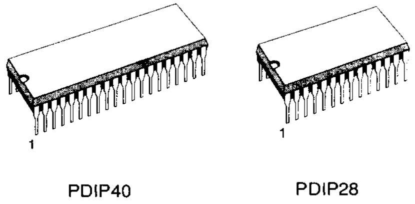
SGS-THOMSON MICROELECTRONICS

Z8611/Z8601/Z8681 Z8610/Z8600

2K/4K ROM, Z8 FAMILY MICROCOMPUTER

- COMPLETE MICROCOMPUTER
- 2K OR 4K BYTES OF ROM
- 28 PINS OR 40 PINS VERSIONS AVAILABLE
- 128 BYTES OF RAM
- 22 to 32 I/O LINES
- UP TO 60K BYTES ADDRESSABLE EXTERNAL SPACE EACH FOR PROGRAM AND DATA MEMORY
- 144-BYTE REGISTER FILE, INCLUDING 124 GENERAL PURPOSE REGISTERS, 4 I/O PORT REGISTERS, AND 16 STATUS AND CONTROL REGISTERS
- MINIMUM INSTRUCTION EXECUTION TIME 1µs, AT 12MHz
- 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 NINE WORKING-REGISTER GROUPS IN 1.5µs (8MHz)
- ON-CHIP OSCILLATOR WHICH ACCEPTS CRYSTAL OR EXTERNAL CLOCK DRIVE
- SINGLE + 5V POWER SUPPLY ALL PINS TTL COMPATIBLE
- AVAILABLE IN 8MHz AND 12MHz VERSIONS



(Ordering Information at the end of the datasheet)

DEVICE SUMMARY

DEVICE	ROM	I/O LINE	PACKAGE
Z8611	4K	32	PDIP40
Z8610	4K	22	PDIP28
Z8601	2K	32	PDIP40
Z8600	2K	22	PDIP28
Z8681	0	24	PDIP40

Figure 1 : Logic Functions

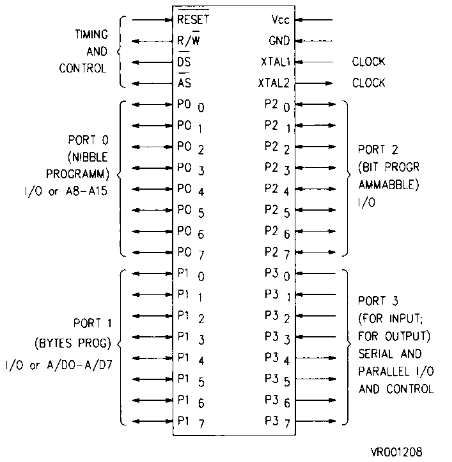


Figure 2 : 40 Pins DIL Configuration.

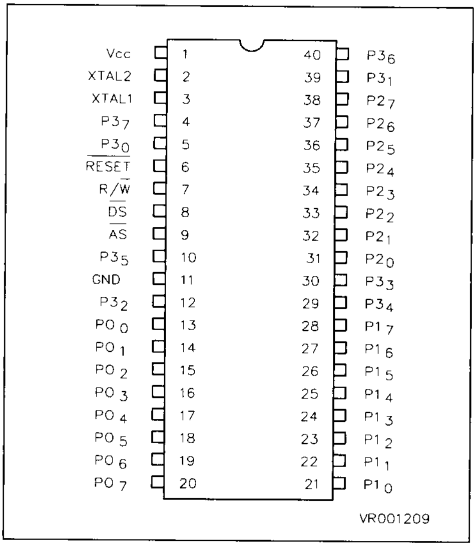
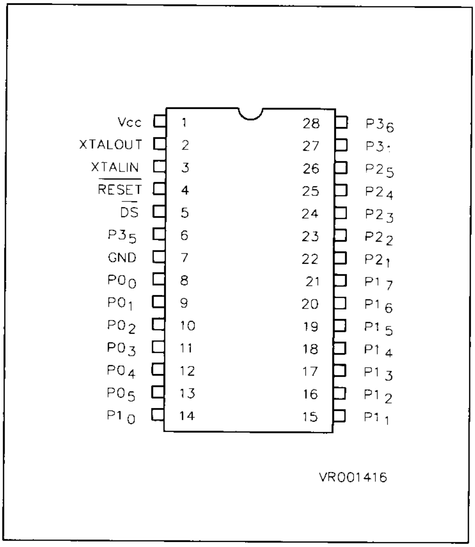


Figure 3 : 28 Pins DIL Configuration.



GENERAL DESCRIPTION

The Z8 microcomputer introduces a new level of sophistication to single-chip architecture. Compared to earlier single-chip microcomputers, the Z8 offers faster execution ; more efficient use of memory ; more sophisticated interrupt, input/output and bit-manipulation capabilities ; and easier system expansion.

Under program control, the Z8 can be tailored to the needs of its user. It can be configured as a stand-alone microcomputer with 2K or 4K bytes of internal ROM, a traditional microprocessor that manages up to 120K bytes of external memory, or a parallel-processing element in a system with other processors and peripheral controllers linked by the Z-BUS. In all configurations, a large number of pins remain available for I/O.

ARCHITECTURE

Z8 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 Z8 fulfills this with 32 pins dedicated to input/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, address/data bus for interfacing external memory.

Because the multiplexed address/data bus is merged with the I/O-oriented ports, the Z8 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 4).

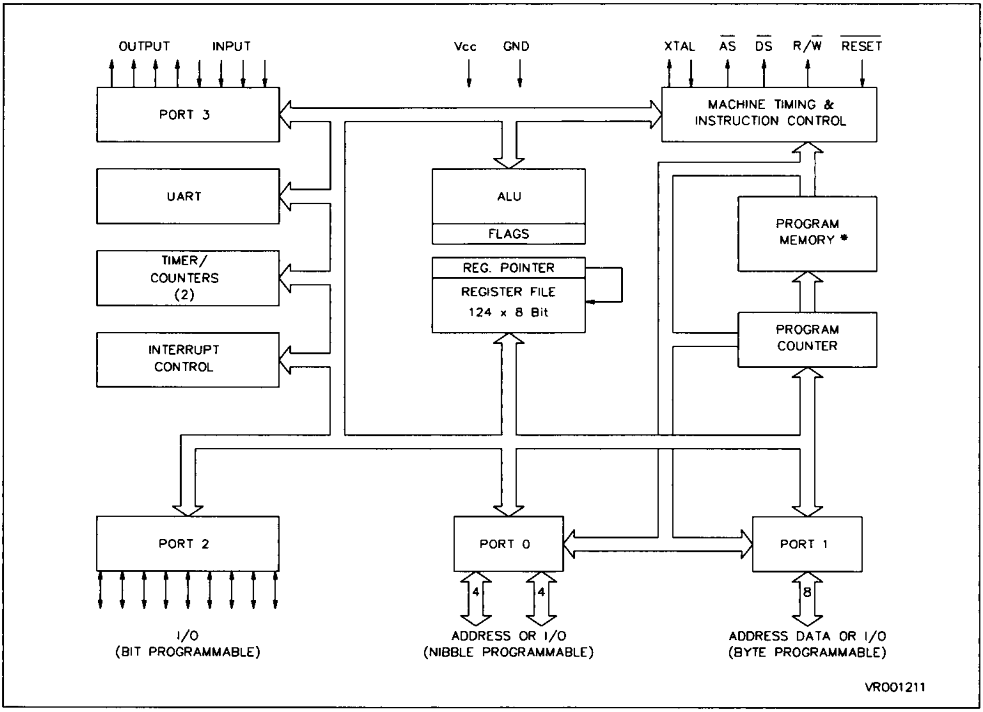
Three basic address spaces are available to support this wide range of configurations : program memory (internal and external), data memory (external) and the register file (internal). The 144-byte random-access register file is composed of 124 general-purpose registers, four I/O port registers, and 16 control and status registers.

To unburden the program from copying 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.

ARCHITECTURE (Continued)

Figure 4 : Block Diagram



* Z8601 = 2048 x 8 bit ; Z8611 = 4096 x 8 bit ; Z8681 = External Prog Memory.

