YSM142

Data Sheet

8 Bit General Purpose OTP MCU

21-I/O with PWM

Preliminary

Revision History

1.0	Initial Version
1.1	Preliminary Version
1.2	DC Characteristic Modify Interrupt table modify – remove SPI INT description
1.21	Add the notes for the OSCI/OSCO pin setting while in IRC mode Add the missed description of the items in SFR-WDTS
1.22	DC Characteristic Modify

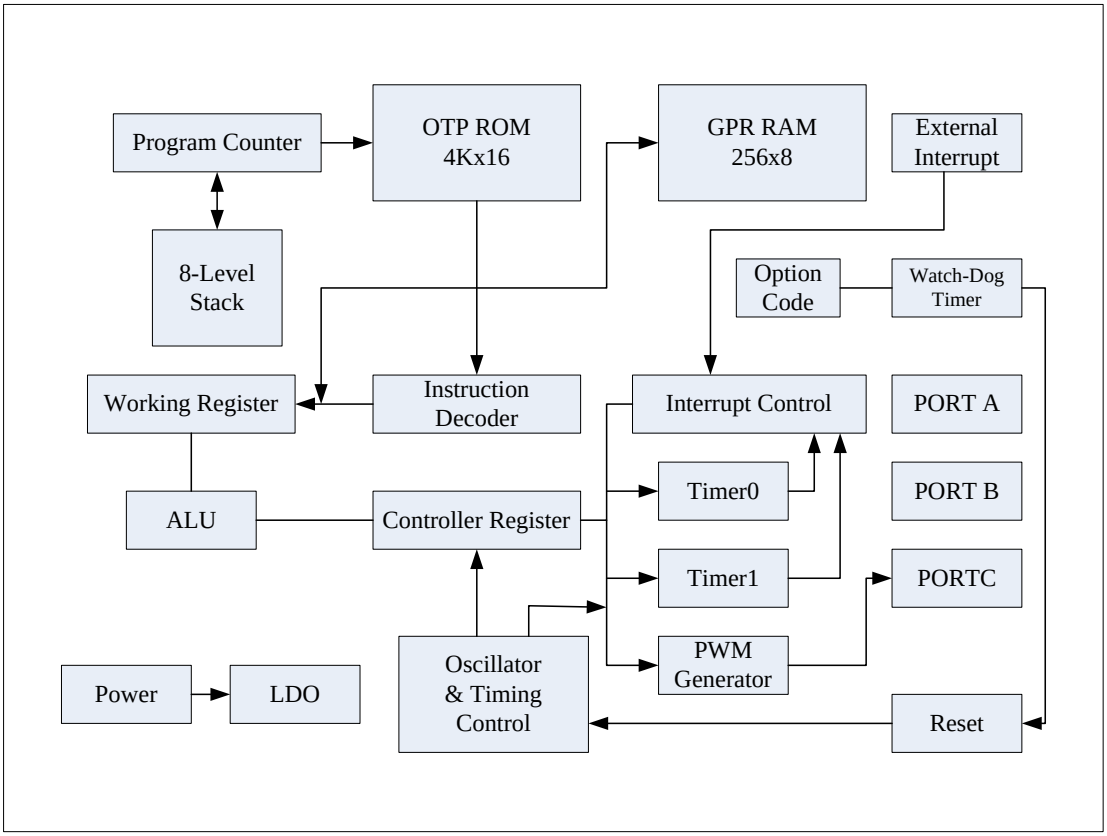
1.0 Device Overview

The YSM142 is a 21 I/O PIN OTP-Base 8-bit microcontroller that is high performance, low cost RISC-MCU.
The YSM142 device has eight-level deep stack, and external interrupt sources.
The YSM142 is a 16-bit wide instruction word, support the direct and indirect addressing mode.
There are four oscillator options, High Speed frequency crystal mode, Low-Speed frequency crystal mode, External Resistor/Capacitor oscillator mode and Internal Resistor/Capacitor mode.
The sleep mode offers power saving , user can wake-up YSM142 from sleep through external , internal and Reset.

