十速科技股份有限公司
tenx technology inc.

Advance Information

TM59PA20

User's Manual

Tenx reserves the right to change or discontinue this product without notice.

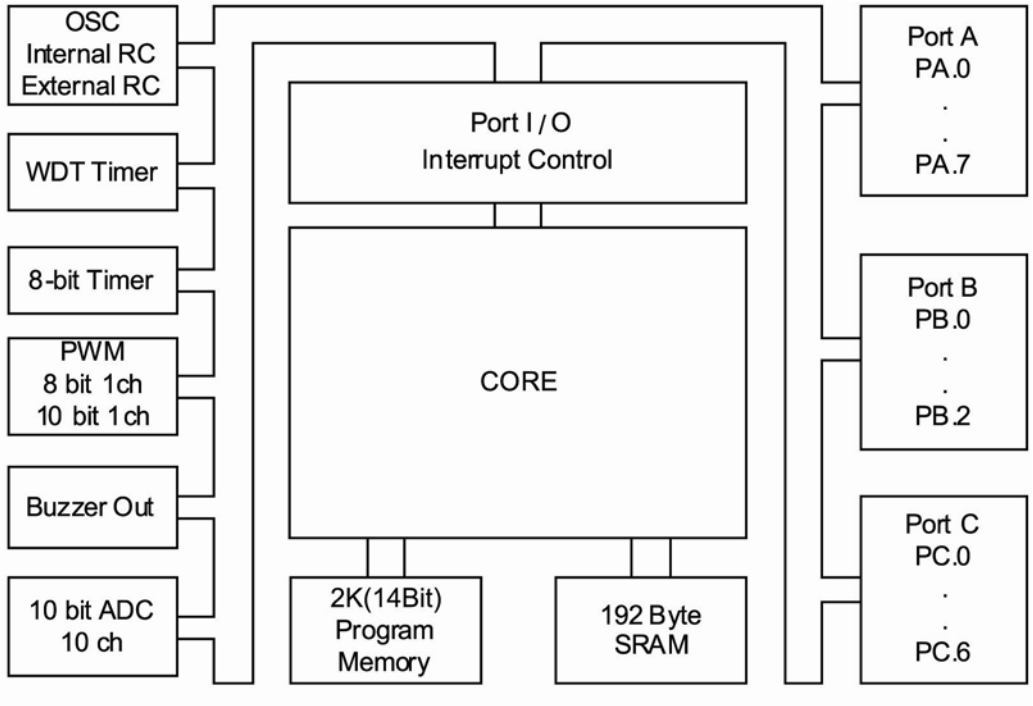
tenx technology, inc.

Contents

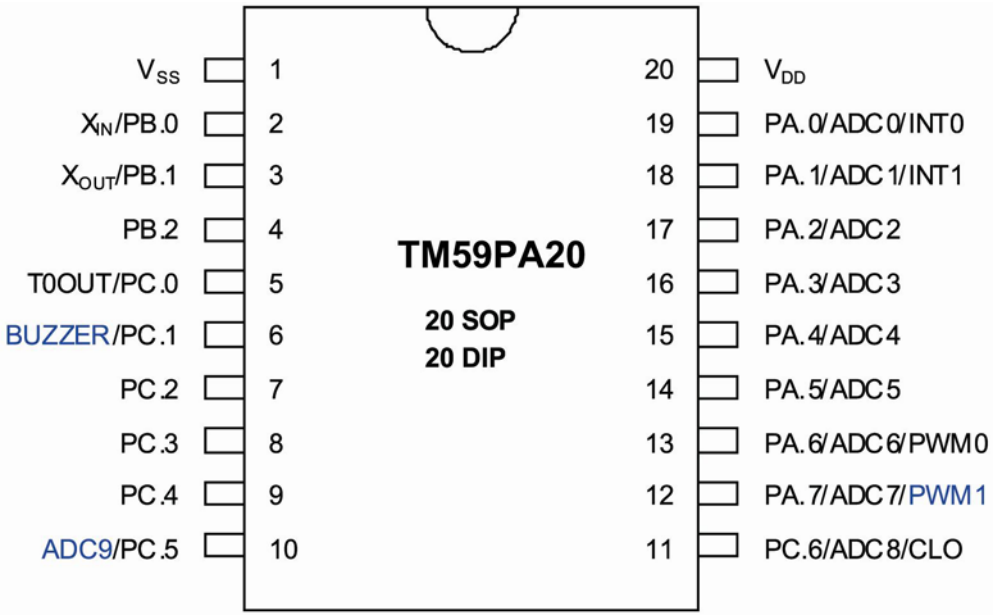
1. Overview

1.1 FEATURE

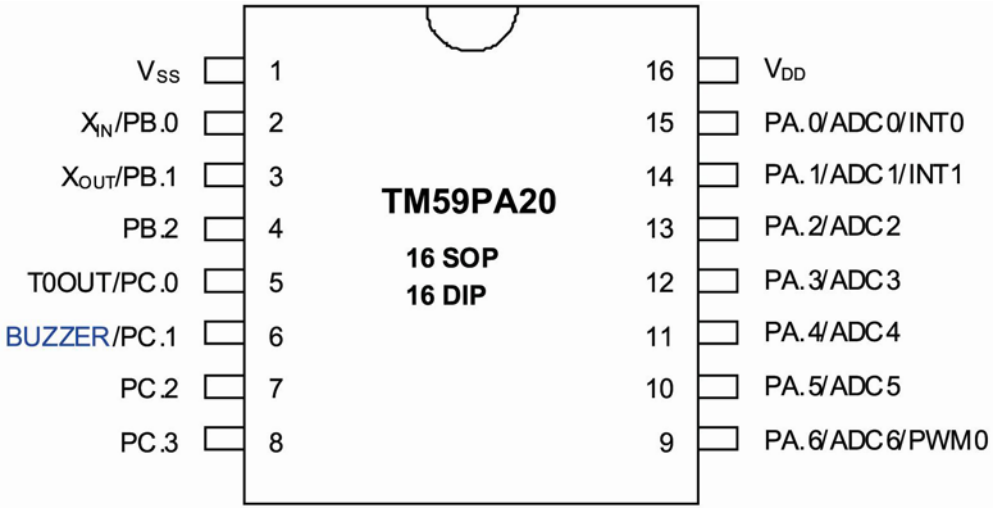
1. ROM: 2K x 14 bits
2. RAM: 192 x 8 bits
3. STACK: 6 Levels
4. I/O ports: Three I/O ports (Max 18 pins) and Bit programmable ports
5. Timer/Counter: One 8-bit timer/counter with time interval modes
6. Watchdog Timer: On chip WDT based on system oscillator
7. Power-On Reset & Watchdog timer overflow Reset & Low Voltage reset
8. Oscillation Frequency:
   - 1 MHz to 12 MHz external crystal oscillator
   - Internal RC: 2.8 MHz (typ.), 470KHz (typ.) in VDD = 5 V
   - External RC
9. High-speed PWM:
   - 8-bit PWM 1-ch, 6-bit base + 2-bit extension (Max: 187 kHz)
   - 10-bit PWM 1-ch, 8-bit base + 2-bit extension (Max: 47 kHz)
10. Operation Voltage: LVR to 5.5V
11. Instruction set: 35 Instructions
12. Execution Time: 167 ns at 12 MHz f_{OSC}
13. A/D Converter: 10-bit conversion resolution with 10-ch analog input pins (MAX)
14. Interrupts: 5 interrupt sources with one vector with one interrupt level
15. Buzzer Out: Frequency Selectable Buzzer Output
16. System Config Option: LVR Level Selection and Clock Source Selection
17. Reset vector: 000H
18. Interrupt vector: 001H
19. Power Down mode
20. Package Types:
    - 20-SOP, DIP
    - 16-SOP, DIP



<Figure 1-1. System Block Diagram>



<Figure 1-2. Pin Assignment Diagram _ Package Types: 20-Pin SOP/DIP>



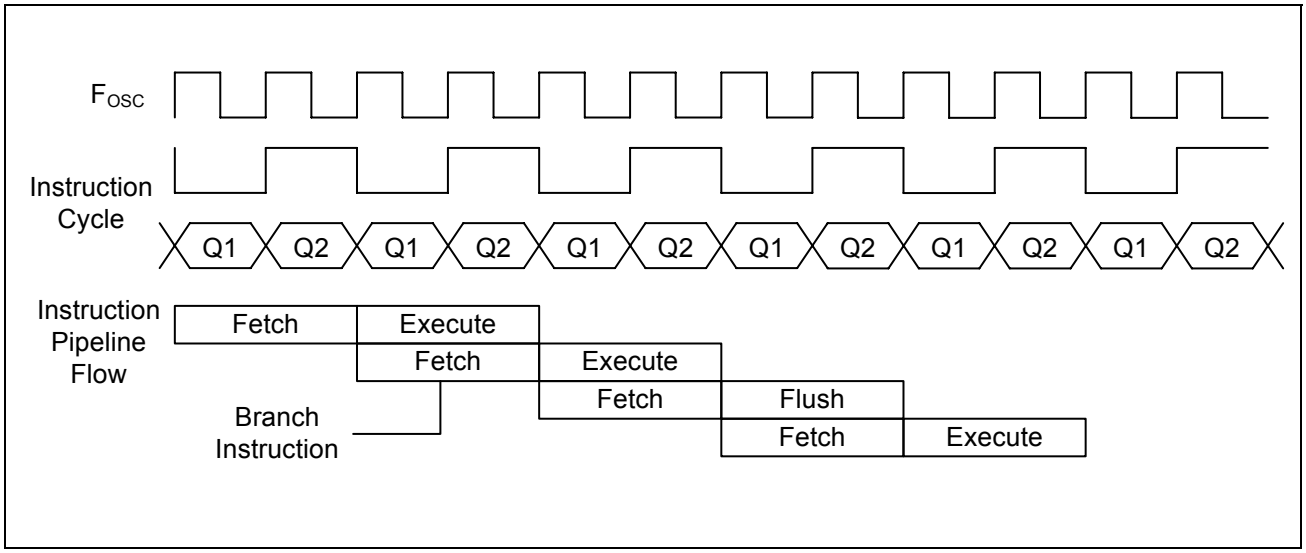
<Figure 1-3. Pin Assignment Diagram _ Package Types: 16-Pin SOP/DIP>

Name	In/Out	Pin Description	Shared Function
PA.0–PA.7	I/O	Bit-programmable I/O port for Schmitt-trigger input or push-pull output. Pull-up resistors are assignable by software. PortA pins can also be used as A/D converter input, PWM output or external interrupt input.	ADC0-ADC7 INT0/INT1 PWM0/PWM1
PB.0–PB.1	I/O	Bit-programmable I/O port for Schmitt-trigger input or push-pull, open-drain output. Pull-up resistors or pull-down resistors are assignable by software.	X_{IN}, X_{OUT}
PB.2	I	Schmitt trigger input port	–
PC.0–PC.6	I/O	Bit-programmable I/O port for Schmitt-trigger input or push-pull, open-drain output. Pull-up resistors are assignable by software.	ADC8-9/CLO T0OUT/BUZZER
X_{IN}, X_{OUT}	–	Crystal/Ceramic, or RC oscillator signal for system clock.	PB.0–PB.1
V_{DD}, V_{SS}	P	Voltage input pin and ground	–
CLO	O	System clock output port	PC.6
INT0–INT1	I	External interrupt input port	PA.0, PA.1
PWM0	O	8-Bit high speed PWM output	PA.6
PWM1	O	10-Bit high speed PWM output	PA.7
T0OUT	O	Timer0 match output	PC.0
ADC0–ADC9	I	A/D converter input	PA.0–PA.7 PC.5–PC.6

<Table 1-1. PIN Description> < I: Input; O: Output; I/O: Bi-direction; P: Power >

1.2 Clock Scheme and Instruction Cycle

The clock input (X_{IN}) is internally divided by two to generate Q1 state and Q2 state for each instruction cycle. The Programming Counter (PC) is updated at Q1 and the instruction is fetched from program ROM and latched into the instruction register in Q2. It is then decoded and executed during the following Q1-Q2 cycle.



< Figure 1-4. Clock/Instruction cycle and pipeline >

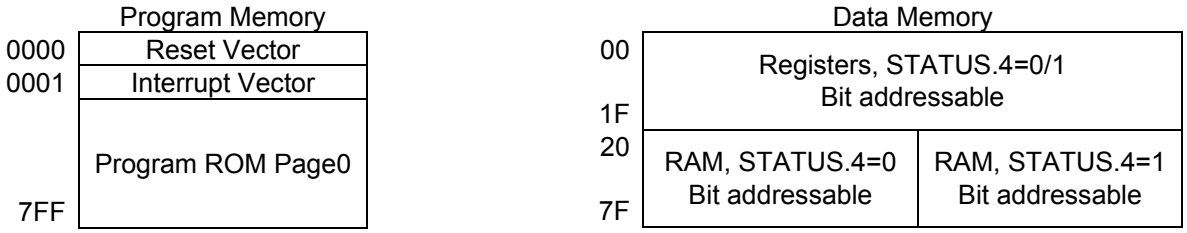
Branch instructions take two cycle since the fetch instruction is 'flushed' from the pipeline, while the new instruction is being feched and then executed.

1.3 Addressing Mode

The Programming Counter is 11-bit wide capable of addressing a 2K x 14 program ROM. As a program instruction is executed, the PC will contain the address of the next program instruction to be executed. The PC value is normally increased by one except the followings. The Reset Vector (000h) and the Interrupt Vector (001h) are provided for PC initialization and Interrupt. For CALL/GOTO instructions, PC loads 11 bits address from instruction word. For RET/RETI/RETLW instructions, PC retrieves its content from the top level STACK. For the other instructions updating PC[7:0], the PC[10:8] keeps unchanged. The STACK is 11-bit wide and 6-level in depth. The CALL instruction and Hardware interrupt will push STACK level in order, While the RET/RETI/RETLW instruction pops the STACK level in order.

The data memory is partitioned into two banks, which contain the General Purpose Data Memory and the Special Function Registers (SFR). STATUS.4 is the bank select bits. Each bank extends up to 7Fh (128 bytes). The lower locations of each bank (00h-1Fh) are reserved for the SFR. Above the SFR is General Purpose Data Memory, implemented as static RAM. SFR area is mirrored in all banks for code reduction and quicker access. The first half of RAM (00h – 3Fh) is bit-addressable.

Data memory can be addressed directly or indirectly. Indirect Addressing is made by INDF register. The INDF register is not a physical register. Addressing INDF actually addresses the register whose address is contained in the FSR register (FSR is a pointer). Reading INDF itself indirectly (FSR=0) will produce 00h. Writing to the INDF register indirectly results in a no-operation.



< Figure 1-5. Address space >

1.4 ALU and Working (W) Register

The ALU is 8 bits wide and capable of addition, subtraction, shift and logical operations. In two-operand instructions, typically one operand is the W register, which is an 8-bit non-addressable register used for ALU operations. The other operand is either a file register or an immediate constant. In single operand instructions, the operand is either W register or a file register. Depending on the instruction executed, the ALU may affect the values of Carry (C), Digit Carry (DC), and Zero (Z) Flags in the STATUS register. The C and DC flags operate as a /Borrow and /Digit Borrow, respectively, in subtraction.

1.5 STATUS Register

This register contains the arithmetic status of ALU and the Bank select for RAM. The STATUS register can be the destination for any instruction, as with any other register. If the STATUS register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. It is recommended, therefore, that only BCF, BSF and MOVWF instructions be used to alter the STATUS Register because these instructions do not affect those bits.

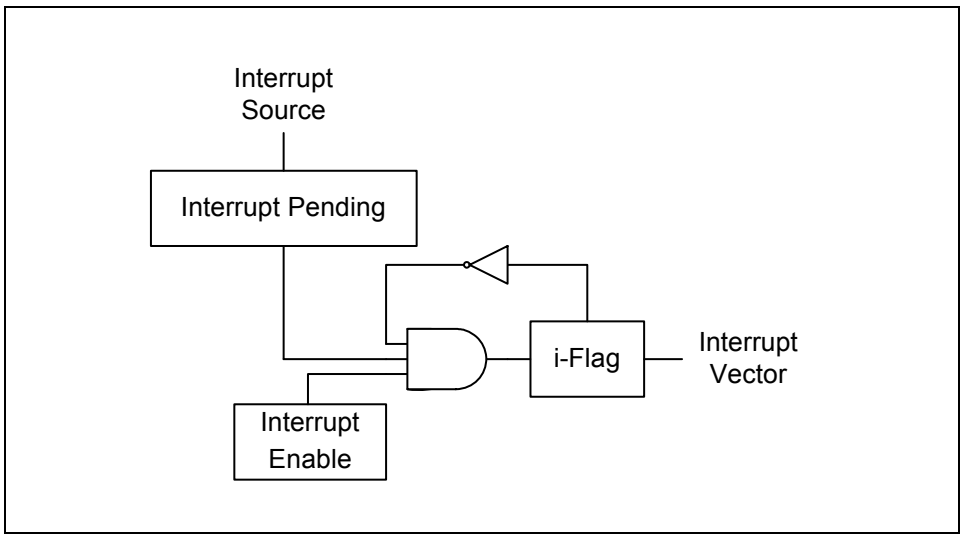
STATUS	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Reset Value	–	–	–	0	–	0	0	0
R/W	–	–	–	R/W	–	R/W	R/W	R/W

Bit	Description	
7-5	Not Used (Must be set to 0)	
4	SRAM: SRAM Bank Selection Bit 0: Page 0 1: Page 1	
3	Not Used (Must be set to 0)	
2	Zero Flag (Z) 0: the result of a logic operation is not zero 1: the result of a logic operation is zero	
1	Decimal Carry Flag or Decimal/Borrow Flag (DC)	
	ADD instruction	SUB instruction
	1: a carry from the low nibble bits of the result occurred 0: no carry	1: no borrow 0: a borrow from the low nibble bits of the result occurred
0	Carry Flag(C) or Borrow Flag	
	ADD instruction	SUB instruction
	1: a carry occurred from the MSB 0: no carry	1: no borrow 0: a borrow occurred from the MSB

<Table 1-2. STATUS — System Flags Register (Address: 03H)>

1.6 Interrupt

The TM59PA20 has 1 level, 1 vector and 5 sources. Each interrupt source has its own enable control bit. An interrupt event will set its individual flag. Because TM59PA20 has only 1 vector, there is not a interrupt priority register. The interrupt priority is determined by F/W.