PIN DESCRIPTIONS

P0$_0$-P0$_7$. *I/O Port Lines* (input/outputs, TTL compatible). 8 lines nibble-programmable that can be configured under program control for I/O or external memory interface.

P1$_0$-P1$_7$. *I/O Port Lines* (input/outputs, TTL compatible). 8 lines byte programmable that can be configured under program control for I/O or multiplexed address (A$_0$-A$_7$) and data (D$_0$-D$_7$) lines used to interface with program/data memory.

P2$_0$-P2$_7$. *I/O Port Lines* (input/outputs, TTL compatible). 8 lines bit programmable. In addition they can be configured to provide open-drain output.

P3$_0$-P3$_7$. *I/O Port Lines* (TTL compatible) 4 lines input (P3$_0$-P3$_3$), 4 lines output P3$_4$-P3$_7$). They can also be configured as control lines.

$\overline{\text{AS}}$. *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 $\overline{\text{AS}}$. Under program control, $\overline{\text{AS}}$ can be placed in the high-impedance state along with ports 0 and 1, Data Strobe and Read/Write.

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

$\overline{\text{RESET}}$. *Reset* (input, active Low). $\overline{\text{RESET}}$ initializes the Z86xx. When $\overline{\text{RESET}}$ is deactivated, program execution begins from internal program location 000CH.

R/$\overline{\text{W}}$. *Read/Write* (output). R/$\overline{\text{W}}$ is Low when the Z86xx is writing to external program or data memory.

PIN DESCRIPTIONS (Continued)

XTAL1, XTAL2. *Crystal 1, Crystal 2* (time-base input and output). These pins connect a parallel-resonant crystal (8 or 12MHz maximum) or an external single-phase clock (8 or 12MHz maximum) to the on-chip clock oscillator and buffer.

ADDRESS SPACES Z8601/Z8600

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 5). The first 2048 bytes consist of on-chip mask programmed ROM. At addresses 2048 and greater, the Z8601 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 Z8601 can address 62K bytes of external data memory beginning at locations 2048 (figure 6). External data memory may be included with or separated from the external program memory space.

$\overline{\text{DM}}$, 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 144-byte register file includes four I/O port registers (R0-R3), 124 general-purpose registers (R4-R127) and 16 control and status registers (R240-R255). These registers are assigned the address locations shown in figure 10.

Z8601 instructions can access registers directly or indirectly with an 8-bit address field. The Z8601 also allows short 4-bit register addressing using the Register Pointer (one of the control registers). In this case, the register file is divided into nine working-register groups, each occupying 16 contiguous locations (figure 11). The Register Pointer addresses the starting location of the active working-register group.

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 2048 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 5 : Program Memory Map.

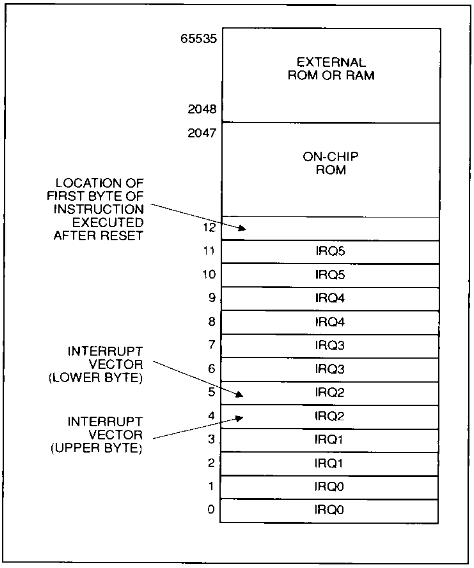
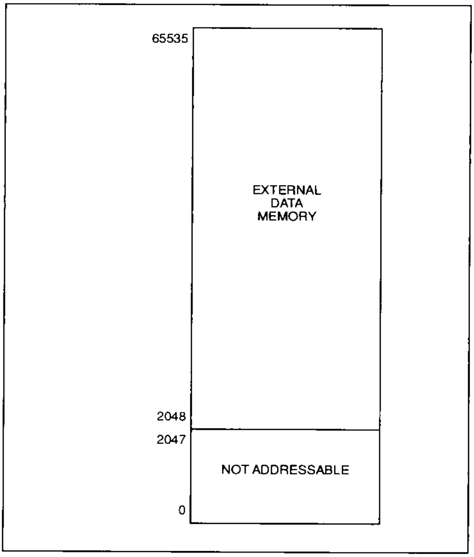

65535
EXTERNAL ROM OR RAM
2048
2047
ON-CHIP ROM
LOCATION OF FIRST BYTE OF INSTRUCTION EXECUTED AFTER RESET
12
11
IRQ5
10
IRQ5
9
IRQ4
8
IRQ4
INTERRUPT VECTOR (LOWER BYTE)
7
IRQ3
6
IRQ3
5
IRQ2
INTERRUPT VECTOR (UPPER BYTE)
4
IRQ2
3
IRQ1
2
IRQ1
1
IRQ0
0
IRQ0

Figure 6 : Data Memory Map.



ADDRESS SPACES Z8611/Z8610

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 7). The first 4096 bytes consist of on-chip mask programmed ROM. At addresses 4096 and greater, the Z8611 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 Z8611 can address 60K bytes of external data memory beginning at locations 4096 (figure 8). External data memory may be included with or separated from the external program memory space.

$\overline{\text{DM}}$, 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 144-byte register file includes four I/O port registers (R0-R3), 124 general-purpose registers (R4-R127) and 16 control and status registers (R240-R255). These registers are assigned the address locations shown in figure 10.

Z8611 instructions can access registers directly or indirectly with an 8-bit address field. The Z8611 also allows short 4-bit register addressing using the Register Pointer (one of the control registers). In this case, the register file is divided into nine working-register groups, each occupying 16 contiguous locations (figure 11). The Register Pointer addresses the starting location of the active working-register group.

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 7 : Program Memory Map.

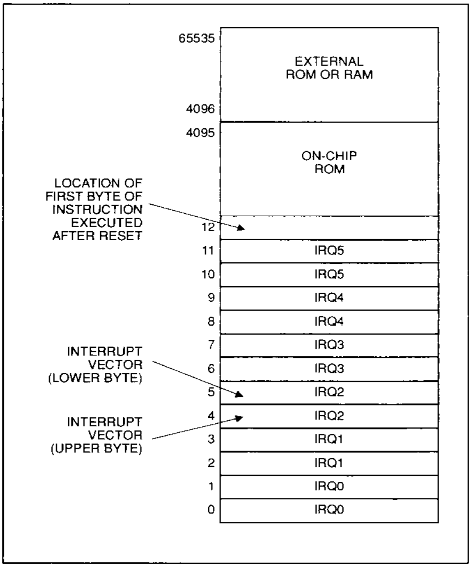
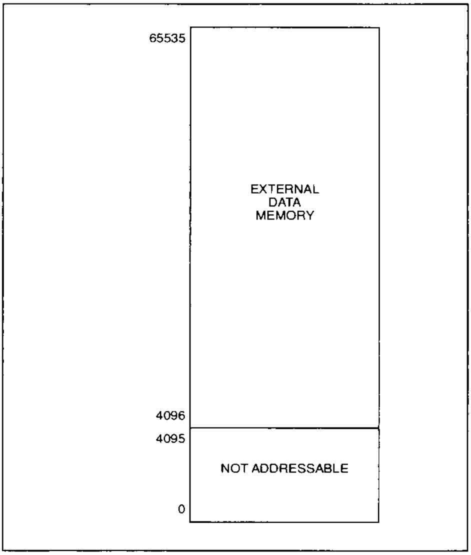

65535
EXTERNAL ROM OR RAM
4096
4095
ON-CHIP ROM
LOCATION OF FIRST BYTE OF INSTRUCTION EXECUTED AFTER RESET
12
11
IRQ5
10
IRQ5
9
IRQ4
8
IRQ4
7
IRQ3
INTERRUPT VECTOR (LOWER BYTE)
6
IRQ3
5
IRQ2
INTERRUPT VECTOR (UPPER BYTE)
4
IRQ2
3
IRQ1
2
IRQ1
1
IRQ0
0
IRQ0

Figure 8 : Data Memory Map.



ADDRESS SPACES Z8681

Program Memory. The Z8681 addresses 64K bytes of external program memory space (Figure 9).

