

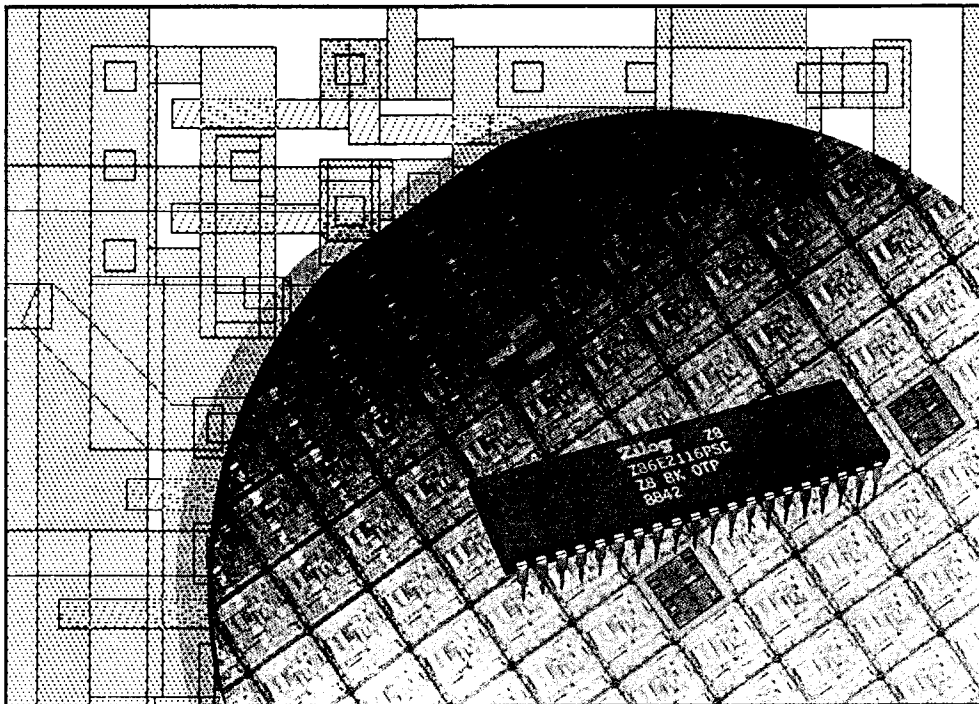


*PRELIMINARY PRODUCT SPECIFICATION*

# Z86C21/E21

CMOS Z8<sup>®</sup> 8K ROM MCU

NOVEMBER 1989





## Z86C21/Z86E21 CMOS CMOS Z8<sup>®</sup> 8K ROM MCU

November 1989

### FEATURES

- Complete microcomputer, **8K** bytes of ROM, **256** bytes of RAM, 32 I/O lines, and up to **56K** bytes addressable external space each for program and data memory.
- **256**-byte register file, including **236** general-purpose registers, **4** I/O port registers, and 16 status and control registers.
- **Minimum instruction execution time of 0.6  $\mu$ s, average of 1.0  $\mu$ s.**
- Vectored, priority interrupts for I/O, counter/timers, and UART.
- Full-duplex UART and two programmable 8-bit counter/timers, each with a 6-bit programmable prescaler.
- **Register Pointer** so that short, fast instructions can access any of **16 working-register groups** in **.6  $\mu$ s**.
- On-chip oscillator which accepts crystal or external clock drive.
- Standby modes—Halt and Stop
- Single +5V power supply—all pins TTL-compatible.
- **12 and 16 MHz.**
- CMOS process
- **Z86E21 compatible field-programmable version — same feature set.**

### GENERAL DESCRIPTION

The Z86C21 microcomputer (Figures 1 and 2) introduces a new level of sophistication to single-chip architecture. Compared to earlier single-chip microcomputers, the Z86C21 offers faster execution;

more efficient use of memory; more sophisticated interrupt, input/output and bit-manipulation capabilities; and easier system expansion.

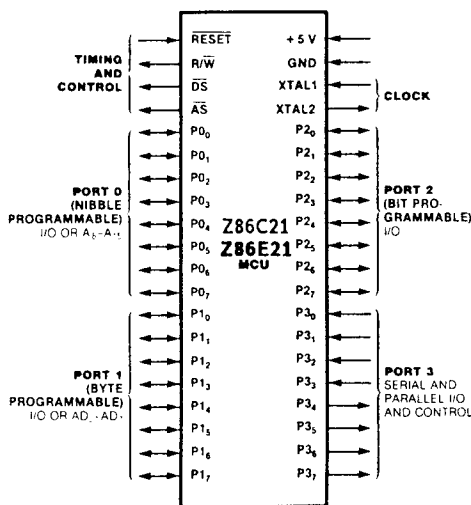


Figure 1. Pin Functions

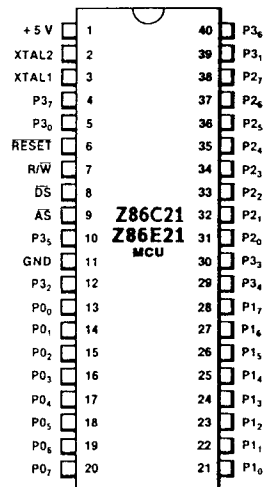


Figure 2. 40-pin Dual-In-Line Package (DIP), Pin Assignments

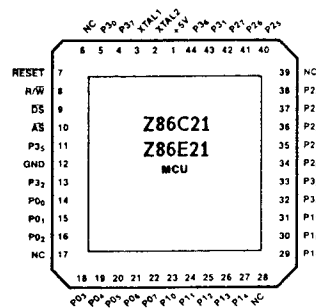
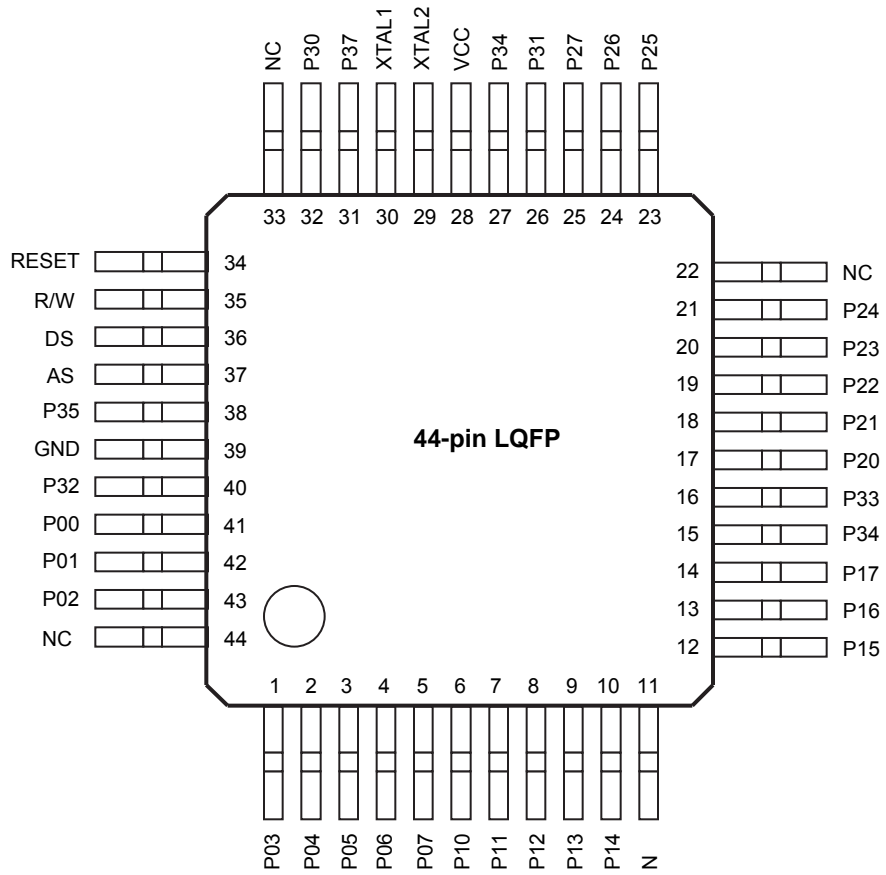


Figure 2b. 44-pin Chip Carrier, Pin Assignments



### General Purpose Microcontroller

Under program control, the Z86C21 can be tailored to the needs of its user. It can be configured as a stand-alone microcomputer with 8K bytes of internal ROM, a traditional microprocessor that manages up to 112K bytes of external memory, or

a parallel-processing element in a system with other processors and peripheral controllers linked by the Z-BUS bus. In all configurations, a large number of pins remain available for I/O.

### Field Programmable Version

The Z86E21 is a pin compatible Onetime Programmable version of the Z86C21. The Z86E21 contains 8K bytes of EPROM memory in place of the 8K bytes of masked ROM on the Z86C21. The

Z86E21 also contains a programmable memory protect feature to provide program security by disabling all external accesses to the internal EPROM array.

## ARCHITECTURE

**Z86C21** architecture is characterized by a flexible I/O scheme, an efficient register and address space structure and a number of ancillary features that are helpful in many applications.

Microcomputer applications demand powerful I/O capabilities. The **Z86C21** fulfills this with 32 pins dedicated to input and output. These lines are grouped into four ports of eight lines each and are configurable under software control to provide timing, status signals, serial or parallel I/O with or without handshake, and an address/data bus for interfacing external memory.

Because the multiplexed address/data bus is merged with the I/O-oriented ports, the **Z86C21** can assume many different memory and I/O configurations. These configurations range from a self-contained microcomputer to a

microprocessor that can address 120K bytes of external memory (Figure 3).

Three basic address spaces are available to support this wide range of configurations: program memory (internal and external), data memory (external) and the register file (internal). The 256-byte random-access register file is composed of 236 general-purpose registers, 4 I/O port registers, and 16 control and status registers.

To unburden the program from coping with real-time problems such as serial data communication and counting/timing, an asynchronous receiver/transmitter (UART) and two counter/timers with a large number of user-selectable modes are offered on-chip. Hardware support for the UART is minimized because one of the on-chip timers supplies the bit rate.

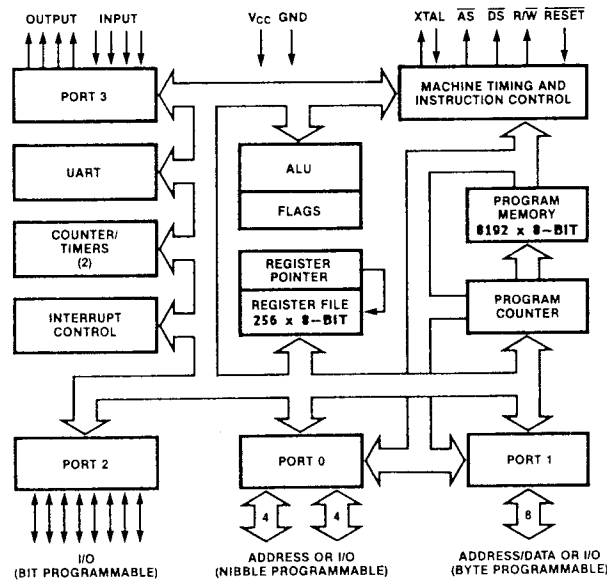


Figure 3. Functional Block Diagram

---

## STANDBY MODE

The Z86C21's standby modes are:

- Stop
- Halt

The Stop instruction stops the internal clock and clock oscillation; the Halt instruction stops the internal clock but not clock oscillation.

