships for programming the write registers, and transferring data and status.

Write Registers

The Z80-SIO contains eight registers (WR0-WR7) in each channel that are programmed separately by the system program to configure the functional personality of the channels. With the exception of WR0, programming the write registers requires two bytes. The first byte contains three bits (D_0-D_2) that point to the selected register; the second byte is the actual control word that is written into the register to configure the Z80-SIO.

WR0 is a special case in that all the basic commands (CMD_0-CMD_2) can be accessed with a single byte. Reset (internal or external) initializes the pointer bits D_0-D_2 to point to WR0.

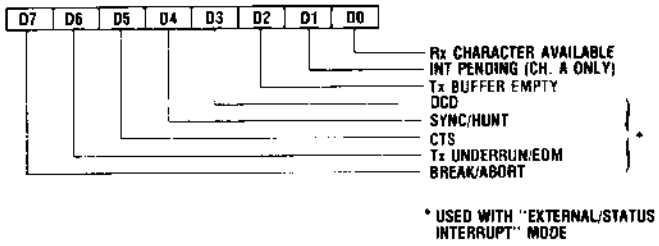
Read Registers

The Z80-SIO contains three registers, RR0-RR2 (Figure 6), that can be read to obtain the status information for each channel (except for RR2 — Channel B only). The status information includes error conditions, interrupt vector and standard communications-interface signals.

To read the contents of a selected read register other than RR0, the system program must first write the pointer byte to WR0 in exactly the same way as a write register operation. Then, by executing an input instruction, the contents of the addressed read register can be read by the CPU.

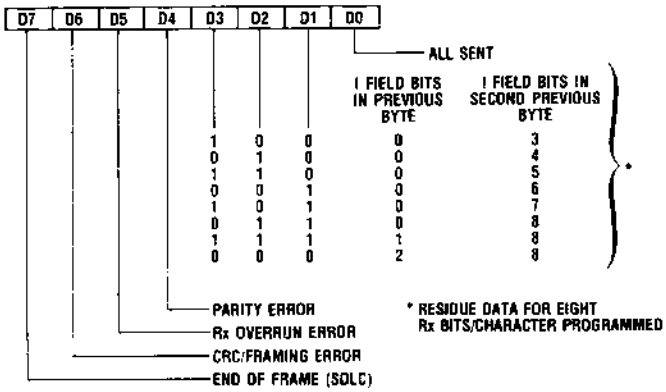
The status bits of RR0 and RR1 are carefully grouped to simplify status monitoring. For example, when the interrupt vector indicates that a Special Receive Condition interrupt has occurred, all the appropriate error bits can be read from a single register (RR1).

READ REGISTER 0



* USED WITH "EXTERNAL/STATUS INTERRUPT" MODE

READ REGISTER 1†



* RESIDUE DATA FOR EIGHT Rx BITS/CHARACTER PROGRAMMED

† USED WITH SPECIAL RECEIVE CONDITION MODE

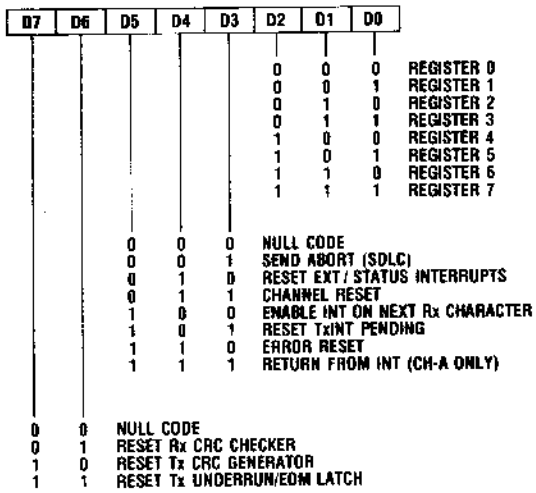
READ REGISTER 2

*VARIABLE IF "STATUS AFFECTS VECTOR" IS PROGRAMMED

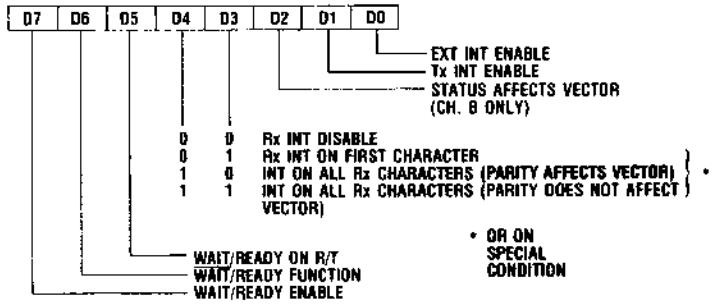
Figure 6. Read Register Bit Functions

Z-80 SIO Programming

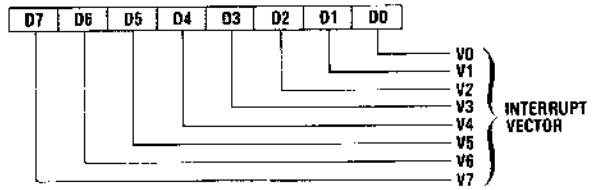
WRITE REGISTER 0



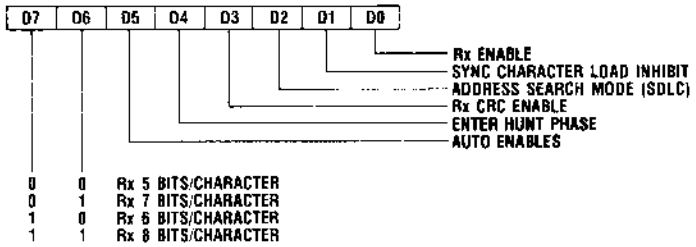
WRITE REGISTER 1



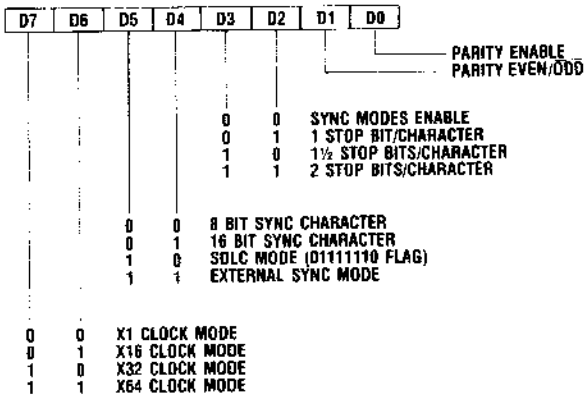
WRITE REGISTER 2 (CHANNEL B ONLY)



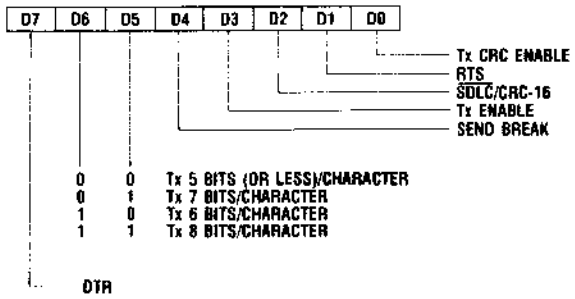
WRITE REGISTER 3



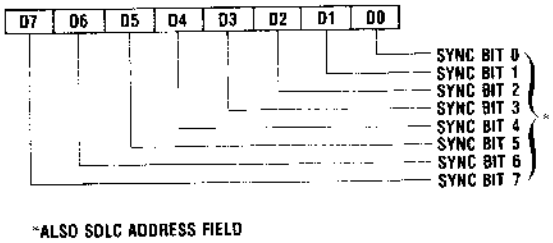
WRITE REGISTER 4



WRITE REGISTER 5

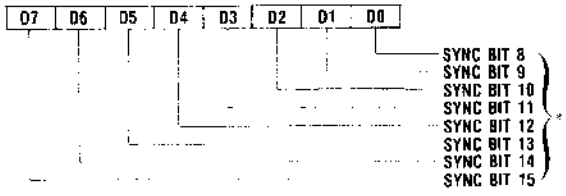


WRITE REGISTER 6



*ALSO SDLC ADDRESS FIELD

WRITE REGISTER 7



*FOR SDLC IT MUST BE PROGRAMMED TO "01111110" FOR FLAG RECOGNITION

Figure 7. Write Register Bit Functions

Z-80 SIO Timing

Timing

Read Cycle. The timing signals generated by a Z80-CPU input instruction to read a Data or Status byte from the Z80-SIO are illustrated in Figure 8a.

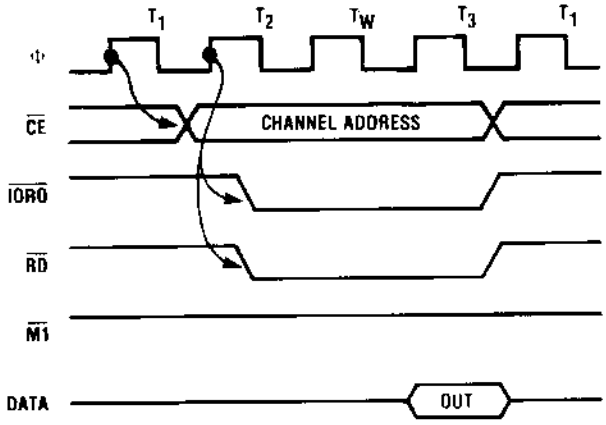


Figure 8a. Read Cycle

Write Cycle. Figure 8b illustrates the timing and data signals generated by a Z80-CPU output instruction to write a Data or Control byte into the Z80-SIO.

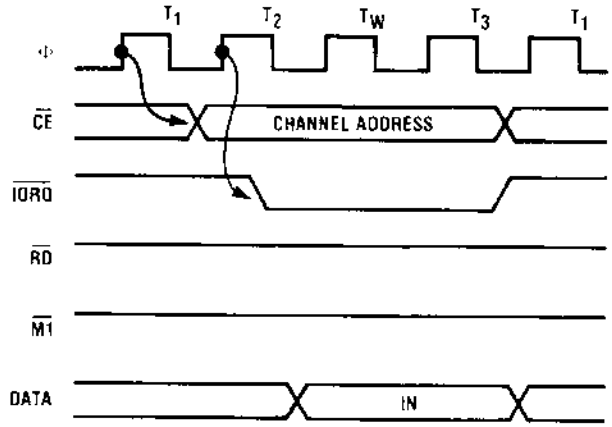


Figure 8b. Write Cycle

Interrupt Acknowledge Cycle. After receiving an Interrupt Request signal ($\overline{INT}$ pulled Low), the Z80-CPU sends an Interrupt Acknowledge signal ($\overline{M1}$ and $\overline{IORQ}$ both Low). The daisy-chained interrupt circuits determine the highest priority interrupt requestor. The IEI of the highest priority peripheral is terminated High. For any peripheral that has no interrupt pending or under service, IEO=IEI. Any peripheral that does have an interrupt pending or under service forces its IEO Low.

To insure stable conditions in the daisy chain, all interrupt status signals are prevented from changing while $\overline{M1}$ is Low. When $\overline{IORQ}$ is Low, the highest priority interrupt requestor (the one with IEI High) places its interrupt vector on the data bus and sets its internal interrupt-under-service latch.

Return From Interrupt Cycle. Normally, the Z80-CPU issues a RETI (RETurn from Interrupt) instruction at the end of an interrupt service routine. RETI is a 2-byte opcode (ED-4D) that resets the interrupt-under-service latch to terminate the interrupt that has just been processed. This is accomplished by manipulating the daisy chain in the following way.

The normal daisy chain operation can be used to detect a pending interrupt; however, it cannot distinguish between an interrupt under service and a pending unacknowledged interrupt of a higher priority. Whenever "ED" is decoded, the daisy chain is modified by forcing High the IEO of any interrupt that has not yet been acknowledged. Thus the daisy chain identifies the device presently under service as the only one with an IEI

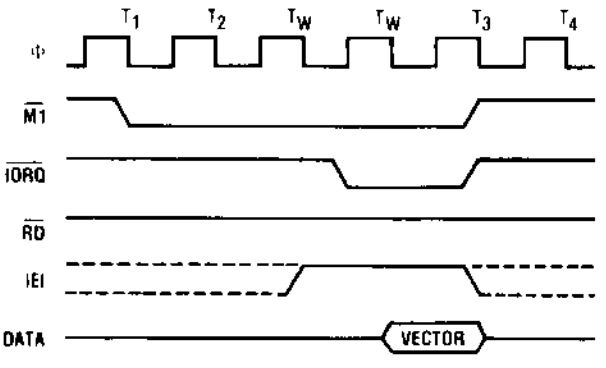


Figure 8c. Interrupt Acknowledge Cycle

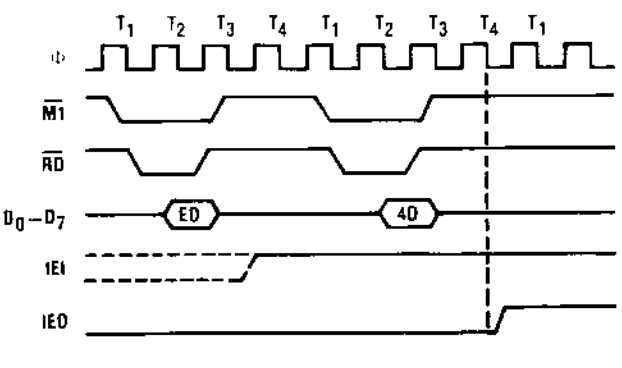


Figure 8d. Return from Interrupt Cycle

High and an IEO Low. If the next opcode byte is "4D," the interrupt-under-service latch is reset.

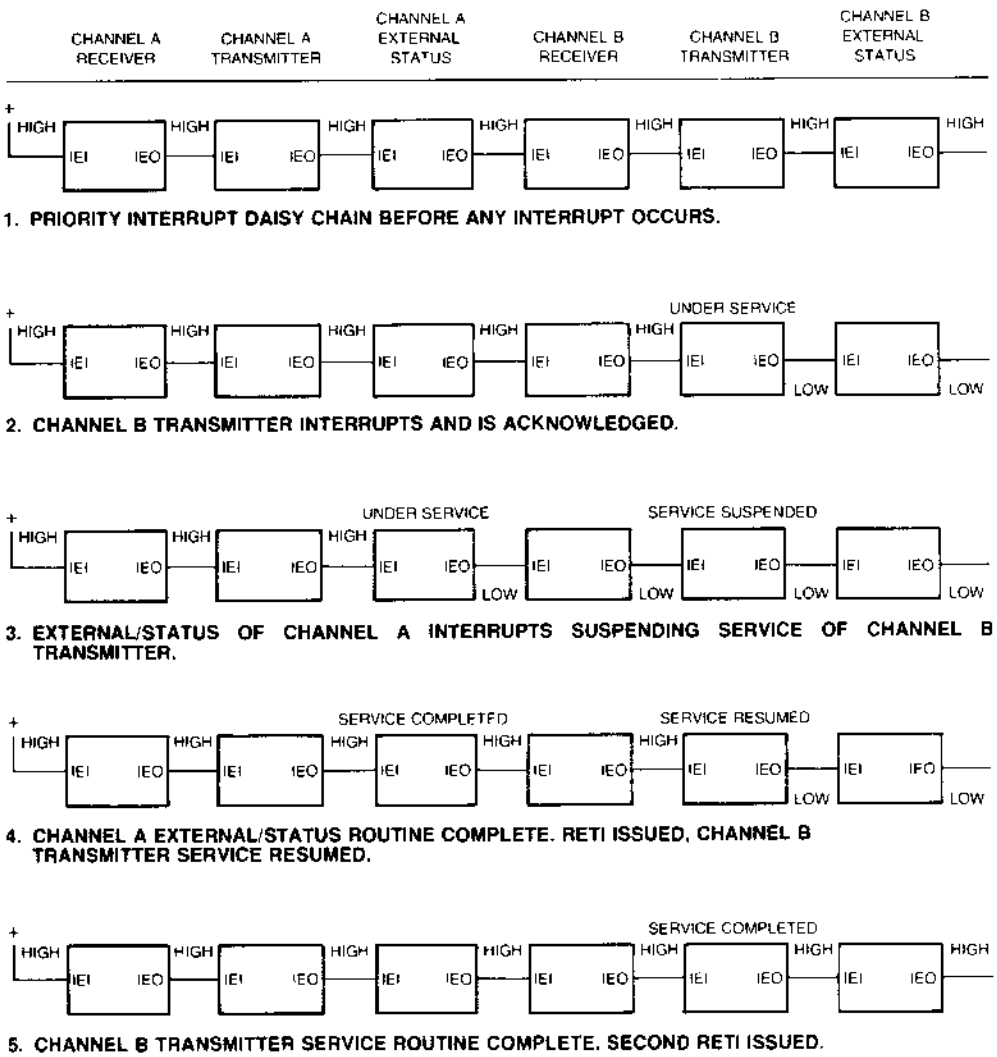
The ripple time of the interrupt daisy chain (both the High-to-Low and the Low-to-High transitions) limits the number of devices that can be placed in the daisy chain. Ripple time can be improved with carry-look-ahead, or by extending the interrupt acknowledge cycle. For further information about techniques for increasing the number of daisy-chained devices, refer to Zilog Application Note 03-0041-01 (*The Z80 Family Program Interrupt Structure*).

Daisy Chain Interrupt Nesting

Figure 9 illustrates the daisy chain configuration of interrupt circuits and their behavior with nested interrupts (an interrupt that is interrupted by another with a higher priority).

Each box in the illustration could be a separate external Z80 peripheral circuit with a user-defined order of interrupt priorities. However, a similar daisy chain structure also exists inside the Z80-SIO, which has six interrupt levels with a fixed order of priorities.

The case illustrated occurs when the transmitter of Channel B interrupts and is granted service. While this interrupt is being serviced, it is interrupted by a higher priority interrupt from Channel A. The second interrupt is serviced and—upon completion—a RETI instruction is executed or a RETI command is written into the Z80-SIO, resetting the interrupt-under-service latch of the Channel A interrupt. At this time, the service rotine for Channel B is resumed. When it is completed, another RETI instruction is executed to complete the interrupt service.



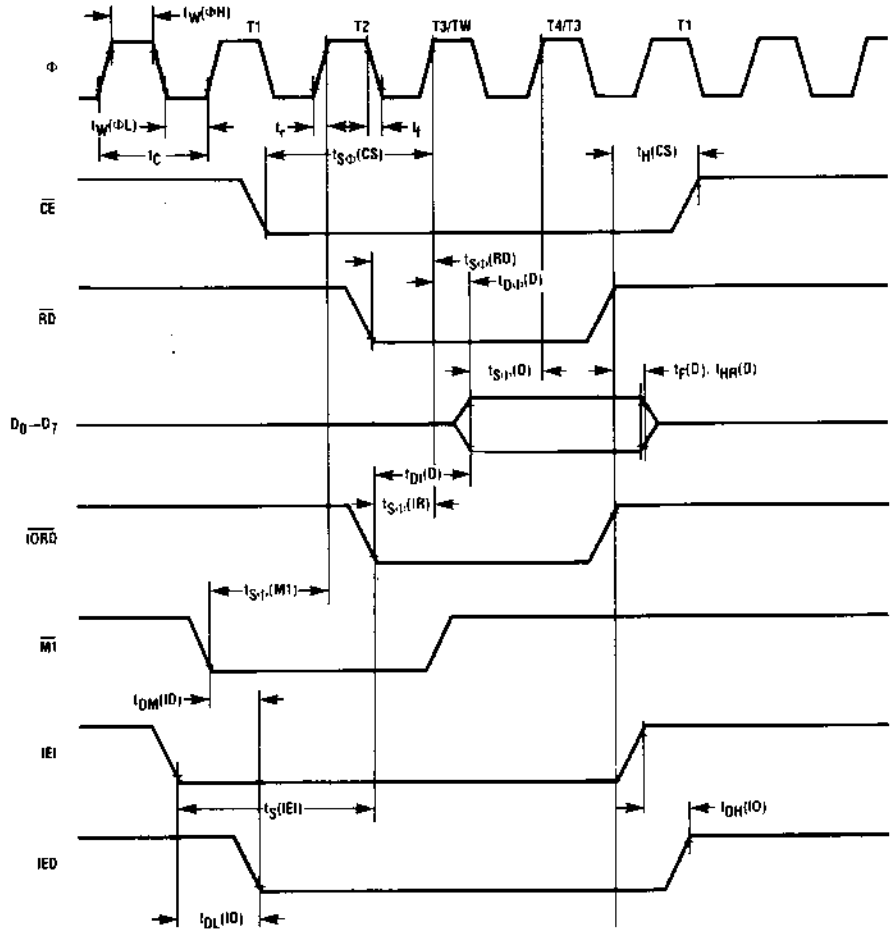
1. PRIORITY INTERRUPT DAISY CHAIN BEFORE ANY INTERRUPT OCCURS.
2. CHANNEL B TRANSMITTER INTERRUPTS AND IS ACKNOWLEDGED.
3. EXTERNAL/STATUS OF CHANNEL A INTERRUPTS SUSPENDING SERVICE OF CHANNEL B TRANSMITTER.
4. CHANNEL A EXTERNAL/STATUS ROUTINE COMPLETE. RETI ISSUED, CHANNEL B TRANSMITTER SERVICE RESUMED.
5. CHANNEL B TRANSMITTER SERVICE ROUTINE COMPLETE. SECOND RETI ISSUED.

Figure 9. Typical Interrupt Sequence

Electrical Characteristics

AC Characteristics

$T_A = 0°C$, $V_{CC} = +5V, \pm 5\%$



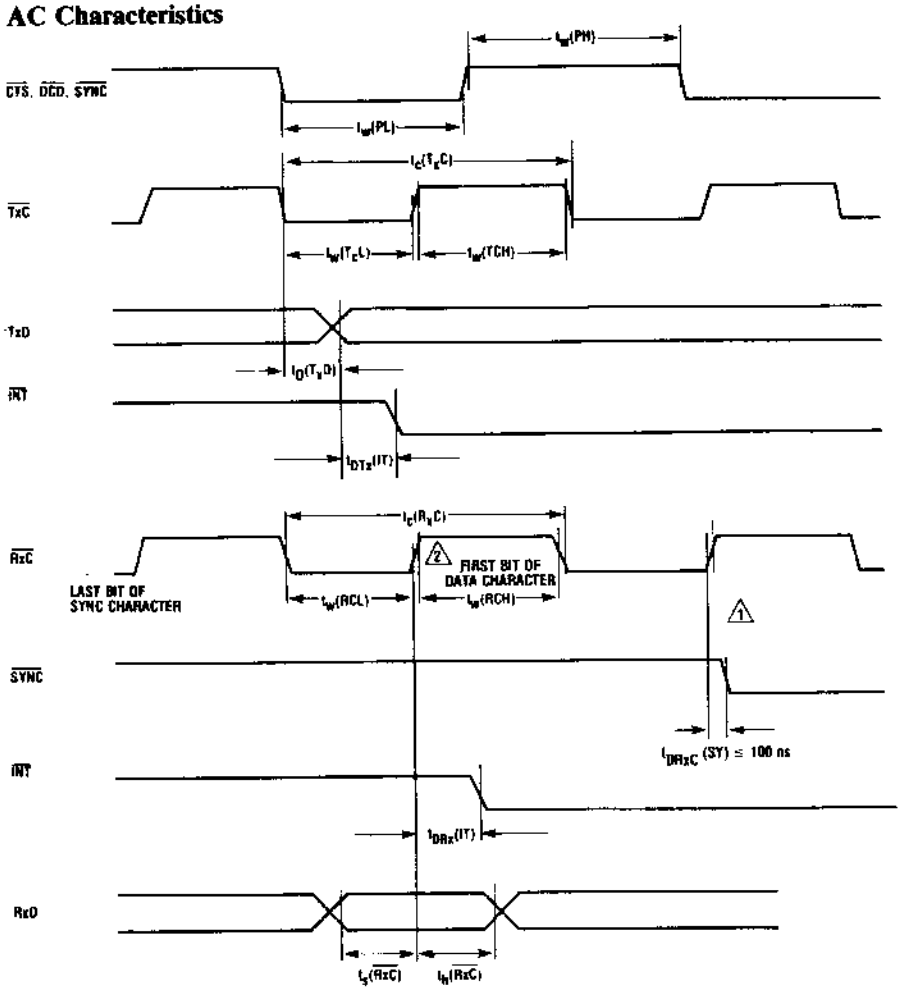
Signal	Symbol	Parameter	Z80-SIO Min	Z80-SIO Max	Z80A-SIO Min	Z80A-SIO Max	Unit
Φ	$t_c(\phi)$	Clock Period	400	4000	250	4000	ns
	$t_w(\phi H)$	Clock Pulse Width, clock HIGH	170	2000	105	2000	ns
	$t_w(\phi L)$	Clock Pulse Width, clock LOW	170	2000	105	2000	ns
	t_r, t_f	Clock Rise and Fall Times	0	30	0	30	ns
	t_H	Any Unspecified Hold Time for setup times specified below	0		0		ns
$\overline{CE}$, B/$\overline{A}$ C/$\overline{D}$, $\overline{IORQ}$	$t_{s(h)}(CS)$	Control Signal Setup Time to rising edge of ϕ during Read or Write Cycle	160		145		ns
D_0-D_7	$t_{d(\phi)}(D)$	Data Output Delay from rising edge of ϕ during Read Cycle		240		220	ns
	$t_{s(\phi)}(D)$	Data Setup Time to rising edge of ϕ during Write or M1 Cycle	50		50		ns
	$t_{d(f)}(D)$	Data Output Delay from falling edge of $\overline{IORQ}$ during INTA Cycle		340		160	ns
	$t_z(D)$	Delay to Floating Bus (output buffer disable time)		230		110	ns
IEI	$t_s(IEI)$	IEI Setup Time to falling edge of $\overline{IORQ}$ during INTA Cycle	200		140		ns
IEO	$t_{d(r)}(IO)$	IEO Delay Time from rising edge of IEI (after ED decode)		150		100	ns
	$t_{d(f)}(IO)$	IEO Delay Time from falling edge of IEI		150		100	ns
	$t_{d(M)}(IO)$	IEO Delay Time from falling edge of $\overline{M1}$ (interrupt occurring just prior to M1)		300		190	ns
M1	$t_{s(f)}(M1)$	$\overline{M1}$ Setup Time to rising edge of ϕ during INTA or M1 Cycle	210		90		ns
$\overline{RD}$	$t_{s(r)}(RD)$	$\overline{RD}$ Setup Time to rising edge of ϕ during Read or M1 Cycle	240		115		ns

*If WAIT from the SIO is to be used, $\overline{CE}$, $\overline{IORQ}$, C/$\overline{D}$ and $\overline{M1}$ must be valid for as long as the Wait condition is to persist.

Figure 9. Typical Interrupt Sequence

Electrical Characteristics

AC Characteristics



NOTES:

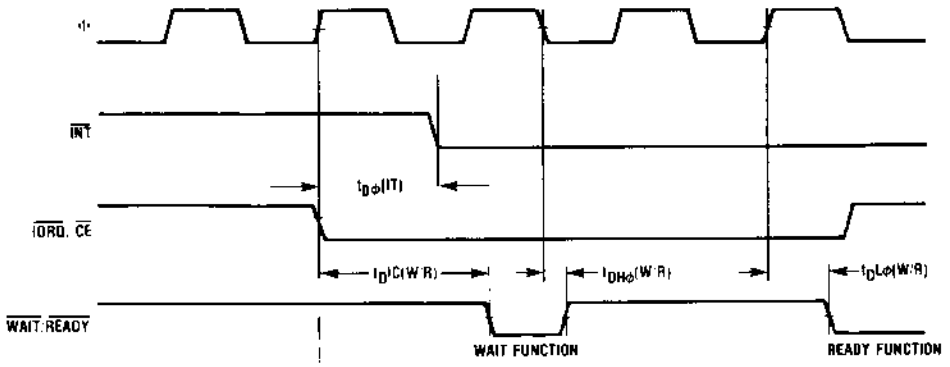
1. The $\overline{SYNC}$ input must be driven Low on the rising edge of $\overline{RxC}$ delayed two complete clock cycles from the last bit of the sync character.
2. Data character assembly begins on the next Receive Clock cycle after the last bit of the sync character is received.

Signal	Symbol	Parameter	Z80-SIO Min	Z80-SIO Max	Z80A-SIO Min	Z80A-SIO Max	Unit
$\overline{INT}$	t_{DIr}(IT)	$\overline{INT}$ Delay Time from rising edge of $\overline{RxC}$	10	13	10	13	φ periods
	t_{DIf}(IT)	$\overline{INT}$ Delay Time from transition of Xmit Data Bit	5	9	5	9	φ periods
$\overline{CTSA}$, $\overline{CTSB}$, $\overline{DCDA}$, $\overline{DCDB}$, $\overline{SYNCA}$, $\overline{SYNCB}$	t_w(PH)	Minimum HIGH Pulse Width for latching state into register and generating interrupt	200		200		ns
	t_w(PL)	Minimum LOW Pulse Width for latching state into register and generating interrupt	200		200		ns
$\overline{SYNCA}$, $\overline{SYNCB}$	t_{dx}(SY)	Sync Pulse Delay Time from rising edge of $\overline{RxC}$, Output Modes	4	7	4	7	φ periods
	t_{dRxC}(SY)	Sync Pulse Delay Time from rising edge of $\overline{RxC}$ External Sync Mode		100		100	ns
$\overline{TxCA}$, $\overline{TxCB}$	t_c(TxC)	Transmit Clock Period	400	*	400	*	ns
	t_w(TCH)	Transmit Clock Pulse Width, clock HIGH	180	*	180	*	ns
	t_w(TCL)	Transmit Clock Pulse Width, clock LOW	180	*	180	*	ns
TxDA, TxDB*	t_d(TxD)	TxD Output Delay from falling Edge of $\overline{TxC}$ (x1 Clock Mode)		400		300	ns
$\overline{RxCA}$, $\overline{RxCB}$	t_c(RxC)	Receive Clock Period	400	*	400	*	ns
	t_w(RCH)	Receive Clock Pulse Width, clock HIGH	180	*	180	*	ns
	t_w(RCL)	Receive Clock Pulse Width, clock LOW	180	*	180	*	ns
RxDA, RxDB*	t_s(RxC)	Setup Time to rising edge of RxC, x1 mode	0		0		ns
	t_h(RxC)	Hold Time from rising edge of RxC, x1 mode	140		140		ns

*In all modes, the system clock (φ) rate must be at least 4.5 times the maximum data rate. $\overline{RESET}$ must be active a minimum of one complete φ cycle.