The first 12 bytes of program memory are reserved for the interrupt vectors. These location contain six 16-bit vectors that correspond to the six available interrupts. Program execution begins at location 000C$_H$ after a reset.

Data Memory. The Z8681 can address 64K bytes of external data memory. 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.

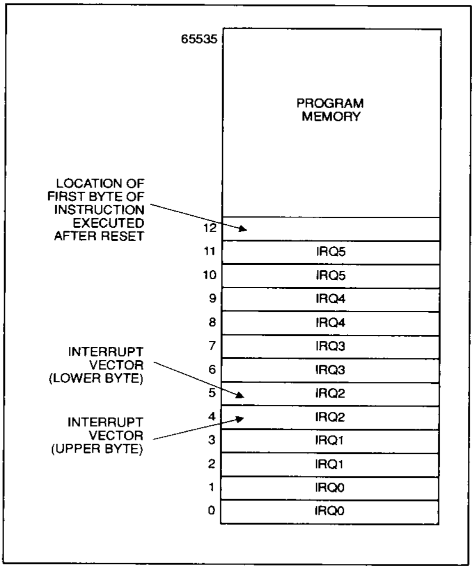
Register File. The 143-byte register file includes three I/O port registers (R0, R2, R3), 124 general-purpose registers (R4-R127) and 16 control and status registers (R240-R255.). These registers are assigned the address locations shown in Figure 10.

Z8681 instructions can access registers directly or indirectly with an 8-bit address field. This 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 nine working-register groups, each occupying 16 contiguous locations (Figure 10).

The Register Pointer addresses the starting location of the active working-register group (Figure 11).

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. An 8-bit Stack Pointer (R255) is used for the internal stack that resides within the 124 general-purpose registers (R4-R127).

Figure 9 : Address Spaces


65535
PROGRAM MEMORY
LOCATION OF FIRST BYTE OF INSTRUCTION EXECUTED AFTER RESET
12
11
IRQ5
10
IRQ5
9
IRQ4
8
IRQ4
7
IRQ3
INTERRUPT VECTOR (LOWER BYTE)
6
IRQ3
5
IRQ2
INTERRUPT VECTOR (UPPER BYTE)
4
IRQ2
3
IRQ1
2
IRQ1
1
IRQ0
0
IRQ0

ADDRESS SPACES (Continued)

Figure 10 : Register File.

LOCATION		IDENTIFIERS
255	STACK POINTER (BITS 7-0)	SPL
254	STACK POINTER (BITS 15-8)	SPH
253	REGISTER POINTER	RP
252	PROGRAM CONTROL FLAGS	FLAGS
251	INTERRUPT MASK REGISTER	IMR
250	INTERRUPT REQUEST REGISTER	IRQ
249	INTERRUPT PRIORITY REGISTER	IPR
248	PORT 0-1 MODE	P01M
247	PORT 3 MODE	P3M
246	PORT 2 MODE	P2M
245	T0 PRESCALER	PRE0
244	TIMER/COUNTER 0	T0
243	T1 PRESCALER	PRE1
242	TIMER/COUNTER 1	T1
241	TIMER MODE	TMR
240	SERIAL I/O	SIO
	NOT IMPLEMENTED	
127	GENERAL-PURPOSE REGISTERS	
4		
3	PORT 3	P3
2	PORT 2	P2
1	PORT 1	P1 *
0	PORT 0	P0

* P1 not used for ROMLESS version.

SERIAL INPUT/OUTPUT

Port 3 lines P30 and P37 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, with a maximum rate of 62.5K bits/second, for the 8MHz version.

The Z8 automatically adds a start bit and two stop bits to transmitted data (figure 12). 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 (IRQ4) 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 IRQ3 interrupt request.

Figure 11 : Register Pointer.

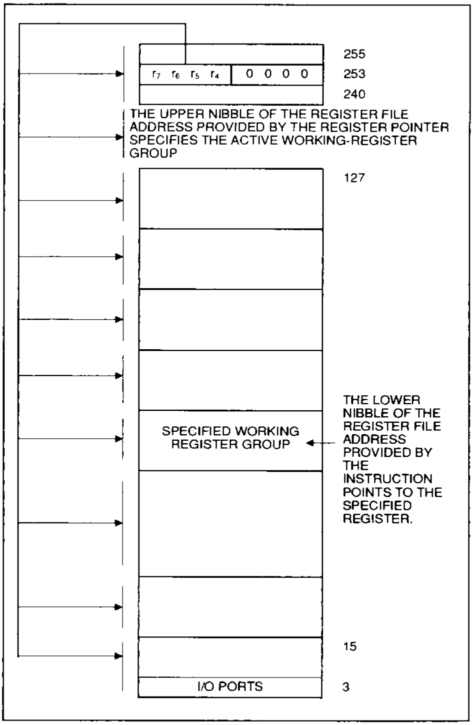
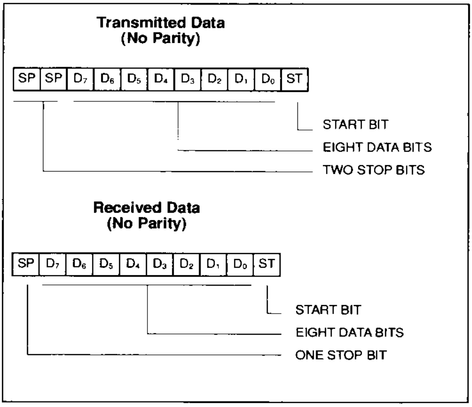
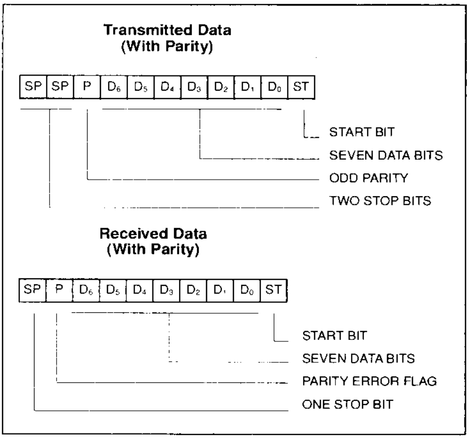

255
r7 r6 r5 r4 | 0 0 0 0
253
240
THE UPPER NIBBLE OF THE REGISTER FILE ADDRESS PROVIDED BY THE REGISTER POINTER SPECIFIES THE ACTIVE WORKING-REGISTER GROUP
127
SPECIFIED WORKING REGISTER GROUP
THE LOWER NIBBLE OF THE REGISTER FILE ADDRESS PROVIDED BY THE INSTRUCTION POINTS TO THE SPECIFIED REGISTER.
15
I/O PORTS
3

Figure 12 : Serial Data Formats.


Transmitted Data (No Parity)
SP SP D7 D6 D5 D4 D3 D2 D1 D0 ST
START BIT
EIGHT DATA BITS
TWO STOP BITS
Received Data (No Parity)
SP D7 D6 D5 D4 D3 D2 D1 D0 ST
START BIT
EIGHT DATA BITS
ONE STOP BIT

SERIAL INPUT/OUTPUT (Continued)

Figure 13 : Serial Data Formats (Continued)



COUNTER/TIMERS

The Z8 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 prescaler 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, 4MHz maximum for the 8MHz device and 6MHz maximum for the 12MHz device, divided by four, or an external signal input via Port 3. The Timer Mode register configures the external timer input as an external clock (1.5MHz 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 (T_{OUT}) through which T_0, T_1 or the internal clock can be output.

I/O PORTS Z8601, Z8611

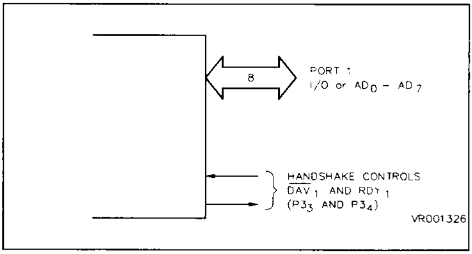
The Z8 has 32 lines dedicated to input and output. These lines are grouped into four ports of eight lines each and are configurable as input, output or address/data. Under software control, the ports can be programmed to provide address outputs, timing, status signals, serial I/O, and parallel I/O with 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 DAV_1 (ready and data available).

