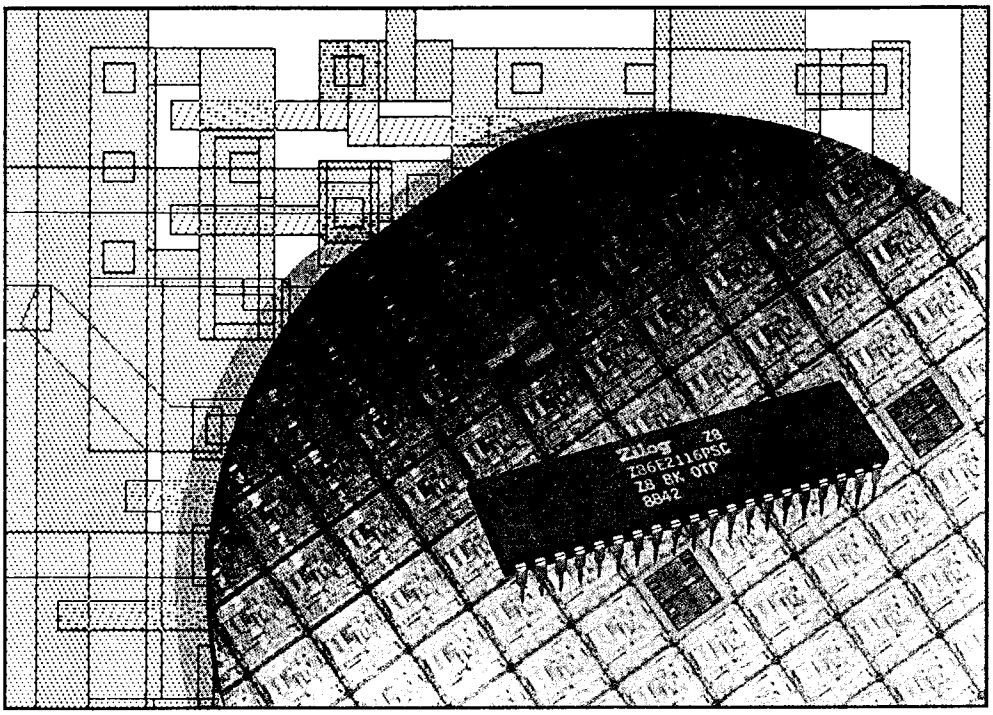
PRELIMINARY PRODUCT SPECIFICATION

zilog

Z86C21/E21

CMOS Z8® 8K ROM MCU

NOVEMBER 1989



zilog

ADVANCED INFORMATION
Product Specification

Z86C21/Z86E21 CMOS
CMOS Z8® 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 μs, average of 1.0 μs.**
- 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 μs.**
- 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.

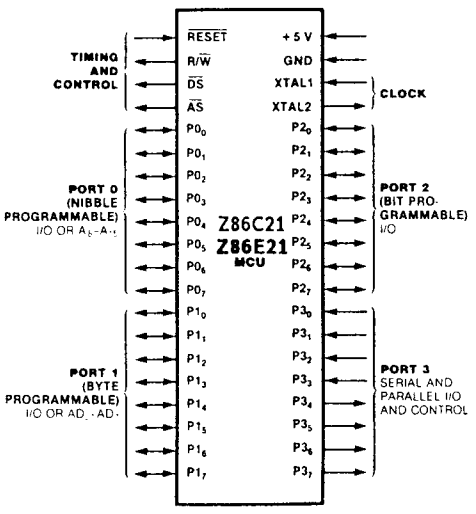


Figure 1. Pin Functions

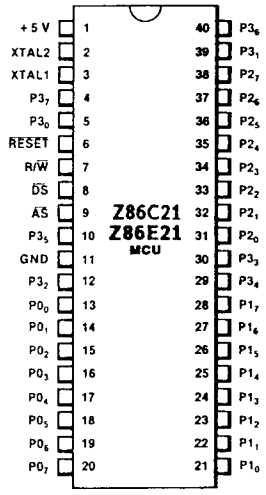


Figure 2. 40-pin Dual-In-Line Package (DIP), Pin Assignments

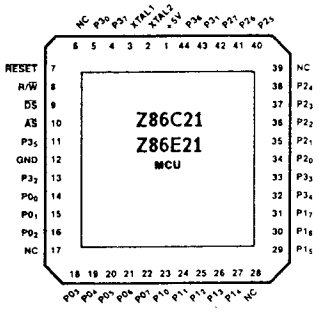
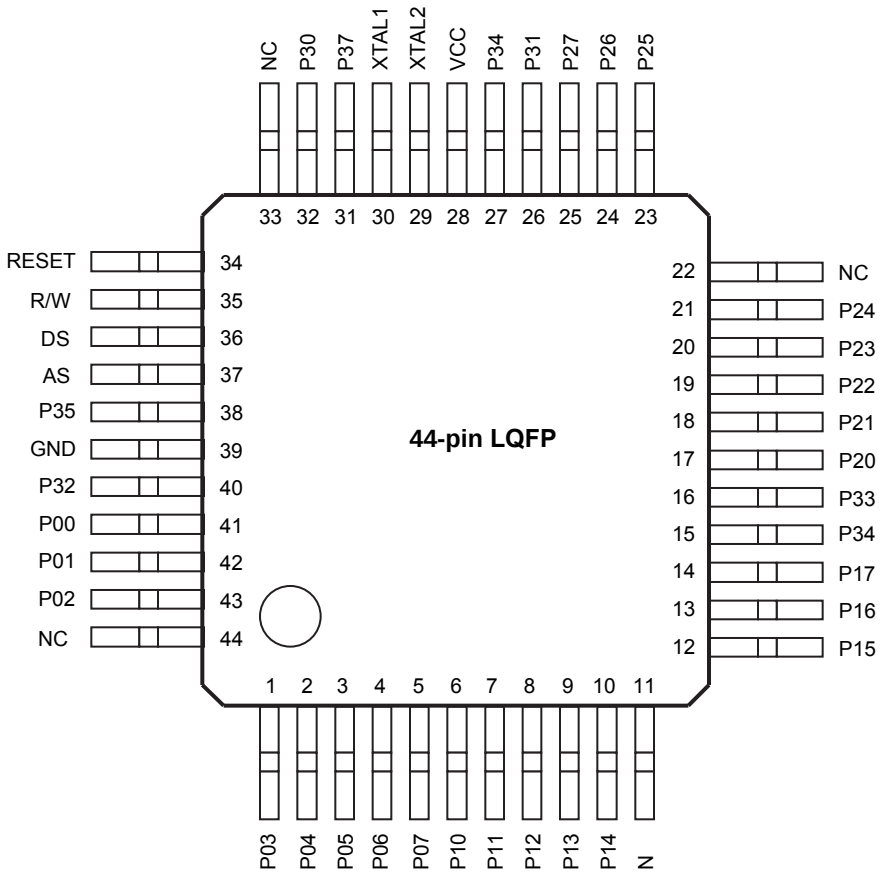


