4-BIT FLASH MICROCONTROLLER

Table of Contents--

GENERAL DESCRIPTION

The W741E20X is a high-performance 4-bit microcontroller (μC) that provides an flash EEPROM for the program memory. The device contains a 4-bit ALU, two 8-bit timers, a divider, a serial port, and five 4-bit I/O ports (including 3 output port for LED driving). There are also seven interrupt sources and 8-level subroutine nesting for interrupt applications. The W741E20X has two power reduction modes, hold mode and stop mode, which help to minimize power dissipation.

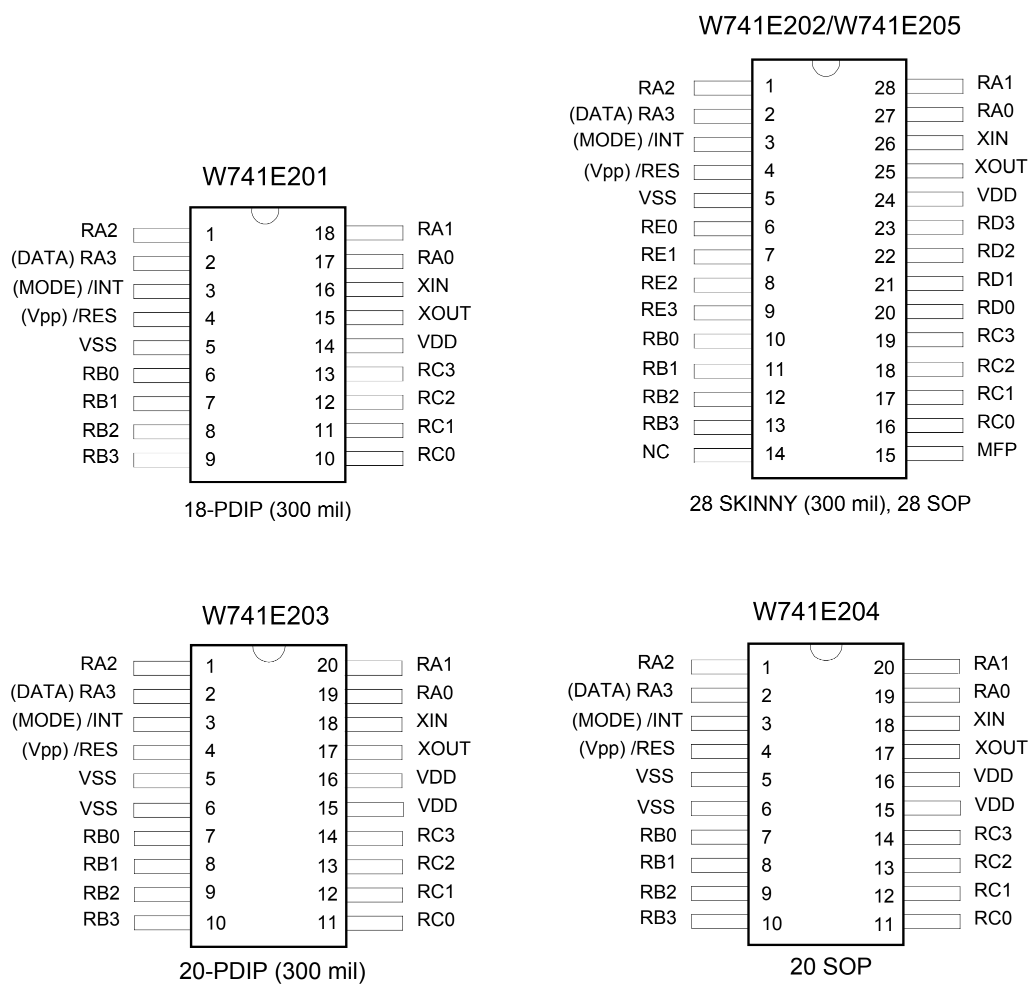
The W741E20X is suitable for end product manufacturer engineering testing and earlier samples before mass production.

FEATURES

- Operating voltage: 2.4V–5.5V
- Crystal or RC oscillation circuit can be selected by the code option
  - Crystal/Ceramic oscillator: up to 4 MHz
  - RC oscillator: up to 4 MHz
- Both in crystal or RC oscillator operation mode, high-frequency (400 KHz to 4 MHz) or low-frequency (32.768 KHz) oscillation must be determined by the code option
- Memory
  - 2048 × 16 bit program flash EEPROM (including 2K × 4 bit look-up table)
  - 128 × 4 bit data RAM (including 16 working registers)
- 21 input/output pins
  - Input/output ports: 4 ports/16 pins
  - Serial input/output port: 1 port /4 pins (high sink current for LED driving)
  - MFP output pin: 1 pin (MFP)
- Power-down mode
  - Hold function: no operation (except for oscillator)
  - Stop function: no operation (including oscillator)
- Seven types of interrupts
  - Five internal interrupts (Divider 0, Timer 0, Timer 1, and Serial I/O)
  - Two external interrupts (Port RC and $\overline{\text{INT}}$ pin)
- MFP output pin
  - Output is software selectable as modulating or nonmodulating frequency
  - Works as frequency output specified by Timer 1
- Built-in 14-bit clock frequency divider circuit
- Two built-in 8-bit programmable countdown timers
  - Timer 0: One of two internal clock frequencies (FOSC/4 or FOSC/1024) can be selected
  - Timer 1: Offers auto-reload function and one of two internal clock frequencies (FOSC or FOSC/64) can be selected or falling edge of pin RC.0 can be selected (output through MFP pin)
- Built-in 18/14-bit watchdog timer selectable for system reset
- Powerful instruction set: 118 instructions
- 8-level subroutine (include interrupt) nesting

- One serial transmission/receiver port specified by software
- Up to 1 µS instruction cycle (with 4 MHz operating frequency)
- Packaged in 18-pin, 20-pin, 28-pin PDIP and 20-pin, 28-pin SOP

PIN CONFIGURATIONS

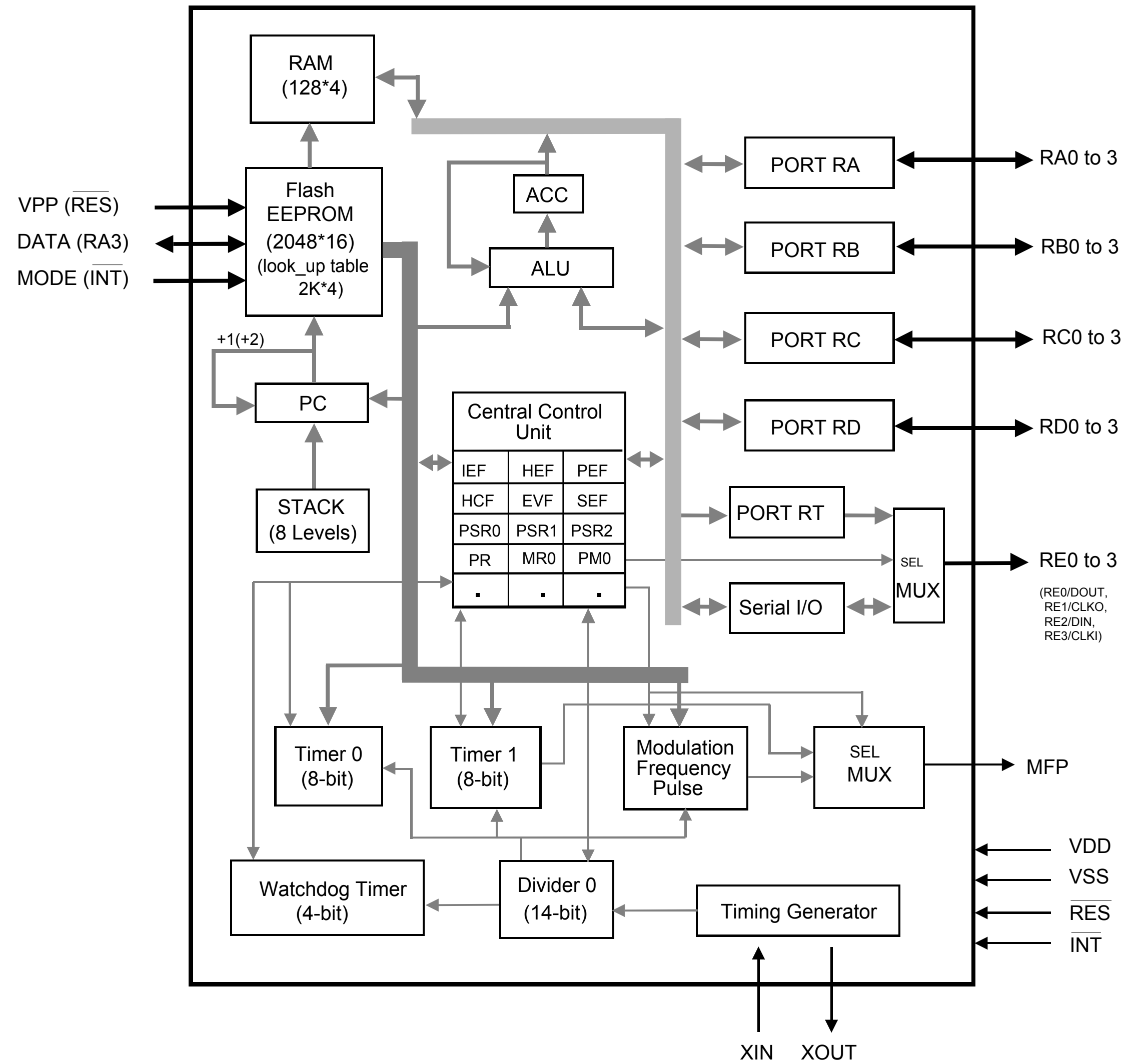


PIN DESCRIPTION

SYMBOL	I/O	FUNCTION
XIN	I	Input pin for oscillator. Connected to crystal or resistor to generate system clock by code option.
XOUT	O	Output pin for oscillator. Connected to crystal or resistor to generate system clock by code option.
RA0−RA2, RA3 (DATA)	I/O	Input/Output port. Input/output mode specified by port mode 1 register (PM1). When used as output port, can provide high sink current for driving LED.
RB0−RB3	I/O	Input/Output port. Input/output mode specified by port mode 2 register (PM2). When used as output port, can provide high sink current for driving LED.
RC0−RC3	I/O	Input/Output port. Input/output mode specified by port mode 4 register (PM4). Each pin has an independent interrupt capability in input mode.
RD0−RD3	I/O	Input/Output port. Input/output mode specified by port mode 5 register (PM5).
RE0/DOUT, RE1/CLKO, RE2/DIN, RE3/CLKI	I/O	Special input/output port. This port can be configured by software to act as the output of internal port RT or the serial I/O port. When used as output port, can provide high sink current for driving LED.
MFP	O	Output pin only. This pin can output modulating or nonmodulating frequency, or Timer 1 clock output specified by mode register 1 (MR1).
$\overline{\text{INT}}$ (MODE)[1]	I	External interrupt pin. This pin must be tied to VDD through an external resistor. It is a low active and floating input pin.
$\overline{\text{RES}}$ (VPP)[1]	I	System reset pin. This pin must be tied to VDD through an external resistor when it is not used to reset this chip. It is a low active and floating input pin.
VDD	I	Positive power supply (+).
VSS	I	Negative power supply (-).

Note: There are internal pull-high resistors in these pins of W741C20X.

BLOCK DIAGRAM



FUNCTIONAL DESCRIPTION

Program Counter (PC)

Organized as an 11-bit binary counter (PC0 to PC10), the program counter generates the addresses of the 2048 × 16 on-chip flash EEPROM containing the program instruction. When the jump or subroutine call instructions or the interrupt or initial reset conditions are to be executed, the address corresponding to the instruction will be loaded into the program counter. The format used is shown below.

ITEM	ADDRESS	INTERRUPT PRIORITY
Initial Reset	000H	-
INT 0 (Divider)	004H	1st
INT 1 (Timer 0)	008H	2nd
INT 2 (Port RC)	00CH	3rd
INT 3 ($\overline{INT}$ pin)	014H	4th
INT 4 (Serial Port Input)	018H	5th
INT 5 (Serial Port Output)	01CH	6th
INT 6 (Timer 1)	020H	7th
JMP Instruction	XXXH	-
Subroutine Call	XXXH	-

Stack Register (STACK)

The stack register is organized as 11 bits × 8 levels (first-in, last-out). When either a call subroutine or an interrupt is executed, the program counter will be pushed onto the stack register automatically. At the end of a call subroutine or an interrupt service subroutine, the RTN instruction must be executed to pop the contents of the stack register into the program counter. When the stack register is pushed over the eighth level, the contents of the first level will be lost. In other words, the stack register is always eight levels deep.

Program Memory (flash EEPROM)

The flash EEPROM is used to store program codes; the look-up table is arranged as 2048 × 4 bits. The first three quarters of flash EEPROM (000H to 5FFH) are used to store instruction codes only, but the last quarter (600H to 7FFH) can store both instruction codes and the look-up table. Each look-up table element is composed of 4 bits, so the look-up table can be addressed up to 2048 elements. There are two registers (TABL and TABH) to be used in look-up table addressing and they are controlled by MOV TABH, R and MOV TABL, R instructions. When the instruction MOVC R is executed, the contents of the look-up table location address specified by TABH, TABL and ACC will be read and transferred to the data RAM. Refer to the instruction table for more details. The organization of the program memory is shown in Figure 1.

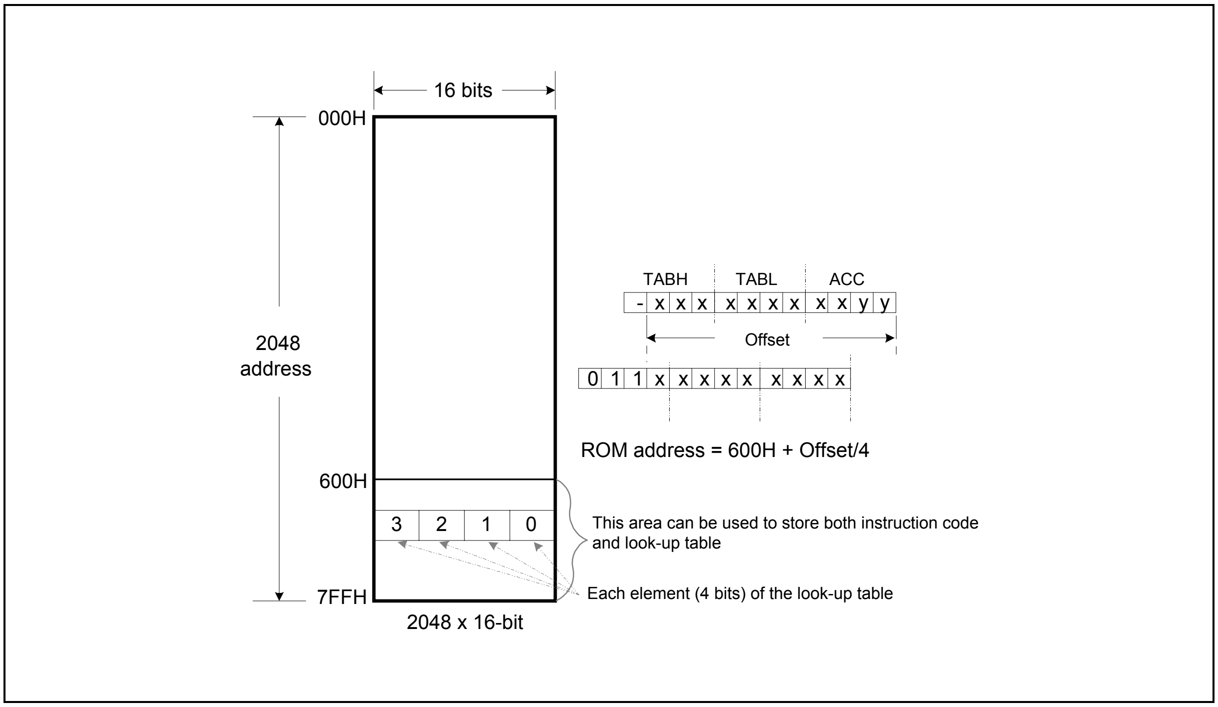


Figure 1. Program Memory Organization

Data Memory (RAM)

1. Architecture

The static data memory (RAM) used to store data is arranged as 128 × 4 bits. The data memory can be addressed directly or indirectly. The organization of the data memory is shown in Figure 2.

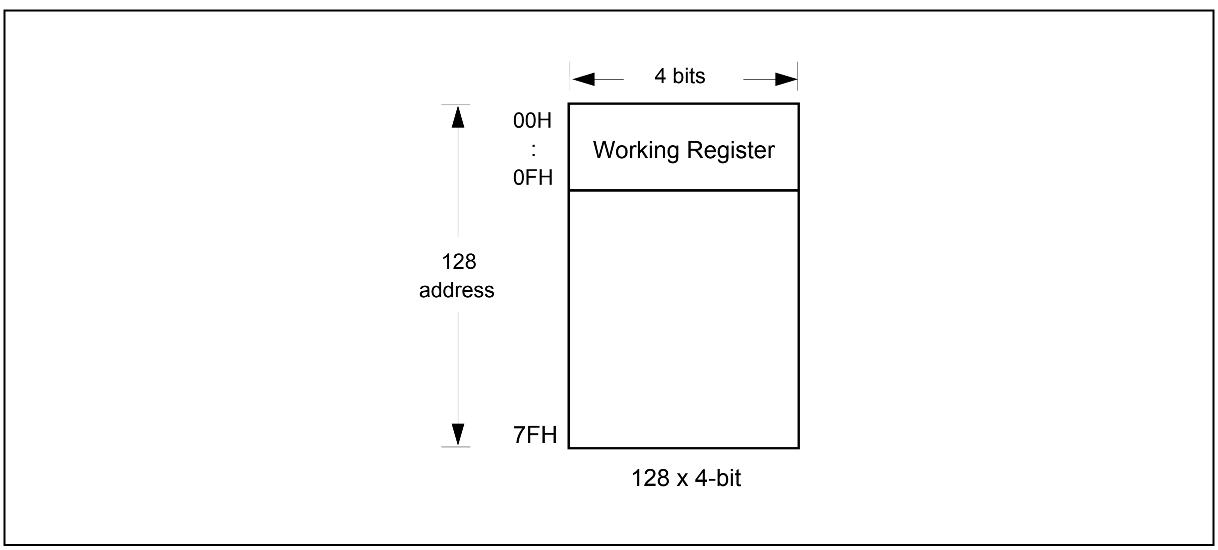


Figure 2. Data Memory Organization

The first sixteen addresses (00H to 0FH) in the data memory are known as the working registers (WR). The other data memory is used as general memory and cannot operate directly with immediate data. The relationship between data memory locations and the page register (PAGE) in indirect addressing mode is described in the next section.

2. Page Register (PAGE)

The page register is organized as a 4-bit binary register. The bit descriptions are as follows:

	3	2	1	0
PAGE	–	R/W	R/W	R/W

Note: R/W means read/write available.

Bit 3 is reserved.

Bit 2, Bit 1, Bit 0 Indirect addressing mode preselect bits:

000 = Page 0 (00H - 0FH)
001 = Page 1 (10H - 1FH)
010 = Page 2 (20H - 2FH)
011 = Page 3 (30H - 3FH)
100 = Page 4 (40H - 4FH)
101 = Page 5 (50H - 5FH)
110 = Page 6 (60H - 6FH)
111 = Page 7 (70H - 7FH)

Accumulator (ACC)

The accumulator (ACC) is a 4-bit register used to hold results from the ALU and transfer data between the memory, I/O ports, and registers.