Memory locations greater than 4096 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 Z8 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 14 : Port 1 Configuration.

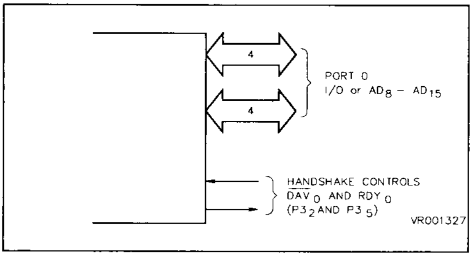


Port 0 can be programmed as a nibble I/O port, or as an address port for interfacing external memory. When used as an I/O port, Port 0 may be placed under handshake control. In this configuration, Port 3 lines $P3_2$ and $P3_5$ are used as the handshake controls DAV_0 and RDY_0. Handshake signal assignment is dictated by the I/O direction of the upper nibble $P0_4$-$P0_7$.

I/O PORTS (Continued)

For external memory references, Port 0 can provide address bits A_8-A_{11} (lower nibble) or A_{12}-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/$\overline{W}$.

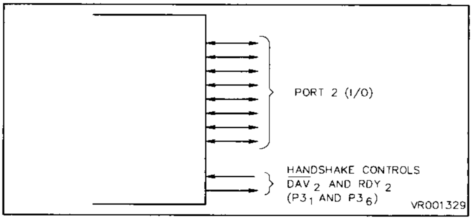
Figure 15 : Port O Configuration.



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 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.

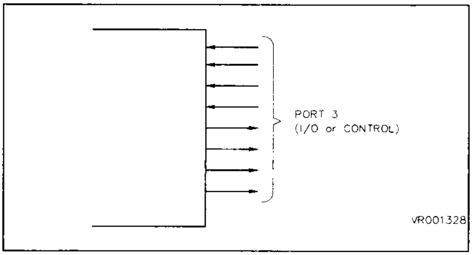
Figure 16 : Port 2 Configuration.



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 17 : Port 3 Configuration.



INTERRUPTS

The Z86x1 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 Z86x1 interrupts are vectored. 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 ($C_1 \leq 15pF$) from each pin to ground. The specifications for the crystal are as follows :

– AT cut, series resonant

– Fundamental types, 8MHz and 12MHz

– Series resistance, $R_S \leq 100\Omega$.

I/O Ports (Z8681 only)

The Z8681 has 24 lines available for input and output. These lines are grouped into four ports of eight lines each and are configurable as input, output or address.

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 I is a dedicated Z-BUS compatible memory interface. The operations of Port 1 are supported by the Address Strobe ($\overline{\text{AS}}$) and Data Strobe ($\overline{\text{DS}}$) lines, and by the Read/Write (R/$\overline{\text{W}}$) and Data Memory ($\overline{\text{DM}}$)control lines. The low-order program and data memory addresses (A_0-A_7) are output through Port 1 (Figure 18) and are multiplexed with data in/out (D_0-D_7).

Instruction fetch and data memory read/write operations are done through this port.

Port 1 cannot be used as a register nor can a handshake mode be used with this port.

The Z8681, wake up with the 8 bits of Port 1 configured as address outputs for external memory. If more than eight address line are required with the Z8681, additional lines can be obtained by programming Port 0 bits as address bits. The least-significant four bits of Port 0 can be configured to supply address bits A_8-A_{11} for 4K byte addressing or both nibbles of Port 0 can be configured to supply address bits A_8-A15 for 64K byte addressing.

Port 0 can be programmed as a nibble I/O port, or as an address port for interfacing external memory (Figure 19).

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 DAV_0 and RDY_0. Handshake signal assignment is dictated by the I/O direction of the upper nubble $P0_4$-$P0_7$.

For external memory references, Port 0 can provide address bits A_8-A_{11} (lower nibble) or A_8-A_{15} (lower and upper nibbles) 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.

Port 0 lines float after reset ; their logic state is unknown until the execution of an initialization routine that configures Port 0.

Such an initialization routine must reside within the first 256 bytes of executable code and must be physically mapped into memory be forcing the Port 0 address lines to a known state. See Figure 20. The proper Port initialization sequence is :

1. Write initial address (A_8-A_{15}) of initialization routine to Port 0 address lines;

2. Configure Port 0 Mode Register to output A_8-A_{15} (or A_8-A_{11}).

To permit the use of slow memory, an automatic wait mode of two oscillator clock cycles is configured for the bus timing of the Z8681 after each reset. The initialization routine could include reconfiguration to eliminate this extended timing mode.

The following example illustrates the manner in which an initialization routine can be mapped in a Z8681 system with 4K of memory.

Figure 18 : Port 1.

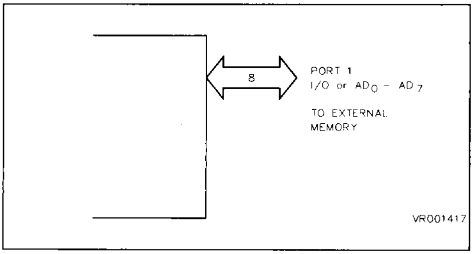
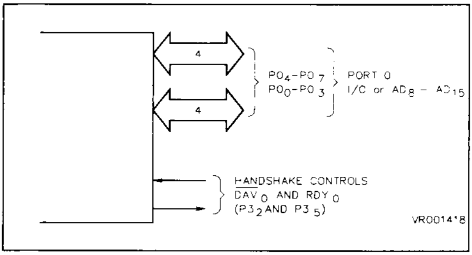


Figure 19 : Port 0.

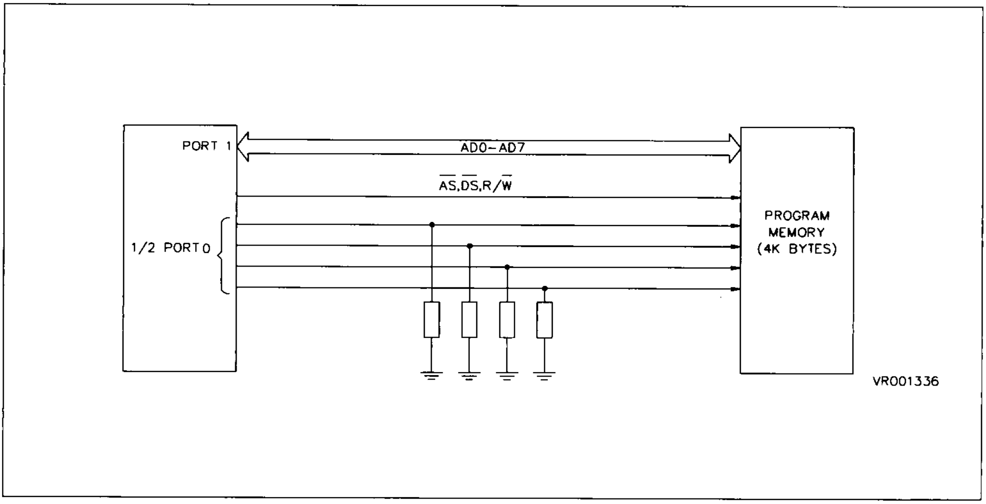


I/O PORTS (Continued)

Example. In figure 20, the intitialization routine is mapped to the first 256 bytes of program memory. Pull-down resistors maintain the address lines at a logic 0 level when these lines are floating. The leakage current caused by fanout must be taken into consideration when selecting the value of the pulldown resistors. The resistor value must be large enough to allow the Port 0 output driver to pull the line to a logic one.

Generally, pulldown resistors are incompatible with TTL loads. If Port 0 drives into TTL input loads (I_{LOW} = 1.6mA) the external resistors should be tied to V_{CC} and the initialization routine put in address space FF00H-FFFFH.

Figure 20 : Z8681, Port 0 Address Lines Tied to Logic 0.


PORT 1
AD0–AD7
$\overline{AS}$,$\overline{DS}$,R/$\overline{W}$
1/2 PORT 0
PROGRAM MEMORY (4K BYTES)
VR001336

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 adress 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	**F1**	user flag
H	Half-carry flag	**F2**	user flag

b7							b0
C	Z	S	V	D	H	F2	F1

Affected flags are indicated by :

O	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

Figure 21 : One-Byte Instruction Format.