A reset input releases the standby mode.

To complete an instruction prior to entering standby mode, use the instructions:

NOP(FF<sub>H</sub>) + STOP(6F<sub>H</sub>)  
NOP(FF<sub>H</sub>) + HALT(7F<sub>H</sub>)

---

## PIN DESCRIPTION

**$\overline{AS}$ .** *Address Strobe* (output, active Low). Address Strobe is pulsed once at the beginning of each machine cycle. Addresses output via Port 1 for all external program or data memory transfers are valid at the trailing edge of  $\overline{AS}$ . Under program control,  $\overline{AS}$  can be placed in the high-impedance state along with Ports 0 and 1, Data Strobe and Read/Write.

**$\overline{DS}$ .** *Data Strobe* (output, active Low). Data Strobe is activated once for each external memory transfer.

**P0<sub>0</sub>-P0<sub>7</sub>, P1<sub>0</sub>-P1<sub>7</sub>, P2<sub>0</sub>-P2<sub>7</sub>, P3<sub>0</sub>-P3<sub>7</sub>.** *I/O Port Lines* (input/outputs, TTL-compatible). These 32 lines are divided into four 8-bit I/O ports that can be configured under program control for I/O or external memory interface (Figure 3).

**$\overline{RESET}$ .** *Reset* (input, active Low).  $\overline{RESET}$  initializes the Z86C21. When  $\overline{RESET}$  is deactivated, program execution begins from internal program location 000C<sub>H</sub>.

**$\overline{R/\overline{W}}$ .** *Read/Write* (output).  $\overline{R/\overline{W}}$  is Low when the Z86C21 is writing to external program or data memory.

**XTAL1, XTAL2.** *Crystal 1, Crystal 2* (time-base input and output). These pins connect a parallel-resonant crystal (12 or 20 MHz maximum) or an external single-phase clock (12 or 20 MHz maximum) to the on-chip clock oscillator and buffer.

---

## ADDRESS SPACE

**Program Memory.** The 16-bit program counter addresses 64K bytes of program memory space. Program memory can be located in two areas: one internal and the other external (Figure 4). The first 8192 bytes consist of on-chip mask-programmed ROM. At addresses 8192 and greater, the Z86C21 executes external program memory fetches.

The first 12 bytes of program memory are reserved for the interrupt vectors. These locations contain six 16-bit vectors that correspond to the six available interrupts.

**Data Memory.** The Z86C21 can address 56K bytes of external data memory beginning at location 4096 (Figure 5). External data memory may be included with or separated from the external program memory space.  $\overline{DM}$ , an optional I/O function that can be programmed to appear on pin P3<sub>4</sub>, is used to distinguish between data and program memory space.

**Register File.** The 256-byte register file includes 4 I/O port registers (R0-R3), 236 general-purpose registers (R4-R239) and 16 control and status registers (R240-R255).

These registers are assigned the address locations shown in Figure 6.

Z86C21 instructions can access registers directly or indirectly with an 8-bit address field. The Z86C21 also allows short 4-bit register addressing using the Register Pointer (one of the control registers). In the 4-bit mode, the register file is divided into 16 working register groups, each occupying 16 contiguous locations (Figure 6). The Register Pointer addresses the starting location of the active working-register group (Figure 7). Note: Register Bank E0-EF can only be accessed through working register and indirect addressing mode.

**Stacks.** Either the internal register file or the external data memory can be used for the stack. A 16-bit Stack Pointer (R254 and R255) is used for the external stack, which can reside anywhere in data memory between locations 4096 and 65535. An 8-bit Stack Pointer (R255) is used for the internal stack that resides within the 124 general-purpose registers (R4-R127).

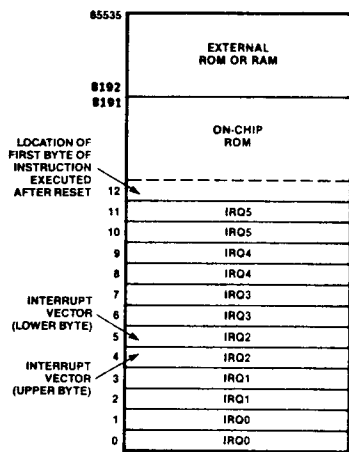


Figure 4. Program Memory Map

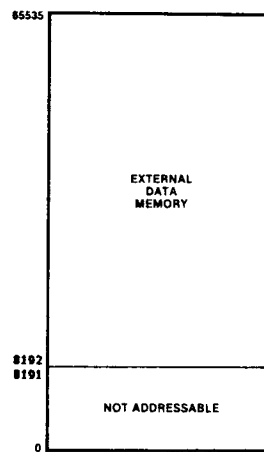


Figure 5. Data Memory Map

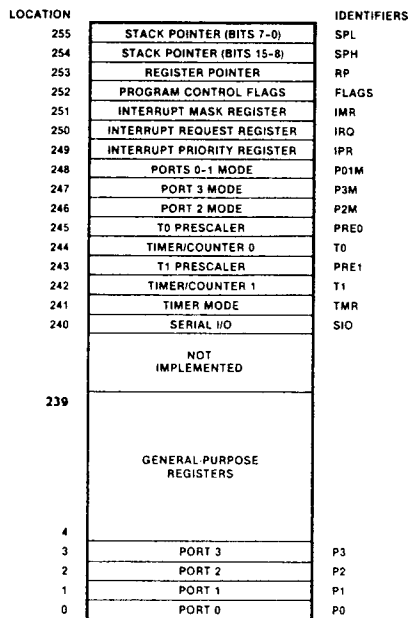


Figure 6. The Register File

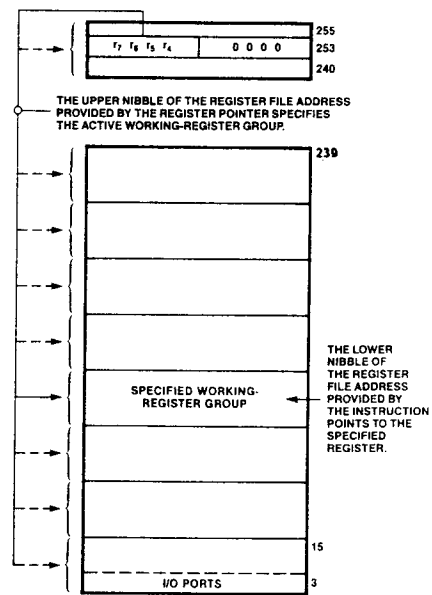


Figure 7. The Register Pointer

## SERIAL INPUT/OUTPUT

Port 3 lines P3<sub>0</sub> and P3<sub>7</sub> can be programmed as serial I/O lines for full-duplex serial asynchronous receiver/transmitter operation. The bit rate is controlled by Counter/Timer 0.

The **Z86C21** automatically adds a start bit and two stop bits to transmitted data (Figure 8). Odd parity is also available as an option. Eight data bits are always transmitted, regardless

of parity selection. If parity is enabled, the eighth bit is the odd parity bit. An interrupt request (IRQ<sub>4</sub>) is generated on all transmitted characters.

Received data must have a start bit, eight data bits and at least one stop bit. If parity is on, bit 7 of the received data is replaced by a parity error flag. Received characters generate the IRQ<sub>3</sub> interrupt request.

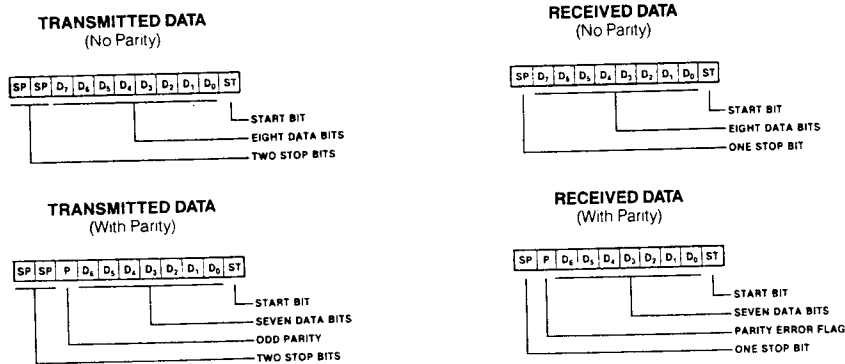


Figure 8. Serial Data Formats

## COUNTER/TIMERS

The **Z86C21** contains two 8-bit programmable counter/timers (T<sub>0</sub> and T<sub>1</sub>), each driven by its own 6-bit programmable prescaler. The T<sub>1</sub> prescaler can be driven by internal or external clock sources; however, the T<sub>0</sub> prescaler is driven by the internal clock only.

The 6-bit prescalers can divide the input frequency of the clock source by any number from 1 to 64. Each prescaler drives its counter, which decrements the value (1 to 256) that has been loaded into the counter. When the counter reaches the end of count, a timer interrupt request—IRQ<sub>4</sub> (T<sub>0</sub>) or IRQ<sub>5</sub> (T<sub>1</sub>)—is generated.

The counters can be started, stopped, restarted to continue, or restarted from the initial value. The counters can also be programmed to stop upon reaching zero (single-pass mode) or to automatically reload the initial value and

continue counting (modulo-n continuous mode). The counters, but not the prescalers, can be read any time without disturbing their value or count mode.

The clock source for T<sub>1</sub> is user-definable and can be the internal microprocessor clock divided by four, or an external signal input via Port 3. The Timer Mode register configures the external timer input as an external clock (1MHz maximum), a trigger input that can be retriggerable or non-retriggerable, or as a gate input for the internal clock. The counter/timers can be programmably cascaded by connecting the T<sub>0</sub> output to the input of T<sub>1</sub>. Port 3 line P3<sub>6</sub> also serves as a timer output (T<sub>OUT</sub>) through which T<sub>0</sub>, T<sub>1</sub> or the internal clock can be output.

## I/O PORTS

The **Z86C21** has 32 lines dedicated to input and output. These lines are grouped into four ports of eight lines each and are configurable as input, output or address/data. Under software control, the ports can be programmed to provide address outputs, timing, status signals, serial I/O, and parallel I/O with or without handshake. All ports have active pull-ups and pull-downs compatible with TTL loads.

**Port 1** can be programmed as a byte I/O port or as an address/data port for interfacing external memory. When used as an I/O port, Port 1 may be placed under handshake control. In this configuration, Port 3 lines P3<sub>3</sub> and P3<sub>4</sub> are used as the handshake controls RDY<sub>1</sub> and DAV<sub>1</sub> (Ready and Data Available).