Arithmetic and Logic Unit (ALU)

This is a circuit which performs arithmetic and logic operations. The ALU provides the following functions:

- Logic operations: ANL, XRL, ORL
- Branch decisions: JB0, JB1, JB2, JB3, JNZ, JZ, JC, JNC, DSKZ, DSKNZ, SKB0, SKB1, SKB2, SKB3
- Shift operations: SHRC, RRC, SHLC, RLC
- Binary additions/subtractions: ADC, SBC, ADD, SUB, ADU, DEC, INC

After any of the above instructions are executed, the status of the carry flag (CF) and zero flag (ZF) is stored in the internal registers. CF can be read out by executing MOVA R, CF.

Clock Generator

The W741E20X provides a crystal or RC oscillation circuit selected by option codes to generate the system clock through external connections. If a crystal oscillator is used, a crystal or a ceramic resonator must be connected to XIN and XOUT, and the capacitor must be connected if an accurate frequency is needed. When a crystal oscillator is used, a high-frequency clock (400 KHz to 4 MHz) or low-frequency clock (32 KHz) can be selected for the system clock by means of option codes. If the RC oscillator is used, a resistor in the range of 20 KΩ to 1.6 MΩ must be connected to XIN and XOUT, as shown in Figure 3. The system clock frequency range is from 32 KHz to 4 MHz. One machine cycle consists of a four-phase system clock sequence and can run up to 1 μS with a 4 MHz system clock.

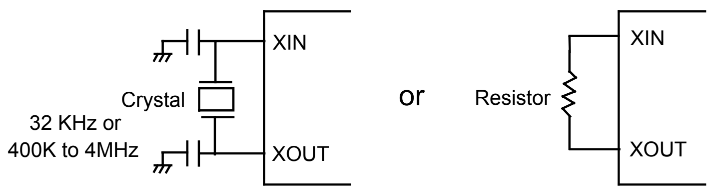


Figure 3. Oscillator Configuration

Divider 0

Divider 0 is organized as a 14-bit binary up-counter designed to generate periodic interrupts, as shown in Figure 4. When the system starts, the divider is incremented by each system clock (Fosc). When an overflow occurs, the divider event flag is set to 1 (EVF.0 = 1). Then, if the divider interrupt enable flag has been set (IEF.0 = 1), the interrupt is executed, while if the hold release enable flag has been set (HEF.0 = 1), the hold state is terminated. The last 4-stage of the Divider 0 can be reset by executing CLR DIVR0 instruction. If the oscillator is connected to the 32768 Hz crystal, the EVF.0 will be set to 1 periodically at each 500 mS interval.

Watchdog Timer (WDT)

The watchdog timer (WDT) is organized as a 4-bit up counter and is designed to protect the program from unknown errors. The WDT is enable when the corresponding option code bit of the WDT is set to 1. If the WDT overflows, the chip will be reset. At initial reset, the input clock of the WDT is FOSC/1024. The input clock of the WDT can be switched to FOSC/16384 (or FOSC/1024) by executing the SET PMF, #08H (or CLR PMF, #08H) instruction. The contents of the WDT can be reset by the instruction CLR WDT. In normal operation, the application program must reset WDT before it overflows. A WDT overflow indicates that the operation is not under control and the chip will be reset. The WDT minimun overflow period is 468.75 mS when the system clock (FOSC) is 32 KHz and WDT clock input is FOSC/1024. When the corresponding option code bit of the WDT is set to 0, the WDT function is disabled. The organization of the Divider0 and watchdog timer is shown in Figure 4.

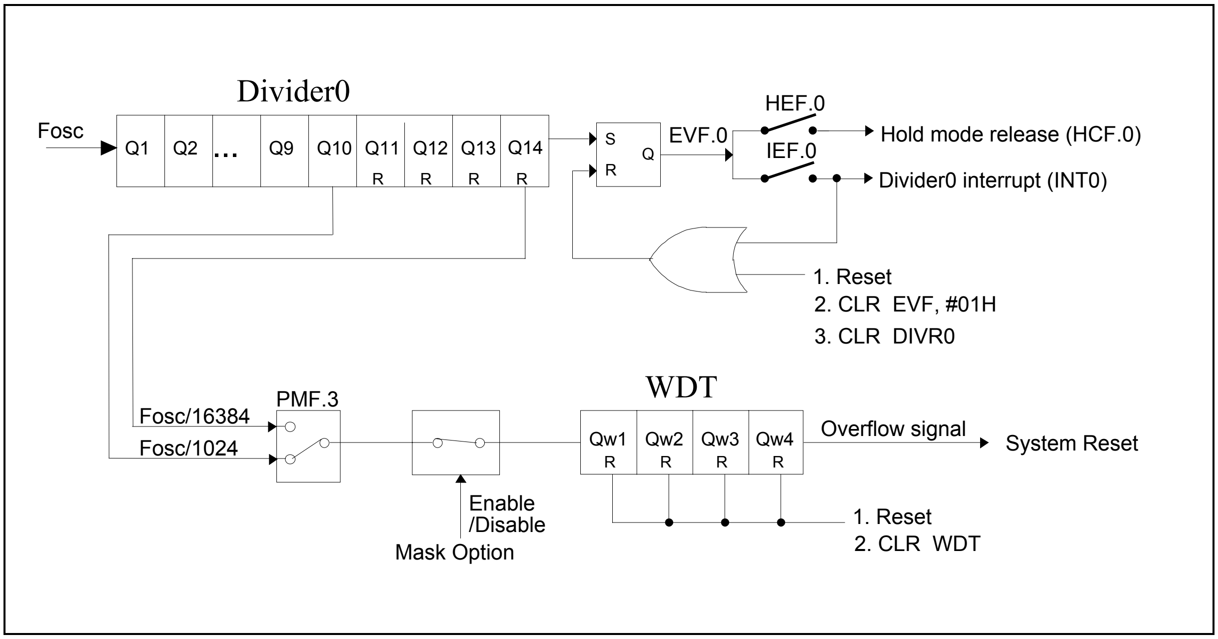


Figure 4. Organization of Divider and Watchdog Timer

Parameter Flag (PMF)

The parameter flag is organized as a 4-bit binary register (PMF.0 to PMF.3). The PMF is controlled by the SET PMF, #I or CLR PMF, #I instruction. The bit descriptions are as follows:

	3	2	1	0
PMF	W	—	—	—

Note: W means write only.

Bit 0, Bit 1 & Bit 2 are reserved.

Bit 3 = 0 The fundamental frequency of the watch dog timer is Fosc/1024.

= 1 The fundamental frequency of the watch dog timer is Fosc/16384.

At initial reset, bit 3 of PMF is set to "0".

Timer/Counter

Timer 0 (TM0)

Timer 0 (TM0) is a programmable 8-bit binary down-counter. The specified value can be loaded into TM0 by executing the MOV TM0L(TM0H), R or MOV TM0, #I instruction. When the MOV TM0L (TM0H), R instructions are executed, the TM0 will stop down-counting (if the TM0 is down-counting), the MR0.3 will be reset to 0, and the specified value is loaded into TM0. If MR0.3 is set to 1, the event flag 1 (EVF.1) is reset and the TM0 starts to count. When it decrements to FFH, Timer 0 stops operating and generates an underflow (EVF.1 = 1). The interrupt is executed if the Timer 0 interrupt enable flag has been set (IEF.1 = 1); and the hold state is terminated if the hold release enable flag 1 has been set (HEF.1 = 1). The Timer 0 clock input can be set as FOSC/1024 or FOSC/4 by setting MR0.0 to 1 or by resetting MR0.0 to 0. The default timer value is FOSC/4. The organization of Timer 0 is shown in Figure 5.

If the Timer 0 clock input is Fosc/4, then:

Desired time 0 interval = (preset value +1) × 4 × 1/Fosc

If the Timer 0 clock input is Fosc/1024, then:

Desired time 0 interval = (preset value +1) × 1024 × 1/Fosc

Preset value: Decimal number of Timer 0 preset value

Fosc: Clock oscillation frequency

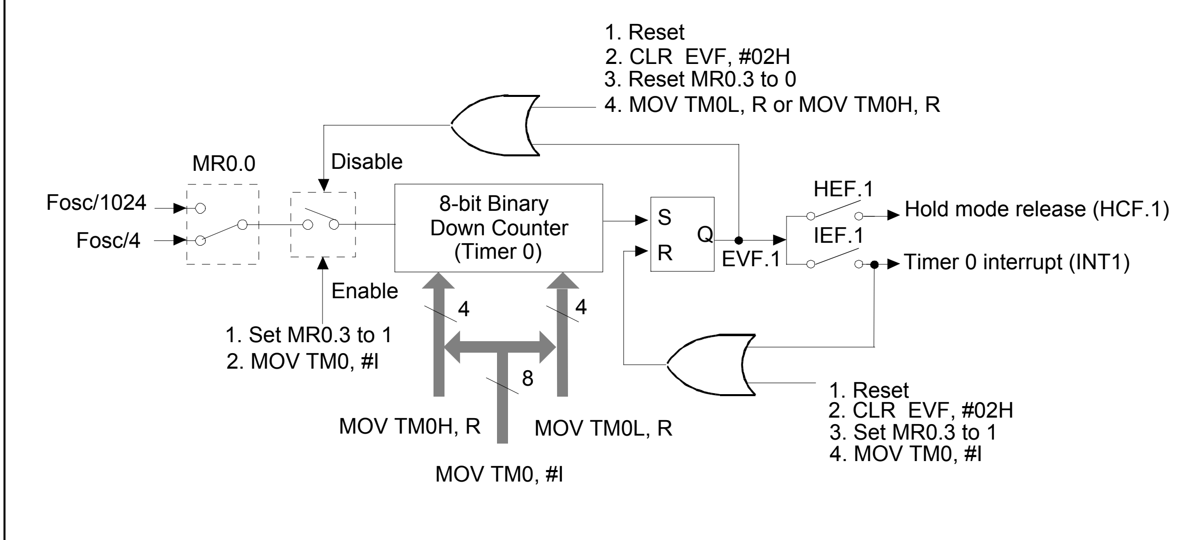


Figure 5. Organization of Timer 0

Timer 1 (TM1)

Timer 1 (TM1) is also a programmable 8-bit binary down counter, as shown in Figure 6. Timer 1 can be used as a counter to count external events or to output an arbitrary frequency to the MFP pin. The input clock of Timer 1 can be one of three sources: Fosc/64, Fosc, or an external clock from the RC.0

input pin. The source can be selected by setting bit 0 and bit 1 of mode register 1 (MR1). At initial reset, the Timer 1 clock input is Fosc. If an external clock is selected as the clock source of Timer 1, the content of Timer 1 is decreased by 1 at the falling edge of RC.0. When the MOV TM1L, R or MOV TM1H,R instruction is executed, the specified data are loaded into the auto-reload buffer and the TM1 down-counting will be disabled (i.e. MR1.3 is reset to 0). If the bit 3 of MR1 is set (MR1.3 = 1), the contents of the auto-reload buffer will be loaded into the TM1 down counter, Timer 1 starts to down count, and the event flag 7 is reset (EVF.7 = 0). When the MOV TM1, #I instruction is executed, the event flag 7 (EVF.7) and MR1.3 are reset and the specified value is loaded into auto-reload buffer and TM1 by the internal hardware, then the MR1.3 is set, that is the TM1 starts to count by the hardware. When the timer decrements to FFH, it will generate an underflow (EVF.7 = 1) and be auto-reloaded with the specified data, after which it will continue to count down. An interrupt is executed if the interrupt enable flag 7 has been set to 1 (IEF.7 = 1), and the hold state is terminated if the hold mode release enable flag 7 is set to 1 (HEF.7 = 1). The specified frequency of Timer 1 can be delivered to the MFP output pin by programming bit 2 of MR1. Bit 3 of MR1 can be used to make Timer 1 stop or start counting.

If the Timer 1 clock input is F_T, then:

Desired Timer 1 interval = (preset value +1) / F_T

Desired frequency for MFP output pin = $F_T \div$ (preset value + 1) $\div$ 2 (Hz)

Preset value: Decimal number of Timer 1 preset value, and

Fosc: Clock oscillation frequency

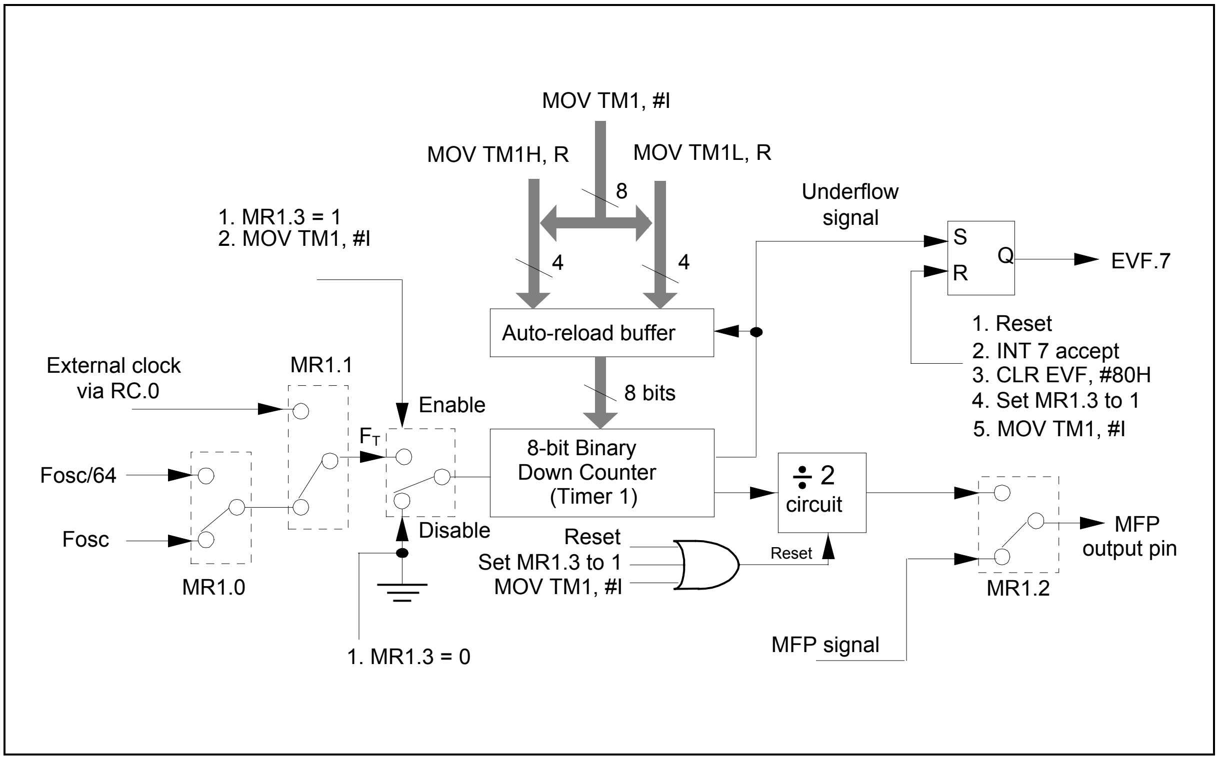


Figure 6. Organization of Timer 1

For example, when FT equals 32768 Hz, depending on the preset value of TM1, the MFP pin will output a single tone signal in the tone frequency range from 64 Hz to 16384 Hz. The relation between the tone frequency and the preset value of TM1 is shown in the table below.

		3			4			5		
		Tone frequency	TM1 preset value & MFP frequency		Tone frequency	TM1 preset value & MFP frequency		Tone frequency	TM1 preset value & MFP frequency	
	C	130.81	7CH	131.07	261.63	3EH	260.06	523.25	1EH	528.51
	C#	138.59	75H	138.84	277.18	3AH	277.69	554.37	1CH	564.96
T	**D**	146.83	6FH	146.28	293.66	37H	292.57	587.33	1BH	585.14
	D#	155.56	68H	156.03	311.13	34H	309.13	622.25	19H	630.15
O	**E**	164.81	62H	165.49	329.63	31H	327.68	659.26	18H	655.36
	F	174.61	5DH	174.30	349.23	2EH	372.36	698.46	16H	712.34
N	**F#**	185.00	58H	184.09	369.99	2BH	390.09	739.99	15H	744.72
	G	196.00	53H	195.04	392.00	29H	420.10	783.99	14H	780.19
E	**G#**	207.65	4EH	207.39	415.30	26H	443.81	830.61	13H	819.20
	A	220.00	49H	221.40	**440.00**	**24H**	**442.81**	880.00	12H	862.84
	A#	233.08	45H	234.05	466.16	22H	468.11	932.23	11H	910.22
	B	246.94	41H	248.24	493.88	20H	496.48	987.77	10H	963.76

Note: Central tone is A4 (440 Hz).

Mode Register 0 (MR0)

Mode Register 0 is organized as a 4-bit binary register (MR0.0 to MR0.3). MR0 can be used to control the operation of Timer 0. The bit descriptions are as follows:

	3	2	1	0
MR0	W	–	–	W

Note: W means write only.

Bit 0 = 0 The fundamental frequency of Timer 0 is FOSC/4.

= 1 The fundamental frequency of Timer 0 is FOSC/1024.

Bit 1 & Bit 2 are reserved

Bit 3 = 0 Timer 0 stops down-counting.

= 1 Timer 0 starts down-counting.

Mode Register 1 (MR1)

Mode Register 1 is organized as a 4-bit binary register (MR1.0 to MR1.3). MR1 can be used to control the operation of Timer 1. The bit descriptions are as follows:

	3	2	1	0
MR1	W	W	W	W

Note: W means write only.

Bit 0 = 0 The internal fundamental frequency of Timer 1 is Fosc.
= 1 The internal fundamental frequency of Timer 1 is Fosc/64.

Bit 1 = 0 The fundamental frequency source of Timer 1 is the internal clock.
= 1 The fundamental frequency source of Timer 1 is the external clock from RC.0 input pin.

Bit 2 = 0 The specified waveform of the MFP generator is delivered at the MFP output pin.
= 1 The specified frequency of Timer 1 is delivered at the MFP output pin.

Bit 3 = 0 Timer 1 stops down-counting.
= 1 Timer 1 starts down-counting.

Input/Output Ports RA, RB

Port RA consists of pins RA.0 to RA.3 and Port RB consists of pins RB.0 to RB.3. At initial reset, input/output ports RA and RB are both in input mode. When RA and RB are used as output ports, CMOS or NMOS open drain output type can be selected by the PM0 register. Each pin of port RA or RB can be specified as input or output mode independently by the PM1 and PM2 registers. The MOVA R, RA or MOVA R, RB instructions operate the input functions and the MOV RA, R or MOV RB, R operate the output functions. For more details, refer to the instruction table and Figure 7.