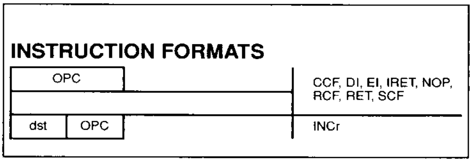


Figure 22 : Two-bytes instruction Format.

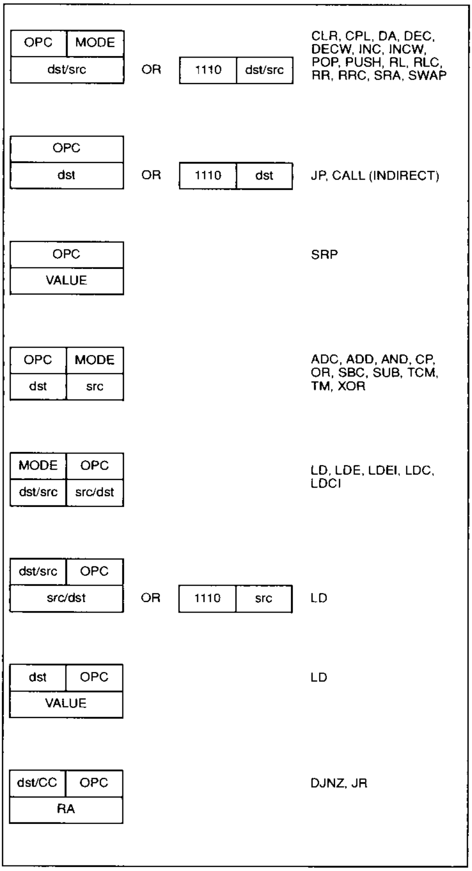
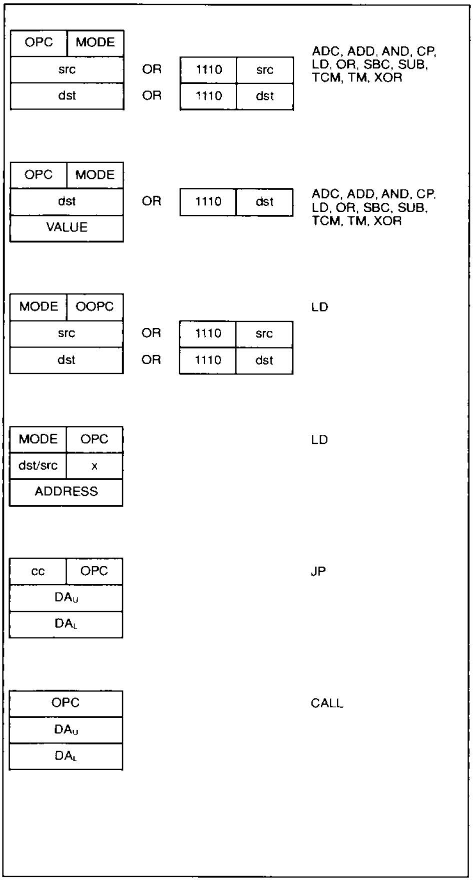
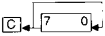
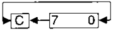
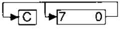
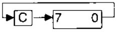
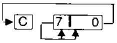
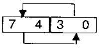


Figure 23 : Three-Bytes Instruction Format.



Instruction and Operation	Addr Mode		Opcode Byte (Hex)	Flags Affected					
	dst	src		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 □	*	*	*	*	-	-
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	-	-	-	-	-	-
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	*	*	*	*	*	*
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 R IR IR	Im R r X r Ir r R IR Im Im R	rC r8 r9 r = 0 - F C7 D7 E3 F3 E4 E5 E6 E7 F5	-	-	-	-	-	-
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	*	*	*	*	-	-
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	*

Note : 1.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.

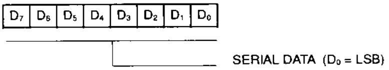
Instruction and Operation	Addr Mode		Opcode Byte (Hex)	Flags Affected					
	dst	src		C	Z	S	V	D	H
SCF C ← 1			DF	1	-	-	-	-	-
SRA dst 	R IR		D0 D1	*	*	*	0	-	-
SRP src RP ← src		Im	31	-	-	-	-	-	-
SUB dst, src dst ← dst - src	(Note 1)		2□	*	*	*	*	1	*
SWAP dst 	R IR		F0 F1	X	*	*	X	-	-
TCM dst, src (NOT dst) AND src	(Note 1)		6□	-	*	*	0	-	-
TM dst, src dst AND src	(Note 1)		7□	-	*	*	0	-	-
XOR dst, src dst ← dst XOR src	(Note 1)		B□	-	*	*	0	-	-

Note : 1.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.

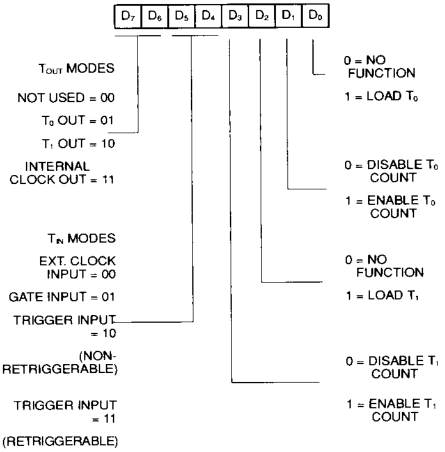
Addr Mode		Lower Opcode Nibble
dst	src	
r	r	2
r	Ir	$\boxed{3}$
R	R	$\boxed{4}$
R	IR	$\boxed{5}$
R	IM	$\boxed{6}$

CONTROL REGISTERS

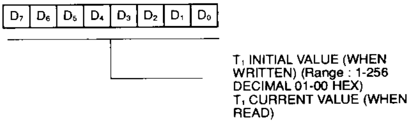
R240 SIO
Serial I/O Register
(F0H ; Read/Write)



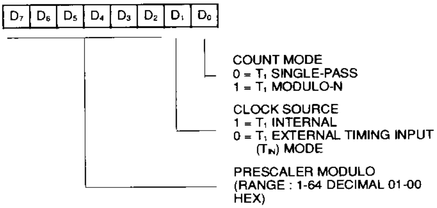
R241 TMR
Timer Mode Register
(F1H ; Read/Write)



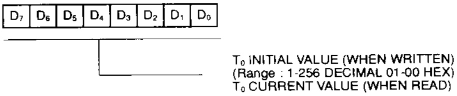
R242 T1
Counter Timer 1 Register
(F2H ; Read/Write)


D7 D6 D5 D4 D3 D2 D1 D0
T1 INITIAL VALUE (WHEN WRITTEN) (Range : 1-256 DECIMAL 01-00 HEX)
T1 CURRENT VALUE (WHEN READ)

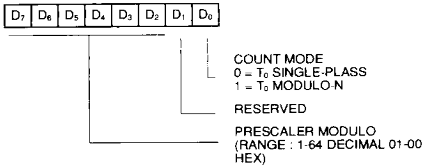
R243 PRE1
Prescaler 1 Register
(F3H ; Write Only)



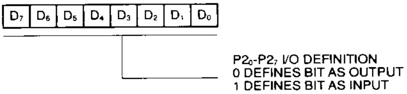
R244 T0
Counter/Timer 0 Register
(F4H ; Read/Write)


D7 D6 D5 D4 D3 D2 D1 D0
T0 INITIAL VALUE (WHEN WRITTEN)
(Range : 1-256 DECIMAL 01-00 HEX)
T0 CURRENT VALUE (WHEN READ)

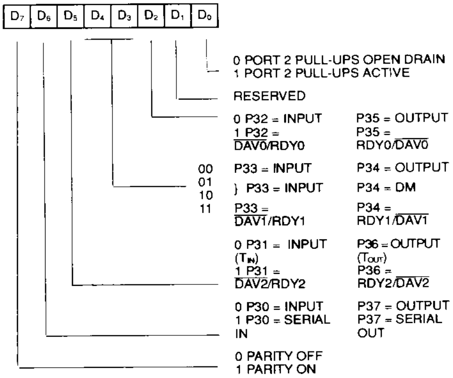
R245 PRE0
Prescaler 0 Register
(F5H ; Write Only)