Electrical Characteristics

AC Characteristics



Signal	Symbol	Parameter	Z80-SIO Min	Z80-SIO Max	Z80A-SIO Min	Z80A-SIO Max	Unit
$\overline{INT}$	$t_{\phi}\phi(IT)$	$\overline{INT}$ Delay Time from rising edge of ϕ		200		200	ns
	$t_{D}IC(W/R)$	$\overline{WAIT/READY}$ Delay Time from $\overline{IORQ}$ or $\overline{CE}$ in Wait Mode		300		210	ns
	$t_{D}H\phi(W/R)$	$\overline{WAIT/READY}$ Delay Time from falling edge of ϕ, $\overline{WAIT/READY}$ HIGH, Wait Mode		150		130	ns
$\overline{WAIT/READY}$	$t_{D}Rx(W/R)$	$\overline{WAIT/READY}$ Delay Time from rising edge of $\overline{RxC}$ Data Bit, Ready Mode	10	13	10	13	ϕ periods
	$t_{D}Tx(W/R)$	$\overline{WAIT/READY}$ Delay Time from center of Transmit Data Bit, Ready Mode	5	9	5	9	ϕ periods
	$t_{D}L\phi(W/R)$	$\overline{WAIT/READY}$ Delay Time from rising edge of ϕ, $\overline{WAIT/READY}$ LOW, Ready Mode		120		120	ns

DC Characteristics

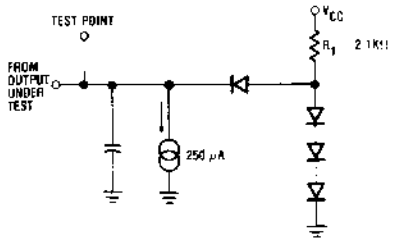
$T_A = 0°C$ to $70°C$, $V_{CC} = +5V, \pm 5\%$

Symbol	Parameter	Min.	Max.	Unit	Test Condition
V_{ILC}	Clock Input Low Voltage	−0.3	+0.45	V	
V_{IHC}	Clock Input High Voltage	$V_{CC}-0.6$	+5.5	V	
V_{IL}	Input Low Voltage	−0.3	+0.8	V	
V_{IH}	Input High Voltage	+2.0	+5.5	V	
V_{OL}	Output Low Voltage		+0.4	V	$I_{OL} = 2.0$ mA
V_{OH}	Output High Voltage	+2.4		V	$I_{OH} = -250\ \mu A$
I_{LI}	Input Leakage Current	−10	+10	μA	$0 \leq V_{IN} \leq V_{CC}$
I_{LD}	3-State Output/Data Bus Input Leakage Current	−10	+10	μA	$0 \leq V_{IN} \leq V_{CC}$
I_{LSY}	$\overline{SYNC}$ Pin Leakage Current	−40	+10	μA	$0 \leq V_{IN} \leq V_{CC}$
I_{CC}	Power Supply Current		100	mA	

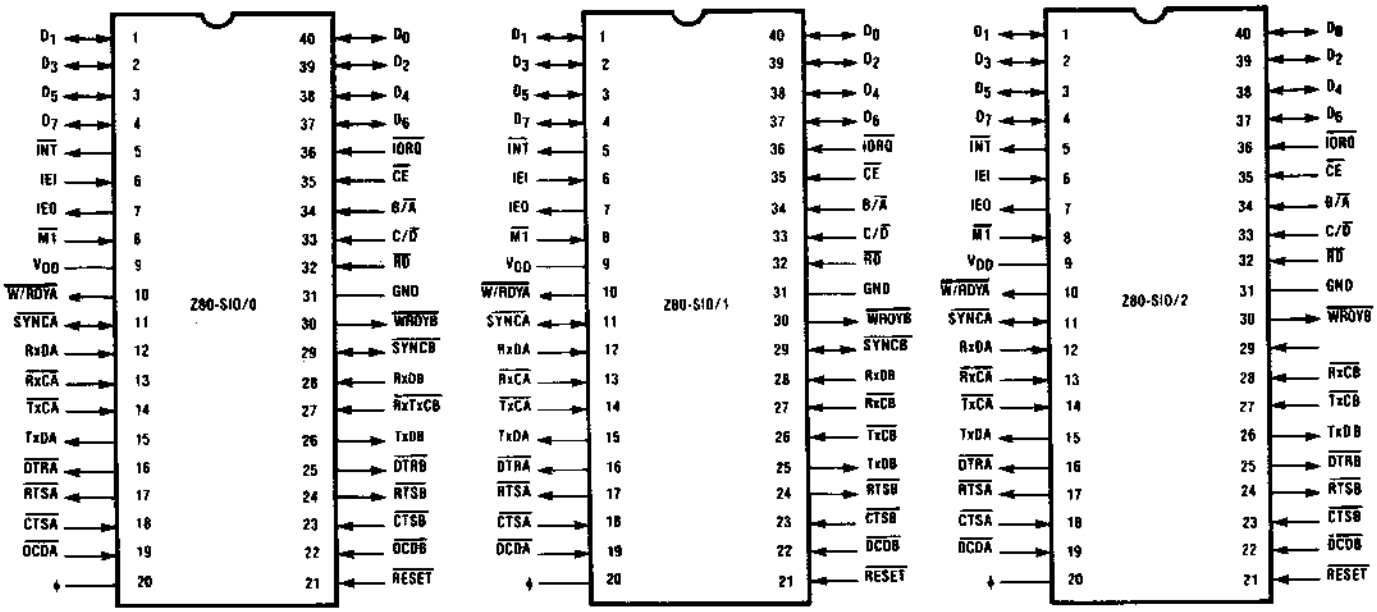
Capacitance

$T_A = 25°C$, $f = 1$ MHz

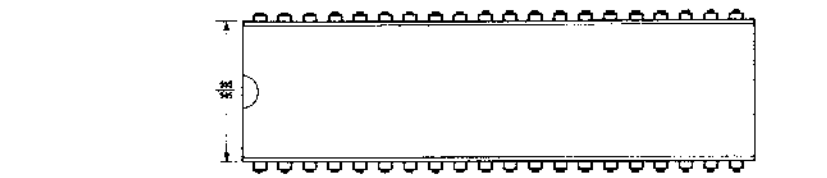
Symbol	Parameter	Min.	Max.	Unit	Test Condition
C	Clock Capacitance		40	pF	Unmeasured pins returned to ground
C_{IN}	Input Capacitance		5	pF	
C_{OUT}	Output Capacitance		10	pF	



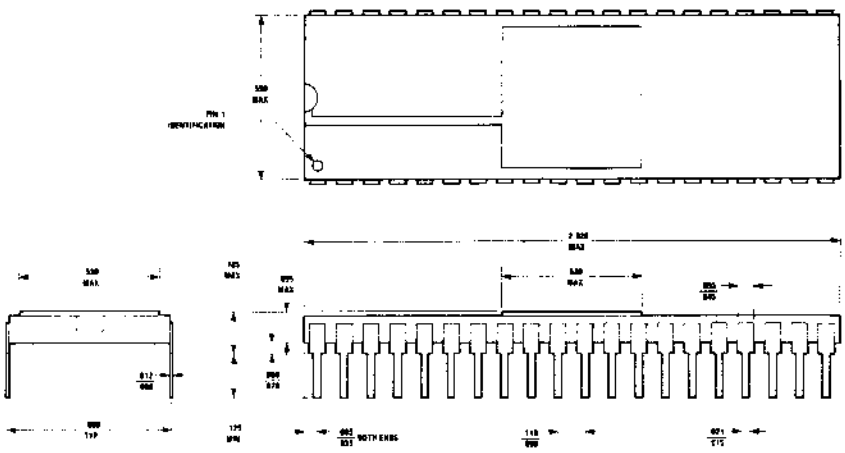
Package Information



Package Outlines



40-Pin Plastic



40-Pin Ceramic

Ordering Information

Bonding, Package and Temperature Identification

/0 — Type 0 Bonding
/1 — Type 1 Bonding
/2 — Type 2 Bonding

C — Ceramic Package
P — Plastic Package

S — Standard Range (5V, ±5%, 0°C to 70°C)
E — Extended Range (5V, ±5%, −40°C to 85°C)
M — Military Range (5V, ±10%, −55°C to 125°C)

Examples:

Z80-SIO/1 CS (Type 1 bonding, ceramic package, standard range)

Z80-SIO/2 PS (Type 2 bonding, plastic package, standard range)

Zilog

Product Specification

Z-80®SIO/9
Z-80A SIO/9

General Description

The Z80-SIO (Serial Input/Output) is a single-channel multi-function peripheral component designed to satisfy a wide variety of serial data communications requirements in microcomputer systems. Its basic function is a serial-to-parallel, parallel-to-serial converter/controller, but—within that role—it is configurable by systems software so its "personality" can be optimized for a given serial data communications application.

The Z80-SIO is capable of handling asynchronous formats, synchronous byte-oriented protocols such as IBM Bisync, and synchronous bit-oriented protocols such as HDLC and SDLC. This versatile device can also be used to support virtually any other serial protocol for applications other than data communications (cassette or floppy disk interfaces, for example).

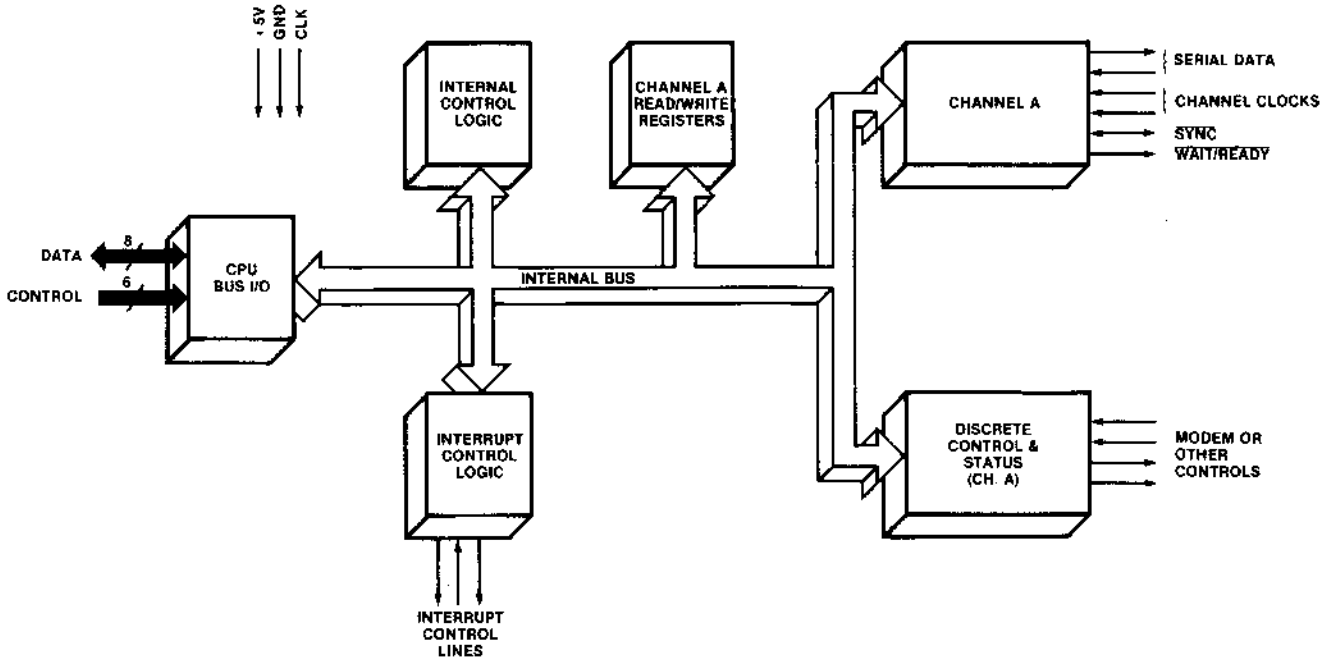
The Z80-SIO can generate and check CRC codes in any synchronous mode and can be programmed to check data integrity in various modes. The device also has facilities for modem controls in both channels. In applications where these controls are not needed, the modem controls can be used for general-purpose I/O.

Structure

- N-channel silicon-gate depletion-load technology
- 40-pin DIP
- Single 5V power supply
- Single-phase 5V clock
- All inputs and outputs TTL compatible

Features

- One full-duplex channel
- Data rates in synchronous or isosynchronous modes:
  - 0–500K bits/second with 2.5 MHz system clock rate
  - 0–800K bits/second with 4.0 MHz system clock rate
- Receiver data registers quadruply buffered; transmitter doubly buffered.
- Asynchronous features:
  - 5, 6, 7 or 8 bits/character
  - 1, 1½ or 2 stop bits
  - Even, odd or no parity



Z-80 SIO/9 BLOCK DIAGRAM

- ×1, ×16, ×32 and ×64 clock modes
- Break generation and detection
- Parity, overrun and framing error detection

■ Binary synchronous features:
- Internal or external character synchronization
- One or two sync characters in separate registers
- Automatic sync character insertion/deletion
- CRC generation and checking

■ HDLC and SDLC features:
- Abort sequence generation and detection
- Automatic zero insertion and deletion
- Automatic flag insertion between messages
- Address field recognition
- Support for one to eight bits/character
- Valid receive messages protected from overrun
- CRC generation and checking

■ Interrupt features:
- Daisy-chain interrupt logic provides automatic interrupt vectoring with no external logic
- Programmable interrupt vector
- Status Affects Interrupt Vector mode for fast interrupt processing

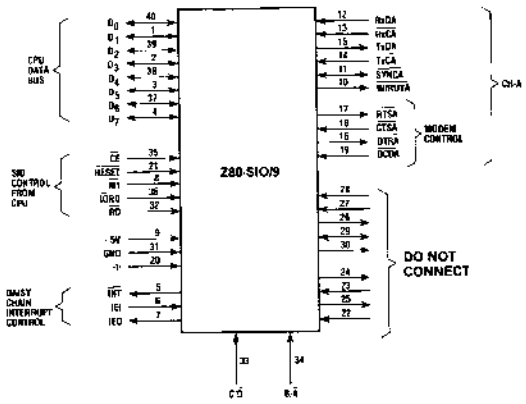
■ CRC-16 or CRC-CCITT block frame check

■ Modem control inputs and outputs

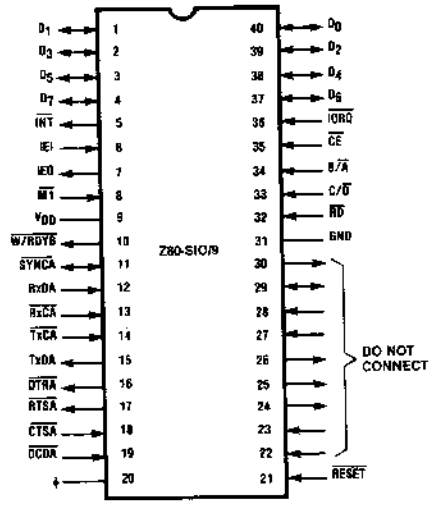
■ Modem status can be monitored

Refer to the *Z80-SIO Product Specification* (August 1978) and the *Z80-SIO Technical Manual* (August 1978) for detailed functional and electrical descriptions. All functional and electrical descriptions in these publications are applicable to the Z80-SIO, except that Channel B cannot be used for data input or output and pins 22 to 30 must not be connected.

Write Register 2 (interrupt vector) and the Status Affects Vector bit in Write Register 1 are, however, still programmed by selecting Channel B with the B/A select input. All other bits in Write Register 1 or Channel B must be programmed to 0.



FUNCTIONAL PIN CONFIGURATION



PACKAGE CONFIGURATION

Ordering Information

Z80A-SIO/9 CS	= 4 MHz maximum clock rate, ceramic package, 0°C to +70°C temperature range	Z80-SIO/9 CS	= 2.5 MHz maximum clock rate, ceramic package, 0°C to +70°C temperature range
Z80A-SIO/9 PS	= 4 MHz maximum clock rate, plastic package, 0°C to +70°C temperature range	Z80-SIO/9 PS	= 2.5 MHz maximum clock rate, plastic package, 0°C to +70°C temperature range

Zilog

Product Specification

Z6104
4096 x 1 Bit Static RAM

Description

The Zilog Z6104 and Z6104L high-performance 4096-bit static read/write random access memories are organized as 4096 1-bit words. Data storage is static; however, addresses are clocked in by the leading edge of Chip Enable.

The Z6104 is fabricated with n-channel silicon-gate depletion-load technology. Polysilicon load resistors used in the memory array result in a very small die size.

Features

- 4096 x 1 organization
- Static storage—no refresh required
- Z6104 offers high speed; Z6104L offers low power consumption
- Separate pins for data input and output
- Industry-standard 18-pin configuration and package
- Single +5 V, ±10% supply voltage
- Fully TTL compatible (Fan-out: two U.L. or eight LSTTL loads)

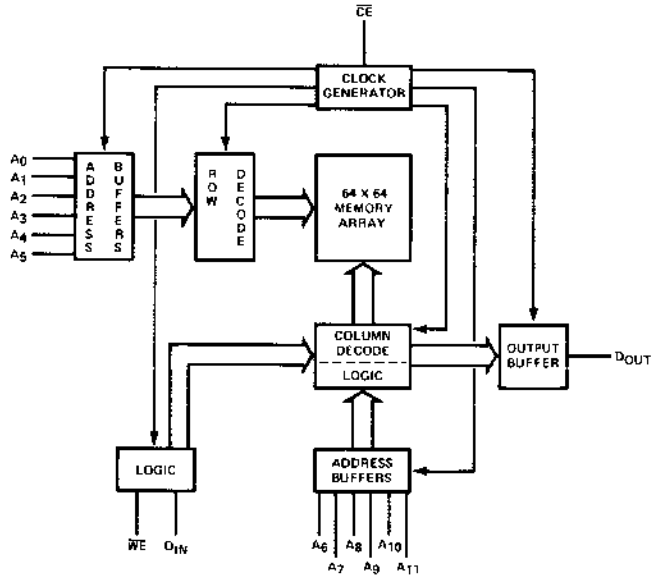
Access Times

Part Number	Access Time	Cycle Time
Z6104-2	150 ns	240 ns
Z6104-3	200 ns	320 ns
Z6104-4	250 ns	380 ns
Z6104L-4	250 ns	380 ns
Z6104L-5	300 ns	440 ns
Z6104L-6	350 ns	510 ns

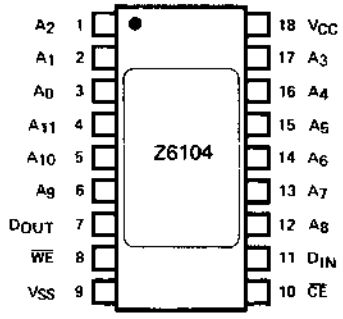
Functional Description

Two control inputs, $\overline{CE}$ (Chip Enable, active Low) and $\overline{WE}$ (Write Enable, active Low) determine device operation. Although data storage is static, the 12-bit address is clocked into the on-chip address register by the High-to-Low transition of $\overline{CE}$. Correct address information is, therefore, required only during a narrow timing window (t_{AH}) after the transition on $\overline{CE}$. Each access of a new address must be initiated by a High-to-Low transition on $\overline{CE}$ after $\overline{CE}$ has been High for at least the minimum specified precharge time (t_P).

Functional Block Diagram

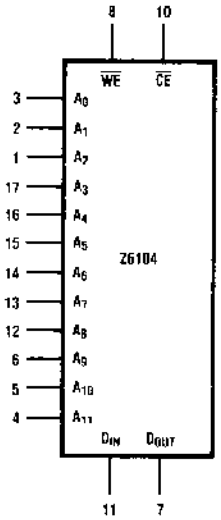


Pin Connections



$A_0 - A_{11}$	ADDRESS INPUTS
$\overline{CE}$	CHIP ENABLE
D_{IN}	DATA INPUT
D_{OUT}	DATA OUTPUT
V_{SS}	GROUND
V_{CC}	POWER (+5V)
$\overline{WE}$	WRITE ENABLE

Logic Symbol



Read Cycle

The data output (D_{OUT}) is active (High or Low) only when $\overline{CE}$ is active (Low) and $\overline{WE}$ is not active (High). D_{OUT} is inactive (also called open, floating or 3-stated) when the chip is deselected or in the write mode.

Write Cycle

Input data is written into the address memory location while both $\overline{CE}$ and $\overline{WE}$ are Low. Correct information at the data input (D_{IN}) is required only during a narrow timing window starting t_{DS} before the end of a write operation that is determined by either $\overline{CE}$ or $\overline{WE}$ going High.

The timing diagram depicts a write cycle with the shortest possible $\overline{WE}$ pulse width (t_{WW}). $\overline{WE}$ can go Low simultaneously with (or even before) $\overline{CE}$ goes Low, but this requires a wider write pulse since $\overline{WE}$ may not go High earlier than t_{CW} after the High-to-Low transition of $\overline{CE}$. If $\overline{WE}$ goes High while the chip is still selected, a read operation is started and D_{OUT} becomes active.

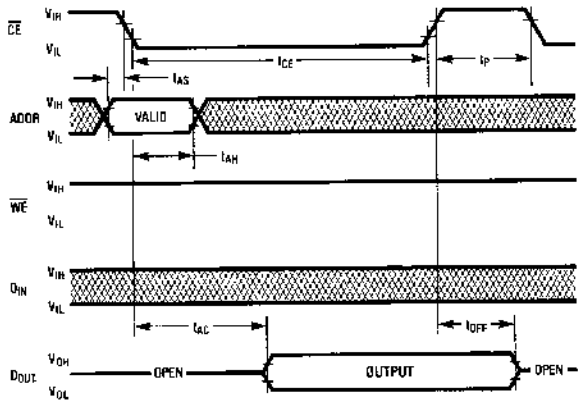
$\overline{WE}$ can stay Low until or beyond the rising edge of $\overline{CE}$. In this case, the Low-to-High transition of $\overline{CE}$ determines the end of the write operation. The timing of D_{IN} and t_{WW} must be referenced to the rising edge of $\overline{CE}$, not of $\overline{WE}$.

Read/Modify/Write Cycle

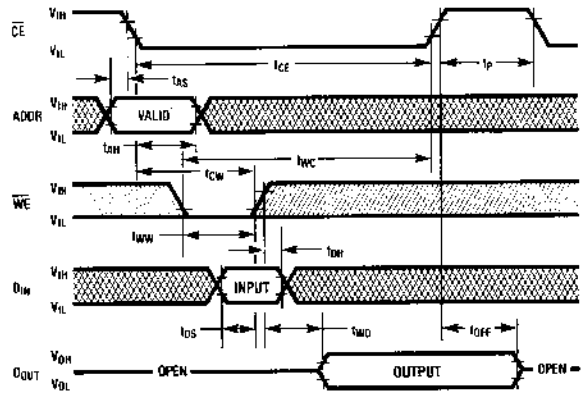
A read/modify/write operation begins like a normal read cycle. Data is read, then modified by circuitry external to the memory. The modified result is written back into the same memory location.

Common D_{IN} and D_{OUT}

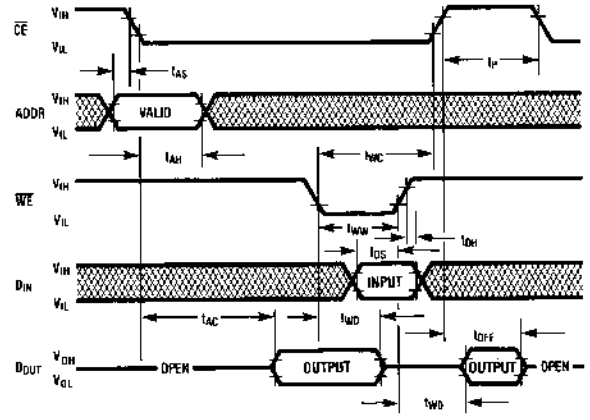
D_{IN} and D_{OUT} can be tied together to form a bidirectional data bus; however, for read/modify/write operation, the timing must be adjusted to avoid bus-contention problems. The write pulse width ($\overline{WE}$ Low) must be long enough to accommodate the Write-Enable-to-Data-Out delay (t_{WD}) plus the maximum delay for activating the external data driver plus the specified data input set-up time.



Read Cycle



Write Cycle



Read/Modify/Write Cycle

Microprocessor Applications

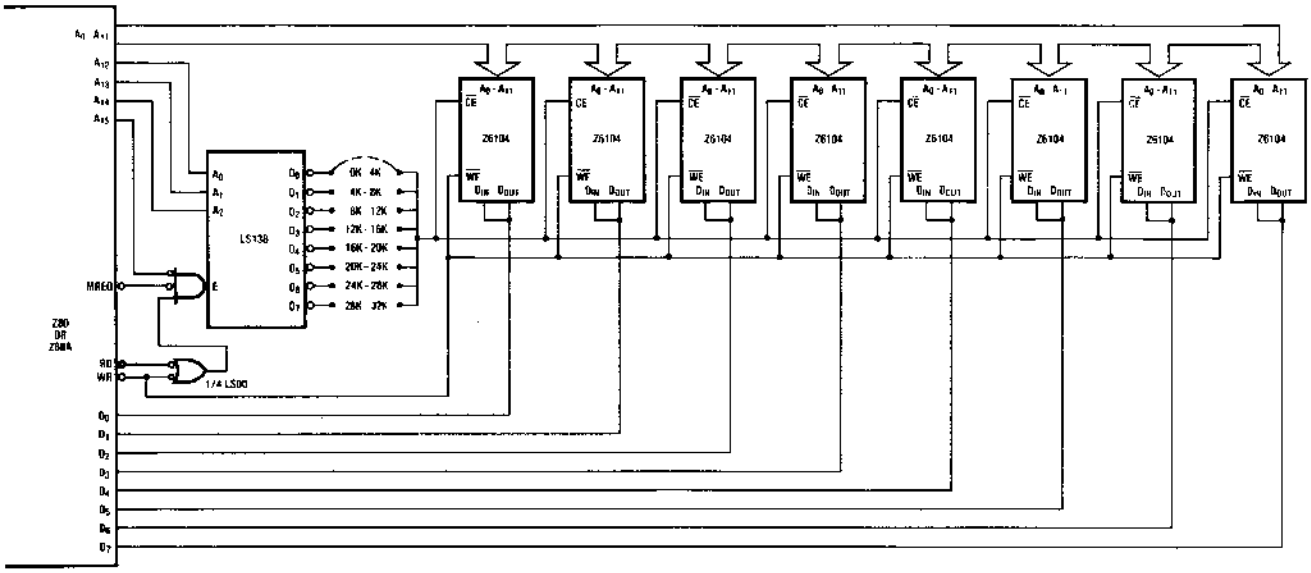
The Z6104 combines the convenience of fully static RAMs with edge-triggered on-chip address registers that simplify timing and address/data busing in microprocessor systems.

The edge-triggered address registers are clocked by the High-to-Low transition of $\overline{CE}$. Any access to a new address must therefore be initiated by $\overline{CE}$ becoming active (going Low). Other than this, the Z6104 is as straightforward and easy to understand as any fully static RAM.

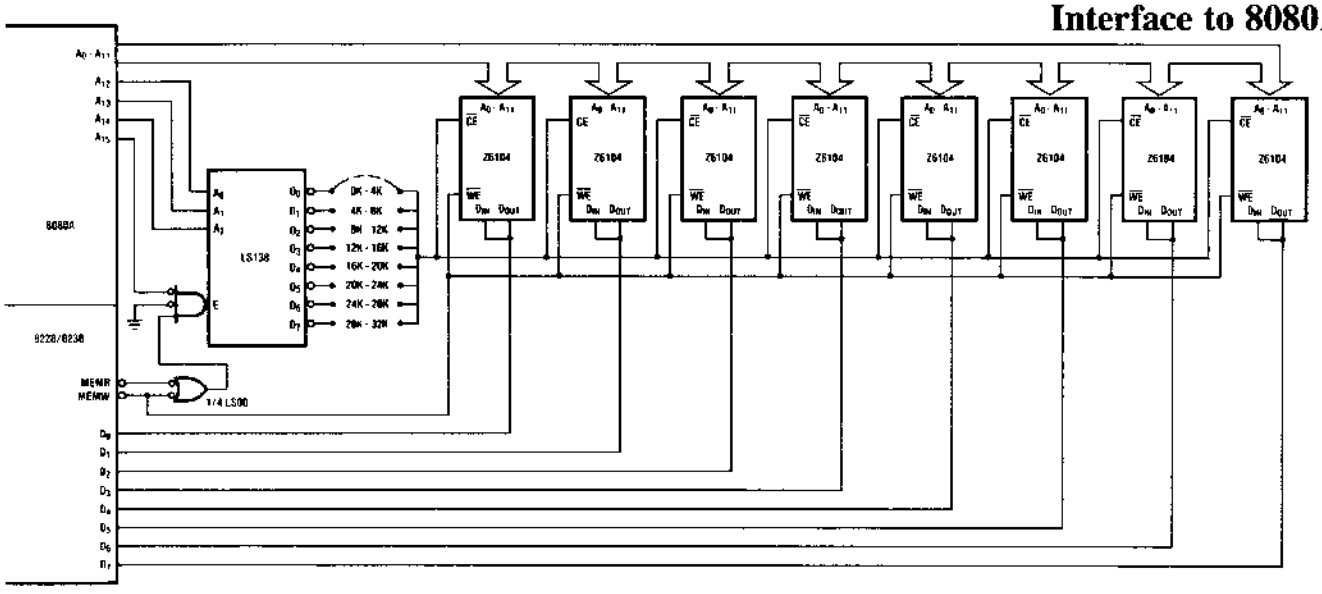
The data output buffers are active only while $\overline{WE}$ is High and $\overline{CE}$ is Low. Data is written into the selected storage cell when $\overline{CE}$ and $\overline{WE}$ are both Low simultaneously. Since the write operation is fully static, there is only one constraint on D_{IN}: it must be stable and valid for a specified set-up time before the end of the write operation, as defined by either $\overline{WE}$ or $\overline{CE}$ going High.

These simple and straightforward timing requirements result in a very easy interface to various microprocessors, as shown in the following figures for the Z80, the 8080A and the 6800.

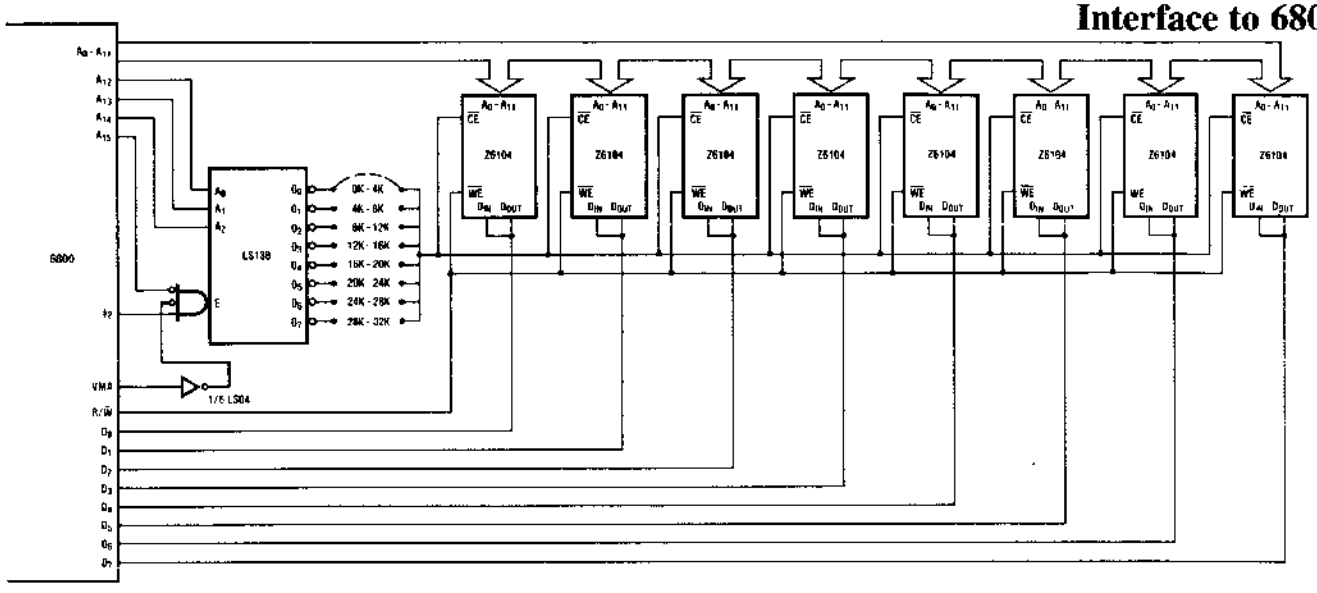
Interface to Z-80



Interface to 8080A



Interface to 6800



Line Reflections, Ringing and Undershoot

Large MOS memory arrays are usually driven by TTL buffers with fairly fast rise and fall times (less than 5 ns). When the lines driven by these buffers are longer than six inches, they do not represent a lumped capacitive load; rather, they act as transmission lines with distributed inductance and capacitance and a typical characteristic impedance of 100Ω to 150Ω, depending on line width.

A transition generated by the TTL buffer travels along the line at a speed of approximately six inches per nanosecond. When the transition reaches the far end of an unterminated line (usually the farthest MOS load), it doubles its amplitude and travels back towards the buffer, where it is again reflected by the low TTL output impedance, thus changing its sign and resulting in the familiar ringing.

The first reflection of a High-to-Low transition can go substantially negative: A +3 to +0.5 V transition at the driver results in a +0.5 to −2 V reflection. On an MOS input, this negative undershoot forward biases the input-to-substrate diode, injects current into the substrate and can disturb the data content of the memory. There are three proven methods for reducing these reflections.

- Terminate the far end of the line with a resistor (≃150Ω) connected either to V_{CC}, to a voltage divider or to a capacitor.
- Series terminate at the TTL buffer with a ≃100Ω resistor.
- Use CMOS buffers, which are inherently much slower than TTL, and thus avoid the problem, but require that the memory timing permits the added delay.

Calculating Average Operating Current

The average operating dc current consumption of the Z6104 and Z6104L, $I_{CC(avg)}$, is a function of three parameters:

- $I_{CC(SB)}$: The standby or precharge current while the memory is disabled.
- $I_{CC(SEL)}$: The select current while the memory is enabled.
- Q: The charge consumed during each memory cycle in the charging and discharging of internal capacitances.

The formula for calculating the average operating current is as follows:

$$I_{CC(avg)} = \frac{I_{CC(SB)} \cdot t_P + I_{CC(SEL)} \cdot t_{CE} + Q}{t_P + t_{CE}}$$

The following examples illustrate the effect of cycle time on average operating current: power consumption is reduced when the cycle time is lengthened.

A. Max Speed: 6104L-4 at t_P = 90 ns, t_{CE} = 250 ns, T_A = +70°C

$$I_{CC(avg)} = \frac{23 \cdot 90 + 30 \cdot 250 + 2000}{90 + 250} = 34 \text{ mA (max)}$$

B. 1 μs Cycle Time: 6104L-4 at t_P = 750 ns, t_{CE} = 250 ns, T_A = +70°C

$$I_{CC(avg)} = \frac{23 \cdot 750 + 30 \cdot 250 + 2000}{750 + 250} = 27 \text{ mA (max)}$$

Absolute Maximum Ratings

Rating	Min	Max	Unit
Voltage on any pin relative to V_{SS}	−0.3	+7.0	V
Storage temperature (ambient)	−65	+150	°C
Temperature under bias	0	+70	°C

Stresses greater than those listed in this table may cause permanent damage to the device. This is a stress rating only: functional operation of the device at these or any other conditions greater than those indicated in this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect reliability.