1.2 Features:

- Only 50 instructions to learn.
- All instructions are single cycle except for program branches which are two cycle.
- 16-bit wide instruction.
- 8-level deep hardware stack.
- 4K x 16-bit OTP-ROM Program Memory.
- 256 x 8-bit RAM Data Memory.
- 21 General Purpose I/O PIN.
- Built-in Internal RC oscillator.
- Two system clocks per instruction cycle.
- 8 I/O Wake up PIN.
- One I/O PIN with PWM Output.
- Timer0 & Timer1 Interrupt Source.
- 8-bit real time clock/counter
- Power On Reset, External Reset & Watch-Dog Timer Reset.

1.3 Function Block



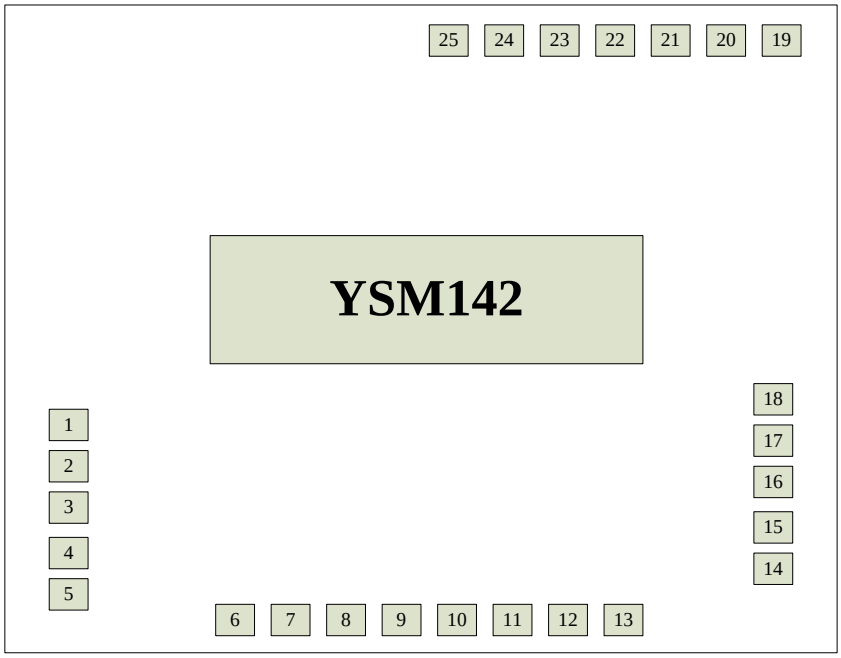
1.4 Pin Description

Pin Name	Pin Number	Type	Description
VDD	14	P	Power PAD, Positive Supply
VDDL	15	P	(a) System Power < 3.6V Connected with VDD. (b) System Power > 3.6V Connected with 0.1uF capacitor to VSS
VSS	1	P	Ground PAD
RSTN	25	I	Reset Pin, Low Active, Smit-Trigger with pull-up
EINT/ PB0	2	I/O	Pin function defined by Code-Option (a) EINT : Interrupt PIN (b)PB0 : Bi-direction I/O with open-drain & pull-up

			Wake-Up PIN
OSCI/ **PC0**	18	I	Pin function defined by Code-Option (a) OSCI : Xtal Mode (Oscillator Crystal Iutput) RC Mode (Connected a resistor to VDD) (b) PC0 : General Input Pin
OSCO/ **PC1**	17	O	Pin function defined by Code-Option (a) OSCO : Xtal Mode (Oscillator Crystal output) RC Mode (Output internal system clock) (b) PC1 : General Input Pin
PA0 **PA1** **PA2** **PA3** **PA4** **PA5** **PA6** **PA7**	19 20 22 23 24 21 9 11	I/O I/O	Bi-direction I/O with pull-up Bi-direction I/O with pull-up
PB1 **PB2** **PB3** **PB4** **PB5** **PB6** **PB7**	4 5 6 7 8 12 13	I/O	Bi-direction I/O with pull-up Wake-Up PIN
PC2 **PC3**	16 3	I/O	Bi-direction I/O
PC4	10	I/O	Bi-direction I/O PWM Output