R246 P2M
Port 2 Mode Register
(F6H ; Write Only)

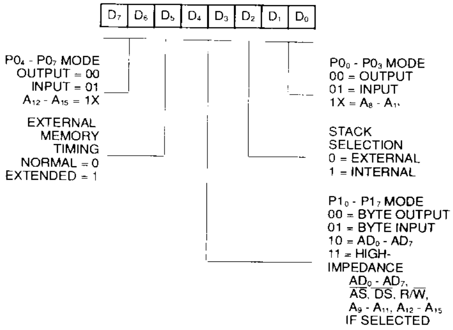


R247 P3M
Port 3 Mode Register
(F7H ; Write Only)

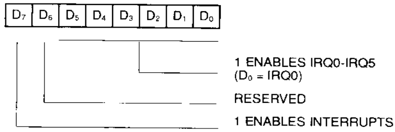


CONTROL REGISTERS (Continued)

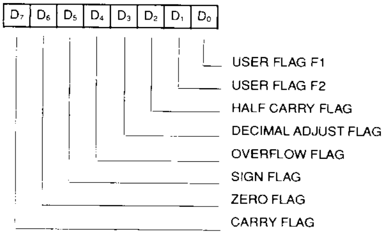
R248 P01M
Port 0 and 1 Mode Register
($F8_H$; Write Only)



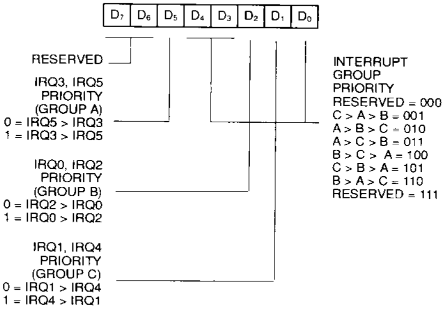
R251 IMR
Interrupt Mask Register
(FB_D ; Read/Write)



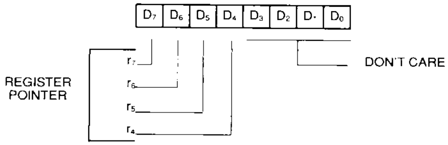
R252 FLAGS
Flag Register
(FC_H ; Read/Write)



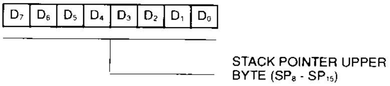
R249 IPR
Interrupt Priority Register
($F9_H$; Write Only)



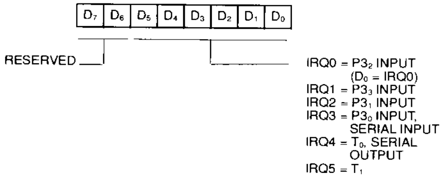
R253 RP
Register Pointer
(FD_H ; Read/Write)



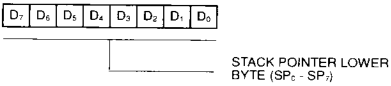
R254 SPH
Stack Pointer
(FE_H ; Read/Write)



R250 IRQ
Interrupt Request Register
(FA_H ; Read/Write)



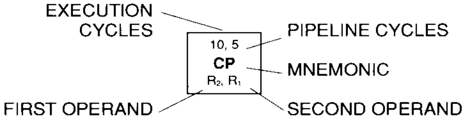
R255 SPL
Stack Pointer
(FF_H ; Read/Write)



OPCODE MAP

Lower Nibble (Hex)

Upper 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** r_1	
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	↓	↓	↓	↓	↓	↓	↓	
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	↓	↓	↓	↓	↓	↓	↓	
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	6, 7 **SWAP** R_1	6, 7 **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
R1 or r1 = Dst address
R2 or R2 = Src address
Sequence: Opcode, First Operand, Second Operand
Note : The blank areas are not defined.

Note : * 2-byte instruction fetch cycle appears as a 3-byte instruction.

ABSOLUTE MAXIMUM RATINGS*

Symbol	Parameter	Value	unit
	Voltage on any pin relative to ground	-0.3 to 7.0	V
T_A	Operating Ambient Temperature	0 to 70	°C
T_{STG}	Storage Temperature	-65 to 150	°C

Note : Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions 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.

TEST CONDITIONS

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

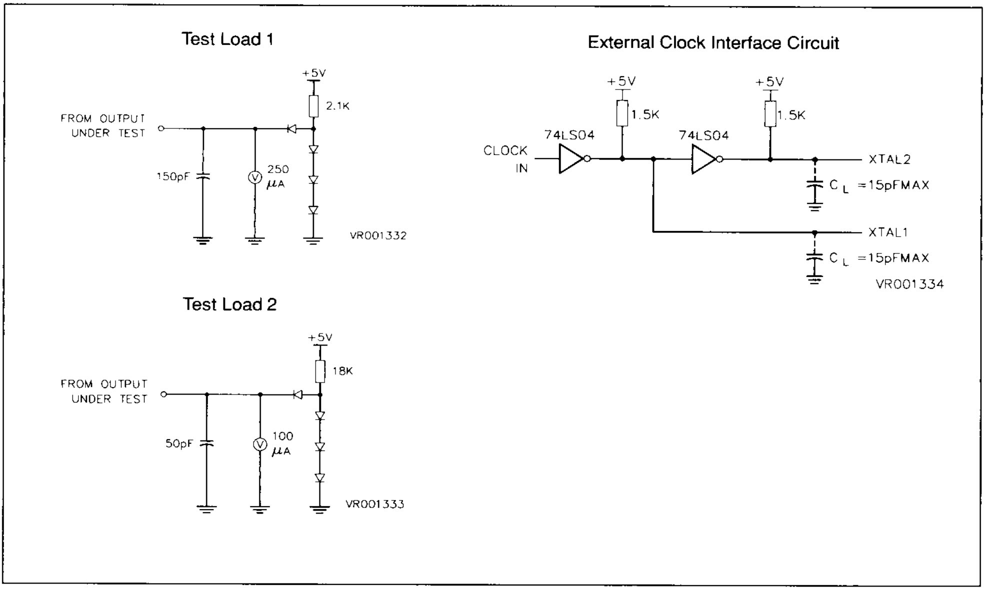
Standard test conditions are as follows :