D.C. Electrical Characteristics

V_{CC} = +5.0 V ±10%; T_A = 0°C to +70°C

Symbol	Parameter	Min	Max	Units	Conditions
V_{OH}	Output High Voltage	+2.4		V	I_{OH} = −200 μA; $\overline{CE}$ at V_{IL}; $\overline{WE}$ at V_{IH}
V_{OL}	Output Low Voltage		+0.4	V	I_{OL} = 3.2 mA; $\overline{CE}$ at V_{IL}; $\overline{WE}$ at V_{IH}
V_{IH}	Input High Voltage	+2.0	+6.0	V	
V_{IL}	Input High Voltage	−0.3		V	
I_Z	I/O Leakage Current	−10	+10	μA	$0 < V_{IN} < 5.5$ V

A.C. Electrical Characteristics

Symbol	Parameter	Z6104-2		Z6104-3		Z6104-4 Z6104L-4		Z6104L-5		Z6104L-6		Unit
		Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	
t_{AC}	Read Access Time[1, 2]		150		200		250		300		350	ns
t_C	Read or Write Cycle Time[3]	240		320		380		440		510		ns
t_{CE}	Chip Enable Pulse Width (Low)	150	10^5	200	10^5	250	10^5	300	10^5	350	10^5	ns
t_P	Chip Enable Precharge Time (High)	50		80		90		100		120		ns
t_{AS}	Address Set-up Time	0		0		0		0		0		ns
t_{AH}	Address Hold Time	50		80		100		120		140		ns
t_{CW}	Chip Enable to Write Disable	90		110		140		180		220		ns
t t_{WC}	Write Enable to Chip Disable	40		50		60		80		100		ns
t_{WW}	Write Enable Pulse Width	40		50		60		80		100		ns
t_{DS}	Data Input Set-up Time	60		70		80		100		120		ns
t_{DH}	Data Input Hold Time	0		0		0		0		0		ns
t_{WD}	Write Enable and Disable to Data Out	0	50	0	60	0	80	0	100	0	120	ns
t_{OFF}	Output Buffer Turn-off Delay[4]		40		50		60		80		100	ns
t_T	Transition Time		50		50		50		50		50	ns

1. Output loaded with 100 pF and TTL loading. Output levels are measured at V_{OH} = 2.0 V and V_{OL} = 0.8 V.
2. Typical change in access time vs load capacitance is 0.1 ns/pF.
3. $t_C = t_{CE} + t_P + 2 \cdot t_T$. Test conditions use t_T = 20 ns.
4. Output waveform is dependent on output load. D_{OUT} is guaranteed to be off within t_{OFF}.

Power Consumption

V_{CC} = 5.5V

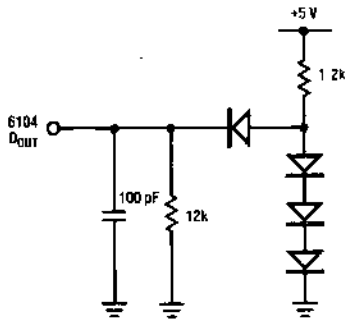
Symbol	Parameter	Z6104-2, 3, 4		Z6104L-4, 5, 6		Unit	Condition
	T_A =	0°C	70°C	0°C	70°C		
$I_{CC(SB)}$	Standby Power Supply Current	60	45	40	30	mA	$\overline{CE}$ = High
$I_{CC(SEL)}$	Selected Power Supply Current	85	65	60	45	mA	$\overline{CE}$ = Low
Q	Dynamic Power Supply Charge	2000	2000	2000	2000	pC[1]	

[1] pC = mA • ns

Capacitance

V_{CC} = +5 V ±10%, T_A = 0°C to +70°C

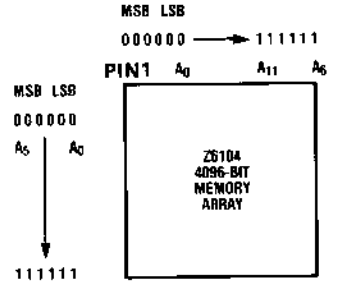
Symbol	Parameter	Min	Max	Unit
C_I	Input Capacitance		5	pF
C_O	Output Capacitance		5	pF



LOAD CIRCUIT

Address Pin Labeling

From a functional point of view, the labeling of RAM address pins is arbitrary. For the Z6104, the address pins are labeled so they address the two-dimensional cell array consistent with the physical layout.



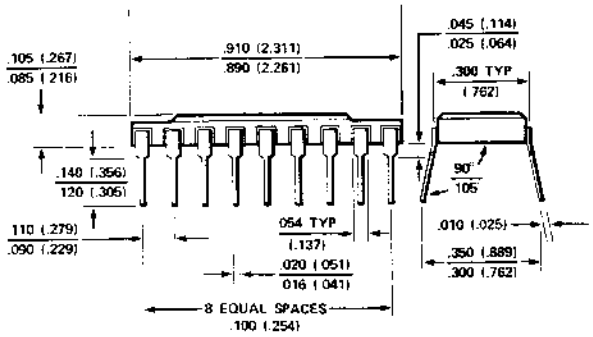
Cell Address Organization. Top View.

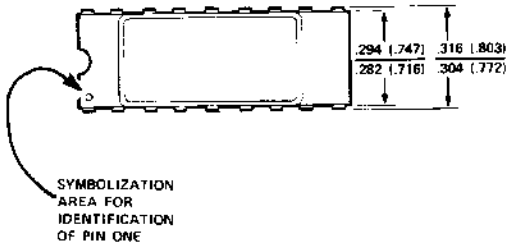
Ordering Information

Suffix	Access Time	Suffix	Package	Suffix	Temperature Range
–2	150 ns	C	Ceramic	S	0°C to +70°C, 5 V ±10%
–3	200 ns	P	Plastic		
–4	250 ns				
–5	300 ns				
–6	350 ns				

Example: Z6104L-4CS = Low power, 250 ns access time, ceramic package, standard range.

Package Outline





NOTE: Dimensions in parentheses are for metric system (cm).

Zilog

Product Specification

PRELIMINARY

Z6114 1024 x 4 Bit Static RAM

General Description

The Zilog Z6114 is a high-performance 4096-bit static RAM organized as 1024 4-bit words. Data storage is static; however, addresses are clocked in by the leading edge of Chip Enable. The four bidirectional I/O lines of the Z6114 make it especially suited for microprocessor systems with less than 4K bytes of RAM because as few as two Z6114 devices can provide byte-wide storage.

The Z6114 is fabricated with n-channel silicon-gate depletion-load technology. Polysilicon load resistors used in the memory array result in a very small die size.

Zilog also manufactures the Z6142, which is functionally identical to the Z6114 except that it contains an additional Chip Enable input (active High) and an Output Enable input.

Features

- 1024 x 4 organization
- Static storage -no refresh required
- Four I/O lines
- Industry-standard 18-pin configuration and package
- Single +5 V, ±10% supply voltage
- Fully TTL compatible (Fan-out: two U.L. or eight LSTTL loads)

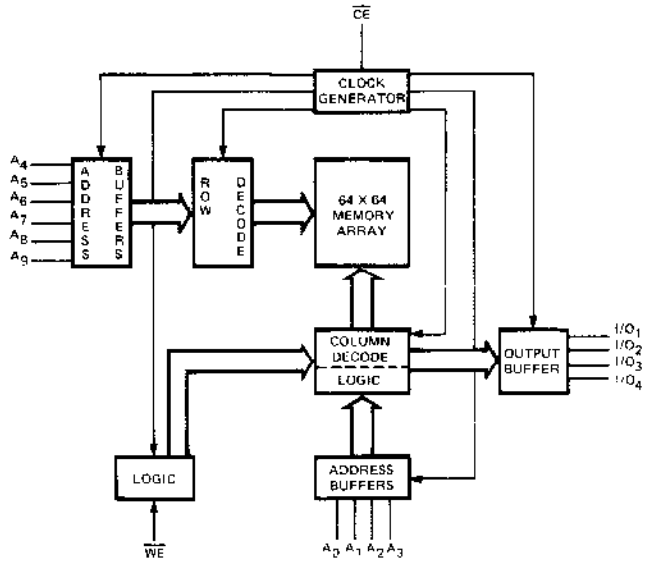
Access Times

Part Number	Access Time	Cycle Time
Z6114-2	150 ns	240 ns
Z6114-3	200 ns	320 ns
Z6114-4	250 ns	380 ns
Z6114-5	300 ns	440 ns
Z6114-6	350 ns	510 ns

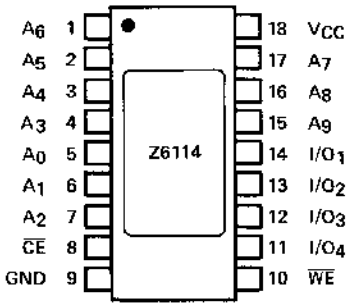
Functional Description

Two control inputs, $\overline{CE}$ (Chip Enable, active Low) and $\overline{WE}$ (Write Enable, active Low) determine device operation. Although data storage is static, the 10-bit address is clocked into the on-chip address register by the High-to-Low transition of $\overline{CE}$. Correct address information is, therefore, required only during a narrow timing window (t_{AH}) after the transition on $\overline{CE}$. Each access of a new address must be initiated by a High-to-Low transition on $\overline{CE}$ after $\overline{CE}$ has been High for at least the minimum specified precharge time (tp).

Functional Block Diagram

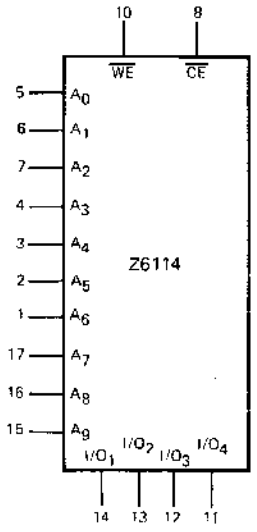


Pin Configuration



A_0–A_{11}	ADDRESS INPUTS
$\overline{CE}$	CHIP ENABLE
V_{SS}	GROUND
V_{CC}	POWER (+5V)
$\overline{WE}$	WRITE ENABLE
I/O_1–I/O_4	DATA INPUT/OUTPUTS

Logic Symbol



Read Cycle

The four I/O pins are active outputs (High or Low) only when $\overline{CE}$ is active (Low) and $\overline{WE}$ is not active (High). The I/O pins are inactive (also called open, floating or 3-stated) when the chip is deselected or in the write mode.

Write Cycle

Input data is written into the addressed memory location while both $\overline{CE}$ and $\overline{WE}$ are Low. Correct information at the I/O pins is required only during a narrow timing window starting t_{DS} before the end of a write operation determined by either $\overline{CE}$ or $\overline{WE}$ going High.

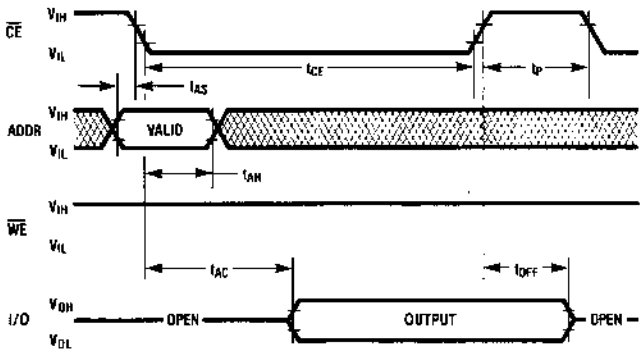
The write timing diagram depicts a write cycle with the shortest possible $\overline{WE}$ pulse width (t_{WW}). $\overline{WE}$ can go Low simultaneously with, (or even before) $\overline{CE}$ goes Low, but this requires a wide write pulse since $\overline{WE}$ may not go High earlier than t_{CW} plus t_{WW} after the High-to-Low transition of $\overline{CE}$. If $\overline{WE}$ goes High while the chip is still selected, a read opeation is started and the I/O pins become active outputs.

$\overline{WE}$ can stay Low until or beyond the rising edge of $\overline{CE}$. In this case, the Low-to-High transition of $\overline{CE}$ determines the end of the write operation. The write timing of t_{DS} and t_{WW} must be referenced to the rising edge of $\overline{CE}$, not of $\overline{WE}$.

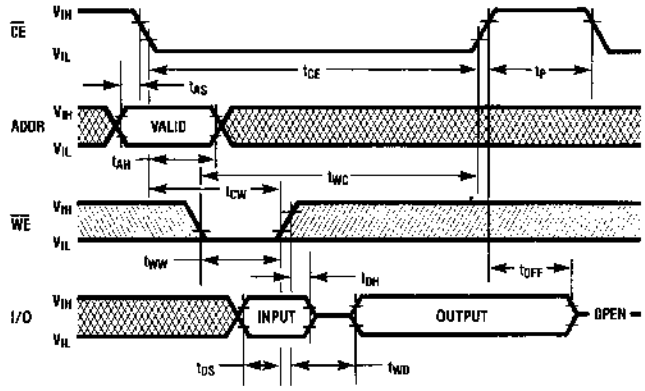
Read/Modify/Write Cycle

A read/modify/write cycle begins like a normal read cycle. Data is read, then modified by circuitry external to the memory. The modified result is written back into the same memory location.

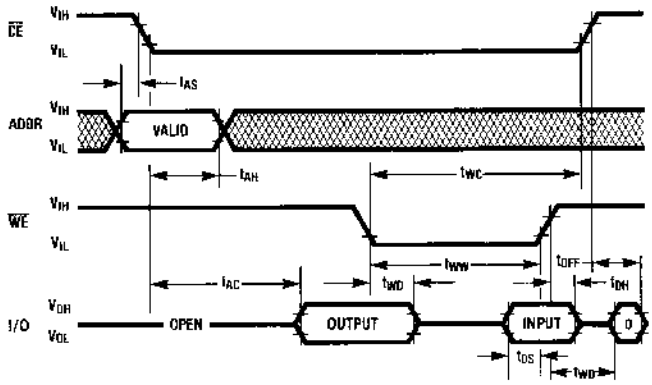
The $\overline{WE}$ pulse width must be long enough to accommodate the maximum output buffer turn-off delay (t_{OFF}) plus the maximum delay in activating the external data driver circuits plus the input data set-up time (t_{DS}).



Read Cycle



Write Cycle



Read/Modify/Write Cycle

Applications

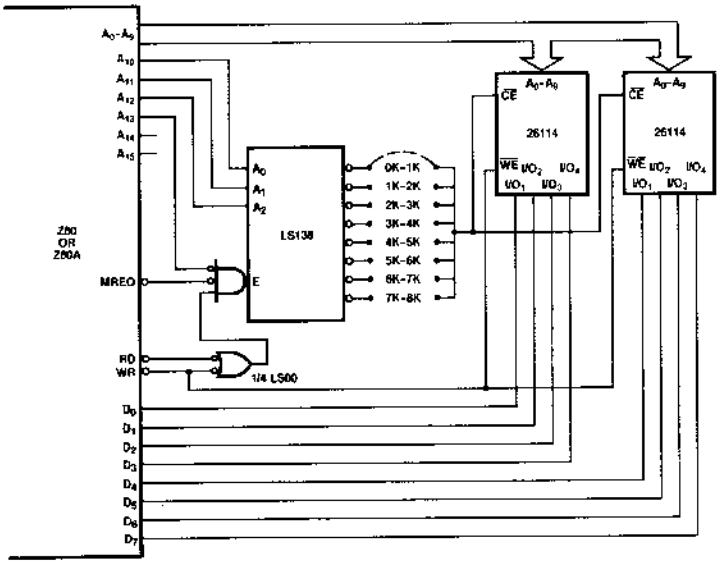
The Z6114 combines the convenience of fully static RAMs with I/O and edge-triggered on-chip address registers that simplify timing and address/data busing in microprocessor systems.

The edge-triggered address registers are clocked by the High-to-Low transition of $\overline{CE}$. Any access to a new address must therefore be initiated by $\overline{CE}$ becoming active (going Low). Other than this, the Z6114 is a straightforward and easy to understand as any fully static RAM.

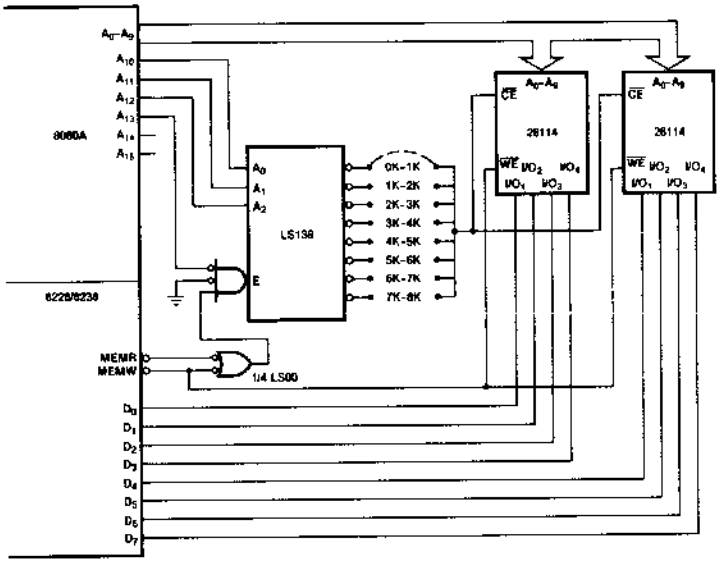
The I/O output buffers are active only while $\overline{WE}$ is High and $\overline{CE}$ is Low. Data is written into the selected storage cell when $\overline{CE}$ and $\overline{WE}$ are both Low simultaneously. Since the write operation is fully static, there is only one constraint on input data: it must be stable and valid for a specified set-up time before the end of the write operation, as defined by either $\overline{WE}$ or $\overline{CE}$ going High.

These simple and straightforward timing requirements result in a very easy interface to various microprocessors, as shown in the following figures for the Z80, the 8080A and the 6800.

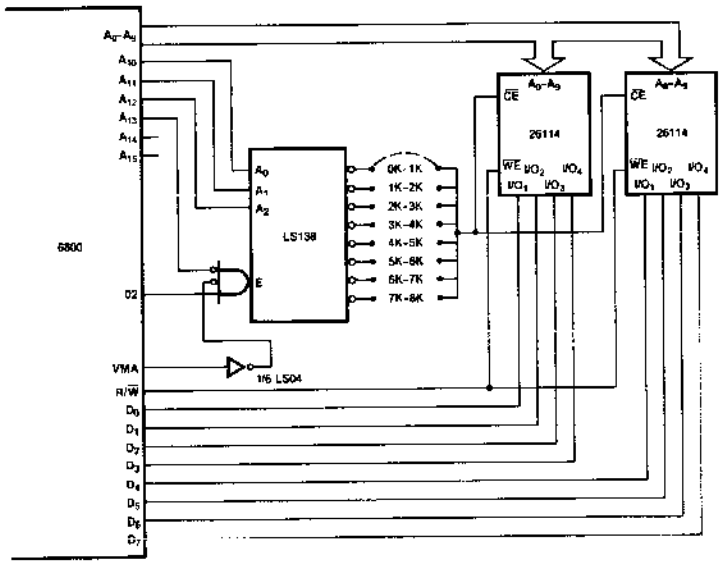
Interface to Z-80



Interface to 8080A



Interface to 6800



Line Reflections, Ringing and Undershoot

Large MOS memory arrays are usually driven by TTL buffers with fairly fast rise and fall times (less than 5 ns). When the lines driven by these buffers are longer than six inches, they do not represent a lumped capacitive load; rather, they act as transmission lines with distributed inductance and capacitance and a typical characteristic impedance of 100Ω to 150Ω, depending on line width.

A transition generated by the TTL buffer travels along the line at a speed of approximately six inches per nanosecond. When the transition reaches the far end of an unterminated line (usually the farthest MOS load), it doubles its amplitude and travels back towards the buffer, where it is again reflected by the low TTL output impedance, thus changing its sign and resulting in the familiar ringing.

The first reflection of a High-to-Low transition can go substantially negative: A +3 to +0.5 V transition at the driver results in a +0.5 to −2 V reflection. On an MOS input, this negative undershoot forward biases the input-to-substrate diode, injects current into the substrate and can disturb the data content of the memory. There are three proven methods for reducing these reflections.

- Terminate the far end of the line with a resistor (≃150Ω) connected either to V_{CC}, to a voltage divider or to a capacitor.
- Series terminate at the TTL buffer with a ≃100Ω resistor.
- Use CMOS buffers, which are inherently much slower than TTL, and thus avoid the problem, but require that the memory timing permits the added delay.

Calculating Average Operating Current

The average operating dc current consumption of the Z6114, $I_{CC(avg)}$, is a function of three parameters:

- $I_{CC(SB)}$: The standby or precharge current while the memory is disabled.
- $I_{CC(SEL)}$: The select current while the memory is enabled.
- Q: The charge consumed during each memory cycle in the charging and discharging of internal capacitances.

The formula for calculating the average operating current is as follows:

$$I_{CC(avg)} = \frac{I_{CC(SB)} \cdot t_P + I_{CC(SEL)} \cdot t_{CE} + Q}{t_P + t_{CE}}$$

The following examples illustrate the effect of cycle time on average operating current: power consumption is reduced when the cycle time is lengthened.

A. **Max Speed** 6114 at $t_P = 90$ ns, $t_{CE} = 250$ ns, $T_A = +70°C$

$$I_{CC(avg)} \leq \frac{40 \cdot 90 + 55 \cdot 250 + 2000}{90 + 250} = 57 \text{ mA (max)}$$

B. **1μs cycle time** 6114-4 at $t_P = 750$ ns, $t_{CE} = 250$ ns, $T_A = +70°C$

$$I_{CC(avg)} \leq \frac{40 \cdot 750 + 55 \cdot 250 + 2000}{750 + 250} = 46 \text{ mA (max)}$$

Absolute Maximum Ratings

Rating	Min	Max	Unit
Voltage on any pin relative to V_{SS}	−0.3	+7.0	V
Storage Temperature (ambient)	−65	+150	°C
Temperature under bias	0	+70	°C

Stresses greater than those listed in this table may cause permanent damage to the device. This is a stress rating only: functional operation of the device at these or any other conditions greater than those indicated in this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect reliability.

DC Electrical Characteristics

V_{CC} = +5.0 V ±10%; T_A = 0°C to +70°C unless otherwise specified.

Symbol	Parameter	Min	Max	Units	Conditions
V_{OH}	Output High Voltage	+2.4		V	I_{OH} = −200 μA; $\overline{CE}$ at V_{IL}; $\overline{WE}$ at V_{IH}
V_{OL}	Output Low Voltage		+0.4	V	I_{OL} = 3.2 mA; $\overline{CE}$ at V_{IL}; $\overline{WE}$ at V_{IH}
V_{IH}	Input High Voltage	+2.0	+6.0	V	
V_{IL}	Input High Voltage	−0.3		V	
I_Z	I/O Leakage Current	−10	+10	μA	$0 < V_{IN} < 5.5$ V

AC Electrical Characteristics

V_{CC} = +5 V ±10%; T_A = 0°C to +70°C

Symbol	Parameter	Z6114-2		Z6114-3		Z6114-4		Z6114-5		Z6114-6		Unit
		Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	
t_{AC}	Read Access Time[1, 2]		150		200		250		300		350	ns
t_C	Read or Write Cycle Time[3]	240		320		380		440		510		ns
t_{CE}	Chip Enable Pulse Width (Low)	150	10^5	200	10^5	250	10^5	300	10^5	350	10^5	ns
t_P	Chip Enable Precharge Time (High)	50		80		90		100		120		ns
t_{AS}	Address Set-up Time	0		0		0		0		0		ns
t_{AH}	Address Hold Time	50		80		100		120		140		ns
t_{CW}	Chip Enable to Write Disable	90		110		140		180		220		ns
t_{WC}	Write Enable to Chip Disable	40		50		60		80		100		ns
t_{WW}	Write Enable Pulse Width	40		50		60		80		100		ns
t_{DS}	Data Input Set-up Time	60		70		80		100		120		ns
t_{DH}	Data Input Hold Time	0		0		0		0		0		ns
t_{WD}	Write Enable and Disable to I/O	0	50	0	60	0	80	0	100	0	120	ns
t_{OFF}	Output Buffer Turn-off Delay[4]		40		50		60		80		100	ns
t_T	Transition Time		50		50		50		50		50	ns

1. I/O loaded with 100 pF and TTL loading. Output levels are measured at V_{OH} = 2.0 V and V_{OL} = 0.8 V.
2. Typical change in access time vs load capacitance is 0.1 ns/pF.
3. $t_C = t_{CE} + t_P + 2 \cdot t_T$. Test conditions use t_T = 20 ns.
4. Output waveform is dependent on output load. Output buffer is guaranteed to be off within t_{OFF}.

Power Consumption

V_{CC} = 5.5 V

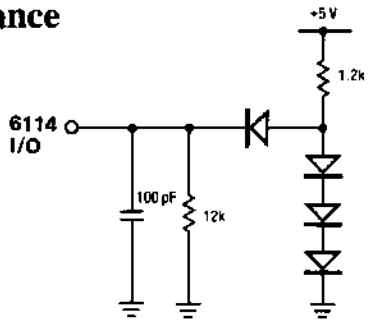
Symbol	Parameter	T_A =	Z6114-2, 3, 4		Z6114-5, 6		Unit	Condition
			0°C	+70°C	0°C	+70°C		
$I_{CC(SB)}$	Standby Power Supply Current		70	55	50	40	mA	$\overline{CE}$ = High
$I_{CC(SEL)}$	Selected Power Supply Current		95	75	70	55	mA	$\overline{CE}$ = Low
Q	Dynamic Power Supply Charge		2000	2000	2000	2000	pC[1]	

[1] pC = mA · ns

Capacitance

V_{CC} = +5 V ±10%, T_A = 0°C to +70°C

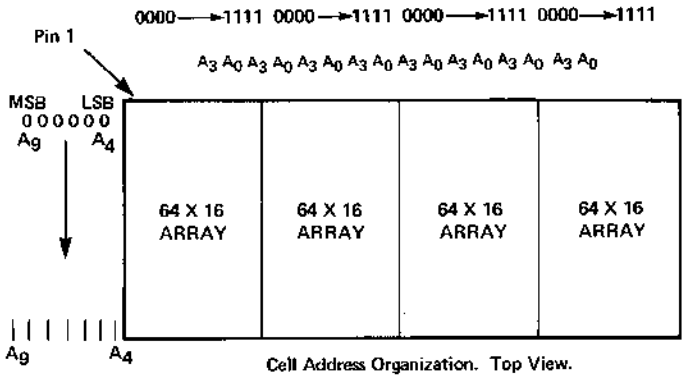
Symbol	Parameter	Min	Max	Unit
C_I	Input Capacitance		5	pF
C_O	Output Capacitance		5	pF



LOAD CIRCUIT

Address Pin Labeling

From a functional point of view, the labeling of RAM address pins is arbitrary. For the Z6114, the address pins are labeled so they address the two-dimensional cell array consistent with the physical layout.



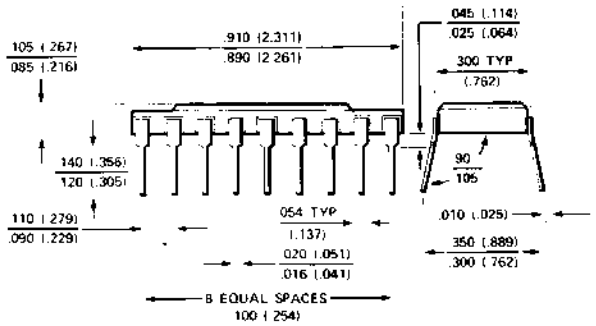
Cell Address Organization. Top View.

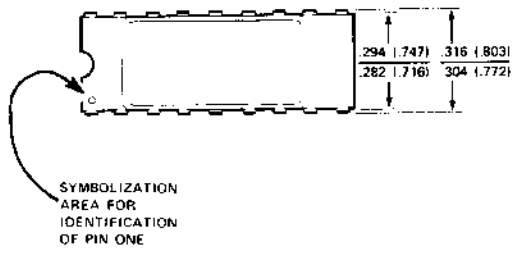
Ordering Information

Suffix	Access Time	Suffix	Package	Suffix	Temperature Range
−2	150 ns	C	Ceramic	S	0°C to +70°C, 5 V ±10%
−3	200 ns	P	Plastic		
−4	250 ns				
−5	300 ns				
−6	350 ns				

Example: Z6114-4CS = 250 ns access time, ceramic package, standard range.

Package Outline





NOTE: Dimensions in parentheses are for metric system (cm)

Zilog

Z6142
1024 x 4 Bit Static RAM

Product Specification

PRELIMINARY

General Description

The Zilog Z6142 is a high-performance 4096-bit static RAM organized as 1024 4-bit words. Data storage is static; however, addresses are clocked in by the leading edge of Chip Enable. The four bidirectional I/O lines and the Output Enable input of the Z6142 make it especially suited for microprocessor systems with less than 4K bytes of RAM because as few as two Z6142 devices can provide byte-wide storage.

The Z6142 contains two Chip Enable inputs and an Output Enable input. Zilog also manufactures the Z6114, which is functionally identical to the Z6142 except that it has one Chip Enable and no Output Enable. The basic operation and electrical characteristics of the Z6142 are the same as those of the Z6114.

The Z6142 is fabricated with n-channel silicon-gate depletion-load technology. Polysilicon load resistors used in the memory array result in a very small die size.

Features

- 1024 x 4 organization
- Static storage—no refresh required
- Four I/O lines
- Two Chip Enable inputs plus an Output Enable
- Industry-standard 20-pin configuration and package
- Single +5 V, ±10% supply voltage
- Fully TTL compatible (Fan-out: two U.L. or eight LSTTL loads)

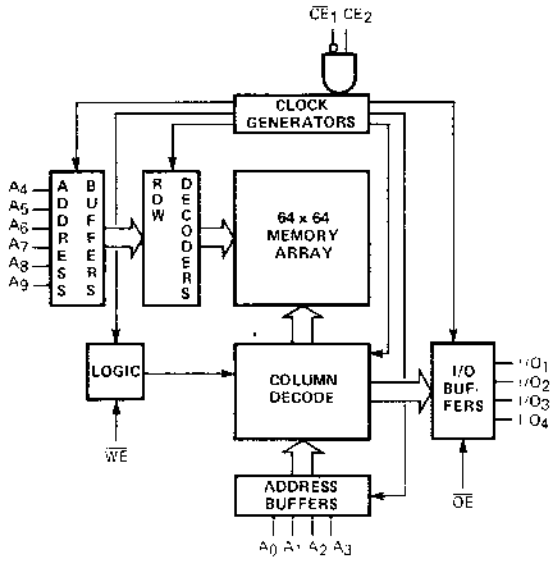
Access Times

Part Number	Access Time	Cycle Time
Z6142-2	150 ns	240 ns
Z6142-3	200 ns	320 ns
Z6142-4	250 ns	380 ns
Z6142-5	300 ns	440 ns
Z6142-6	350 ns	510 ns

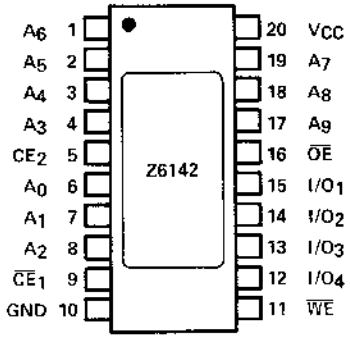
Functional Description

Three control inputs—$\overline{CE}_1$, CE_2 (Chip Enables) and $\overline{WE}$ (Write Enable)—determine device operation. Although data storage is static, the 10-bit address is clocked into the on-chip address register whenever the two ANDed Chip Enable inputs make the Chip Enable function go true. Correct address information is, therefore, required only during a narrow timing window (t_{AH}) after this transition. Each access of a new address must be initiated by the transition of the Chip Enable function going active, after this function has been inactive ($\overline{CE}_1$ = High or CE_2 = Low) for at least the minimum specified precharge time (tp). A High level on the Output Enable input ($\overline{OE}$) forces the four outputs into the inactive (also called high-impedance or floating) state; however, $\overline{OE}$ does not affect the internal operation of the device.

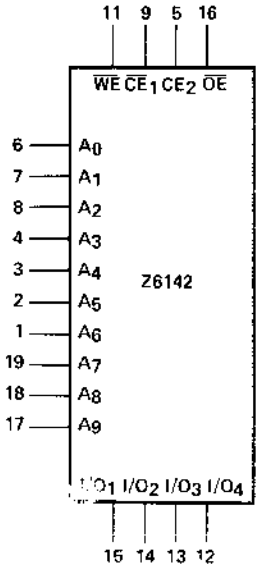
Functional Block Diagram



Pin Configuration



Logic Symbol



Section 2
Boards

Zilog

Section 2
Boards

Zilog brings to the OEM board market the same advantages it brought to the microprocessor market—the powerful architecture of the Z-80 and Z-80A CPU, the world's most advanced 8-bit microprocessor, *clever design* with maximum thru-put per board due to an intelligent mix of processor, program memory and I/O functions, and Zilog's overall *"systems concept"* of integrating complex functions in single board and multi-board configurations.

The leading member of the Zilog Board Family is the **Z-80 MCB.** This single board computer provides Z-80 CPU, 16K RAM standard (4K optional), 4K ROM, PROM or EPROM, two 8-bit parallel I/O ports and a serial I/O port, either RS-232 or 20 mil amp. current loop. All this on a single 7.7" x 7.5" computer card! In addition, this powerful single-board computer requires only a +5 volt supply! For system expansion, it mates to the standard Zilog bus, the same bus used in Zilog's mainframe MCZ-1 computers.