Figure 2b. 44-pin Chip Carrier, Pin Assignments



44-Pin LQFP 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.

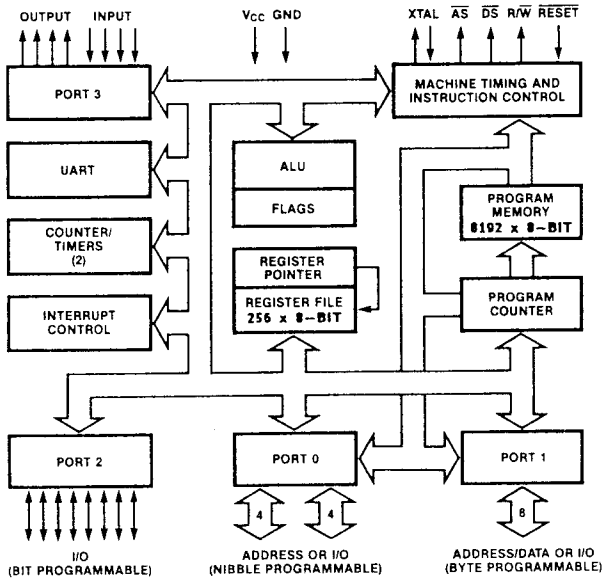


Figure 3. Functional Block Diagram

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:

$NOP(FF_H) + STOP(6F_H)$
$NOP(FF_H) + HALT(7F_H)$

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.

$P0_0$-$P0_7$, $P1_0$-$P1_7$, $P2_0$-$P2_7$, $P3_0$-$P3_7$. *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 $000C_H$.

R/$\overline{W}$. *Read/Write* (output). 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 $P3_4$, 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).

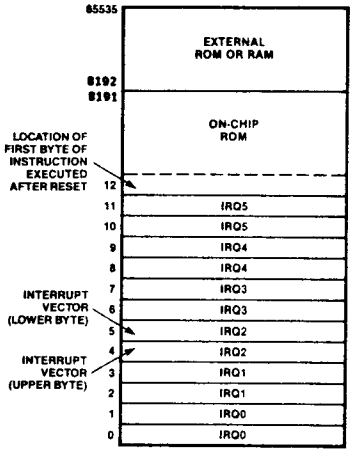


Figure 4. Program Memory Map

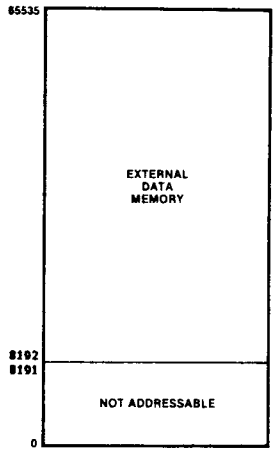


Figure 5. Data Memory Map

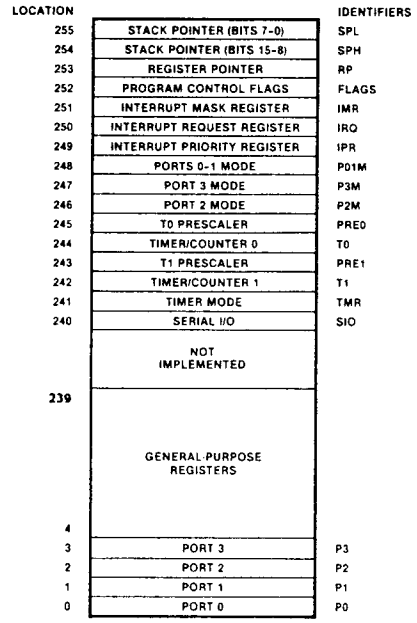


Figure 6. The Register File

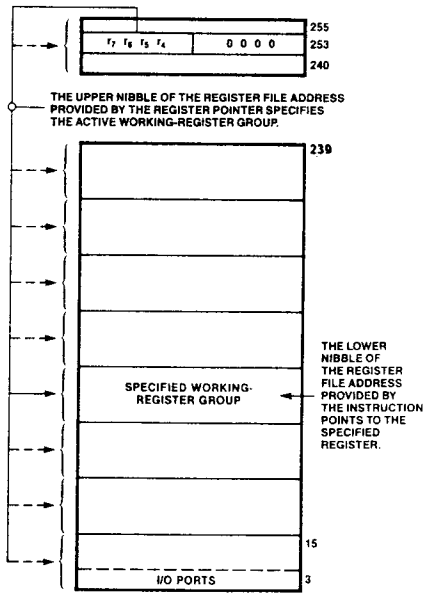


Figure 7. The Register Pointer

SERIAL INPUT/OUTPUT

Port 3 lines $P3_0$ and $P3_7$ 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_4) 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_3 interrupt request.