– $+4.5 \leq V_{CC} \leq +5.5V$

– GND = 0V

– $0°C \leq T_A \leq +70°C$

Figure 24 : Test Circuits



Test Load 1

Test Load 2

External Clock Interface Circuit

DC CHARACTERISTICS

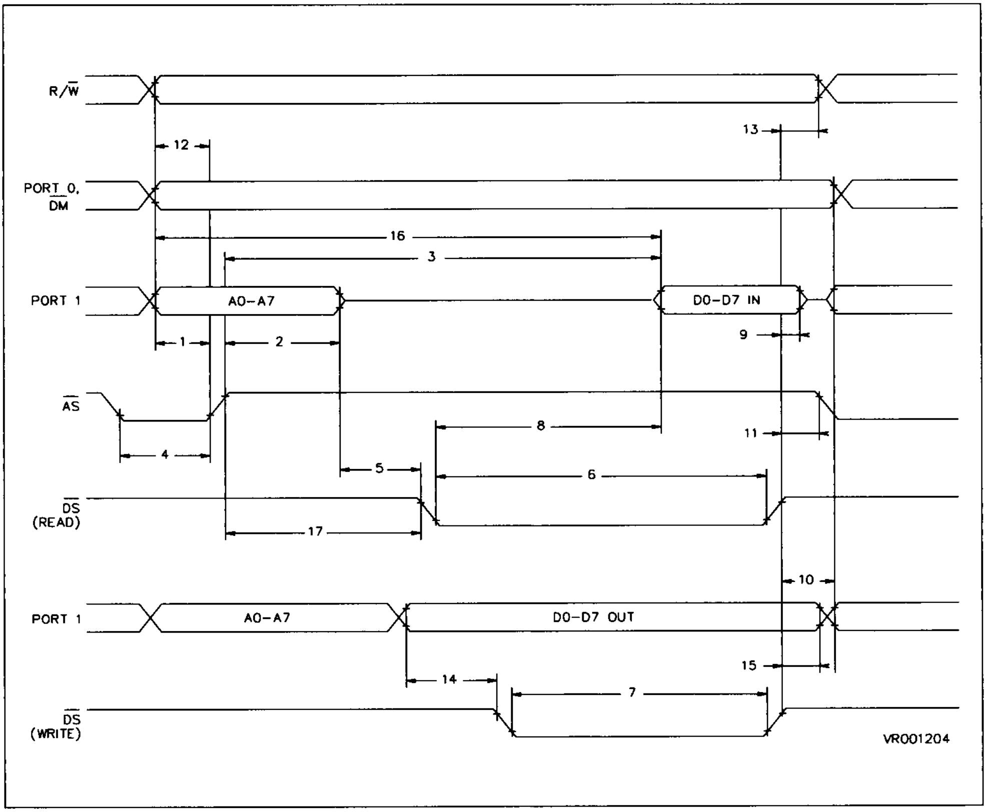
Symbol	Parameter	Min.	Max.	unit	Condition
V_{CH}	Clock input High Voltage	3.8	V_{CC}	V	Driven by External Clock Generator
V_{CL}	Clock Input Low Voltage	-0.3	0.8	V	Driven by External Clock Generator
V_{IH}	nput High Voltage	2.0	V_{CC}	V	
V_{IL}	Input Low Voltage	-0.3	0.8	V	
V_{RH}	Reset Input High Voltage	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}= -250µA
V_{OL}	Output Low Voltage		0.4	V	I_{OL} = +2.0mA
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}	V_{CC} Supply Current		180	mA	
I_{MM}	V_{MM} Supply Current		10	mA	Power Down Mode
V_{MM}	Backup Supply Voltage	3	V_{CC}	V	Power Down

EXTERNAL I/O OR MEMORY, READ WRITE AND CLOCK CYCLE TIMING

N°	Symbol	Parameter	Z86xx/8MHz			Z86xx/12MHz			Notes
			Min.	Max.	Equation	Min.	Max.	Equation	
1	$T_{DA(AS)}$	Address Valid to $\overline{AS}$ ↑ Delay	50		T_{PC} - 75	35		T_{PC} - 50	1, 2, 3
2	$T_{DAS(A)}$	$\overline{AS}$ ↑ to Address Float Delay	70		T_{PC} - 55	45		T_{PC} - 40	1, 2, 3
3	$T_{DAS(DR)}$	$\overline{AS}$ ↑ to Read Data Required Valid		360	$4T_{PC}$ - 140		220	$4T_{PC}$ - 110	1, 2, 3, 4
4	T_{WAS}	$\overline{AS}$ Low Width	80		T_{PC} - 45	55		T_{PC} - 30	1, 2, 3
5	$T_{DAZ(DS)}$	Address Float to $\overline{DS}$ ↓	0		$3T_{PC}$ - 125	0		$3T_{PC}$ - 65	1
6	T_{WDSR}	$\overline{DS}$ (Read) Low Width	250		$2T_{PC}$ - 90	185		$2T_{PC}$ - 55	1, 2, 3, 4
7	T_{WDSW}	$\overline{DS}$ (Write) Low Width	160		$3T_{PC}$ - 175	110		$3T_{PC}$ - 120	1, 2, 3, 4
8	$T_{DDSR(DR)}$	$\overline{DS}$ ↓ to Read Data Required Valid		200	T_{PC} - 55		130	T_{PC} - 40	1, 2, 3, 4
9	$T_{HDR(DS)}$	Read Data to $\overline{DS}$ ↓ Hold Time	0			0			1
10	$T_{DDS(A)}$	$\overline{DS}$ ↑ to Address Active Delay	70		T_{PC} - 55	45		T_{PC} - 30	1, 2, 3
11	$T_{DDS(AS)}$	$\overline{DS}$ ↑ to $\overline{AS}$ ↓ Delay	70		T_{PC} - 75	55		T_{PC} - 55	1, 2, 3
12	$T_{DR/W(AS)}$	R/$\overline{W}$ Valid to $\overline{AS}$ ↑ Delay	50		T_{PC} - 65	30		T_{PC} - 50	1, 2, 3
13	$T_{DDS(R/W)}$	$\overline{DS}$ ↑ to R/$\overline{W}$ Not Valid	60		T_{PC} - 75	35		T_{PC} - 50	1, 2, 3
14	$T_{DDW(DSW)}$	Write Data Valid to $\overline{DS}$ (Write) ↓ Delay	50		T_{PC} - 55	35		T_{PC} - 40	1, 2, 3
15	$T_{DDS(DW)}$	$\overline{DS}$ ↑ to Write Data Not Valid Delay	70		$5T_{PC}$ - 215	45		$5T_{PC}$ - 160	1, 2, 3
16	$T_{DA(DR)}$	Address Valid to Read Data Required Valid		410	T_{PC} - 45		255	T_{PC} - 30	1, 2, 3, 4
17	$T_{DAS(DS)}$	$\overline{AS}$ ↑ to $\overline{DS}$ ↓ Delay	80			55			1, 2, 3

Notes : All values in ns.
1. Test Load 1.
2. Timing numbers given are for minimum T_{PC}.
3. Also see clock cycle time dependent characteristics table.
4. When using extended memory timing add 2 T_{PC}.

Figure 25 : External I/O or Memory Read/Write Timing.



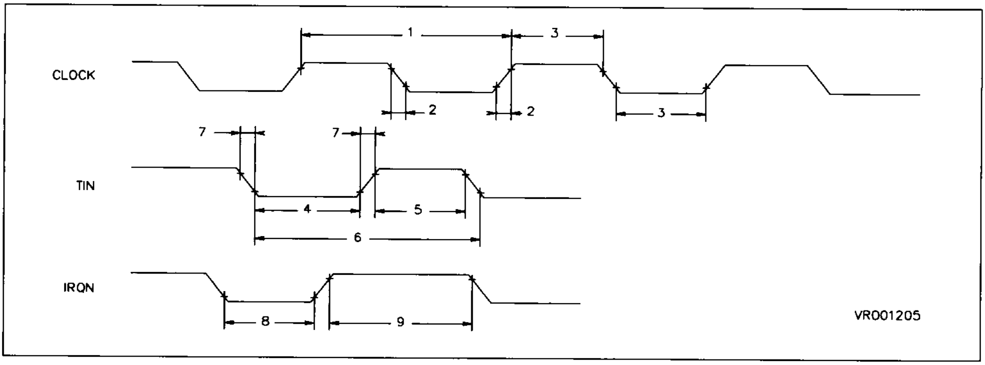
ADDITIONAL TIMING TABLE

N°	Symbol	Parameter	Z86xx/8MHz		Z86xx/12MHz		Unit	Notes
			Min.	Max.	Min.	Max.		
1	T_{PC}	Input Clock Period	125	1000	83	1000	ns	1
2	T_{RC}, T_{FC}	Clock Input Rise And Fall Times		25		15	ns	1
3	T_{WC}	Input Clock Width	37		26		ns	1
4	T_{WTINL}	Timer Input Low Width	100		70		ns	2
5	T_{WTINH}	Timer Input High Width	$3T_{PC}$		$3T_{PC}$		ns	2
6	T_{PTIN}	Timer Input Period	$8T_{PC}$		$8T_{PC}$		ns	2
7	T_{RTIN}, T_{FTIN}	Timer Input Rise And Fall Times		100		100	ns	2
8a	T_{WIL}	Interrupt Request Input Low Time	100		70		ns	2, 3
8b	T_{WIL}	Interrupt Request Input Low Time	$3T_{PC}$		$3T_{PC}$		ns	2, 4
9	T_{WIH}	Interrupt Request Input High Time	$3T_{PC}$		$3T_{PC}$		ns	2, 3

Notes :

1. Clock timing references uses 3.8V for a logic "1" and 0.8 for a logic "0".
2. Timing reference uses 2.0V for a logic "1" and 0.8V for a logic "0".
3. Interrupt request via Port 3 ($P3_1$-$P3_3$).
4. Interrupt request via Port 3 ($P3_0$).

Figure 26 : Additional Timing.