The Zilog Board Family includes:

- **The Z-80 RMB—RAM Memory Board**
  16K to 64K dynamic memory cards allow for maximum memory expansion
- **The Z-80 IOB—Input/Output Board**
  Four Z-80 PIOs give the user 64 programmable parallel I/O lines
- **The Z-80 SIB—Serial I/O Board**
  Four full duplex serial I/O channels for synchronous, asynchronous or bi-sync operation
- **The Z-80 AIO/AIB—Analog Input/Output Board**
  32 Single-ended or 16 differential input channels and two analog output channels. Offers 12-bit resolution and requires only +5 volt supply!
- **The Z-80 PMB—PROM Memory Board**
  Allows up to 32K ROM, PROM or EPROM on-board plus dual 8-bit parallel I/O ports and four 8-bit programmable counter/timers
- **The Z-80 PPB—PROM Programmer Board**
  Offers programming capabilities for industry standard PROMs and EPROMs.
- **The Z-80 MDC—Floppy Disk Controller Board**
  Controls up to 8 disk drives plus contains 48K of RAM memory
- **The Z-80 VDB—Video Display Board**
  Displays text in 24 lines of 80 characters each or displays graphics in a 240 x 560 dot matrix

Accessories:

- **SCE-4—Standard Card Enclosure**
  Includes power supply, 4-slot card cage, fan and enclosures

Additional hardware such as wire-wrap boards, extender cards, card cages, enclosures and power supplies are also available.

Zilog Microcomputer Boards offer what the OEM needs most—solutions. A conveniently sized board series (7.7" x 7.5"), a powerful mix of modular subsystems, the Zilog standard bus, easy hardware interface (boards require +5 volts only) and a wide range of accessories.

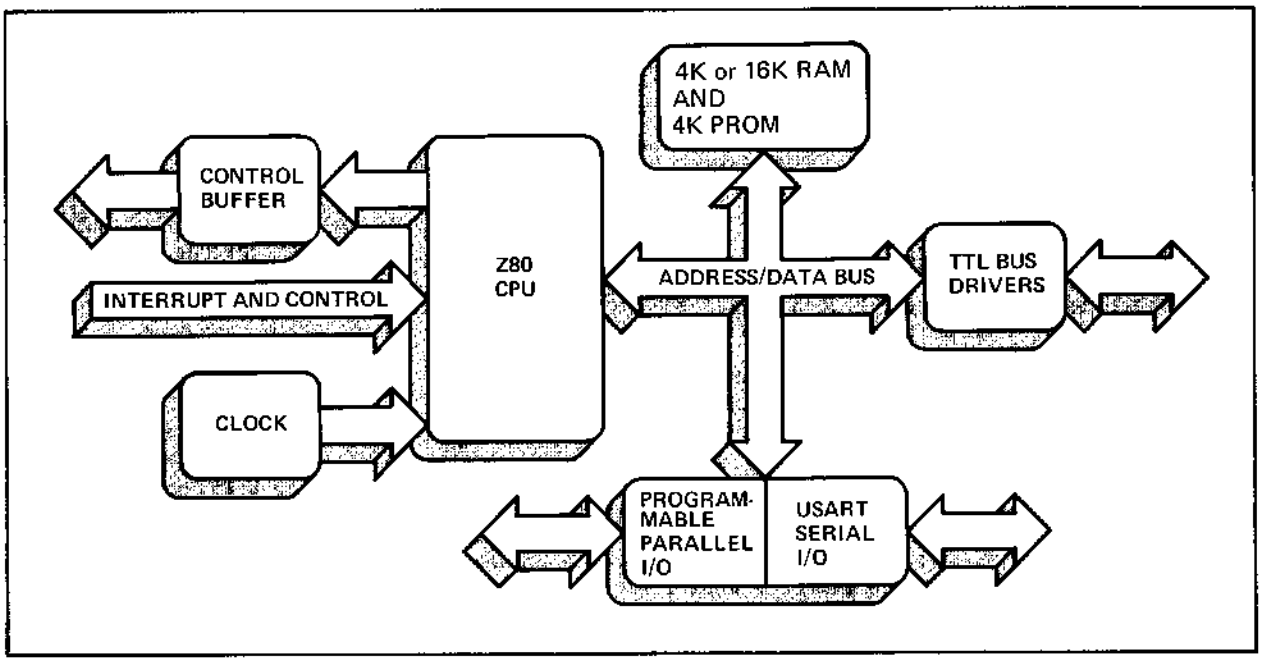
Zilog Microcomputer Boards—A Generation Ahead—Today!

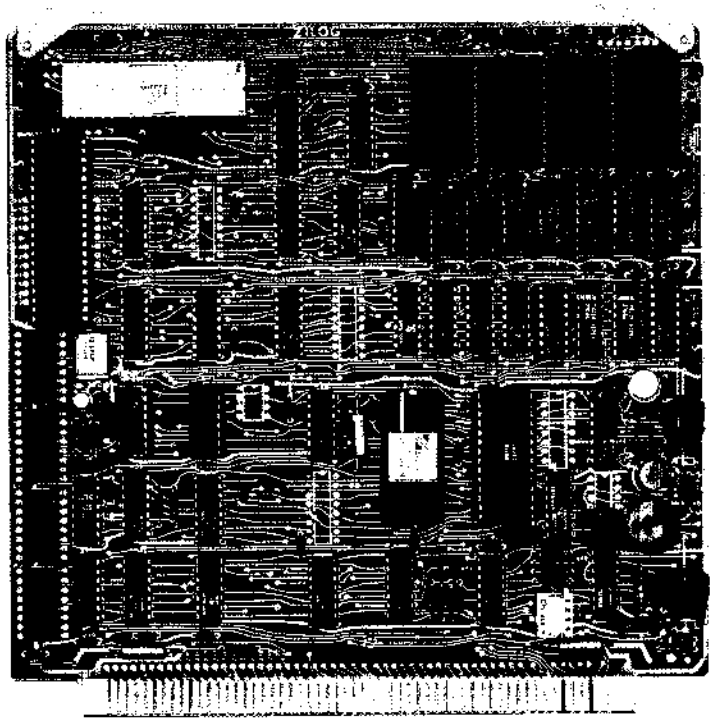
Zilog

Product Specification

Z-80® MCB Microcomputer Board

The Z-80 MCB Microcomputer Board is a complete single board computer with its own self-contained memory plus serial and parallel I/O ports. It features the use of the Z-80 CPU, Z-80 CTC, Z-80 PIO, and Z-6616 devices that have become standard components in the microcomputer industry.





Description

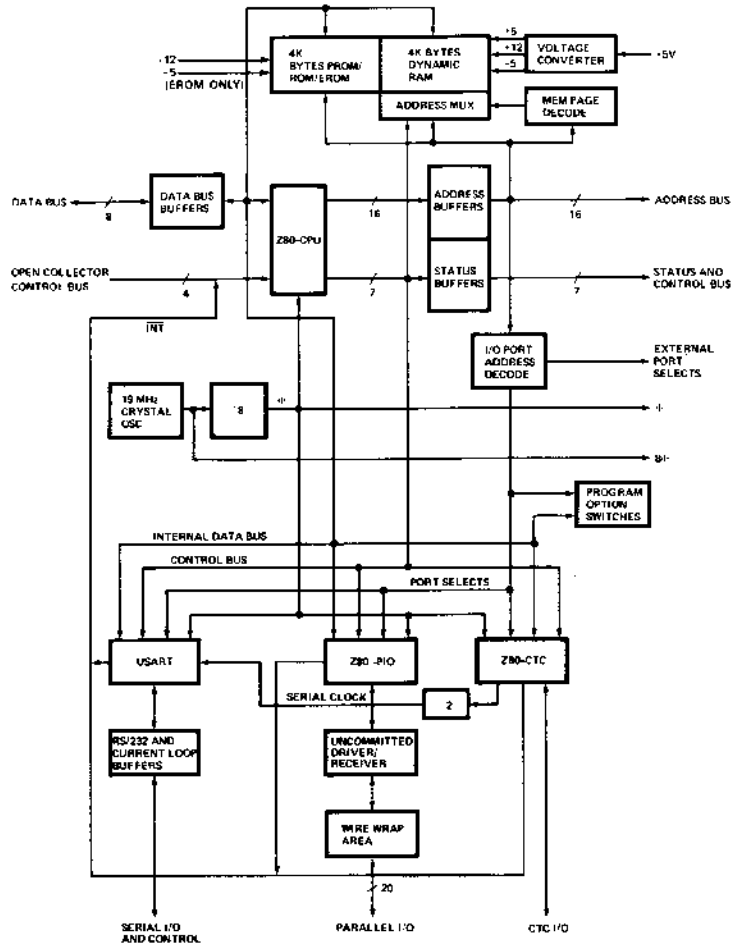
The Z80-Microcomputer Board (MCB) is a modular, single-board computer. It is designed around the Zilog Z80-CPU and employs an on-board DC converter to allow operation from a single +5 volt power supply. This board is highly flexible, and can be customized by the user for specific applications.

The basic configuration consists of the Z-80 CPU, 4K or 16K bytes of dynamic RAM, provision for up to 4K bytes of PROM, ROM, or EPROM, both parallel and serial I/O ports, I/O port decoders, and a crystal controlled clock. The parallel port is implemented with the Z-80 PIO with area reserved for user-supplied driver and/or receiver logic. One of the four timers in the Z-80 CTC is used as a baud rate generator for the serial interface implemented with an 8251 USART. Strapping options are available for selecting several memory and I/O port configurations, terminal interface schemes, and operating modes. Expansion of the card is made possible by feeding all buffered address, data, and control lines to a 122-pin edge connector. Two versions of monitor software (1K and 3K bytes) are available in bipolar PROMs for insertion into the four 24-pin PROM sockets, allowing software debugging and terminal interface (TTY, CRT, or disk).

Features

- Z-80 CPU single-chip n-channel processor with 158 instructions (including all of the 8080A's 78 instructions with total software compatibility). New instructions include memory-to-memory block transfers, I/O block transfers, 16-bit arithmetic, 9 types of rotates and shifts, bit manipulation and many new addressing modes. (See *Z-80 CPU Product Specification* for specific details.)
- 16K or 4K bytes of low-power dynamic RAM.
- MOS EPROM, fuseable bipolar PROM or masked ROM sockets for user's program storage. Optional Zilog monitor software is available in 1K, and 3K byte versions..
- Strapping options can relocate both the RAM and ROM memories into any segment of the 64K address space.
- Programmable full duplex serial I/O port with RS-232 or current-loop interface. Can be programmed to operate at 14 separate baud rates from 50 baud to 38.4K baud. Can operate using any asynchronous or synchronous protocol.
- Modem control signals including Request To Send and Clear To Send are provided with the RS-232 interface.
- A separate reader control line is available for teletype terminals not equipped with automatic reader control.
- Universal parallel I/O can be programmed to define any direction and data-transfer characteristics for two 8-bit ports. Full flexibility in buffering and terminating the parallel ports is provided by uncommitted driver/termination device locations. Data transfer can be accomplished under full interrupt control. (See *Z-80 PIO Product Specification* for specific details.)
- Z-80 CTC includes four programmable counter/timer circuit channels. It is used as the programmable baud rate generator; additional channels can be used as real-time clocks. (See *Z-80 CTC Product Specification* for specific details.)
- Switches on the board can be read by the CPU for various options. The software provided in the standard 1K byte monitor can read these switches and set the communication frequency to any of 14 common rates by programming the CTC. Switches can also be used for other similar functions.
- Board contains I/O port address decoders which can decode 32 unique contiguous port addresses. Several of these are used to select the channels of the USART, PIO and CTC. Additional decoded port select signals are available for various peripherals attached to the system.
- 19.6608MHz crystal oscillator divided to 2.457MHz for Z-80 CPU operation and dividable by Z-80 CTC to provide the serial I/O baud rates or any other desired system frequencies.
- Bus drivers are provided for memory and I/O expansion to other boards that are a part of the series.
- Optional 1K byte monitor software has terminal handler, load and punch routines as well as set and display memory commands. A Go command begins execution of user programs. There are debug aids such as set and display registers and breakpoints. The 3K byte version includes a floppy disk controller and even more debug capacity.
- Tri-state buffers on all data, address and control lines.
- One nonmaskable and three maskable interrupts.

MCB Block Diagram



Summary of Z-80 CPU Instruction Set

ADD REGISTER/MEMORY/VALUE TO A/WITH CARRY
ADD REGISTER PAIR TO HL/IX/IY
ADD REGISTER PAIR TO HL WITH CARRY
SUBTRACT REGISTER/MEMORY/VALUE FROM A/ WITH BORROW
SUBTRACT REGISTER PAIR FROM HL WITH BORROW
DECIMAL ADJUST

AND REGISTER/MEMORY/VALUE WITH A
OR REGISTER/MEMORY/VALUE WITH A
EXCLUSIVE OR REGISTER/MEMORY/VALUE WITH A
COMPARE REGISTER/MEMORY/VALUE WITH A
COMPARE/INCREMENT/DECREMENT/HL, DECREMENT BC/REPEAT
INCREMENT/DECREMENT REGISTER/REGISTER PAIR MEMORY
COMPLEMENT A/2'S/1'S/

LOAD REGISTER/REGISTER PAIR/MEMORY WITH REGISTER/REGISTER PAIR/MEMORY/VALUE
LOAD (DE) WITH (HL), INC/DEC/HL AND DE, DEC BC/REPEAT

ROTATE RIGHT/LEFT, CIRCULAR/THRU CARRY/ REGISTER/MEMORY
ROTATE RIGHT/LEFT DIGIT
SHIFT REGISTER/MEMORY, LEFT ARITHMETIC
SHIFT REGISTER/MEMORY, RIGHT LOGICAL

SET/RESET/TEST BIT IN REGISTER/MEMORY

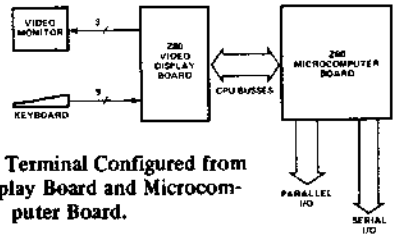
CALL SUBROUTINE IF CONDITION TRUE/UNCONDITIONAL CALL
JUMP IS CONDITION TRUE/UNCONDITIONAL JUMP
JUMP RELATIVE IF CONDITION TRUE/UNCONDITIONAL RELATIVE JUMP
JUMP TO (HL), (IX), OR (IY)
RESTART CALL
EXCHANGE REGISTERS
EXCHANGE REGISTER PAIR AND STACK
RETURN FROM SUBROUTINE IF CONDITION TRUE/ UNCONDITIONAL RETURN
POP/PUSH REGISTER PAIR FROM/TO STACK
SET INTERRUPT MODE 0/1/2
ENABLE/DISABLE INTERRUPTS
RETURN FROM INTERRUPT/NMI INTERRUPT
DECREMENT B, JUMP IF B = 0

INPUT/OUTPUT REGISTER
INPUT/OUTPUT, (HL), INC (DEC) HL, DEC B/REPEAT

SET/COMPLEMENT CARRY FLAG
NO OPERATION
HALT

MCB Applications

The Z-80 MCB can be used in many applications traditonally not available to single microcomputer boards. Many of the functions previously performed by external hardware are now implemented in the Z-80 CPU and peripherals. The performance per unit board area has been greatly optimized, thus eliminating the need for extra cards, card cages, back planes, and connectors for many applications. The Z-80 MCB can be used for machine controllers, customized small business computers, automated test stands, customized data acquistion systems, process control systems, communications or display controllers, multiprocessor systems, and word processing systems, just to name a few. The diagram below shows the Z-80 MCB used with the Zilog Video Display Board to configure an intelligent terminal.



Intelligent Terminal Configured from Video Display Board and Microcomputer Board.

Specifications

POWER SUPPLY:	+5V DC ±5%, current: 2 amps max (with 3 PROMs)
CONNECTOR:	122-pin edge (100 mil spacing) Augat PN 14005-19P1
SIZE:	Length, 7.7" Depth, 7.5" Spacing, 0.5" centers
ENVIRONMENTAL:	0° – 50°C temperature range. Up to 90% humidity without condensation.
MEMORY CAPACITY:	4K or 16K bytes dynamic RAM plus up to 4K bytes PROM, ROM or EPROM. Expandable by use of Z-80 RMB RAM board to 64K bytes of main memory.
I/O CHANNELS:	Serial I/O port with RS-232 or 20 mA current loop interface; Two (2) software configurable bidirectional 8-bit parallel I/O ports.

Pin Assignments

MCB – PIN OUT (COMPONENT SIDE)

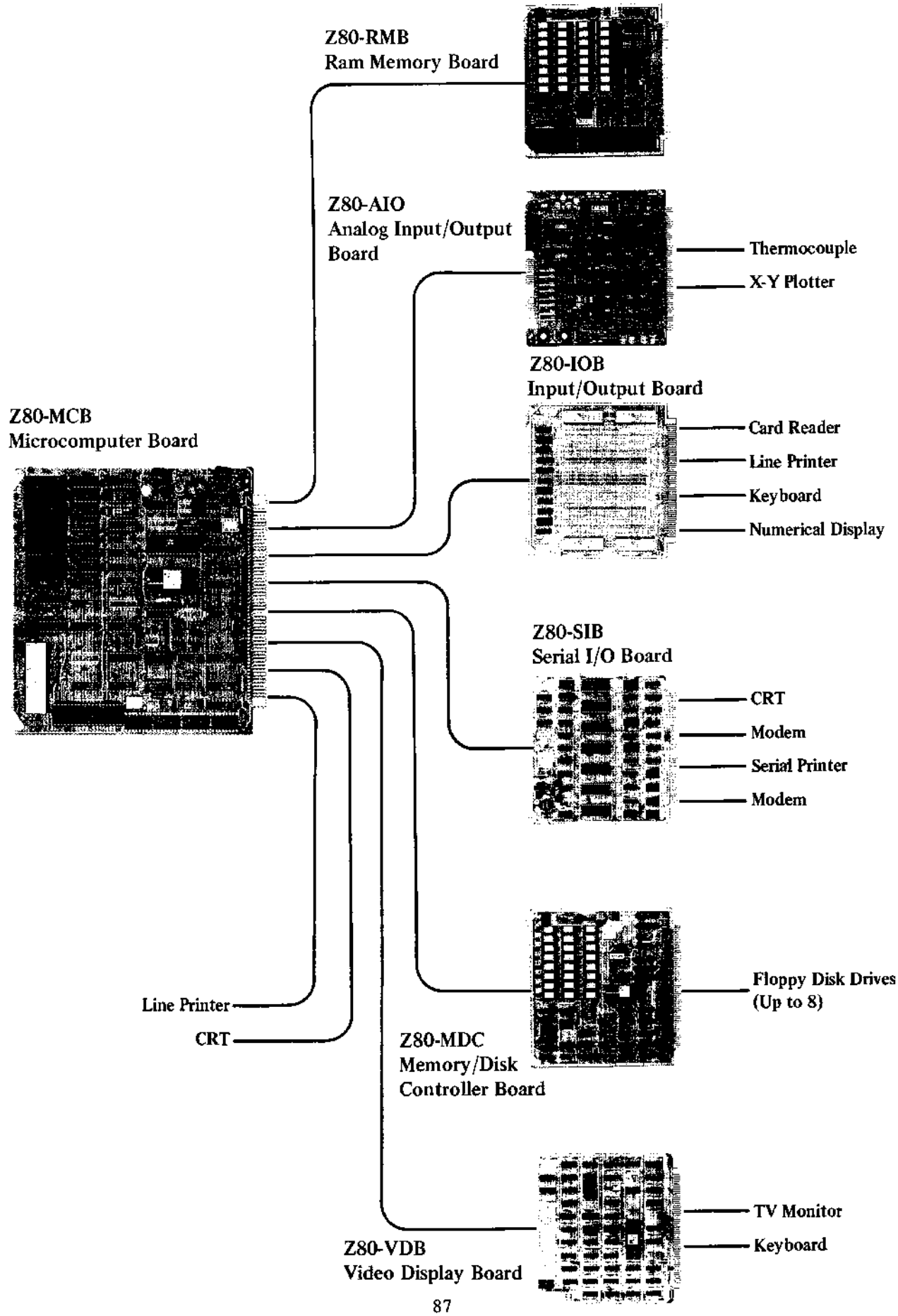
1	+5	34	I/O SUB GROUPΦ–
2	+5	35	RFSH–
3	+5	36	ADDRESS BUS 13
4	IORQ–	37	ADDRESS BUS 11
5	DATA BUS 5	38	OPEN (Z 21) (PULL UP)
6	20 mA DATA	39	OPEN (Z 20)
7	RECEIVE DATA	40	OPEN (Z 19)
8	DATA BUS 3	41	OPEN (Z 18)
9	MASTER RESET	42	OPEN (Z 17)
10	MASTER RESET–	43	OPEN (Z 16)
11	CLEAR TO SEND	44	OPEN (Z 15)
12	DATA BUS 6	45	OPEN (Z 14)
13	DATA BUS Φ	46	OPEN (Z 13)
14	REQ TO SEND	47	OPEN (Z 12)
15	XMITED DATA	48	OPEN (Z 11)
16	MEM SEL IN	49	OPEN (Z 10)
17	DISK C/T	50	OPEN (Z 9)
18	C/T 2	51	OPEN (Z 8)
19	20 mA DATA RET	52	OPEN (Z 7)
20	TTY TAPE CNTL RET	53	OPEN (Z 6)
21	MEMORY SEL OUT	54	OPEN (Z 5)
22	INTE IN CTC	55	OPEN (Z 4)
23	WR–	56	OPEN (Z 3)
24	USER STRB 3	57	OPEN (Z 2)
25	DISK STRB	58	OPEN (Z 1)
26	ADDRESS BUS 7	59	+5
27	ADDRESS BUS 8	60	+5
28	IOWR–	61	+5
29	ADDRESS BUS 5		
30	ADDRESS BUS 6		
31	RESET–		
32	ADDRESS BUS 15		
33	I/O SUB GROUP 2–		

NOTE: Open pins are definable by user. One is pulled up for use as a source for interrupt enable daisy chain.

MCB – PIN OUT (SOLDER SIDE)

62	GND	93	8Φ–
63	GND	94	ADDRESS BUS 14
64	GND	95	I/O SUB GROUP 3–
65	TTY TAPE CNTL	96	I/O SUB GROUP 1–
66	–5V EXTERNAL	97	ADDRESS BUS 12
67	–5V EXTERNAL	98	ADDRESS BUS 4
68	DATA BUS 4	99	Φ–
69	+12V EXTERNAL	100	ADDRESS BUS 3
70	+12V EXTERNAL	101	ADDRESS BUS 2
71	DATA BUS 2	102	ADDRESS BUS 1
72	–12V EXTERNAL	103	ADDRESS BUSΦ
73	DATA BUS 7	104	I/O GROUPΦ–
74	DATA SET READY	105	I/O GROUP 1–
75	DATA BUS 1	106	I/O GROUP 2–
76	DATA TERM. RDY/ XMIT CLK	107	I/O GROUP 3–
		108	I/O GROUP 4–
77	20mA RECV. RETN./ RECV. CLK	109	BUS RQ–
		110	NMI–
78	SYNC DETECT	111	PIO INTE OUT
79	INT–	112	SERIAL CLK IN (2X)
80	LINE SIGNAL DET.		
81	20mA RECV.	113	HALT–
82	USER C/T 3	114	INTE IN PIO
83	ROM SEL OUT	115	M 1–
84	USER RTC C/T	116	RD–
85	MRQ–	117	INTE IN SER
86	CTC INTE OUT	118	½Φ
87	2X SERIAL CLK	119	WAIT–
88	BUSAK–	120	GND
89	ADDRESS BUS 9	121	GND
90	IORD–	122	GND
91	ADDRESS BUS 10		
92	ROM SEL IN–		

MCB Performs Supervisory Control in Z-80 Based Systems

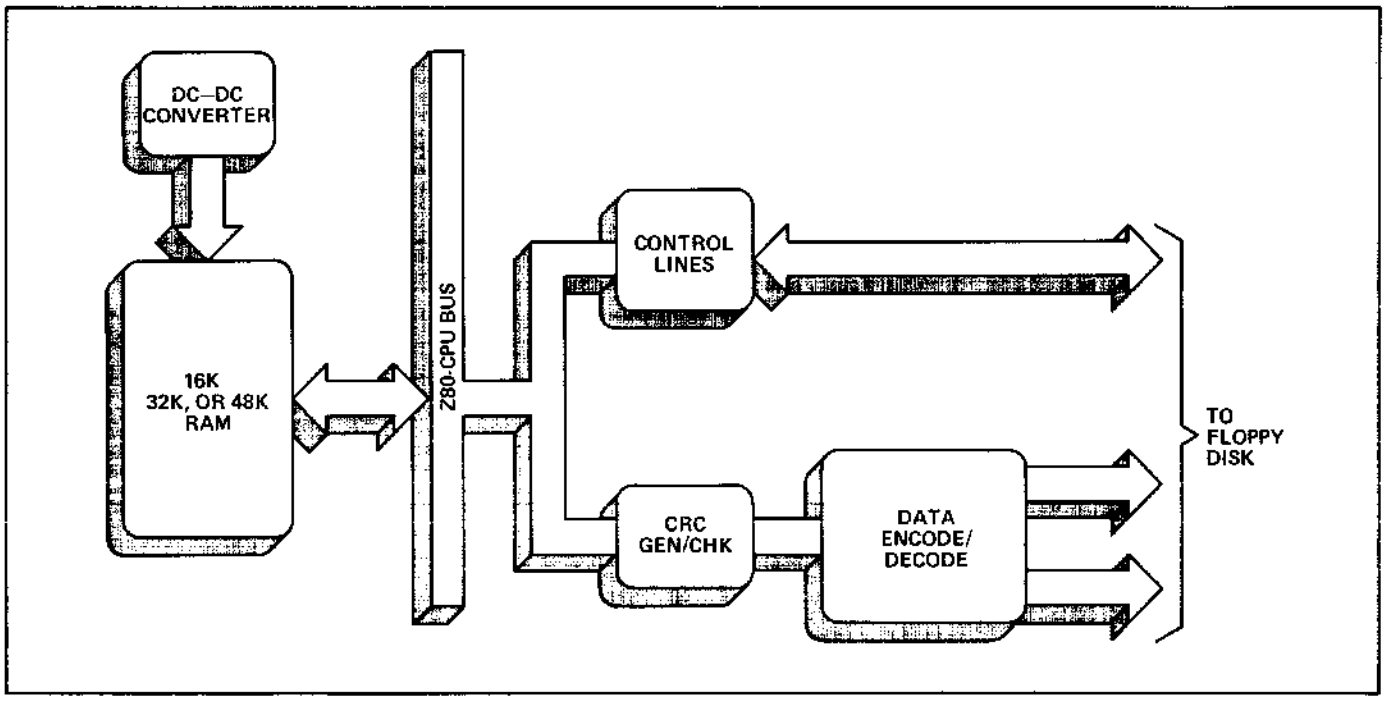


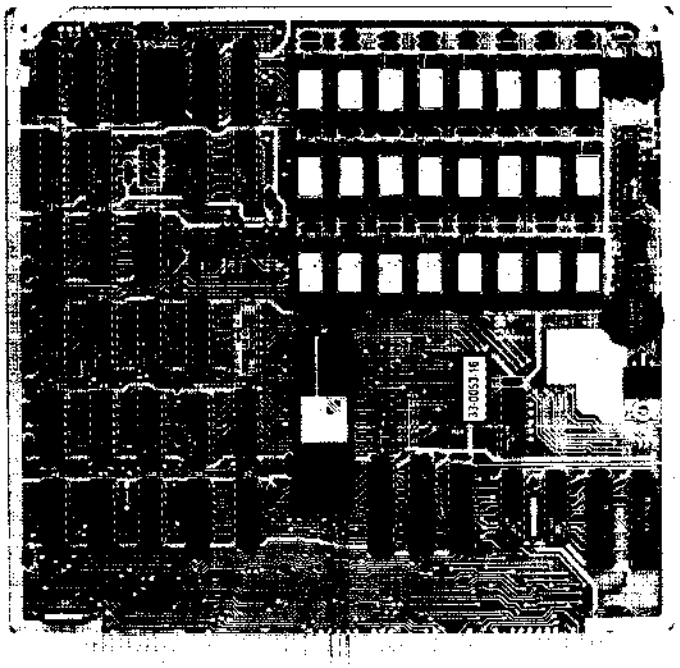
Zilog

Product Specification

Z-80® MDC Memory Disk Controller

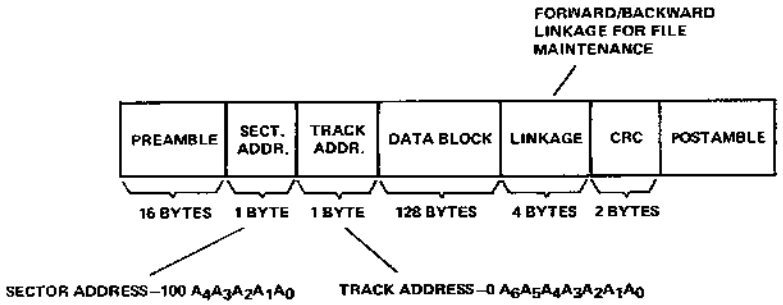
The Z-80 Memory Disk Controller provides the Z-80 Microcomputer Board series and MCZ-1 System with all data formatting required for reading and writing onto a floppy disk system from RAM storage. Up to 8 floppy disks may be controlled by one MDC. The MDC also provides up to 48K bytes of dynamic RAM memory for program or data storage. The card interfaces directly to all other boards in the MCB series.





Features

- Recommended configuration of MDC and two Shugart 801R Floppy Disk Drives provides 600K bytes of peripheral mass storage.
- The MDC is designed to support a Z-80 MCB and provide a minimum microcomputer system. In addition to the floppy disk controller the MDC contains up to 48K bytes of dynamic RAM memory.
- Only a single +5v power supply is required. Voltage levels of +12v and -5v needed by the dynamic memory array are provided by a voltage converter on the board.
- PROM memory mapping allows memory to be mapped in 4K bytes to one of 16 starting address locations.
- Memory array utilizes 16K bit dynamic RAM devices.
- Disk read/write accuracy ensured by 16 bit Cyclic Redundancy Check (CRC) code circuitry.
- WAIT control logic synchronizes the speed of the CPU to the disk read or write speed by setting the CPU WAIT line to an active state until a complete byte of data has been assembled.
- MDC employs a Z-80 PIO device to control the disk drives and provide status information to the CPU.
- Data is recorded onto the floppy diskette in a serial format. The necessary parallel-to-serial and serial-to-parallel data conversion is performed by circuitry on the MDC. Frequency modulation is the recording mode where each data bit recorded on the diskette has an associated clock bit recorded with it.
- Formatting of serial data into the disk is accomplished under program control by the Z-80 CPU on a MCB Microcomputer Board. Optional PROM-based firmware is available from Zilog to control up to eight Shugart 801R Floppy Disk Drives. This firmware assumes that 32 data sectors (records) are utilized per track and 77 tracks are utilized per disk. The firmware provides all control functions for the disk and performs all data transfer. The Sector Data Format is illustrated below:



Pin Assignments

MDC—PIN OUT (COMPONENT SIDE)

1	+5v	21	OPEN	41	OPEN
2	+5v	22	OPEN	42	OPEN
3	+5v	23	OPEN	43	OPEN
4	IORQ–	24	OPEN	44	OPEN
5	DATA BUS 5	25	OPEN	45	OPEN
6	OPEN	26	ADDRESS BUS 7	46	OPEN
7	DR 1–	27	ADDRESS BUS 8	47	OPEN
8	DATA BUS 3	28	OPEN	48	OPEN
9	STEP–	29	ADDRESS BUS 5	49	OPEN
10	SEL 3–	30	ADDRESS BUS 6	50	OPEN
11	WRITE GATE–	31	OPEN	51	OPEN
12	DATA BUS 6	32	ADDRESS BUS 15	52	OPEN
13	DATA BUS 0	33	READY–	53	OPEN
14	DR 6–	34	DRIVE PRESENT–	54	OPEN
15	DR 5–	35	REFRESH–	55	OPEN
16	OPEN	36	ADDRESS BUS 13	56	OPEN
17	START SECTOR	37	ADDRESS BUS 11	57	OPEN
18	DR 0	38	OPEN	58	OPEN
19	SECTOR–	39	OPEN	59	+5v
20	OPEN	40	OPEN	60	+5v
				61	+5v

MDC—PIN OUT (SOLDER SIDE)

62	GND	82	SEL 0–	102	ADDRESS BUS 1
63	GND	83	ROM SEL OUT–	103	ADDRESS BUS 0
64	GND	84	OPEN	104	OPEN
65	OPEN	85	MEM REQUEST	105	OPEN
66	READ DATA–	86	SEL 2–	106	OPEN
67	TRACK 0–	87	OPEN	107	I/O GROUP 3–
68	DATA BUS 4	88	OPEN	108	I/O GROUP 4–
69	SPARE OUT–	89	ADDRESS BUS 9	109	OPEN
70	DIRECTION–	90	OPEN	110	OPEN
71	DATA BUS 2	91	ADDRESS BUS 10	111	OPEN
72	OPEN	92	OPEN	112	OPEN
73	DATA BUS 7	93	8 X CLOCK	113	OPEN
74	DR 3–	94	ADDRESS BUS 14	114	MDC PIO INT IN
75	DATA BUS 1	95	SUBG3–	115	M1–
76	DR 2–	96	WRITE PROTECT	116	READ
77	WRITE DATA–	97	ADDRESS BUS 12	117	OPEN
78	DR 1–	98	ADDRESS BUS 4	118	OPEN
79	OPEN	99	CLOCK–	119	WAIT–
80	SEL 1–	100	ADDRESS BUS 3	120	GND
81	DR 4–	101	ADDRESS BUS 2	121	GND
				122	GND

Specifications

POWER SUPPLY: +5 VDC ±5%
1.6 amps maximum current

CONNECTOR: 122-pin edge (100 mil spacing)

SIZE: Length, 7.7"
Depth, 7.5"
Spacing, 0.5" centers

ENVIRONMENTAL: 0° – 50° C temperature range. Up to 90% humidity without condensation.

