ADVANCED INFORMATION
Product Specification

T-49-19-07

Z86C00/C10/C20 CMOS Z8® MCU

FEATURES

- Complete microcomputer, 2K (86C00), 4K (86C10), or 8K (86C20) bytes of ROM, 124 bytes of RAM (256 bytes - Z86C20), and 22 I/O lines.
- 144-byte register file, including 124 (238 - Z86C20) general-purpose registers, four I/O port registers, and 14 status and control registers.
- Average instruction execution time of 1.5 us, maximum of 2.8 us.
- Vectored, priority interrupts for I/O and counter/timers.
- 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 1.0 us.
- On-chip oscillator which accepts crystal, external clock drive, LC, ceramic resonator.
- Standby modes — Halt and Stop.
- Single +5V power supply — all pins TTL-compatible.
- 8 and 12 MHz
- CMOS process.

GENERAL DESCRIPTION

Z86C10/C20 microcomputer (Figures 1 and 2) introduces a new level of sophistication to single-chip architecture. Compared to earlier single-chip microcomputers, the **Z86C10/C20** offers faster execution; more efficient use of memory; more sophisticated interrupt, input/output and bit-manipulation capabilities; and easier system expansion.

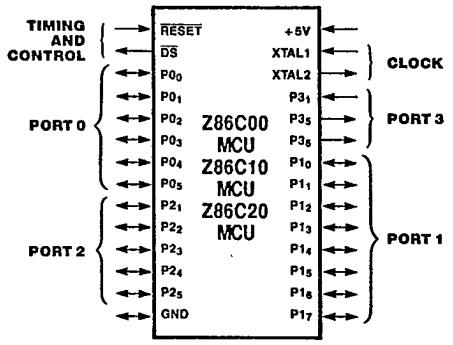


Figure 1. Pin Functions

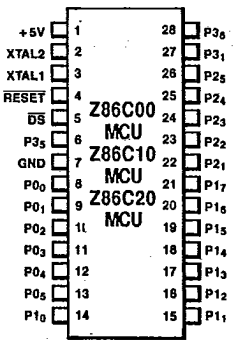


Figure 2. Pin Assignments

PIN DESCRIPTIONS

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

$P0_0$-$P0_5$, $P1_0$-$P1_7$, $P2_1$-$P2_5$, $P3_1$, $P3_5$, $P3_6$. *I/O Port lines* (bidirectional, TTL-compatible). These 22 I/O lines are grouped in four ports that can be configured under program control for I/O.

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

XTAL1, XTAL2. *Crystal 1, Crystal 2* (time-base input and output). These pins connect a parallel-resonant crystal to the on-chip clock oscillator and buffer.

ARCHITECTURE

The MCU's 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. (Figure 3).

Microcomputer applications demand powerful I/O capabilities. The MCU fulfills this with 22 pins dedicated to input and output. These lines are grouped in four ports and are configurable under software control to provide timing, status signals, and parallel I/O.

Two basic internal address spaces are available to support this wide range of configurations: program memory and the register file. The 144-byte random-access register file is composed of 124 general-purpose registers, four I/O port registers, and 14 control and status registers.

To unburden the program from coping with real-time problems such as counting/timing, two counter/timers with a large number of user-selectable modes are offered on-chip.

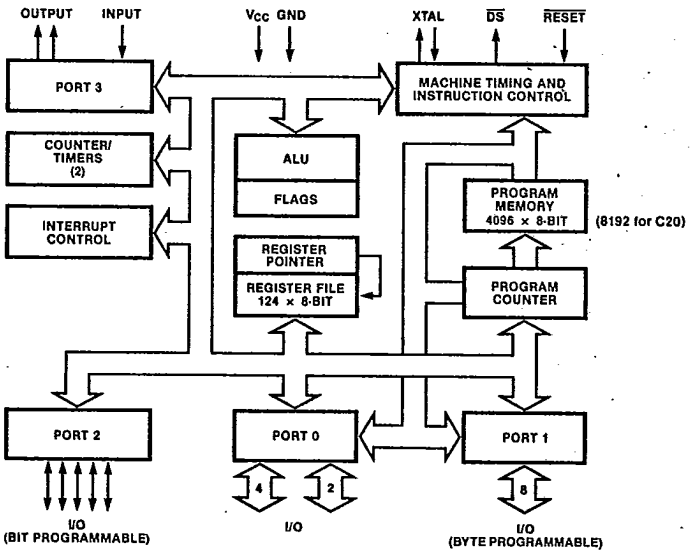


Figure 3. Functional Block Diagram

STANDBY MODE

The Z86C00/C10/C20'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:

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

ADDRESS SPACES

Program Memory. The 16-bit program counter addresses 4K or 8K bytes of program memory space as shown in Figure 4.