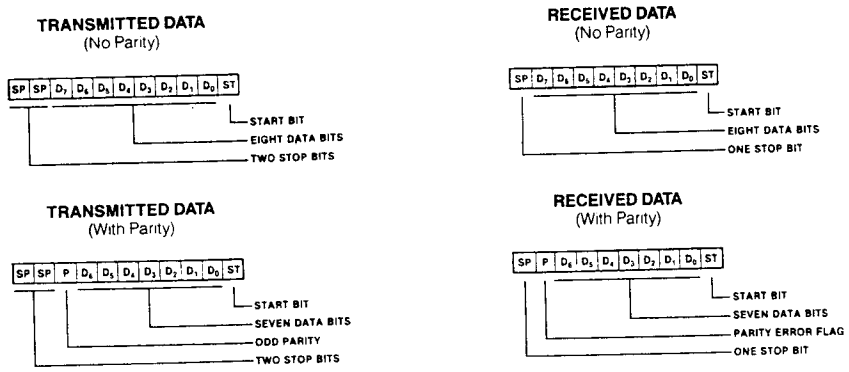


Figure 8. Serial Data Formats

COUNTER/TIMERS

The **Z86C21** contains two 8-bit programmable counter/timers (T_0 and T_1), each driven by its own 6-bit programmable prescaler. The T_1 prescaler can be driven by internal or external clock sources; however, the T_0 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_4 (T_0) or IRQ_5 (T_1)—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_1 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_0 output to the input of T_1. Port 3 line $P3_6$ also serves as a timer output ($\overline{T}_{OUT}$) through which T_0, T_1 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 $P3_3$ and $P3_4$ are used as the handshake controls RDY_1 and $\overline{DAV}_1$ (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 1 can be placed in the high-impedance state along with Port 0, $\overline{AS}$, $\overline{DS}$ and R/$\overline{W}$, allowing the **Z86C21** to share common resources in multiprocessor and DMA applications. Data transfers can be controlled by assigning $P3_3$ as a Bus Acknowledge input, and $P3_4$ as a Bus Request output.

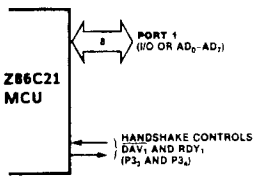


Figure 9a. Port 1

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 $P3_2$ and $P3_5$ are used as the handshake controls $\overline{DAV}_0$ and RDY_0. Handshake signal assignment is dictated by the I/O direction of the upper nibble $P0_4$-$P0_7$.

For external memory references, Port 0 can provide address bits A_8-A_{11} (lower nibble) or A_8-A_{15} (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 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 $\overline{AS}$, $\overline{DS}$ and R/$\overline{W}$.

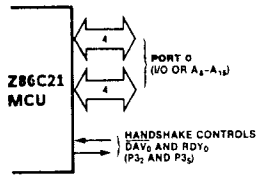


Figure 9b. Port 0

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 $P3_1$ and $P3_6$ are used as the handshake controls lines $\overline{DAV}_2$ and RDY_2. The handshake signal assignment for Port 3 lines $P3_1$ and $P3_6$ is dictated by the direction (input or output) assigned to bit 7 of Port 2.

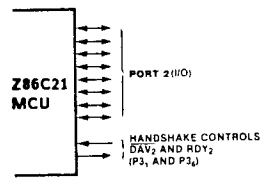


Figure 9c. Port 2

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 ($P3_0$-$P3_3$) and four output ($P3_4$-$P3_7$). For serial I/O, lines $P3_0$ and $P3_7$ 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 ($\overline{DAV}$ and RDY); four external interrupt request signals (IRQ_0-IRQ_3); timer input and output signals (T_{IN} and T_{OUT}) and Data Memory Select ($\overline{DM}$).

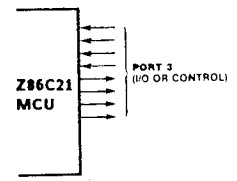


Figure 9d. Port 3

INTERRUPTS

The **Z86C21** allows six different interrupts from eight sources: the four Port 3 lines $P3_0$-$P3_3$, 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 appli-cations where the final system configura-tion -- 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 (Outpus 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 SIZEn pins.

/SCLK (Output, TTL compatible) System Clock. This line is teh 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 Acknow-ledge. This output, when low, indicates that the Z86C12 is an interrupt cycle.

/SIZE0, /SIZE1, /SIZE2, /SIZE3, SIZE4 (Inputs, TTL compatible). The /SIZEn 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:

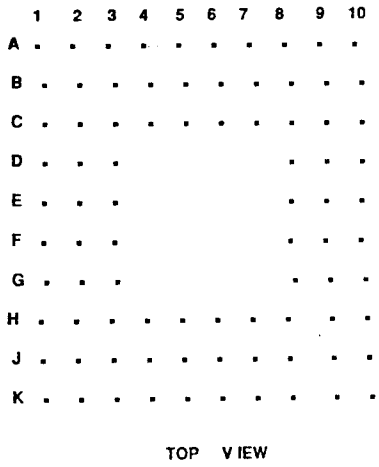
The SIZE pins may be configured to make the memory control signals (/MAS, /MDS, R/W, /AS, and /DS) look like the Z86C91 ROMless device, however on power-up or reset ports 0 and 1 are configured as inputs, rather than A15 - A8 and AD7 - AD0, respectively.