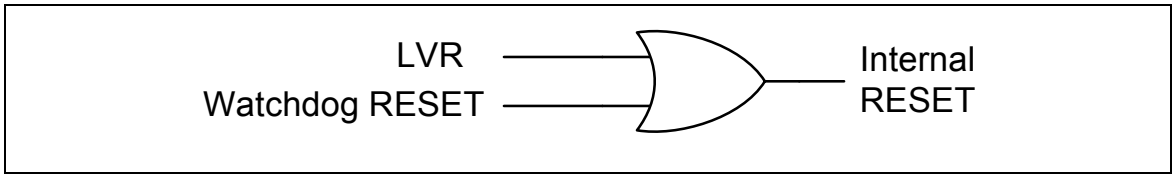


< Figure 1-6. Interrupt Function Diagram >

If the corresponding interrupt enable bit has been set (INTCON), it would trigger CPU to service the interrupt. CPU accepts interrupt in the end of current executed instruction cycle. In the mean while, A “CALL 0001” instruction is inserted to CPU, and the i-flag is set to prevent recursive interrupt nesting. The i-flag is cleared in the instruction after the “RETI” instruction. That is, at least one instruction in main program is executed before service the pending interrupt. The interrupt event is edge trigged. F/W must clear the interrupt event register while serves the interrupt routine.

1.7 Reset

The TM59PA20 can be RESET in four ways.
- Power-On-Reset
- Low Voltage Reset (LVR)
- Watchdog Reset



< Figure 1-7. Reset Circuit Diagram >

After the Power-On-Reset, all system and peripheral control registers are then set to their default hardware Reset values. And the clock source, LVR level is selected by SYSL register value. After the clock source selection, clock oscillation starts, and oscillation stabilization time must be needed. The minimum required oscillation stabilization time is approximately 2.5 ms (f_{OSC} = 10 MHz). The Low Voltage Reset features static reset when supply voltage is below a reference value. The four levels of reference voltage can be configured in SYSL register.

The Watchdog Timer is disabled after Reset. F/W can use the CLRWDT instruction to clear and enable the Watchdog Timer. If once enabled, the Watchdog Timer overflow and generate a chip reset signal if no

CLRWDT executed in a period of 2^{21} oscillator's cycle (0.25 Second for 8.192MHz crystal). The Watchdog Timer does not work in Power-down mode to provide wake-up function. It is only designed to prevent F/W goes into endless loop.

1.8 Power-Down Mode

The Power-down mode is activated by SLEEP instruction. During the Power-down mode, the crystal clock oscillation stops to minimize power consumption and all the peripherals are not working. Therefore, The Power down mode can be terminated by Reset or enabled external Interrupts (External Interrupt 0, 1). When the Power down mode is released, the clock circuit requires oscillation stabilization time also.

PWRDN	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Reset Value	–	–	–	–	–	–	–	–
R/W	–	–	–	–	–	–	–	–
Bit	Description							
7-0	Power Down Control Register							
	This register is not physical register. The device can enter STOP mode by writing any value into this register. The SLEEP instruction is equivalent to "MOVWF PWRDN".							

<Table 1-3. PWRDN — Power Down Control Register (Address: 0AH)>

1.9 System Config Register

The System Config Register (SYSL) is the ROM option for initial condition of the MCU. The address 2000H is virtual address which is not reachable in F/W. It can be written by MDS and system use only. You can config clock source, LVR reference voltage control by SYSL register. The default value of SYSL is 3FFFh. The 13th bit is code protection selection bit. If write this bit to 0, the data of ROM will be all 3FFFh, when user read ROM.

NAME	Bit 13	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SYSL	–	–	–	–	–	–	–	–	–
Reset Value	1	1	1	1	1	1	1	1	1
Bit	Description								
13	Code protection selection bit								
	1: No protect								
	0: Code protection								
7	Not Used (Must Set be '1')								
6-5	CSS1	CSS0	CSS1~0Clock Source Selection Bit						
	0	0	External crystal / ceramic oscillator						
	0	1	External RC						
	1	0	Internal RC (0.47 MHz in V_{DD} = 5 V)						
	1	1	Internal RC (2.8 MHz in V_{DD} = 5 V)						
4-0	LVS: LVR Level Selection Byte								
	11001		2.0V						
	11010		2.3V						
	10001		3.0V						
	01111		3.9V						

<Table 1-4. SYSL — System Config Register (Address: 2000H)>

1.10 Instruction Set

Each instruction is a 14-bit word divided into an OPCODE, which specified the instruction type, and one or more operands, which further specify the operation of the instruction. The instructions can be categorized as byte-oriented, bit-oriented and literal operations list in the following table.

For byte-oriented instructions, “f” represents address designator and “d” represents destination designator. The address designator is used to specify which address in Program memory is to be used by the instruction. The destination designator specifies where the result of the operation is to be placed. If “d” is “0”, the result is placed in the W register. If “d” is “1”, the result is placed in the address specified in the instruction.

For bit-oriented instructions, “b” represents a bit field designator, which selects the number of the bit affected by the operation, while “f” represents the address designator. For literal operations, “k” represents the literal or constant value.

Field	Description
f	Register File Address
b	Bit address
k	Literal. Constant data or label
d	Destination selection field. 0 : Working register 1 : Register file
W	Working Register
Z	Zero Flag
C	Carry Flag
DC	Decimal Carry Flag
PC	Program Counter
TOS	Top Of Stack
GIE	Global Interrupt Enable Flag (i-Flag)
( )	Contents
.	Bit Field
←	Assign direction

< Table 1-5. OP-CODE Field Description >

Mnemonic		Op Code	Cycle	Flag Affect	Description
Byte-Oriented File Register Instruction					
ADDWF	f,d	00 0111 dfff ffff	1	C,DC,Z	Add W and "f"
ANDWF	f,d	00 0101 dfff ffff	1	Z	AND W with "f"
CLRF	f	00 0001 1fff ffff	1	Z	Clear "f"
CLRW		00 0001 0100 0000	1	Z	Clear W
COMF	f,d	00 1001 dfff ffff	1	Z	Complement "f"
DECF	f,d	00 0011 dfff ffff	1	Z	Decrement "f"
DECFSZ	f,d	00 1011 dfff ffff	1 or 2	-	Decrement "f", skip if zero
INCF	f,d	00 1010 dfff ffff	1	Z	Increment "f"
INCFSZ	f,d	00 1111 dfff ffff	1 or 2	-	Increment "f", skip if zero
IORWF	f,d	00 0100 dfff ffff	1	Z	OR W with "f"
MOVFW	f	00 1000 0fff ffff	1	-	Move "f" to "w"
MOVWF	f	00 0000 1fff ffff	1	-	Move W to "f"
RLF	f,d	00 1101 dfff ffff	1	C	Rotate left "f" through carry
RRF	f,d	00 1100 dfff ffff	1	C	Rotate right "f" through carry
SUBWF	f,d	00 0010 dfff ffff	1	C,DC,Z	Subtract W from "f"
SWAPF	f,d	00 1110 dfff ffff	1	-	Swap nibbles in "f"
TESTZ	f	00 1000 1fff ffff	1	Z	Test if "f" is zero
XORWF	f,d	00 0110 dfff ffff	1	Z	XOR W with "f"
Bit-Oriented File Register Instruction					
BCF	f,b	01 000b bbff ffff	1	-	Clear "b" bit of "f"
BSF	f,b	01 001b bbff ffff	1	-	Set "b" bit of "f"
BTFSC	f,b	01 010b bbff ffff	1 or 2	-	Test "b" bit of "f", skip if clear
BTFSS	f,b	01 011b bbff ffff	1 or 2	-	Test "b" bit of "f", skip if set
Literal and Control Instruction					
ADDLW	k	01 1100 kkkk kkkk	1	C,DC,Z	Add Literal "k" and W
ANDLW	k	01 1011 kkkk kkkk	1	Z	AND Literal "k" with W
CALL	k	10 0kkk kkkk kkkk	2	-	Call subroutine "k"
CLRWDT		00 0000 1000 1001	1	-	Clear and enable Watch Dog Timer
GOTO	k	11 0kkk kkkk kkkk	2	-	Jump to branch "k"
IORLW	k	01 1010 kkkk kkkk	1	Z	OR Literal "k" with W
MOVLW	k	01 1001 kkkk kkkk	1	-	Move Literal "k" to W
NOP		00 0000 0000 0000	1	-	No operation
RET		00 0000 0100 0000	2	-	Return from subroutine
RETI		00 0000 0110 0000	2	-	Return from interrupt
RETLW	k	01 1000 kkkk kkkk	2	-	Return with Literal "k" in W
SLEEP		00 0000 1000 1010	1	-	Go into standby mode, Clock oscillation stops
XORLW	k	01 1111 kkkk kkkk	1	Z	XOR Literal "k" with W

< Table 1-6. Instruction Summary >

ADDLW Add Literal “k” and W

Syntax	ADDLW k
Operands	k : 00h ~ FFh
Operation	(W) ← (W) + k
Status Affected	C, DC, Z
OP-Code	01 1100 kkkk kkkk
Description	The contents of the W register are added to the eight-bit literal ’k’ and the result is placed in the W register.
Cycle	1
Example	ADDLW 0x15 B : W = 0x10 A : W = 0x25

ADDWF Add W and ‘f’

Syntax	ADDWF f [,d]
Operands	f : 00h ~ 7Fh d : 0, 1
Operation	(Destination) ← (W) + (f)
Status Affected	C, DC, Z
OP-Code	00 0111 dfff ffff
Description	Add the contents of the W register with register ‘f’. If ‘d’ is 0, the result is stored in the W register. If ‘d’ is 1, the result is stored back in register ‘f’.
Cycle	1
Example	ADDWF FSR, 0 B : W = 0x17, FSR = 0xC2 A : W = 0xD9, FSR = 0xC2

ANDLW Logical AND Literal "k" with W

Syntax	ANDLW k
Operands	k : 00h ~ FFh
Operation	(W) ← (W) ‘AND’ (f)
Status Affected	Z
OP-Code	01 1011 kkkk kkkk
Description	The contents of W register are AND’ed with the eight-bit literal 'k'. The result is placed in the W register.
Cycle	1
Example	ANDLW 0x5F B : W = 0xA3 A : W = 0x03

ANDWF AND W with f

Syntax	ANDWF f [,d]
Operands	f : 00h ~ 7Fh d : 0, 1
Operation	(Destination) ← (W) ‘AND’ (f)
Status Affected	Z
OP-Code	00 0101 dfff ffff
Description	AND the W register with register ’f’. If ’d’ is 0, the result is stored in the W register. If ’d’ is 1, the result is stored back in register ’f’.
Cycle	1
Example	ANDWF FSR, 1 B : W = 0x17, FSR = 0xC2 A : W = 0x17, FSR = 0x02

BCF	Clear "b" bit of "f"	
Syntax	BCF f [,b]	
Operands	f : 00h ~ 3Fh b : 0 ~ 7	
Operation	(f.b) ← 0	
Status Affected	-	
OP-Code	01 000b bbff ffff	
Description	Bit 'b' in register 'f' is cleared.	
Cycle	1	
Example	BCF FLAG_REG, 7	B : FLAG_REG = 0xC7 A : FLAG_REG = 0x47

BSF	Set "b" bit of "f"	
Syntax	BSF f [,b]	
Operands	f : 00h ~ 3Fh b : 0 ~ 7	
Operation	(f.b) ← 1	
Status Affected	-	
OP-Code	01 001b bbff ffff	
Description	Bit 'b' in register 'f' is set.	
Cycle	1	
Example	BSF FLAG_REG, 7	B : FLAG_REG = 0x0A A : FLAG_REG = 0x8A

BTFSC	Test 'b' bit of 'f', skip if clear(0)	
Syntax	BTFSC f [,b]	
Operands	f : 00h ~ 3Fh b : 0 ~ 7	
Operation	Skip next instruction if (f.b) = 0	
Status Affected	-	
OP-Code	01 010b bbff ffff	
Description	If bit 'b' in register 'f' is '1', then the next instruction is executed. If bit 'b' in register 'f' is '0', then the next instruction is discarded, and a NOP is executed instead, making this a 2nd cycle instruction.	
Cycle	1 or 2	
Example	LABEL1 BTFSC FLAG, 1 TRUE GOTO SUB1 FALSE ...	B : PC = LABEL1 A : if FLAG.1 = 0, PC = FALSE if FLAG.1 = 1, PC = TRUE

BTFSS	Test "b" bit of "f", skip if set(1)	
Syntax	BTFSS f [,b]	
Operands	f : 00h ~ 3Fh b : 0 ~ 7	
Operation	Skip next instruction if (f.b) = 1	
Status Affected	-	
OP-Code	01 011b bbff ffff	
Description	If bit 'b' in register 'f' is '0', then the next instruction is executed. If bit 'b' in register 'f' is '1', then the next instruction is discarded, and a NOP is executed instead, making this a 2nd cycle instruction.	
Cycle		
Example	LABEL1 BTFSS FLAG, 1 TRUE GOTO SUB1 FALSE ...	B : PC = LABEL1 A : if FLAG.1 = 0, PC = TRUE if FLAG.1 = 1, PC = FALSE

CALL	Call subroutine "k"	
Syntax	CALL k	
Operands	K : 00h ~ 7FFh	
Operation	Operation: TOS ← (PC)+ 1, PC.10~0 ← k	
Status Affected	-	
OP-Code	10 0kkk kkkk kkkk	
Description	Call Subroutine. First, return address (PC+1) is pushed onto the stack. The eleven-bit immediate address is loaded into PC bits <10:0>. CALL is a two-cycle instruction.	
Cycle	2	
Example	LABEL1 CALL SUB1	B : PC = LABEL1 A : PC = SUB1, TOS = LABEL1+1

CLRF	Clear f	
Syntax	CLRF f	
Operands	f : 00h ~ 7Fh	
Operation	(f) ← 00h, Z ← 1	
Status Affected	Z	
OP-Code	00 0001 1fff ffff	
Description	The contents of register ‘f’ are cleared and the Z bit is set.	
Cycle	1	
Example	CLRF FLAG_REG	B : FLAG_REG = 0x5A A : FLAG_REG = 0x00, Z = 1

CLRW	Clear W	
Syntax	CLRW	
Operands	-	
Operation	(W) ← 00h, Z ← 1	
Status Affected	Z	
OP-Code	00 0001 0100 0000	
Description	W register is cleared and Zero bit (Z) is set.	
Cycle	1	
Example	CLRW	B : W = 0x5A A : W = 0x00, Z = 1

CLRWDT	Clear Watchdog Timer	
Syntax	CLRWDT	
Operands	-	
Operation	WDTE ← 00h	
Status Affected	-	
OP-Code	00 0000 1000 1001	
Description	CLRWDT instruction enables and resets the Watchdog Timer.	
Cycle	1	
Example	CLRWDT	B : WDT counter = ? A : WDT counter = 0x00

COMF	Complement f	
Syntax	COMF f [,d]	
Operands	f : 00h ~ 7Fh, d : 0, 1	
Operation	(destination) ← ($\bar{f}$)	
Status Affected	Z	
OP-Code	00 1001 dfff ffff	
Description	The contents of register 'f' are complemented. If 'd' is 0, the result is stored in W. If 'd' is 1, the result is stored back in register 'f'.	
Cycle	1	
Example	COMF REG1,0	B : REG1 = 0x13 A : REG1 = 0x13, W = 0xEC

DECF	Decrement f	
Syntax	DECF f [,d]	
Operands	f : 00h ~ 7Fh, d : 0, 1	
Operation	(destination) ← (f) - 1	
Status Affected	Z	
OP-Code	00 0011 dfff ffff	
Description	Decrement register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.	
Cycle	1	
Example	DECF CNT, 1	B : CNT = 0x01, Z = 0 A : CNT = 0x00, Z = 1

DECFSZ	Decrement f, Skip if 0	
Syntax	DECFSZ f [,d]	
Operands	f : 00h ~ 7Fh, d : 0, 1	
Operation	(destination) ← (f) - 1, skip next instruction if result is 0	
Status Affected	-	
OP-Code	00 1011 dfff ffff	
Description	The contents of register 'f' are decremented. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'. If the result is 1, the next instruction is executed. If the result is 0, then a NOP is executed instead, making it a 2 cycle instruction.	
Cycle	1 or 2	
Example	LABEL1 DECFSZ CNT, 1 GOTO LOOP CONTINUE	B : PC = LABEL1 A : CNT = CNT – 1 if CNT=0, PC = CONTINUE if CNT≠0, PC = LABEL1+1

GOTO	Unconditional Branch	
Syntax	GOTO k	
Operands	k : 00h ~ 7FFh	
Operation	PC.10~0 ← k	
Status Affected	-	
OP-Code	11 0kkk kkkk kkkk	
Description	GOTO is an unconditional branch. The 11-bit immediate value is loaded into PC bits <10:0>. GOTO is a two-cycle instruction.	
Cycle	2	
Example	LABEL1 GOTO SUB1	B : PC = LABEL1 A : PC = SUB1

INCF — Increment f

Syntax	INCF f [,d]
Operands	f : 00h ~ 7Fh
Operation	(destination) ← (f) + 1
Status Affected	Z
OP-Code	00 1010 dfff ffff
Description	The contents of register 'f' are incremented. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'.
Cycle	1
Example	INCF CNT, 1 B : CNT = 0xFF, Z = 0 A : CNT = 0x00, Z = 1

INCFSZ — Increment f, Skip if 0

Syntax	INCFSZ f [,d]
Operands	f : 00h ~ 7Fh, d : 0, 1
Operation	(destination) ← (f) + 1, skip next instruction if result is 0
Status Affected	-
OP-Code	00 1111 dfff ffff
Description	The contents of register 'f' are incremented. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'. If the result is 1, the next instruction is executed. If the result is 0, a NOP is executed instead, making it a 2 cycle instruction.
Cycle	1 or 2
Example	LABEL1 INCFSZ CNT, 1 B : PC = LABEL1 GOTO LOOP A : CNT = CNT + 1 CONTINUE if CNT=0, PC = CONTINUE if CNT≠0, PC = LABEL1+1

IORLW — Inclusive OR Literal with W

Syntax	IORLW k
Operands	k : 00h ~ FFh
Operation	(W) ← (W) OR k
Status Affected	Z
OP-Code	01 1010 kkkk kkkk
Description	The contents of the W register is OR'ed with the eight-bit literal 'k'. The result is placed in the W register.
Cycle	1
Example	IORLW 0x35 B : W = 0x9A A : W = 0xBF, Z = 0

IORWF — Inclusive OR W with f

Syntax	IORWF f [,d]
Operands	f : 00h ~ 7Fh, d : 0, 1
Operation	(destination) ← (W) OR k
Status Affected	Z
OP-Code	00 0100 dfff ffff
Description	Inclusive OR the W register with register 'f'. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'.
Cycle	1
Example	IORWF RESULT, 0 B : RESULT = 0x13, W = 0x91 A : RESULT = 0x13, W = 0x93, Z = 0

MOVFW	Move f to W	
Syntax	MOVFW f	
Operands	f : 00h ~ 7Fh	
Operation	(W) ← (f)	
Status Affected	-	
OP-Code	00 1000 0fff ffff	
Description	The contents of register f are moved to W register.	
Cycle	1	
Example	MOVF FSR, 0	B : W = ? A : W ← f, if W = 0 Z = 1