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

Register File. The 144-byte register file includes four I/O port registers (R_0-R_3), 124 general-purpose registers (R_4-R_{127}) and 15 control and status registers (R_{241}-R_{255}). These registers are assigned the address locations shown in Figure 5.

Instructions can access registers directly or indirectly with an 8-bit address field. The MCU 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 6). The Register Pointer addresses the starting location of the active working-register group.

Stacks. An 8-bit Stack Pointer (R_{255}) is used for the internal stack that resides within the 124 general-purpose registers (R_4-R_{127}).

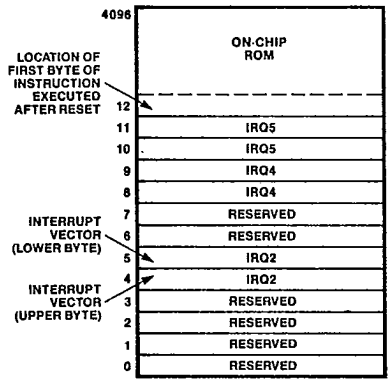


Figure 4. Program Memory Map

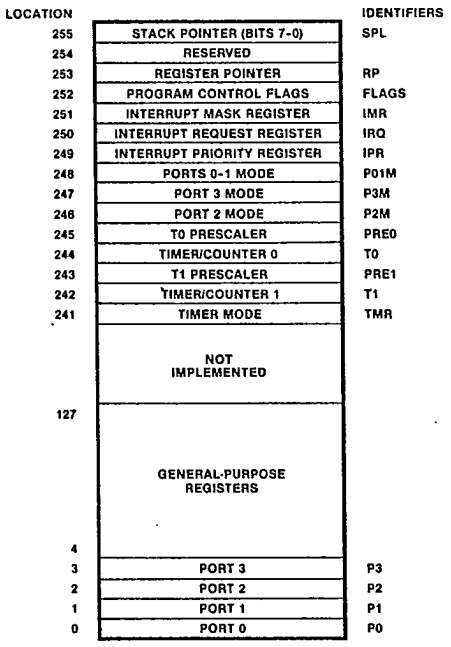


Figure 5. Register File

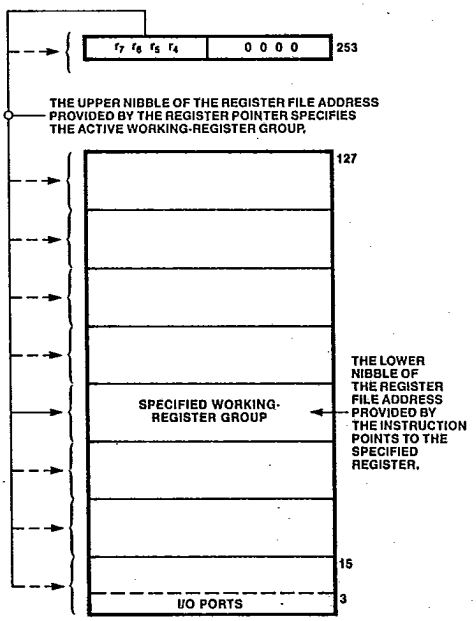


Figure 6. Register Pointer

COUNTER/TIMERS

The MCU 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 , 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

The MCU has 22 lines dedicated to input and output grouped in four ports. Under software control, the ports can be programmed to provide address outputs, timing, status signals, and parallel I/O. All ports have active pull-ups and pull-downs compatible with TTL loads.

Port 0 can be programmed as an I/O port.

Port 1 can be programmed as a byte I/O port.