Table 1. Z86C12 Pin Assignments

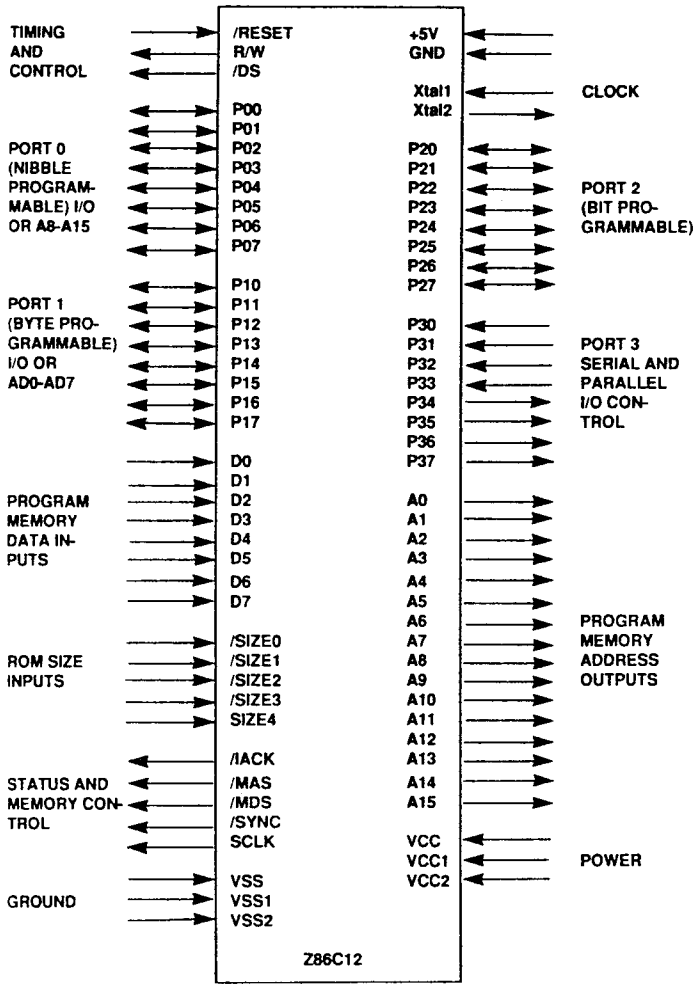
NAME		NAME	PIN	NAME	PIN	NAME	PIN
/AS	B2	A8	J5	P07	J1	P36	A7
/DS	C4	A9	K4	P10	G8	P37	A5
/MAS	E1	D0	H3	P11	G9	R/W	A1
/MDS	G3	D1	K2	P12	G10	SCLK	G2
/RESET	B3	D2	J3	P13	F8	SIZE4	F10
/SIZE0	A3	D3	K3	P14	D10	VCC	A4
/SIZE1	C5	D4	H8	P15	C10	VCC1	B6
/SIZE2	A6	D5	J10	P16	B10	VCC2	F9
/SIZE3	C6	D6	H9	P17	E9	VSS	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



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.

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

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

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

Assignment of a value is indicated by the symbol "←". For example,

dst ← dst + 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,

dst (7)

refers to bit 7 of the destination operand.

Flags. Control Register R252 contains the following six flags:

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

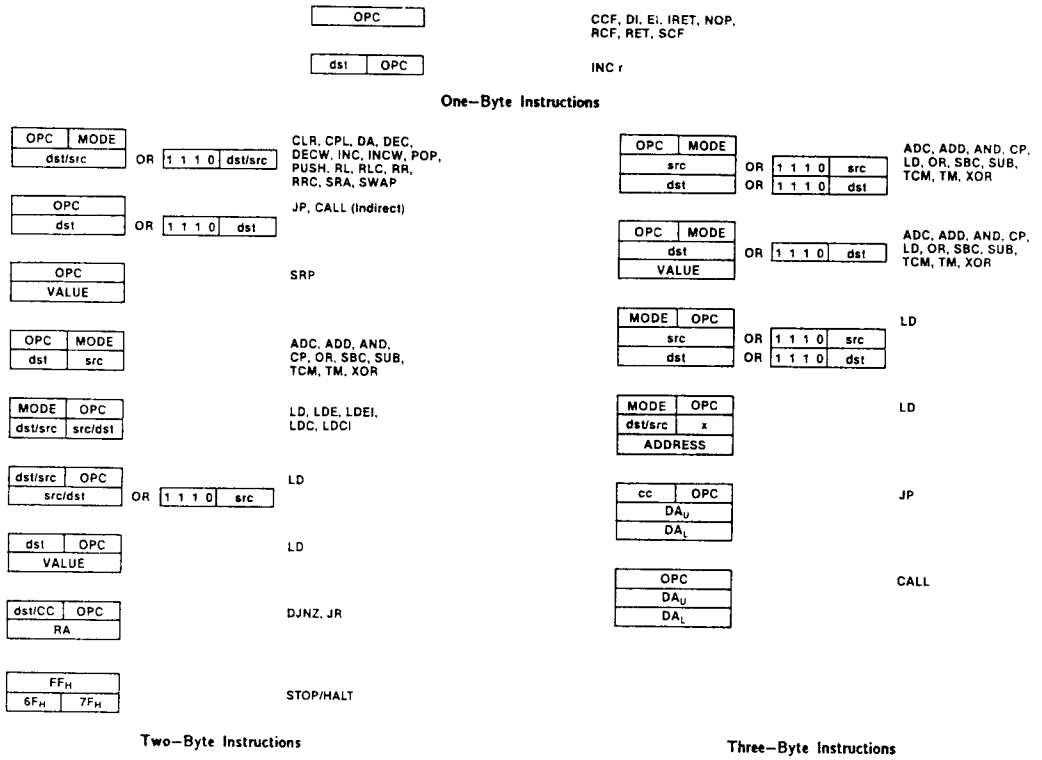
Affected flags are indicated by:

0	Cleared to zero
1	Set to one
*	Set or cleared according to operation
—	Unaffected
X	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

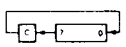
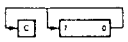
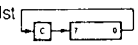
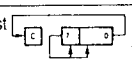
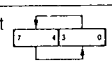


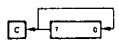
INSTRUCTION SUMMARY

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

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

INSTRUCTION SUMMARY (Continued)

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

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

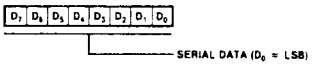
NOTE: These instructions have an identical set of addressing modes, which are encoded for brevity. The first opcode nibble is found in the instruction set table above. The second nibble is expressed symbolically by a ☐ in this table, and its value is found in the following table to the left of the applicable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

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

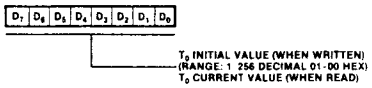
Addr Mode dst	Addr Mode src	Lower Opcode Nibble
r	r	2
r	Ir	3
R	R	4
R	IR	5
R	IM	6
IR	IM	7

REGISTERS

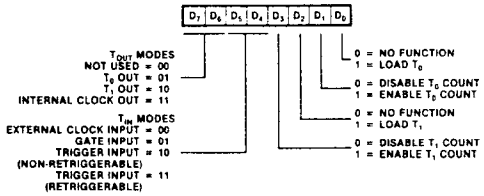
R240 SIO
SERIAL I/O REGISTER
($F0_H$; Read/Write)



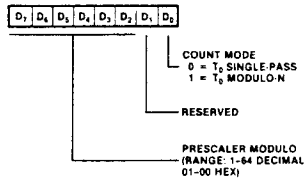
R244 T0
COUNTER/TIMER 0 REGISTER
($F4_H$; Read/Write)



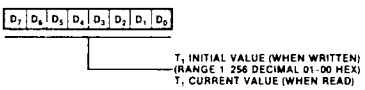
R241 TMR
TIMER MODE REGISTER
($F1_H$; Read/Write)



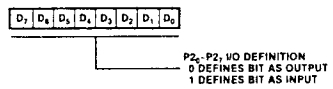
R245 PRE0
PRESCALER 0 REGISTER
($F5_H$; Write Only)



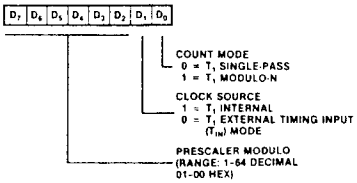
R242 T1
COUNTER TIMER 1 REGISTER
($F2_H$; Read/Write)



R246 P2M
PORT 2 MODE REGISTER
($F6_H$; Write Only)



R243 PRE1
PRESCALER 1 REGISTER
($F3_H$; Write Only)



R247 P3M
PORT 3 MODE REGISTER
($F7_H$; Write Only)

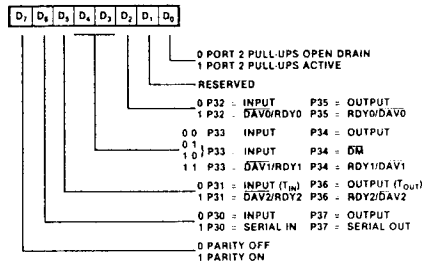
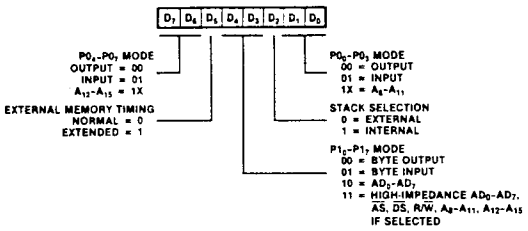
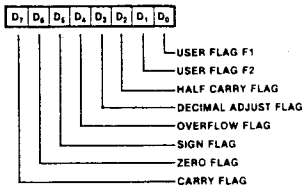


Figure 11. Control Registers

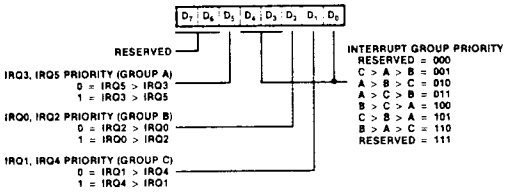
R248 P01M
PORT 0 AND 1 MODE REGISTER
($F8_H$; Write Only)



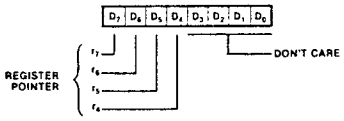
R252 FLAGS
FLAG REGISTER
(FC_H; Read/Write)



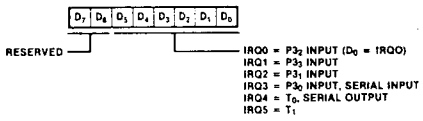
R249 IPR
INTERRUPT PRIORITY REGISTER
($F9_H$; Write Only)



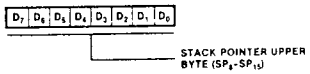
R253 RP
REGISTER POINTER
(FD_H; Read/Write)



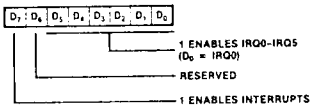
R250 IRQ
INTERRUPT REQUEST REGISTER
(FA_H; Read/Write)



R254 SPH
STACK POINTER
(FE_H; Read/Write)



R251 IMR
INTERRUPT MASK REGISTER
(FB_H; Read/Write)



R255 SPL
STACK POINTER
(FF_H; Read/Write)

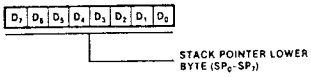
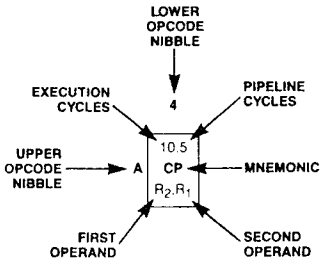