MOVLW	Move Literal to W	
Syntax	MOVLW k	
Operands	k : 00h ~ FFh	
Operation	(W) ← k	
Status Affected	-	
OP-Code	01 1001 kkkk kkkk	
Description	The eight-bit literal ‘k’ is loaded into W register. The don’t cares will assemble as 0’s.	
Cycle	1	
Example	MOVLW 0x5A	B : W = ? A : W = 0x5A

MOVWF	Move W to f	
Syntax	MOVWF f	
Operands	f : 00h ~ 7Fh	
Operation	(f) ← (W)	
Status Affected	-	
OP-Code	00 0000 1fff ffff	
Description	Move data from W register to register ‘f’.	
Cycle	1	
Example	MOVWF REG1	B : REG1 = 0xFF, W = 0x4F A : REG1 = 0x4F, W = 0x4F

NOP	No Operation	
Syntax	NOP	
Operands	-	
Operation	No Operation	
Status Affected	Z	
OP-Code	00 0000 0000 0000	
Description	No Operation	
Cycle	1	
Example	NOP	-

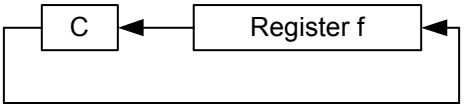
RETI	Return from Interrupt	
Syntax	RETI	
Operands	-	
Operation	PC ← TOS, GIE ← 1	
Status Affected	-	
OP-Code	00 0000 0110 0000	
Description	Return from Interrupt. Stack is POPed and Top-of-Stack (TOS) is loaded in to the PC. Interrupts are enabled. This is a two-cycle instruction.	
Cycle	2	
Example	RETFIE	A : PC = TOS, GIE = 1

RETLW	Return with Literal in W	
Syntax	RETLW k	
Operands	k : 00h ~ FFh	
Operation	PC ← TOS, (W) ← k	
Status Affected	-	
OP-Code	01 1000 kkkk kkkk	
Description	The W register is loaded with the eightbit literal 'k'. The program counter is loaded from the top of the stack (the return address). This is a two-cycle instruction.	
Cycle	2	
Example	CALL TABLE	B : W = 0x07
	:	A : W = value of k8
	TABLE ADDWF PCL,1	
	RETLW k1	
	RETLW k2	
	:	
	RETLW kn	

RET	Return from Subroutine	
Syntax	RET	
Operands	-	
Operation	PC ← TOS	
Status Affected	-	
OP-Code	00 0000 0100 0000	
Description	Return from subroutine. The stack is POPed and the top of the stack (TOS) is loaded into the program counter. This is a two-cycle instruction.	
Cycle	2	
Example	RETURN	A : PC = TOS

RLF Rotate Left f through Carry

Syntax: RLF f [,d]
Operands: f : 00h ~ 7Fh, d : 0, 1
Operation:

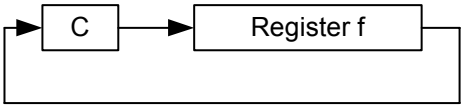


Status Affected: C
OP-Code: 00 1101 dfff ffff
Description: The contents of register ‘f’ are rotated one bit to the left through the Carry Flag. If ‘d’ is 0, the result is placed in the W register. If ‘d’ is 1, the result is stored back in register ‘f’.
Cycle: 1
Example: RLF REG1,0

B : REG1 = 1110 0110, C = 0
A : REG1 = 1110 0110
W = 1100 1100, C = 1

RRF Rotate Right f through Carry

Syntax: RRF f [,d]
Operands: f : 00h ~ 7Fh, d : 0, 1
Operation:



Status Affected: C
OP-Code: 00 1100 dfff ffff
Description: The contents of register ‘f’ are rotated one bit to the right through the Carry Flag. If ‘d’ is 0, the result is placed in the W register. If ‘d’ is 1, the result is placed back in register ‘f’.
Cycle: 1
Example: RRF REG1,0

B : REG1 = 1110 0110, C = 0
A : REG1 = 1110 0110
W = 0111 0011, C = 0

SLEEP Go into standby mode, Clock oscillation stops

Syntax: SLEEP
Operands: -
Operation: -
Status Affected: -
OP-Code: 00 0000 1000 1010
Description: Go into SLEEP mode with the oscillator stopped.
Cycle: 1
Example: SLEEP -

SUBWF — Subtract W from f

Syntax	SUBWF f [,d]
Operands	f : 00h ~ 7Fh, d : 0, 1
Operation	(destination) ← (f) – (W)
Status Affected	C, DC, Z
OP-Code	00 0010 dfff ffff
Description	Subtract (2's complement method) W register from register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.
Cycle	1
Example	SUBWF REG1,1 — B : REG1 = 3, W = 2, C = ?, Z = ? A : REG1 = 1, W = 2, C = 1, Z = 0
	SUBWF REG1,1 — B : REG1 = 2, W = 2, C = ?, Z = ? A : REG1 = 0, W = 2, C = 1, Z = 1
	SUBWF REG1,1 — B : REG1 = 1, W = 2, C = ?, Z = ? A : REG1 = FFh, W = 2, C = 0, Z = 0

SWAPF — Swap Nibbles in f

Syntax	SWAPF f [,d]
Operands	f : 00h ~ 7Fh, d : 0, 1
Operation	(destination,7~4) ← (f.3~0), (destination.3~0) ← (f.7~4)
Status Affected	-
OP-Code	00 1110 dfff ffff
Description	The upper and lower nibbles of register 'f' are exchanged. If 'd' is 0, the result is placed in W register. If 'd' is 1, the result is placed in register 'f'.
Cycle	1
Example	SWAPF REG, 0 — B : REG1 = 0xA5 A : REG1 = 0xA5, W = 0x5A

TESTZ — Test if 'f' is zero

Syntax	TESTZ f
Operands	f : 00h ~ 7Fh
Operation	Set Z flag if (f) is 0
Status Affected	Z
OP-Code	00 1000 1fff ffff
Description	If the content of register 'f' is 0, Zero flag is set to 1.
Cycle	1
Example	TESTZ REG1 — B : REG1 = 0, Z = ? A : REG1 = 0, Z = 1

XORLW — Exclusive OR Literal with W

Syntax	XORLW k
Operands	k : 00h ~ FFh
Operation	(W) ← (W) XOR k
Status Affected	Z
OP-Code	01 1111 kkkk kkkk
Description	The contents of the W register are XOR'ed with the eight-bit literal 'k'. The result is placed in the W register.
Cycle	1
Example	XORLW 0xAF — B : W = 0xB5 A : W = 0x1A

XORWF	Exclusive OR W with f	
Syntax	XORWF f [,d]	
Operands	f : 00h ~ 7Fh, d : 0, 1	
Operation	(destination) ← (W) XOR (f)	
Status Affected	Z	
OP-Code	00 0110 dfff ffff	
Description	Exclusive OR the contents of the W register with register ‘f’. If ‘d’ is 0, the result is stored in the W register. If ‘d’ is 1, the result is stored back in register ‘f’.	
Cycle	1	
Example	XORWF REG 1	B : REG = 0xAF, W = 0xB5 A : REG = 0x1A, W = 0xB5

2. Control Register

Description	Mnemonic	Dec	Hex	R/W
System Config Reg Low	SYSL	-	2000	−
Indirect File Reg	INDF	0	00H	−
Timer 0 Counter Reg	T0CNT	1	01H	R
Program Counter Low	PCL	2	02H	R/W
System Flags Reg	STATUS	3	03H	R/W
File Select Reg	FSR	4	04H	R/W
Port A Data Reg	PAD	5	05H	R/W
Port B Data Reg	PBD	6	06H	R/W
Port C Data Reg	PCD	7	07H	R/W
Clock control Reg	CLKCON	8	08H	R/W
WatchDog Timer Control Reg	WDTE	9	09H	−
Stop mode Control Reg	PWRDN	10	0AH	−
Interrupt Control Reg	INTCON	11	0BH	R/W
Interrupt Pending Reg	INTPND	12	0CH	R/W
External Interrupt Signal Control Reg	PINTD	13	0DH	R/W
Timer 0 Control Reg	T0CON	14	0EH	R/W
Timer 0 Data Reg	T0DATA	15	0FH	R/W
PWM 0 Control Reg	PWM0CON	16	10H	R/W
PWM 0 Data Reg	PWM0DAT	17	11H	R/W
PWM 1 Control Reg	PWM1CON	18	12H	R/W
PWM 1 Data Reg	PWM1DAT	19	13H	R/W
Buzzer Control Reg	BZCON	20	14H	R/W
Port A Control Reg Low	PACONL	21	15H	R/W
Port A Control Reg High	PACONH	22	16H	R/W
Port B Control Reg	PBCON	23	17H	R/W
Port C Control Reg Low	PCCONL	24	18H	R/W
Port C Control Reg High	PCCONH	25	19H	R/W
ADC Control Reg	ADCCON	26	1AH	R/W
ADC DATA Reg Low	ADCDATL	27	1BH	R
ADC DATA Reg High	ADCDATH	28	1CH	R
Location 1DH is factory use only				
General Purpose Register 0	GPR0	30	1EH	R/W
General Purpose Register 1	GPR1	31	1FH	R/W

ADCCON — A/D Converter Control Register **Address: 1AH**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	0	0	0	0	0	0	0	0	
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	

Bit	Description				
7-4	**Input Pin Selection Bits**				
	0	0	0	0	ADC0 (PA.0)
	0	0	0	1	ADC1 (PA.1)
	0	0	1	0	ADC2 (PA.2)
	0	0	1	1	ADC3 (PA.3)
	0	1	0	0	ADC4 (PA.4)
	0	1	0	1	ADC5 (PA.5)
	0	1	1	0	ADC6 (PA.6)
	0	1	1	1	ADC7 (PA.7)
	1	0	0	0	ADC8 (PC.6)
	1	0	0	1	ADC9 (PC.5)
	1	1	1	1	Connected with V_{DD} internally
	Others				Connected with GND internally
3	**End-of-Conversion Status Bit**				
	0	A/D conversion is in progress			
	1	A/D conversion complete			
2-1	**Clock Source Selection Bit** (NOTE 1)				
	0	0	f_{osc} /16		
	0	1	f_{osc} /8		
	1	0	f_{osc} /4		
	1	1	f_{osc} /1		
0	**Conversion Start Bit**				
	0	No meaning			
	1	A/D conversion start			

NOTE :

1. Maximum ADC Input Clock is 4MHz.

ADCDATL — ADC Data Register Low Byte

Address: 1BH

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	–	–	–	–	–	–	–	–	
R/W	–	–	–	–	–	–	R	R	

Bit	Description	
1-0	**ADC Data Low Byte**	
	XX	ADC Data Value Lower 2Bit

ADCDATH — ADC Data Register High Byte

Address: 1CH

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	–	–	–	–	–	–	–	–	
R/W	R	R	R	R	R	R	R	R	

Bit	Description	
7-0	**ADC Data High Byte**	
	XXXXXXXX	ADC Data Value Higher 8Bit

BZCON — Buzzer Out Control Register

Address: 14H

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	1	1	1	1	1	1	1	1	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Bit	Description		
7-6	**Input Clock Selection**		
	0	0	f_{osc} / 8
	0	1	f_{osc} / 16
	1	0	f_{osc} / 32
	1	1	f_{osc} / 64
5-0	**Buzzer Period Data**		
	XXXXXX	Period Data	

CLKCON — Clock Control Register

Address: 08H

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	0	–	–	–	–	–	0	0	
R/W	R/W	–	–	–	–	–	R/W	R/W	

Bit	Description		
7	**System Divider Clear bit**		
	0	No effect	
	1	Clear Divider (Auto Clear)	
6-2	Not Used		
1-0	**Divided by Selection Bits for CPU Clock frequency**		
	0	0	Divide by f_{OSC} /16
	0	1	Divide by f_{OSC} /8
	1	0	Divide by f_{OSC} /4
	1	1	Divide by f_{OSC} /2

FSR — File Select Register

Address: 04H

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	–	–	–	–	–	–	–	–	
R/W	–	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Bit	Description	
7	Not Used	
6-0	**File Select Register**	
	000 0000	Not Used.
	1 ~ 7Fh	Indirect Addressing Location

GPR0/1 — General Purpose Register

Address: 1EH/1FH

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	0	0	0	0	0	0	0	0	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Bit	Description
7-0	**General Purpose Register**
	GPR0, GPR1 are mirrored all bank. It is useful to pass arguments to SUB routine or backup Working register (W) and STATUS register in ISR or SUB routine.

INTCON — Interrupt Control Register **Address: 0BH**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	–	–	–	0	0	0	0	0	
R/W	–	–	–	R/W	R/W	R/W	R/W	R/W	

Bit	Description	
7-5	Not Used	
4	**PWM 1 Overflow Interrupt Enable Bit**	
	0	PWM 1 Interrupt Disable
	1	PWM 1 Interrupt Enable
3	**PWM 0 Overflow Interrupt Enable Bit**	
	0	PWM 0 Interrupt Disable
	1	PWM 0 Interrupt Enable
2	**Timer 0 Interrupt Enable Bit**	
	0	Timer 0 Interrupt Disable
	1	Timer 0 Interrupt Enable
1	**Port A.1 EXTINT1 Interrupt Enable Bit**	
	0	EXTINT1 Interrupt Disable
	1	EXTINT1 Interrupt Enable
0	**Port A.0 EXTINT0 Interrupt Enable Bit**	
	0	EXTINT0 Interrupt Disable
	1	EXTINT0 Interrupt Enable

INTPND — Interrupt Pending Register **Address: 0CH**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	–	–	–	0	0	0	0	0	
R/W	–	–	–	R/W	R/W	R/W	R/W	R/W	

Bit	Description	
7-5	Not Used	
4	**PWM 1 Overflow Interrupt Pending Bit**	
	0	No interrupt pending (read) / Pending bit clear (write)
	1	Interrupt is pending (read) / No effect (write)
3	**PWM 0 Overflow Interrupt Pending Bit**	
	0	No interrupt pending (read) / Pending bit clear (write)
	1	Interrupt is pending (read) / No effect (write)
2	**Timer 0 Interrupt Pending Bit**	
	0	No interrupt pending (read) / Pending bit clear (write)
	1	Interrupt is pending (read) / No effect (write)
1	**Port A.1 EXTINT1 Interrupt Pending Bit**	
	0	No interrupt pending (read) / Pending bit clear (write)
	1	Interrupt is pending (read) / No effect (write)
0	**Port A.0 EXTINT0 Interrupt Pending Bit**	
	0	No interrupt pending (read) / Pending bit clear (write)
	1	Interrupt is pending (read) / No effect (write)

PACONL — Port A Control Register (Low Byte) **Address: 15H**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	0	0	0	0	0	0	0	0	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Bit	Description		
7-6	**Port A.3 Configuration Bits**		
	0	0	Schmitt trigger input (pull-up enable)
	0	1	Schmitt trigger input
	1	0	Push-pull output
	1	1	ADC3 Input (Schmitt trigger input off)
5-4	**Port A.2 Configuration Bits**		
	0	0	Schmitt trigger input (pull-up enable)
	0	1	Schmitt trigger input
	1	0	Push-pull output
	1	1	ADC2 Input (Schmitt trigger input off)
3-2	**Port A.1 Configuration Bits**		
	0	0	Schmitt trigger input (pull-up enable) / External Interrupt 1 Input
	0	1	Schmitt trigger input / External Interrupt 1 Input
	1	0	Push-pull output
	1	1	ADC1 Input (Schmitt trigger input off)
1-0	**Port A.0 Configuration Bits**		
	0	0	Schmitt trigger input (pull-up enable) / External Interrupt 0 Input
	0	1	Schmitt trigger input / External Interrupt 0 Input
	1	0	Push-pull output
	1	1	ADC0 Input (Schmitt trigger input off)

PACONH — Port A Control Register (High Byte) **Address: 16H**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	0	0	0	0	0	0	0	0	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Bit	Description		
7-6	**Port A.7 Configuration Bits**		
	0	0	Schmitt trigger input (pull-up enable)
	0	1	PWM 1 output
	1	0	Push-pull output
	1	1	ADC7 Input (Schmitt trigger input off)
5-4	**Port A.6 Configuration Bits**		
	0	0	Schmitt trigger input (pull-up enable)
	0	1	PWM 0 output
	1	0	Push-pull output
	1	1	ADC6 Input (Schmitt trigger input off)
3-2	**Port A.5 Configuration Bits**		
	0	0	Schmitt trigger input (pull-up enable)
	0	1	Schmitt trigger input
	1	0	Push-pull output
	1	1	ADC5 Input (Schmitt trigger input off)
1-0	**Port A.4 Configuration Bits**		
	0	0	Schmitt trigger input (pull-up enable)
	0	1	Schmitt trigger input
	1	0	Push-pull output
	1	1	ADC4 Input (Schmitt trigger input off)

PBCON — Port B Control Register **Address: 17H**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	–	–	0	0	1	0	0	1	
R/W	–	–	–	–	–	–	–	–	

Bit	Description			
7-6	Not Used			
5-3	**Port B.1 Configuration Bits**			
	0	0	0	Schmitt trigger input (pull-up enable)
	0	0	1	Schmitt trigger input
	0	1	0	Push-pull output
	0	1	1	Schmitt trigger input (pull-down)
	1	0	0	Open-drain Output
	Other Value			Not Used
2-0	**Port B.0 Configuration Bits**			
	0	0	0	Schmitt trigger input (pull-up enable)
	0	0	1	Schmitt trigger input
	0	1	0	Push-pull output
	0	1	1	Schmitt trigger input (pull-down)
	1	0	0	Open-drain Output
	Other Value			Not Used

PCCONL — Port C Control Register (Low Byte) **Address: 18H**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	0	0	0	0	0	0	0	0	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Bit	Description		
7-6	**Port C.3 Configuration Bits**		
	0	0	Schmitt trigger input (pull-up)
	0	1	Not Used
	1	0	Push-pull output
	1	1	Open-drain output
5-4	**Port C.2 Configuration Bits**		
	0	0	Schmitt trigger input (pull-up)
	0	1	Not Used
	1	0	Push-pull output
	1	1	Open-drain output
3-2	**Port C.1 Configuration Bits**		
	0	0	Schmitt trigger input (pull-up)
	0	1	Buzzer Out
	1	0	Push-pull output
	1	1	Open-drain output
1-0	**Port C.0 Configuration Bits**		
	0	0	Schmitt trigger input(pull-up)
	0	1	Not Used
	1	0	Push-pull output
	1	1	T0 match output

PCCONH — Port C Control Register (High Byte) **Address: 19H**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	0	0	0	0	0	0	0	0	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Bit	Description			
7-5	**Port C.6 Configuration Bits**			
	0	0	0	Schmitt trigger input (pull-up)
	0	0	1	Schmitt trigger input
	0	1	X	ADC8 Input
	1	0	0	Push-pull output
	1	0	1	Open-drain output (pull-up)
	1	1	0	Open-drain output
	1	1	1	Clock Output
4-2	**Port C.5 Configuration Bits**			
	0	0	0	Schmitt trigger input (pull-up)
	0	0	1	Schmitt trigger input
	0	1	X	ADC9 Input
	1	0	0	Push-pull output
	1	0	1	Open-drain output (pull-up)
	1	1	0	Open-drain output
	1	1	1	Not Used
1-0	**Port C.4 Configuration Bits**			
	0	0	Schmitt trigger input (pull-up)	
	0	1	Schmitt trigger input	
	1	0	Push-pull output	
	1	1	Open-drain output	

PAD — Port A Data Register **Address: 05H**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	0	0	0	0	0	0	0	0	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Bit	Description
7-0	Port A.7-0 Data Bits

PBD — Port B Data Register **Address: 06H**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	–	–	–	–	–	0	0	0	
R/W	–	–	–	–	–	–	–	–	

Bit	Description
7-3	Not Used
2-0	Port B.2-0 Data Bits

PCD — Port C Data Register **Address: 07H**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	–	0	0	0	0	0	0	0	
R/W	–	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Bit	Description
7	Not Used
6-0	Port C.6-0 Data Bits

PCL — Program Counter Low Byte **Address: 02H**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	0	0	0	0	0	0	0	0	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Bit	Description
7-0	**Program Counter Low Byte**
	This register represents Lower 8-Bit of PC+1. The PC can be changed writing any value (00h~FFh) into this register. It is similar to GOTO instruction. But the branch instruction by PCL can access only higher address than PC.