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

Port 3 can be configured as I/O or control lines. $P3_1$ is a general purpose input or can be used for an external interrupt request signal (IRQ_2). $P3_5$ and $P3_6$ are general purpose outputs. $P3_6$ is also used for timer input (T_{IN}) and output (T_{OUT}) signals.

INTERRUPTS

The MCU allows three different interrupts from three sources, the Port 3 line $P3_1$ and the two counter/timers. These interrupts are both maskable and prioritized. The Interrupt Mask register globally or individually enables or disables the three 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 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 locations 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).

Crystal source is connected across XTAL1 and XTAL2 using the recommended capacitors (C1 ≤ 15 pf) from each pin to ground. The specifications are as follows:

- AT cut, parallel resonant
- **Fundamental type, 16 MHz maximum.**
- Series resistance, Rs ≤ 100 n

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

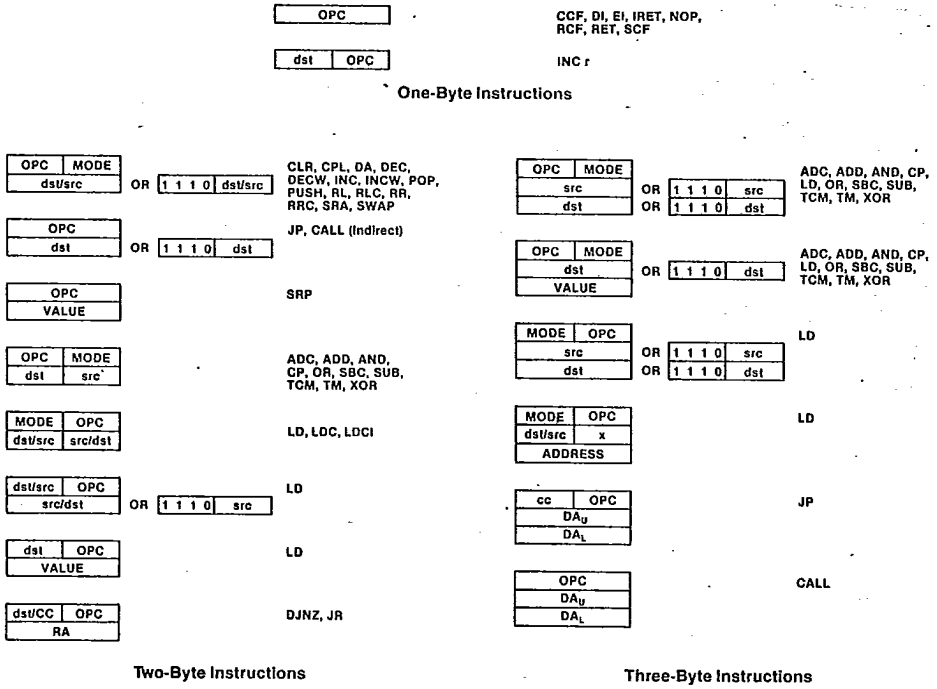


Figure 7. Instruction Formats

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	—	—

Instruction and Operation	Addr Mode dst	Addr Mode src	Opcode Byte (Hex)	Flags Affected C	Z	S	V	D	H
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	—	—	—	—	—	—

INSTRUCTION SUMMARY (Continued)