Figure 11. Control Registers (Continued)

OPCODE MAP

Upper Nibble (Hex) \ Lower Nibble (Hex)	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	6.5 **DEC** R_1	6.5 **DEC** IR_1	6.5 **ADD** r_1,r_2	6.5 **ADD** r_1,Ir_2	10.5 **ADD** R_2,R_1	10.5 **ADD** IR_2,R_1	10.5 **ADD** R_1,IM	10.5 **ADD** IR_1,IM	6.5 **LD** r_1,R_2	6.5 **LD** r_2,R_1	12/10.5 **DJNZ** r_1,RA	12/10.0 **JR** cc,RA	6.5 **LD** r_1,IM	12/10.0 **JP** cc,DA	6.5 **INC** r1	
1	6.5 **RLC** R_1	6.5 **RLC** IR_1	6.5 **ADC** r_1,r_2	6.5 **ADC** r_1,Ir_2	10.5 **ADC** R_2,R_1	10.5 **ADC** IR_2,R_1	10.5 **ADC** R_1,IM	10.5 **ADC** IR_1,IM								
2	6.5 **INC** R_1	6.5 **INC** IR_1	6.5 **SUB** r_1,r_2	6.5 **SUB** r_1,Ir_2	10.5 **SUB** R_2,R_1	10.5 **SUB** IR_2,R_1	10.5 **SUB** R_1,IM	10.5 **SUB** IR_1,IM								
3	8.0 **JP** IRR_1	6.1 **SRP** IM	6.5 **SBC** r_1,r_2	6.5 **SBC** r_1,Ir_2	10.5 **SBC** R_2,R_1	10.5 **SBC** IR_2,R_1	10.5 **SBC** R_1,IM	10.5 **SBC** IR_1,IM								
4	8.5 **DA** R_1	8.5 **DA** IR_1	6.5 **OR** r_1,r_2	6.5 **OR** r_1,Ir_2	10.5 **OR** R_2,R_1	10.5 **OR** IR_2,R_1	10.5 **OR** R_1,IM	10.5 **OR** IR_1,IM								
5	10.5 **POP** R_1	10.5 **POP** IR_1	6.5 **AND** r_1,r_2	6.5 **AND** r_1,Ir_2	10.5 **AND** R_2,R_1	10.5 **AND** IR_2,R_1	10.5 **AND** R_1,IM	10.5 **AND** IR_1,IM								
6	6.5 **COM** R_1	6.5 **COM** IR_1	6.5 **TCM** r_1,r_2	6.5 **TCM** r_1,Ir_2	10.5 **TCM** R_2,R_1	10.5 **TCM** IR_2,R_1	10.5 **TCM** R_1,IM	10.5 **TCM** IR_1,IM								6.0 **STOP**
7	10/12.1 **PUSH** R_2	12/14.1 **PUSH** IR_2	6.5 **TM** r_1,r_2	6.5 **TM** r_1,Ir_2	10.5 **TM** R_2,R_1	10.5 **TM** IR_2,R_1	10.5 **TM** R_1,IM	10.5 **TM** IR_1,IM								7.0 **HALT**
8	10.5 **DECW** RR_1	10.5 **DECW** IR_1	12.0 **LDE** r_1,Irr_2	18.0 **LDEI** Ir_1,Irr_2												6.1 **DI**
9	6.5 **RL** R_1	6.5 **RL** IR_1	12.0 **LDE** r_2,Irr_1	18.0 **LDEI** Ir_2,Irr_1												6.1 **EI**
A	10.5 **INCW** RR_1	10.5 **INCW** IR_1	6.5 **CP** r_1,r_2	6.5 **CP** r_1,Ir_2	10.5 **CP** R_2,R_1	10.5 **CP** IR_2,R_1	10.5 **CP** R_1,IM	10.5 **CP** IR_1,IM								14.0 **RET**
B	6.5 **CLR** R_1	6.5 **CLR** IR_1	6.5 **XOR** r_1,r_2	6.5 **XOR** r_1,Ir_2	10.5 **XOR** R_2,R_1	10.5 **XOR** IR_2,R_1	10.5 **XOR** R_1,IM	10.5 **XOR** IR_1,IM								16.0 **IRET**
C	6.5 **RRC** R_1	6.5 **RRC** IR_1	12.0 **LDC** r_1,Irr_2	18.0 **LDCI** Ir_1,Irr_2				10.5 **LD** $r_1 x,R_2$								6.5 **RCF**
D	6.5 **SRA** R_1	6.5 **SRA** IR_1	12.0 **LDC** r_2,Irr_1	18.0 **LDCI** Ir_2,Irr_1	20.0 **CALL*** IRR_1		20.0 **CALL** DA	10.5 **LD** $r_2 x,R_1$								6.5 **SCF**
E	6.5 **RR** R_1	6.5 **RR** IR_1		6.5 **LD** r_1,IR_2	10.5 **LD** R_2,R_1	10.5 **LD** IR_2,R_1	10.5 **LD** R_1,IM	10.5 **LD** IR_1,IM								6.5 **CCF**
F	8.5 **SWAP** R_1	8.5 **SWAP** IR_1		6.5 **LD** Ir_1,r_2		10.5 **LD** R_2,IR_1										6.0 **NOP**
Bytes per Instruction	2				3				2					3	1	



Legend:
R = 8-bit address
r = 4-bit address
R_1 or r_1 = Dst address
R_2 or r_2 = 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 ≤ Vcc ≤ + 5.5V**
- GND = 0V
- **0 C ≤ T_A ≤ +70 C for S (Standard temperature)**