Notes. *OSCI/OSCO pins should pull-high or pull-low if the system clock source is from the internal RC and the port being set as input.*

1.5 Pad Location :



NO.	PAD NAME	X	Y		NO.	PAD NAME	X	Y
1	VSS	-503	-32		15	VDDL	469	-324
2	EINT/PB0	-469	-119		16	PC2	469	-231
3	PC3	-469	-209		17	OSCO/PC1	469	-142
4	PB1	-469	-299		18	OSCI/PC0	469	-52
5	PB2	-469	-389		19	PA0	467	542
6	PB3	-313	-542		20	PA1	377	542
7	PB4	-223	-542		21	PA5	287	542
8	PB5	-133	-542		22	PA2	197	542
9	PA6	-43	-542		23	PA3	107	542
10	PC4	47	-542		24	PA4	17	542
11	PA7	137	-542		25	RSTN	-73	542
12	PB6	227	-542					
13	PB7	317	-542					
14	VDD	469	-414					

Table of Contents

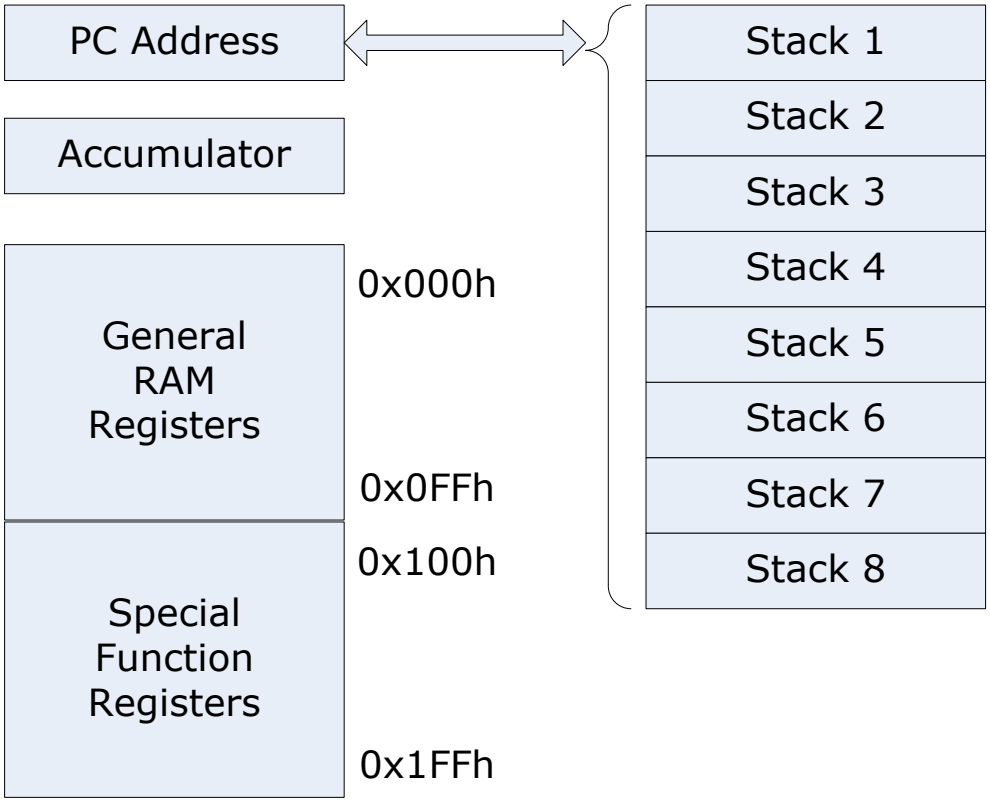
2.0 Memory Organization

2.1 ROM Structure

Address	***Description***
0x0000	**Program Area** General coding area.
0x0001	**Hardware Interrupt Vector** Timer0, Timer1, External Interrupt.
0x0002	**Software Interrupt Vector** When do the INT Instruction.
0x0003 ~ 0x0FFE	**Program Area** General coding area.
0x0FFF	**Reset Vector** Power-On , RSTN PIN go low.