PINTD — External Interrupt Signal Control Register **Address: 0DH**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	–	–	–	–	0	0	0	0	
R/W	–	–	–	–	R/W	R/W	R/W	R/W	

Bit	Description		
7-4	Not Used		
3-2	**External Interrupt 1 Input Signal Selection Bits**		
	0	0	Falling Edge
	0	1	Rising Edge
	1	X	Both Edge
1-0	**External Interrupt 0 Input Signal Selection Bits**		
	0	0	Falling Edge
	0	1	Rising Edge
	1	X	Both Edge

PWM0CON — PWM0 Control Register **Address: 10H**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	–	–	0	0	–	0	0	0	
R/W	–	–	R/W	R/W	–	R/W	R/W	R/W	

Bit	Description		
7-6	Not Used		
5-4	**PWM0 Input Clock Selection Bit**		
	0	0	f_{osc} / 64
	0	1	f_{osc} / 8
	1	0	f_{osc} / 2
	1	1	f_{osc} / 1
3	Not Used		
2	**PWM0 DATA Reload Interval Selection Bit**		
	0	Reload from 8-bit up counter overflow	
	1	Reload from 6-bit up counter overflow	
1	**PWM0 Counter Clear Bit (Auto Cleared)**		
	0	No effect	
	1	Clear the PWM counter (when write)	
0	**PWM0 Enable Bit**		
	0	Stop counter	
	1	Start (Resume countering)	

PWM0DAT — PWM0 Data Register **Address: 11H**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	0	0	0	0	0	0	0	0	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Bit	Description	
7-2	**PWM Period Data**	
	XXXXXX	Period Data
1-0	**Extension Cycle Selection Bit**	
	0	0 –
	0	1 2
	1	0 1, 3
	1	1 1, 2, 3

PWM1CON — PWM1 Control Register **Address: 12H**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	0	0	0	0	–	0	0	0	
R/W	R/W	R/W	R/W	R/W	–	R/W	R/W	R/W	

Bit	Description	
7-6	**PWM1 Extension Cycle Selection Bit**	
	0	0 –
	0	1 2
	1	0 1, 3
	1	1 1, 2, 3
5-4	**PWM1 Input Clock Selection**	
	0	0 f_{osc} / 64
	0	1 f_{osc} / 8
	1	0 f_{osc} / 2
	1	1 f_{osc} / 1
3	Not Used	
2	**PWM1 DATA Reload Interval Selection Bit**	
	0	Reload from 10-bit up counter overflow
	1	Reload from 8-bit up counter overflow
1	**PWM1 Counter Clear Bit (Auto Cleared)**	
	0	No effect
	1	Clear the PWM counter (when write)
0	**PWM1 Enable Bit**	
	0	Stop counter
	1	Start (Resume counting)

PWM1DAT — PWM1 Data Register **Address: 13H**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	0	0	0	0	0	0	0	0	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Bit	Description	
7-0	**PWM1 Period Data Low Byte**	
	XXXXXXXX	Period Data

PWRDN — Power Down Control Register **Address: 0AH**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	–	–	–	–	–	–	–	–	
R/W	–	–	–	–	–	–	–	–	

Bit	Description
7-0	**Power Down Control Register**
	This register is not physical register. The device can enter STOP mode by writing any value into this register. The SLEEP instruction is equivalent to “MOVWF PWRDN”.

STATUS — System Flags Register **Address: 03H**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	–	–	–	0	–	0	0	0	
R/W	–	–	–	R/W	–	R/W	R/W	R/W	

Bit	Description	
7-5	Not Used (Must be set to 0)	
4	**SRAM Bank Selection Bit**	
	0	Page 0
	1	Page 1
3	Not Used (Must be set to 0)	
2	**Zero Flag(Z)**	
	0	The result of a logic operation is not zero
	1	The result of a logic operation is zero
1	**Decimal Carry Flag or Decimal/Borrow Flag (DC)**	
	ADD instruction	SUB instruction
	1: a carry from the low nibble bits of the result occurred 0: no carry	1: no borrow 0: a borrow from the low nibble bits of the result occurred
0	**Carry Flag(C) or Borrow Flag**	
	ADD instruction	SUB instruction
	1: a carry occurred from the MSB 0: no carry	1: no borrow 0: a borrow occurred from the MSB

SYSL — System Config Register **Address: 2000H**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	1	1	1	1	1	1	1	1	
R/W	–	–	–	–	–	–	–	–	

Bit	Description		
13	**Code protection selection bit**		
	1	No protect	
	0	Code protection	
7	**Not Used (Must Set be '1')**		
6-5	**Clock Source Selection Bit**		
	CSS1	CSS0	CSS1 ~ 0 Clock Source Selection Bit
	0	0	External crystal / ceramic oscillator
	0	1	External RC
	1	0	Internal RC (0.47 MHz in V_{DD} = 5 V)
	1	1	Internal RC (2.8 MHz in V_{DD} = 5 V)
4-0	**LVS: LVR Level Selection Bit**		
	11001		2.0V
	11010		2.3V
	10001		3.0V
	01111		3.9V

T0CON — TIMER 0 Control Register **Address: 0EH**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	–	–	0	0	–	–	–	0	
R/W	–	–	R/W	R/W	–	–	–	R/W	

Bit	Description		
7-6	Not Used		
5-4	**Timer 0 Input Clock Selection Bits**		
	0	0	f_{osc} /4096
	0	1	f_{osc} /256
	1	0	f_{osc} /8
	1	1	f_{osc} /1
3-1	Not Used		
0	**Timer 0 Counter Clear Bit**		
	0	No effect	
	1	Clear the timer 0 counter (when write)	

T0CNT — TIMER 0 Counter Register **Address: 01H**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	0	0	0	0	0	0	0	0	
R/W	R	R	R	R	R	R	R	R	

Bit	Description
7-0	Timer 0 Counter Value

T0DATA — TIMER 0 Data Register **Address: 0FH**

Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	1	1	1	1	1	1	1	1	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Bit	Description
7-0	Period Data

WDTE — WatchDog Timer Control Register **Address: 09H**

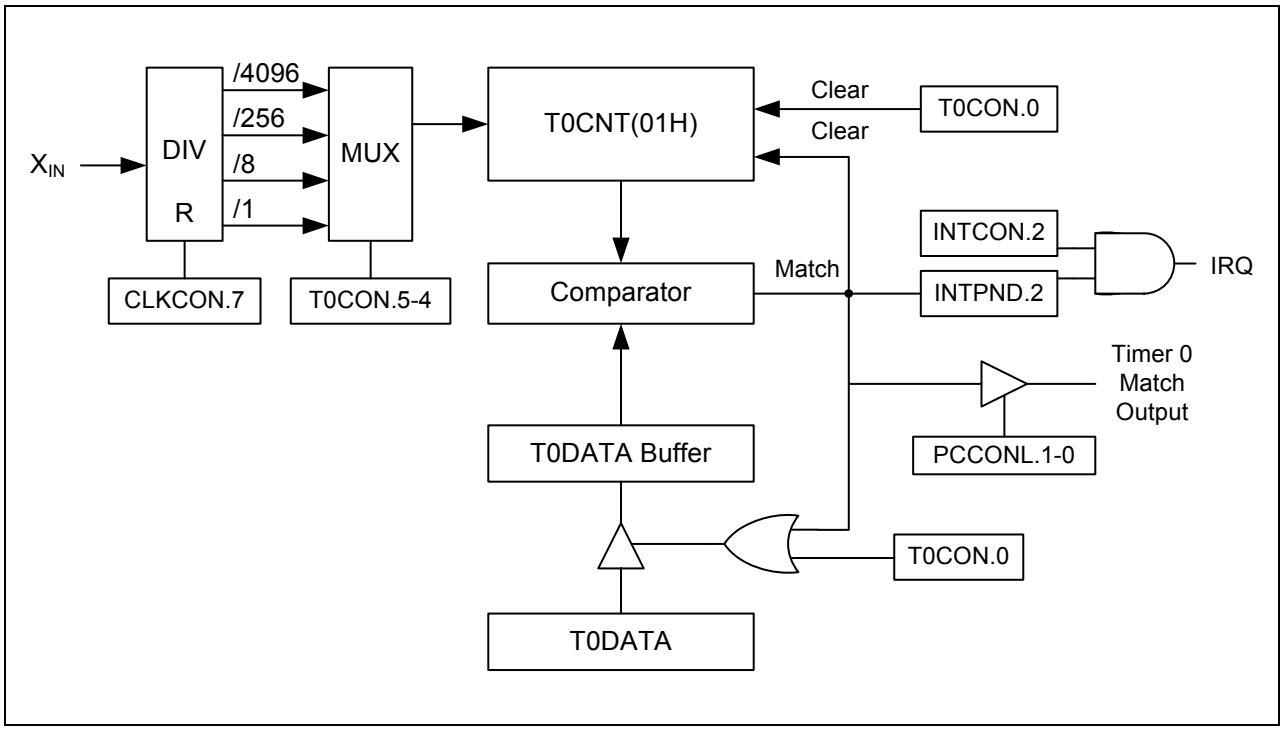
Bit	7	6	5	4	3	2	1	0	Related Register
Reset Value	–	–	–	–	–	–	–	–	
R/W	–	–	–	–	–	–	–	–	

Bit	Description
7-0	**WatchDog Timer Control Register**
	This register is not physical register. The WatchDog timer can be enabled and refreshed by CLRWDT or writing any value into this register. The CLRWDT instruction is equivalent to “MOVWF WDTE”.

3. 8-Bit Timer

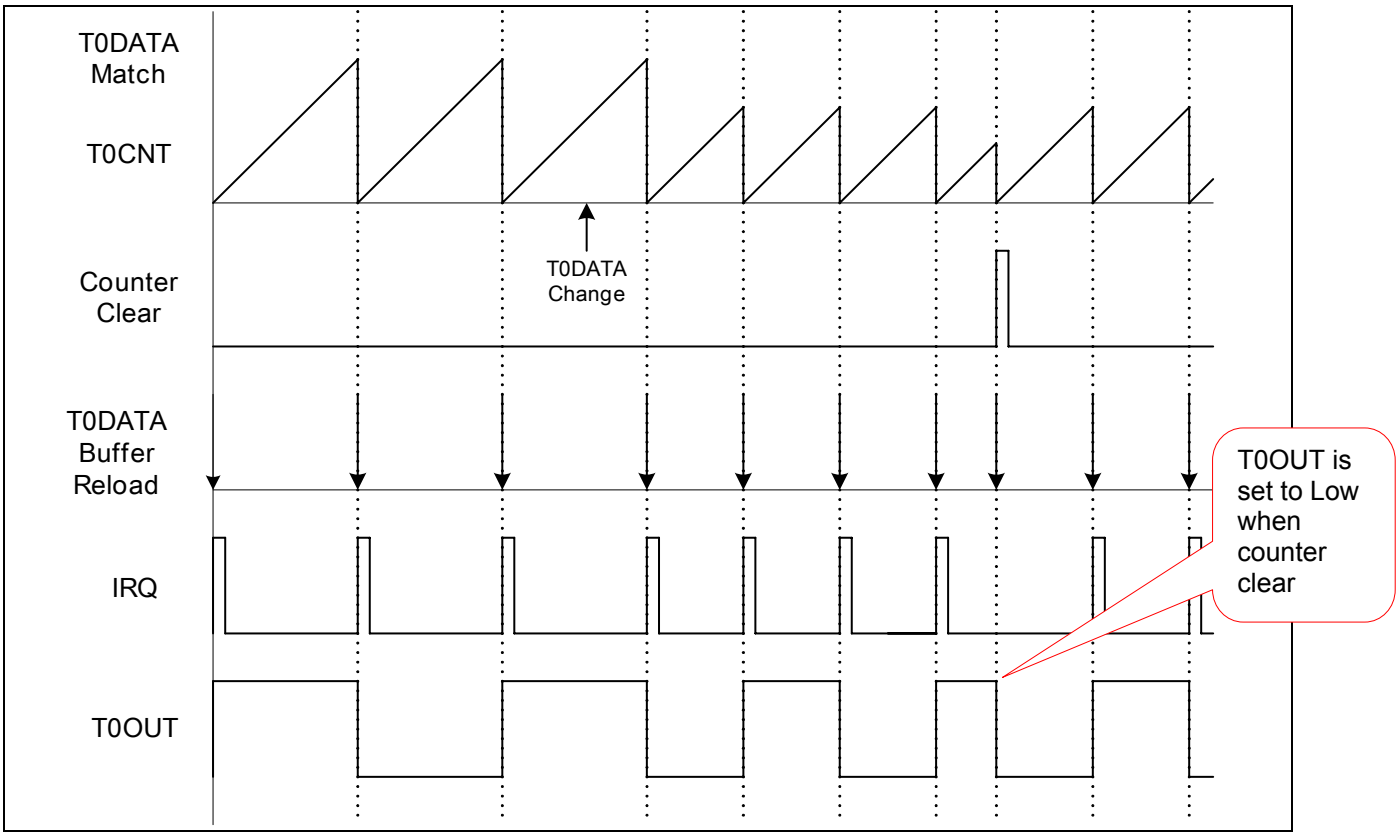
TIMER0 has the following functional components:

- Clock frequency selector
- 8-bit counter (T0CNT), 8-bit comparator, 8-bit data register (T0DATA), and T0DATA buffer.
- TIMER0 control register (T0CON)

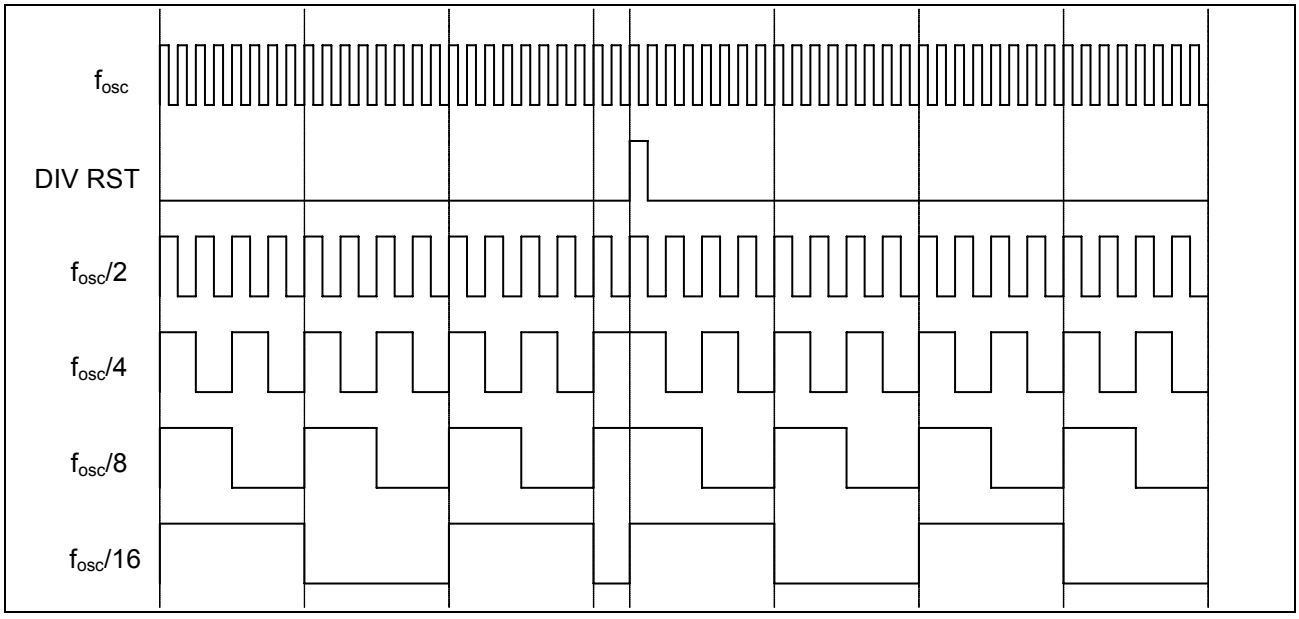


< Figure 3-1 Block Diagram >

T0CON is used to select input clock frequency, to clear the timer 0 counter. Interrupt enable and pending bit for Timer0 interrupt is controlled by INTCON and INTPND. In interval timer mode, a match signal is generated when the counter value is identical to the value T0DATA. The match signal generates a TIMER0 match interrupt, clears the counter and counting resumes. If the TIMER0 interrupt is disabled (INTCON.2 = 0), the match signal do not generates match interrupt request. The clock divider is not the constituent of Timer 0, then the divided clock is asynchronous with Timer interrupt enable signal. Therefore, there is discrepancy in first match interval. To minimize this discrepancy, divider reset can be used (CLKCON).