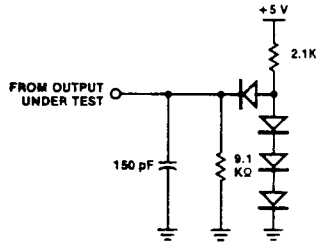


Figure 12. Test Load 1

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 µA
V_{OH}	**Output High Voltage**	**V_{CC} -100mV**			**V**	**I_{CC} = -100µA**
V_{OL}	Output Low Voltage			0.4	V	I_{OL} = +2.0 mA
I_{IL}	Input Leakage	−10		10	µA	0V ≤ V_{IN} ≤ + 5.25V
I_{OL}	Output Leakage	−10		10	µA	0V ≤ V_{IN} ≤ + 5.25V
I_{IR}	Reset Input Current			−50	µA	V_{CC} = +5.25V, V_{RL} = 0V
I_{CC}	Supply Current				mA	All outputs and I/O pins floating, 12 MHz
I_{CC_1}	Standby Current		5		mA	Halt Mode
I_{CC_2}	Standby Current			10	µA	Stop Mode

I_{CC}2 requires loading TMR (%F1) with any value prior to STOP execution.

Use the sequence:

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

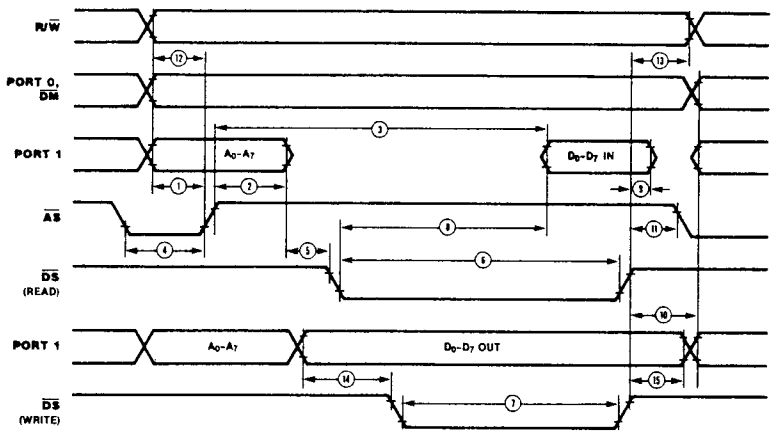


Figure 13. External I/O or Memory Read/Write

AC CHARACTERISTICS

External I/O or Memory Read and Write Timing

Number	Symbol	Parameter	12MHz Min	12MHz Max	16MHz Min	16MHz Max	20MHz Min	20MHz Max	Units	Notes
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 Min	12 MHz Max	16 MHz Min	16 MHz Max	20 MHz Min	20 MHz Max	Notes
1	TpC	Input Clock Period	83	1000	62.5	1000	50	1000	1
2	TrC,TfC	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,TfTin	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 ($P3_1$ - $P3_3$)
5. Interrupt request via P30
6. 20 MHz timing is preliminary and subject to change.
   Units in nanoseconds (ns)

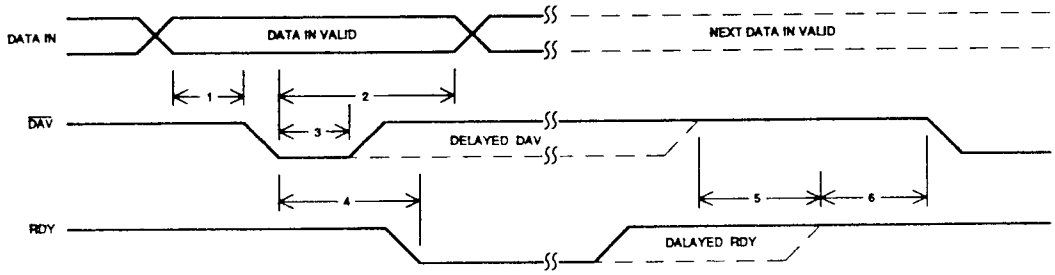


Figure 15a. Input Handshake Timing

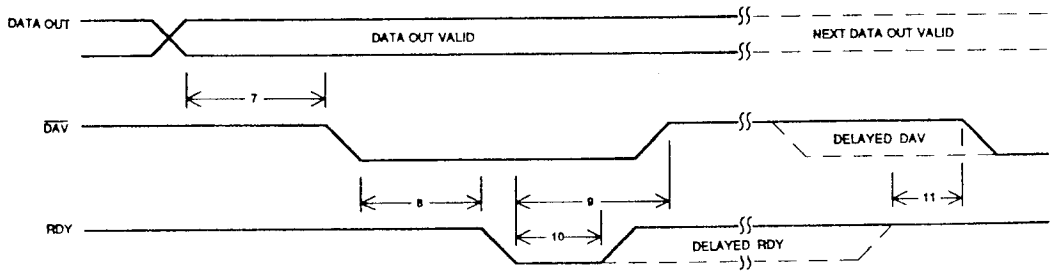
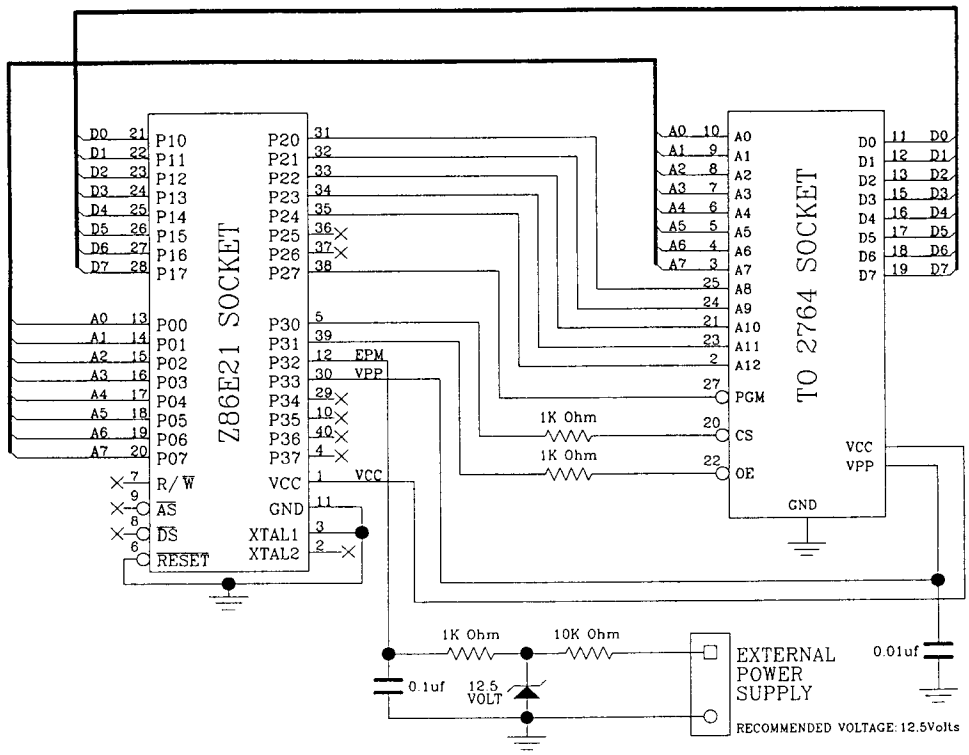