Address	***Description***
N/A	**Option Codes** The ROM space which used to store the device options.

2.2 RAM Structure



There are 256 bytes general purpose registers(GPRs) in RAM of the MCU, User can use these GPRs by direct addressing without any page/bank change commands.
The 2nd block of the RAM space are the Special Function Registers(SFRs). These are being accessed like GPRs by direct addressing.
Notes. GPRs/SFRs can only move the value to/from the accumulator.

2.3 RAM – Special Function Registers(SFRs) Overview

YSM142 Special Function Registers									
NAME	**ADDR**	**BIT 7**	**BIT 6**	**BIT 5**	**BIT 4**	**BIT 3**	**BIT 2**	**BIT 1**	**BIT 0**
INDR	0x100	**Indirect Addressing Register**							
TMR0	0x101	**8 bit Real Time Clock / Counter**							
PCL	0x102	**Low 8 bits Program Counter Register**							
STATUS	0x103	GPR	-	-	/TO	/PD	Z	DC	C
RAMS	0x104	**Indirect Data Memory Address Point**							
PORTA	0x105	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
PORTB	0x106	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
PORTC	0x107				PC4	PC3	PC2	PC1	PC0
MCRW	0x122	/PUE	GIE	-	-	PAB	PS2	PS1	PS0
MCRR	0x124	/PUE	GIE	-	-	PAB	PS2	PS1	PS0
IORA	0x125	IOA7	IOA6	IOA5	IOA4	IOA3	IOA2	IOA1	IOA0
IORB	0x126	IOB7	IOB6	IOB5	IOB4	IOB3	IOB2	IOB1	IOB0
IORC	0x127				IOC4	IOC3	IOC2	IOC1	IOC0
T1CON	0x12C	-	-	-	-	-	TR1S	TR1P1	TR1P0
IOPH	0x12D							/PUB	/PUA
WDTS	0x12E	-	ODE	WDTE	SLP2E	-	-	-	/WUE
INTE	0x12F	-	-	-	-	TR1IE		EXIE	TR0IE
PWM1D	0x130	**PWM1 Duty Cycle Data Register**							
PWM1P	0x132	**PWM1 Period Data Register**							
PWMC	0x134		PW1E	-	-	-	PWS2	PWS1	PWS0
TMR1	0x13E	**Timer1 Value Register**							
TR1CV	0x13F	**Timer 1 Comparator Value Register**							
PAL	0x14B	**Low- Byte Address of Data Latch**							
PAH	0x14C	**High-Byte Address of Data Latch**							
INTF	0x17F	-	-	-	-	TR1IF		EXIF	TR0IF

2.4 Special Function Registers(SFRs) Briefing

◆ **INDR** (0x100) : ***Indirect Addressing Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
INDR	Indirect Addressing Register							
R/W	R/W							
RESET	-	-	-	-	-	-	-	-

The INDR Register is not a physical register. Addressing INDR actually address the register that RAMS address value. User should use SFR:INDR with SFR:RAMS to access the indirect addressing data.
See : Page 15

◆ **TMR0** (0x101) : ***8Bits Real Timer Clock/Counter***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
TMR0	8 bit Real Time Clock / Counter							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RESET	0	0	0	0	0	0	0	0

The TMR0 special function register is the clock counter value which the clock source is from an 8bits real time clock. And the clock source of the 8bits real time clock is from the instruction cycle(Fosc / 2).

◆ **PCL** (0x102) : ***Program Counter Lower 8 bits data***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
TMR0	Low 8 bits Program Counter Register							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RESET								

The PCL special function register is the lower 8 bits data of the program counter.

◆ **STATUS** (0x103) : ***Status Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
STATUS	GPR	-	-	/TO	/PD	Z	DC	C
R/W	R/W			R/W	R/W	R/W	R/W	R/W
RESET	0	0	0	1	1	0	0	0

GPR : ***General Purpose Read/Write bit***

/TO : ***Time- Out Flag***

/TO=0 Watch dog Timer Overflow.