< Figure 3-2 Timimg diagram >



< Figure 3-3 Divider reset >

Example 3-1> Timer 0 Sample Code (f_{OSC} = 8.192 MHz, Interval = 1ms, T0OUT = 500 Hz)

```
      org     01h
int_vector:
      BTFSS   INTCON, 2   ; Timer 0 Interrupt Check
      GOTO    NEXT_INT    ; Jump to Other Interrupt Rotine
      .                   ; Timer 0 Interrupt Rotine
      .
      .
NEXT_INT:
      .
      .
      RETI
```

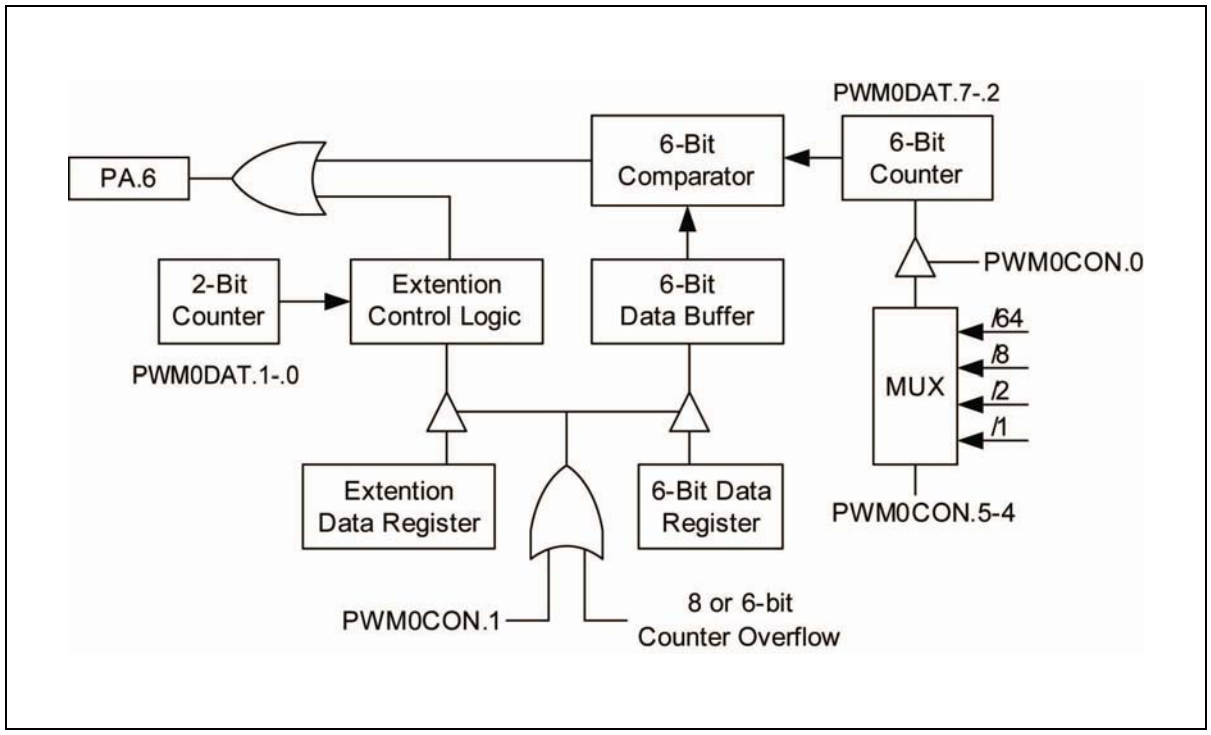
```
      MOVLW   1Fh         ; Set T0DATA 1FH
      MOVWF   T0DATA

      MOVLW   00010000b   ; fosc/256
      MOVWF   T0CON       ; Set T0CON Control Register

      BSF     T0CON,  0   ; Timer0 Counter Clear

      BSF     PCCONL, 0   ; Select PX.0 match output.
      BSF     PCCONL, 1   ; PCCONL Bit [1-0]:[11] is match output

      BSF     INTCON, 2   ; Timer0 Interrupt Enable
      .
      .
```

4. 8-Bit PWM

PWM0 has the following functional components:

- Clock frequency selector
- 8-bit up-counter, 6-bit comparator, 6-bit data register and 6-bit data buffer.
- 2-bit extension control logic, 2-bit extension register and extension data buffer.
- Control register (PWM0CON)

To determine the PWM0 operating frequency, the upper 6-bits of counter is compared to the PWM0 data register (PWM0DAT.7–.2). In order to achieve higher resolutions, the lower 2-bits of the counter can be used to modulate the "extended" cycle.



< Figure 4-1 Block Diagram >

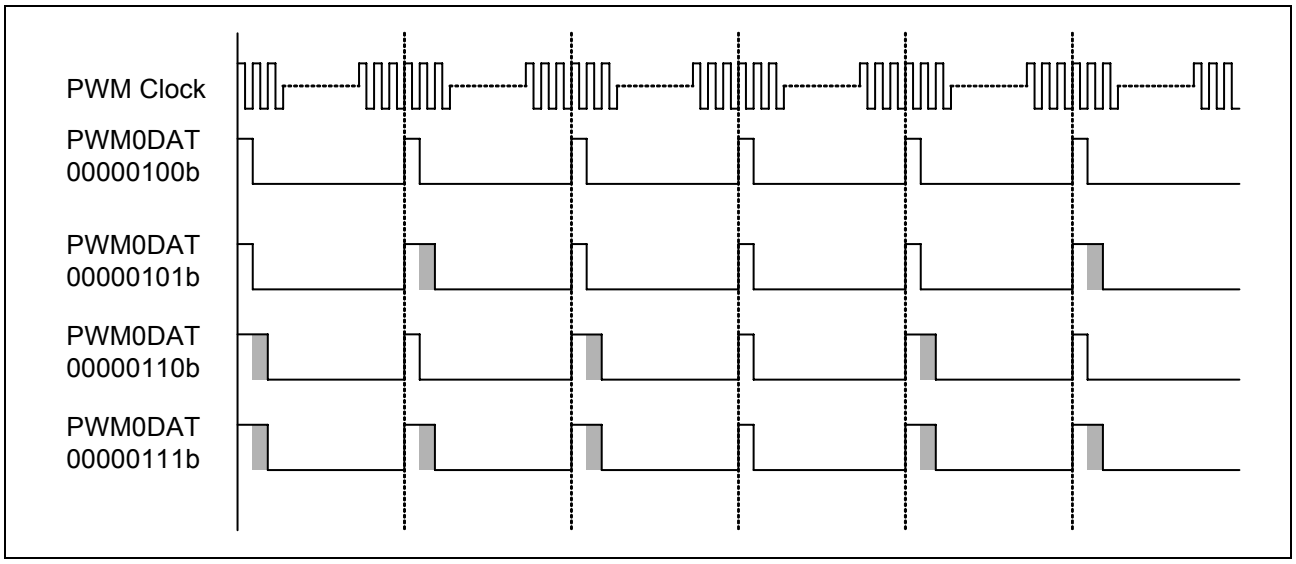
The PWM output signal toggles to Low level whenever the lower 6-bit of counter matches the reference data register (PWM0DAT.7–.2). If the value in the PWM0DAT.7–.2 register is not zero, an overflow of the lower 6-bits of counter causes the PWM output to toggle to High level. In this way, the reference value written to the reference data register determines the module's base duty cycle.

The value in the upper 2-bits of counter is compared with the extension settings in the 2-bit extension data register (PWM0DAT.1–.0). This lower 2-bits of counter value is used to "extend" the duty cycle of the PWM output. The "extension" value is one extra clock period at specific cycles (see Table 4-1).

PWM0DAT.1-0	Extended Cycle
00	None
01	2
10	1, 3
11	1, 2, 3

< Table 4-1 PWM output extended cycle >

For example, if the value in the extension data register is '01B', the 2nd cycle will be one pulse longer than the other 3 cycles. (see Figure 4-2).



< Figure 4-2 Extended Output >

Example 4-1> PWM0 Sample Code (f_{OSC} = 8 MHz, 1 Cycle = 500μs, Extend 2nd Cycle)

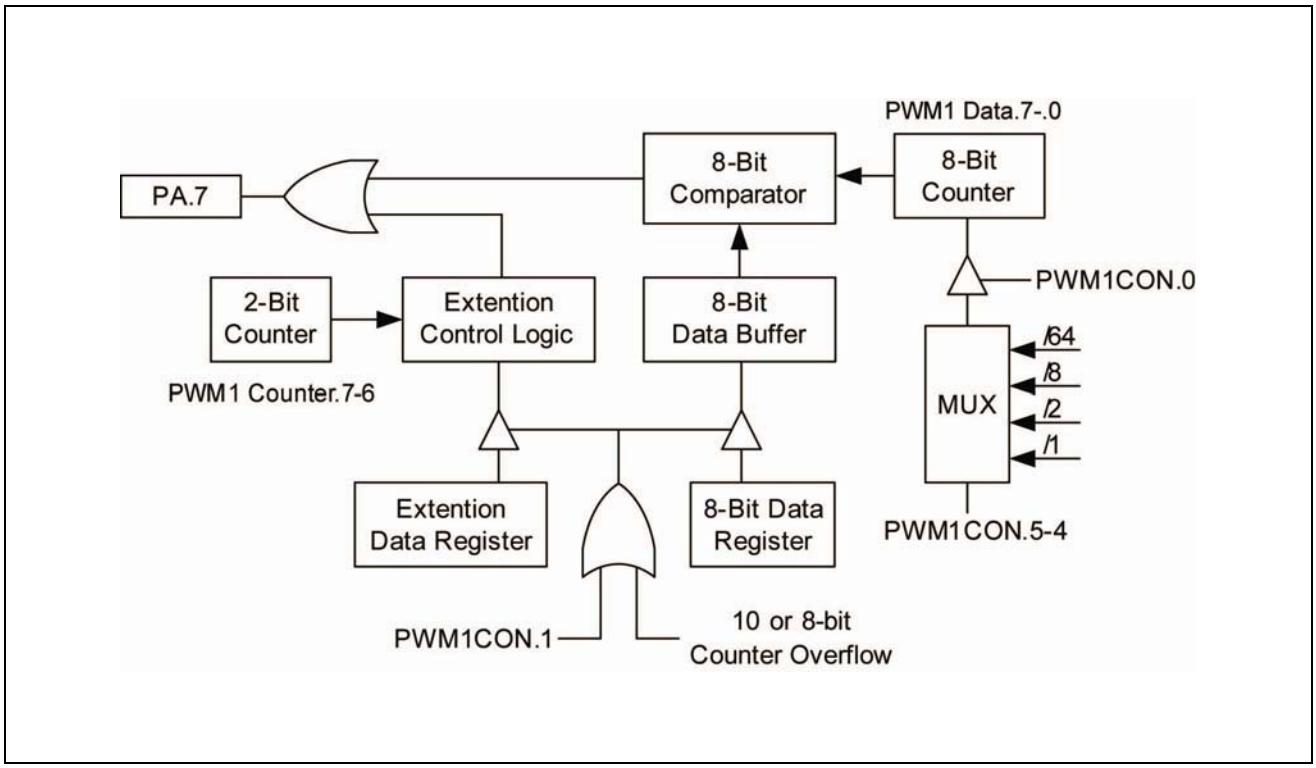
```
MOVLW    05h          ; Set PWM0 Data Register
MOVWF    PWM0DAT      ; Data = 1, Extension = 1

CLRF     PACONH
BSF      PACONH, 4    ; Select PACONH.54 '01' PWM0 Out.

CLRF     PWM0CON      ; fosc/64, 8-bit Overflow Reload, PWM Stop

BSF      PWM0CON, 1   ; PWM0 Counter Clear
BSF      PWM0CON, 0   ; PWM0 Start
.
.
BCF      PWM0CON, 0   ; PWM0 Stop
```

5. 10-Bit PWM

PWM1 has the following functional components:

- Clock frequency selector
- 10-bit up-counter, 8-bit comparator, 8-bit data register and 8-bit data buffer.
- 2-bit extension control logic, 2-bit extension register and extension data buffer.
- Control register (PWM1CON)

To determine the PWM1 operating frequency, the upper 8-bit counter is compared to the PWM1 data register (PWM1DAT). In order to achieve higher resolutions, the 2-bits of the counter can be used to modulate the "extended" cycle.



< Figure 5-1 Block Diagram >

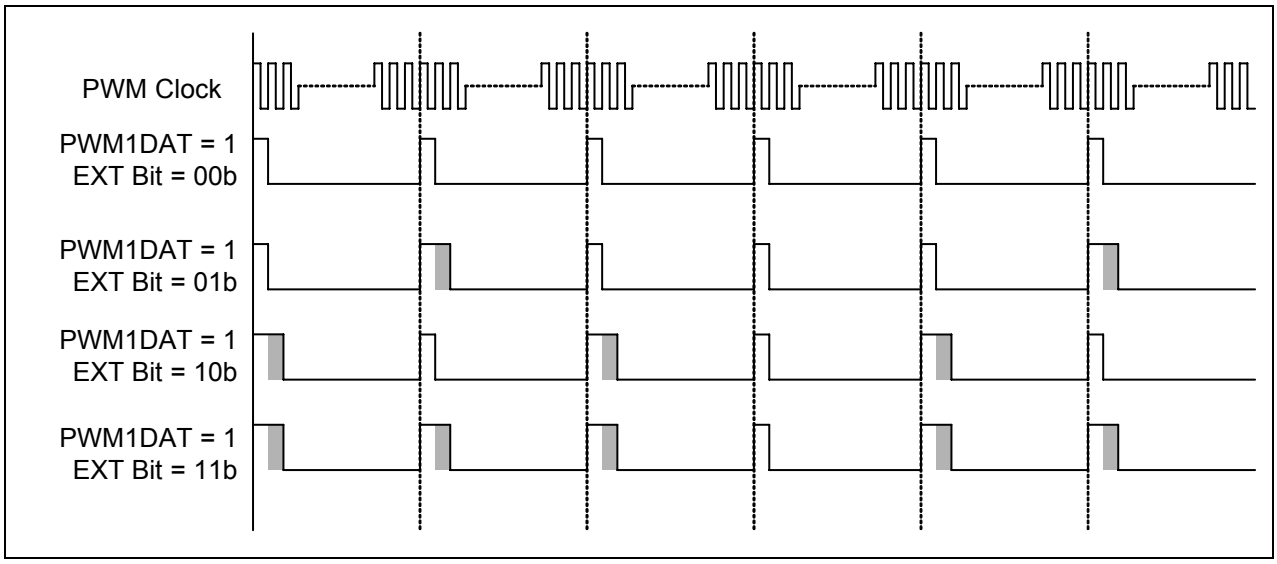
The PWM output signal toggles to Low level whenever the lower 8-bit of counter matches the reference data register (PWM1DAT). If the value in the PWM1DAT register is not zero, an overflow of the lower 8-bits of counter causes the PWM output to toggle to High level. In this way, the reference value written to the reference data register determines the module's base duty cycle.

The value in the upper 2-bits of counter is compared with the extension settings in the 2-bit extension data register (PWM1CON.7-6). This lower 2-bits is used to "extend" the duty cycle of the PWM output. The "extension" value is one extra clock period at specific cycles (see Table 5-1).

PWM1CON.7-6	Extended Cycle
00	None
01	2
10	1, 3
11	1, 2, 3

< Table 5-1 PWM output extended cycle >

For example, if the value in the extension data register is '01B', the 2nd cycle will be one pulse longer than the other 3 cycles. (see Figure 5-2).



< Figure 5-2 Extended Output >

6. Analog to Digital Converter

The 10-bit CMOS ADC (Analog to Digital Converter) consists of a 10-channel analog input multiplexer, control register, clock generator, 10 bit successive approximation register, and output register.

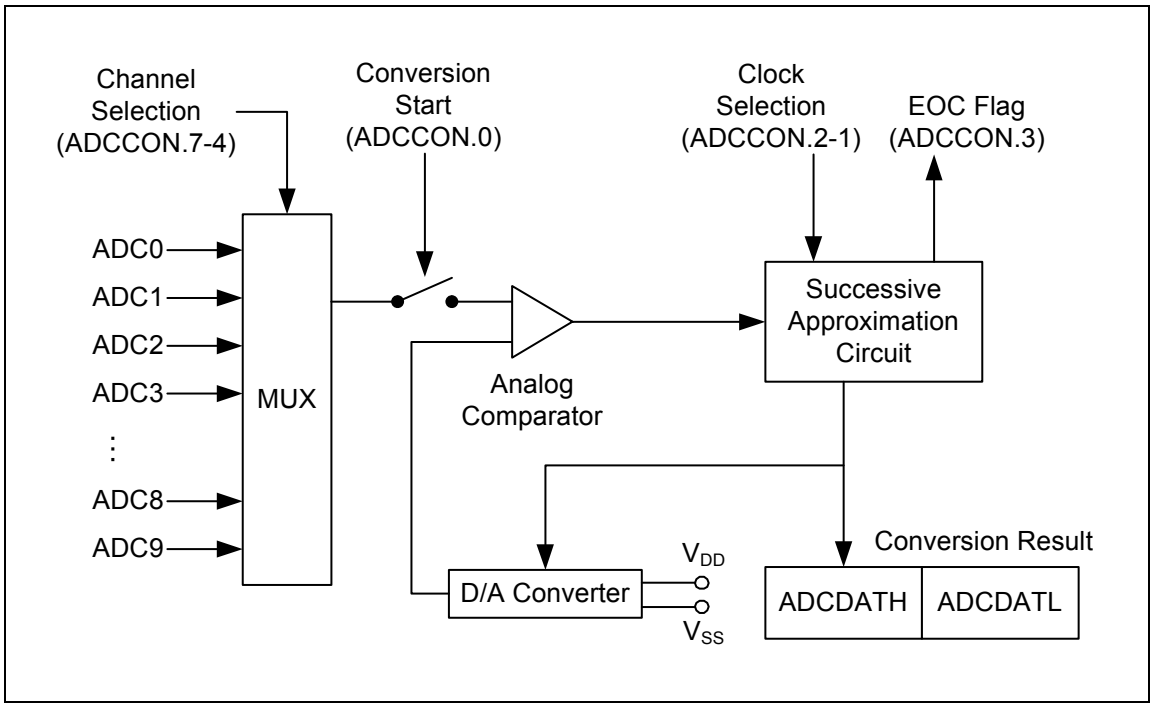
A/D CONVERSION PROCEDURE

1. Configure the analog input pins to ADC input mode by making the appropriate settings in the I/O port control registers.
2. Select ADC input channel.
3. Start conversion by set the ADCCON.0 to '1'.
4. When conversion has been completed, the EOC flag is set to '1'.
5. The converted digital value is loaded to the ADCDATL, ADCDATH register, and then the ADC module enters an idle state.
6. The digital conversion result can now be read from the ADDATAH, ADDATAL register.