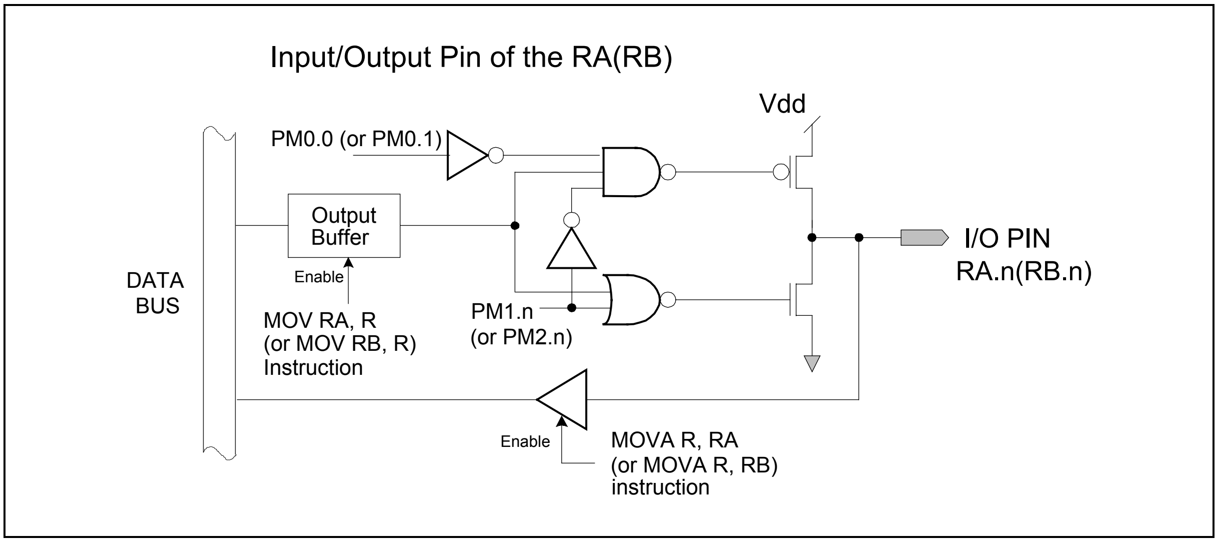


Figure 7. Architecture of RA & RB Input/Output Pins

Port Mode 0 Register (PM0)

The port mode 0 register is organized as 4-bit binary register (PM0.0 to PM0.3). PM0 can be used to determine the structure of the input/output ports; it is controlled by the MOV PM0, #I instruction. The bit descriptions are as follows:

	3	2	1	0
PM0	–	–	w	w

Note: W means write only.

Bit 0 = 0 RA port is CMOS output type. Bit 0 = 1 RA port is NMOS open drain output type.

Bit 1 = 0 RB port is CMOS output type. Bit 0 = 1 RB port is NMOS open drain output type.

Bit 2 & Bit 3 are reserved.

Port Mode 1 Register (PM1)

The port mode 1 register is organized as 4-bit binary register (PM1.0 to PM1.3). PM1 can be used to control the input/output mode of port RA. PM1 is controlled by the MOV PM1, #I instruction. The bit descriptions are as follows:

	3	2	1	0
PM1	w	w	w	w

Note: W means write only.

Bit 0 = 0 RA.0 works as output pin; Bit 0 = 1 RA.0 works as input pin

Bit 1 = 0 RA.1 works as output pin; Bit 1 = 1 RA.1 works as input pin

Bit 2 = 0 RA.2 works as output pin; Bit 2 = 1 RA.2 works as input pin

Bit 3 = 0 RA.3 works as output pin; Bit 3 = 1 RA.3 works as input pin

At initial reset, port RA is input mode (PM1 = 1111B).

Port Mode 2 Register (PM2)

The port mode 2 register is organized as 4-bit binary register (PM2.0 to PM2.3). PM2 can be used to control the input/output mode of port RB. PM2 is controlled by the MOV PM2, #I instruction. The bit descriptions are as follows:

	3	2	1	0
PM2	w	w	w	w

Note: W means write only.

Bit 0 = 0 RB.0 works as output pin; Bit 0 = 1 RB.0 works as input pin

Bit 1 = 0 RB.1 works as output pin; Bit 1 = 1 RB.1 works as input pin

Bit 2 = 0 RB.2 works as output pin; Bit 2 = 1 RB.2 works as input pin

Bit 3 = 0 RB.3 works as output pin; Bit 3 = 1 RB.3 works as input pin

At initial reset, the port RB is input mode (PM2 = 1111B).

Port Mode 3 Register (PM3)

Port Mode 3 Register is organized as a 4-bit binary register (PM3.0 to PM3.3). PM3 can be used to determine the operating mode of the output port RE and the clock rate of the serial I/O function. The PM3 control diagram is shown in Figure 8. The bit descriptions are as follows:

	3	2	1	0
PM3	W	—	W	—

Note: W means write only.

Bit 0 is reserved.

Bit 1 = 0 The output of the port RE is the output of the internal parallel port RT.

= 1 The port RE works as the serial input/output port.

Bit 2 is reserved.

Bit 3 = 0 Serial Tx rate = Fosc/2

= 1 Serial Tx rate = Fosc/256

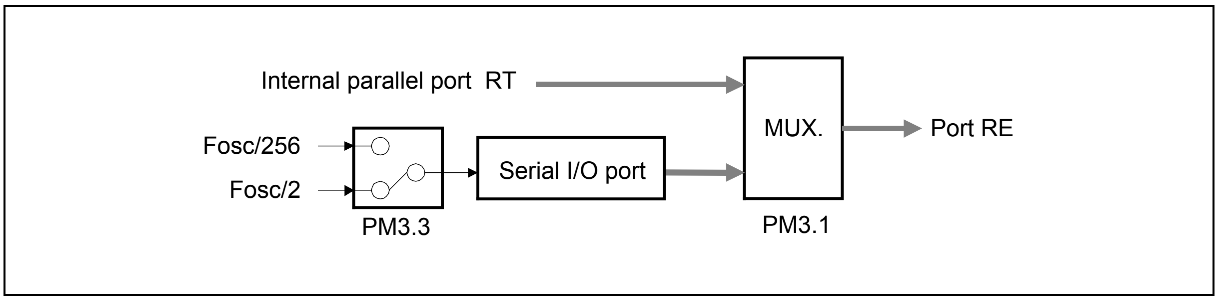


Figure 8. PM3 Control Diagram

Port Mode 4 Register (PM4)

The port mode 4 register is organized as 4-bit binary register (PM4.0 to PM4.3). PM4 can be used to control the input/output mode of port RC. PM4 is controlled by the MOV PM4, #I instruction. The bit descriptions are as follows:

	3	2	1	0
PM4	w	w	w	w

Note: W means write only.

Bit 0 = 0 RC.0 works as output pin; Bit 0 = 1 RC.0 works as input pin

Bit 1 = 0 RC.1 works as output pin; Bit 1 = 1 RC.1 works as input pin

Bit 2 = 0 RC.2 works as output pin; Bit 2 = 1 RC.2 works as input pin

Bit 3 = 0 RC.3 works as output pin; Bit 3 = 1 RC.3 works as input pin

At initial reset, port RC is input mode (PM4 = 1111B).

Port Mode 5 Register (PM5)

The port mode 5 register is organized as 4-bit binary register (PM5.0 to PM5.3). PM5 can be used to control the input/output mode of port RD. PM5 is controlled by the MOV PM5, #I instruction. The bit descriptions are as follows:

	3	2	1	0
PM5	w	w	w	w

Note: W means write only.

Bit 0 = 0 RD.0 works as output pin; Bit 0 = 1 RD.0 works as input pin

Bit 1 = 0 RD.1 works as output pin; Bit 1 = 1 RD.1 works as input pin

Bit 2 = 0 RD.2 works as output pin; Bit 2 = 1 RD.2 works as input pin

Bit 3 = 0 RD.3 works as output pin; Bit 3 = 1 RD.3 works as input pin

At initial reset, the port RB is input mode (PM2 = 1111B).

Input/Output Ports RC, RD

Port RC consists of pins RC.0 to RC.3, and port RD consists of pins RD.0 to RD.3. At initial reset, input/output ports RC and RD are both in input mode. When RC and RD are used as output ports, the CMOS type is the only ouput driving type. Each pin of port RC or RD can be specified as input or output mode independently by the PM4 and PM5 registers. The MOVA R, RC or MOVA R, RD instructions operate the input functions and the MOV RC, R or MOV RD, R operate the output functions. When the PEF, HEF, and IEF corresponding to the RC port are set, a signal change at the specified pins of port RC will execute the hold mode release or interrupt subroutine. Port status register 0 (PSR0) records the status of port RC, and that can be read out and cleared by the MOV R, PSR0, and CLR PSR0 instructions. Before the port mode of the RC port is changed from output mode to input mode in the hold mode release and interrupt application, the output value must be preset to the same as the system status to prevent the undesired signal change being accepted. When the interrupt of RC port is accepted, the corresponding event flag (EVF.2) will be reset, but the content of PSR0 should not be changed except the CLR PSR0 or MOV PEF,#I instruction being executed or performing the reset function. In addition, the falling edge signal on the pin of port RC specified by the instruction MOV SEF, #I will cause the device to exit the stop mode. The RD port is used as the I/O port only. Refer to Figure 9, Figure 10 and the instruction table for more details.

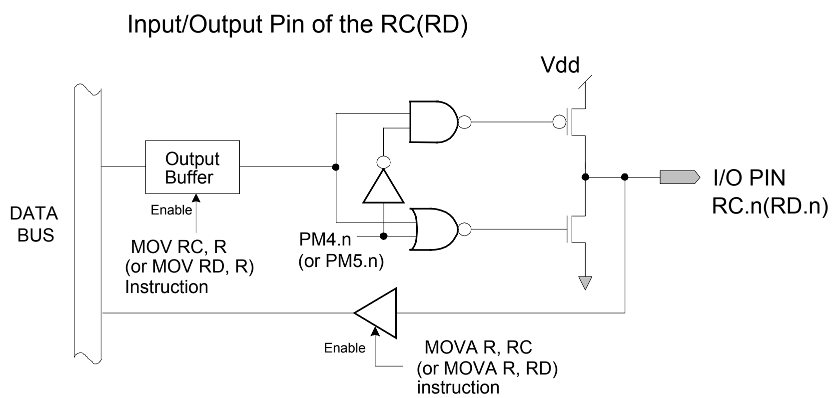


Figure 9. Architecture of RC & RD Input/Output Pins

Port Enable Flag (PEF)

The port enable flag is organized as 4-bit binary register (PEF.0 to PEF.3). Before port RC may be used to release the hold mode or preform interrupt function, the content of the PEF must be set first. The PEF is controlled by the MOV PEF, #I instruction. The bit descriptions are as follows:

	3	2	1	0
PEF	W	W	W	W

Note: W means write only.

PEF.0: Enable/disable the signal change at pin RC.0 to release hold mode or perform interrupt.

PEF.1: Enable/disable the signal change at pin RC.1 to release hold mode or perform interrupt.

PEF.2: Enable/disable the signal change at pin RC.2 to release hold mode or perform interrupt.

PEF.3: Enable/disable the signal change at pin RC.3 to release hold mode or perform interrupt.

Port Status Register 0 (PSR0)

Port status register 0 is organized as 4-bit binary register (PSR0.0 to PSR0.3). PSR0 can be read or cleared by the MOVA R, PSR0, and CLR PSR0 instructions. The bit descriptions are as follows:

	3	2	1	0
PSR0	R	R	R	R

Note: R means read only.

Bit 0 = 1   Signal change at RC.0

Bit 1 = 1   Signal change at RC.1

Bit 2 = 1   Signal change at RC.2

Bit 3 = 1   Signal change at RC.3

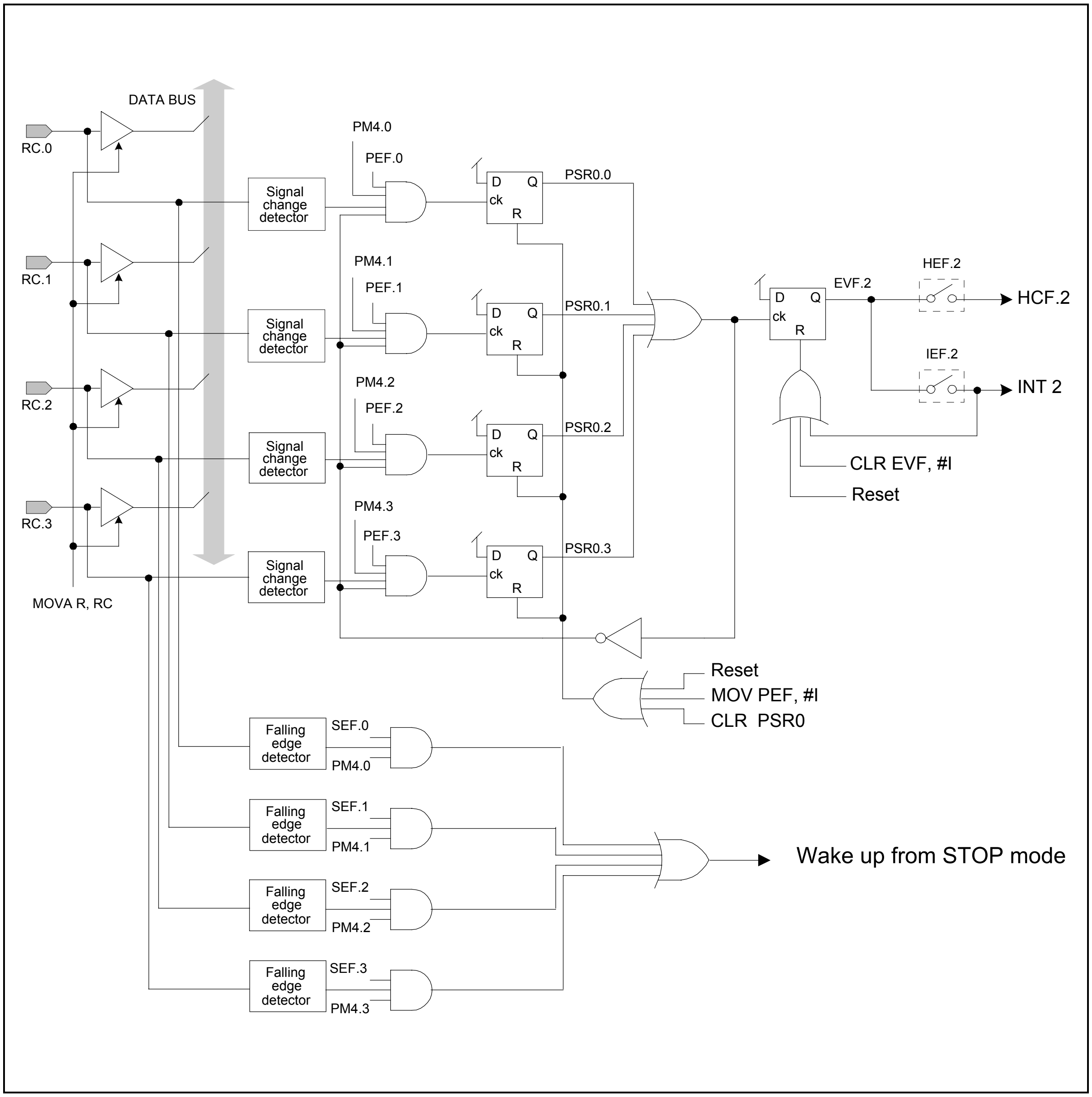
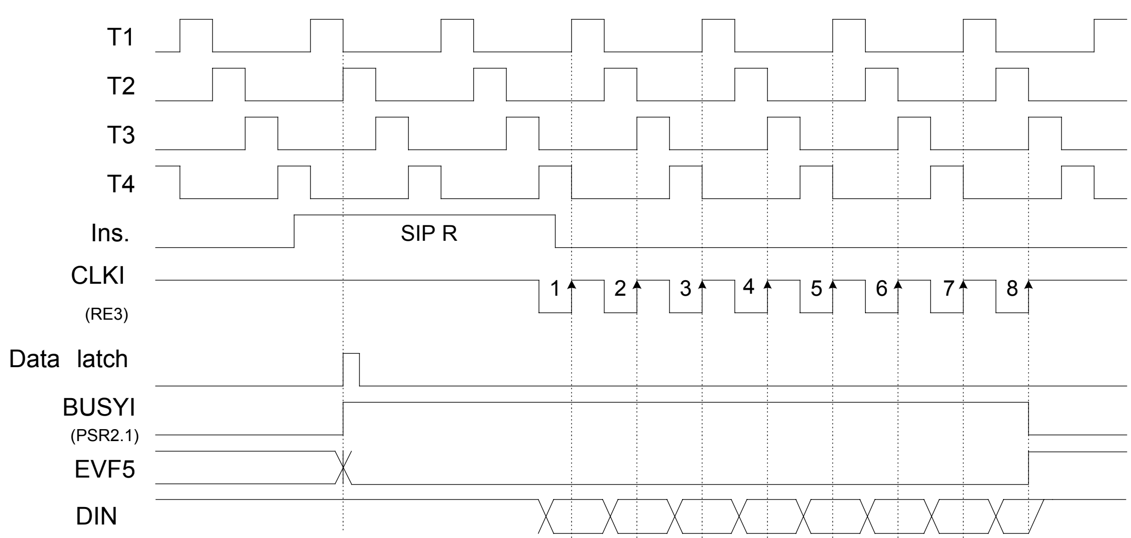


Figure 10. Input Architecture of Ports RC

Output Port RE

Output port RE can be used as an output of the internal RT port, or as a serial input/output port. The control flow is shown in Figure 8. When bit 1 of port mode 3 register (PM3) equals to 0, port RE works as an output of internal port RT. When the MOV RE, R instruction is executed, the data in the RAM will be output to port RT through port RE. When RE works as a parallel output port, it provides a high sink current to drive LEDs. When bit 1 of MR0 equals to 1, the RE port works as a serial input/output port, and RE.0 to RE.3 are used as DOUT, CLKO, DIN, and CLKI, respectively. In this case, the DIN pin will has a built-in pull-high resistor. The serial I/O functions are controlled by the instructions SOP R and SIP R. The functions of the two instructions are described below:

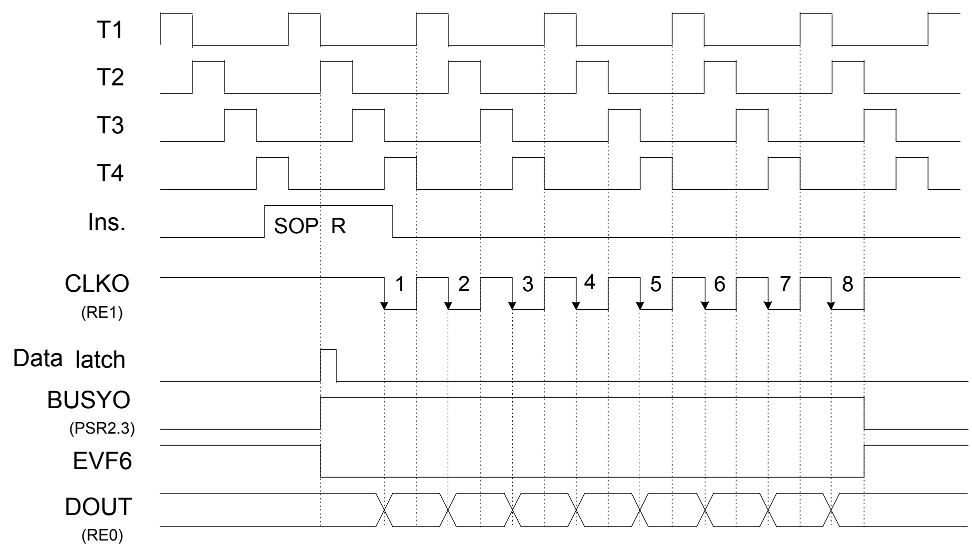
(1) When the SIP R instruction is executed, the data will be loaded from the serial input buffer to the ACC and RAM first, and bit 1 of port status register 2 will automatically be set to "1" (BUSYI = 1). Then the CLKI pin will send out 8 clocks and the data from the DIN pin will be loaded to SIB at the rising edge of the CLKI pin. After the 8 clocks have been sent, BUSYI will be reset to "0" and EVF.5 will be set to "1." At this time, if IEF.5 has been set (IEF.5 = 1), an interrupt is executed; if HEF.5 has been set (HEF.5 = 1), the hold state is terminated. Users can check the status of PSR2.1 (BUSYI) to know whether the serial input process is completed or not. If a serial input process is not completed, and the SIP R instruction is executed again, the data will be lost. The timing is shown in Figure 11.