Memory locations greater than **8192** are referenced through Port 1. To interface external memory, Port 1 must be programmed for the multiplexed Address/Data mode. If more than 256 external locations are required, Port 0 must output the additional lines.

**Port 0** can be programmed as a nibble I/O port, or as an address port for interfacing external memory. When used as an I/O port, Port 0 may be placed under handshake control. In this configuration, Port 3 lines P3<sub>2</sub> and P3<sub>5</sub> are used as the handshake controls  $\overline{\text{DAV}}_0$  and RDY<sub>0</sub>. Handshake signal assignment is dictated by the I/O direction of the upper nibble P0<sub>4</sub>-P0<sub>7</sub>.

For external memory references, Port 0 can provide address bits A<sub>8</sub>-A<sub>11</sub> (lower nibble) or A<sub>8</sub>-A<sub>15</sub> (lower and upper nibble) depending on the required address space. If the address range requires 12 bits or less, the upper nibble of Port 0 can be programmed independently as I/O while the lower nibble

**Port 2** bits can be programmed independently as input or output. This port is always available for I/O operations. In addition, Port 2 can be configured to provide open-drain outputs.

Like Ports 0 and 1, Port 2 may also be placed under handshake control. In this configuration, Port 3 lines P3<sub>1</sub> and P3<sub>6</sub> are used as the handshake controls lines  $\overline{\text{DAV}}_2$  and RDY<sub>2</sub>. The handshake signal assignment for Port 3 lines P3<sub>1</sub> and P3<sub>6</sub> is dictated by the direction (input or output) assigned to bit 7 of Port 2.

**Port 3** lines can be configured as I/O or control lines. In either case, the direction of the eight lines is fixed as four input (P3<sub>0</sub>-P3<sub>3</sub>) and four output (P3<sub>4</sub>-P3<sub>7</sub>). For serial I/O, lines P3<sub>0</sub> and P3<sub>7</sub> are programmed as serial in and serial out respectively.

Port 3 can also provide the following control functions: handshake for Ports 0, 1 and 2 ( $\overline{\text{DAV}}$  and RDY); four external interrupt request signals (IRQ<sub>0</sub>-IRQ<sub>3</sub>); timer input and output signals (T<sub>IN</sub> and T<sub>OUT</sub>) and Data Memory Select (DM).

Port 1 can be placed in the high-impedance state along with Port 0,  $\overline{\text{AS}}$ ,  $\overline{\text{DS}}$  and R/W, allowing the **Z86C21** to share common resources in multiprocessor and DMA applications. Data transfers can be controlled by assigning P3<sub>3</sub> as a Bus Acknowledge input, and P3<sub>4</sub> as a Bus Request output.

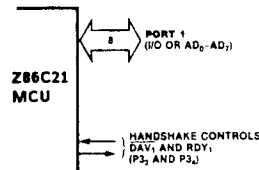


Figure 9a. Port 1

is used for addressing. When Port 0 nibbles are defined as address bits, they can be set to the high-impedance state along with Port 1 and the control signals  $\overline{\text{AS}}$ ,  $\overline{\text{DS}}$  and R/W.

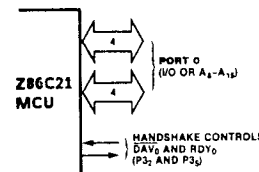


Figure 9b. Port 0

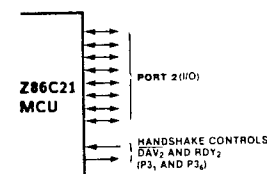


Figure 9c. Port 2

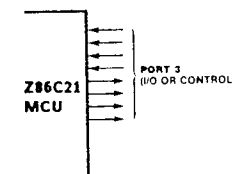


Figure 9d. Port 3



---

## INTERRUPTS

The **Z86C21** allows six different interrupts from eight sources: the four Port 3 lines P3<sub>0</sub>-P3<sub>3</sub>, Serial In, Serial Out, and the two counter/timers. These interrupts are both maskable and prioritized. The Interrupt Mask register globally or individually enables or disables the six interrupt requests. When more than one interrupt is pending, priorities are resolved by a programmable priority encoder that is controlled by the Interrupt Priority register.

**All Z86C21 interrupts are vectored through locations in program memory. When an interrupt request is granted, an interrupt machine cycle is entered. This disables all**

**subsequent interrupts, saves the Program Counter and status flags, and branches to the program memory vector location reserved for that interrupt. This memory location and the next byte contain the 16-bit address of the interrupt service routine for that particular interrupt request.**

Polled interrupt systems are also supported. To accommodate a polled structure, any or all of the interrupt inputs can be masked and the Interrupt Request register polled to determine which of the interrupt requests needs service.

---

## CLOCK

The on-chip oscillator has a high-gain, parallel-resonant amplifier for connection to a crystal or to any suitable external clock source (XTAL1 = Input, XTAL2 = Output).

The crystal source is connected across XTAL1 and XTAL2, using the recommended capacitors ( $C_1 \leq 15$  pF) from each

pin to ground. The specifications for the crystal are as follows:

- AT cut, parallel resonant
  - Fundamental type, 16 MHz maximum
  - Series resistance,  $R_s \leq 100 \Omega$
- 

## GENERAL DESCRIPTION

The Z86C12 development device allows users to prototype a system with an actual hardware device and to develop the code. This code is eventually mask-programmed into the on-chip ROM for any of the 86Cxx devices (except the 86C91). Development devices are also useful in emulator applications where the final system configuration -- memory configuration, I/O, interrupt inputs, etc. -- are unknown. The Z86C12 development device is identical to its equivalent Z86C21 microcomputer with the following exceptions:

- No internal ROM is provided, so that code is developed in off-chip memory. Five "size" inputs configure the memory boundaries.

■ The normally internal ROM address and data lines are buffered and brought out to external pins to interface with the external memory.

- Control lines (/MAS and /MDS) are added to interface with external program memory.

The Timing and Control, I/O ports, and clock pins on the Z86C12 are identical in function to those on the 86C21. This section covers those pins that do not appear on the Z86C21 8K ROM device. The pin functions and pin assignments are shown on figure 00.

---

## Z86C12 PIN DESCRIPTION

**D0 - D7 (Inputs, TTL compatible) Data bus.** These 8 lines provide the input data bus to access external memory emulating on the on-chip ROM. During read cycles in the internal memory space the data on these lines is latched in just prior to the rise of the /MDS data strobe.

**A0 - A15 (Output, TTL compatible) Address bus.** During T1 these lines output the current memory address. All addresses, whether internal or external, are output.

**/MAS (Output, TTL compatible) Memory Address Strobe.** This line is active during every T1 cycle. The rising edge of this signal may be used to latch the current memory address on the lines A0 - A15. This line is always valid; it is not tri-stated when /AS is tri-stated.

**/MDS (Output, TTL compatible) Memory Data Strobe.** This is a timing signal used to enable the external memory to emulate the on-chip ROM. It is active only during accesses to the on-chip ROM memory space, as selected by the configuration of the SIZE<sub>n</sub> pins.

**/SCLK (Output, TTL compatible) System Clock.** This line is the internal system clock.

**/SYNC (Output, TTL, compatible) Sync signal.** This signal indicates the last clock cycle of the currently executing instruction.

**/IACK (Output, TTL, compatible) Interrupt Acknowledge.** This output, when low, indicates that the Z86C12 is in an interrupt cycle.

---

**/SIZE0, /SIZE1, /SIZE2, /SIZE3, SIZE4** (inputs, TTL compatible). The /SIZE<sub>n</sub> lines control the emulation mode of the 86C12. Note that /SIZE0 - /SIZE3 are active low, while SIZE4 is active high. The functions are defined as shown in figure 00. The 86C12 should be in RESET when the state of these lines are changed.

**NOTE:**

The SIZE pins may be configured to make the memory control signals (/MAS, /MDS, R/W, /AS, and /DS) look like the Z86C91 ROMless device, however on power-up or reset ports 0 and 1 are configured as inputs, rather than A15 - A8 and AD7 - AD0, respectively.

**Table 1. Z86C12 Pin Assignments**

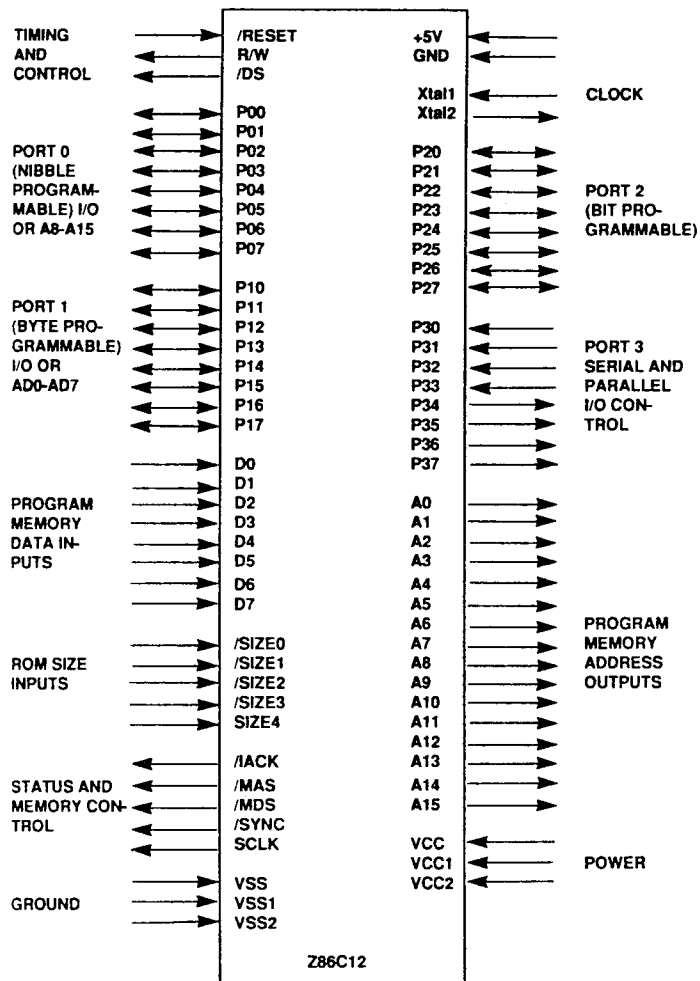
NAME		NAME	PIN	NAME	PIN	NAME	PIN
/AS	B2	A8	J5	P07	J1	P36	A7
/DS	C4	A9	K4	P10	G8	P37	A5
/MAS	E1	D0	H3	P11	G9	R/W	A1
/MDS	G3	D1	K2	P12	G10	SCLK	G2
/RESET	B3	D2	J3	P13	F8	SIZE4	F10
/SIZE0	A3	D3	K3	P14	D10	VCC	A4
/SIZE1	C5	D4	H8	P15	C10	VCC1	B6
/SIZE2	A6	D5	J10	P16	B10	VCC2	F9
/SIZE3	C6	D6	H9	P17	E9	VSS	F3
/SYNC	F1	D7	H10	P20	C9	VSS1	E2
A0	J9	IACK	F2	P21	A10	VSS2	H6
A1	H7	NC	J2	P22	B9	VSS3	E8
A10	J4	NC	C3	P23	C8	Xtal1	B5
A11	H4	NC	D8	P24	A9	Xtal2	A2
A12	K9	NC	H2	P25	B8		
A13	K7	NC	K1	P26	A8		
A14	K5	P00	C1	P27	C7		
A15	H5	P01	D3	P30	B4		
A2	K10	P02	D2	P31	B7		
A3	J8	P03	D1	P32	C2		
A4	J7	P04	E3	P33	D9		
A5	K6	P05	G1	P34	E10		
A6	J6	P06	H1	P35	B1		
A7	K8						