Instruction and Operation	Addr Mode dst	Addr Mode src	Opcode Byte (Hex)	Flags Affected C	Z	S	V	D	H
EI IMR (7) ← 1			9F	—	—	—	—	—	—
HALT			7F						
INC dst dst ← dst + 1	r		rE r = 0 – F	—	*	*	*	—	—
	R		20						
	IR		21						
INCW dst dst ← dst + 1	RR		A0	—	*	*	*	—	—
	IR		A1						
IRET FLAGS ← @SP; SP ← SP + 1 PC ← @SP; SP ← SP + 2; IMR (7) ← 1			BF	*	*	*	*	*	*
JP cc,dst if cc is true	DA		cD c = 0 – F	—	—	—	—	—	—
PC ← dst	IRR		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	Im	rC	—	—	—	—	—	—
	r	R	r8						
	R	r	r9 r = 0 – F						
	r	X	C7						
	X	r	D7						
	r	Ir	E3						
	Ir	r	F3						
	R	R	E4						
	R	IR	E5						
	R	IM	E6						
	IR	IM	E7						
	IR	R	F5						
LDC dst,src dst ← src	r	Irr	C2	—	—	—	—	—	—
	Irr	r	D2						
LDCI dst,src dst ← src r ← r + 1; rr ← rr + 1	Ir	Irr	C3	—	—	—	—	—	—
	Irr	Ir	D3						
LDE dst,src dst ← src	r	Irr	82	—	—	—	—	—	—
	Irr	r	92						
LDEI dst,src dst ← src r ← r + 1; rr ← rr + 1	Ir	Irr	83	—	—	—	—	—	—
	Irr	Ir	93						
NOP			FF	—	—	—	—	—	—
OR dst,src dst ← dst OR src	(Note 1)		4□	—	*	*	0	—	—
POP dst dst ← @SP; SP ← SP + 1	R		50	—	—	—	—	—	—
	IR		51						
PUSH src SP ← SP – 1; @SP ← src		R	70	—	—	—	—	—	—
		IR	71						

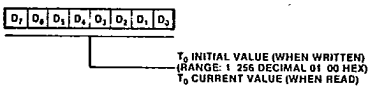
Instruction and Operation	Addr Mode dst	Addr Mode src	Opcode Byte (Hex)	Flags Affected C	Z	S	V	D	H
RCF C ← 0			CF	0	—	—	—	—	—
RET PC ← @SP; SP ← SP + 2			AF	—	—	—	—	—	—
RL dst	R		90	*	*	*	*	—	—
	IR		91						
RLC dst	R		10	*	*	*	*	—	—
	IR		11						
RR dst	R		E0	*	*	*	*	—	—
	IR		E1						
RRC dst	R		C0	*	*	*	*	—	—
	IR		C1						
SBC dst,src dst ← dst ← src ← C	(Note 1)		3□	*	*	*	*	1	*
SCF C ← 1			DF	1	—	—	—	—	—
SRA dst	R		D0	*	*	*	0	—	—
	IR		D1						
SRP src RP ← src		Im	31	—	—	—	—	—	—
STOP			6F						
SUB dst,src dst ← dst ← src	(Note 1)		2□	*	*	*	*	1	*
SWAP dst	R		F0	X	*	*	X	—	—
	IR		F1						
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: 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.

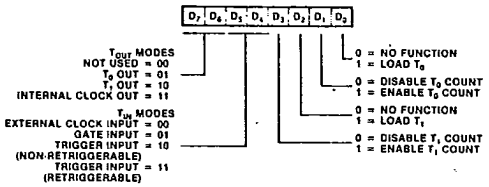
For example, the opcode of an ADC instruction using the addressing modes r (destination) and Ir (source) is 13.

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

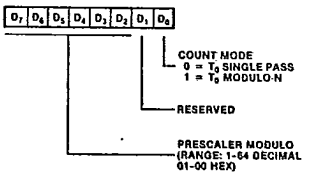
R244 T0
COUNTER/TIMER 0 REGISTER
($F4_H$; Read/Write)



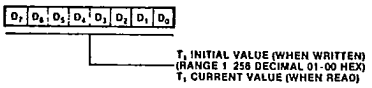
R241 TMR
TIMER MODE REGISTER
($F1_H$; Read/Write)



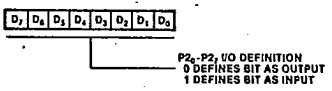
R245 PRE0
PRESCALER 0 REGISTER
($F5_H$; Write Only)



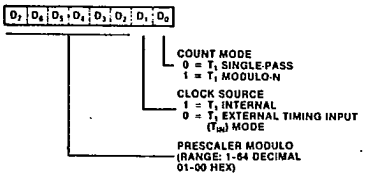
R242 T1
COUNTER TIMER 1 REGISTER
($F2_H$; Read/Write)



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)

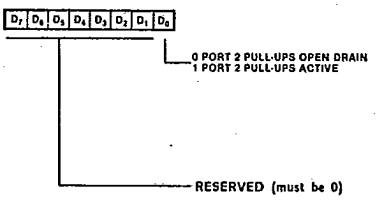
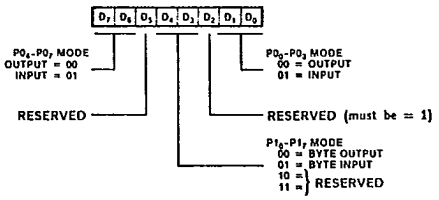