Notes : 1. These clocks at the CLKI pin are internal clock and its frequency is Fosc/2.
2. When the internal signal of the data latch equals to "1," then the data in SIB will be loaded into RAM and ACC.

Figure 11. Timing of the Serial Input Function (SIP R)

(2) When the SOP R instruction is executed, the data will be loaded to the serial output buffer (SOB) and bit 3 of port status register 2 will be set to "1" (BUSYO = 1). Then the CLKO pin will send out 8 clocks and the data in SOB will be sent out at the falling edge of the CLKO pin. After the 8 clocks have been sent, BUSYO will be reset to "0" and EVF.6 will be set to "1." At this time, if IEF.6 has been set (IEF.6 = 1), an interrupt is executed; if HEF.6 has been set (HEF.6 = 1), the hold state is terminated. Users can check the status of PSR2.3 (BUSYO) to know whether the serial output process is completed or not. If a serial output process is not completed, and the SOP R instruction is executed again, the data will be lost. The timing is shown in Figure 12.



Notes : 1. These clocks at the CLKO pin are internal clock and its frequency is Fosc/2.
2. When the internal signal of the data latch equals to "1," then the data of the RAM and ACC be loaded to SOB.

Figure 12. Timing of the Serial Output Function (SOP R)

In the above description, the low nibble location of the serial input/output register is contributed to the ACC, and the high nibble is to R. The port status register 2 (PSR2) including BUSYI, and BUSYO can be read out or cleared by the MOVA R, PSR2, or CLR PSR2 instruction.

Port Status Register 2 (PSR2)

Port status register 2 is organized as 4-bit binary register (PSR2.0 to PSR2.3). PSR2 is controlled by the MOVA R, PSR2, and CLR PSR2 instructions. The bit descriptions are as follows:

	3	2	1	0
PSR2	R	–	R	–

Note: R means read only.

Bit 0 is reserved.

Bit 1 (BUSYI): Serial port input busy flag.

Bit 2 is reserved.

Bit 3 (BUSYO): Serial port output busy flag.

MFP Output Pin (MFP)

The MFP output pin can output the Timer 1 clock or the modulation frequency; the output of the pin is determined by mode register 1 (MR1). The organization of MR1 is shown in Figure 6. When bit 2 of MR1 is reset to "0," the MFP output can deliver a modulation output in any combination of one signal from among DC, 4096Hz, 2048Hz, and one or more signals from among 128 Hz, 64 Hz, 8 Hz, 4 Hz, 2 Hz, or 1 Hz (when using a 32.768 KHz crystal). The MOV MFP, #I instruction is used to specify the modulation output combination. The data specified by the 8-bit operand and the MFP output pin are shown as below.

(Fosc = 32.768 KHz)

R7 R6	R5	R4	R3	R2	R1	R0	FUNCTION
0 0	0	0	0	0	0	0	Low level
	0	0	0	0	0	1	128 Hz
	0	0	0	0	1	0	64 Hz
	0	0	0	1	0	0	8 Hz
	0	0	1	0	0	0	4 Hz
	0	1	0	0	0	0	2 Hz
	1	0	0	0	0	0	1 Hz
0 1	0	0	0	0	0	0	High level
	0	0	0	0	0	1	128 Hz
	0	0	0	0	1	0	64 Hz
	0	0	0	1	0	0	8 Hz
	0	0	1	0	0	0	4 Hz
	0	1	0	0	0	0	2 Hz
	1	0	0	0	0	0	1 Hz
1 0	0	0	0	0	0	0	2048 Hz
	0	0	0	0	0	1	2048 Hz * 128 Hz
	0	0	0	0	1	0	2048 Hz * 64 Hz
	0	0	0	1	0	0	2048 Hz * 8 Hz
	0	0	1	0	0	0	2048 Hz * 4 Hz
	0	1	0	0	0	0	2048 Hz * 2 Hz
	1	0	0	0	0	0	2048 Hz * 1 Hz

Continued

R7 R6	R5	R4	R3	R2	R1	R0	FUNCTION
1 1	0	0	0	0	0	0	4096 Hz
	0	0	0	0	0	1	4096 Hz * 128 Hz
	0	0	0	0	1	0	4096 Hz * 64 Hz
	0	0	0	1	0	0	4096 Hz * 8 Hz
	0	0	1	0	0	0	4096 Hz * 4 Hz
	0	1	0	0	0	0	4096 Hz * 2 Hz
	1	0	0	0	0	0	4096 Hz * 1 Hz

Interrupts

The W741E20X provides five internal interrupt sources (Divider 0, Timer 0, Timer 1, serial I/O) and two external interrupt sources ($\overline{\text{INT}}$, port RC). Vector addresses for each of the interrupts are located in the range of program memory (ROM) addresses 004H to 020H. The flags IEF, PEF, and EVF are used to control the interrupts. When EVF is set to "1" by hardware and the corresponding bits of IEF and PEF have been set by software, an interrupt is generated. When an interrupt occurs, all of the interrupts are inhibited until the EN INT or MOV IEF, #I instruction is invoked. The interrupts can also be disabled by executing the DIS INT instruction. When an interrupt is generated in hold mode, the hold mode will be released momentarily and interrupt subroutine will be executed. After the RTN instruction is executed in an interrupt subroutine, the µC will enter hold mode again. The operation flow chart is shown in Figure 14. The control diagram is shown below.

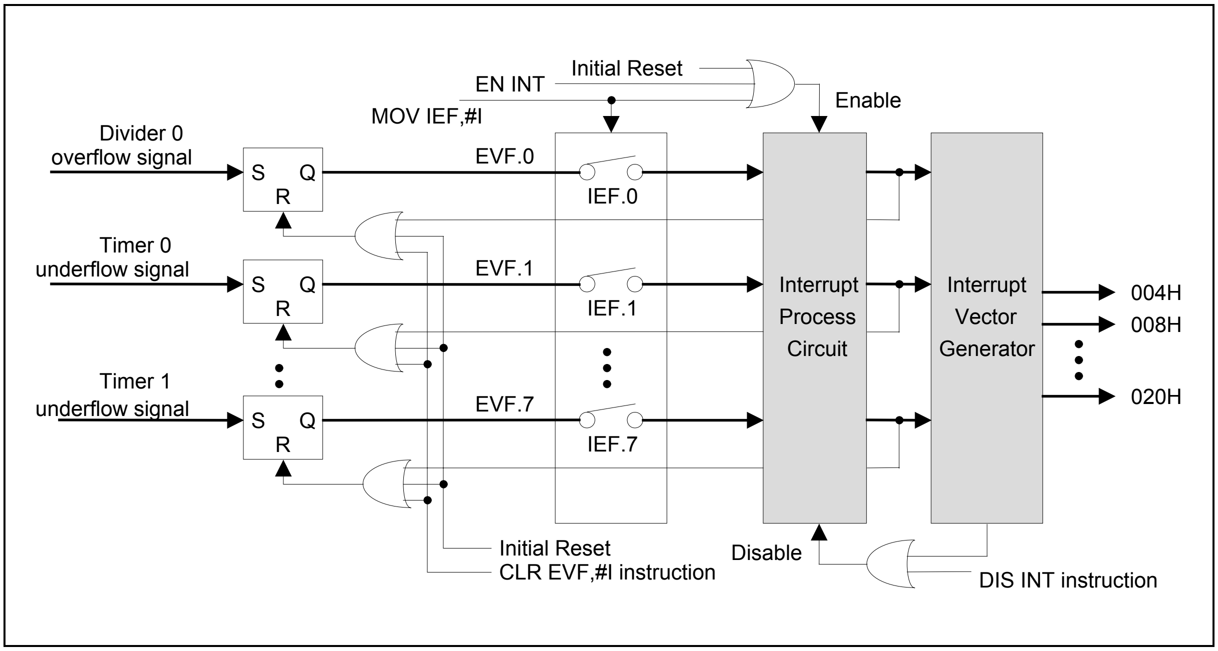


Figure 13. Interrupt event control diagram

Interrupt Enable Flag (IEF)

The interrupt enable flag is organized as an 8-bit binary register (IEF.0 to IEF.7). These bits are used to control the interrupt conditions. It is controlled by the MOV IEF, #I instruction. When one of these interrupts is accepted, the corresponding to the bit of the event flag will be reset, but the other bits are unaffected. In interrupt subroutine, these interrupts will be disable till the instruction MOV IEF, #I or EN INT is executed again. To enable these interrupts, the instructions MOV IEF, #I or EN INT must be executed again. Otherwise, these interrupts can be disable by executing DIS INT instruction. The bit descriptions are as follows:

	7	6	5	4	3	2	1	0
IEF	w	w	w	w	—	w	w	w

Note: W means write only.

IEF.0 = 1 Interrupt 0 is accepted by overflow from the Divider 0.

IEF.1 = 1 Interrupt 1 is accepted by underflow from the Timer 0.

IEF.2 = 1 Interrupt 2 is accepted by a signal change at port RC.

IEF.3 is reserved.

IEF.4 = 1 Interrupt 4 is accepted by a falling edge signal at the $\overline{\text{INT}}$ pin.

IEF.5 = 1 Interrupt 5 is accepted by the serial port received completely.

IEF.6 = 1 Interrupt 6 is accepted by the serial port transmitted completely.

IEF.7 = 1 Interrupt 7 is accepted by underflow from Timer 1.

External INT

The external interrupt $\overline{\text{INT}}$ pin contains a pull-up resistor. When the HEF.4 or IEF.4 flag is set, the falling edge of the $\overline{\text{INT}}$ pin will execute the hold mode release or interrupt subroutine. A low level on the $\overline{\text{INT}}$ pin will release the stop mode.

Stop Mode Operation

In stop mode, all operations of the µC cease (including the operation of the oscillator). The µC enters stop mode when the STOP instruction is executed and exits stop mode when an external trigger is activated (by a low level on the $\overline{\text{INT}}$ pin or a falling signal on the RC port). When the designated signal is accepted, the µC awakens and warm-up, and then executes the next instruction.

Stop Mode Wake-up Enable Flag for Ports RC (SEF)

The stop mode wake-up flag for ports RC is organized as an 4-bit binary register (SEF.0 to SEF.3). Before port RC may be used to make the device exit the stop mode, the content of the SEF must be set first. The SEF is controlled by the MOV SEF, #I instruction. The bit descriptions are as follows:

	3	2	1	0
SEF	w	w	w	w

Note: W means write only.

SEF 0 = 1 Device will exit stop mode when falling edge signal is applied to pin RC.0

SEF 1 = 1 Device will exit stop mode when falling edge signal is applied to pin RC.1

SEF 2 = 1 Device will exit stop mode when falling edge signal is applied to pin RC.2

SEF 3 = 1 Device will exit stop mode when falling edge signal is applied to pin RC.3

Hold Mode Operation

In hold mode, all operations of the µC cease, except for the operation of the oscillator and timer. The µC enters hold mode when the HOLD instruction is executed. The hold mode can be released in one of five ways: by the action of timer 0, timer 1, the divider, the $\overline{\text{INT}}$ pin, the RC port. Before the device enters the hold mode, the HEF, PEF, and IEF flags must be set to define the hold mode release conditions. For more details, refer to the instruction-set table and the following flow chart.

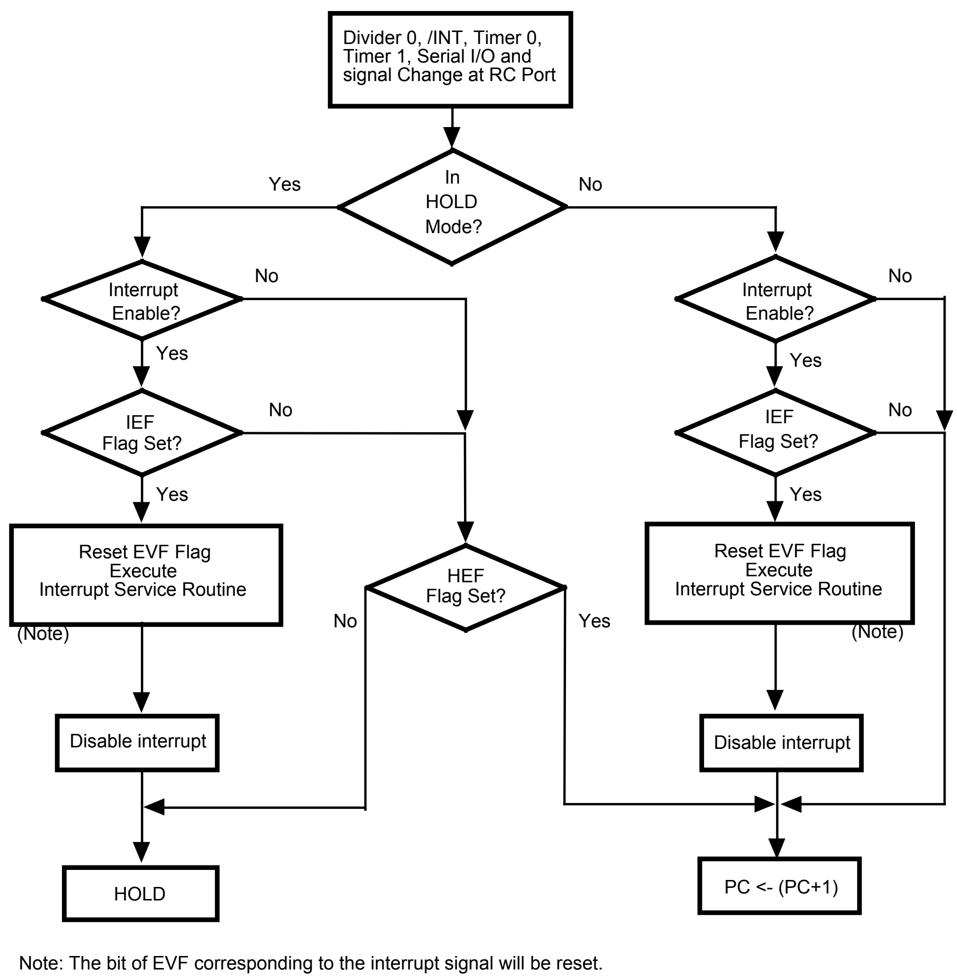


Figure 14. Hold Mode and Interrupt Operation Flow Chart

Hold Mode Release Enable Flag (HEF)

The hold mode release enable flag is organized as an 8-bit binary register (HEF.0 to HEF.7). The HEF is used to control the hold mode release conditions. It is controlled by the MOV HEF, #I instruction. The bit descriptions are as follows:

	7	6	5	4	3	2	1	0
HEF	w	w	w	w	–	w	w	w

Note: W means write only.

HEF.0 = 1   Overflow from the Divider 0 causes Hold mode to be released.

HEF.1 = 1   Underflow from Timer 0 causes Hold mode to be released.

HEF.2 = 1   Signal change at port RC causes Hold mode to be released.

HEF.3 is reserved.

HEF.4 = 1   Falling edge signal at the $\overline{\text{INT}}$ pin causes Hold mode to be released.

HEF.5 = 1   The serial port received completely causes Hold mode to be released.

HEF.6 = 1   The serial port transmitted completely causes Hold mode to be released.

HEF.7 = 1   Underflow from Timer 1 causes Hold mode to be released.

Hold Mode Release Condition Flag (HCF)

The hold mode release condition flag is organized as a 8-bit binary register (HCF0 to HCF7). It indicates by which interrupt source the hold mode has been released, and is loaded by hardware. The HCF can be read out by the MOVA R, HCFL and MOVA R, HCFH instructions. When any of the HCF bits is "1," the hold mode will be released and the HOLD instruction is invalid. The HCF can be reset by the CLR EVF or MOV HEF,#I (HEF = 0) instructions. When EVF and HEF have been reset, the corresponding bit of HCF is reset simultaneously. The bit descriptions are as follows:

	7	6	5	4	3	2	1	0
HCF	R	R	R	R	–	R	R	R

Note: R means read only.

HCF.0 = 1 Hold mode was released by overflow from the Divider 0

HCF.1 = 1 Hold mode was released by underflow from the timer 0

HCF.2 = 1 Hold mode was released by a signal change at port RC

HCF.3 is reserved.

HCF.4 = 1 Hold mode was released by a falling edge signal at the $\overline{\text{INT}}$ pin

HCF.5 = 1 Hold mode was released by underflow from the timer 1

HCF.6 = 1 Hold mode was released by the serial port received completely.

HCF.7 = 1 Hold mode was released by the serial port transmitted completely.

Event Flag (EVF)

The event flag is organized as a 8-bit binary register (EVF0 to EVF7). It is set by hardware and reset by CLR EVF, #I instruction or the occurrence of an interrupt. The bit descriptions are as follows:

	7	6	5	4	3	2	1	0
EVF	R	R	R	R	—	R	R	R

Note: R means read only.

EVF.0 = 1 Overflow from Divider 0 occurred.

EVF.1 = 1 Underflow from Timer 0 occurred.

EVF.2 = 1 Signal change at port RC occurred.

EVF.3 is reserved.

EVF.4 = 1 Falling edge signal at the $\overline{\text{INT}}$ pin occurred.

EVF.5 = 1 The serial port received completely.

EVF.6 = 1 The serial port transmitted completely.

EVF.7 = 1 Underflow from Timer 1 occurred.

Reset Function

The W741E20X is reset either by a power-on reset or by using the external $\overline{\text{RES}}$ pin. The initial state of the W741C200 after the reset function is executed is described below.

Program Counter (PC)	000H
TM0, TM1	Reset
MR0, MR1, PAGE registers	Reset
PSR0, PSR2, PM3 registers	Reset
IEF, HEF, HCF, PEF, EVF, SEF flags	Reset
Timer 0 input clock	Fosc/4
Timer 1 input clock	Fosc
MFP output	Low
Input/output ports RA, RB	Input mode
Input/output ports RC, RD	Input mode
Output port RE	High
RA and RB ports output type	CMOS type
Input clock of the watchdog timer	Fosc/1024

EEPROM Program/Erase Description

The built-in program code memory of the W741E20X is the EEPROM structure. This memory can be programmed, erased and verified through the VPP, MODE and DATA pins. The on board program/erase connection is shown below.