**Table 2. Memory Size Configuration**

SIZE4	/SIZE3	/SIZE2	/SIZE1	/SIZE0	MEMORY
0	1	1	1	1	ROMless
0	1	1	1	0	2K ROM
0	1	1	0	1	4K ROM
0	1	0	1	1	8K ROM
0	0	1	1	1	16K ROM
1	1	1	1	1	32K ROM

	1	2	3	4	5	6	7	8	9	10
A	.	.	.	.	.	.	.	.	.	.
B	.	.	.	.	.	.	.	.	.	.
C	.	.	.	.	.	.	.	.	.	.
D	.	.	.	.	.	.	.	.	.	.
E	.	.	.	.	.	.	.	.	.	.
F	.	.	.	.	.	.	.	.	.	.
G	.	.	.	.	.	.	.	.	.	.
H	.	.	.	.	.	.	.	.	.	.
J	.	.	.	.	.	.	.	.	.	.
K	.	.	.	.	.	.	.	.	.	.

TOP VIEW



Z86C12 Pin Functions

## INSTRUCTION SET NOTATION

**Addressing Modes.** The following notation is used to describe the addressing modes and instruction operations as shown in the instruction summary.

<b>IRR</b>	Indirect register pair or indirect working-register pair address
<b>Irr</b>	Indirect working-register pair only
<b>X</b>	Indexed address
<b>DA</b>	Direct address
<b>RA</b>	Relative address
<b>IM</b>	Immediate
<b>R</b>	Register or working-register address
<b>r</b>	Working-register address only
<b>IR</b>	Indirect-register or indirect working-register address
<b>Ir</b>	Indirect working-register address only
<b>RR</b>	Register pair or working register pair address

**Symbols.** The following symbols are used in describing the instruction set.

<b>dst</b>	Destination location or contents
<b>src</b>	Source location or contents
<b>cc</b>	Condition code (see list)
<b>@</b>	Indirect address prefix
<b>SP</b>	Stack pointer (control registers 254-255)
<b>PC</b>	Program counter
<b>FLAGS</b>	Flag register (control register 252)
<b>RP</b>	Register pointer (control register 253)
<b>IMR</b>	Interrupt mask register (control register 251)

Assignment of a value is indicated by the symbol " $\leftarrow$ ". For example,

$$\text{dst} \leftarrow \text{dst} + \text{src}$$

indicates that the source data is added to the destination data and the result is stored in the destination location. The notation "addr(n)" is used to refer to bit "n" of a given location. For example,

$$\text{dst}(7)$$

refers to bit 7 of the destination operand.

**Flags.** Control Register R252 contains the following six flags:

<b>C</b>	Carry flag
<b>Z</b>	Zero flag
<b>S</b>	Sign flag
<b>V</b>	Overflow flag
<b>D</b>	Decimal-adjust flag
<b>H</b>	Half-carry flag

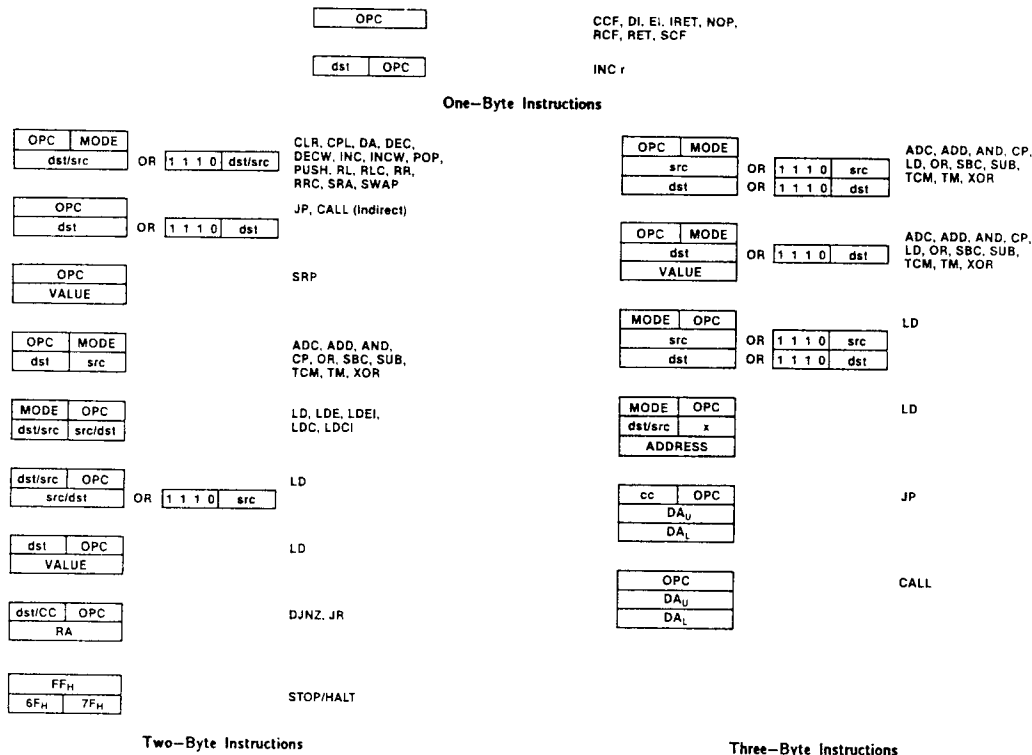
Affected flags are indicated by:

<b>0</b>	Cleared to zero
<b>1</b>	Set to one
<b>*</b>	Set or cleared according to operation
<b>—</b>	Unaffected
<b>X</b>	Undefined

## CONDITION CODES

Value	Mnemonic	Meaning	Flags Set
1000		Always true	—
0111	C	Carry	C = 1
1111	NC	No carry	C = 0
0110	Z	Zero	Z = 1
1110	NZ	Not zero	Z = 0
1101	PL	Plus	S = 0
0101	MI	Minus	S = 1
0100	OV	Overflow	V = 1
1100	NOV	No overflow	V = 0
0110	EQ	Equal	Z = 1
1110	NE	Not equal	Z = 0
1001	GE	Greater than or equal	(S XOR V) = 0
0001	LT	Less than	(S XOR V) = 1
1010	GT	Greater than	[Z OR (S XOR V)] = 0
0010	LE	Less than or equal	[Z OR (S XOR V)] = 1
1111	UGE	Unsigned greater than or equal	C = 0
0111	ULT	Unsigned less than	C = 1
1011	UGT	Unsigned greater than	(C = 0 AND Z = 0) = 1
0011	ULE	Unsigned less than or equal	(C OR Z) = 1
0000		Never true	—

# INSTRUCTION FORMATS



# INSTRUCTION SUMMARY

Instruction and Operation	Addr Mode		Opcode Byte (Hex)	Flags Affected						
	dst	src		C	Z	S	V	D	H	
<b>ADC</b> dst,src dst ← dst + src + C	(Note 1)		1□	*	*	*	*	0	*	
<b>ADD</b> dst,src dst ← dst + src	(Note 1)		0□	*	*	*	*	0	*	
<b>AND</b> dst,src dst ← dst AND src	(Note 1)		5□	-	*	*	0	-	-	
<b>CALL</b> dst SP ← SP - 2 @SP ← PC, PC ← dst	DA		D6	-	-	-	-	-	-	
	IRR		D4	-	-	-	-	-	-	
<b>CCF</b> C ← NOT C			EF	*	-	-	-	-	-	
<b>CLR</b> dst dst ← 0	R		B0	-	-	-	-	-	-	
	IR		B1	-	-	-	-	-	-	
<b>COM</b> dst dst ← NOT dst	R		60	-	*	*	0	-	-	
	IR		61	-	*	*	0	-	-	
<b>CP</b> dst,src dst - src	(Note 1)		A□	*	*	*	*	-	-	

Instruction and Operation	Addr Mode		Opcode Byte (Hex)	Flags Affected						
	dst	src		C	Z	S	V	D	H	
<b>JP</b> cc,dst if cc is true PC ← dst	DA		cD	-	-	-	-	-	-	
	IRR		30	c = 0 - F						
<b>JR</b> cc,dst if cc is true, PC ← PC + dst Range: +127, -128	RA		cB	-	-	-	-	-	-	
				c = 0 - F						
<b>LD</b> dst,src dst ← src	r	im	rC	-	-	-	-	-	-	
	r	R	r8							
	R	r	r9							
				r = 0 - F						
	r	X	C7							
	X	r	D7							
	r	lr	E3							
	lr	r	F3							
	R	R	E4							
	R	IR	E5							
	R	IM	E6							
	IR	IM	E7							
	IR	R	F5							

## INSTRUCTION SUMMARY (Continued)

Instruction and Operation	Addr Mode		Opcode Byte (Hex)	Flags Affected					
	dst	src		C	Z	S	V	D	H
<b>DA</b> dst dst ← DA dst	R		40	*	*	*	X	--	--
	IR		41						
<b>DEC</b> dst dst ← dst - 1	R		00	--	*	*	*	--	--
	IR		01						
<b>DECW</b> dst dst ← dst - 1	RR		80	--	*	*	*	--	--
	IR		81						
<b>DI</b> IMR(7) ← 0			8F	--	--	--	--	--	--
<b>DJNZ</b> r,dst r ← r - 1 if r ≠ 0 PC ← PC + dst Range: +127, -128	RA		rA	--	--	--	--	--	--
			r = 0 - F						
<b>EI</b> IMR(7) ← 1			9F	--	--	--	--	--	--
<b>HALT</b>			7F						
<b>INC</b> dst dst ← dst + 1	r		rE	--	*	*	*	--	--
			r = 0 - F						
	R		20						
	IR		21						
<b>INCW</b> dst dst ← dst + 1	RR		A0	--	*	*	*	--	--
	IR		A1						
<b>IRET</b> FLAGS ← @SP; SP ← SP + 1 PC ← @SP; SP ← SP + 2; IMR(7) ← 1			BF	*	*	*	*	*	*
<b>RLC</b> dst	R		10	*	*	*	*	--	--
	IR		11						
<b>RR</b> dst	R		E0	*	*	*	*	--	--
	IR		E1						
<b>RRC</b> dst	R		C0	*	*	*	*	--	--
	IR		C1						
<b>SBC</b> dst,src dst ← dst ← src ← C	(Note 1)		3□	*	*	*	*	1	*
<b>SCF</b> C ← 1			DF	1	--	--	--	--	--
<b>SRA</b> dst	R		D0	*	*	*	*	0	--
	IR		D1						
<b>SRP</b> src RP ← src		Im	31	--	--	--	--	--	--
<b>STOP</b>			6F						
<b>SUB</b> dst,src dst ← dst ← src	(Note 1)		2□	*	*	*	*	1	*
<b>SWAP</b> dst	R		F0	X	*	*	X	--	--
	IR		F1						
<b>TCM</b> dst,src (NOT dst) AND src	(Note 1)		6□	--	*	*	*	0	--