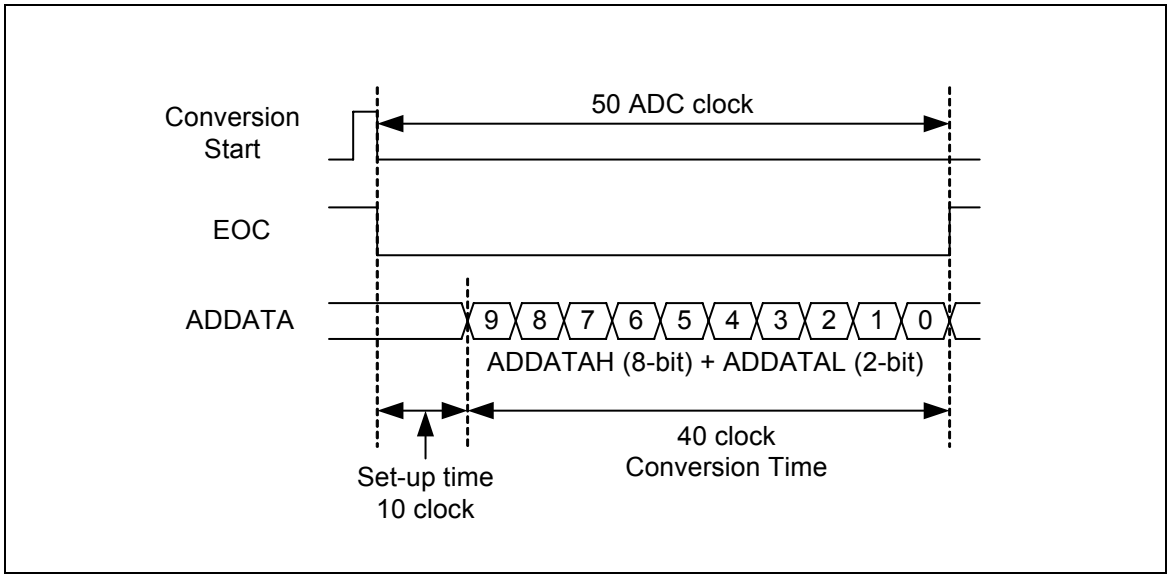
If the chip enters to STOP mode in conversion process, there will be a leakage current path in A/D block. The ADC operation must be finished before the chip enters STOP mode.

※ There is not sampling/hold circuit in ADC. Therefore, it is important that any fluctuations in the analog level at the ADC0–ADC9 input pins during a conversion procedure be kept to an absolute minimum. Any change in the input level, perhaps due to circuit noise, will invalidate the result.

```
      MOVLW    00000100b    ; fosc/4, ADC0
      MOVWF    ADCCON       ; Configure ADCCON

      CLRF     PACONL
      BSF      PACONL, 0
      BSF      PACONL, 1    ; Configure PA.0 ADC Input 0

      BSF      ADCCON, 0    ; Start Conversion
ADC_LOOP:
      BTFSS    ADCCON, 3
      GOTO     ADC_LOOP     ; Wait until EOC bit is set

                            ; Converted value can be read from ADDATL and
                            ; ADDATH.
```



< Figure 6-1 Analog to Digital Converter Block Diagram >



< Figure 6.2 A/D Conversion Timing Diagram >

※ Maximum ADC Input Clock is 4MHz.

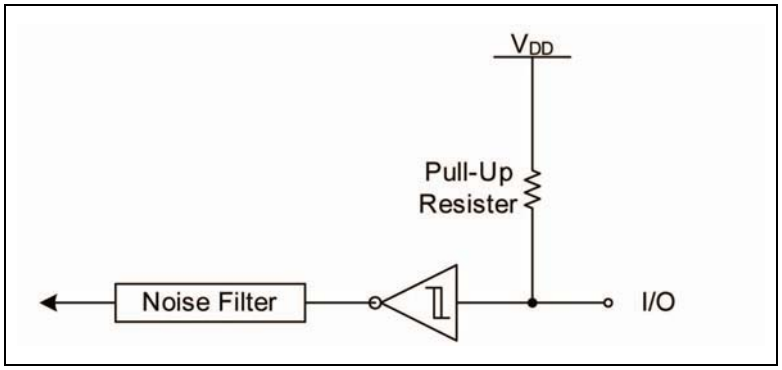
7. I/O Ports

The TM59PA20 has three I/O port, PORTA, PORTB and PORTC (MAX 18 Pin). These ports can be accessed directly by writing or reading port data register.

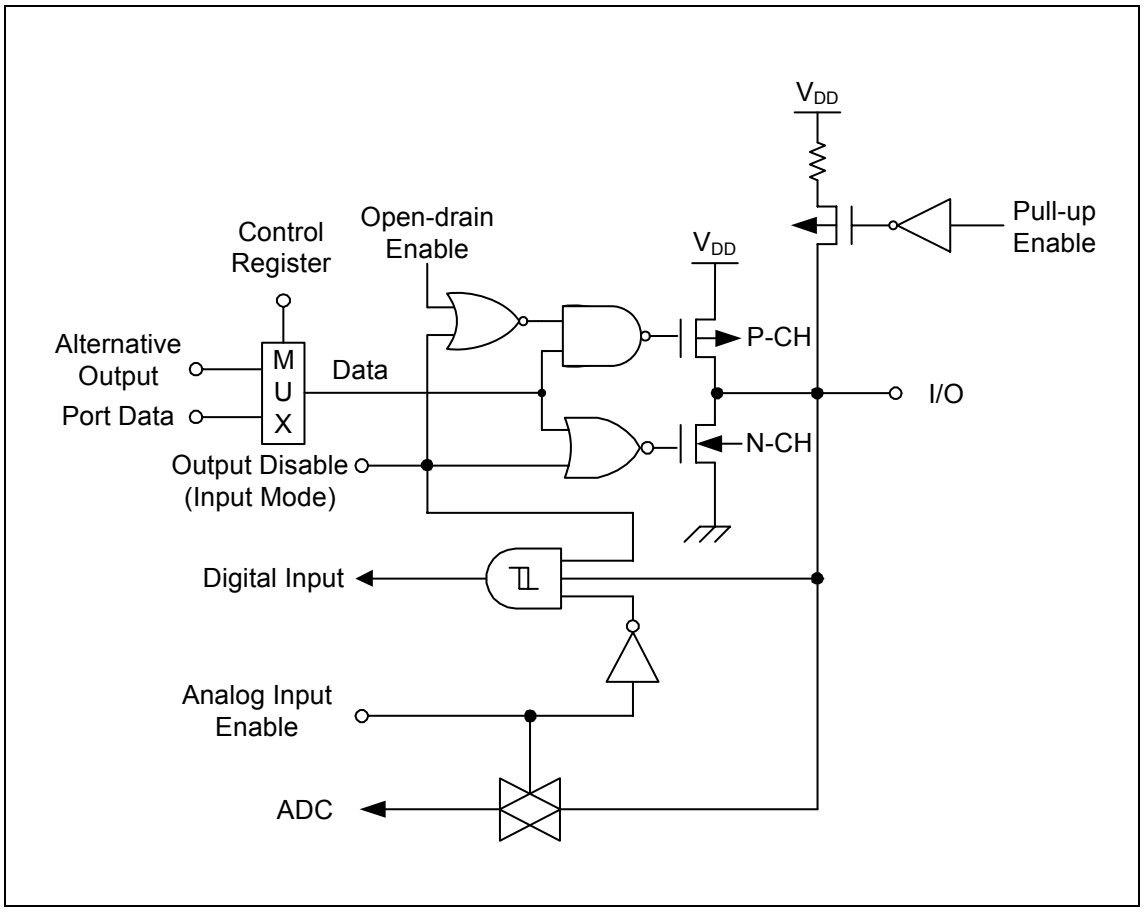
PORT	Bit	Pin No	Pin Description	Input/ Output	PIN Type
PORT A	0	19	Schmitt trigger input, Push-pull output, ADC0, External Interrupt 0	I/O	C
	1	18	Schmitt trigger input, Push-pull output, ADC1, External Interrupt 1	I/O	
	2	17	Schmitt trigger input, Push-pull output, ADC2	I/O	
	3	16	Schmitt trigger input, Push-pull output, ADC3	I/O	
	4	15	Schmitt trigger input, Push-pull output, ADC4	I/O	
	5	14	Schmitt trigger input, Push-pull output, ADC5	I/O	
	6	13	Schmitt trigger input, Push-pull output, ADC6, PWM0	I/O	
	7	12	Schmitt trigger input, Push-pull output, ADC7, PWM1	I/O	
PORT B	0	2	Schmitt-trigger input, Push-pull output, Open-drain Output	I/O	D
	1	3	Schmitt-trigger input, Push-pull output, Open-drain Output	I/O	
	2	4	Schmitt-trigger input	I	A
PORT C	0	5	Schmitt-trigger input, Push-pull output, Open-drain Output, Timer0 match Output	I/O	C
	1	6	Schmitt-trigger input, Push-pull output, Open-drain Output, Buzzer Out	I/O	B
	2	7	Schmitt-trigger input, Push-pull output, Open-drain Output	I/O	
	3	8	Schmitt-trigger input, Push-pull output, Open-drain Output	I/O	
	4	9	Schmitt-trigger input, Push-pull output, Open-drain Output	I/O	
	5	10	Schmitt-trigger input, Push-pull output, Open-drain Output, ADC9	I/O	C
	6	11	Schmitt-trigger input, Push-pull output, Open-drain Output, ADC8, Clock Out	I/O	

< Table 7-1 Port Configuration Overview >

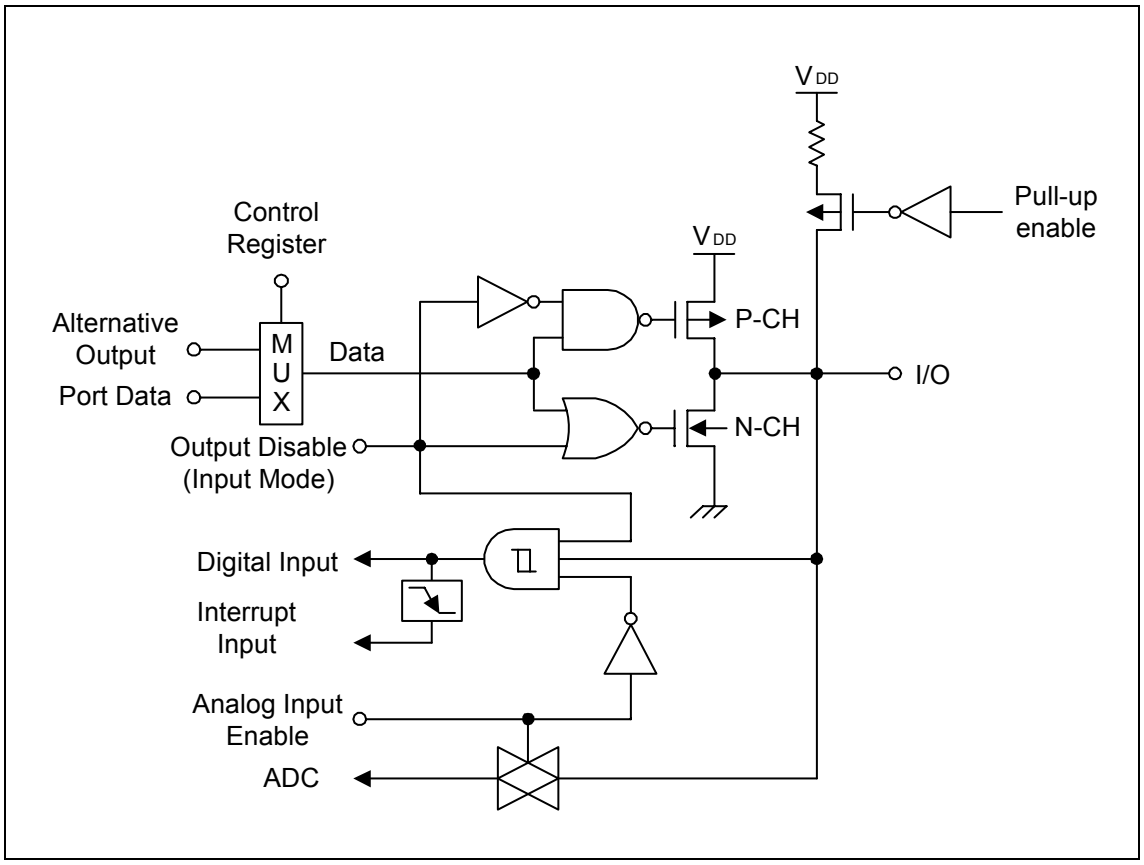
Pin Circuit



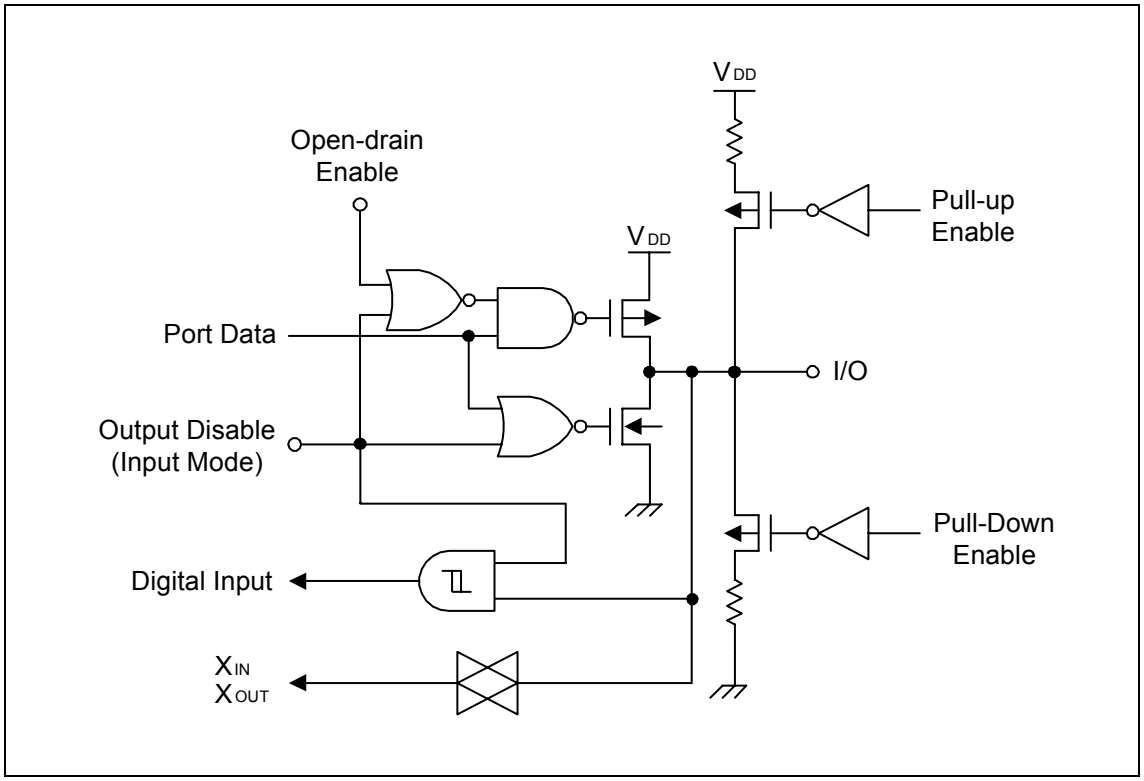
< Figure 7-1 Pin Circuit Type A >



< Figure 7-2 Pin Circuit Type B >



< Figure 7-3 Pin Circuit Type C >



< Figure 7-4 Pin Circuit Type D >

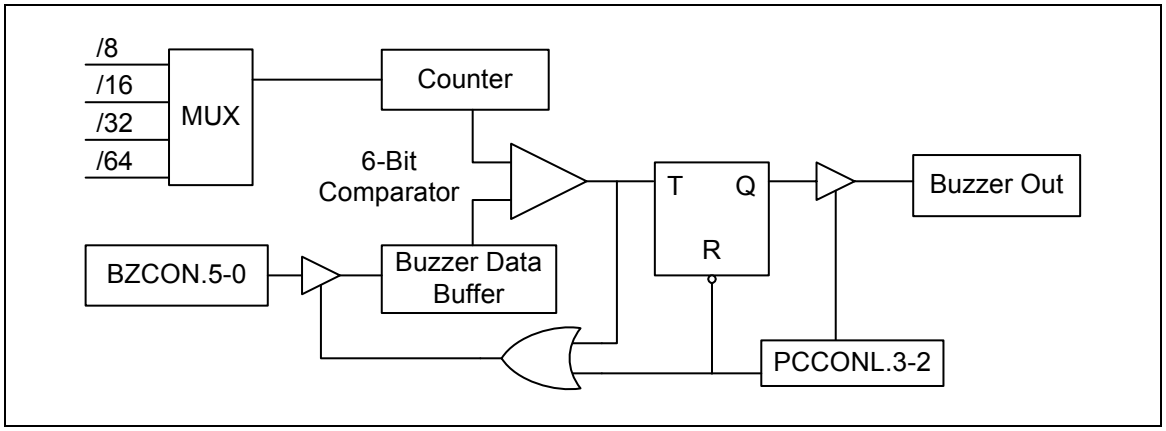
PORTA
Port A has 8-bit I/O Pins. It can be used for normal I/O (Schmitt trigger input, push-pull output, open-drain output) or some alternative function (ADC, External interrupt 0, 1, PWM output).

PORTB
Port B has 3-bit I/O Pins. PortB.1-0 can be used clock input or normal I/O. If the PortB.1-0 pins are used as external clock Input, the control register (PBCON) must be set to output port to prevent current consumption. PortB.2 can be used for input only pin.

PORTC
Port C has 7-bit I/O Pins. It can be used for normal I/O (Schmitt trigger input, push-pull output, open-drain output) or some alternative function (ADC, Clock output, T0 clock output, Buzzer out).

8. Buzzer Out

The TM59PA20 has Buzzer driver that consist of 6-bit counter, clock divider, control register. It generates 50% duty square-wave and the frequency cover a wide range.



< Figure 8-1 Block Diagram >

It can be enabled by setting the bit PC.1 as Buzzer out function. When the Buzzer Out is enabled, the 6-bit counter is cleared and PC.1 output status is '0' and start counting up. If the counter value is match up to period data (BZCON.5-0), then PC.1 output status is toggle and the counter is cleared. Also, the counter is cleared by 6-bit counter overflow. BZCON.5-0 determines output frequency. Frequency calculation is as follows.