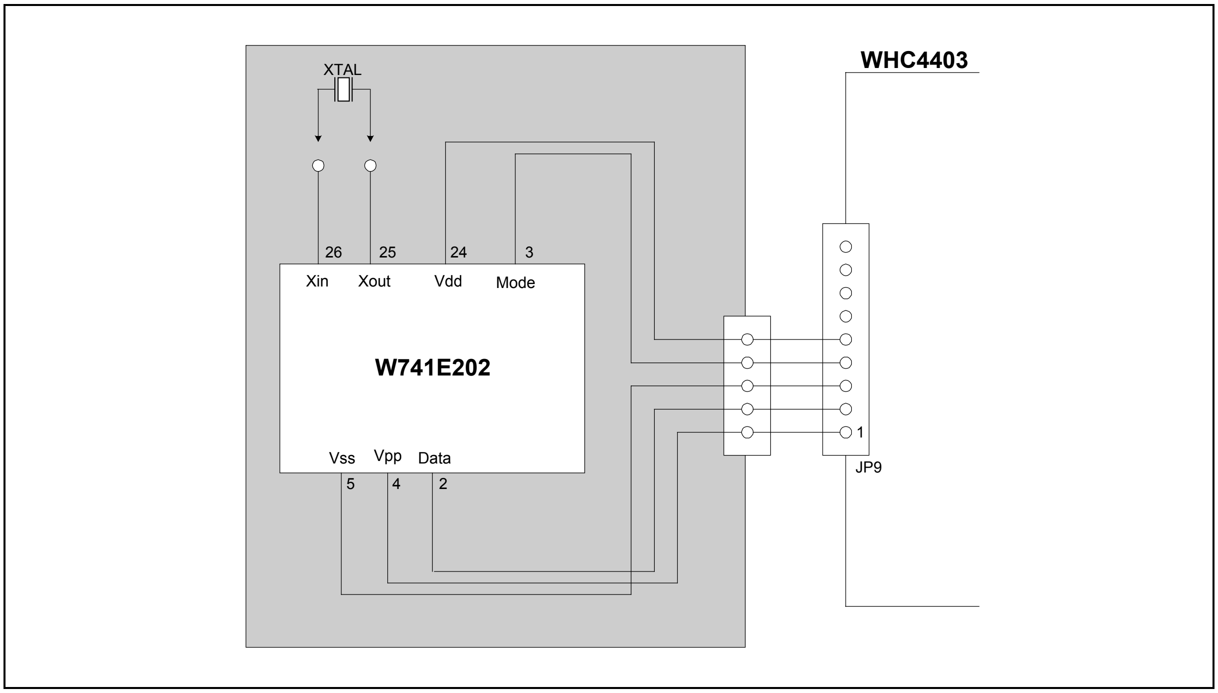


Figure 15. The W741E202 Program/Erase Configuration

ABSOLUTE MAXIMUM RATINGS

PARAMETER	RATING	UNIT
Supply Voltage to Ground Potential	-0.3 to +7.0	V
Applied Input/Output Voltage	-0.3 to +7.0	V
Power Dissipation	120	mW
Ambient Operating Temperature	0 to +70	°C
Storage Temperature	-55 to +150	°C

Note: Exposure to conditions beyond those listed under Absolute Maximum Ratings may adversely affect the life and reliability of the device.

DC CHARACTERISTICS

(VDD-VSS = 3.0 V, Fosc. = 32.768 KHz, Ta = 25° C; unless otherwise specified)

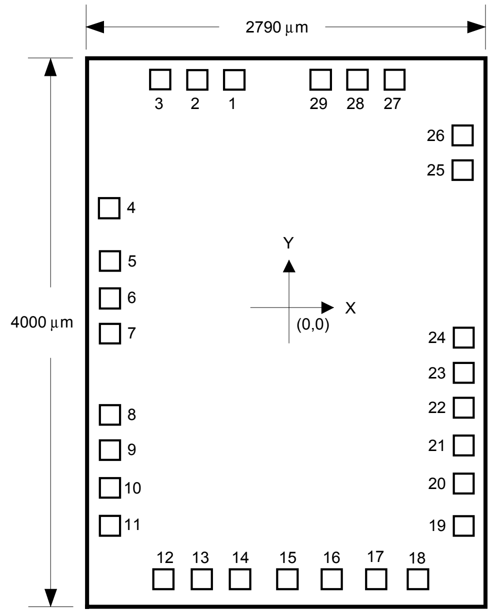
PARAMETER	SYM.	CONDITIONS	MIN.	TYP.	MAX.	UNIT
Op. Voltage	VDD	-	2.4	-	5.5	V
Op. Current (Crystal type)	IOP1	No load (Ext-V)	-	8.5	20	μA
Op. Current (RC type)	IOP2	No load (Ext-V)	-	36	65	μA
Hold Current (Crystal type)	IHM1	Hold mode No load (Ext-V)	-	4	6	μA
Hold Current (RC type)	IHM2	Hold mode No load (Ext-V)	-	16	40	μA
Stop Current (Crystal type)	ISM1	Stop mode No load (Ext-V)	-	0.1	2	μA
Stop Current (RC type)	ISM2	Stop mode No load (Ext-V)	-	0.1	2	μA
Input Low Voltage	VIL	-	VSS	-	0.3 VDD	V
Input High Voltage	VIH	-	0.7 VDD	-	VDD	V
MFP Output Low Voltage	VML	IOL = 3.5mA	-	-	0.4	V
MFP Output High Voltage	VMH	IOH = -3.5mA	2.4	-	-	V
Port RA, RB Sink Current	IABL	VOL = 0.9V	9	-	-	mA
Port RA, RB Source Current	IABH	VOH = 2.4V	0.4	1.2	-	mA
Port RC, RD Output Low Voltage	VCDL	IOL = 2.0 mA	-	-	0.4	V
Port RC, RD Output High Voltage	VCDH	IOH = -2.0 mA	2.4	-	-	V
Port RE Sink Current	IEL	VOL = 0.9V	9	-	-	mA
Port RE Source Current	IEH	VOH = 2.4V	0.4	1.2	-	mA
DIN Pin Pull-up Resistor	RDIN	RE.2 used as serial input pin	50	250	1000	KΩ

AC CHARACTERISTICS

(VDD-VSS = 3.0 V, Ta = 25° C; unless otherwise specified)

PARAMETER	SYM.	CONDITIONS	MIN.	TYP.	MAX.	UNIT
Op. Frequency	Fosc	RC type	-	-	4000	KHz
		Crystal type 1 (Option low speed type)	-	32.768	-	
		Crystal type 2 (Option high speed type)	400	-	4190	
Frequency Deviation by Voltage Drop for RC Oscillator	$\frac{\Delta f}{f}$	$\frac{f(3V) - f(2.4V)}{f(3V)}$	-	-	10	%
Instruction Cycle Time	T_I	One machine cycle	-	4/Fosc	-	S
Serial Port Data Ready Time	T_{DR}	-	200	-	-	nS
Serial Port Data Hold Time	T_{DH}	-	200	-	-	nS
Reset Active Width	T_{RAW}	Fosc = 32.768 KHz	1	-	-	μS
Interrupt Active Width	T_{IAW}	Fosc = 32.768 KHz	1	-	-	μS

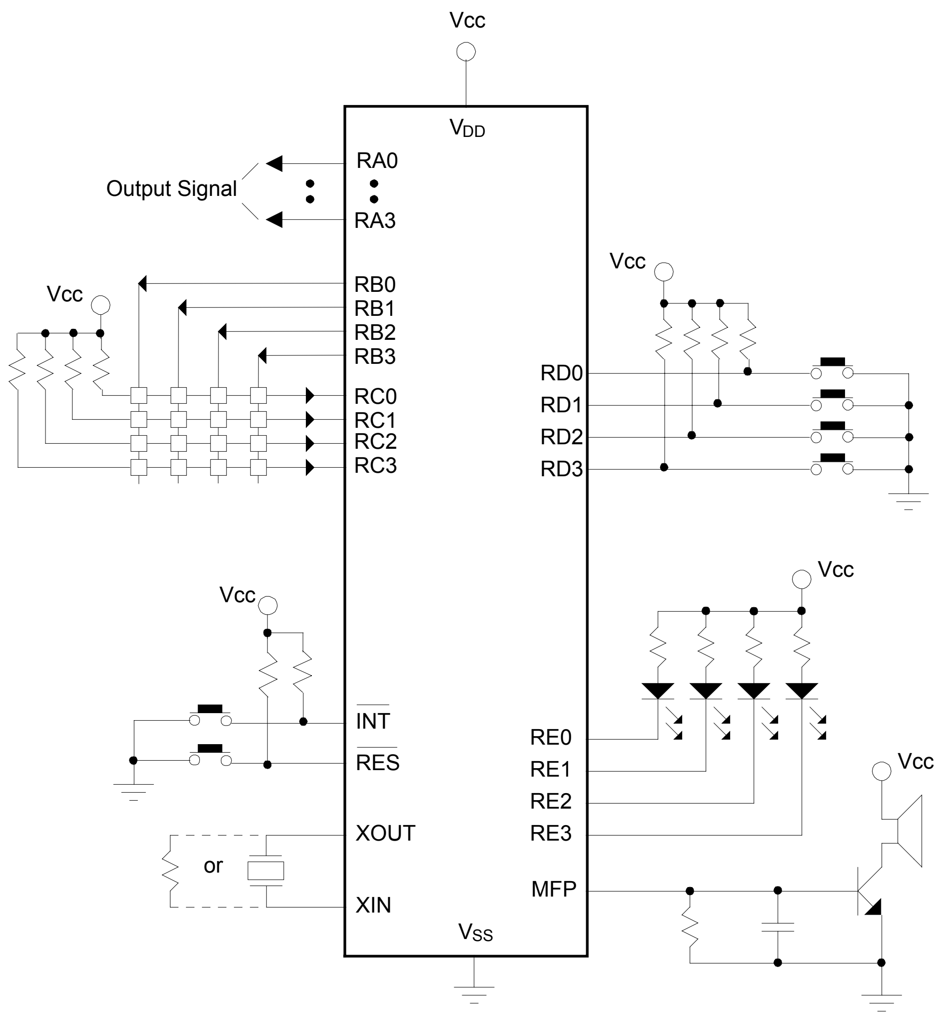
PAD ASSIGNMENT & POSITIONS



Note: The chip substrate must be connected to system ground (Vss).

PAD NO.	PAD NAME	X	Y	PAD NO.	PAD NAME	X	Y
1	RA2	-207.80	1744.11	16	RC0	275.20	-1763.79
2	RA3	-482.00	1744.11	17	RC1	600.00	-1763.79
3	$\overline{\text{INT}}$	-804.20	1746.11	18	RC2	878.40	-1763.79
4	$\overline{\text{RES}}$	-1160.80	828.61	19	RC3	1152.90	-1563.29
5	VSS	-1160.80	441.01	20	VDD	1152.90	-1329.49
6	RE0	-1158.80	155.31	21	RD0	1152.90	-1083.29
7	RE1	-1158.80	-115.69	22	RD1	1152.90	-812.29
8	RE2	-1158.80	-752.09	23	RD2	1152.90	-533.89
9	RE3	-1158.80	-1023.09	24	RD3	1152.90	262.89
10	VSS	-1160.80	-1242.29	25	VDD	1152.90	1298.81
11	RB0	-1158.80	-1522.29	26	XOUT	1152.90	1524.11
12	RB1	-903.10	-1763.79	27	XIN	825.80	1742.71
13	RB2	-636.10	-1763.79	28	RA0	505.00	1744.11
14	RB3	-361.90	-1763.79	29	RA1	230.80	1744.11
15	MFP	-3.20	-1763.79				

TYPICAL APPLICATION CIRCUIT



INSTRUCTION SET TABLE

Symbol Description

ACC:	Accumulator
ACC.n:	Accumulator bit n
WR:	Working Register
PAGE:	Page Register
MR0:	Mode Register 0
MR1:	Mode Register 1
PM0:	Port Mode 0
PM1:	Port Mode 1
PM2:	Port Mode 2
PM3:	Port Mode 3
PM4:	Port Mode 4
PM5:	Port Mode 5
PSR0:	Port Status Register 0
PSR2:	Port Status Register 2
R:	Memory (RAM) of address R
R.n:	Memory bit n of address R
I:	Constant parameter
L:	Branch or jump address
CF:	Carry Flag
ZF:	Zero Flag
PC:	Program Counter
TM0L:	Low nibble of the Timer 0 counter
TM0H:	High nibble of the Timer 0 counter
TM1L:	Low nibble of the Timer 1 counter
TM1H:	High nibble of the Timer 1 counter
TABL:	Low nibble of the look-up table address buffer
TABH:	High nibble of the look-up table address buffer
IEF.n:	Interrupt Enable Flag n
HCF.n:	HOLD mode release Condition Flag n
HEF.n:	HOLD mode release Enable Flag n
SEF.n:	STOP mode wake-up Enable Flag n
PEF.n:	Port Enable Flag n
EVF.n:	Event Flag n

Continued

! =:	Not equal
&:	AND
^:	OR
EX:	Exclusive OR
←:	Transfer direction, result
[PAGE*10H+()]:	Contents of address PAGE (bit2, bit1, bit0)*10H+()
[P()]:	Contents of port P

INSTRUCTION SET TABLE 1

MNEMONIC		FUNCTION	FLAG AFFECTED	CYCLE
Arithmetic				
ADD	R, ACC	ACC←(R) + (ACC)	ZF, CF	1
ADD	WR, #I	ACC←(WR) + I	ZF, CF	1
ADDR	R, ACC	ACC, R←(R) + (ACC)	ZF, CF	1
ADDR	WR, #I	ACC, WR←(WR) + I	ZF, CF	1
ADC	R, ACC	ACC←(R) + (ACC) + (CF)	ZF, CF	1
ADC	WR, #I	ACC←(WR) + I + (CF)	ZF, CF	1
ADCR	R, ACC	ACC, R←(R) + (ACC) + (CF)	ZF, CF	1
ADCR	WR, #I	ACC, WR←(WR) + I + (CF)	ZF, CF	1
ADU	R, ACC	ACC←(R) + (ACC)	ZF	1
ADU	WR, #I	ACC←(WR) + I	ZF	1
ADUR	R, ACC	ACC, R←(R) + (ACC)	ZF	1
ADUR	WR, #I	ACC, W R←(WR) + I	ZF	1
SUB	R, ACC	ACC←(R) - (ACC)	ZF, CF	1
SUB	WR, #I	ACC←(WR) - I	ZF, CF	1
SUBR	R, ACC	ACC, R←(R) - (ACC)	ZF, CF	1
SUBR	WR, #I	ACC, WR←(WR) - I	ZF, CF	1
SBC	R, ACC	ACC←(R) - (ACC) - (CF)	ZF, CF	1
SBC	WR, #I	ACC←(WR) - I - (CF)	ZF, CF	1
SBCR	R, ACC	ACC, R←(R) - (ACC) - (CF)	ZF, CF	1
SBCR	WR, #I	ACC, WR←(WR) - I - (CF)	ZF, CF	1

Instruction Set Table 1, continued

MNEMONIC		FUNCTION	FLAG AFFECTED	CYCLE
Arithmetic				
INC	R	ACC, R←(R) + 1	ZF, CF	1
DEC	R	ACC, R←(R) - 1	ZF, CF	1
Logic Operations				
ANL	R, ACC	ACC←(R) & (ACC)	ZF	1
ANL	WR, #I	ACC←(WR) & I	ZF	1
ANLR	R, ACC	ACC, R←(R) & (ACC)	ZF	1
ANLR	W, R #I	ACC, WR←(WR) & I	ZF	1
ORL	R, ACC	ACC←(R) ∧ (ACC)	ZF	1
ORL	WR, #I	ACC←(WR) ∧ I	ZF	1
ORLR	R, ACC	ACC, R←(R) ∧ (ACC)	ZF	1
ORLR	WR, #I	ACC, WR←(WR) ∧ I	ZF	1
XRL	R, ACC	ACC←(R) EX (ACC)	ZF	1
XRL	WR, #I	ACC←(WR) EX I	ZF	1
XRLR	R, ACC	ACC, R←(R) EX (ACC)	ZF	1
XRLR	WR, #I	ACC, WR←(WR) EX I	ZF	1
Branch				
JMP	L	PC10–PC0←L10–L0		1
JB0	L	PC10~PC0←L10~L0; if ACC.0 = "1"		1
JB1	L	PC10~PC0←L10~L0; if ACC.1 = "1"		1
JB2	L	PC10~PC0←L10~L0; if ACC.2 = "1"		1
JB3	L	PC10~PC0←L10~L0; if ACC.3 = "1"		1
JZ	L	PC10~PC0←L10~L0; if ACC = 0		1
JNZ	L	PC10~PC0←L10~L0; if ACC ! = 0		1
JC	L	PC10~PC0←L10~L0; if CF = "1"		1
JNC	L	PC10~PC0←L10~L0; if CF ! = "1"		1
DSKZ	R	ACC, R←(R) - 1; skip if ACC = 0	ZF, CF	1
DSKNZ	R	ACC, R←(R) - 1; skip if ACC ! = 0	ZF, CF	1
SKB0	R	Skip if R.0 = "1"		1
SKB1	R	Skip if R.1 = "1"		1
SKB2	R	Skip if R.2 = "1"		1
SKB3	R	Skip if R.3 = "1"		1

Instruction Set Table 1, continued

MNEMONIC		FUNCTION	FLAG AFFECTED	CYCLE
Data Move				
MOV	WR, R	WR←(R)		1
MOV	R, WR	R←(WR)		1
MOVA	WR, R	ACC, WR←(R)	ZF	1
MOVA	R, WR	ACC, R←(WR)	ZF	1
MOV	R, ACC	R←(ACC)		1
MOV	ACC, R	ACC←(R)	ZF	1
MOV	R, #I	R←I		1
MOV	WR, @R	WR←[PR(bit2, bit1, bit0) × 10H +(R)]		2
MOV	@R, WR	[PR(bit2, bit1, bit0) × 10H +(R)]←WR		2
MOV	TABL, R	TABL←(R)		1
MOV	TABH, R	TABH←(R)		1
MOVC	R	R←[(TABH) × 10H + (TABL)]		2
MOVC	WR, #I	WR←[(I6–I0) × 10H + (ACC)]		2
Input & Output				
MOVA	R, RA	ACC, R←[RA]	ZF	1
MOVA	R, RB	ACC, R←[RB]	ZF	1
MOVA	R, RC	ACC, R←[RC]	ZF	1
MOVA	R, RD	ACC, R←[RD]	ZF	1
MOV	RA, R	[RA]←(R)		1
MOV	RB, R	[RB]←(R)		1
MOV	RC, R	[RC]←(R)		1
MOV	RD, R	[RD]←(R)		1
MOV	RE, R	[RE]←(R)		1
SOP	R	RE0←(R), (ACC); RE1←CLK		1
SIP	R	R, ACC ←SIB; RE3←CLK		2
MOV	MFP, #I	[MFP]← I		1
Flag & Register				
MOVA	R, PAGE	ACC, R←PAGE (Page Register)	ZF	1
MOV	PAGE, R	PAGE←(R)		1
MOV	MR0, #I	MR0←I		1
MOV	MR1, #I	MR1←I		1
MOV	PAGE, #I	PAGE←I		1
MOVA	R, CF	ACC.0, R.0←CF	ZF	1
MOV	CF, R	CF←(R.0)	CF	1

Instruction Set Table 1, continued

MNEMONIC		FUNCTION	FLAG AFFECTED	CYCLE
MOVA	R, HCFL	ACC, R←HCF0–HCF3	ZF	1
MOVA	R, HCFH	ACC, R←HCF4–HCF7	ZF	1
CLR	PMF, #I	Clear Parameter Flag if In = 1		1
SET	PMF, #I	Set Parameter Flag if In = 1		1
MOV	PM0, #I	Port Mode 0← I		1
MOV	PM1, #I	Port Mode 1← I		1
MOV	PM2, #I	Port Mode 2← I		1
MOV	PM3, #I	Port Mode 3← I		1
MOV	PM4, #I	Port Mode 4← I		1
MOV	PM5, #I	Port Mode 5← I		1
CLR	EVF, #I	Clear Event Flag if In = 1		1
MOV	PEF, #I	Set/Reset Port Enable Flag		1
MOV	IEF, #I	Set/Reset Interrupt Enable Flag		1
MOV	HEF, #I	Set/Reset HOLD mode release Enable Flag		1
MOV	SEF, #I	Set/Reset STOP mode wake-up Enable Flag for RC port		1
MOVA	R, PSR0	ACC, R←Port Status Register 0	ZF	1
CLR	PSR0	Clear Port Status Register 0		1
MOVA	R, PSR2	ACC, R←Port Status Register 2	ZF	1
CLR	PSR2	Clear Port Status Register 2		1
SET	CF	Set Carry Flag	CF	1
CLR	CF	Clear Carry Flag	CF	1
CLR	DIVR0	Clear the last 4-bit of the Divider 0		1
CLR	WDT	Clear WatchDog Timer		1
Shift & Rotate				
SHRC	R	ACC.n, R.n←(R.n+1); ACC.3, R.3←0; CF←R.0	ZF, CF	1
RRC	R	ACC.n, R.n←(R.n+1); ACC.3, R.3←CF; CF←R.0	ZF, CF	1
SHLC	R	ACC.n, R.n←(R.n-1); ACC.0, R.0←0; CF←R.3	ZF, CF	1
RLC	R	ACC.n, R.n←(R.n-1); ACC.0, R.0←CF; CF←R.3	ZF, CF	1