12

Instruction and Operation	Addr Mode		Opcode Byte (Hex)	Flags Affected					
	dst	src		C	Z	S	V	D	H
<b>LDC</b> dst,src dst ← src	r	lrr	C2	--	--	--	--	--	--
	lrr	r	D2						
<b>LDCI</b> dst,src dst ← src r ← r + 1; rr ← rr + 1	lr	lrr	C3	--	--	--	--	--	--
	lrr	lr	D3						
<b>LDE</b> dst,src dst ← src	r	lrr	82	--	--	--	--	--	--
	lrr	r	92						
<b>LDEI</b> dst,src dst ← src r ← r + 1; rr ← rr + 1	lr	lrr	83	--	--	--	--	--	--
	lrr	lr	93						
<b>NOP</b>			FF	--	--	--	--	--	--
<b>OR</b> dst,src dst ← dst OR src	(Note 1)		4□	--	*	*	*	0	--
<b>POP</b> dst dst ← @SP; SP ← SP + 1	R		50	--	--	--	--	--	--
	IR		51						
<b>PUSH</b> src SP ← SP - 1; @SP ← src	R		70	--	--	--	--	--	--
	IR		71						
<b>RCF</b> C ← 0			CF	0	--	--	--	--	--
<b>RET</b> PC ← @SP; SP ← SP + 2			AF	--	--	--	--	--	--
<b>RL</b> dst	R		90	*	*	*	*	--	--
	IR		91						
<b>TM</b> dst,src dst AND src	(Note 1)		7□	--	*	*	*	0	--
<b>XOR</b> dst,src dst ← dst XOR src	(Note 1)		B□	--	*	*	*	0	--

NOTE: These instructions have an identical set of addressing modes which are encoded for brevity. The first opcode nibble is found in the instruction set table above. The second nibble is expressed symbolically by a □ in this table, and its value is found in the following table to the left of the applicable addressing mode pair.

For example, the opcode of an ADC instruction using the addressing modes r (destination) and lr (source) is 13

Addr Mode		Lower Opcode Nibble
dst	src	
r	r	2
r	lrr	3
R	R	4
R	IR	5
R	IM	6
IR	IM	7

# REGISTERS

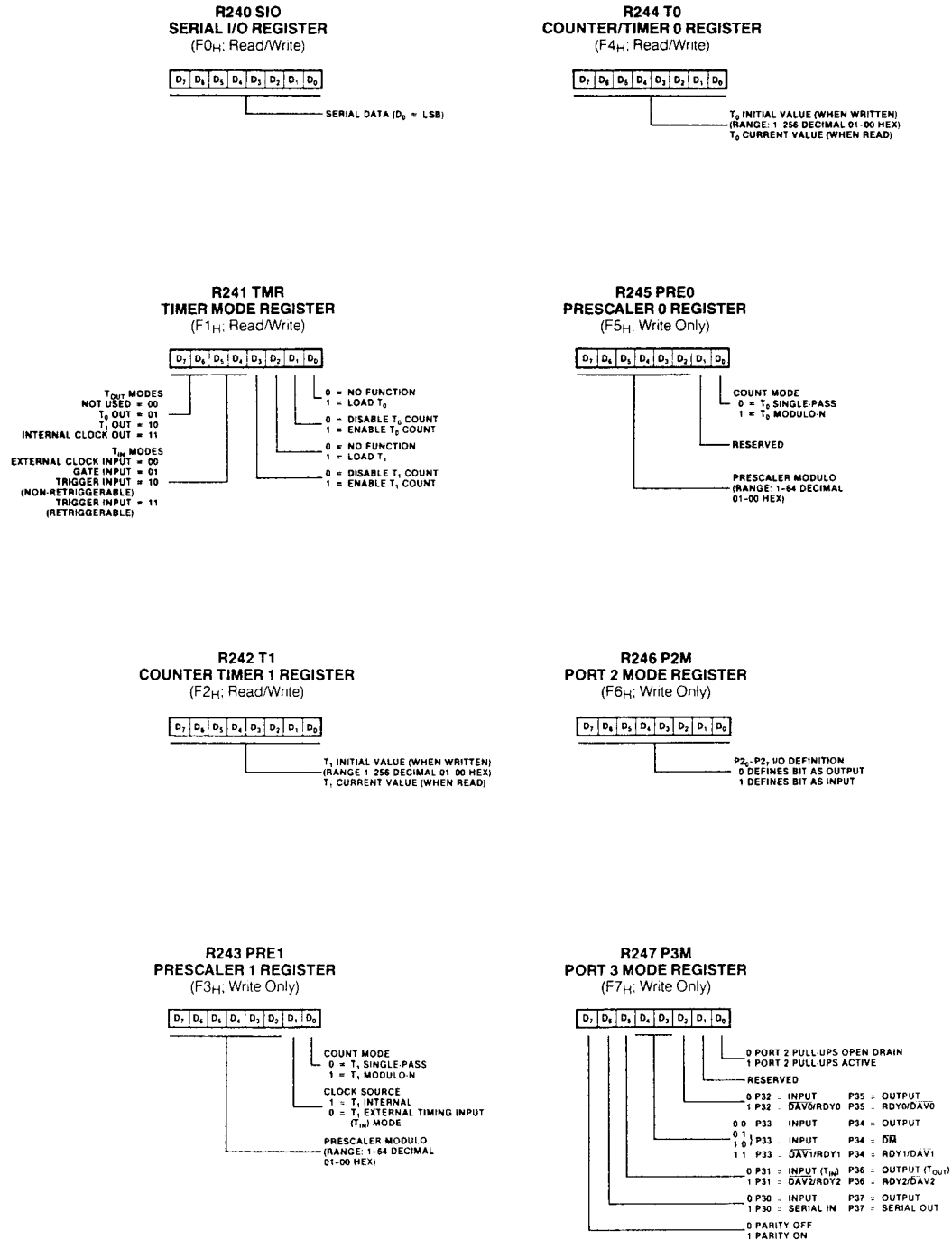
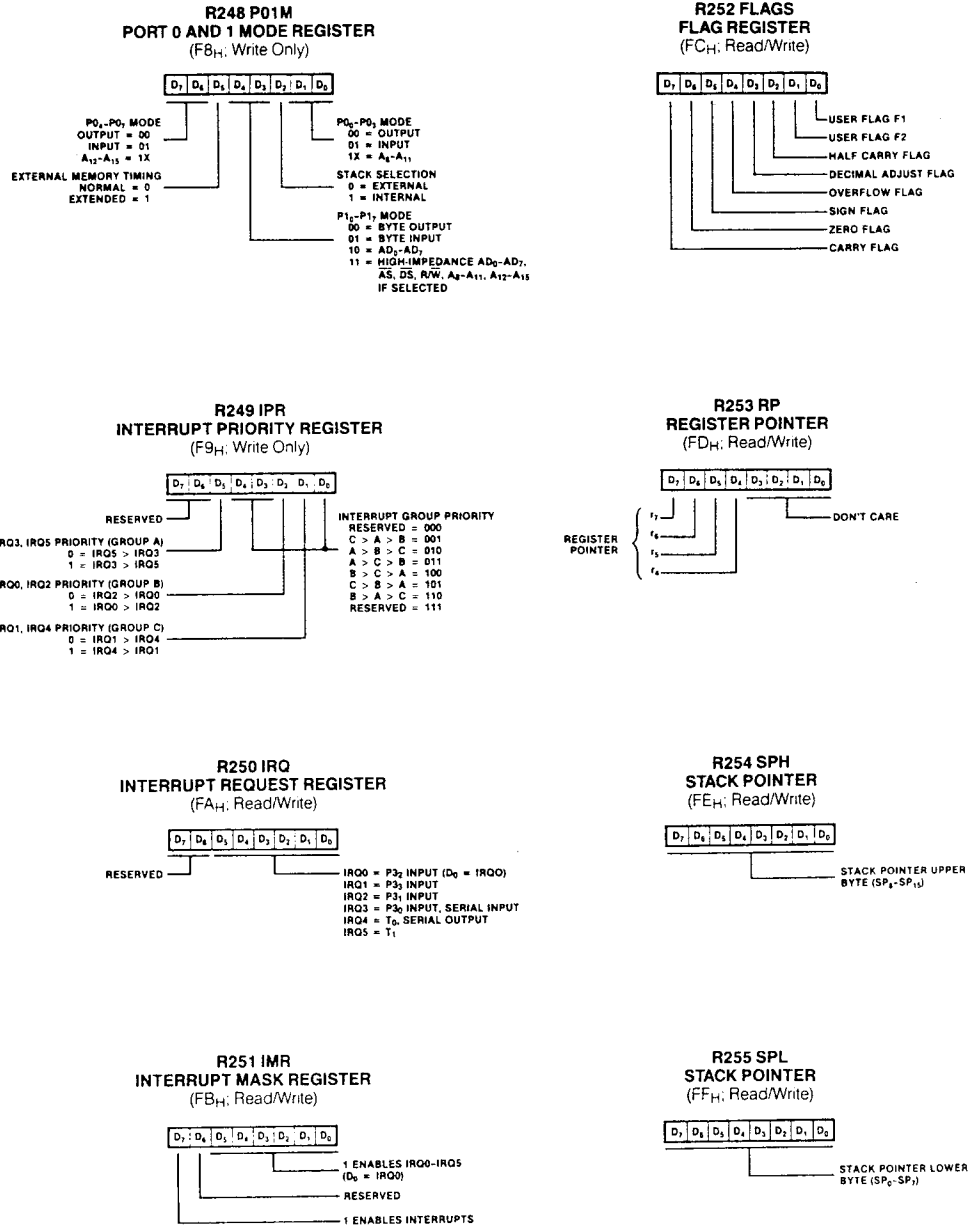


Figure 11. Control Registers

**REGISTERS** (Continued)



### R250 IRQ INTERRUPT REQUEST REGISTER (FA<sub>H</sub>; Read/Write)

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------

**RESERVED**

IRQ0 = P<sub>2</sub> INPUT (D<sub>0</sub> = IRQ0)  
 IRQ1 = P<sub>3</sub> INPUT  
 IRQ2 = P<sub>3</sub> INPUT  
 IRQ3 = P<sub>3</sub> INPUT, SERIAL INPUT  
 IRQ4 = T<sub>0</sub> SERIAL OUTPUT  
 IRQ5 = T<sub>1</sub>