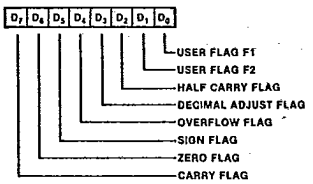
Figure 11. Control Registers

REGISTERS (Continued)

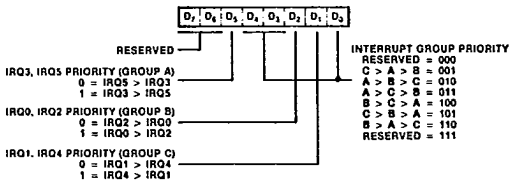
R248 P01M
PORT 0 AND 1 MODE REGISTER
($F8_H$; Write Only)



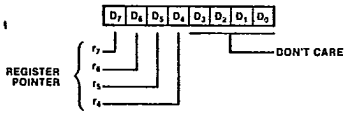
R252 FLAGS
FLAG REGISTER
(FC_H; Read/Write)



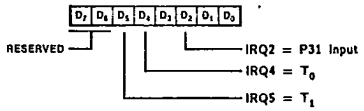
R249 IPR
INTERRUPT PRIORITY REGISTER
($F9_H$; Write Only)



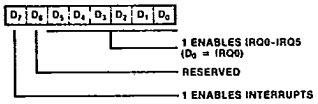
R253 RP
REGISTER POINTER
(FD_H; Read/Write)



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



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



R255 SPL
STACK POINTER
(FF_H; Read/Write)

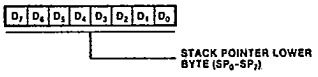
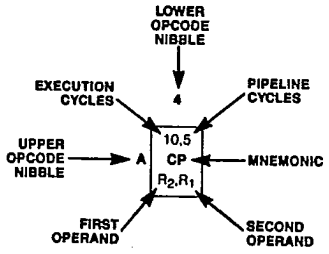


Figure 11. Control Registers (Continued)

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 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														6.1 DI
9	6.5 RL R_1	6.5 RL IR_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:

- $+4.5 \leq V_{CC} \leq +5.5$
- GND = 0V
- $0°C \leq T_A \leq +70°C$

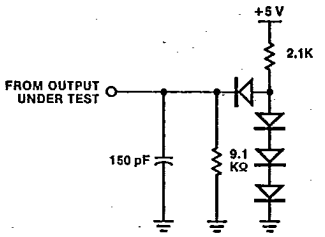


Figure 12. Test Load 1

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

NOTE:

Icc2 low power requires loading TMR (%F1) with any value prior to stop execution.
Use sequence:

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

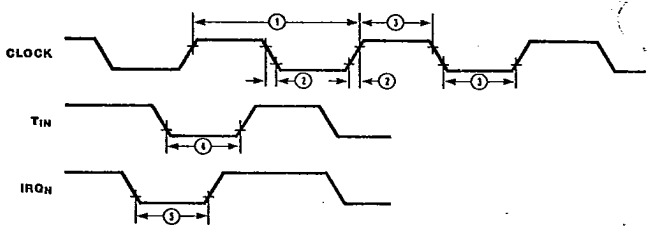


Figure 14. Additional Timing

AC CHARACTERISTICS

Additional Timing Table

Number	Symbol	Parameter	Z86C10 Min	Z86C10 Max	Notes*
1	TpC	Input Clock Period	**83**	**100,000**	1
2	TrC,TfC	Clock Input Rise and Fall Times		**15**	1
3	TwC	Input Clock Width	**70**		1
4	TwTinL	Timer Input Low Width	**70**		2
5	TwIL	Interrupt Request Input Low Time	**70**		2,3

NOTES:
1. Clock timing references use 3.8V for a logic "1" and 0.8V for a logic "0".
2. Timing references use 2.0V for a logic "1" and 0.8V for a logic "0".
3. Interrupt request via Port 3.
* Units in nanoseconds (ns).