Instruction Set Table 1, continued

MNEMONIC		FUNCTION	FLAG AFFECTED	CYCLE
Timer				
MOV	TM0L, R	TM0L←(R)		1
MOV	TM0H, R	TM0H←(R)		1
MOV	TM0, #I	Timer 0 set		1
MOV	TM1L, R	TM1L←(R)		1
MOV	TM0H, R	TM0H←(R)		1
MOV	TM1, #I	Timer 1 set		1
Subroutine				
CALL	L	STACK ← (PC)+1; PC10 ~ PC0 ← L10 ~ L0		1
RTN		(PC)← STACK		1
Other				
HOLD		Enter Hold mode		1
STOP		Enter Stop mode		1
NOP		No Operation		1
EN	INT	Enable Interrupt Function		1
DIS	INT	Disable Interrupt Function		1

INSTRUCTION SET TABLE 2

ADC R, ACC	**Add R to ACC with CF**
Machine Code:	0 0 0 0 1 0 0 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC ← (R) + (ACC) + (CF)
Description:	The contents of the data memory location addressed by R6 to R0, ACC, and CF are binary added and the result is loaded into the ACC.
Flag Affected:	CF & ZF
ADC WR, #I	**Add immediate data to WR with CF**
Machine Code:	0 0 0 0 1 1 0 0 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC ← (WR) + I + (CF)
Description:	The contents of the Working Register (WR), I and CF are binary added and the result is loaded into the ACC.
Flag Affected:	CF & ZF
ADCR R, ACC	**Add R to ACC with CF**
Machine Code:	0 0 0 0 1 0 0 1 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) + (ACC) + (CF)
Description:	The contents of the data memory location addressed by R6 to R0, ACC, and CF are binary added and the result is placed in the ACC and the data memory.
Flag Affected:	CF & ZF

Instruction Set Table 2, continued

ADCR WR, #I	**Add immediate data to WR with CF**
Machine Code:	0 0 0 0 1 1 0 1 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC, WR ← (WR) + I + (CF)
Description:	The contents of the Working Register (WR), I, CF are binary added and the result is placed in the ACC and the WR.
Flag Affected:	CF & ZF
ADD R, ACC	**Add R to ACC**
Machine Code:	0 0 0 1 1 0 0 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC ← (R) + (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and ACC are binary added and the result is loaded into the ACC.
Flag Affected:	CF & ZF
ADD WR, #I	**Add immediate data to WR**
Machine Code:	0 0 0 1 1 1 0 0 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC ← (WR) + I
Description:	The contents of the Working Register (WR) and the immediate data I are binary added and the result is loaded into the ACC.
Flag Affected:	CF & ZF

Instruction Set Table 2, continued

ADDR R, ACC	Add R to ACC
Machine Code:	0 0 0 1 1 0 0 1 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) + (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and ACC are binary added and the result is placed in the ACC and the data memory.
Flag Affected:	CF & ZF
ADDR WR, #I	**Add immediate data to WR**
Machine Code:	0 0 0 1 1 1 0 1 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC, WR ← (WR) + I
Description:	The contents of the Working Register (WR) and the immediate data I are binary added and the result is placed in the ACC and the WR.
Flag Affected:	CF & ZF
ADU R, ACC	**Add R to ACC and Carry Flag unchange**
Machine Code:	0 0 1 0 1 0 0 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC ← (R) + (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and ACC are binary added and the result is loaded into the ACC.
Flag Affected:	ZF

Instruction Set Table 2, continued

ADU WR, #I	**Add immediate data to WR and Carry Flag unchange**
Machine Code:	0 0 1 0 1 1 0 0 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC ← (WR) + I
Description:	The contents of the Working Register (WR) and the immediate data I are binary added and the result is loaded into the ACC.
Flag Affected:	ZF
ADUR R, ACC	**Add R to ACC and Carry Flag unchange**
Machine Code:	0 0 1 0 1 0 0 1 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) + (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and ACC are binary added and the result is placed in the ACC and the data memory.
Flag Affected:	ZF
ADUR WR, #I	**Add immediate data to WR and Carry Flag unchange**
Machine Code:	0 0 1 0 1 1 0 1 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC, WR ← (WR) + I
Description:	The contents of the Working Register (WR) and the immediate data I are binary added and the result is placed in the WR and the ACC.
Flag Affected:	ZF

Instruction Set Table 2, continued

ANL R, ACC	And R to ACC
Machine Code:	0 0 1 0 1 0 1 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC ← (R) & (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and the ACC are ANDed and the result is loaded into the ACC.
Flag Affected:	ZF
ANL WR, #I	**And immediate data to WR**
Machine Code:	0 0 1 0 1 1 1 0 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC ← (WR) & I
Description:	The contents of the Working Register (WR) and the immediate data I are ANDed and the result is loaded into the ACC.
Flag Affected:	ZF
ANLR R, ACC	**And R to ACC**
Machine Code:	0 0 1 0 1 0 1 1 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) & (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and the ACC are ANDed and the result is placed in the data memory and the ACC.
Flag Affected:	ZF

Instruction set table 2, continued

ANLR WR, #I	**And immediate data to WR**
Machine Code:	0 0 1 0 1 1 1 1	I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC, WR ← (WR) & I
Description:	The contents of the Working Register (WR) and the immediate data I are ANDed and the result is placed in the WR and the ACC.
Flag Affected:	ZF
CALL L	**Call subroutine**
Machine Code:	0 1 1 0 0 L10 L9 L8	L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	STACK ← (PC)+1; PC10 ~ PC0 ← L10 ~ L0
Description:	The next program counter (PC10 to PC0) is saved in the STACK and then the direct address (L10 to L0) is loaded into the program counter. A subroutine is called.
CLR CF	**Clear CF**
Machine Code:	0 1 0 1 0 0 0 0	0 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	Clear CF
Description:	Clear Carry Flag to 0.
Flag Affected:	CF

Instruction Set Table 2, continued

CLR DIVR0	**Reset the last 4 bits of the DIVideR 0**
Machine Code:	0 0 0 1 0 1 1 1 / 0 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	Reset the last 4 bits of the divider 0
Description:	When this instruction is executed, the last 4 bits of the divider 0 (14 bits) are reset.

CLR EVF, #I	**Clear EVent Flag**
Machine Code:	0 1 0 0 0 0 0 0 / I7 I6 I5 I4 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Clear event flag
Description:	The condition corresponding to the data specified by I7 to I0 is controlled.

I0~I7	Mode after execution of instruction
I0 = 1	EVF0 caused by overflow from the divider 0 is reset.
I1 = 1	EVF1 caused by underflow from the timer 0 is reset.
I2 = 1	EVF2 caused by the signal change at port RC is reset.
I3	Reserved
I4 = 1	EVF4 caused by the falling edge signal on INT pin is reset.
I5 = 1	EVF5 caused by the serial port received completely.
I6 = 1	EVF6 caused by the serial port transmitted completely.
I7 = 1	EVF7 caused by underflow from the timer 1 is reset.

Instruction Set Table 2, continued

CLR PMF, #I	**Clear ParaMeter Flag**
Machine Code:	0 0 0 1 0 1 1 0	1 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Clear Parameter Flag
Description:	Description of each flag: I0, I1, I2: Reserved I3 = 1: The input clock of the watchdog timer is Fosc/1024.
CLR PSR0	**Clear Port Status Register 0 (RC port signal change flag)**
Machine Code:	0 1 0 0 0 0 1 0	0 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	Clear Port Status Register 0 (RC port signal change flag)
Description:	When this instruction is executed, the RC port signal change flag (PSR0) is cleared.
CLR PSR2	**Clear Port Status Register 2 (serial port status flags)**
Machine Code:	0 1 0 1 0 1 0 1	0 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	Clear Port Status Register 2 (serial port status flags)
Description:	When this instruction is executed, the serial port status flags (PSR2) are cleared.
CLR WDT	**Reset the last 4 bits of the Watchdog Timer**
Machine Code:	0 0 0 1 0 1 1 1	1 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	Reset the last 4 bits of the watchdog timer
Description:	When this instruction is executed, the last 4 bits of the watchdog timer are reset.

Instruction Set Table 2, continued

DEC R	**Decrement R content**
Machine Code:	0 1 0 0 1 0 1 0 | 1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) - 1
Description:	Decrement the data memory content and load result into the ACC and the data memory.
Flag Affected:	CF & ZF
DIS INT	**Disable Interrupt function**
Machine Code:	0 1 0 1 0 0 0 0 | 1 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	Disable interrupt function
Description:	Interrupt function is inhibited by executing this instruction.
DSKNZ R	**Decrement R content then skip if ACC ! = 0**
Machine Code:	0 1 0 0 1 0 0 0 | 1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) - 1; PC ← (PC) + 2 if ACC ! = 0
Description:	Decrement the data memory content and load result into the ACC and the data memory. If ACC ! = 0, the program counter is incremented by 2 and produces a skip.
Flag Affected:	CF & ZF

Instruction Set Table 2, continued

DSKZ R	**Decrement R content then skip if ACC is zero**
Machine Code:	0 1 0 0 1 0 0 0	0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) - 1; PC ← (PC) + 2 if ACC = 0
Description:	Decrement the data memory content and load result into the ACC and the data memory. If ACC = 0, the program counter is incremented by 2 and produces a skip.
Flag Affected:	CF & ZF
EN INT	**Enable Interrupt function**
Machine Code:	0 1 0 1 0 0 0 0	1 1 0 0 0 0 0 0
Machine Cycle:	1
Operation:	Enable interrupt function
Description:	This instruction enables the interrupt function.
HOLD	**Enter the HOLD mode**
Machine Code:	0 0 0 0 0 0 0 0	1 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	Enter the HOLD mode
Description:	The following two conditions cause the HOLD mode to be released. (1) An interrupt is accepted. (2) The HOLD release condition specified by the HEF is met. In HOLD mode, when an interrupt is accepted the HOLD mode will be released and the interrupt service routine will be executed. After completing the interrupt service routine by executing the RTN instruction, the µC will enter HOLD mode again.

Instruction Set Table 2, continued

INC R	**Increment R content**
Machine Code:	0 1 0 0 1 0 1 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) + 1
Description:	Increment the data memory content and load the result into the ACC and the data memory.
Flag Affected:	CF & ZF
JB0 L	**Jump when bit 0 of ACC is "1"**
Machine Code:	1 0 0 0 0 L10 L9 L8 | L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	PC10 ~ PC0 ← L10 ~ L0; if ACC.0 = "1"
Description:	If bit 0 of the ACC is "1," PC10 to PC0 of the program counter are replaced with the data specified by L10 to L0 and a jump occurs. If bit 0 of the ACC is "0," the program counter (PC) is incremented.
JB1 L	**Jump when bit 1 of ACC is "1"**
Machine Code:	1 0 0 1 0 L10 L9 L8 | L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	PC10 ~ PC0 ← L10 ~ L0; if ACC.1 = "1"
Description:	If bit 1 of the ACC is "1," PC10 to PC0 of the program counter are replaced with the data specified by L10 to L0 and a jump occurs. If bit 1 of the ACC is "0," the program counter (PC) is incremented.

Instruction Set Table 2, continued

JB2 L	**Jump when bit 2 of ACC is "1"**
Machine Code:	1 0 1 0 0 L10 L9 L8 | L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	PC10 ~ PC0 ← L10 ~ L0; if ACC.2="1"
Description:	If bit 2 of the ACC is "1," PC10 to PC0 of the program counter are replaced with the data specified by L10 to L0 and a jump occurs. If bit 2 of the ACC is "0," the program counter (PC) is incremented.
JB3 L	**Jump when bit 3 of ACC is "1"**
Machine Code:	1 0 1 1 0 L10 L9 L8 | L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	PC10 ~ PC0 ← L10 ~ L0; if ACC.3 = "1"
Description:	If bit 3 of the ACC is "1," PC10 to PC0 of the program counter are replaced with the data specified by L10 to L0 and a jump occurs. If bit 3 of the ACC is "0," the program counter (PC) is incremented.
JC L	**Jump when CF is "1"**
Machine Code:	1 1 1 1 0 L10 L9 L8 | L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	PC10 ~ PC0 ← L10 ~ L0; if CF = "1"
Description:	If CF is "1," PC10 to PC0 of the program counter are replaced with the data specified by L10 to L0 and a jump occurs. If the CF is "0," the program counter (PC) is incremented.
JMP L	**Jump absolutely**
Machine Code:	0 1 1 1 0 L10 L9 L8 | L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	PC10 ~ PC0 ← L10 ~ L0
Description:	PC10 to PC0 of the program counter are replaced with the data specified by L10 to L0 and an unconditional jump occurs.

Instruction Set Table 2, continued

JNC L	**Jump when CF is not "1"**
Machine Code:	1 1 0 1 0 L10 L9 L8 | L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	PC10 ~ PC0 ← L10 ~ L0; if CF = "0"
Description:	If CF is "0," PC10 to PC0 of the program counter are replaced with the data specified by L10 to L0 and a jump occurs. If CF is "1," the program counter (PC) is incremented.
JNZ L	**Jump when ACC is not zero**
Machine Code:	1 1 0 0 0 L10 L9 L8 | L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	PC10 ~ PC0 ← L10 ~ L0; if ACC ! = 0
Description:	If the ACC is not zero, PC10 to PC0 of the program counter are replaced with the data specified by L10 to L0 and a jump occurs. If the ACC is zero, the program counter (PC) is incremented.
JZ L	**Jump when ACC is zero**
Machine Code:	1 1 1 0 0 L10 L9 L8 | L7 L6 L5 L4 L3 L2 L1 L0
Machine Cycle:	1
Operation:	PC10 ~ PC0 ← L10 ~ L0; if ACC = 0
Description:	If the ACC is zero, PC10 to PC0 of the program counter are replaced with the data specified by L10 to L0 and a jump occurs. If the ACC is not zero, the program counter (PC) is incremented.

Instruction Set Table 2, continued

MOV ACC, R	**Move R content to ACC**
Machine Code:	0 1 0 0 1 1 1 0 | 1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC ← (R)
Description:	The contents of the data memory location addressed by R6 to R0 are loaded into the ACC.
	ZF
MOV CF, R	**Move R.0 content to CF**
Machine Code:	0 1 0 1 1 0 0 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	CF ← (R.0)
Description:	The bit 0 content of the data memory location addressed by R6 to R0 is loaded into CF.
Flag Affected:	CF

Instruction Set Table 2, continued

MOV HEF, #I	Set/Reset Hold mode release Enable Flag
Machine Code:	0 1 0 0 0 0 0 1	I7 I6 I5 I4 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Hold mode release enable flag control
Description:	(see table below)

I0~I7	Operation
I0 = 1	The HEF0 is set so that overflow from the divider 0 caused the HOLD mode to be released.
I1 = 1	The HEF1 is set so that underflow from the Timer 0 caused the HOLD mode to be released.
I2 = 1	The HEF2 is set so that signal change at port RC caused the HOLD mode to be released.
I3	Reserved
I4 = 1	The HEF4 is set so that the falling edge signal at the INT pin caused the HOLD mode to be released.
I5 = 1	The HEF5 is set so that the serial port received completely caused the HOLD mode to be released.
I6 = 1	The HEF6 is set so that the serial port transmitted completely caused the HOLD mode to be released.
I7 = 1	The HEF7 is set so that underflow from the Timer 1 caused the HOLD mode to be released.

MOV IEF, #I	Set/Reset Interrupt Enable Flag
Machine Code:	0 1 0 1 0 0 0 1	I7 I6 I5 I4 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Interrupt Enable flag Control
Description:	The interrupt enable flag corresponding to the data specified by I7 - I0 is controlled:

Instruction Set Table 2, continued

I0~I7	Operation
I0 = 1	The IEF0 is set so that interrupt 0 (overflow from the divider 0) is accepted.
I1 = 1	The IEF1 is set so that interrupt 1 (underflow from the Timer 0) is accepted.
I2 = 1	The IEF2 is set so that interrupt 2 (signal change at port RC) is accepted.
I3	Reserved
I4 = 1	The IEF4 is set so that interrupt 4 (falling edge signal at the INT pin) is accepted.
I5 = 1	The IEF5 is set so that interrupt 5 (the serial port received completely) is accepted.
I6 = 1	The IEF6 is set so that interrupt 6 (the serial port transmitted completely) is accepted.
I7 = 1	The IEF7 is set so that interrupt 7 (underflow from the Timer 1) is accepted.

MOV MFP, #I — **Modulation Frequency Pulse generator**

Machine Code:

0	0	0	1	0	0	1	0	I7	I6	I5	I4	I3	I2	I1	I0

Machine Cycle: 1

Operation: [MFP] ← I

Description: If the bit 2 of MR1 is "0," the waveform specified by I7 to I0 is delivered at the MFP output pin (MFP). The relation between the waveform and immediate data I is shown as follows:

I5~I0	I0 = 1	I1 = 1	I2 = 1	I3 = 1	I4 = 1	I5 = 1
Signal	$\frac{Fosc}{256}$	$\frac{Fosc}{512}$	$\frac{Fosc}{4096}$	$\frac{Fosc}{8192}$	$\frac{Fosc}{16384}$	$\frac{Fosc}{32768}$

I7	I6	Signal
0	0	Low
0	1	High
1	0	Fosc/16
1	1	Fosc/8

Instruction Set Table 2, continued

MOV MR0, #I	**Load immediate data to Mode Register 0 (MR0)**
Machine Code:	0 0 0 1 0 0 1 1 / 1 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	MR0 ← I
Description:	The immediate data I are loaded to the MR0. MR0 bits description: bit 0: = 0 The fundamental frequency of Timer 0 is Fosc/4; = 1 The fundamental frequency of Timer 0 is Fosc/1024 bit 1: Reserved bit 2: Reserved bit 3: = 0 Timer 0 stop down-counting; = 1 Timer 0 start down-counting
MOV MR1, #I	**Load immediate data to Mode Register 1 (MR1)**
Machine Code:	0 0 0 1 0 0 1 1 / 0 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	MR1 ← I
Description:	The immediate data I are loaded to the MR1. MR1 bit description: bit0: = 0 The internal fundamental frequency of Timer 1 is Fosc; = 1 The internal fundamental frequency of Timer 1 is Fosc/64 bit1: = 0 The fundamental frequency source of Timer 1 is internal clock; = 1 The fundamental frequency source of Timer 1 is external clock via RC.0 input pin bit2: = 0 The specified waveform of the MFP generator is delivered at the MFP output pin; = 1 The specified frequency of the Timer 1 is delivered at the MFP output pin bit3: = 0 Timer 1 stop down-counting; = 1 Timer 1 start down-counting

MR0 bits description:

bit 0	= 0 The fundamental frequency of Timer 0 is Fosc/4 = 1 The fundamental frequency of Timer 0 is Fosc/1024
bit 1	Reserved
bit 2	Reserved
bit 3	= 0 Timer 0 stop down-counting = 1 Timer 0 start down-counting

MR1 bit description:

bit0	= 0 The internal fundamental frequency of Timer 1 is Fosc = 1 The internal fundamental frequency of Timer 1 is Fosc/64
bit1	= 0 The fundamental frequency source of Timer 1 is internal clock = 1 The fundamental frequency source of Timer 1 is external clock via RC.0 input pin
bit2	= 0 The specified waveform of the MFP generator is delivered at the MFP output pin = 1 The specified frequency of the Timer 1 is delivered at the MFP output pin
bit3	= 0 Timer 1 stop down-counting = 1 Timer 1 start down-counting

Instruction Set Table 2, continued

MOV PAGE, #I	**Load immediate data to Page Register**
Machine Code:	0 1 0 1 0 1 1 0	1 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Page Register ← I
Description:	The immediate data I are loaded to the PR. Bit 3 is reserved. Bit 0, bit 1, and bit 2 indirect addressing mode preselect bits:

bit2	bit1	bit0	
0	0	0	= Page 0 (00H~0FH)
0	0	1	= Page 1 (10H~1FH)
0	1	0	= Page 2 (20H~2FH)
0	1	1	= Page 3 (30H~3FH)
1	0	0	= Page 4 (40H~4FH)
1	0	1	= Page 5 (50H~5FH)
1	1	0	= Page 6 (60H~6FH)
1	1	1	= Page 7 (70H~7FH)

MOV PEF, #I	**Set/Reset Port Enable Flag**
Machine Code:	0 1 0 0 0 0 1 1	0 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Port enable flag control
Description:	The data specified by I can cause HOLD mode to be released or an interrupt to occur. The signal change on port RC is specified.