Figure 15b. Output Handshake Timing



Z86E21 Z8 OTP Programming Adapter

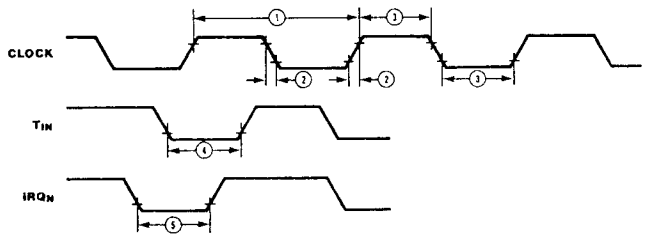


Figure 14. Additional Timing

AC CHARACTERISTICS

Handshake Timing

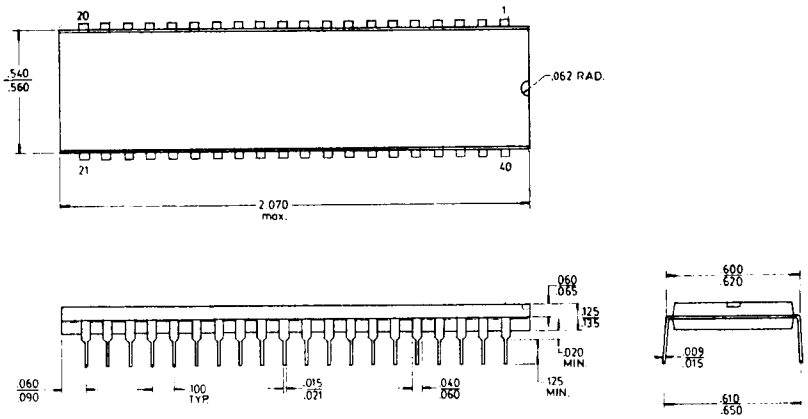
Number	Symbol	Parameter	12,16,20 MHz Min	Max	Notes (Data Direction)
1	TsDI(DAV)	Data In Setup Time	0		In
2	ThDI(DAV)	Data In Hold Time	145		In
3	TwDAV	Data Available Width	110		In
4	TdDAV(RDY)	DAV↓to RDY↓Delay		115	In
5	TdDAV(RDY)	DAV↑to RDY↑Delay		115	In
6	TdRDY(DAV)	RDY↑to DAV↓Delay	0		In
7	TdDO(DAV)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

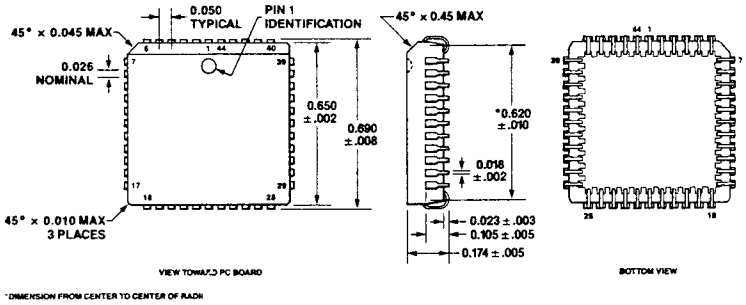
External I/O or Memory Read and Write Timing

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

PACKAGING INFORMATION



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



View Toward PC Board **Bottom View**

44-Pin Plastic Chip Carrier (PCC)

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

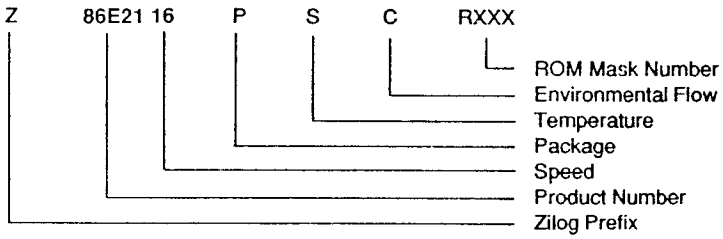
TEMPERATURE
Preferred
S = 0°C to +70°C

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

Customer Support

For answers to technical questions about the product, documentation, or any other issues with Zilog's offerings, please visit Zilog's Knowledge Base at http://www.zilog.com/kb.

For any comments, detail technical questions, or reporting problems, please visit Zilog's Technical Support at http://support.zilog.com.