### R254 SPH STACK POINTER (FE<sub>H</sub>; Read/Write)

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------

STACK POINTER UPPER BYTE (SP<sub>U</sub>, SP<sub>H</sub>)

### R251 IMR INTERRUPT MASK REGISTER (FB<sub>H</sub>; Read/Write)

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------

- 1 ENABLES IRQ0-IRQ5 (D<sub>0</sub> = IRQ0)
- RESERVED
- 1 ENABLES INTERRUPTS

### R255 SPL STACK POINTER (FF<sub>H</sub>; Read/Write)

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------

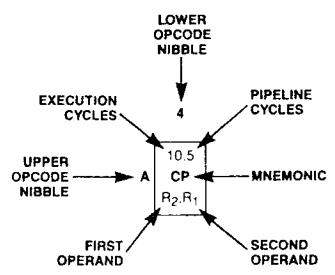
STACK POINTER LOWER BYTE (SP<sub>L</sub>, SP<sub>L</sub>)

Figure 11. Control Registers (Continued)



# OPCODE MAP

		Lower Nibble (Hex)																
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
Upper Nibble (Hex)	0	6.5 DEC R <sub>1</sub>	6.5 DEC IR <sub>1</sub>	6.5 ADD r <sub>1</sub> .r <sub>2</sub>	6.5 ADD r <sub>1</sub> .r <sub>2</sub>	10.5 ADD R <sub>2</sub> .R <sub>1</sub>	10.5 ADD IR <sub>2</sub> .R <sub>1</sub>	10.5 ADD R <sub>1</sub> .IM	10.5 ADD IR <sub>1</sub> .IM	6.5 LD r <sub>1</sub> .R <sub>2</sub>	6.5 LD r <sub>2</sub> .R <sub>1</sub>	12/10.5 DJNZ r <sub>1</sub> .RA	12/10.0 JR cc.RA	6.5 LD r <sub>1</sub> .IM	12/10.0 JP cc.DA	6.5 INC r1		
	1	6.5 RLC R <sub>1</sub>	6.5 RLC IR <sub>1</sub>	6.5 ADC r <sub>1</sub> .r <sub>2</sub>	6.5 ADC r <sub>1</sub> .r <sub>2</sub>	10.5 ADC R <sub>2</sub> .R <sub>1</sub>	10.5 ADC IR <sub>2</sub> .R <sub>1</sub>	10.5 ADC R <sub>1</sub> .IM	10.5 ADC IR <sub>1</sub> .IM									
	2	6.5 INC R <sub>1</sub>	6.5 INC IR <sub>1</sub>	6.5 SUB r <sub>1</sub> .r <sub>2</sub>	6.5 SUB r <sub>1</sub> .r <sub>2</sub>	10.5 SUB R <sub>2</sub> .R <sub>1</sub>	10.5 SUB IR <sub>2</sub> .R <sub>1</sub>	10.5 SUB R <sub>1</sub> .IM	10.5 SUB IR <sub>1</sub> .IM									
	3	8.0 JP IRR <sub>1</sub>	6.1 SRP IM	6.5 SBC r <sub>1</sub> .r <sub>2</sub>	6.5 SBC r <sub>1</sub> .r <sub>2</sub>	10.5 SBC R <sub>2</sub> .R <sub>1</sub>	10.5 SBC IR <sub>2</sub> .R <sub>1</sub>	10.5 SBC R <sub>1</sub> .IM	10.5 SBC IR <sub>1</sub> .IM									
	4	8.5 DA R <sub>1</sub>	8.5 DA IR <sub>1</sub>	6.5 OR r <sub>1</sub> .r <sub>2</sub>	6.5 OR r <sub>1</sub> .r <sub>2</sub>	10.5 OR R <sub>2</sub> .R <sub>1</sub>	10.5 OR IR <sub>2</sub> .R <sub>1</sub>	10.5 OR R <sub>1</sub> .IM	10.5 OR IR <sub>1</sub> .IM									
	5	10.5 POP R <sub>1</sub>	10.5 POP IR <sub>1</sub>	6.5 AND r <sub>1</sub> .r <sub>2</sub>	6.5 AND r <sub>1</sub> .r <sub>2</sub>	10.5 AND R <sub>2</sub> .R <sub>1</sub>	10.5 AND IR <sub>2</sub> .R <sub>1</sub>	10.5 AND R <sub>1</sub> .IM	10.5 AND IR <sub>1</sub> .IM									
	6	6.5 COM R <sub>1</sub>	6.5 COM IR <sub>1</sub>	6.5 TCM r <sub>1</sub> .r <sub>2</sub>	6.5 TCM r <sub>1</sub> .r <sub>2</sub>	10.5 TCM R <sub>2</sub> .R <sub>1</sub>	10.5 TCM IR <sub>2</sub> .R <sub>1</sub>	10.5 TCM R <sub>1</sub> .IM	10.5 TCM IR <sub>1</sub> .IM									6.0 STOP
	7	10/12.1 PUSH R <sub>2</sub>	12/14.1 PUSH IR <sub>2</sub>	6.5 TM r <sub>1</sub> .r <sub>2</sub>	6.5 TM r <sub>1</sub> .r <sub>2</sub>	10.5 TM R <sub>2</sub> .R <sub>1</sub>	10.5 TM IR <sub>2</sub> .R <sub>1</sub>	10.5 TM R <sub>1</sub> .IM	10.5 TM IR <sub>1</sub> .IM									7.0 HALT
	8	10.5 DECW RR <sub>1</sub>	10.5 DECW IR <sub>1</sub>	12.0 LDE r <sub>1</sub> .rr <sub>2</sub>	18.0 LDEI rr <sub>1</sub> .rr <sub>2</sub>													6.1 DI
	9	6.5 RL R <sub>1</sub>	6.5 RL IR <sub>1</sub>	12.0 LDE r <sub>2</sub> .rr <sub>1</sub>	18.0 LDEI rr <sub>2</sub> .rr <sub>1</sub>													6.1 EI
	A	10.5 INCW RR <sub>1</sub>	10.5 INCW IR <sub>1</sub>	6.5 CP r <sub>1</sub> .r <sub>2</sub>	6.5 CP r <sub>1</sub> .r <sub>2</sub>	10.5 CP R <sub>2</sub> .R <sub>1</sub>	10.5 CP IR <sub>2</sub> .R <sub>1</sub>	10.5 CP R <sub>1</sub> .IM	10.5 CP IR <sub>1</sub> .IM									14.0 RET
	B	6.5 CLR R <sub>1</sub>	6.5 CLR IR <sub>1</sub>	6.5 XOR r <sub>1</sub> .r <sub>2</sub>	6.5 XOR r <sub>1</sub> .r <sub>2</sub>	10.5 XOR R <sub>2</sub> .R <sub>1</sub>	10.5 XOR IR <sub>2</sub> .R <sub>1</sub>	10.5 XOR R <sub>1</sub> .IM	10.5 XOR IR <sub>1</sub> .IM									16.0 IRET
	C	6.5 RRC R <sub>1</sub>	6.5 RRC IR <sub>1</sub>	12.0 LDC r <sub>1</sub> .rr <sub>2</sub>	18.0 LDCl rr <sub>1</sub> .rr <sub>2</sub>				10.5 LD r <sub>1</sub> .x.R <sub>2</sub>									6.5 RCF
	D	6.5 SRA R <sub>1</sub>	6.5 SRA IR <sub>1</sub>	12.0 LDC r <sub>2</sub> .rr <sub>1</sub>	18.0 LDCl rr <sub>2</sub> .rr <sub>1</sub>	20.0 CALL* IRR <sub>1</sub>		20.0 CALL DA	10.5 LD r <sub>2</sub> .x.R <sub>1</sub>									6.5 SCF
	E	6.5 RR R <sub>1</sub>	6.5 RR IR <sub>1</sub>		6.5 LD r <sub>1</sub> .R <sub>2</sub>	10.5 LD R <sub>2</sub> .R <sub>1</sub>	10.5 LD IR <sub>2</sub> .R <sub>1</sub>	10.5 LD R <sub>1</sub> .IM	10.5 LD IR <sub>1</sub> .IM									6.5 CCF
	F	8.5 SWAP R <sub>1</sub>	8.5 SWAP IR <sub>1</sub>		6.5 LD r <sub>1</sub> .r <sub>2</sub>			10.5 LD R <sub>2</sub> .R <sub>1</sub>										6.0 NOP



**Legend:**  
R = 8-bit address  
r = 4-bit address  
R<sub>1</sub> or r<sub>1</sub> = Dst address  
R<sub>2</sub> or r<sub>2</sub> = Src address

**Sequence:**  
Opcode, First Operand, Second Operand

**NOTE:** The blank areas are not defined

\*2 byte instruction fetch cycle appears as a 3 byte instruction

## ABSOLUTE MAXIMUM RATINGS

Voltages on all pins with respect to GND ..... - 0.3V to + 7.0V  
 Operating Ambient Temperature ..... See Ordering Information  
 Storage Temperature ..... - 65°C to + 150°C

Stresses greater than those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; operation of the device at any condition above those indicated in the operational sections of these specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

## STANDARD TEST CONDITIONS

The DC characteristics listed below apply for the following standard test conditions, unless otherwise noted. All voltages are referenced to GND. Positive current flows into the referenced pin.

Standard conditions are as follows:

- $+4.5 \leq V_{CC} \leq +5.5V$
- $GND = 0V$
- $0 \leq T_A \leq +70 \text{ C for S (Standard temperature)}$

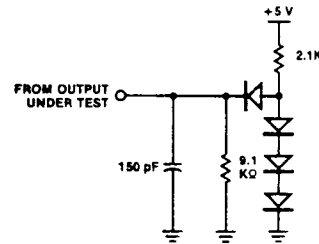


Figure 12. Test Load 1

## DC CHARACTERISTICS

Symbol	Parameter	Min	Typ	Max	Unit	Condition
$V_{CH}$	Clock Input High Voltage	3.8		$V_{CC}$	V	Driven by External Clock Generator
$V_{CL}$	Clock Input Low Voltage	-0.3		0.8	V	Driven by External Clock Generator
$V_{IH}$	Input High Voltage	2.0		$V_{CC}$	V	
$V_{IL}$	Input Low Voltage	-0.3		0.8	V	
$V_{RH}$	Reset Input High Voltage	3.8		$V_{CC}$	V	
$V_{RL}$	Reset Input Low Voltage	-0.3		0.8	V	
$V_{OH}$	Output High Voltage	2.4			V	$I_{OH} = -250 \mu A$
$V_{OH}$	Output High Voltage	$V_{CC} - 100mV$			V	$I_{CC} = -100 \mu A$
$V_{OL}$	Output Low Voltage			0.4	V	$I_{OL} = +2.0 mA$
$I_{IL}$	Input Leakage	-10		10	$\mu A$	$0V \leq V_{IN} \leq +5.25V$
$I_{OL}$	Output Leakage	-10		10	$\mu A$	$0V \leq V_{IN} \leq +5.25V$
$I_{IR}$	Reset Input Current			-50	$\mu A$	$V_{CC} = +5.25V, V_{RL} = 0V$
$I_{CC}$	Supply Current				mA	All outputs and I/O pins floating, 12 MHz
$I_{CC1}$	Standby Current		5		mA	Halt Mode
$I_{CC2}$	Standby Current			10	$\mu A$	Stop Mode

$I_{CC2}$  requires loading TMR (%F1) with any value prior to STOP execution.