MEMORY CAPACITY: 16K, 32K, or 48K bytes dynamic RAM. Memory may be organized into 4K pages with each page assigned to any of 16 possible starting addresses.

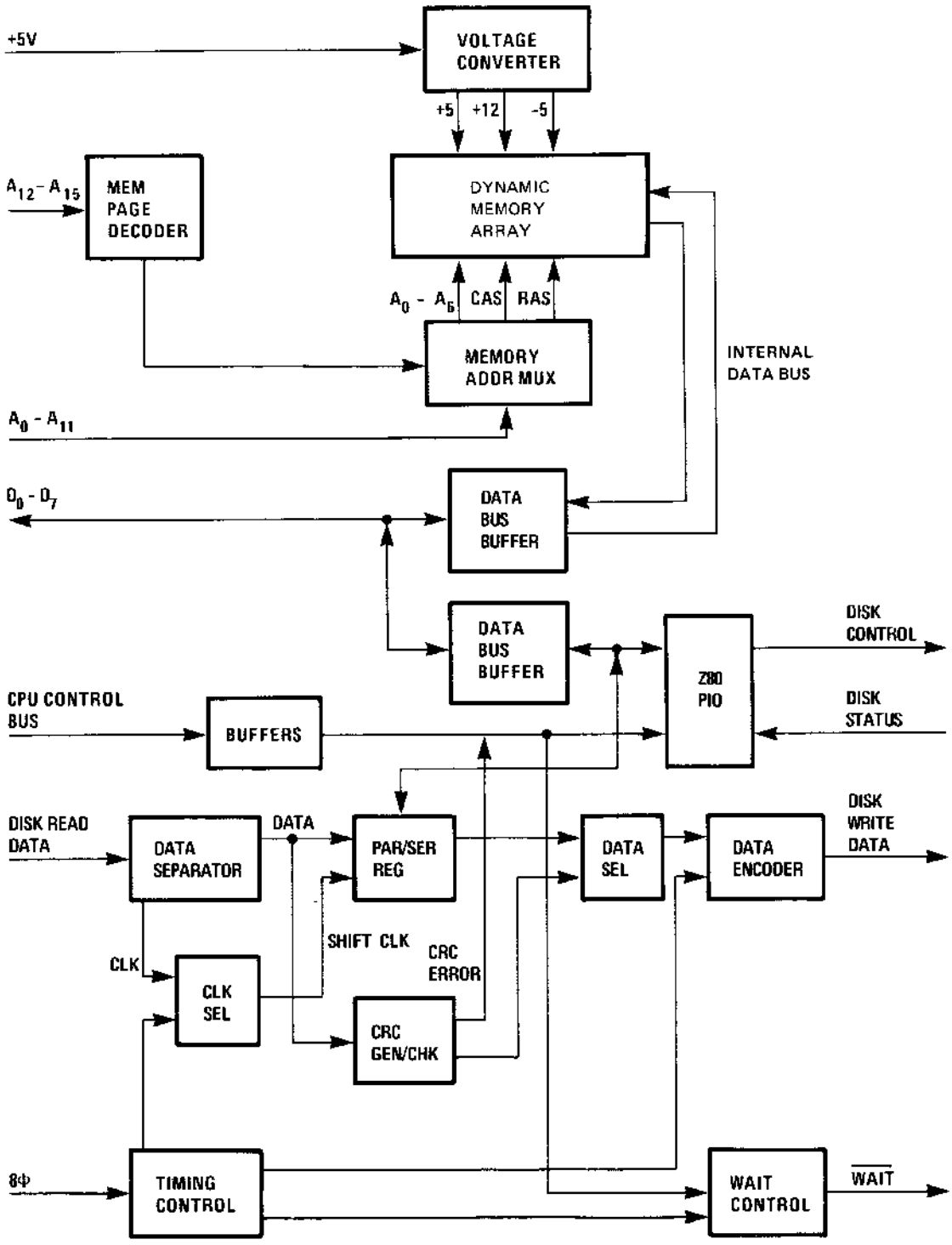


FIGURE 1
Z-80 MDC BLOCK DIAGRAM

ADDITIONAL INFORMATION

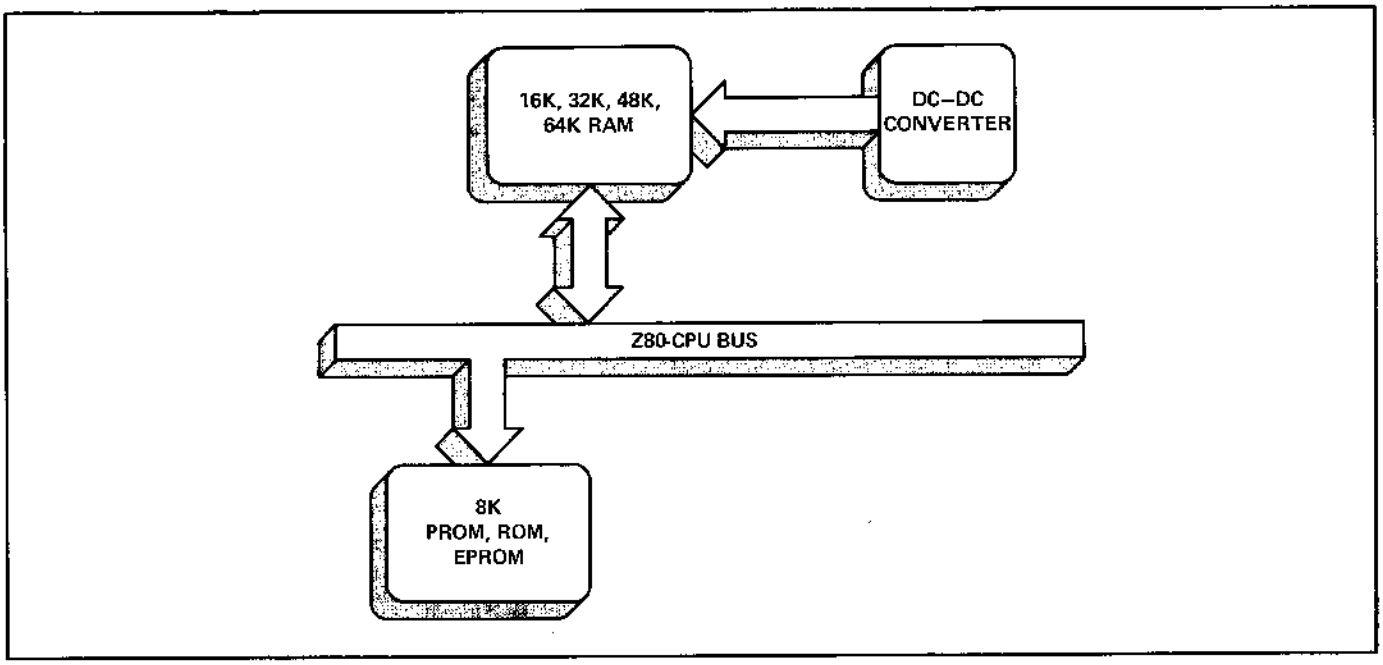
A complete set of Hardware and Software User's Manuals is available for use with the MCB. These manuals contain both logic and schematic diagrams and give detailed information on the operation of the MCB. Technical manuals on the Z80-CPU, PIO and CTC devices are also available.

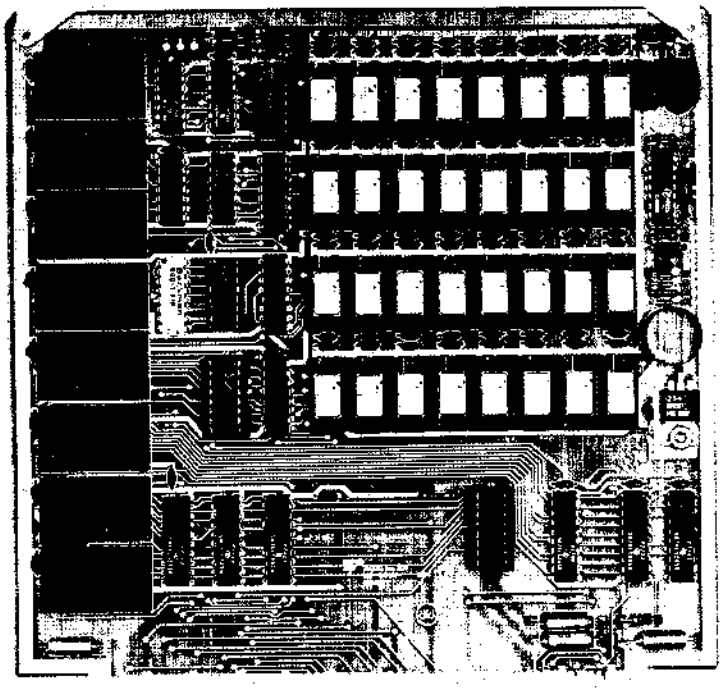
Zilog

Product Specification

Z-80® RMB RAM/ROM Memory Board

The Z-80 RMB RAM/ROM Memory Board provides the Zilog Microcomputer Board Series with additional memory capacity, consisting of 16K, 32K, 48K, or 64K bytes of dynamic RAM and sockets for up to 8K bytes of ROM, PROM, or EPROM.





Features

- 16K, 32K, 48K, or 64K bytes of dynamic RAM per board utilizing 16K bit dynamic RAMs.
- Sockets for up to eight ROM or PROM devices per board. These sockets are intended for 24-pin packages in either a 512 x 8 or a 1K x 8 configuration. Pin compatible PROMs include: 6341, 6381, 82S181, 2704, 2708, 2716. NOTE: For EPROMs an external -5 and +12 volt power supply is required.
- PROM decoder allows each 4K segment of memory to have a starting address at any of 16 boundaries within the 64K of addressable memory space.
- Only a single +5 volt supply is required. An on-board voltage converter is included to provide the +12 volt and -5 volt supplies needed by the dynamic RAM devices.

Specifications

POWER SUPPLY:	+5 VDC ±5% 1.6 amps maximum current
CONNECTOR:	122-pin edge (100mil spacing)
SIZE:	Length 7.7" Depth 7.5" Spacing 0.5" centers
ENVIRONMENTAL:	0° – 50° C temperature range. Up to 90% humidity without condensation.
MEMORY CAPACITY:	16K, 32K, 48K, or 64K bytes dynamic RAM. Each 4K page may have its starting address assigned to any of 16 possible values. Also sockets on board for up to 8K bytes of non-volatile memory (ROM, PROM, or EPROM).

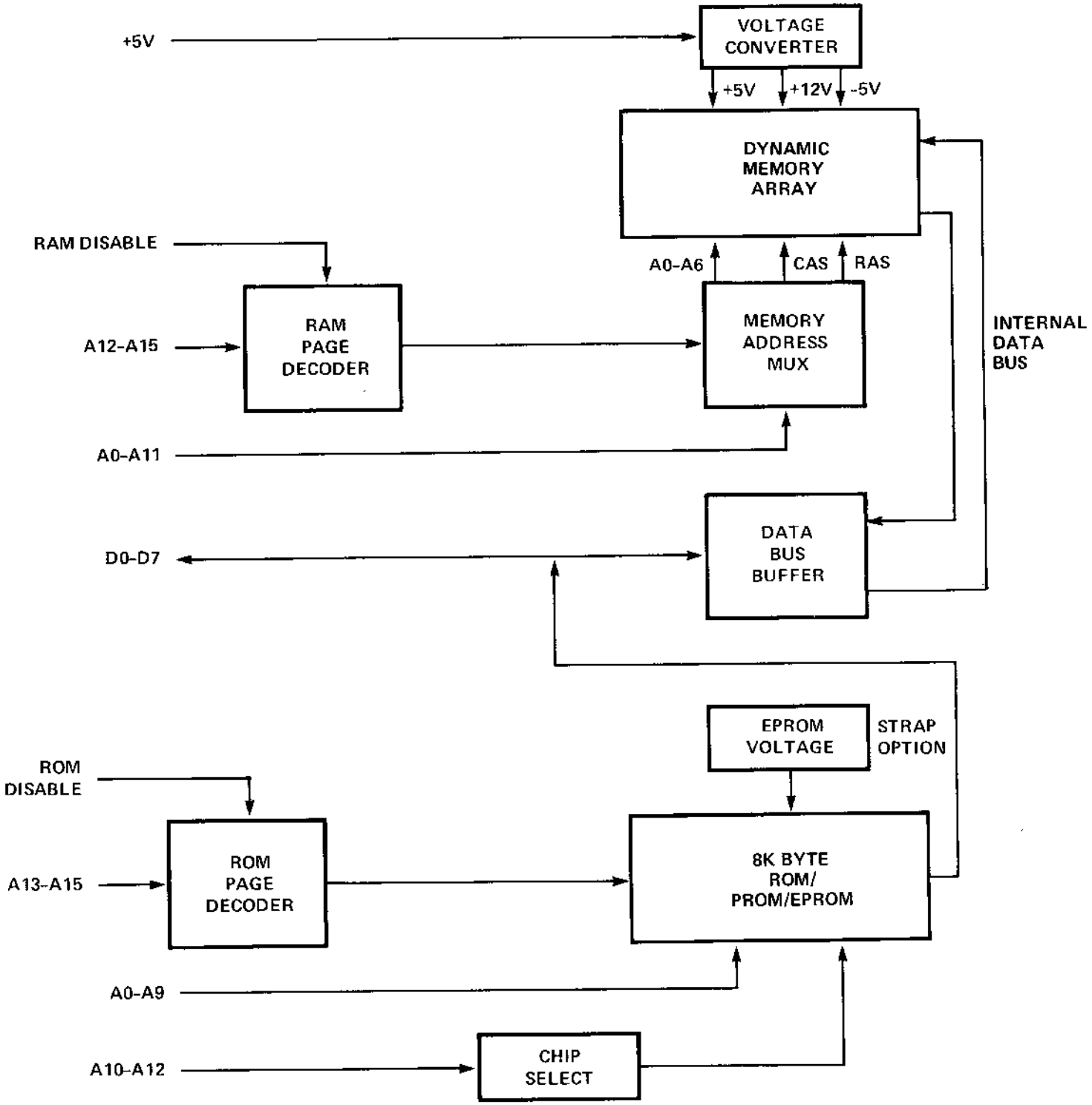
Pin Assignments

RMB–PIN OUT (COMPONENT SIDE)

1	+5v	29	ADDRESS BUS 5
2	+5v	30	ADDRESS BUS 6
3	+5v	32	ADDRESS BUS 15
5	DATA BUS 5	35	RFSH –
8	DATA BUS 3	36	ADDRESS BUS 13
12	DATA BUS 6	37	ADDRESS BUS 11
31	DATA BUS Ø	59	+5v
23	WRITE –	60	+5v
26	ADDRESS BUS 7	61	+5v
27	ADDRESS BUS 8		

RMB–PIN OUT (SOLDER SIDE)

62	GND	97	ADDRESS BUS 12
63	GND	98	ADDRESS BUS 4
64	GND	100	ADDRESS BUS 3
66	–5v	101	ADDRESS BUS 2
67	–5v	102	ADDRESS BUS 1
68	DATA BUS 4	103	ADDRESS BUS Ø
69	+12v	108	RAM DISABLE –
70	+12v	116	READ –
71	DATA BUS 2	117	ROM DISABLE –
73	DATA BUS 7	120	GND
75	DATA BUS 1	121	GND
85	MEM REQUEST –	122	GND
89	ADDRESS BUS 9		
91	ADDRESS BUS 10		
94	ADDRESS BUS 14		



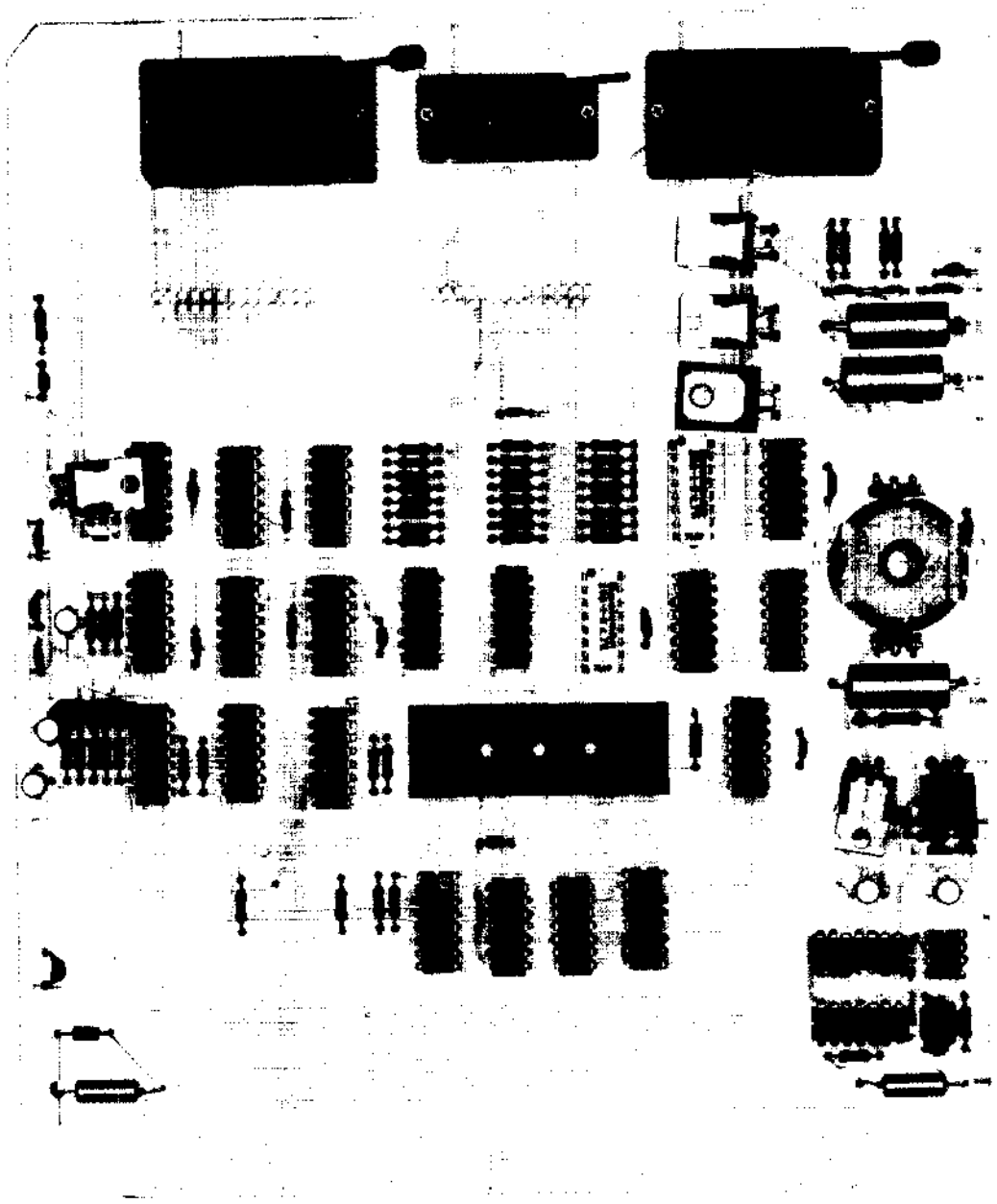
Z-80 RMB BLOCK DIAGRAM

Zilog

Product Specification

Z-80® PPB PROM Programmer Board

The Zilog PROM Programmer Board (PPB) provides the capability to program 16 and 24-pin bipolar PROMs and 24-pin EPROMs. The board incorporates zero insertion force sockets and power conversion to permit a single +5 volt power supply operation. System bus interface is implemented by the Z-80 PIO and is simply addressed as I/O with the device address of EØH. The compatible software utility ZPROG provides the capability to program from a disk file and verify PROM contents from the disk file, to list the PROM contents onto a terminal or printer, and to duplicate a PROM.



Operation

The I/O address EØ is decoded by the ADDRESS DECODE to enable the PIO. The two Least Significant Bits of the ADDRESS BUS are used to select port A or B and configure the PIO for control or data transmission. Port A is set in the 'bidirectional' mode for reading and writing PROM data in the VERIFY and PROGRAMMING mode respectively. Port B is set in the 'output' mode to control the ADDRESS COUNTER, CHIP SELECT / BUS CONTROL and PROGRAM PULSE circuitry. The ADDRESS COUNTER is cleared and clocked by PIO lines B_0 and B_1 respectively, and produces ten address lines to the programming socket. The CHIP SELECT and PROGRAM PULSE: EPROM are controlled by PIO lines B_2 and B_3. The programming pulse width is a function of the ZPROG utility and is set in software for the EPROM device being programmed. The BUS CONTROL circuitry accomodates the reading of PROM contents or writing to PROM via PIO Port A and is controlled by PIO line B_6. Bipolar devices are programmed by enabling the high voltage drivers on the data bus with the chip select ($\overline{CE}$) controlled by PIO line B_2 and generating the PROGRAM PULSE: BIPOLAR from the PIO line B_5. This pulse width is also a function of the ZPROG utility.

Features

- Programs 16 and 24-pin bipolar PROMs: 7610, 7611, 7620, 7621, 7640, 7641, and 24-pin EPROMs: 2704, 2708
- Programs 7603 PROMs using supplied adapter.
- Program from a disk file
- Verify PROM contents from a disk file
- List PROM contents to a terminal
- Duplicate a PROM
- Zero insertion force sockets
- Single +5V power supply
- Compatible with all MCB based systems and Development system.

Specifications

POWER SUPPLY:	+5 VDC ±5% 1.5 Amps (standby) 2.5 Amps (programming)
CONNECTOR:	122-pin edge (100 mil spacing)
SIZE:	Length: 7.7" Depth: 7.5" Spacing: 0.5" centers
ENVIRONMENTAL:	0° – 50°C temperature range Up to 90% humidity

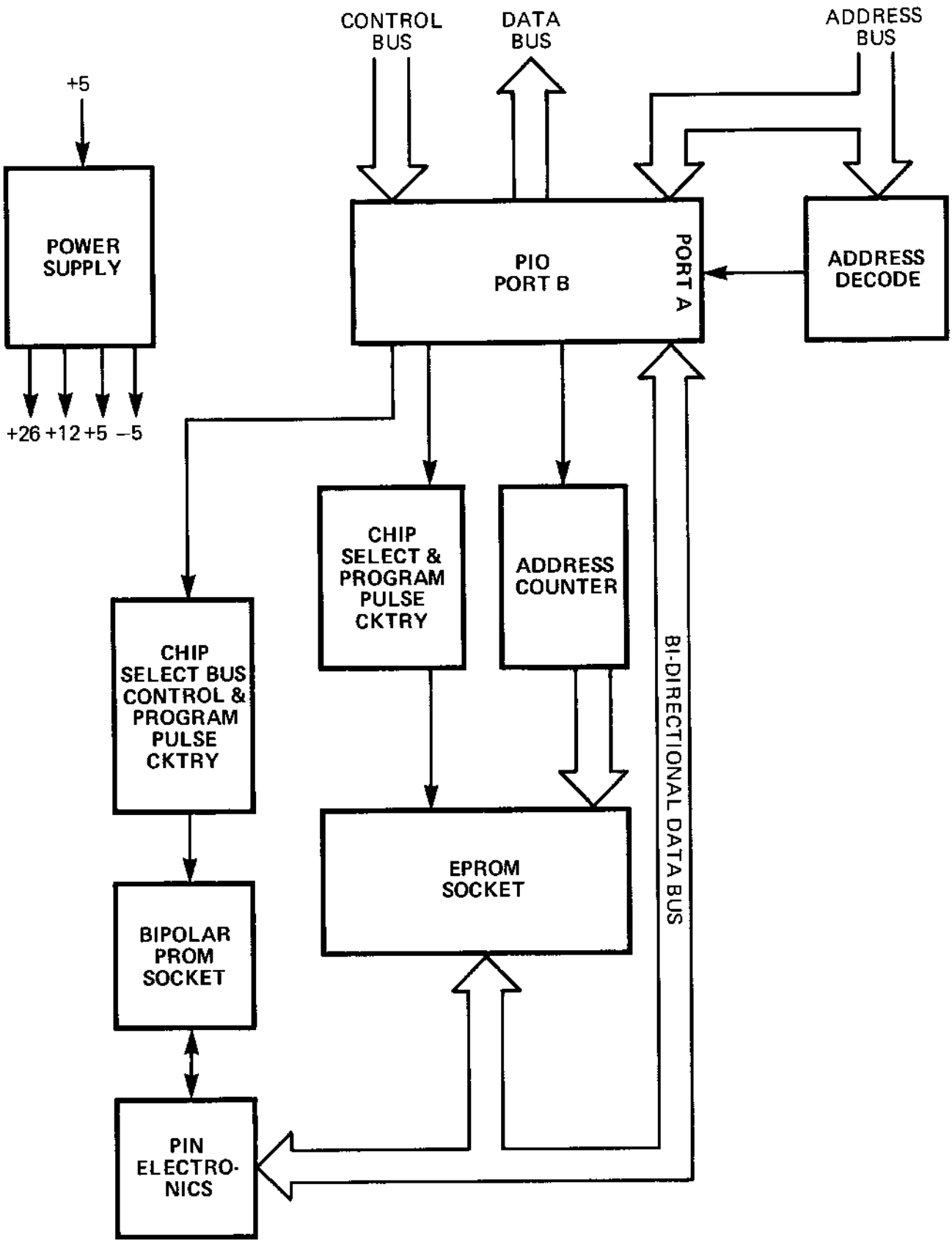
Pin Assignment

ZDS DEVELOPMENT SYSTEM VERSION

9	ADDRESS BIT 6	47	DATA BIT 4
10	ADDRESS BIT 7	48	DATA BIT 5
16	ADDRESS BIT Ø	51	DATA BIT 6
17	ADDRESS BIT 1	52	DATA BIT 7
18	ADDRESS BIT 2	54	DATA BIT Ø
19	ADDRESS BIT 3	55	DATA BIT 1
20	ADDRESS BIT 4	56	DATA BIT 3
21	ADDRESS BIT 5	57	DATA BIT 2
24	$\overline{Ø}$	88	$\overline{IEI}$
28	$\overline{M1}$	109	$\overline{INT}$
30	$\overline{IORQ}$	112	$\overline{IEO}$
45	$\overline{RD}$		

MCZ VERSION

4	$\overline{IORQ}$	73	DATA BIT 7
5	DATA BIT 5	75	DATA BIT 1
8	DATA BIT 3	79	$\overline{INT}$
12	DATA BIT 6	98	ADDRESS BIT 4
13	DATA BIT Ø	99	$\overline{Ø}$
26	ADDRESS BIT 7	100	ADDRESS BIT 3
29	ADDRESS BIT 5	101	ADDRESS BIT 2
30	ADDRESS BIT 6	102	ADDRESS BIT 1
50	$\overline{IEI}$	103	ADDRESS BIT Ø
51	$\overline{IEO}$	115	$\overline{M1}$
68	DATA BIT 4	116	$\overline{RD}$
71	DATA BIT 2		



PPB Block Diagram

Zilog PROM Programmer Utility

The ZILOG PROM Programmer Utility (ZPROG), operates with the ZDS-PPB, MCZ-PPB, ZDS-PPB/16, MCZ-PPB/16 or PROLOG Interface (CIB) boards.

ZPROG provides an environment in which the user can:

- –Duplicate PROMs of the same type
- –Duplicate PROMs of different types
- –Write the contents of one or more PROMs to a disk file
- –Program PROMs from a disk file
- –List the contents of a PROM on the console
- –Display and modify any byte of a PROM

ZPROG COMMANDS

The following parameter definitions apply to the ZPROG commands:

pname	a PROM name from the applicable PROMs.
begadr	specifies the address at which the command should begin reading or programming.
numbytes	specifies the number of bytes to read or program.
U,L	specifies which nibble (upper or lower) is to be read or programmed.
M	specifies that the user wishes to enter modify mode to modify the buffer after reading from the source PROM, and before programming the destination.
V	specifies that the contents of the destination is verified against the source and the programming step skipped.
entadr	specifies the entry address of the file to be created (File command only).
reclen	specifies the record length of the file to be created (File command only).

Summary of Commands

The following notation is used in the command description.

- | is used to denote 'or', e.g., pname|* means either a pname or an asterisk '*' may be used.
- [ ] is used to denote an optional parameter, e.g., [M] means 'M' may be included in the command line or left out.

All commands can be abbreviated by giving the first letter only. Commands and options can be entered in either upper or lower case except in the PROM names which must be entered as listed.

AGAIN	The AGAIN command is used to repeat the previous command.
BYTE	pname|* The BYTE command is used to read or program one byte of a PROM in the same manner in which the buffer is displayed in modify mode.
COPY	S: pname|* [B=begadrl] [N=numbytes] [U|L] D: pname|* [B=begadr2] [N=numbytes2] [U|L] [M] [V] The COPY command is used to copy (or verify) the contents of one PROM to another PROM which may have different attributes.
DUPLICATE	pname|* [B=begadr] [N=numbytes] [M] [V] The DUPLICATE command is used to copy (or verify) the contents of one PROM to another PROM of the same type.
FILE	pname|* filename [B=begadr] [N=numbytes] [U|L] [E=entadr] [RL=reclen] The FILE command is used to copy the contents of a PROM to a RIO file.
LIST	pname|* [B=begadr] [N=numbytes] The LIST command is used to list the contents of a PROM to CONOUT (Logical Unit for console output).
NEXT	The NEXT command is used to repeat the previous command (with begadr adjusted for the program command).
PROGRAM	filename pname|* [B=begadr] [N=numbytes] [U|L] [V] The PROGRAM command is used to copy the contents of a RIO procedure type file to a PROM.
QUIT	The QUIT command is used to exit the ZPROG environment and return to the RIO executive. Any open files are closed.