HANDSHAKE TIMING

N°	Symbol	Parameter	Z86xx/8MHz		Z86xx/12MHz		Unit	Notes
			Min.	Max.	Min.	Max.		
1	$T_{SDI(DAV)}$	Data In Setup Time	0		0		ns	
2	$T_{HDI(DAV)}$	Data In Hold Time	230		160		ns	
3	T_{WDAV}	Data Available Width	175		120		ns	
4	$T_{DDAVIF(RDY)}$	$\overline{DAV}$ ↓ Input to RDY ↓ Delay		175		120	ns	1, 2
5	$T_{DDAVOF(RDY)}$	$\overline{DAV}$ ↓ Output to RDY ↓ Delay	0		0		ns	1, 3
6	$T_{DDAVIR(RDY)}$	$\overline{DAV}$ ↑ Input to RDY ↑ Delay		175		120	ns	1, 2
7	$T_{DDAVOR(RDY)}$	$\overline{DAV}$ ↑ Output to RDY ↑ Delay	0		0		ns	1, 3
8	$T_{DDO(DAV)}$	Data Out to $\overline{DAV}$ ↓ Delay	50		30		ns	1
9	$T_{DRDY(DAV)}$	Rdy ↓ Input to $\overline{DAV}$ ↑ Delay	0	200	0	140	ns	1

Notes :
1. Test Load 1.
2. Input handshake.
3. Output handshake..
4. All timing references use 2.0V for a logic "1" and 0.8V for a logic "0".

Figure 27 : Input Handshake Timing.

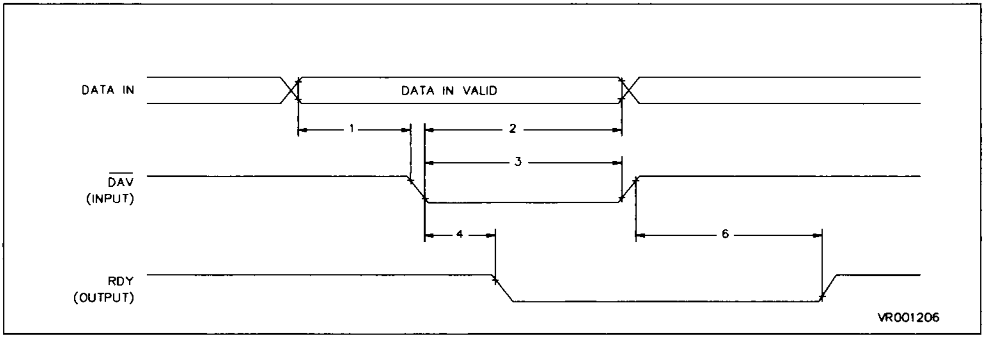


Figure 28 : Output Handshake Timing.

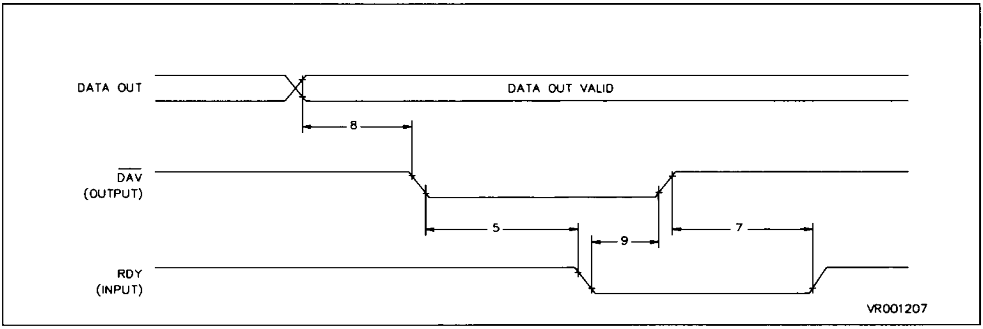
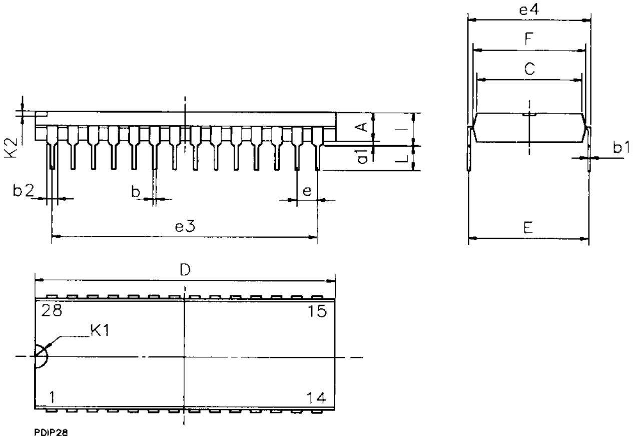
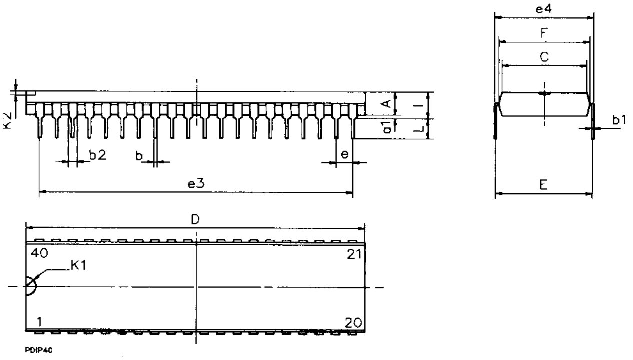


Figure 29 : 28-Lead Plastic Dual In Line Package (B)



Dim.	mm			inches		
	Min	Typ	Max	Min	Typ	Max
A	–	–	–	–	–	–
a1		0.63			0.025	
b		0.45			0.018	
b1	0.23		0.31	0.009		0.012
b2		1.27			0.050	
C	–	–	–	–	–	–
D			37.34			1.470
E	15.20		16.68	0.598		0.657
e		2.54			0.100	
e3		33.02			1.300	
e4	–	–	–	–	–	–
F			14.10			0.555
I		4.45			0.175	
L		3.30			0.130	

Figure 30 : 40-Lead Plastic Dual In Line Package (B)



Dim.	mm			inches		
	Min	Typ	Max	Min	Typ	Max
A	–	–	–	–	–	–
a1		0.63			0.025	
b		0.45			0.018	
b1	0.23		0.31	0.009		0.012
b2		1.27			0.050	
C	–	–	–	–	–	–
D			52.58			2.070
E	15.20		16.68	0.598		0.656
e		2.54			0.100	
e3		48.26			1.900	
e4	–	–	–	–	–	–
F			14.10			0.555
I		4.45			0.175	
L		3.30			0.129	

ORDERING INFORMATION

Type	Description	Frequency	Range	Package
Z8601xxB1	2K ROM	8MHz	0 to + 70°C	PDIP40
Z8600xxB1	2K ROM	8MHz	0 to + 70°C	PDIP28
Z8601AxxB1	2K ROM	12MHz	0 to + 70°C	PDIP40
Z8600AxxB1	2K ROM	12MHz	0 to + 70°C	PDIP28
Z8611xxB1	4K ROM	8MHz	0 to + 70°C	PDIP40
Z8610xxB1	4K ROM	8MHz	0 to + 70°C	PDIP28
Z8611AxxB1	4K ROM	12MHz	0 to + 70°C	PDIP40
Z8610AxxB1	4K ROM	12MHz	0 to + 70°C	PDIP28
Z8681B1	ROMLESS	8MHz	0 to + 70°C	PDIP40
Z8681AB1	ROMLESS	12MHz	0 to + 70°C	PDIP40

Information furnished is believed to be accurate and reliable. However, SGS-THOMSON Microelectronics assumes no responsability for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of SGS-THOMSON Microelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all informations previously supplied. SGS-THOMSON Microelectronics products are not authorized for use as critical components in life support devices or system without express written approval of SGS-THOMSON Microelectronics.

SGS-THOMSON Microelectronics GROUP OF COMPANIES
Australia - Brazil - France - Germany - Hong Kong - Italy - Japan - Korea - Malaysia - Malta - Morocco - The Netherlands -
Singapore - Spain - Sweden - Switzerland - Taiwan - United Kingdom - U.S.A.