Use the sequence:

```
LD TMR, #00
NOP
STOP
```

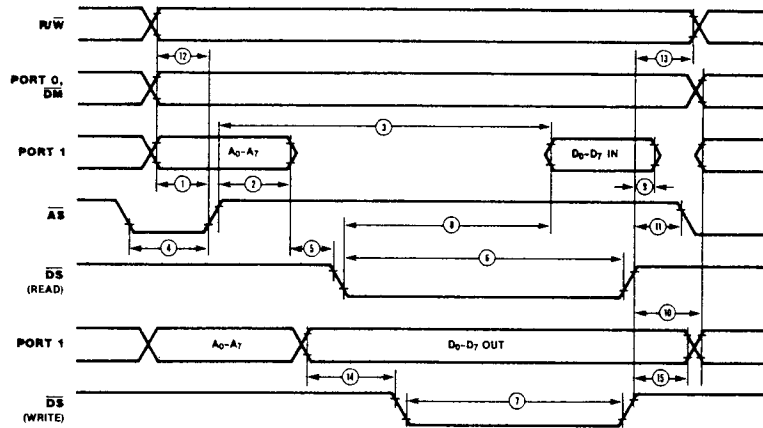


Figure 13. External I/O or Memory Read/Write

## AC CHARACTERISTICS

External I/O or Memory Read and Write Timing

Number	Symbol	Parameter	12MHz		16MHz		20MHz		Units	Notes
			Min	Max	Min	Max	Min	Max		
1	TdA(AS)	Address Valid to $\overline{AS}$ ↑ Delay	35		25		20		ns	2,3,4
2	TdAS(A)	$\overline{AS}$ ↑ to Address Float Delay	45		35		25		ns	2,3,4
3	TdAS(DR)	$\overline{AS}$ ↑ to Read Data Req'd Valid		250		180		150	ns	1,2,3,4
4	TwAS	$\overline{AS}$ Low Width	55		40		30		ns	2,3,4
5	TdAZ(DS)	Address Float to $\overline{DS}$ ↓	0		0		0		ns	
6	TwDSR	$\overline{DS}$ (Read) Low Width	185		135		105		ns	1,2,3,4
7	TwDSW	$\overline{DS}$ (Write) Low Width	110		80		65		ns	1,2,3,4
8	TdDSR(DR)	$\overline{DS}$ ↓ to Read Data Req'd Valid		130		75		55	ns	1,2,3,4
9	ThDR(DS)	Read Data to $\overline{DS}$ ↑ Hold Time	0		0		0		ns	2,3,4
10	TdDS(A)	$\overline{DS}$ ↑ to Address Active Delay	65		50		40		ns	2,3,4
11	TdDS(AS)	$\overline{DS}$ ↑ to $\overline{AS}$ ↓ Delay	45		35		25		ns	2,3,4
12	TdR/W(AS)	R/ $\overline{W}$ Valid to $\overline{AS}$ ↑ Delay	33		25		20		ns	2,3,4
13	TdDS(R/W)	$\overline{DS}$ ↑ to R/ $\overline{W}$ Not Valid	50		35		25		ns	2,3,4
14	TdDW(DSW)	Write Data Valid to $\overline{DS}$ ↓ (Write) Delay	35		25		20		ns	2,3,4
15	TdDS(DW)	$\overline{DS}$ ↑ to Write Data Not Valid Delay	55		35		25		ns	2,3,4
16	TdA(DR)	Address Valid to Read Data Req'd Valid		310		230		180	ns	1,2,3,4
17	TdAS(DS)	$\overline{AS}$ ↑ to $\overline{DS}$ ↓ Delay	65		45		35		ns	2,3,4
18	TdDI(DS)	Data Input Setup to $\overline{DS}$ ↑	75		60		50		ns	1,2,3,4
19	TdDM(AS)	$\overline{DM}$ Valid to $\overline{AS}$ ↓ Delay	50		30		20		ns	2,3,4

### Notes

1. When using extended memory timing add 2TpC
2. Timing numbers given are for minimum TpC
3. See clock cycle dependent characteristics table
4. 20 MHz timing is preliminary and subject to change

+ Test Load 1

\* All timing references use 2.0V for a logic "1" and 0.8V for a logic "0"

## AC CHARACTERISTICS

### Additional Timing Table

Number	Symbol	Parameter	12 MHz		16 MHz		20 MHz		Notes
			Min	Max	Min	Max	Min	Max	
1	TpC	Input Clock Period	83	1000	62.5	1000	50	1000	1
2	TrC, TtC	Clock Input Rise & Fall Times		15		10		10	1
3	TwC	Input Clock Width	37		21		15		1
4	TwTinL	Timer Input Low Width	75		75		75		2
5	TwTinH	Timer Input High Width	3TpC		3TpC		3TpC		2
6	TpTin	Timer Input Period	8TpC		8TpC		8TpC		2
7	TrTin, TtTin	Timer Input Rise and Fall Times	100		100		100		2
8A	TwIL	Interrupt Request Input Low Time	70		70		70		2,4
8B	TwIL	Interrupt Request Input Low Time	3TpC		3TpC		3TpC		2,5
9	TwIH	Interrupt Request Input High Time	3TpC		3TpC		3TpC		2,3

#### Notes:

1. Clock timing references use 3.8 V for a logic "1" and 0.8 V for a logic "0"
  2. Timing references use 2.0 V for a logic "1" and 0.8 V for a logic "0"
  3. Interrupt references request via Port 3
  4. Interrupt request via Port 3 (P3<sub>1</sub>, P3<sub>2</sub>)
  5. Interrupt request via P30
  6. 20 MHz timing is preliminary and subject to change.
- Units in nanoseconds (ns)

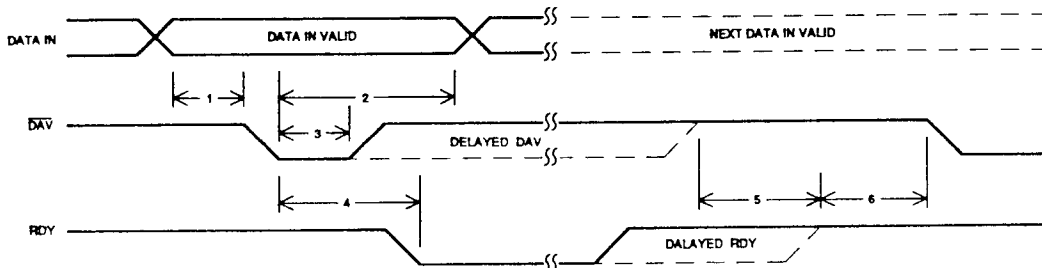


Figure 15a. Input Handshake Timing

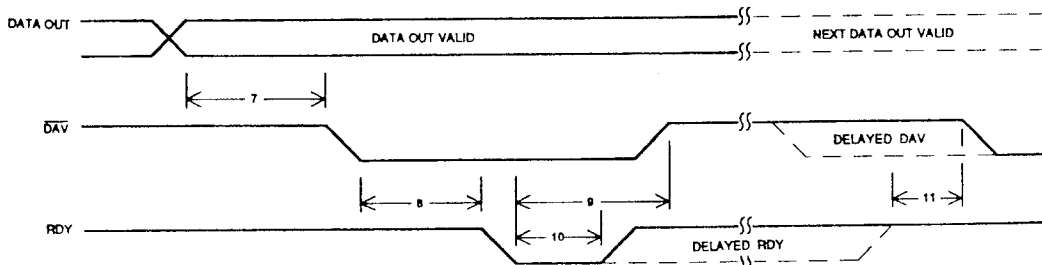
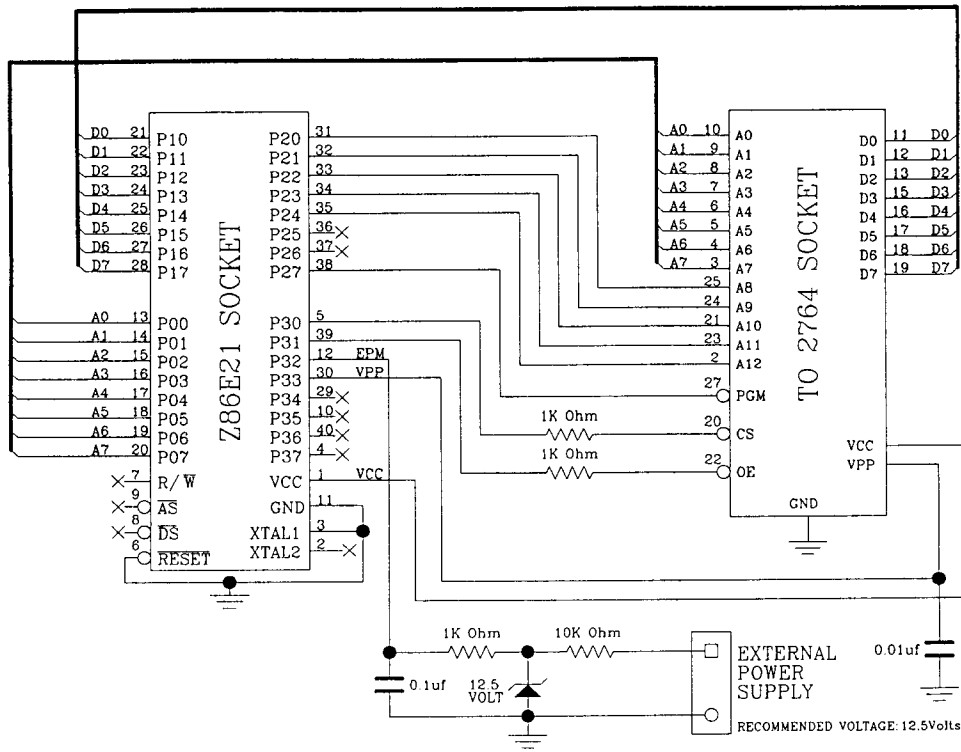