Zilog

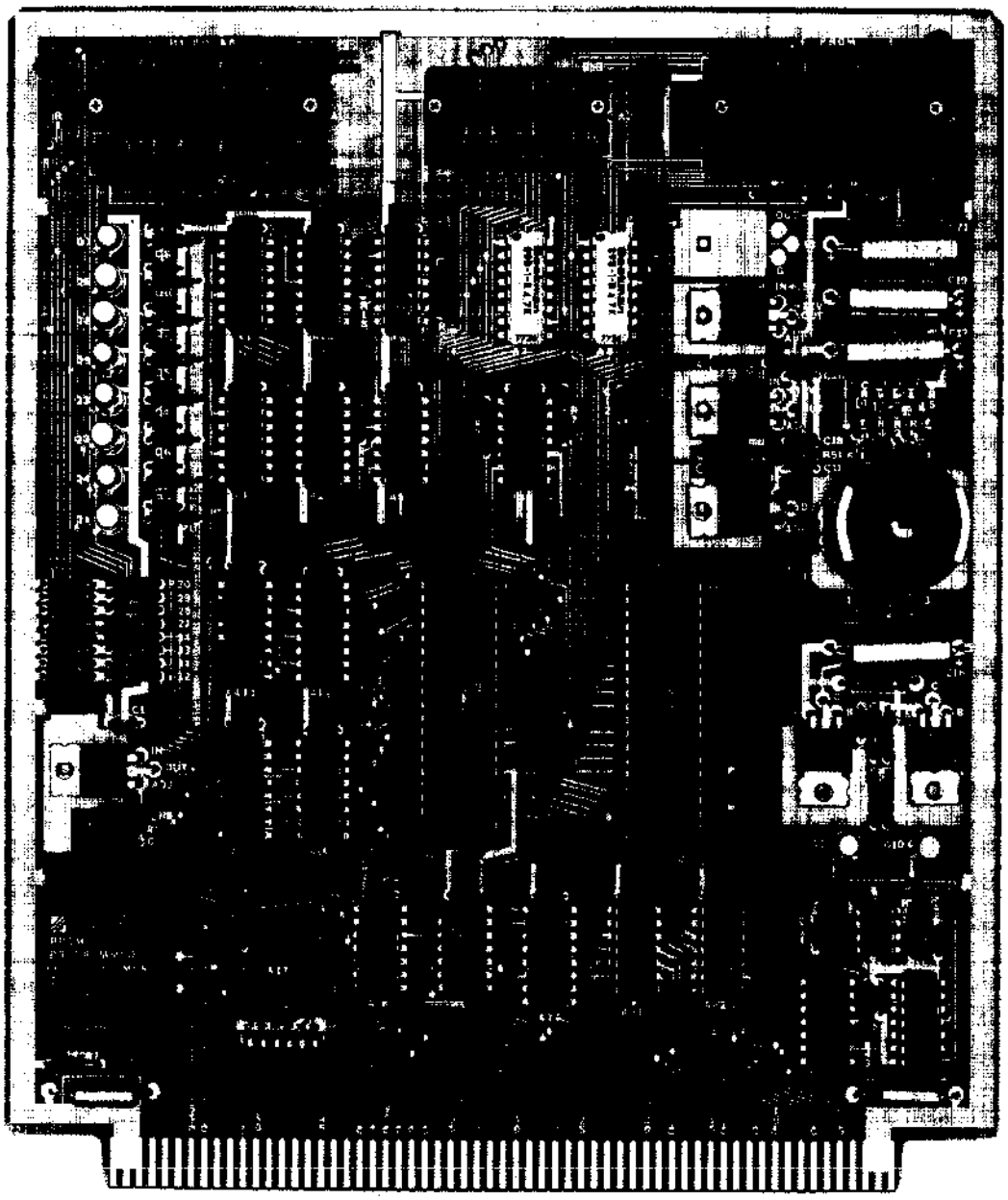
Product Specification

Z-80® PPB/16 PROM Programmer Board

The Z-80 PPB/16 provides the ZDS and MCZ systems with PROM programming capability for the following PROMs:

2716

82S126 82S129 82S130 82S131
82S140 82S141 82S180 82S181
82S2708



Description

The PPB/16 incorporates three zero insertion force sockets that are mounted at the top of the card. The card is inserted into the appropriate system and extends beyond the end of the card cage to allow easy access to the sockets. One 24-pin socket is used for the 2K x 8 EPROM (2716). Another 24-pin socket is used for the 1K x 8 PROMs and a 16-pin socket for 1K x 4 PROMs. PROMs not listed above may also be programmed provided their pinout and programming procedure matches that for the above EPROMs and PROMs.

Each card is a completely self-contained unit requiring only a +5 volt supply. An on-board converter transforms the +5 volts to the voltage required by the PROMs and EPROMs. All communication with the Z-80 CPU is done through the Z-80 PIO peripheral interface via a 122-pin edge connector.

PROM Programmer Utility (ZPROG)

The Zilog PROM Programmer Software Utility (ZPROG) is provided on the RIO operating system utilities diskette. ZPROG enables the user to:

- Duplicate PROMs of the same type
- Write the contents of one or more PROMs to a disk file
- Program PROMs from a disk file
- List the contents of a PROM on the console
- Display and modify any byte of a PROM

Features

- Two board versions—the MCZ-PPB/16 and ZDS-PPB/16
- Z-80 control for microprocessor flexibility
- Quick load, zero insertion force, PROM sockets
- Power from single +5 volt power supply utility
- ZPROG included with RIO operating system
- Unique Program/User interface with prompt statements
  READY TO READ
  READY TO PROGRAM?
  READY TO VERIFY?
- Program, List Duplicate and Verify modes of operation

Edge Connector Pin Assignments (MCZ Systems)

PIN	DESCRIPTION
1, 3, 59–61	+5V P.S.
4	IORQ–
5	DB5
8	DB3
12	DB6
13	DB0
26	AB7
29	AB5
30	AB6
50	IEI · PPB · PIO
51	IEO · PPB · PIO
62–64, 120–122	GND
68	DB4
71	DB2
73	DB7
75	DB1
79	INT–
98	AB4
99	Φ–. (System Clock–)
100	AB3
101	AB2
102	AB1
103	AB0
115	M1–
116	RD–

Edge Connector Pin Assignments (ZDS Systems)

PIN	DESCRIPTION
1–3, 59–61	+5V
9	A6
10	A7
16	A0
17	A1
18	A2
19	A3
20	A4
21	A5
24	Φ (System Clock)
28	M1–
30	IORQ–
45	RD–
47	D4
48	D5
51	D6
52	D7
54	D0
55	D1
56	D3
57	D2
62–64, 120–122	GND
88	IEI
109	INT–
112	IEO

Zilog

PPB/16 Specifications

PROM TYPES:

24-pin EPROM

PROM TYPES:

24-pin EPROM—2716

24-pin Bipolar—	82S140	(512 x 8)
	82S141	(512 x 8)
	82S180	(1024 x 8)
	82S181	(1024 x 8)
	82S2708	(1024 x 8)
16-pin Bipolar—	82S126	(256 x 4)
	82S129	(256 x 4)
	82S130	(512 x 4)
	82S131	(512 x 4)

CONTROL INTERFACE:

TTL interface with MCZ series data, address, and control signals.

ELECTRICAL SPECIFICATIONS:

+5 VDC ± 5%
Maximum Current—2.5 AMPS during programming
1.5 AMPS during read

CONNECTOR:

122-pin edge (100 mil spacing) available from:

VENDOR	PART NO.
Garry Mfg. Co.	4000-2
Augat	14005-19P1

PHYSICAL CHARACTERISTICS:

Width: 7.7 inches (19.6 cm)
Height: 9.0 inches (22.9 cm)
Thickness: 0.062 inches (0.16 cm)
Spacing between cards: 1.0 inches (2.54 cm) centers
Maximum component height: 0.9 inches (2.29 cm)
Etch layers: two

ENVIRONMENT:

0° to 50° C ambient operating

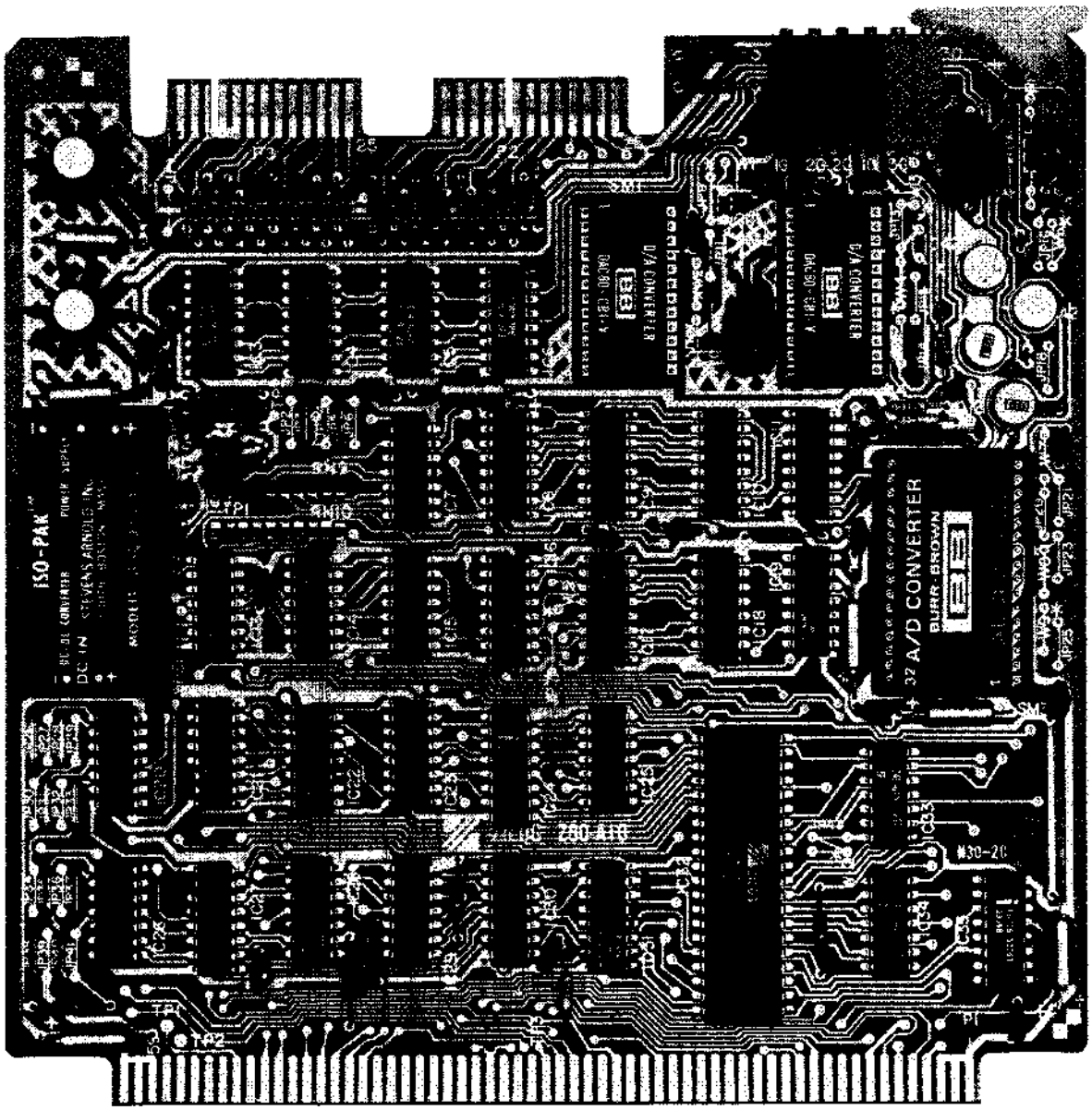
Zilog

Product Specification

Z-80® AIO/AIB Analog Input/Output Board

The Z-80 AIO is a completely compatible 12-bit analog input/output system for the Z-80 Microcomputer Board Series or any system composed of these boards.

The analog to digital portion has 16 differential or 32 single-ended channels with input voltages ranging from 2.5mV full scale to 10V full scale. The digital to analog portion provides two 12-bit D/A converters with double buffering to minimize output switching transients. The output voltage is selectible for bipolar or unipolar operation ranging from 2.5mV full scale to 10V full scale. This analog system is interfaced as I/O to the CPU and may operate in a polling or interrupt mode. The system's I/O addressing may be changed by on-board jumper selections.



Features

Z-80 AIO

- 16 differential or 32 single-ended analog channels
- Input voltage range: 2.5mV full scale to 10V full scale
- High gain instrumentation amplifier
- Sample and hold amplifier
- 12-bit converter
- Maximum throughput: 45μ per channel
- Two 12-bit converters
- Double buffered output
- Single +5V power supply

Z-80 AIB

- 16 differential or 32 single-ended analog channels
- Input voltage range: 2.5mV full scale to 10V full scale
- High gain instrumentation amplifier
- Sample and hold amplifier
- 12-bit converter
- Single +5V power supply

Operation

Interfacing the AIO or AIB to the system bus is accommodated by the on-board PIO which is simply addressed as I/O. The ADDRESS DECODER (block diagram) uses ten addresses to direct all board operations. Five addresses are used in conjunction with the system control lines and data bus to program the PIO operating mode, select the analog input channel and read the 12-bit A/D conversion result. One address is used to read the status register. The remaining four addresses are characteristic to the AIO board and are used in conjunction with the data bus to operate the two 12-bit D/A converters.

By selecting the PIO port A or B control addresses, the PIO may be programmed to interrupt the CPU system and supply an interrupt vector address upon completion of an A/D conversion. The selection of an analog input channel automatically requests an A/D conversion. The CONTROL and TIMING will gate the requested analog input channel to the ANALOG MULTIPLEXER, strobe the SAMPLE and HOLD AMPLIFIER and request conversion of the A/D CONVERTER. Upon completing the conversion, the A/D CONVERTER will respond to the PIO through the CONTROL and TIMING that the converted data is ready. At this time, the PIO may interrupt the system or the system may read the status register to find that the conversion data is ready and the results have not previously been read.

The two 12-bit D/A converters on the AIO have separate I/O addresses for the upper and lower bytes of the data word.

A word is formed by loading the eight least significant bits into the D/A converter where they are latched and buffered from the D/A inputs until the final four bits of the data word are received. The combined 12 bits of data are then gated and latched simultaneously to the D/A inputs. This double buffering scheme prevents conversion of partial words, and therefore, eliminates spiking in the output signal.

Specifications

Analog Input Section

INPUT CHARACTERISTICS

Number of Channels	32 single-ended/16 differential
ADC Gain Ranges (Jumper Selectable)	0–5V, 0–10V, ±2.5V, ±5V, ±10V
Amplifier Gain Ranges (Resistor Programmable)	1 to 1000
Maximum Input Voltage without Damage	±26 volts
Input Impedance	100 mΩ, 10 pF OFF Channel 100 mΩ, 100 pF ON Channel
Bias Current	20 nA
Differential Bias Current	10 nA

TRANSFER CHARACTERISTICS

Resolution	12 bits
Throughput Time (max.) G = 1	45 μsec/channel

ACCURACY

System Accuracy at +25°C (max.) (1)	±0.025%FSR (2)
Linearity	±1/2 LSB
Differential Linearity	±1/2 LSB
Quantizing Error	±1/2 LSB
Monotonicity (3)	Guaranteed 0°C to +70°C

Specifications (Cont'd)

Analog Output Section

OUTPUT CHARACTERISTICS

Number of Channels	2
Output Voltage Ranges (Strap Selectable)	±10V, 0 to 10V, ±5V, 0 to 5V, ±2.5V at 5 mA
Output Impedance	1Ω

TRANSFER CHARACTERISTICS

Resolution	12 bits
Output Settling Time (max.)	10 μsec

ACCURACY

Output Accuracy	±0.0125% FSR
Temperature Coefficient of Accuracy	±30 ppm of FSR/°C

NOTES:

1. Includes offset errors, gain errors, linearity errors at gain = 1.
2. FSR mean Full Scale Range.
3. No missing codes guaranteed.
4. Includes offset drift, gain drift and linearity drift.

Typical at 25°C and rated power supplies unless otherwise noted.

STABILITY OVER TEMPERATURE (4)

System Accuracy Drift (max.) G = 1	±30 ppm of FSR/°C

DYNAMIC ACCURACY

Sample and Hold Aperture Time	30 ns
Aperture Time Uncertainty	±5 ns
Differential Amplifier CMR	74 dB (DC to 1 kHz)
Channel Crosstalk	80 dB down at 1 kHz, for OFF channel to ON channel

POWER REQUIREMENTS

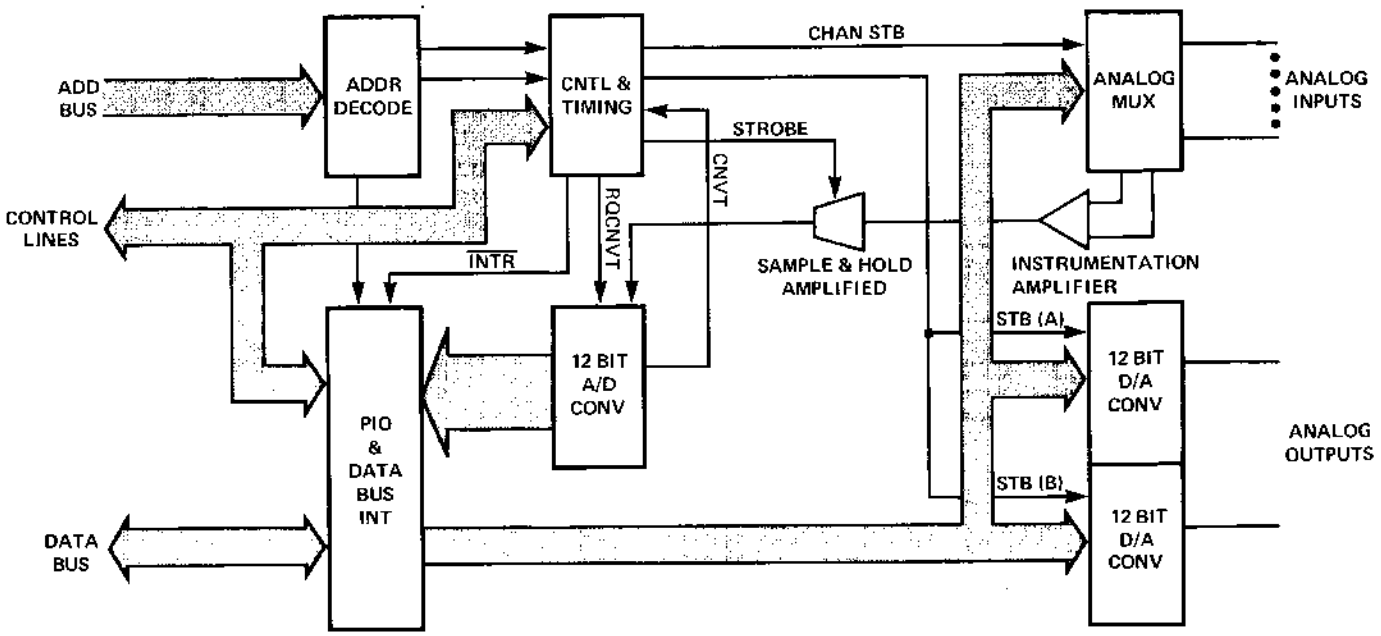
+5V ±5% at 1.5 amp

CONNECTOR

Ansley: cable. . . . . P/N 171-26
Ansley: socket. . . . . P/N 609-258

ENVIRONMENTAL

Operating Temperature	0°C to +70°C
Storage Temperature	-25°C to +85°C
Relative Humidity	95% noncondensing



AIO BLOCK DIAGRAM

System Bus Connections

PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION
1	+5V	13	DBØ	63	GND	100	AB3
2	+5V	23	$\overline{WR}$	64	GND	101	AB2
3	+5V	26	AB7	68	DB4	102	AB1
4	$\overline{IORQ}$	29	AB5	71	DB2	103	ABØ
5	DB5	30	AB6	73	DB7	115	$\overline{M1}$
6	IEO	59	+5V	75	DB1	116	$\overline{RD}$
7	IEI	60	+5V	79	$\overline{INT}$	120	GND
8	DB3	61	+5V	98	AB4	121	GND
12	DB6	62	GND	99	$\overline{Ø}$	122	GND

Analog Interconnections

	SIGNAL	BOARD P2
ANALOG INPUTS	CH 23, or CH 7 RTN	1, 2
	CH 22, or CH 6 RTN	3
	CH 21, or CH 5 RTN	4
	CH 20, or CH 4 RTN	5
	CH 19, or CH 3 RTN	6
	CH 18, or CH 2 RTN	7
	CH 17, or CH 1 RTN	8
	CH 16, or CH 0 RTN	9
	CH 7	10
	CH 6	11
	CH 5	12
	CH 4	13
	CH 3	14
	CH 2	15
	CH 1	16
	CH 0	17
ANALOG OUTPUTS	DAC–GND	18
	DAC–FB	19
	Analog Common	20
	DAC–GND	21
	DAC–OUT	22
	DAC–OUT	23
	DAC–FB	24
VOLTAGE OUTPUTS	-15V	25
	+15V	26

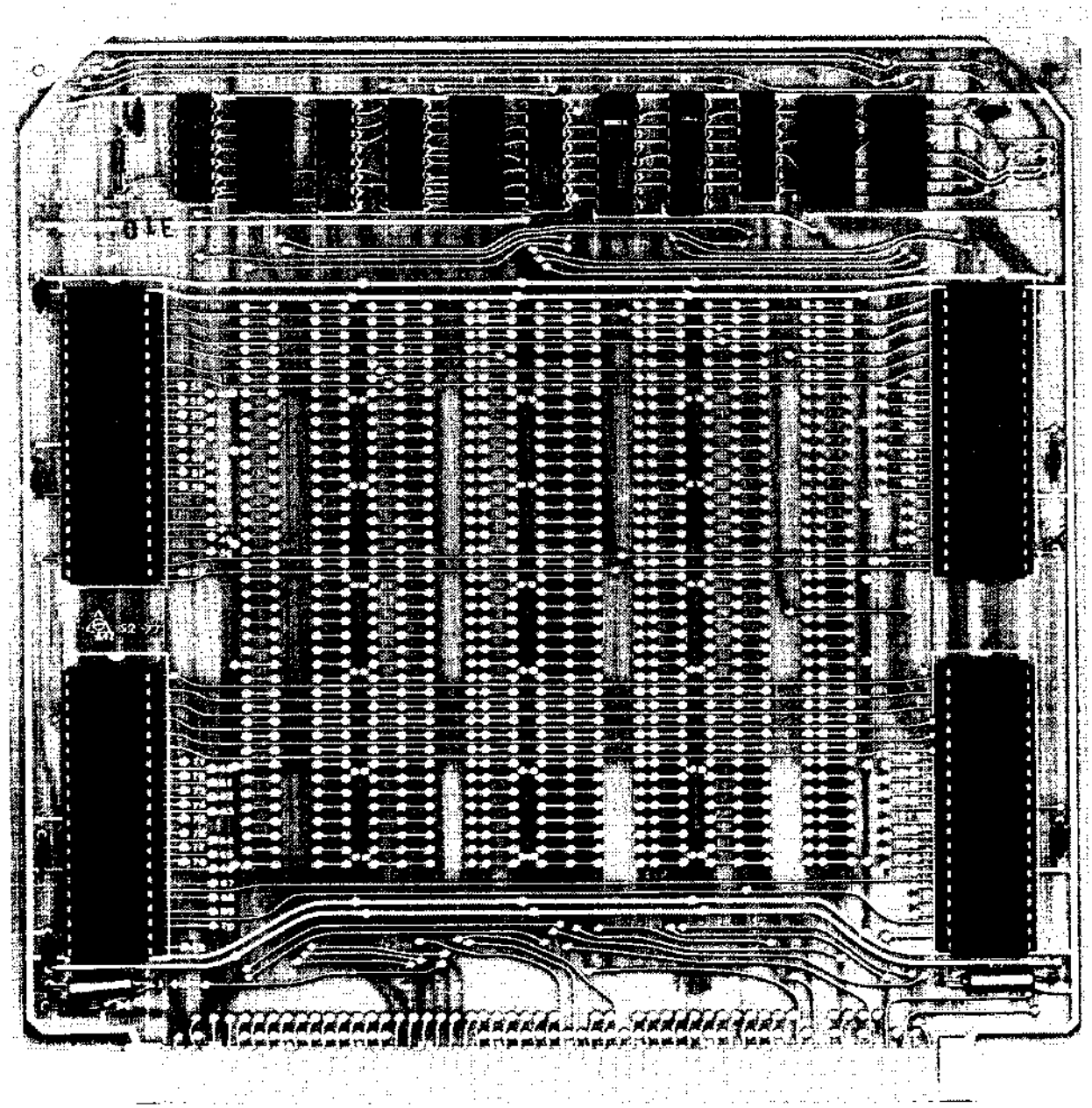
	SIGNAL	BOARD P3
	Remote Common	1, 2
	-15V	3
	+15V	4
	Analog Common	5
	Analog Common	6
	Analog Common	7
	Analog Common	8
	Analog Common	9
	Analog Common	10
ANALOG INPUTS	CH 30, or CH 14 RTN	11
	CH 31, or CH 15 RTN	12
	CH 28, or CH 12 RTN	13
	CH 29, or CH 13 RTN	14
	CH 26, or CH 10 RTN	15
	CH 27, or CH 11 RTN	16
	CH 24, or CH 8 RTN	17
	CH 25, or CH 9 RTN	18
	CH 14	19
	CH 15	20
	CH 12	21
	CH 13	22
	CH 10	23
	CH 11	24
	CH 8	25
	CH 9	26

Zilog

Product Specification

Z-80® IOB Parallel I/O Board

The Z-80 IOB Parallel I/O Board is part of the Zilog Microcomputer Board Series which provides a modular approach to a complete computing and processing system. The IOB is bus compatible and directly interfaces to all other cards in the MCB Series. It provides eight additional programmable 8-bit parallel I/O ports to augment the two contained on the Z-80 MCB Microcomputer Board. The IOB features the use of the Z-80 PIO device that has become the standard of the microcomputer industry.



Features

- 64 programmable I/O lines
- Handshake lines for byte mode control and fast interrupt response
- 4 Modes of operation
- 32 lines with Darlington drive capability
- Selectable daisy chain priority interrupt structure
- Automatic interrupt vectoring with 128 possible vectors
- Single +5 volt power supply
- Open space provided for user supplied interface hardware

Specifications

POWER SUPPLY:	+5 VDC ±5% Supports up to 10 Watts total power dissipation.
CONNECTOR:	122-pin edge (100 mil spacing)
SIZE:	Length' 7.7" Depth, 7.5" Spacing, 0.5" centers.
ENVIRONMENTAL:	0° – 50°C temperature range. Up to 90% humidity without condensation.
I/O CHANNELS:	Eight 8-bit parallel I/O channels can be programmed for byte or bit transfer in either direction.

Pin Assignments

IOB–PIN OUT(COMPONENT SIDE)

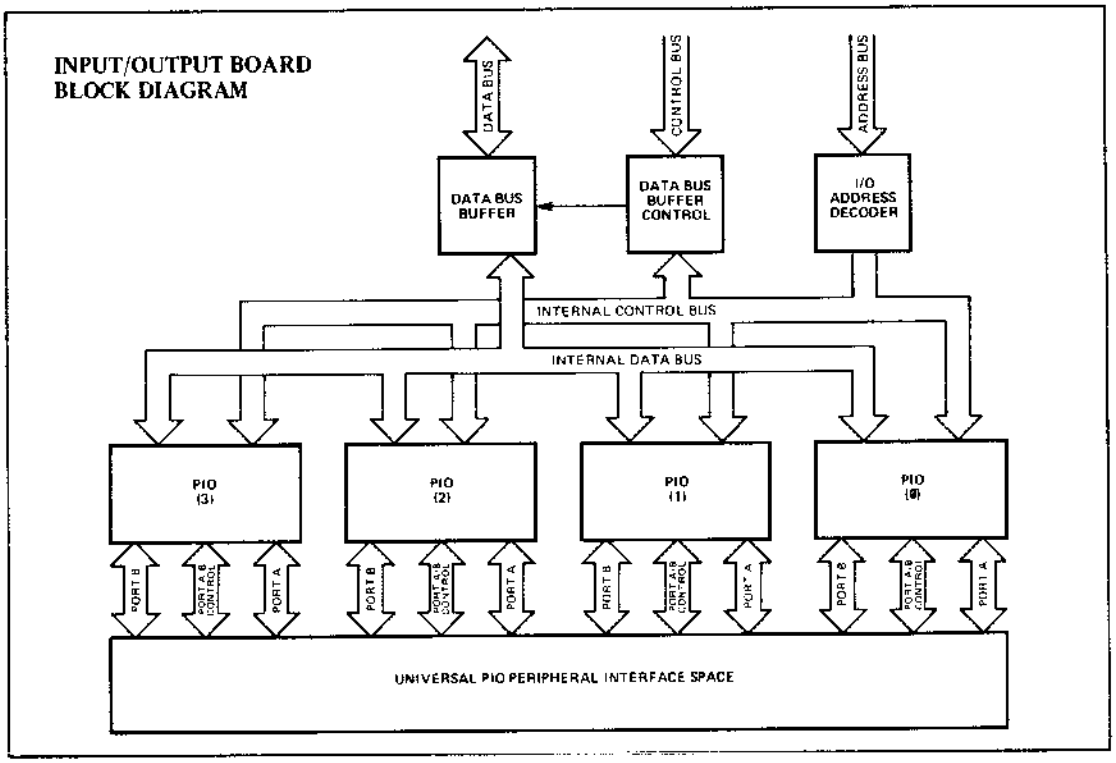
1	+5 volts	13	DATA BIT Ø
2	+5 volts	26	ADDRESS BIT 7
3	+5 volts	29	ADDRESS BIT 5
4	$\overline{IORQ}$	30	ADDRESS BIT 6
5	DATA BIT 5	55	INT ENABLE IN, PIO2
6	INT ENABLE OUT, PIOØ	58	INT ENABLE OUT, PIO3
7	INT ENABLE IN, PIOØ	59	+5 volts
8	DATA BIT 3	60	+5 volts
12	DATA BIT 6	61	+5 volts

NOTE: PINS 9–11, 14–25, 27, 28, 31–54: UNIVERSAL PIO INTERFACE

IOB–PIN OUT (SOLDER SIDE)

62	GND	100	ADDRESS BIT 3
63	GND	101	ADDRESS BIT 2
64	GND	102	ADDRESS BIT 1
65	INT ENABLE OUT PIO1	103	ADDRESS BIT Ø
66	INT ENABLE IN, PIO1	115	$\overline{M1}$
68	DATA BIT 4	116	$\overline{RD}$
71	DATA BIT 2	118	INT ENABLE IN, PIO3
73	DATA BIT 7	119	INT ENABLE OUT, PIO2
75	DATA BIT 1	120	GND
79	$\overline{INTERRUPT}$	121	GND
98	ADDRESS BIT 4	122	GND
99	$\overline{\Phi}$		

NOTE: PINS 56, 57, 67, 69, 70, 72, 74, 76–78, 80–97, 104–114, 117: UNIVERSAL PIO INTERFACE



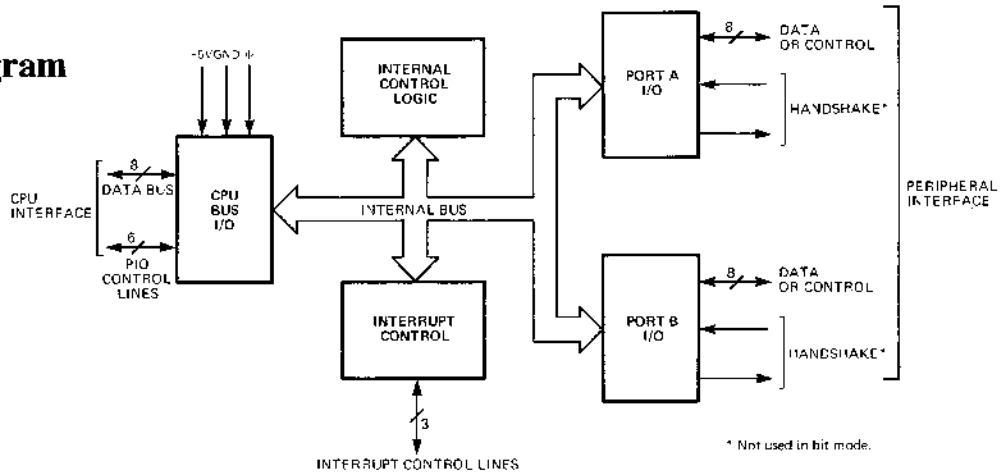
IOB Applications

Z-80 PIO ARCHITECTURE

A block diagram of the Z-80 PIO is shown in Figure 1. The internal structure of the Z-80 PIO consists of a Z-80 CPU bus interface, internal control logic, Port A I/O logic, Port B I/O logic, and interrupt control logic. A typical application might use Port A as the data transfer channel and Port B for the status and control monitoring.

The Port I/O logic is composed of 6 registers with "handshake" control logic as shown in Figure 2. The registers include: an 8-bit input register, an 8-bit output register, a 2-bit mode control register, an 8-bit mask register, an 8-bit input/output select register, and a 2-bit mask control register. The last three registers are used only when the port has been programmed to operate in the bit mode.

Figure 1
PIO Block Diagram



REGISTER DESCRIPTION

Mode Control Register—2 bits, loaded by CPU to select the operating mode: byte output, byte input, byte bidirectional bus or bit mode.

Data Output Register—8 bits, permits data to be transferred from the CPU to the peripheral.

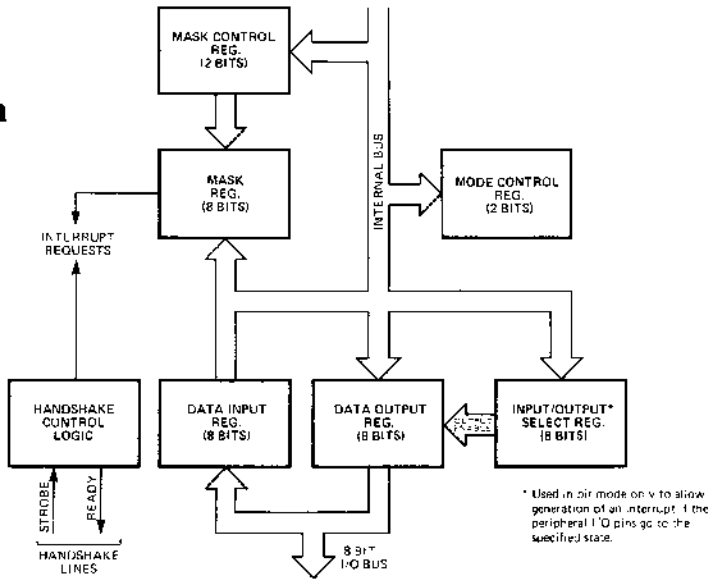
Data Input Register—8 bits, accepts data from the peripheral for transfer to the CPU.

Mask Control Register—2 bits, loaded by the CPU to specify the active state (high or low) of any peripheral device interface pins that are to be monitored and, if an interrupt should be generated when all unmasked pins are active (AND condition), or when any unmasked pin is active (OR condition).