I0~I7	Signal change at port RC
I0 = 1	RC0
I1 = 1	RC1
I2 = 1	RC2
I3 = 1	RC3

Instruction Set Table 2, continued

MOV PM0, #I	**Set/Reset Port Mode 0 register**
Machine Code:	0 1 0 1 0 0 1 1	0 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Set/Reset Port mode 0 register
Description:	I0 = 0: RA port is CMOS type; I0 = 1: RA port is NMOS type. I1 = 0: RB port is CMOS type; I1 = 1: RB port is NMOS type. I2 = 0: RC port pull-high resistor is disabled; I2 = 1: RC port pull-high resistor is enabled. I3 = 0: RD port pull-high resistor is disabled; I3 = 1: RD port pull-high resistor is enabled.
MOV PM1, #I	**RA port independent Input/Output control**
Machine Code:	0 1 0 1 0 1 1 1	0 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	RA port 4 pins input/output control is independent.
Description:	I0 = 0: RA.0 is output pin; I0 = 1: RA.0 is input pin. I1 = 0: RA.1 is output pin; I1 = 1: RA.1 is input pin. I2 = 0: RA.2 is output pin; I2 = 1: RA.2 is input pin. I3 = 0: RA.3 is output pin; I3 = 1: RA.3 is input pin. Default condition RA port is input mode (PM = 1111B).
MOV PM2, #I	**RB port independent Input/Output control**
Machine Code:	0 1 0 1 0 1 1 1	1 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	RB port 4 pins input/output control is independent.
Description:	I0 = 0: RB.0 is output pin; I0 = 1: RB.0 is input pin. I1 = 0: RB.1 is output pin; I1 = 1: RB.1 is input pin. I2 = 0: RB.2 is output pin; I2 = 1: RB.2 is input pin. I3 = 0: RB.3 is output pin; I3 = 1: RB.3 is input pin. Default condition RB port is input mode (PM2 = 1111B).

Instruction Set Table 2, continued

MOV PM3, #I	**Set/Reset Port Mode 3 register**
Machine Code: Machine Cycle: Operation: Description:	0 1 0 1 0 1 1 0 | 0 0 0 0 I3 I2 I1 I0 1 Set/Reset Port mode 3 register I0 is reserved. I1 = 0: The port RE is used as the output of the internal parallel port RT. I1 = 1: The port RE works as the serial input/output port. I2 is reserved. I3 = 0: Serial Tx rate = FOSC/2 I3 = 1: Serial Tx rate = FOSC/256
MOV PM4, #I	**RC port independent Input/Output control**
Machine Code: Machine Cycle: Operation: Description:	0 0 1 1 0 1 1 1 | 0 0 0 0 I3 I2 I1 I0 1 RC port 4 pins input/output control is independent. I0 = 0: RC.0 is output pin; I0 = 1: RC.0 is input pin. I1 = 0: RC.1 is output pin; I1 = 1: RC.1 is input pin. I2 = 0: RC.2 is output pin; I2 = 1: RC.2 is input pin. I3 = 0: RC.3 is output pin; I3 = 1: RC.3 is input pin. Default condition RC port is input mode (PM4 = 1111B).
MOV PM5, #I	**RD port independent Input/Output control**
Machine Code: Machine Cycle: Operation: Description:	0 0 1 1 0 1 1 1 | 1 0 0 0 I3 I2 I1 I0 1 RD port 4 pins input/output control is independent. I0 = 0: RD.0 is output pin; I0 = 1: RD.0 is input pin. I1 = 0: RD.1 is output pin; I1 = 1: RD.1 is input pin. I2 = 0: RD.2 is output pin; I2 = 1: RD.2 is input pin. I3 = 0: RD.3 is output pin; I3 = 1: RD.3 is input pin. Default condition RD port is input mode (PM5 = 1111B).

Instruction Set Table 2, continued

MOV R, ACC	**Move ACC content to R**
Machine Code:	0 1 0 1 1 0 0 1	1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	R ← (ACC)
Description:	The contents of the ACC are loaded to the data memory location addressed by R6 to R0.
MOVA R, RA	**Input RA port data to ACC & R**
Machine Code:	0 1 0 1 1 0 1 1	0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC , R ← [RA]
Description:	The data on port RA are loaded into the data memory location addressed by R6 to R0 and the ACC.
Flag Affected:	ZF
MOVA R, RB	**Input RB port data to ACC & R**
Machine Code:	0 1 0 1 1 0 1 1	1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC , R ← [RB]
Description:	The data on port RB are loaded into the data memory location addressed by R6 to R0 and the ACC.
Flag Affected:	ZF

Instruction Set Table 2, continued

MOVA R, RC	**Input RC port data to ACC & R**
Machine Code:	0 1 0 0 1 0 1 1 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC , R ← [RC]
Description:	The input data on the input port RC are loaded into the data memory location addressed by R6 to R0 and the ACC.
Flag Affected:	ZF
MOVA R, RD	**Input RD port data to ACC & R**
Machine Code:	0 1 0 0 1 0 1 1 | 1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC , R ← [RD]
Description:	The input data on the input port RD are loaded into the data memory location addressed by R6 to R0 and the ACC.
Flag Affected:	ZF
MOV R, WR	**Move WR content to R**
Machine Code:	1 1 1 1 1 W3 W2 W1 | W0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	R ← (WR)
Description:	The contents of the WR are loaded to the data memory location addressed by R6 to R0.

Instruction Set Table 2, continued

MOV R, #I	**Load immediate data to R**
Machine Code:	[1 0 1 1 1 I3 I2 I1] [I0 R6 R5 R4 R3 R2 R1 R0]
Machine Cycle:	1
Operation:	R ← I
Description:	The immediate data I are loaded to the data memory location addressed by R6 to R0.
MOV RA, R	**Output R content to RA port**
Machine Code:	[0 1 0 1 1 0 1 0] [0 R6 R5 R4 R3 R2 R1 R0]
Machine Cycle:	1
Operation:	[RA] ← (R)
Description:	The data in the data memory location addressed by R6 to R0 are output to the port RA.
MOV RB, R	**Output R content to RB port**
Machine Code:	[0 1 0 1 1 0 1 0] [1 R6 R5 R4 R3 R2 R1 R0]
Machine Cycle:	1
Operation:	[RB] ← (R)
Description:	The contents of the data memory location addressed by R6 to R0 are output to the port RB.
MOV RC, R	**Output R content to RC port**
Machine Code:	[1 0 0 0 1 1 0 0] [0 R6 R5 R4 R3 R2 R1 R0]
Machine Cycle:	1
Operation:	[RC] ← (R)
Description:	The data in the data memory location addressed by R6 to R0 are output to the port RC.

Instruction Set Table 2, continued

MOV RD, R	**Output R content to RD port**
Machine Code:	1 0 0 0 1 1 0 0 | 1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	[RD] ← (R)
Description:	The contents of the data memory location addressed by R6 to R0 are output to the port RD.
MOV RE, R	**Output R content to port RE**
Machine Code:	0 1 0 1 1 1 1 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	[RE] ← (R)
Description:	The contents of the data memory location addressed by R6 to R0 are output to port RE.
MOV SEF, #I	**Set/Reset STOP mode waked-up Enable Flag for port RC**
Machine Code:	0 1 0 1 0 0 1 0 | 0 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Set/reset STOP mode wake-up enable flag for port RC
Description:	The data specified by I cause a wake-up from the STOP mode. The falling-edge signal on port RC can be specified independently.

I0~I7	Falling edge signal at port RC
I0 = 1	RC0
I1 = 1	RC1
I2 = 1	RC2
I3 = 1	RC3

Instruction Set Table 2, continued

MOV TM0, #I	**Timer 0 set**
Machine Code:	0 0 0 1 0 0 0 0	I7 I6 I5 I4 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Timer 0 set
Description:	The data specified by I7 to I0 is loaded to the Timer 0 to start the timer.
MOV TM0L, R	**Move R content to TM0L**
Machine Code:	0 0 0 1 0 1 0 0	0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	TM0L ← (R)
Description:	The content of the data memory location addressed by R6 to R0 are loaded into the TM0L.
MOV TM0H, R	**Move R content to TM0H**
Machine code:	0 0 0 1 0 1 0 0	1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	TM0H ← (R)
Description:	The content of the data memory location addressed by R6 to R0 are loaded into the TM0H.
MOV TM1, #I	**Timer 1 set**
Machine Code:	0 0 0 1 0 0 0 1	I7 I6 I5 I4 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Timer 1 set
Description:	The data specified by I7 to I0 is loaded to the Timer 1 to start the timer.

Instruction Set Table 2, continued

MOV TM1L, R	**Move R content to TM1L**
Machine Code:	0 0 0 1 0 1 0 1 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	TM1L ← (R)
Description:	The content of the data memory location addressed by R6 to R0 are loaded into the TM1L.
MOV TM1H, R	**Move R content to TM1H**
Machine code:	0 0 0 1 0 1 0 1 | 1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	TM1H ← (R)
Description:	The content of the data memory location addressed by R6 to R0 are loaded into the TM1H.
MOV WR, R	**Move R content to WR**
Machine Code:	1 1 1 0 1 W3 W2 W1 | W0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	WR ← (R)
Description:	The contents of the data memory location addressed by R6 to R0 are loaded to the WR.
MOV WR, @R	**Indirect load from R to WR**
Machine Code:	1 1 0 0 1 W3 W2 W1 | W0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	2
Operation:	WR ← [PR (bit2, bit1, bit0) × 10H + (R)]
Description:	The data memory contents of address [PR (bit2, bit1, bit0) × 10H + (R)] are loaded to the WR.

Instruction Set Table 2, continued

MOV @R, WR	**Indirect load from WR to R**
Machine Code:	1 1 0 1 1 W3 W2 W1 | W0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	2
Operation:	[PR (bit2, bit1, bit0) × 10H + (R)] ← WR
Description:	The contents of the WR are loaded to the data memory location addressed by [PR (bit2, bit1, bit0) × 10H + (R)] .
MOV PAGE, R	**Move R content to Page Register**
Machine Code:	0 1 0 1 1 1 1 0 | 1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	PR ← (R)
Description:	The contents of the data memory location addressed by R6 to R0 are loaded to the PR.
MOVA R, CF	**Move CF content to ACC.0 & R.0**
Machine Code:	0 1 0 1 1 0 0 1 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC.0, R.0 ← (CF)
Description:	The content of CF is loaded to bit 0 of the data memory location addressed by R6 to R0 and the ACC. The other bits of the data memory and ACC are reset to "0."
Flag Affected:	ZF

Instruction Set Table 2, continued

MOVA R, HCFH	**Move HCF4~7 to ACC & R**
Machine Code:	0 1 0 0 1 0 0 1	1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← HCF4~7
Description:	The contents of HCF bit 4 to bit 7 (HCF4 to HCF7) are loaded to the data memory location addressed by R6 to R0 and the ACC. The ACC contents and the meaning of the bits after execution of this instruction are as follows: Bit 0: HCF4: "1" when the HOLD mode is released by the falling edge signal at the INT pin. Bit 1: HCF5: "1" when the HOLD mode is released by underflow from Timer 1. Bit 2: HCF6: "1" when the HOLD mode is released by the serial port receiving completely Bit 3: HCF7: "1" when the HOLD mode is released by the serial port transmitting completely.
Flag Affected:	ZF
MOVA R, HCFL	**Move HCF0~3 to ACC & R**
Machine Code:	0 1 0 0 1 0 0 1	0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← HCF0~3
Description:	The contents of HCF bit 0 to bit 3 (HCF0 to HCF3) are loaded to the data memory location addressed by R6 to R0 and the ACC. The ACC contents and the meaning of the bits after execution of this instruction are as follows: Bit 0: HCF0: "1" when the HOLD mode is released by overflow from the Divider 0. Bit 1: HCF1: "1" when the HOLD mode is released by underflow from Timer 0. Bit 2: HCF2: "1" when the HOLD mode is released by a signal change on port RC. Bit 3: Reserved.
Flag Affected:	ZF

Instruction Set Table 2, continued

MOVA R, PAGE	**Move Page Register content to ACC & R**
Machine Code:	0 1 0 1 1 1 1 1	1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC , R ← (Page Register)
Description:	The contents of the Page Register (PR) are loaded to the data memory location addressed by R6 to R0 and the ACC.
Flag Affected:	ZF
MOVA R, PSR0	**Move Port Status Register 0 content to ACC & R**
Machine Code:	0 1 0 0 1 1 1 1	0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← RC port signal change flag (PSR0)
Description:	The contents of the RC port signal change flag (PSR0) are loaded to the data memory location addressed by R6 to R0 and the ACC. When the signal changes on any pin of the RC port, the corresponding signal change flag should be set to 1. Otherwise, it should be 0.
Flag Affected:	ZF
MOVA R, PSR2	**Move Port Status Register 0 content to ACC & R**
Machine Code:	0 1 0 1 1 1 1 1	0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← Serial I/O port status flags (PSR2)
Description:	The contents of the serial I/O port status flags (PSR2) are loaded to the data memory location addressed by R6 to R0 and the ACC.
Flag Affected:	ZF

Instruction Set Table 2, continued

MOVA R, WR	**Move WR content to ACC & R**
Machine Code:	0 1 1 1 1 W3 W2 W1 | W0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (WR)
Description:	The contents of the WR are loaded to the ACC and the data memory location addressed by R6 to R0.
Flag Affected:	ZF
MOVA WR, R	**Move R content to ACC & WR**
Machine Code:	0 1 1 0 1 W3 W2 W1 | W0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, WR ← (R)
Description:	The contents of the data memory location addressed by R6 to R0 are loaded to the WR and the ACC.
Flag Affected:	ZF
MOV TABL, R	**Move R content to TABL**
Machine Code:	1 0 0 1 1 0 0 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	TABL ← (R)
Description:	The content of the data memory location addressed by R6 to R0 are loaded into the TABL.
MOV TABH, R	**Move R content to TABH**
Machine code:	1 0 0 1 1 0 0 0 | 1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	TABH ← (R)
Description:	The content of the data memory location addressed by R6 to R0 are loaded into the TABH.

Instruction Set Table 2, continued

MOVC R	**Move look-up table ROM addressed by TABL and TABH to R**
Machine code:	1 0 0 1 1 0 0 1 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	2
Operation:	WR ← [(TABH) × 10H + (TABL)]
Description:	The contents of the look-up table ROM location addressed by TABH and TABL are loaded to R.
NOP	**No Operation**
Machine Code:	0 0 0 0 0 0 0 0 | 0 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	No Operation
ORL R, ACC	**OR R to ACC**
Machine Code:	0 0 1 1 1 0 1 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC ← (R) ∧ (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and the ACC are ORed and the result is loaded into the ACC.
Flag Affected:	ZF
ORL WR , #I	**OR immediate data to WR**
Machine Code:	0 0 1 1 1 1 1 0 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC ← (WR) ∧ I
Description:	The contents of the Working Register (WR) and the immediate data I are ORed and the result is loaded into the ACC.
Flag Affected:	ZF

Instruction Set Table 2, continued

ORLR R, ACC	**OR R to ACC**
Machine Code:	0 0 1 1 1 0 1 1 / 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) ∧ (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and the ACC are ORed and the result is placed in the data memory and the ACC.
Flag Affected:	ZF
ORLR WR , #I	**OR immediate data to WR**
Machine Code:	0 0 1 1 1 1 1 1 / I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC, WR ← (WR) ∧ I
Description:	The contents of the Working Register(WR) and the immediate data I are ORed and the result is placed in the WR and the ACC.
Flag Affected:	ZF
RLC R	**Rotate Left R with CF**
Machine Code:	0 1 0 0 1 1 0 0 / 1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC.n, R.n ← (R.n-1); ACC.0, R.0 ← CF; CF ← R.3
Description:	The contents of the ACC and the data memory location addressed by R6 to R0 are rotated left one bit, bit 3 is rotated into CF, and CF rotated into bit 0 (LSB). The same contents are loaded into the ACC.
Flag Affected:	CF & ZF

Instruction Set Table 2, continued

RRC R	**Rotate Right R with CF**
Machine Code:	0 1 0 0 1 1 0 1 | 1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC.n, R.n ← (R.n+1); ACC.3, R.3 ← CF; CF ← R.0
Description:	The contents of the ACC and the data memory location addressed by R6 to R0 are rotated right one bit, bit 0 is rotated into CF, and CF is rotated into bit 3 (MSB). The same contents are loaded into the ACC.
Flag Affected:	CF & ZF
RTN	**Return from subroutine**
Machine Code:	0 0 0 0 0 0 0 1 | 0 0 0 0 0 0 0 0
Machine Cycle:	1
Operation:	(PC) ← STACK
Description:	The program counter (PC10 to PC0) is restored from the stack. A return from a subroutine occurs.
SBC R, ACC	**Subtract ACC from R with Borrow**
Machine Code:	0 0 0 0 1 0 1 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC ← (R) - (ACC) - (CF)
Description:	The contents of the ACC and CF are binary subtracted from the contents of the data memory location addressed by R6 to R0 and the result is loaded into the ACC.
Flag Affected:	CF & ZF

Instruction Set Table 2, continued

SBC WR, #I	**Subtract immediate data from WR with Borrow**
Machine Code:	0 0 0 0 1 1 1 0	I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC ← (WR) - I - (CF)
Description:	The immediate data I and CF are binary subtracted from the contents of the WR and the result is loaded into the ACC.
Flag Affected:	CF & ZF
SBCR R, ACC	**Subtract ACC from R with Borrow**
Machine Code:	0 0 0 0 1 0 1 1	0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) - (ACC) - (CF)
Description:	The contents of the ACC and CF are binary subtracted from the contents of the data memory location addressed by R6 to R0 and the result is placed in the ACC and the data memory.
Flag Affected:	CF & ZF
SBCR WR, #I	**Subtract immediate data from WR with Borrow**
Machine Code:	0 0 0 0 1 1 1 1	I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC, R ← (WR) - I - (CF)
Description:	The immediate data I and CF are binary subtracted from the contents of the WR and the result is placed in the ACC and the WR.
Flag Affected:	CF & ZF

Instruction Set Table 2, continued

SET CF	**Set CF**
Machine Code:	0 1 0 1 0 0 0 0 | 0 1 0 0 0 0 0 0
Machine Cycle:	1
Operation:	Set CF
Description:	Set Carry Flag to 1.
Flag Affected:	CF
SET PMF, #I	**Set ParaMeter Flag**
Machine Code:	0 0 0 1 0 1 1 0 | 0 0 0 0 I3 I2 I1 I0
Machine Cycle:	1
Operation:	Set Parameter Flag
Description:	Description of each flag: I0, I1, I2 : Reserved I3 = 1 : The input clock of the watchdog timer is Fosc/16384.
SHLC R	**SHift Left R with CF and LSB = 0**
Machine Code:	0 1 0 0 1 1 0 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC.n, R.n ← (R.n-1); ACC.0, R.0 ← 0; CF ← R.3
Description:	The contents of the ACC and the data memory location addressed by R6 to R0 are shifted left one bit, but bit 3 is shifted into CF, and bit 0 (LSB) is replaced with "0." The same contents are loaded into the ACC.
Flag Affected:	CF & ZF

Instruction Set Table 2, continued

SHRC R	**SHift Right R with CF and MSB = 0**
Machine Code:	0 1 0 0 1 1 0 1 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC.n, R.n ← (R.n+1); ACC.3, R.3 ← 0; CF ← R.0
Description:	The contents of the ACC and the data memory location addressed by R6 to R0 are shifted right one bit, but bit 0 is shifted into CF, and bit 3 (MSB) is replaced with "0." The same contents are loaded into the ACC.
Flag Affected:	CF & ZF
SKB0 R	**If bit 0 of R is equal to 1 then skip**
Machine Code:	1 0 0 0 1 0 0 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	PC ← (PC) + 2; if R.0 = 1"1"
Description:	If bit 0 of R is equal to 1, the program counter is incremented by 2 and a skip is produced. If bit 0 of R is not equal to 1, the program counter (PC) is incremented.
SKB1 R	**If bit 1 of R is equal to 1 then skip**
Machine Code:	1 0 0 0 1 0 0 0 | 1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	PC ← (PC) + 2; if R.1 = 1"1"
Description:	If bit 1 of R is equal to 1, the program counter is incremented by 2 and a skip is produced. If bit 1 of R is not equal to 1, the program counter (PC) is incremented.