Figure 15b. Output Handshake Timing



Z86E21 Z8 OTP Programming Adapter

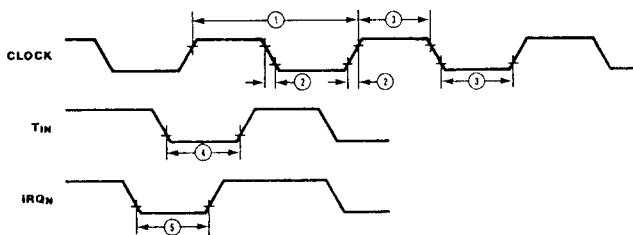


Figure 14. Additional Timing

## AC CHARACTERISTICS

### Handshake Timing

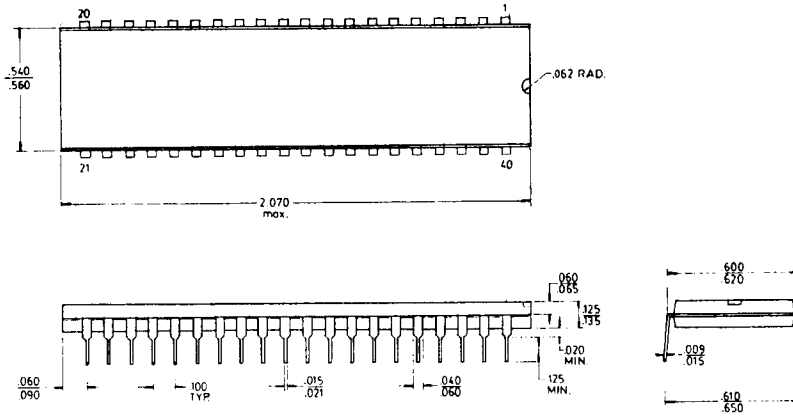
Number	Symbol	Parameter	12,16,20 MHz		Notes (Data Direction)
			Min	Max	
1	TsDI(DAV)	Data In Setup Time	0		In
2	ThDI(DAV)	Data In Hold Time	145		In
3	TwDAV	Data Available Width	110		In
4	TdDAV(RDY)	DAV↓to RDY↓Delay		115	In
5	TdDAV(RDY)	DAV↑to RDY↑Delay		115	In
6	TdRDY(DAV)	RDY↑to DAV↓Delay	0		In
7	TdDO(DAV)	Data Out to DAV↓Delay	TpC		Out
8	TdDAVd(RDY)	DAV↓to RDY↓Delay	0		Out
9	TdRDY(DAV)	RDY↓to DAV↑Delay		115	Out
10	TwRDY	RDY Width	110		Out
11	TdRDY(DAV)	RDY↑to DAV↓Delay		115	Out

## CLOCK DEPENDENT AC CHARACTERISTICS

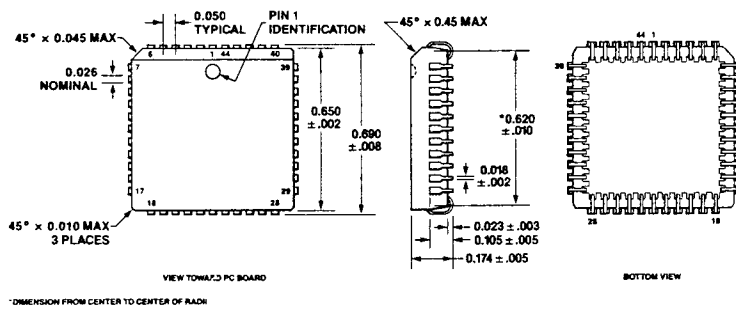
### External I/O or Memory Read and Write Timing

Number	Symbol	Equation
1	TdA(AS)	$0.4T_{pC} + 0.32$
2	TdAS(A)	$0.59T_{pC} - 3.25$
3	TdAS(DR)	$2.83T_{pC} + 6.14$
4	TwAS	$0.66T_{pC} - 1.65$
6	TwDSR	$2.33T_{pC} - 10.56$
7	TwDSW	$1.27T_{pC} + 1.67$
8	TdDSR(DR)	$1.97T_{pC} - 42.5$
10	TdDS(A)	$0.8T_{pC}$
11	TdDS(AS)	$0.59T_{pC} - 3.14$
12	TdR/W(AS)	$0.4T_{pC}$
13	TdDS(R/W)	$0.8T_{pC} - 15$
14	TdDW(DSW)	$0.4T_{pC}$
15	TdDS(DW)	$0.88T_{pC} - 19$
16	TdA(DR)	$4T_{pC} - 20$
17	TdAS(DS)	$0.91T_{pC} - 10.7$
18	TsDI(DS)	$0.8T_{pC} - 10$
19	TdDM(AS)	$0.9T_{pC} - 26.3$

**PACKAGING INFORMATION**



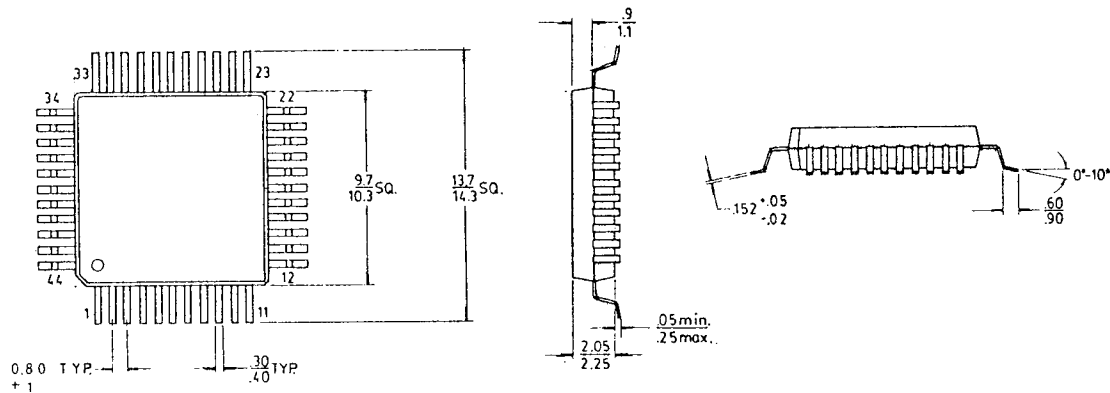
**40-Pin Dual-in-Line Package (DIP),  
Plastic**



**View Toward PC Board**

**Bottom View**

**44-Pin Plastic Chip Carrier (PCC)**



**44-Pin Quad Flat Pack (QFP)**

NOTE: Package dimensions are given in inches. To convert to millimeters, multiply by 25.4.

## ORDERING INFORMATION

### Z86C21

12 MHz	16 MHz	20 MHz
Z86C2112PSC	Z86C2116PSC	Z86C2120PSC
Z86C2112PEC	Z86C2116PEC	Z86C2120PEC
Z86C2112CEA	Z86C2116CEA	Z86C2120CEA
Z86C2112FSC	Z86C2116FSC	Z86C2120FSC
Z86C2112FEC	Z86C2116FEC	Z86C2120FEC
Z86C2112VSC	Z86C2116VSC	Z86C2120VSC
Z86C2112VEC	Z86C2116VEC	Z86C2120VEC

### Z86E21

12MHz	16MHz
Z86E2112PSC	Z86E2116PSC
Z86E2112VSC	Z86E2116VSC

### Codes

#### PACKAGE

Preferred  
 D = Cerdip  
 P = Plastic  
 V = Plastic Chip Carrier

#### Longer Lead Time

C = Ceramic  
 F = Plastic Quad Flat Pack  
 G = Ceramic PGA (Pin Grid Array)  
 L = Ceramic LCC  
 Q = Ceramic Quad-in-Line  
 R = Protopack  
 T = Low Profile Protopack

#### TEMPERATURE

Preferred  
 S = 0°C to +70°C

#### Longer Lead Time

E = -40°C to +85°C  
 M = -55°C to +125°C

#### ENVIRONMENTAL

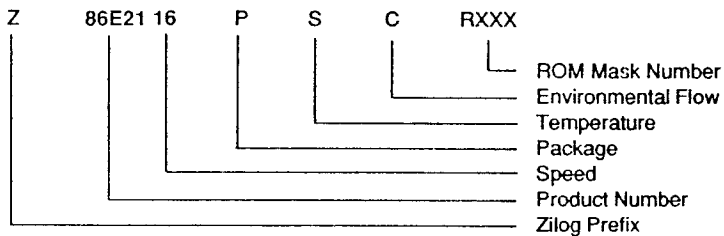
Preferred  
 C = Plastic Standard  
 E = Hermetic Standard  
 F = Protopack Standard

#### Longer Lead Time

A = Hermetic Stressed  
 B = 833 Class B Military  
 D = Plastic Stressed  
 J = JAN 38510 Military

### Example:

Z86E2116PSC is a CMOS 86E21, 16 MHz, Plastic DIP, 0°C to +70°C, Plastic Standard Flow.





---

**ZILOG DOMESTIC SALES OFFICES AND  
TECHNICAL CENTERS****CALIFORNIA**

Agoura ..... 818-707-2160  
Campbell ..... 408-370-8120  
Tustin ..... 714-838-7800

**COLORADO**

Boulder ..... 303-494-2905

**FLORIDA**

Largo ..... 813-585-2533

**GEORGIA**

Norcross ..... 404-448-9370

**ILLINOIS**

Schaumburg ..... 312-517-8080

**NEW HAMPSHIRE**

Nashua ..... 603-888-8590

**MINNESOTA**

Edina ..... 612-831-7611

**NEW JERSEY**

Clark ..... 201-382-5700

**OHIO**

Seven Hills ..... 216-447-1480

**PENNSYLVANIA**

Ambler ..... 215-653-0230

**TEXAS**

Dallas ..... 214-987-9987

**WASHINGTON**

Seattle ..... 206-523-3591

**INTERNATIONAL SALES OFFICES****CANADA**

Toronto ..... 416-673-0634

**GERMANY**

Munich ..... 49-89-672-045

**JAPAN**

Tokyo ..... 81-3-587-0528

**HONG KONG**

Kowloon ..... 852-3-723-8979

**KOREA**

Seoul ..... 82-2-552-5401

**SINGAPORE**

Singapore ..... 65-235-7155

**TAIWAN**

Taipei ..... 886-2-741-3125

**UNITED KINGDOM**

Maidenhead ..... 44-628-392-00

© 1989 by Zilog, Inc. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of Zilog.

The information contained herein is subject to change without notice. Zilog will not be responsible for any such changes. Zilog will not be responsible for notifying any user of changes. Zilog assumes no responsibility for the use of any circuitry or other technology embodied in a Zilog product. No patent licenses, industrial property rights, or other rights are implied.

Zilog will not be responsible for any damage to the user that may result from accidents or any other reasons during operations of the products described herein.

All specifications (parameters) are subject to change without notice. Zilog will not be responsible for any such changes. Zilog will not be responsible for notifying any user of changes. The applicable Zilog test documentation will specify which parameters are tested.

**Zilog, Inc. 210 Hacienda Ave., Campbell, CA 95008-6609**  
**Telephone (408) 370-8000 TWX 910-338-7621**