Mask Register—8 bits, loaded by the CPU to determine which peripheral device interface pins are to be monitored for the specified status condition.

Input/Output Select Register—8 bits, loaded by the CPU to allow any pin to be an output or an input during bit mode operation.

Figure 2
A Typical Port I/O Block Diagram

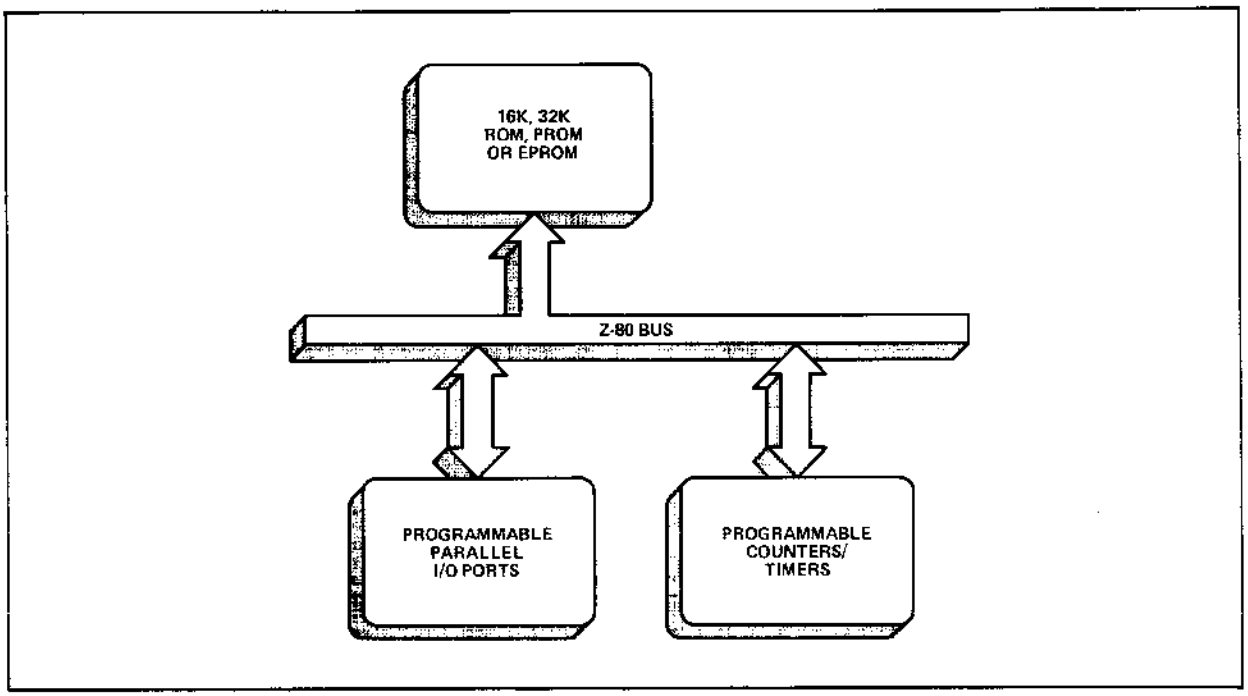


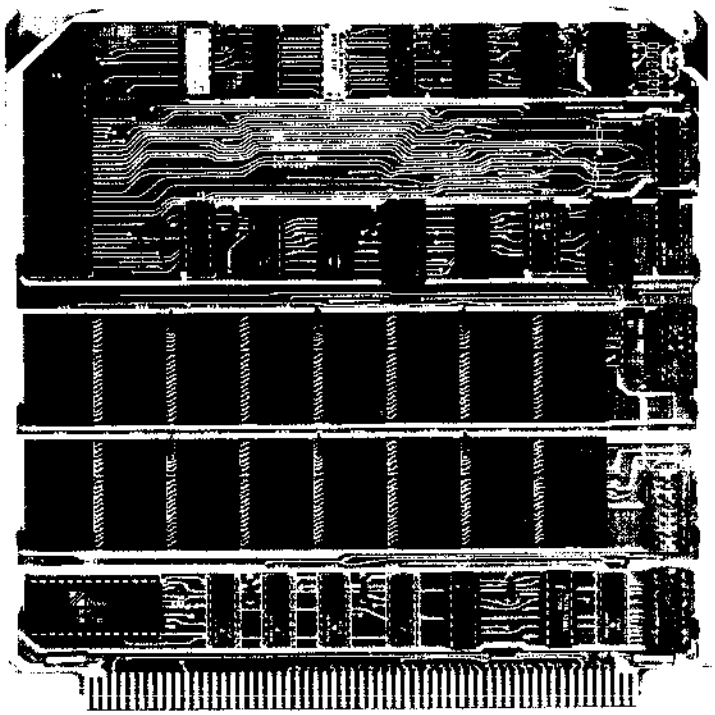
Zilog

Product Specification

Z-80® PMB PROM Memory Board

The Z-80 PMB provides the Zilog Microcomputer Board Series with additional memory and I/O capacity. It consists of sockets for up to 32K bytes of PROM memory using ROM, PROM or EPROM, a Z-80 PIO dual port parallel I/O device, and a Z-80 CTC providing four 8-bit counter timers.





Features

- 16K bytes per board utilizing 2708 EPROMs or 6381 type PROMs.
- 32K bytes per board utilizing 2716 type EPROMs or 2316 type ROMs.
- Program memory addressing is implemented by using a PROM-based memory address decoder.
- 5 volt operation when using 5 volt ROMs or 5 volt 2716 EPROMs. *NOTE: For certain EPROMs an external −5 and +12 volt power supply is required.*
- On board PIO provides sixteen parallel I/O lines. On-board CTC provides four 8-bit counters/timers.
- Accepts all industry standard ROM, PROMs and EPROMs, both bipolar and NMOS–2708, 2716 (5 volt only), 82S181, 2708, 7640, 7641, 2516, 2316, 2758.

Specifications

POWER SUPPLY: +5V ±5%

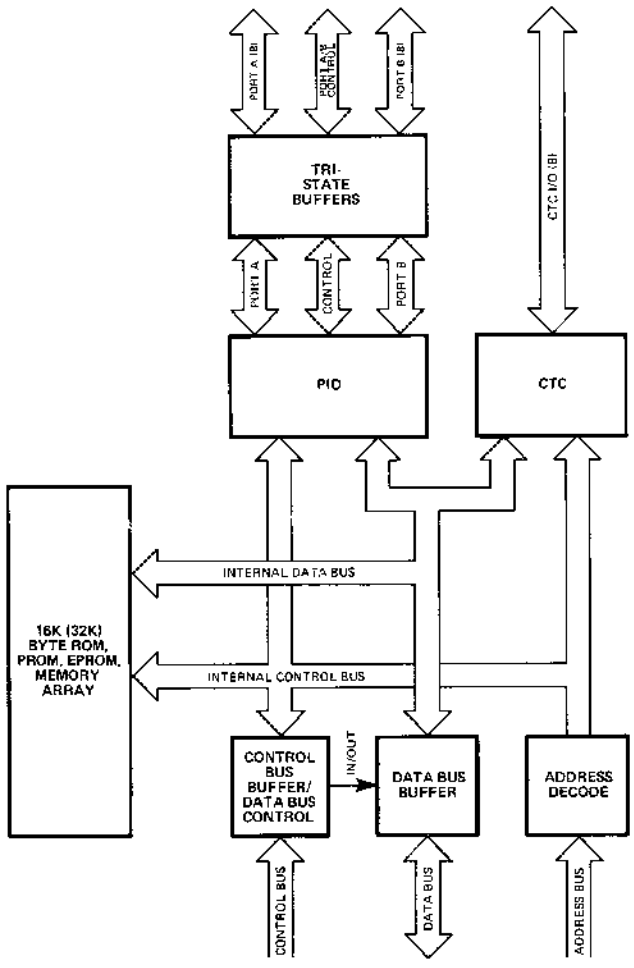
	Without Memory	With 2708 Max	With 2716 Max	With 6381 Max
5V	0.60A	.28A	2.28A	3.40A
−5V	–	.96A	–	–
+12V	–	1.28A	–	–

CONNECTOR: 122-pin edge (100 mil. spacing)

SIZE: Length: 7.7"
Depth: 7.5"
Spacing: .5" centers

ENVIRONMENTAL: 0°–50°C temperature range

MEMORY CAPACITY: NON-VOLATILE Up to 16K or 32K bytes of memory depending on which type of ROM, EPROM or PROM used. Each 2K page may be assigned any one of 32 possible starting addresses.



Pinout for PROM Memory Board

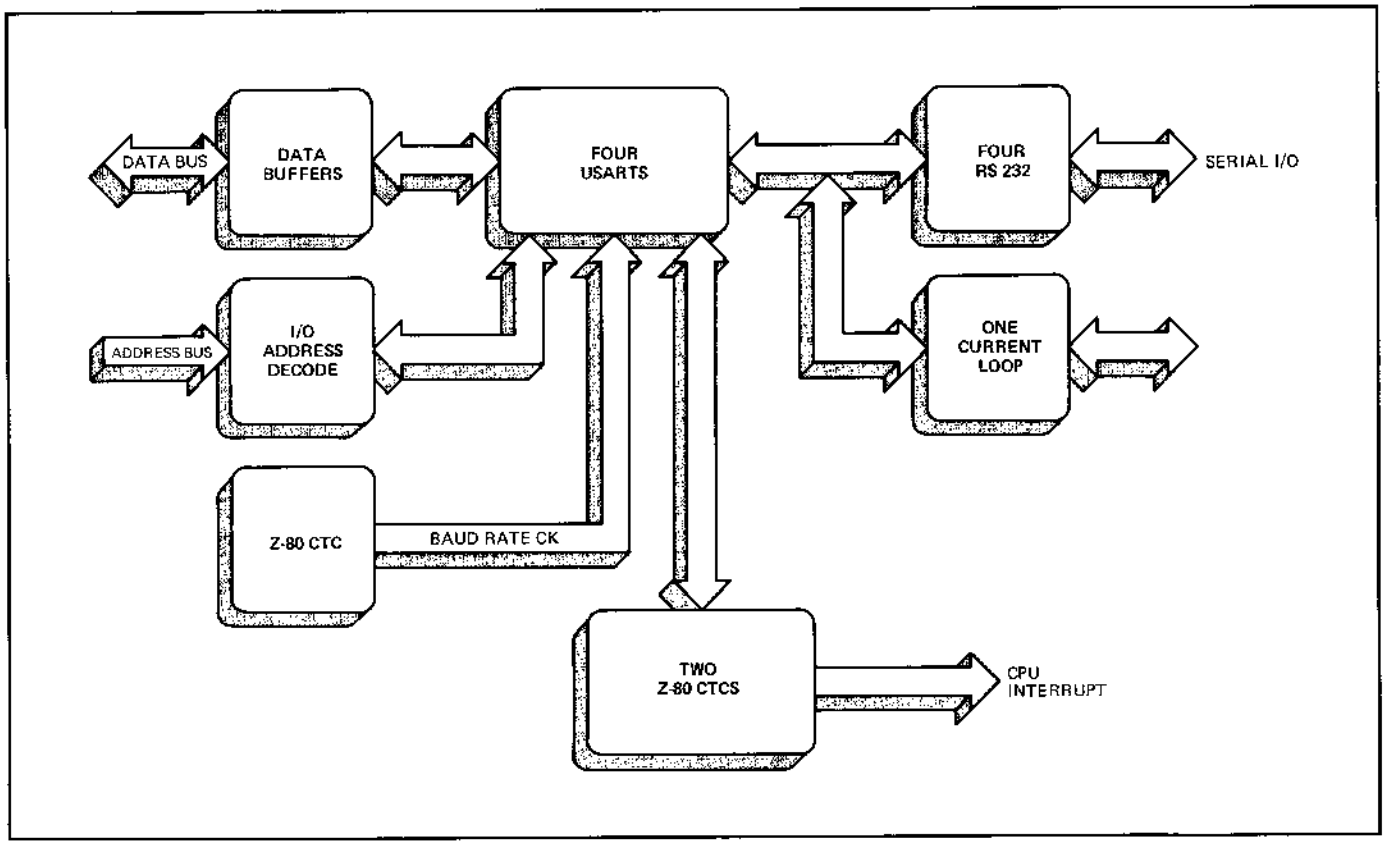
PIN NUMBER	SIGNAL NAME	PIN NUMBER	SIGNAL NAME	PIN NUMBER	SIGNAL NAME
001	+5V	047	.	093	.
002	+5V	048	.	094	AB14
003	+5V	049	.	095	.
004	IORQ–	050	IEO . PMB . PIO	096	.
005	DB5	051	.	097	AB12
006	.	052	.	098	AB4
007	IEI . PMB . PIO	053	.	099	Φ–(System Clock–)
008	DB3	054	.	100	AB3
009	PIO . 0 . A0	055	.	101	AB2
010	PIO . 0 . A1	056	.	102	AB1
011	PIO . 0 . A2	057	.	103	AB0
012	DB6	058	IEO . PMB . CTC	104	.
013	DB0	059	+5V	105	.
014	PIO . 0 . A3	060	+5V	106	.
015	PIO . 0 . A4	061	+5V	107	.
016	PIO . 0 . A5	062	GND	108	.
017	PIO . 0 . A6	063	GND	109	.
018	PIO . 0 . A7	064	GND	110	CK/T0
019	PIO . 0 . A . RDY	065	.	111	CK/T1
020	PIO . 0 . A . STRB–	066	–5V	112	CK/T2
021	PIO . 0 . B . RDY	067	–5V	113	CK/T3
022	PIO . 0 . B . STRB–	068	DB4	114	ZC/TO0
023	.	069	+12V	115	M1–
024	PIO . 0 . B0	070	+12V	116	RD–
025	PIO . 0 . B1	071	DB2	117	ROM . DISABLE–. (IN)
026	AB7	072	.	118	ZC/TO2
027	AB8	073	DB7	119	ZC/TO1
028	PIO . 0 . B2	074	.	120	GND
029	AB5	075	DB1	121	GND
030	AB6	076	.	122	GND
031	PIO . 0 . B3	077	.		
032	AB15	078	.		
033	PIO . 0 . B4	079	INT–		
034	PIO . 0 . B5	080	.		
035	.	081	.		
036	AB13	082	.		
037	AB11	083	.		
038	PIO . 0 . B6	084	.		
039	PIO . 0 . B7	085	MRQ–		
040	.	086	.		
041	.	087	.		
042	.	088	.		
043	.	089	AB9		
044	.	090	.		
045	.	091	AB10		
046	.	092	.		

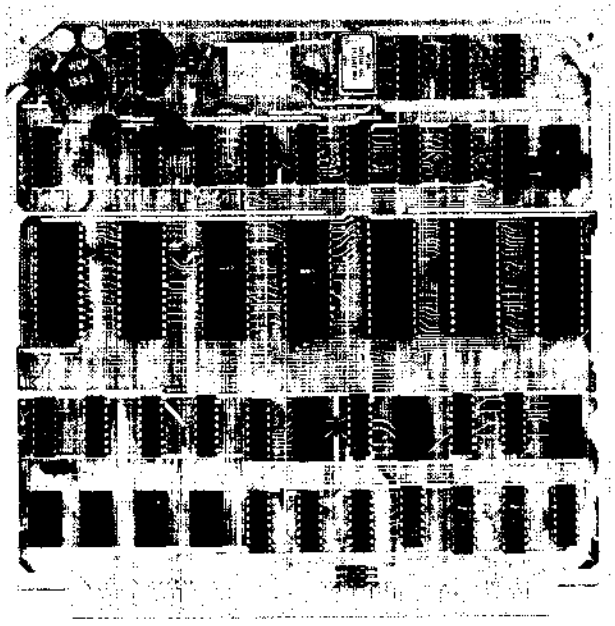
Zilog

Product Specification

Z-80® SIB Serial Interface Board

The Z-80 SIB contains four software controlled serial channels for handling most synchronous and asynchronous communication protocols.





Description

The Z-80 Serial Interface Board (SIB) provides a programmable serial input/output interface for the MCB OEM board series. The SIB contains four programmable, bi-directional serial communication channels (USARTS), which support both synchronous and asynchronous data formats. Each USART is programmable by means of a control byte loaded by the Z-80 microprocessor. The USART accepts data characters from the Z-80 in a parallel format and converts them into a continuous serial data stream for transmission. The USART can simultaneously receive serial data streams and convert them into parallel data characters for the Z-80 CPU.

All four channels are RS-232-buffered, and one channel will accomodate a 20 ma-TTY current-loop interface. An on-board DC/DC converter generates the RS-232 voltage levels, allowing the SIB to be powered by a single +5 volt supply.

Three Z-80 CTC (Counter Timer Circuit) devices are on the SIB. One CTC is dedicated to establishing one of 14 baud rates for each of the four USARTS. Two CTCs are used to accomodate Z-80 interrupt capability for receive and transmit operation for each USART.

Environment

0° to 50°C ambient operating temperature

Interrupt Configuration

Interrupt interface to Z-80 CPU via two Z-80 CTCs. Request generated for transmitter or receiver ready (TXRDY–RXRDY).

Edge Connector Pin Assignments

PIN	DESCRIPTION
1, 3, 59–61	+5V P.S.
4	IORQ–
5	DB5
8	DB3
9	MASTER RESET
12	DB6
13	DB0
23	WR–
26	AB7
29	AB5
30	AB6
62–64, 120–122	GND
68	DB4
71	DB2
73	DB7
75	DB1
79	INT–
98	AB4
99	Φ–. (SYSTEM CLOCK–)
100	AB3
101	AB2
102	AB1
103	AB0
115	M1–
116	RD–

Features

- Bus compatible with MCB series of boards
- Four programmable, bi-directional serial communication channels
- Half or full duplex data transfer
- Synchronous or asynchronous operation
- Four RS-232 interfaces, one 20 ma current-loop interface
- Three Z-80 CTCs, one for baud rate generation, two for interrupt control
- Baud rates from 50 to 9600 (asynchronous) and DC to 56K baud (synchronous)
- Three clocking options: on-board clock, system clock, external clock
- Programmable I/O address selection

SIB Specifications

SERIAL I/O:

Four serial input and four serial output channels configured with 8251 USARTS.

SERIAL MODES:

Synchronous or asynchronous using virtually any serial protocol.

SERIAL INTERFACE:

CHANNEL	INTERFACE
0	RS232
1	RS232
2	RS232
3	RS232 or 20ma current loop

CONTROL INTERFACE:

TTL interface with MCZ series data, address, and control signals.

I/O PORT ADDRESSING:

20 jumper selectable I/O ports.

BAUD RATE:

Programmable from 50 to 9600.

ELECTRICAL SPECIFICATIONS:

+5 VDC ± 5%
Maximum current = 1.5 amps

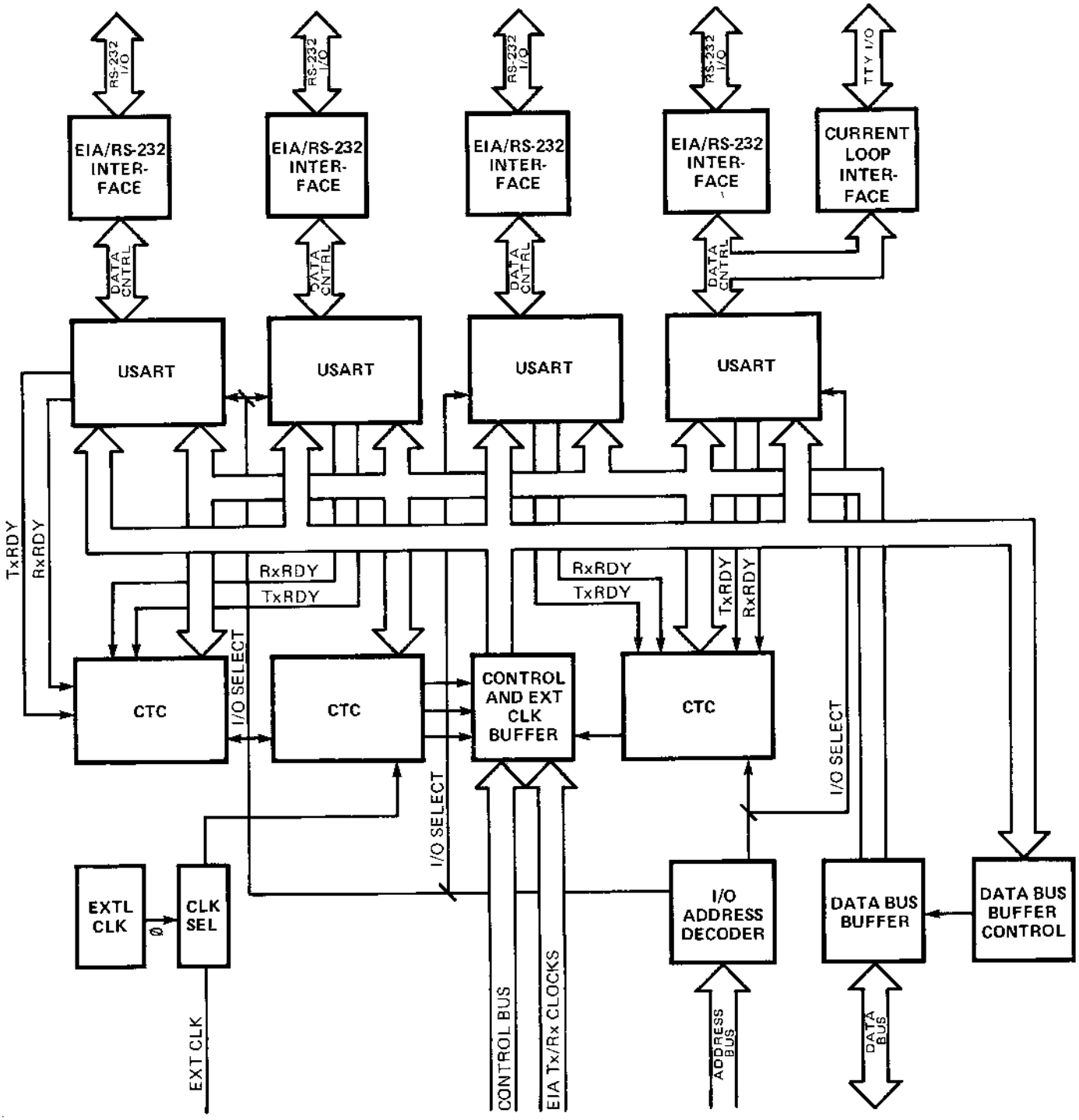
CONNECTOR:

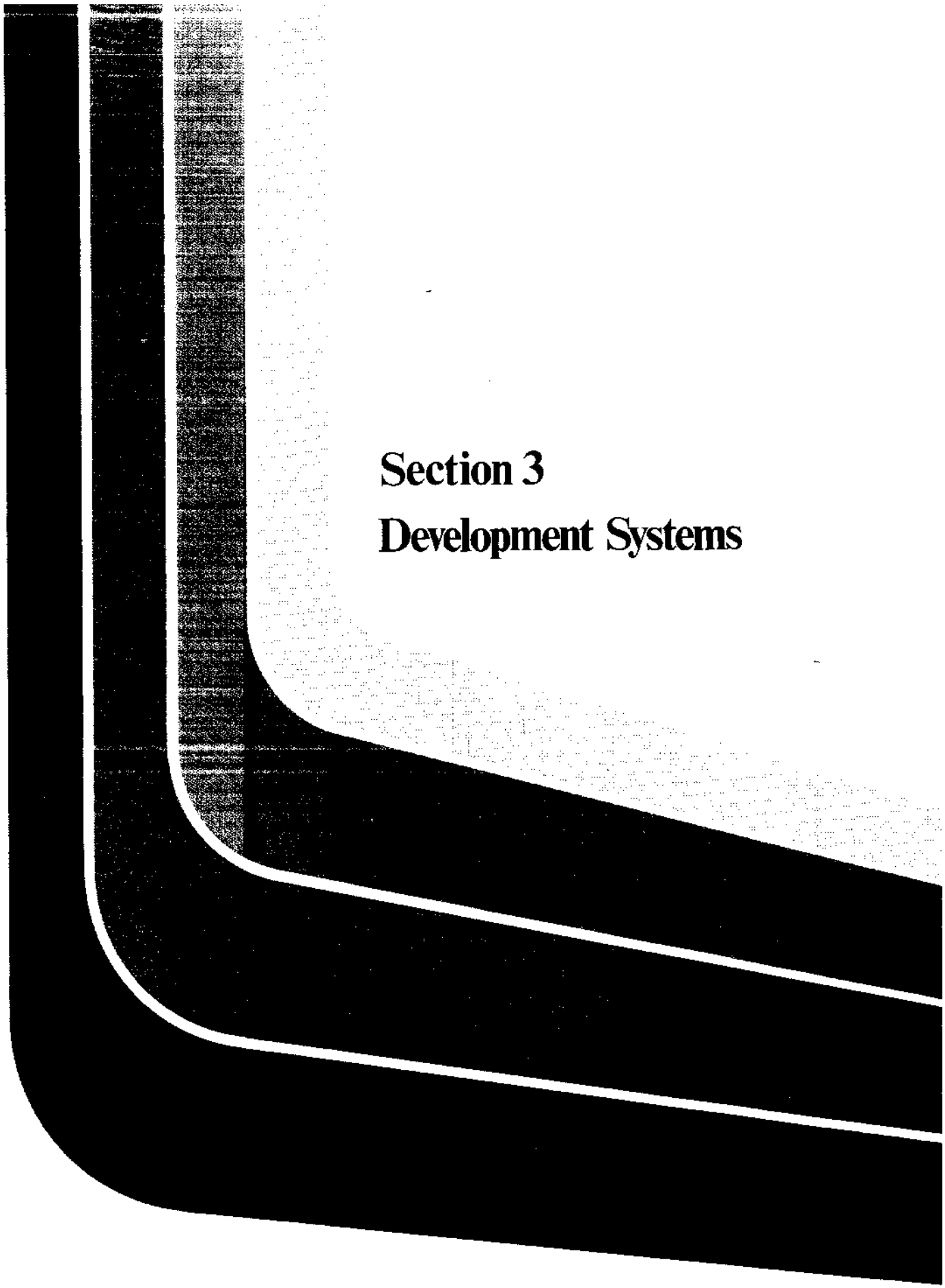
122-pin edge (100 mil spacing). AUGAT PN 14005-19P1

PHYSICAL CHARACTERISTICS:

Length: 7.7" (19.7 cm)
Depth: 7.5" (19.1 cm)
Thickness: .062" (0.16 cm)
Etch Layers: Two

Block Diagram





Section 3
Development Systems

Zilog

Section 3
Development Systems

For Hardware Development

The ZDS-1 Series provide the tools necessary for the designer to easily debug and test Z80 and Z80A Microprocessor-based hardware. Functions normally supplied by many pieces of special purpose design equipment are integrated in by the ZDS-1 Series Development Systems.

As an Emulator the Development System connects directly to the prototype system and behaves like the Z80 or Z80A Microprocessor, only under direct control of the user, to enable:

- Setting or resetting of all registers and RAM memory locations.
- Setting or resetting of interrupt controls.
- Single or multiple instruction stepping.
- Setting or displaying of I/O Port information.
- Start/Stop control of program execution.

As a Logic Analyzer the Development System monitors and stores pertinent CPU bus activity and terminates program execution in an orderly manner when user defined events occur. Some examples are:

- Storage of address, data and control bus information during each bus transaction.
- Storage of only user defined bus transactions; e.g. Memory read, I/O write.
- Suspension of program execution on occurance of a user defined event.
- Display a history of the last n bus events, where n equals from one through 256 events.

As a ROM/RAM Simulator the Development System lends its memory to the user's prototype system with the memory mapping features.

- Memory Mapping allows the user to describe the physical nature of the memory to be used by the prototype system. A memory map is defined by the user to specify which addresses exist in the prototype and which address locations are to be "borrowed" from the Development System.
- The ZDS-1/40 Development System also permits memory address to be declared write protected or nonexistent so that if any access is attempted a break in program execution will occur.

As a PROM/EPROM Programmer the Development System provides the functions necessary to write, read, verify and duplicate PROMS. This is an optional feature provided by the ZDS/PPB and ZDS/PPB/16 PROM Programmer Option, or ZDS/CIB Printer and Prolog PROM Programmer Interface.

For Software Development

ZDS-1 Series Development Systems provide a standalone microcomputer system utilizing the versatile RIO Operating System for the creation, editing, assembly and debug of the software for the powerful Z80 Microprocessor.

The RIO Operating System with relocatable modules and I/O management is a general purpose computing system with an architecture that is designed to facilitate the development process, provide straight-forward linkage to various systems routines, and enable expansion of system features to meet the particular needs of individual users.

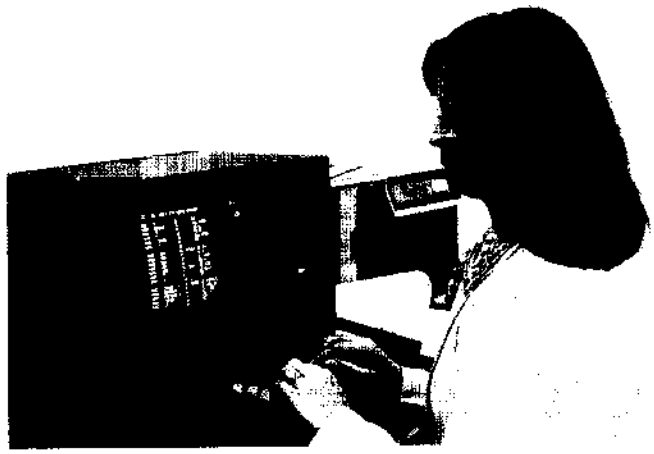
Features

- **OS EXECUTIVE**
  - Map requests for operations on logical units to specific device handling programs.
  - Commands may be issued to OS from the console or by an executing program.
  - Any number of user defined commands may be added to the system.
  - Command sequences may be recorded in files and executed as a group.
- **ADVANCED ASSEMBLER**
  - Relocatable or absolute object code format.
  - External Symbol references.
  - Global Symbol definitions.
  - Macros and conditional Assembly.
  - Paged symbol table permits assembly of arbitrarily large programs in standard memory.
  - Include directive permits additional files to be merged into the source program at assembly time.
- **LINKER**
  - Assigns absolute addresses to program modules.
  - Resolves External references.
  - Permits overlays or memory gaps.
  - Produces memory map and Global address table.
- **TEXT EDITOR**
  - Paged work space permits any size of file to be edited.
  - Automatic backup file creation for protection.
  - Access to other files during edit session.
  - All edit operations for locating and modifying lines within a file are based on character string matching.
- **PROM MONITOR**
  - Low level device handlers for system console and floppy disc.
  - Bootstrap loader for ease in system initialization.
  - Machine language Debug package.
- **ZDOSII DISC FILE HANDLING PACKAGE**
  - Allocates all disc space automatically on an "as needed basis.
  - Supports sequential access to file or direct access to any specific disc address.
- **TABBING**
  - Supported throughout the system.
  - Compact storage of tabbed text on disc.

Utilization of a standalone microcomputer permits all program preparation and verification to take place in the same environment. Smooth and rapid transition from program input, to assembly and linkage to execution and debug provide a rapid and inexpensive solution for software development.

Optional software languages include a BASIC interpreter, a Fortran compiler and Zilog's own powerful PLZ family of system programming language processors.

All software is supplied on floppy diskette media. Each package comes complete with object software, technical and user documentation, and instructions for use with the ZDS-1 System.



Zilog

Product Specification

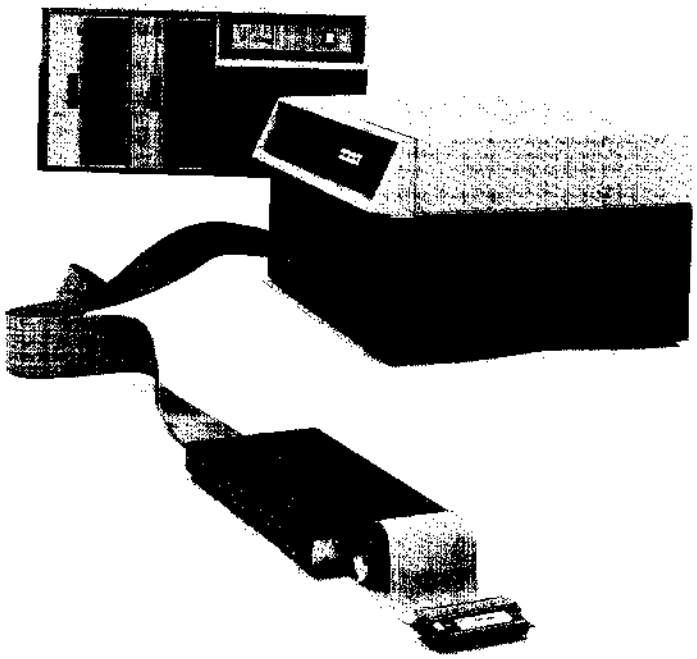
ZDS-1Series

ZDS-1/40 4MHz Development System

The ZDS-1/40 provides a powerful standalone development system for use with the Z80 or Z80A Microprocessor and associated 4MHz peripheral components. Precise emulation is provided by using two microprocessors, a Z80A-CPU which is inserted into the prototype system and a second Z80-CPU which is internal to the development system. This method allows precise emulation up to clock frequencies of 4MHz.