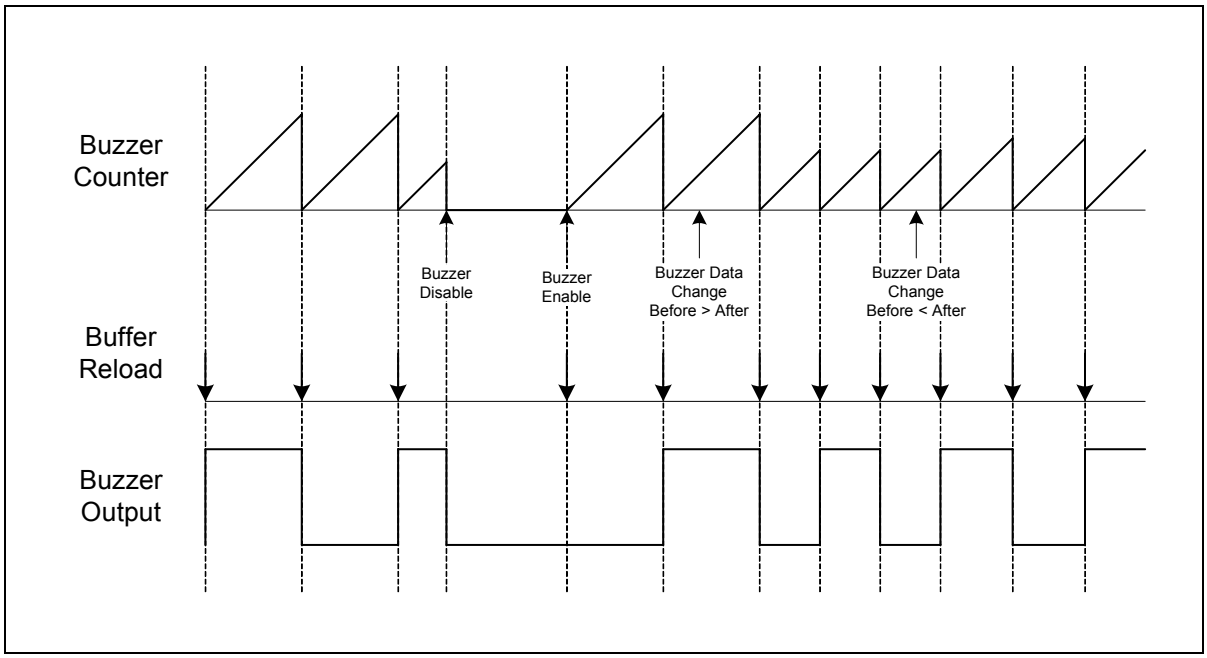
$F_{BZ} = f_{OSC}/2/\text{Prescaler Ratio}/(\text{Period Data} + 1)$

Example 8-1> Output frequency calculation
CPU Clock (f_{OSC}) : 8.192MHz
Prescaler Ratio (BZCON.7-6) : 11 (f_{OSC} /64),
Period Data (BZCON.5-0) : 9

F_{BZ} 8.192M / 2 / 64 / (9+1) = 6400 (Hz)

Example 8-2> Sample Code

```
CLRF    PCCONL      ; Clear PCCONL
MOVLW   11001001b   ; fosc/64, Period Data 9 ( 6.4 KHz Output )
MOVWF   BZCON       ; Set Buzzer 6.4KHz Output
BSF     PCCONL, 2   ; Set PORTC.1 Buzzer Out. Buzzer Enable
.
.
.
BCF     PCCONL, 2   ; Set PORTC.1 Input mode. Buzzer Disable
```



< Figure 8-2 Timing Diagram >

9. Electrical Characteristics

9.1 Absolute Maximum Ratings (T_A = 25°C)

Parameter	Rating	Unit
Supply voltage	– 0.3 to + 6.5	V
Input voltage	– 0.3 to V_{DD} + 0.3	
Output voltage	– 0.3 to V_{DD} + 0.3	
Output current high per 1 PIN	– 25	mA
Output current high per all PIN	– 80	
Output current low per 1 PIN	+ 30	
Output current low per all PIN	+ 150	
Maximum Operating Voltage	5.5	V
Operating temperature	– 40 to + 85	°C
Storage temperature	– 65 to + 150	

9.2 DC Characteristics (T_A = – 40°C to + 85°C, V_{DD} = 2.0V to 5.5V)

Parameter	Symbol	Conditions		Min	Typ	Max	Unit
Input High Voltage	V_{IH1}	Except X_{IN} , X_{OUT}	V_{DD} = 2.0 to 5.5V	0.8 V_{DD}	–	V_{DD}	V
	V_{IH2}	X_{IN} , X_{OUT}		V_{DD} - 0.1			
Input Low Voltage	V_{IL1}	Except X_{IN} , X_{OUT}	V_{DD} = 2.0 to 5.5V	–	–	0.2 V_{DD}	V
	V_{IL2}	X_{IN} and X_{OUT}				0.1	
Output High Voltage(NOTE 1)	V_{OH}	PORT A,B,C	V_{DD} = 4.5 to 5.5V	V_{DD} -1.5	V_{DD} - 0.4	–	V
Output Low Voltage(NOTE 2)	V_{OL}	PORT A,B,C	V_{DD} = 4.5 to 5.5V	–	0.4	2.0	V
Input Leakage Current(pin high)	I_{ILH}	Except X_{IN} , X_{OUT}	V_{IN} = V_{DD}	–	–	1	uA
		X_{IN} and X_{OUT}	V_{IN} = V_{DD}			20	
Input Leakage Current(pin low)	I_{ILL}	Except X_{IN} , X_{OUT}	V_{IN} = 0V	–	–	–1	uA
		X_{IN} and X_{OUT}	V_{IN} = 0V			–20	
Output Leakage Current(pin high)	I_{OLH}	All output pins	V_{OUT} = V_{DD}	–	–	2	uA
Output Leakage Current(pin low)	I_{OLL}	All output pins	V_{OUT} = 0V	–	–	–2	uA
Power Supply Current	I_{DD}	Run 10MHz	V_{DD} = 4.5 to 5.5V	–	7	12	mA
		Run 4MHz	V_{DD} = 2.0V		2	4	
		Stop mode	V_{DD} = 4.5 to 5.5V	–	100	200	uA
			V_{DD} = 2.6V		30	60	
Pull-Up Resistor	R_P	V_{IN} = 0V Ports A, B, C	V_{DD} = 5V	25	50	100	kΩ
Pull-Down Resistor	R_P	V_{IN} = 0V Ports B	V_{DD} = 5V	25	50	100	

NOTE:

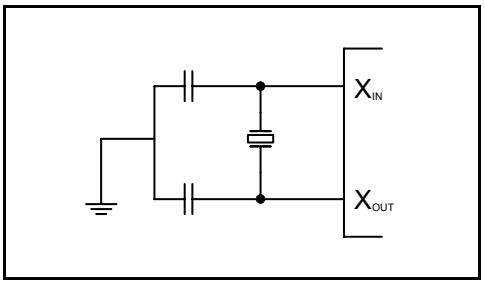
1. Output current high = –10mA
2. Output current Low = 25mA

9.3 Clock Timing Constants (T_A = – 40°C to + 85°C)

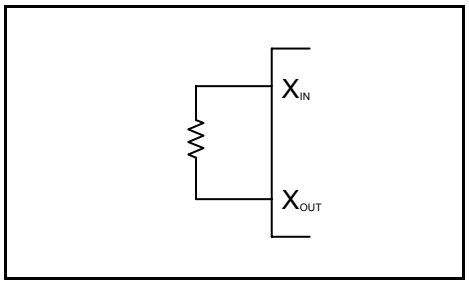
Oscillator	Condition	Min	Typ	Max	Unit
External Clock	V_{DD} = 2.5 to 5.5V	1	–	12	MHz
	V_{DD} = 2.0 to 5.5V	1	–	4	
External RC(NOTE 1)	V_{DD} = 4.75 to 5.25V	–	4	–	
Internal RC(NOTE 2)	V_{DD} = 4.75 to 5.25V		2.8		
			0.47		

NOTE:

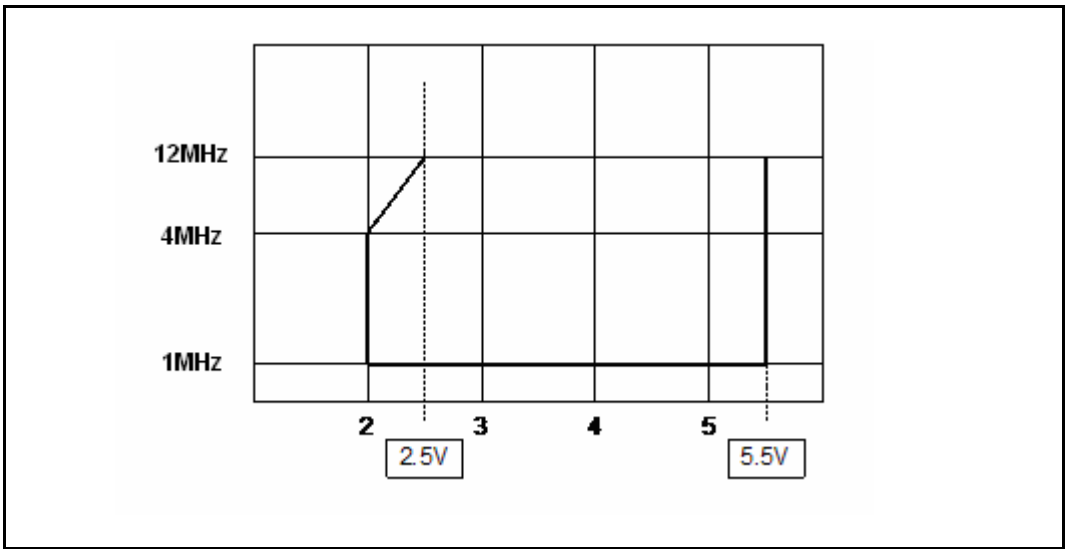
1. Tolerance : ±10 % at T_A = 25°C, Resister Voltage around 2.7K (ohm).
2. Tolerance : ±20 % at T_A = 25°C



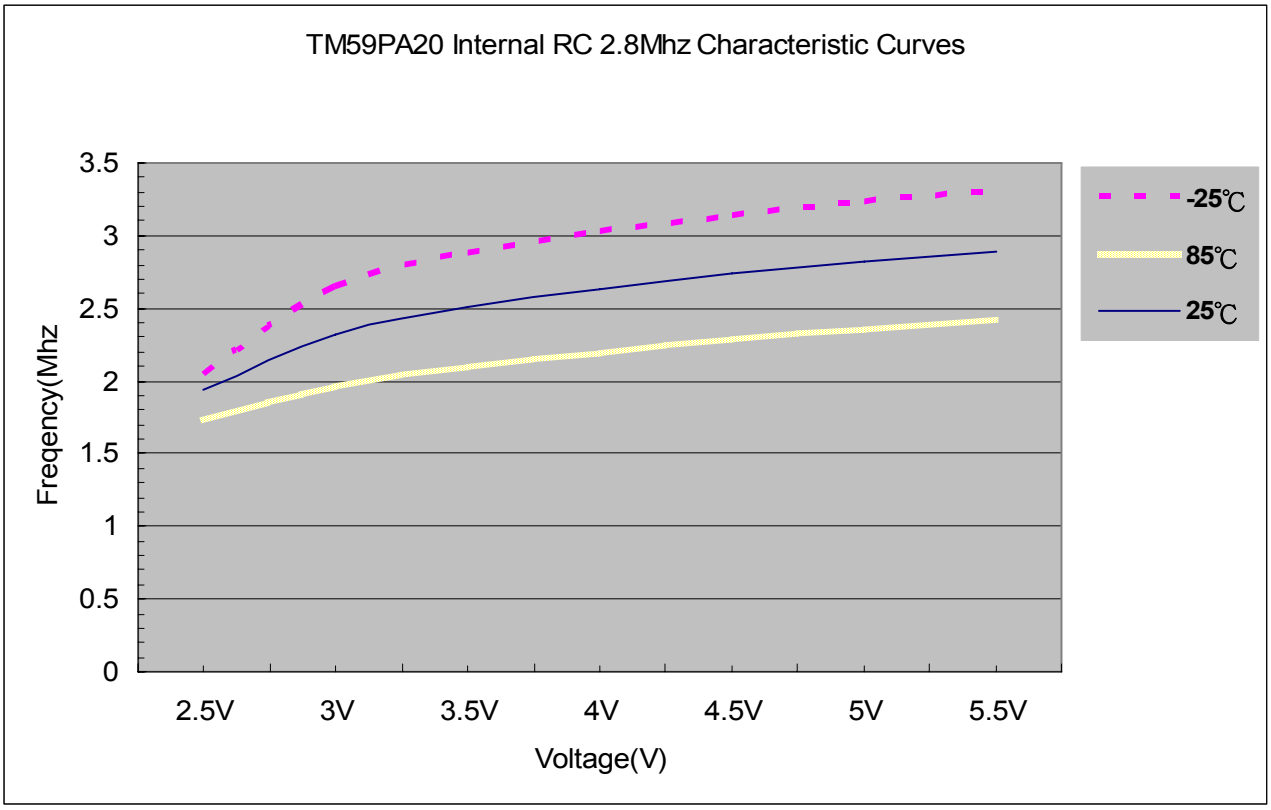
External Oscillator Circuit
(Crystal or Ceramic)

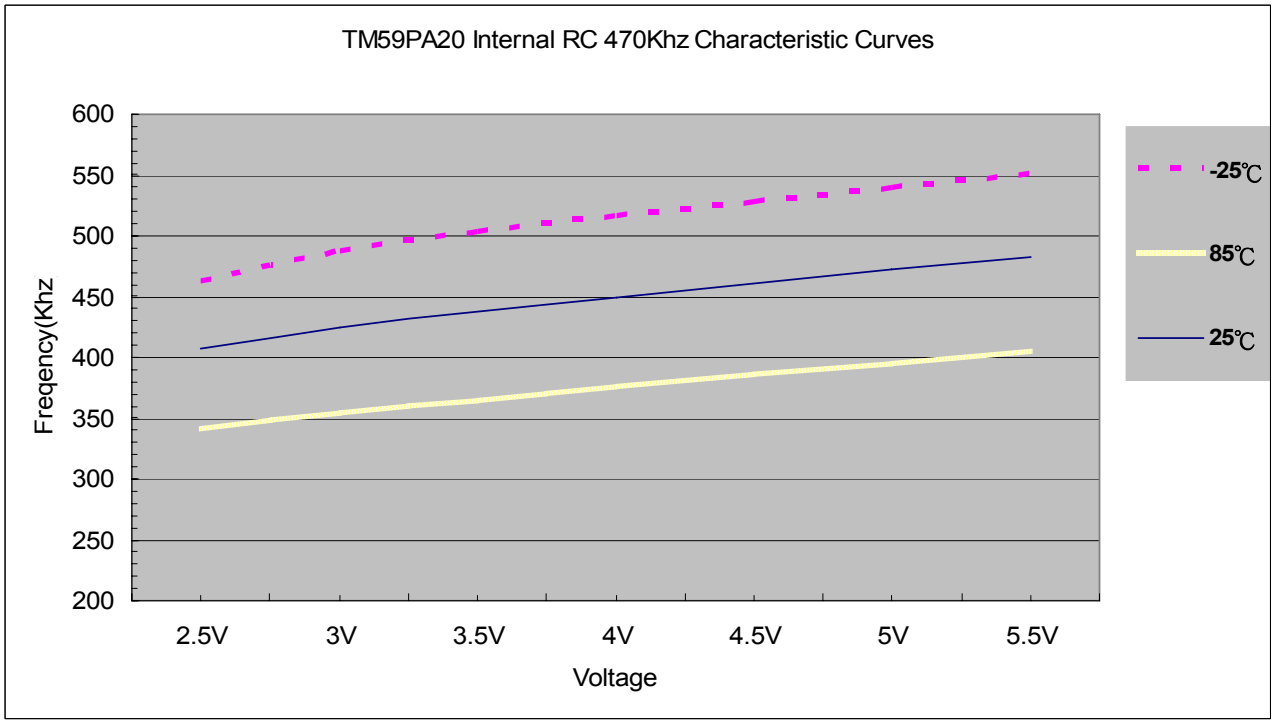


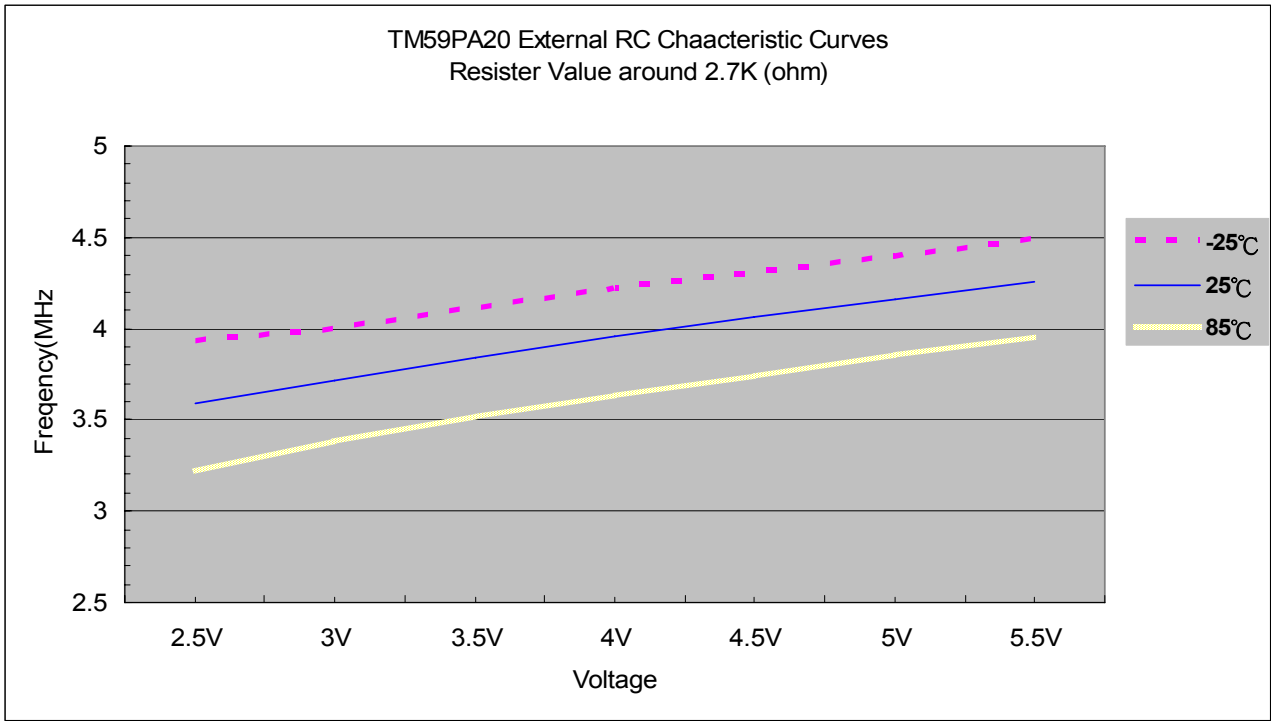
External R-C Oscillator



System Operating Frequency Range

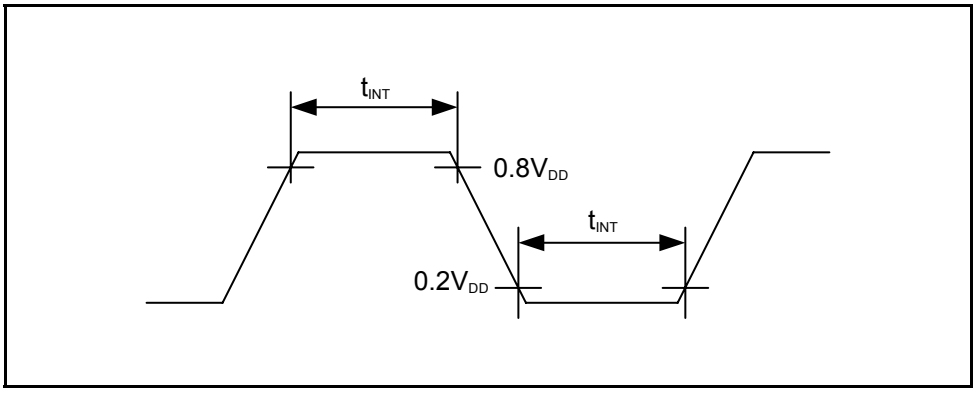

TM59PA20 Internal RC 2.8Mhz Characteristic Curves
Freqency(Mhz
3.5
3
2.5
2
1.5
1
0.5
0
2.5V
3V
3.5V
4V
4.5V
5V
5.5V
Voltage(V)
-25℃
85℃
25℃