Instruction Set Table 2, continued

SKB2 R	**If bit 2 of R is equal to 1 then skip**
Machine Code:	1 0 0 0 1 0 1 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	PC ← (PC) + 2; if R.2 = 1"1"
Description:	If bit 2 of R is equal to 1, the program counter is incremented by 2 and a skip is produced. If bit 2 of R is not equal to 1. The program counter (PC) is incremented.
SKB3 R	**If bit 3 of R is equal to 1 then skip**
Machine Code:	1 0 0 0 1 0 1 0 | 1 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	PC ← (PC) + 2; if R.3 = 1"1"
Description:	If bit 3 of R is equal to 1, the program counter is incremented by 2 and a skip is produced. If bit 3 of R is not equal to 1, the program counter (PC) is incremented.
STOP	**Enter the STOP mode**
Machine Code:	0 0 0 0 0 0 0 0 | 1 1 0 0 0 0 0 0
Machine Cycle:	1
Operation:	STOP oscillator
Description:	Device enters STOP mode. When the falling edge signal of RC port is accepted, the µC will wake up and execute the next instruction.

Instruction Set Table 2, continued

SUB R, ACC	**Subtract ACC from R**
Machine Code:	0 0 0 1 1 0 1 0	0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC ← (R) - (ACC)
Description:	The contents of the ACC are binary subtracted from the contents of the data memory location addressed by R6 to R0 and the result is loaded into the ACC.
Flag Affected:	CF & ZF
SUB WR , #I	**Subtract immediate data from WR**
Machine Code:	0 0 0 1 1 1 1 0	I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC ← (WR) - I
Description:	The immediate data I are binary subtracted from the contents of the WR and the result is loaded into the ACC.
Flag Affected:	CF & ZF
SUBR R, ACC	**Subtract ACC from R**
Machine Code:	0 0 0 1 1 0 1 1	0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) - (ACC)
Description:	The contents of the ACC are binary subtracted from the contents of the data memory location addressed by R6 to R0 and the result is placed in the ACC and the data memory.
Flag Affected:	CF & ZF

Instruction Set Table 2, continued

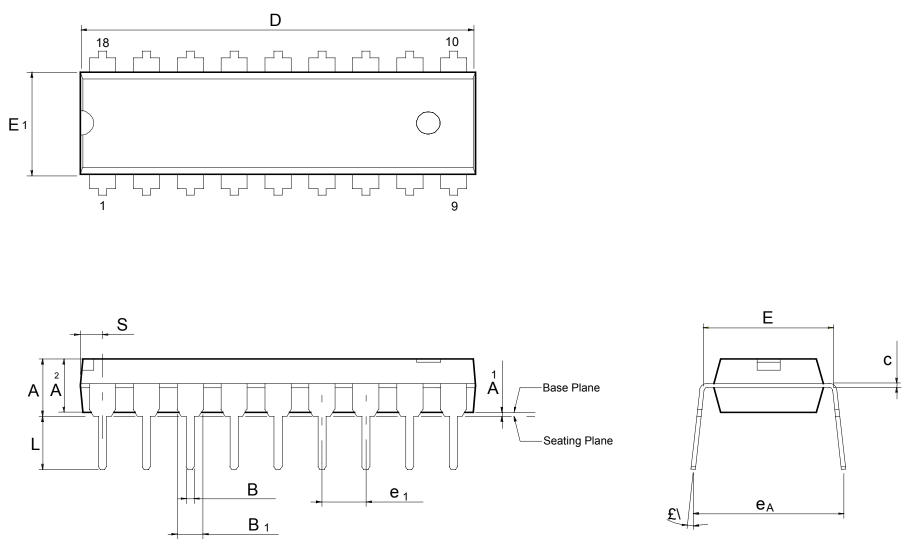
SUBR WR, #I	**Subtract immediate data from WR**
Machine Code:	0 0 0 1 1 1 1 1 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC, WR ← (WR) - I
Description:	The immediate data I are binary subtracted from the contents of the WR and the result is placed in the ACC and the WR.
Flag Affected:	CF & ZF
XRL R, ACC	**Exclusive OR R to ACC**
Machine Code:	0 0 1 1 1 0 0 0 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC ← (R) EX (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and the ACC are exclusive-ORed and the result is loaded into the ACC.
Flag Affected:	ZF
XRL WR, #I	**Exclusive OR immediate data to WR**
Machine Code:	0 0 1 1 1 1 0 0 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC ← (WR) EX I
Description:	The contents of the Working Register (WR) and the immediate data I are exclusive-ORed and the result is loaded into the ACC.
Flag Affected:	ZF

Instruction Set Table 2, continued

XRLR R, ACC	**Exclusive OR R to ACC**
Machine Code:	0 0 1 1 1 0 0 1 | 0 R6 R5 R4 R3 R2 R1 R0
Machine Cycle:	1
Operation:	ACC, R ← (R) EX (ACC)
Description:	The contents of the data memory location addressed by R6 to R0 and the ACC are exclusive-ORed and the result is placed in the data memory and the ACC.
Flag Affected:	ZF
XRLR WR, #I	**Exclusive OR immediate data to WR**
Machine Code:	0 0 1 1 1 1 0 1 | I3 I2 I1 I0 W3 W2 W1 W0
Machine Cycle:	1
Operation:	ACC, WR ← (WR) EX I
Description:	The contents of the Working Register(WR) and the immediate data I are exclusive-ORed and the result is placed in the WR and the ACC.
Flag Affected:	ZF

PACKAGE DIAMENSIONS

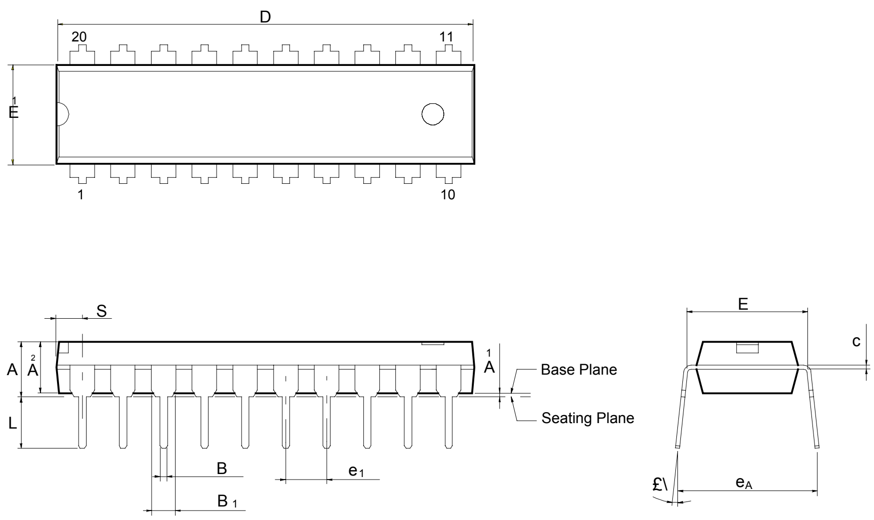
18-Lead PDIP (300 mil)



Symbol	Dimension in inch			Dimension in mm		
	Min.	Nom.	Max	Min	Nom.	Max.
A	—	—	0.175	—	—	4.45
A 1	0.010	—	—	0.25	—	—
A 2	0.125	0.130	0.135	3.18	3.30	3.43
B	0.016	0.018	0.022	0.41	0.46	0.56
B1	0.058	0.060	0.064	1.47	1.52	1.63
c	0.008	0.010	0.014	0.20	0.25	0.36
D	—	0.900	0.910	—	22.86	23.11
E	0.290	0.300	0.310	7.37	7.62	7.87
E1	0.245	0.250	0.255	6.22	6.35	6.48
e 1	0.090	0.100	0.110	2.29	2.54	2.79
L	0.120	0.130	0.140	3.05	3.30	3.56
£\	0	—	15	0	—	15
e A	0.335	0.355	0.375	8.51	9.02	9.53
S	—	—	0.055	—	—	1.40

Package Dimensions, continued

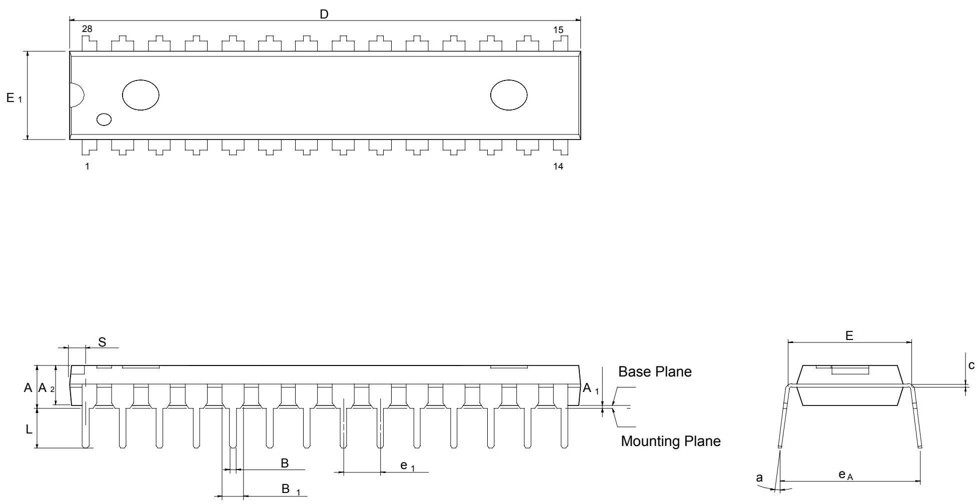
20-Lead PDIP



Symbol	Dimension in inch			Dimension in mm		
	Min.	Nom.	Max.	Min.	Nom.	Max.
A	—	—	0.175	—	—	4.45
A_1	0.010	—	—	0.25	—	—
A_2	0.125	0.130	0.135	3.18	3.30	3.43
B	0.016	0.018	0.022	0.41	0.46	0.56
B_1	0.058	0.060	0.064	1.47	1.52	1.63
c	0.008	0.010	0.014	0.20	0.25	0.36
D	—	1.026	1.040	—	20.06	26.42
E	0.290	0.300	0.310	7.37	7.62	7.87
E_1	0.245	0.250	0.255	6.22	6.35	6.48
e_1	0.090	0.100	0.110	2.29	2.54	2.79
L	0.120	0.130	0.140	3.05	3.30	3.56
£\	0	—	15	0	—	15
e_A	0.335	0.355	0.375	8.51	9.02	9.53
S	—	—	0.075	—	—	1.91

Package Dimensions, continued

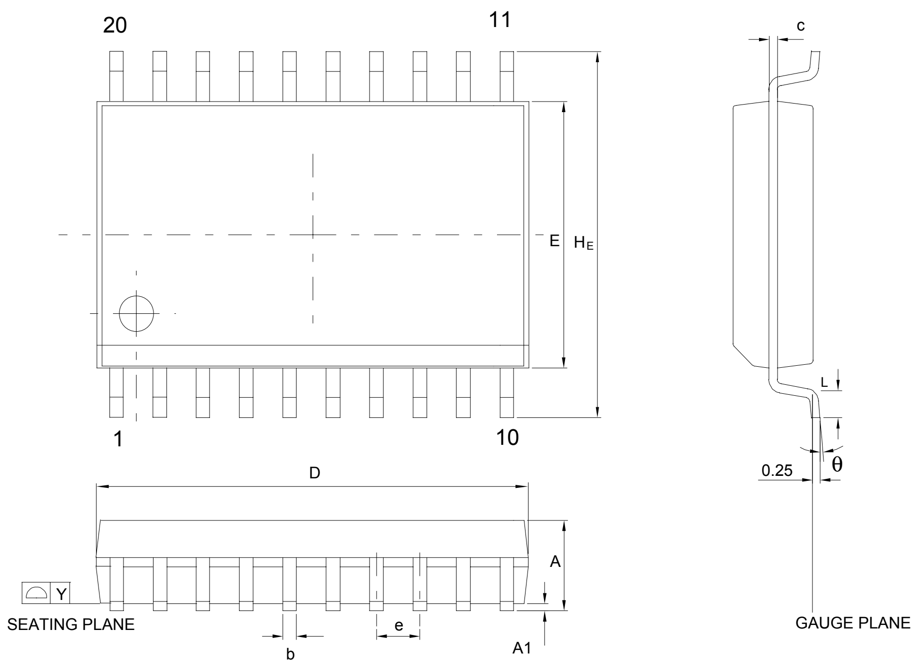
28-Lead PDIP Skinny



Symbol	Dimension in Inches			Dimension in mm		
	Min.	Nom.	Max.	Min.	Nom.	Max.
A	—	—	0.175	—	—	4.45
A_1	0.010	—	—	0.25	—	—
A_2	0.125	0.130	0.135	3.18	3.30	3.43
B	0.016	0.018	0.022	0.41	0.46	0.56
B_1	0.058	0.060	0.064	1.47	1.52	1.63
c	0.008	0.010	0.014	0.20	0.25	0.36
D	—	1.388	1.400	—	35.26	35.56
E	0.300	0.310	0.320	7.62	7.87	8.13
E_1	0.283	0.288	0.293	7.19	7.32	7.44
e_1	0.090	0.100	0.110	2.29	2.54	2.79
L	0.120	0.130	0.140	3.05	3.30	3.56
a	0°	—	15°	0°	—	15°
e_A	0.330	0.350	0.370	8.38	8.89	9.40
S	—	—	0.055	—	—	1.40

Package Dimensions, continued

20-Lead SOP (300mil)

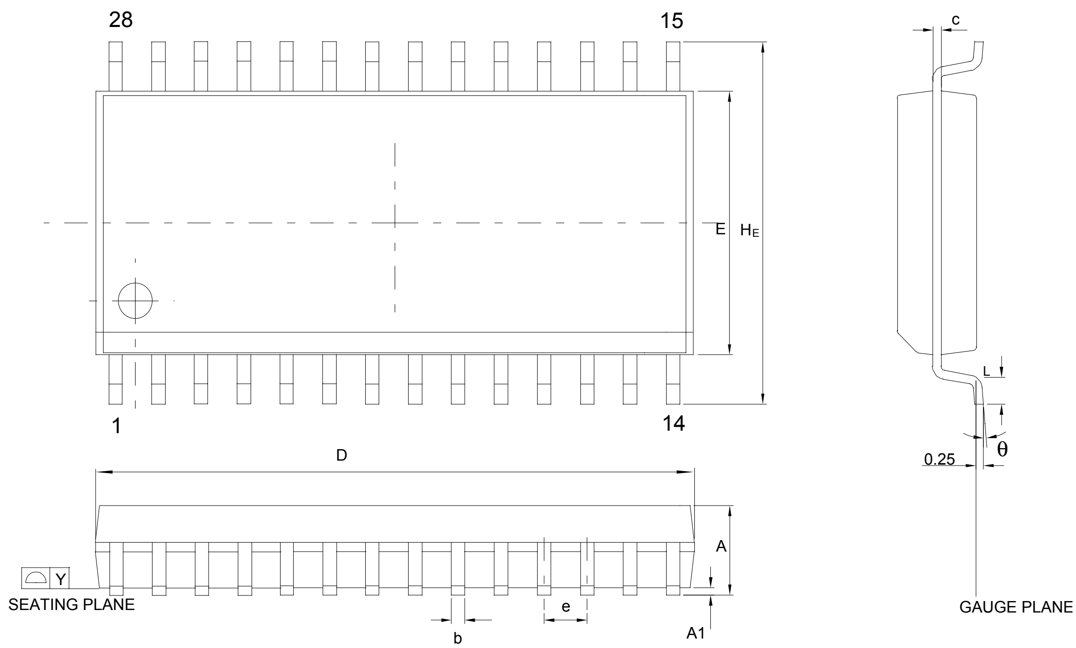


Control demensions are in milmeters.

Symbol	Dimension in mm		Dimension in Inches	
	Min.	Max.	Min.	Max.
A	2.35	2.65	0.093	0.104
A1	0.10	0.30	0.004	0.012
b	0.33	0.51	0.013	0.020
c	0.23	0.32	0.009	0.013
E	7.40	7.60	0.291	0.299
D	12.60	13.00	0.496	0.512
e	1.27 BSC		0.050 BSC	
H_E	10.00	10.65	0.394	0.419
Y	—	0.10	—	0.004
L	0.40	1.27	0.016	0.050
θ	0	8	0	8

Package Dimensions, continued

28-Lead SOP (300 mil)



Control demensions are in milmeters .

Symbol	Dimension in mm		Dimension in Inches	
	Min.	Max.	Min.	Max.
A	2.35	2.65	0.093	0.104
A1	0.10	0.30	0.004	0.012
b	0.33	0.51	0.013	0.020
c	0.23	0.32	0.009	0.013
E	7.40	7.60	0.291	0.299
D	17.70	18.10	0.697	0.713
e	1.27 BSC		0.050 BSC	
H_E	10.00	10.65	0.394	0.419
Y	—	0.10	—	0.004
L	0.40	1.27	0.016	0.050
θ	0	8	0	8

Notes:

Headquarters
No. 4, Creation Rd. III,
Science-Based Industrial Park,
Hsinchu, Taiwan
TEL: 886-3-5770066
FAX: 886-3-5792766
http://www.winbond.com.tw/
Voice & Fax-on-demand: 886-2-27197006

Taipei Office
11F, No. 115, Sec. 3, Min-Sheng East Rd.,
Taipei, Taiwan
TEL: 886-2-27190505
FAX: 886-2-27197502

Winbond Electronics (H.K.) Ltd.
Rm. 803, World Trade Square, Tower II,
123 Hoi Bun Rd., Kwun Tong,
Kowloon, Hong Kong
TEL: 852-27513100
FAX: 852-27552064

Winbond Electronics North America Corp.
Winbond Memory Lab.
Winbond Microelectronics Corp.
Winbond Systems Lab.
2727 N. First Street, San Jose,
CA 95134, U.S.A.
TEL: 408-9436666
FAX: 408-5441798