While similar in functional capability to the ZDS-1/25 Development System the ZDS-1/40 provide special features, shown below.

- External Z80A-CPU for direct connection to the user's prototype system.
- Precise emulation for systems employing clock rates to 4MHz.
- Memory mapping and protection in blocks of 1024 bytes.
- User memory refresh, even when an emulation is suspended.
- Verification of user clock integrity.
- Memory address translation.
- Detection of memory accesses to write protected or non-existent blocks.
- New powerful disc-based debug software.



Features

- **Z80A-CPU**—The 4MHz version of the powerful Z80 micro-processor.
- **Main Memory**—Capacity of up to 60K bytes on a single board, allowing more room for I/O options. Standard system includes 32K bytes.
- **Floppy Disc Drives**—Each providing for the storage of up to 300,000 bytes on an inexpensive, removable diskette.
- **RIO Operating System**—With Relocating Assembler, Linker, Text Editor and Logical File Structure.
- **Real-Time Storage Module**—Enables the monitoring of specific address, data and control bus lines during selected operations, e.g. Memory Read, Memory Write, I/O Read, I/O Write.
- **Breakpoint Module**—Enables the monitoring and testing of specific address, data and control bus states to stop program execution or to create a scope sync.
- **Memory Mapper**—Enables the user to describe the physical nature of the memory to be used. The user may utilize the memory in the Development System, his own system or a combination of both in blocks of 1024 bytes. In addition each block may be assigned properties of being non-existent or write protected to aid in the debug and development of the prototype system.
- **Memory Address Translator**—Is an additional feature provided with the Memory Mapper. It enables a block (1024 bytes) of memory in the user system to be physically located at a different address in the Development System.
- **User Clock Integrity**—is verified to insure the clock supplied by the user does not fall below 250KHz.
- **Refresh**—will always occur even when emulation is suspended, to maintain data integrity if dynamic RAM's are employed.
- **I/O Ports**—The user may have access to all I/O Ports when in User Mode and User Clock is selected.

ZDS-1/40 Debug Command Package

The ZDS-1/40 Debug Command Package provides powerful disc based software to support the Z80A-CPU Emulator module. The command repertoire not only includes debug commands for monitoring and analyzing execution of programs under test, but also includes initialization commands to describe the nature of the memory storage being utilized. Once the memory configuration has been established, it may be saved on disc and reloaded as required, thereby eliminating the need to define the memory configuration prior to each emulation session. The following list describes the available commands:

MAP Creates a map that describes the physical nature of the user's memory address space. This enables the user to define memory in blocks of 1024 bytes to exist in either the user system or the Development System. In addition, the properties of each block may be specified as either non-existent or write protected.

DMAP (Display Map) –Allows the user to display a single entry, range of entries or the entire map and allows the user to edit the map.

WRITE Determines whether a break occurs when a memory write is attempted to an address which has the property of being write protected.

BREAK Is used to specify when the Real-Time execution of a user's program will be terminated. The option fields of the command may establish a break to occur on any combination of the following events:
–Address Compare
–Data Compare or Compare with Mask
–Memory Read or Write
–Port Read or Write

DISPLAY MEMORY Is used to display the contents of a memory location or group of locations. If single bytes are displayed the user may change the contents of each byte.

FILL Causes a given data string to be stored in all locations specified by the command.

GET Loads a memory image file from disc and loads its entry address into the program counter.

GO Begins execution of the user's program. An address may be specified from which program execution will begin. If the program had previously terminated execution due to a break, then the GO command will restore EMULATOR status and resume execution.

HISTORY Enables the display of the last n events stored in the Real-Time Storage Module, where n equals from 1 through 256. Data displayed will be:
–Address Bus (16 bits)
–Data Bus (8 bits)
–Control Bus (7 bits)

INTERRUPT MODE Provides for the display, setting and resetting of the interrupt mode of the emulator (mode 0, 1, 2 or disable).

NEXT Provides a means to single step from 1 to n instructions. All CPU registers are displayed after the selected number of instructions has been executed.

OS Is used to exit the Debug environment and cause the operating system to be loaded and initialized.

PORT Enables a single character to be input from any selected port and displayed. The user may then enter a data byte to output to the specified port.

PROM Transfers program control to the resident Debug software contained in the Development System PROM's.

PULSE Is similar to the BREAK command except that program execution is not suspended. Instead a Sync pulse is generated. This pulse is available as a sync for external test equipment.

QUIT Is used to enable return from current program to the calling program.

REGISTER Enables the examination and modification of any or all CPU registers.

SAVE Enables the contents of memory to be copied onto disc along with the entry address. These files may then be retrieved using GET command. The contents are assigned a file name to permit each of location on the disc.

SET Enables the sequential storage of data bytes into memory at the address specified by the user.

STATUS Displays the emulation status, e.g. interrupt mode, clock source, break and trace arguments.

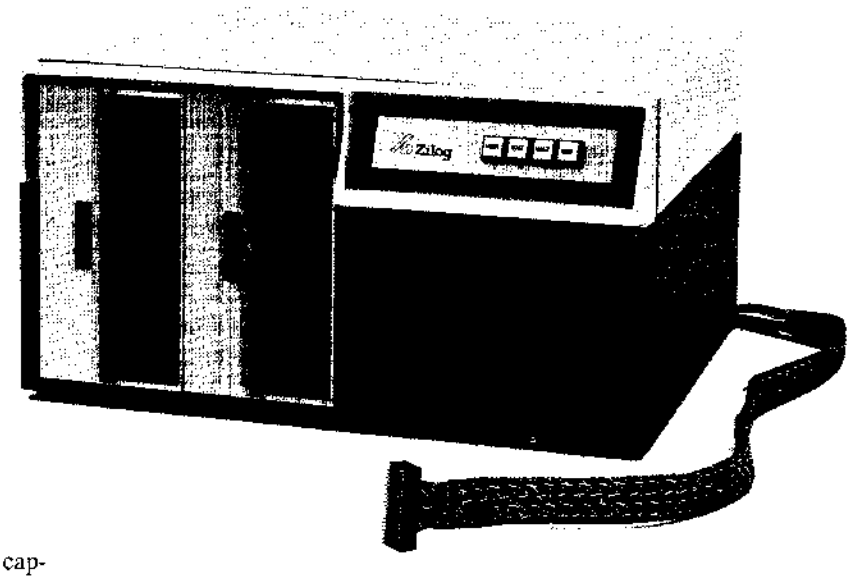
TRACE Allows the user to specify which types of bus transactions will be stored in the Real-Time Storage Module. The user may specify any one or a combination of the following types of operations:
–Memory Reads
–Memory Writes
–Port Reads
–Port Writes

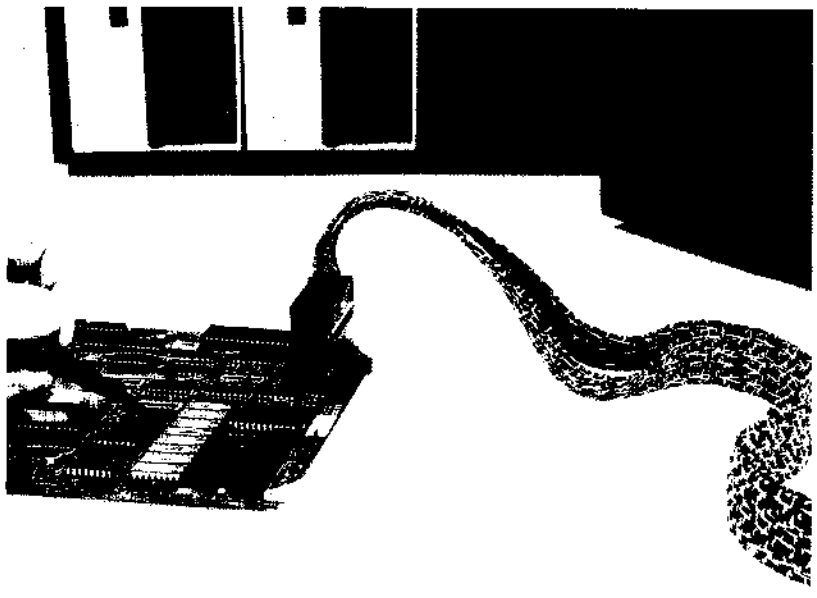
ZDS-1/25 2.5MHz Development System

The ZDS-1/25 Development System is a low cost standalone development system used with the Z80 CPU for prototype development of systems employing clock rates not exceeding 2.5MHz. This system provides a standalone microcomputer with dual floppy discs, optional peripherals and interfaces to assist the user in the software and hardware development activity of a Z80-based system. Two modes of operation are provided to accomplish this: User Mode and Monitor Mode. In Monitor Mode the system enables the user to utilize the RIO operating system to develop, edit and modify his software. In User Mode, the system memory and peripheral devices may partially or totally be allocated to the prototype system, with the exception of I/O ports EØH through EFH. This enables the user to execute his program in a real time environment.

Features

- 6ØK bytes of RAM memory
- 3K PROM Monitor Program with 1K of dedicated Monitor Scratch pad area.
- Programmable Hardware Breakpoint Module enabling the suspension of instruction execution at a given address or activity (Memory or I/O). Programmable Real Time Storage Module to store CPU activity (address, data and control bus), for up to 256 events, to monitor memory or I/O Port activity.
- In-circuit emulator with three foot connection cable to user system.
- Memory mapping to describe the nature of the memory used, either user or ZDS memory. Mapping occurs using blocks containing 256 bytes each.
- Dual Floppy disc drives having a combined storage capacity of 600,000 bytes.
- RIO operating system providing a combination of a Text Editor, Assembler, Linker and ZDOSII File Management System.





ZDS-1/25 Debug Command Package

The ZDS-1/25 Debug Command Package provides the capability to control, analyze and debug programs whic may reside in internal or external prototype memory, or a combination of both. Debug commands are contained in 3K of PROM and use 1K of RAM for a "scratchpad" area. The following is a list of the debug commands and their functions:

BREAK Sets an automatic hardware breakpoint into the real-time debug module. This break can be on a memory read, memory write, I/O port read, or I/O port write. Addresses, data and data masks can also be specified. A break from the user program can also be caused by pressing the Monitor button on the front panel. In either case, a break causes the state of the user's CPU to be stored so that execution can be resumed later, Control returns to the debug level.

COMPARE Allows the user to compare blocks of memory.

DISPLAY Provides access to memory locations. These memory locations may be displayed as a block or observed one at a time for examination and modification.

FILL Allows the user to store a specified data byte throughout a range of memory addresses.

GET Transfers file images formed by the SAVE command into system memory, ready to be executed by the GO Command.

GO Begins execution of the user's program. Execution can begin at any specified address, or it can continue from a previous breakpoint. A programmed or manual break is required to return control back to the debug level.

INTERRUPT STATUS Allows the user to display and modify the state of the interrupt enable flip-flop.

HISTORY Is normally issued after a break from a user program. This instruction lists on the terminal the state of the address, data and control busses of the CPU during the execution of up to 255 bus transactions that occurred in the user's program just prior to a break.

JUMP Transfers control to a starting address of program, but the system remains in the Monitor Mode.

MOVE Allows the user to transfer a block of memory of any size from any location to any other location.

NEXT Executes one or "N" instructions and prints the contents of the CPU registers after each instruction.

PORT Permits examination and/or modification of I/O port data.

PULSE Is identical to Break except that a pulse is provided, via a BNC connector, each time the specified condition occurs, and the program continues to execute. Pulse can be used to synchronize external test equipment.

QUIT Returns control to the OS level.

REGISTER Provides access to the Z80-CPU registers. They may be displayed in their entirety or opened one at a time for examination and change. The register command also allows the user to display and modify the address flag and interrupt mode.

SAVE Stores the RAM image of linked programs and subroutines on the user's disk.

SET Stores data entered from the terminal into specified or memory locations.

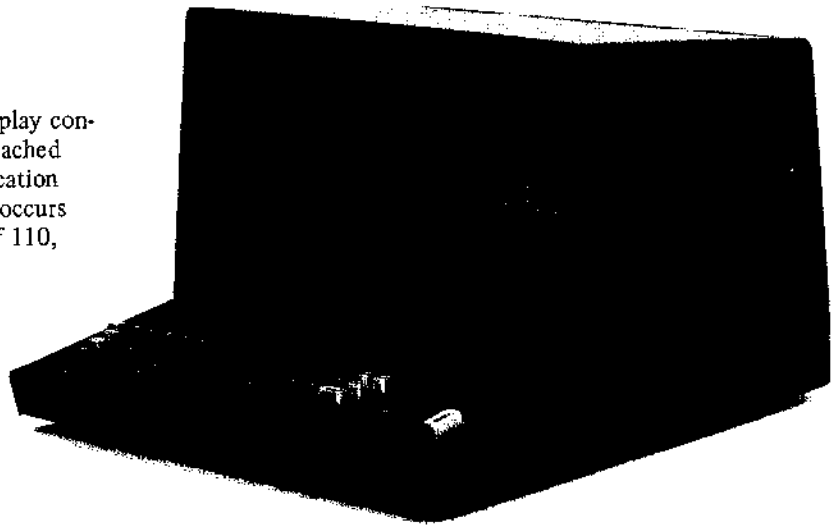
TRACE Specifies if memory read, memory write, port read and/or port write conditions are to be stored in the Real-Time Debug Module during execution of the user's program.

Peripheral Equipment

Zilog provides peripheral equipment to enhance the performance of the ZDS systems to get the job done.

ZDS/CRT Terminal

The CRT Terminal provides an alphanumeric display containing 1920 (80 characters/line x 24 lines) with attached keyboard to serve as the system console. Communication between the terminal and the ZDS-1 Series System occurs in a character-by-character mode at transfer rates of 110, 300, 1200, 2400, 4800 or 9600 baud.



Specifications

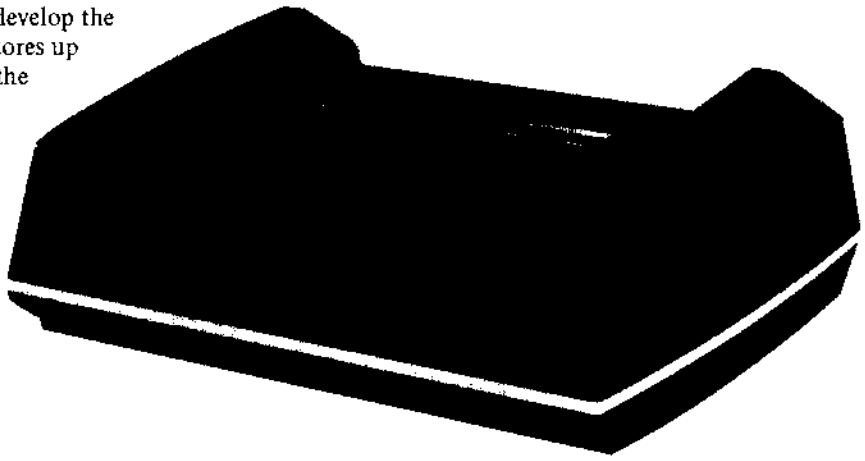
SCREEN FORMAT	
Characters per line	80
Lines per display	24
Character set	96 character ASCII
Character format	5 x 7 dot matrix
MECHANICAL	
TERMINAL	
Size	12" high, 17" wide, 15" deep
Weight	35 pounds
KEYBOARD	
Size	3" high, 17" wide, 8" deep
Weight	5 pounds
ENVIRONMENTAL	
Operating Temperature	0° to 50°C.
Storage Temperature	−30°to 70°C.
Humidity	0 to 95% non-condensing
ELECTRICAL	
Power Consumption	150 watts
Domestic Power	105–130 volts; 60 Hz
Export Power	105–130, 210–260 volts; 50 Hz

ZDS/Printer Terminal

The ZDS/Printer provides a low cost, highly reliable character printer for use with the ZDS-1 Series Development Systems. This printer features excellent print quality with sharply defined characters. Its small size, mobility and attractive appearance make it well suited to the laboratory or office environment.

Both upper and lower case characters of the full 96-character set are impact printed within a 9 x 7 dot matrix to maintain a printing speed of 120 characters per second.

Bidirectional printing allows the printer to develop the shortest printing path. A read/write memory stores up to 132 printable characters that will comprise the next line, allowing "look ahead" for the next print position. The result is faster throughput and increased reliability.



Specifications

Print Speed	120 character/second (bidirectional) 55 lines/min. (132 characters/line) 250 lines/min. (10 characters/line)
Characters/Inch	10
Characters/Line	132
Lines/Inch Vertical	6
Paper Width (Maximum)	15 inches
Character Set	96 USASCII
Forms Length Selector	2 Channel VFU
Forms Specification:	
Type	Continuous fanfold, edge perforated
Width	4 variable 4 inches to 15 inches
Copies	Original and 5 copies
Weight	15 lb. single part, 12 lb. with 7 lb. carbon multipart
Thickness	0.003 to 0.020 inches
Ribbon	Cartridge with continuous loop (0.25" x 15 yds.)

Hardware Expansion Options

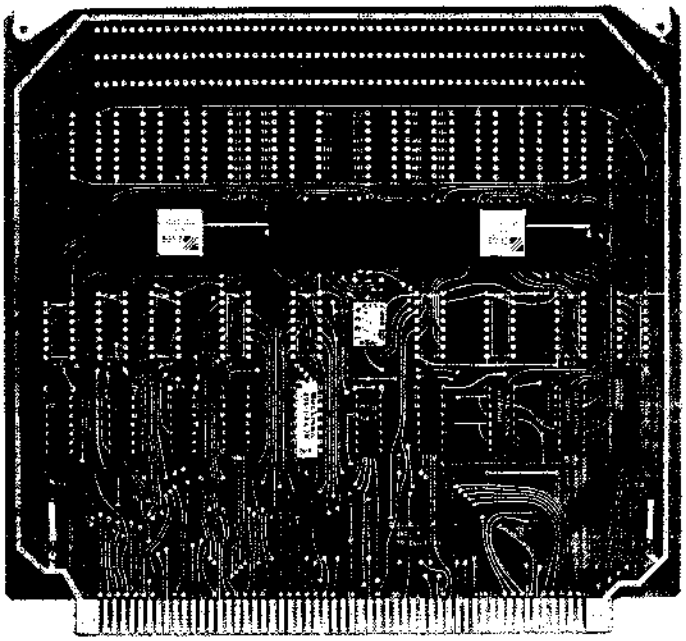
Zilog Offers a variety of hardware expansion features to tailor ZDS-1 Series Development Systems to best suit the users' microprocessor development needs. A standard bus interface structure allows the use of modular boards, each packaged with very high density LSI components for maximum reliability and performance. In addition, software is provided with standard interface to enable immediate use the interface.

ZDS/PIB

PARALLEL INTERFACE BOARD

The PIB provides a basic parallel interface between the ZDS-1 Series Development System and external devices provided by the user. Two Z80-PIO components are provided with supporting logic to give the user 32 bidirectional I/O bits. The card also contains plated through holes for insertion of a Z80-CTC to provide four counter-timer channels. The uncommitted PIO's and/or user supplied CTC can be interfaced with the system's daisy-chain priority interrupt structure. Unused space is provided with 16-pin dip locations (Vcc on pin 16, and GND on pin 8) to allow the addition of logic at the user's discretion. A four-bit dip switch enables I/O port address selection while additional control logic directs port transaction with the system bus.

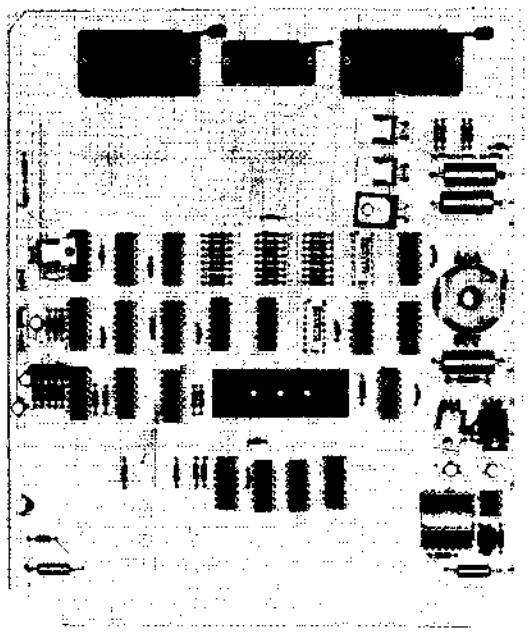
The flexability of the PIB is demonstrated by Zilog in using it to design our own family of interface boards; ASPIO, CIB and RXB.



ZDS/ASPIO

AUXILIARY SERIAL I/O BOARD

The ASPIO board provides one auxiliary parallel communications interface and one auxiliary serial communications interface for the ZDS-1. A parallel interface is designed for communications with the printer supplied by Zilog. Also a printer employing the interface characteristics of the Centronics Model 306C may be used (e.g. Tally Model 1202 or 1602, Wang 200W). A serial interface, utilizing a Universal Asynchronous, Receiver/Transmitter, is provided to enable synchronous or asynchronous, full duplex communication with external devices employing RS-232C interface characteristics. Utility software is provided for the parallel printer interface and diagnostic (exerciser) software is provided for the serial interface.



ZDS/CIB

CENTRONICS INTERFACE BOARD

Two specialized parallel interfaces are provided, one dedicated to interface with a Prolog 90 Series Prom Programmer and the other for a parallel printer interface previously described under ASPIO. Utility software is provided for each interface eliminating the need for the user to develop his own utilities.

ZDS/PPB/16 PROM Programmer Board

The PPB/16 is a functional equivalent of the PPB, previously described, except it is designed to accomodate the following types:

EPROM's – 2716
BIPOLAR – 82S27, 82S181, 82S131

ZDS/PPB PROM Programmer Board

A PPB is a combination PROM/EPROM programmer board which is designed to handle the programming, reading, duplication and verification of the following:

EPROM's – 2704 and 2708
BIPOLAR – 7610, 7611, 7620, 7621, 7640 and 7641

All programming activity is under direct control of the software utility program provided.

Software Expansion

In addition to the RIO Operating System Zilog provides a broad range of higher level languages to simplify and accelerate microprocessor software development activity.

BASIC

Zilog's newest Extended BASIC Interpreter sets a new standard in precision and speed performance for the most popular microcomputer language. Two versions of the system—a version of business BASIC with a 13-digit BDC data type—and a scientific version with the conventional 32-bit floating binary data type—guarantee the optimal mix of speed and precision for any application from professional quality business data processing to extensive scientific calculations. In both of these versions, a 16-bit integer data type provides compact storage and rapid computation for status and control variables. Finally, the string data type permits manipulation of variable length text in a uniquely powerful subscripted notation. Both interpreters have extensive access to disk files through Zilog's ZDOSII disk operating system.

Internally, the Zilog BASIC Interpreter has advanced the state of the art in microcomputer language both in its mathematical packages and in its method of variable reference. Transcendental functions supplied in the Zilog BASIC package use the powerful technique of rational approximations to limit the number of arithmetic operations performed in their evaluation while guaranteeing accuracy. The fundamental arithmetic operations all use rounding of results rather than truncation to further enhance system accuracy. The Interpreter itself maintains all variable references in an encoded form to eliminate the need for execution time symbol table searches. As a result of these advances, BASIC provides the convenience of an interpretive environment, speed normally not found in an interpreter and random access to disk files through a sophisticated disk operating system.

FORTRAN IV

FORTRAN has long been accepted as the standard for scientific programming and is the "native" language of many part-time programmers. Zilog supports FORTRAN with a compiler conforming to the ANSI 1966 specification. This level of FORTRAN is commonly referred to as FORTRAN IV.

Full conformance, with the exception of COMPLEX data types, to the generally accepted ANSI standard insures that accumulated libraries of FORTRAN programs will be immediately usable in the ZDS-1 environment. There is no need to translate to a special microcomputer language. Zilog FORTRAN opens the door to the richest traditions of scientific programming in a small, inexpensive environment.

PLZ

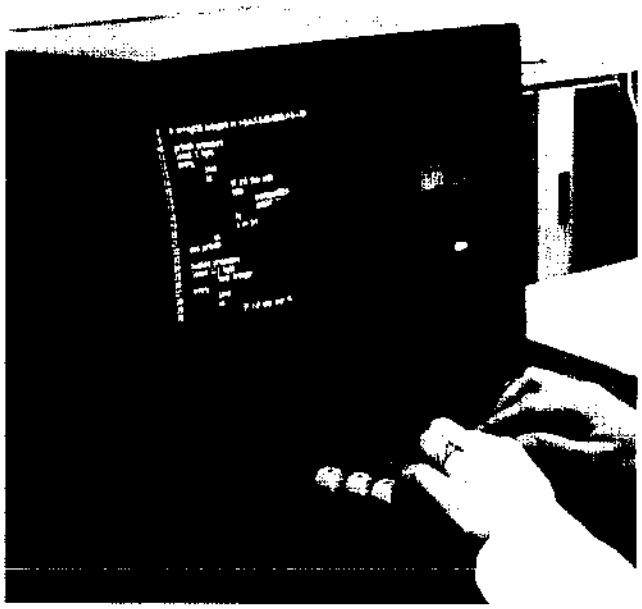
PLZ is Zilog's own family of system implementation languages. Designed specifically with the system programmer in mind, PLZ provides an extremely sophisticated structured programming environment. Implemented at two separate language levels and in distinct translator packages, PLZ offers a complete solution to microcomputer system programming problems.

In its highest level, PLZ is a fully block structured language with an IF-THEN-ELSE clause, a case selector (actually implemented in the syntax of the IF statement), block REPEAT and EXIT statements, procedure references and, of course, RETURN's. Data types include BYTE, WORD, INTEGER, SHORT INTEGER and POINTER as well as ARRAYS of any type and RECORD's composed of groupings of the various basic types, arrays, or even other records.

At this language level, both an Interpreter and a Compiler are available. The interpreter provides a fast and convenient means of program development and debugging, while the highly efficient Compiler generates Z80 Object code.

For applications requiring total control of hardware resources, a second level of PLZ provides all the block structuring and control conventions of the full compiler, but uses assembly language statements rather than compiler level expressions and assignments. Thus, it acts as a PLZ compatible structured assembler and produces code which can be linked directly to modules translated by the compiler.

Regardless of the scope or complexity of a systems programming problem, PLZ will provide a simple, elegant, and complete environment for its solution.



Section 4
Miscellaneous

Zilog

Section 4
Miscellaneous

WE WANT YOU TO KNOW MORE . . .

To assist you in selecting the right Zilog products for your application and to keep you abreast of the latest developments in our technology, each Zilog field office has one or more Field Applications Engineers. They can be reached at the locations below for quick and accurate information about Zilog products:

Regional Offices

EASTERN

Sales and Technical Center
Zilog, Inc.
76 Treble Cove Road
North Billerica, MA 01862
Telephone: (617) 667-2179
TWX 710-347-6660

MIDATLANTIC

Technical Center
Zilog, Inc.
P.O. Box 92
Bergenfield, NJ 07621
Telephone: (201) 385-9158
TWX 710-991-9771

MIDWESTERN

Sales and Technical Center
Zilog, Inc.
830 East Higgins, Suite 114
Schaumburg, IL 60195
Telephone: (312) 885-8080
TWX 910-291-1064

SOUTHWESTERN

Sales and Technical Center
Zilog, Inc.
18001 Sky Park Blvd., Suite K
Irvine, CA 92714
Telephone: (714) 549-2891
TWX 910-595-2803

NORTHWESTERN

Sales and Technical Center
Zilog, Inc.
10340 Bubb Road
Cupertino, CA 95014
Telephone: (408) 446-4666

Additional Zilog Literature

For information on products not covered in this Data Book or for more detailed information than provided here, you may order additional literature direct from Zilog by filling out the literature order form which appears on the next page and sending it to:

Zilog, Inc.
10460 Bubb Road
Cupertino, California 95014
Telephone: (408) 446-4666
TWX 910-338-7621

LITERATURE ORDER LIST

Zilog

10340 Bubb Road
Cupertino, California 95014
Telephone (408) 446-4666
TWX 910-338-7621

TO ORDER ZILOG LITERATURE:

- Give fullname, title, company name and address, and telephone at the top of this literature order list.
- For items requiring payment: Manual orders totaling less than $100.00 must be accompanied by a check or money order. Orders exceeding $100.00 will be processed if accompanied by a purchase order number.
- Large orders i.e., more than five manuals, are shipped via UPS and cannot be shipped to a P.O. Box. UPS delivery time takes from 3–7 working days.
- Small orders are sent Third Class via U.S. Mail. Delivery time for Third Class mail takes from 10–14 days.
- International/Overseas orders are sent Air Mail Printed Matter via U.S. Mail. Delivery time takes up to three weeks.

QTY.	SEND		BACK ORDER
		APPLICATION NOTES	
_____	☐	Z-80 Ap Note (Interrupt)	☐
_____	☐	Z-80 Ap Note (Dynamic RAM)	☐
		BOARDS	
_____	☐	Z-80 Microcomputer Board Catalog	☐
		SYSTEMS	
_____	☐	Z-80 ZDS-1 Series Brochure	☐
_____	☐	Z-80 MCZ-1 Series Brochure	☐
_____	☐	Z-80 MCZ 1/05 Spec Sheet	☐
_____	☐	Z-80 MCZ 1/35 Spec Sheet	☐
_____	☐	Z-80 RIO Spec Sheet	☐
_____	☐	Z-80 Basic Spec Sheet	☐

QTY.	SEND		BACK ORDER
		SOFTWARE (CONT'D)	
_____	☐	Z-80 PLZ Spec Sheet	☐
_____	☐	Z-80 COBOL Spec Sheet	☐
_____	☐	Z-80 Fortran Spec Sheet	☐
		Z-8000	
_____	☐	Z-8000 Architectural Overview	☐
_____	☐	Z-8000 Brochure	☐
		MISCELLANEOUS	
_____	☐	U.S. & Canada Rep & Dist. List	☐
_____	☐	Price List	☐
_____	☐	Other	☐

Orders for these items will not be processed prior to receipt of check or money order. No P.O. numbers please.

QTY.	SEND		BACK ORDER
_____	☐	**Zilog Data Book**	5.00 ☐
		COMPONENTS	
_____	☐	Z-80 CPU Technical Manual	7.50 ☐
_____	☐	Z-80 Program Reference Card	1.95 ☐
_____	☐	Z-80 PIO Technical Manual	4.50 ☐
_____	☐	Z-80 CTC Technical Manual	4.50 ☐
_____	☐	Z-80 SIO Technical Manual	7.50 ☐
		BOARDS	
_____	☐	Z-80 MCB H/W Manual	4.50 ☐
_____	☐	Z-80 MDC/E H/W Manual	4.50 ☐
_____	☐	Z-80 RMB/E H/W Manual	4.50 ☐
_____	☐	Z-80 IOB H/W Manual	2.50 ☐
_____	☐	Z-80 CIB H/W Manual	2.50 ☐
_____	☐	Z-80 SIB H/W Manual	4.50 ☐
_____	☐	*Z-80 PMB H/W Manual	4.50 ☐
_____	☐	Z-80 PPB H/W Manual	4.50 ☐
_____	☐	Z-80 AIO/AIB H/W Manual	4.50 ☐
_____	☐	*Z-80 PPB/16 H/W Manual	4.50 ☐
_____	☐	*Z-80 MCZ 1/05 H/W Manual	4.50 ☐

QTY.	SEND		BACK ORDER
		SYSTEMS (CONT'D)	
_____	☐	Z-80 MCZ 1/20, 1/25 H/W Manual	4.50 ☐
_____	☐	Z-80 MCZ 1/30 H/W Manual	4.50 ☐
_____	☐	Z-80 MCZ 1/35 H/W Manual	4.50 ☐
		SOFTWARE	
_____	☐	Z-80 MCB S/W Manual	4.50 ☐
_____	☐	Z-80 OS 2.1 S/W Manual	10.00 ☐
_____	☐	Z-80 ZDS PROM User Manual	2.00 ☐
_____	☐	*Z-80 MCZ PROM User Manual	2.00 ☐
_____	☐	Z-80 Ass'y Lang. Prog. Manual	7.50 ☐
_____	☐	Z-80 Basic Interpreter User Manual	10.00 ☐
		Z-80 RIO S/W User's Manual (3-piece package:)	
_____	☐	1) Operating System User Manual	15.00 ☐
_____	☐	2) Assembler Linker User Manual	7.50 ☐
_____	☐	3) Text Editor User Manual	7.50 ☐
_____	☐	Z-80 PLZ Language Report	7.50 ☐
_____	☐	Z-80 PLZ Version User Guide	2.50 ☐
_____	☐	Z-80 MCZ COBOL Language Report	15.00 ☐
_____	☐	Z-80 MCZ COBOL User Guide	5.00 ☐

COMMENTS

*Not available at this time. Will not back order.

CUT ALONG DOTTED LINE

Price: $5.00
Printed in U.S.A.

Zilog, Inc.
10340 Bubb Road
Cupertino, California 95014
Telephone (408) 446-4666
TWX 910-338-7621