TM59PA20 Internal RC 470Khz Characteristic Curves
Freqency(Khz
600
550
500
450
400
350
300
250
200
2.5V
3V
3.5V
4V
4.5V
5V
5.5V
Voltage
-25℃
85℃
25℃



9.4 External Interrupt Characteristics (T_A = – 40°C to + 85°C, V_{DD} = 2.0V to 5.5V)

Parameter	Conditions	Min	Typ	Max	Unit
Input High Voltage	–	0.8 V_{DD}	–	V_{DD}	V
Input Low Voltage	–	–	–	0.2 V_{DD}	V
External Interrupt Input Width(t_{INT})	V_{DD} = 5V ± 10 %	–	200	–	ns



9.5 A/D Converter Electrical Characteristics (T_A = – 40°C to + 85°C, V_{DD} = 2.0V to 5.5V, V_{SS} = 0V)

Parameter	Conditions	Min	Typ	Max	Units
Total Accuracy	V_{DD} = 5.12V, V_{SS} = 0V CPU clock = 10MHz	–	–	± 3	LSB
Integral Non-Linearity		–	–	± 2	
Differential Non-Linearity		–	–	± 1	
Offset Error of Top		–	± 1	± 3	
Offset Error of Bottom		–	± 1	± 2	
Max Input Clock (f_{ADC})	–	–	–	4	MHz
Conversion Time (NOTE 1)	f_{ADC} = 4MHz	–	20	–	µs
Analog Input Voltage	–	V_{SS}	–	V_{DD}	V
Analog Input Impedance	–	2	–	–	MΩ
Analog Input Current	V_{DD} = 5V	–	–	10	µA
Analog Block Current (NOTE 2)	V_{DD} = 5V	–	1	3	mA
	V_{DD} = 3V	–	0.5	1.5	mA
	V_{DD} = 5V stop mode	–	100	500	nA

NOTE:

1. "Conversion time" is the time required from the moment a conversion operation starts until it ends.
2. I_{ADC} is operating current during A/D conversion.

9.6 Reset Timing Characteristics (T_A = – 40°C to + 85°C, V_{DD} = 2.0V to 5.5V)

Parameter	Conditions	Min	Typ	Max	Unit
Input High Voltage	–	0.8 V_{DD}	–	V_{DD}	V
Input Low Voltage	–	–	–	0.2 V_{DD}	V
RESET Input Low Width	Input V_{DD} = 5V ± 10 %	–	1	–	µs
Oscillation stabilization time	f_{OSC} = 1 ~ 12MHz	2.4	–	6.1	ms

9.7 LVR Circuit Characteristics (T_A = – 40°C to + 85°C, V_{DD} = 2.0V to 5.5V)

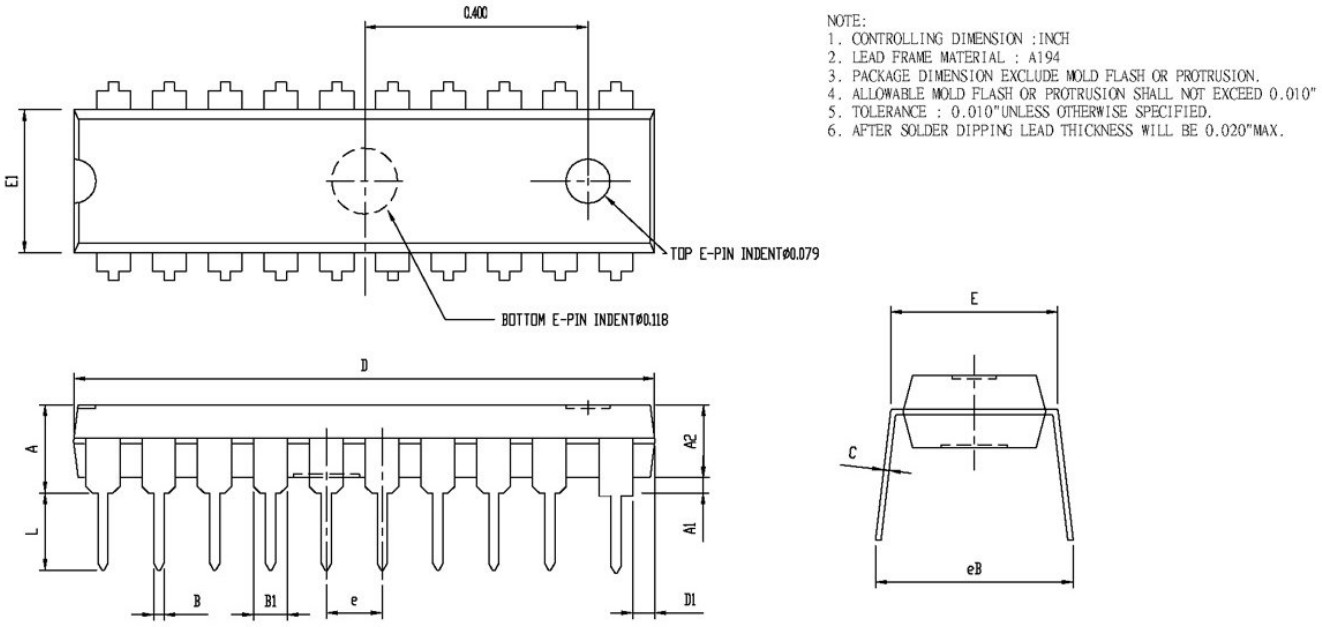
Parameter	Symbol	Min	Typ	Max	Unit
LVR reference Voltage	V_{LVR}	–	2.0 2.3 3.0 3.9	–	V
LVR Hysteresis Voltage	V_{HYST}	–	±0.3	–	V
Low Voltage Detection time	t_{LVR}	1	–	–	µs

10. Packaging Information

The TM59PA20 order information: “IC Type” “XX” “YY” “C” “Z”.

- “IC TYPE”: TM59PA20
- “XX”: Package Type
  - **(1).** DIP Code: D
  - **(2).** SOP Code: S
- “YY”: IC Pin Number
  - **(1).** Pin Number: 16 Code: 16
  - **(2).** Pin Number: 20 Code: 20
- “C”: Reserve (Must write be “C”)
- “Z”: Package material
  - **(1).** Package material: Pb-free Code: W
  - **(2).** Package material: Green Package Code: G

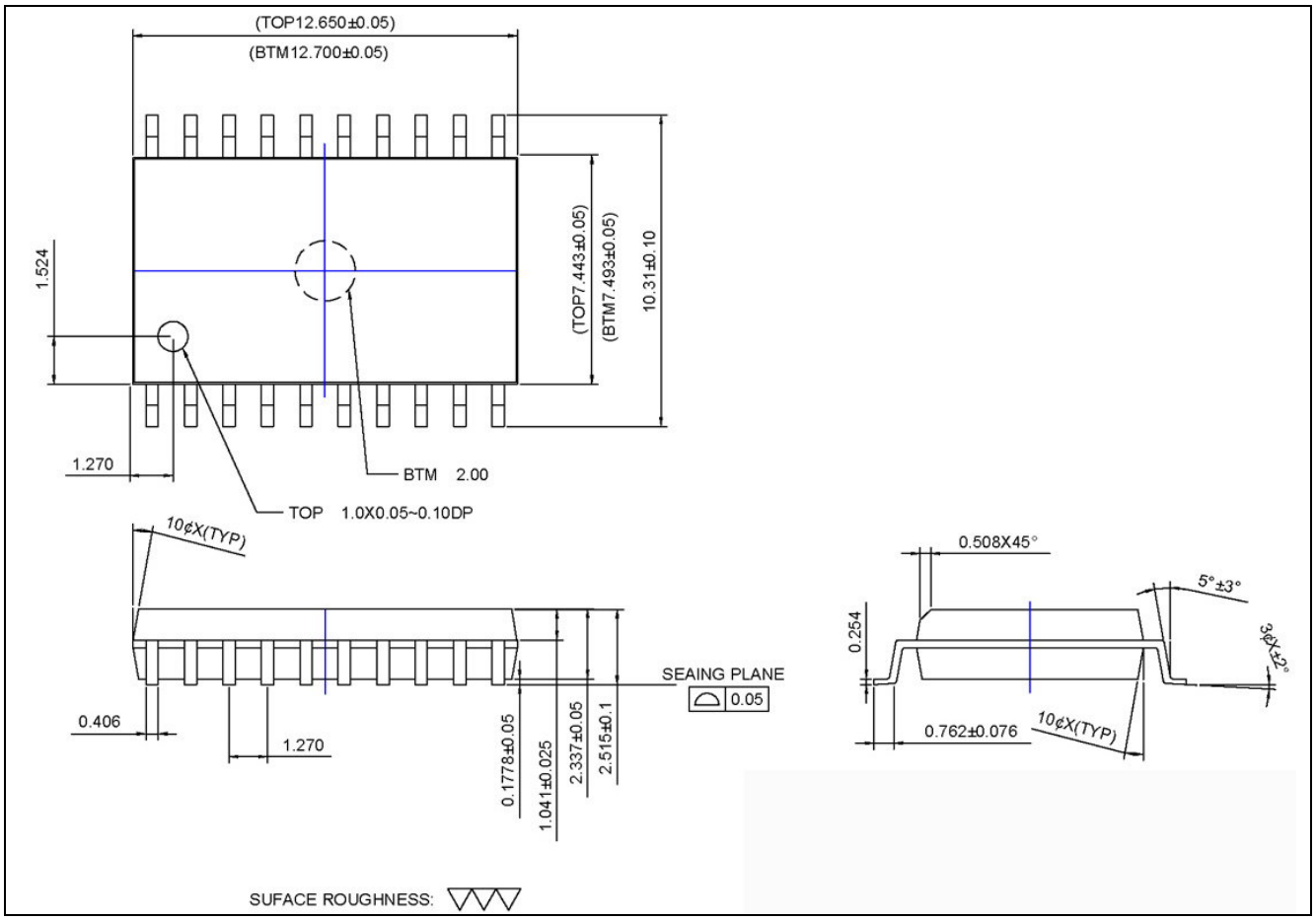
10.1 20-DIP Package Dimension
20 lead, Dual In-line Package
Dimension in Millimeters



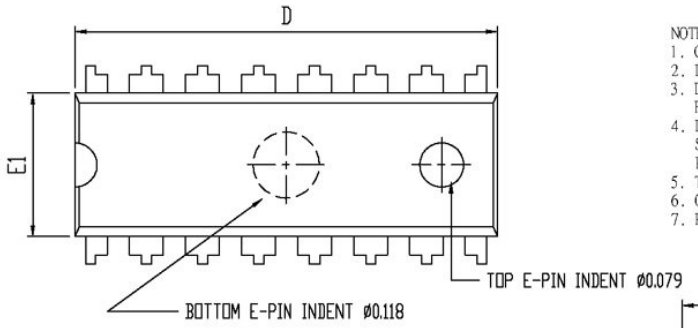
NOTE:
1. CONTROLLING DIMENSION :INCH
2. LEAD FRAME MATERIAL : A194
3. PACKAGE DIMENSION EXCLUDE MOLD FLASH OR PROTRUSION.
4. ALLOWABLE MOLD FLASH OR PROTRUSION SHALL NOT EXCEED 0.010"
5. TOLERANCE : 0.010"UNLESS OTHERWISE SPECIFIED.
6. AFTER SOLDER DIPPING LEAD THICKNESS WILL BE 0.020"MAX.

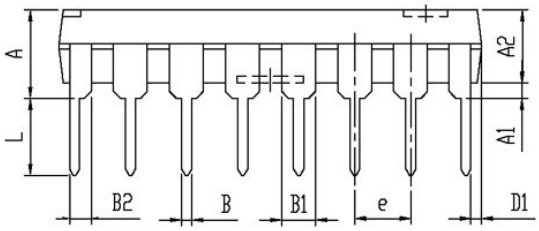
SYMBOLS	DIMENSIONS IN MILLIMETERS			DIMENSIONS IN INCHES		
	MIN	NOM	MAX	MIN	NOM	MAX
A	—	—	4.57	—	—	0.180
A1	0.38	—	—	0.015	—	—
A2	—	3.30	3.56	—	0.130	0.140
B	0.36	0.46	0.56	0.014	0.018	0.022
B1	1.27	1.52	1.78	0.050	0.060	0.070
C	0.20	0.25	0.33	0.008	0.010	0.013
D	26.32	26.416	26.52	1.036	1.040	1.044
D1	0.43	0.56	0.69	0.017	0.022	0.027
E	7.62	—	8.26	0.300	—	0.325
E1	6.40	6.50	6.65	0.252	0.256	0.262
e	—	2.54	—	—	0.100	—
L	3.18	—	—	0.125	—	—
eB	8.38	—	9.65	0.330	—	0.380

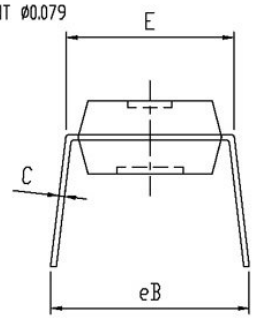
10.2 20-SOP Package Dimension
20 lead, Small Outline Package
Dimension in Millimeters



10.3 16-DIP Package Dimension
16 lead, Dual In-line Package
Dimension in Millimeters



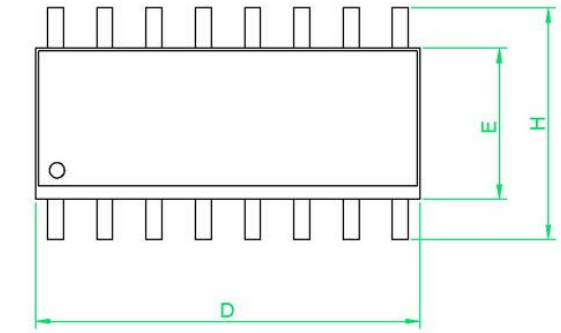


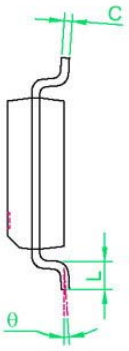


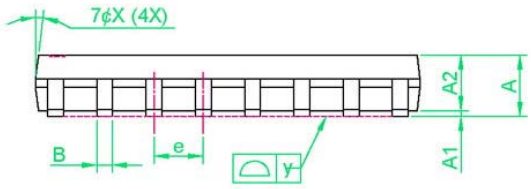
NOTE:
1. CONTROLLING DIMENSION : INCH
2. LEAD FRAME MATERIAL : C194
3. DIMENSION D AND E1 DO NOT INCLUDE MOLD FLASH OR PROTRUSIONS. MOLD FLASH OR PROTRUSIONS SHALL NOT EXCEED 0.010"(0.25mm)
4. DIMENSION "B1" DO NOT INCLUDE DAMBAR PROTRUSION.DAMBAR PROTRUSIONS SHALL NOT EXCEED 0.010"(0.25mm).DISTANCE BETWEEN LEADS INCLUDING DAMBER PROTRUSIONS TO BE 0.005"(0.13mm) INCH MININUM.
5. TOLERANCE : ±0.010"(0.25mm)UNLESS OTHERWISE SPECIFIED.
6. OTHERWISE DIMENSION FOLLOW ACCEPTABLE SPEC.
7. REFERENCE DOCUMENT:JEDEC SPEC MS-001-AC

SYMBOLS	DIMENSIONS IN MILLIMETERS			DIMENSIONS IN INCHES		
	MIN	NOM	MAX	MIN	NOM	MAX
A	—	—	4.57	—	—	0.180
A1	0.38	—	—	0.015	—	—
A2	3.25	3.30	3.45	0.128	0.130	0.136
B	0.36	0.46	0.56	0.014	0.018	0.022
B1	1.27	1.52	1.78	0.050	0.060	0.070
B2	0.81	0.99	1.17	0.032	0.039	0.046
C	0.20	0.25	0.33	0.008	0.010	0.013
D	18.90	19.15	19.30	0.744	0.754	0.760
D1	0.33	0.46	0.58	0.013	0.018	0.023
E	7.62	—	8.26	0.300	—	0.325
E1	6.35	6.50	6.65	0.250	0.256	0.262
e	—	2.54	—	—	0.100	—
L	3.18	—	—	0.125	—	—
eB	8.64	—	9.65	0.340	—	0.380

10.4 16-SOP Package Dimension
16 lead, Small Outline Package
Dimension in Millimeters







SYMBOLS	DIMENSIONS IN MILLIMETERS			DIMENSIONS IN INCHES		
	MIN	NOM	MAX	MIN	NOM	MAX
A	1.35	1.60	1.75	0.053	0.063	0.069
A1	0.10	----	0.25	0.004	----	0.010
A2	----	1.45	----	----	0.057	----
B	0.33	----	0.51	0.013	----	0.020
C	0.19	----	0.25	0.007	----	0.010
D	9.80	----	10.00	0.386	----	0.394
E	3.80	----	4.00	0.150	----	0.157
e	----	1.27	----	----	0.050	----
H	5.80	----	6.20	0.228	----	0.244
L	0.40	----	1.27	0.016	----	0.050
y	----	----	0.10	----	----	0.004
θ	0°	----	8°	0°	----	8°

NOTE
1. PACKAGE BODY SIZES EXCLUDE MOLD FLASH AND GATE BURRS
2. DIMENSION L IS MEASURED IN GAGE PLANE
3. TOLERANCE 0.10 mm UNLESS OTHERWISE SPECIFIED
4. CONTROLLING DIMENSION IS MILLIMETER. CONVERTED INCH DIMENSIONS ARE NOT NECESSARILY EXACT.
5. FOLLOWED FROM JEDEC MS-012