/TO=1 (a) Power On

(b) Execute WDTC or SLEEP Instruction.

/PD : ***Power Down Flag***

/PD=0 After SLEEP Instruction.

/PD=1 (a) Power On

(b) Execute WDTC Instruction.

Z : ***ALU Operation Zero Flag***

Z=0 Operation Result is not Zero.

Z=1 Operation Result is Zero.

DC : ***ALU Operation Half Carry /Borrow Flag***

DC=0 Half Carry/Borrow does not occur.

DC=1 Half Carry/Borrow occurred.

C : ***ALU Operation Carry /Borrow Flag***

C=0 Carry/Borrow does not occur.

C=1 Carry/Borrow occurred.

◆ **RAMS** (0x104) : ***Indirect Addressing Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RAMS	Indirect Data Memory Address Point							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RESET								

The RAMS register is a pointer for Indirect Addressing . User should use SFR:INDR with SFR:RAMS to access the indirect addressing data.
See : Page 15

◆ **PORTA** (0x105) : ***Port A Data***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
PORTA	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RESET	0	0	0	0	0	0	0	0

PA0~PA7 : *I/O Port Read/Write Data*

◆ **PORTB** (0x106) : ***Port B Data***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
PORTB	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RESET	0	0	0	0	0	0	0	0

PB0~PB7 : *I/O Port Read/Write Data*

◆ **PORTC** (0x107) : ***Port C Data***

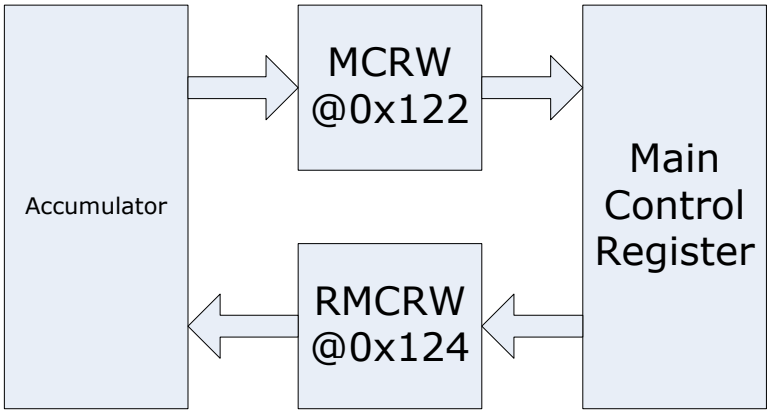
NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
PORTC				PC4	PC3	PC2	PC1	PC0
R/W				R/W	R/W	R/W	R/W	R/W
RESET				0	0	0	0	0

PC0~PC4 : *I/O Port Read/Write Data*

◆ **MCRW** (0x122) : ***Main Control Register for Write***

◆ **MCRR** (0x124) : ***Main Control Register for Read***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MCRR	/PUE	GIE	-	-	PAB	PS2	PS1	PS0
R/W	R/W	R/W			R/W	R/W	R/W	R/W
RESET								



/PUE : ***I/O Port Pull-Up Resistor Enable*** .

/PUE=0 : I/O Connected with Pull-Up Resistor.

/PUE=1 : Pull-Resistor Disconnected.

GIE : ***Global Interrupt Enable.***

GIE=0 Disable all interrupts. Execute DISI Instruction or accept Interrupt then GIE will be set to '0'. Reject any other Interrupt.

GIE=1 Global Enable all the Interrupt (Enable or not depend on others Register Enable Bit.
Execute ENI or RETI will be set to '1'. then accept others Interrupt.

PAB : ***Prescaler bit Assignment .***

PAB=0 Assign to TMR0 (Timer 0).

PAB=1 Assign to WDT (Watch-Dog Timer).

(Continue)

PS2~PS0 : ***Prescaler bits.***

PS2	PS1	PS0	TMR0 Rate	WDT Rate
0	0	0	1:2	1:1
0	0	1	1:4	1:2
0	1	0	1:8	1:4
0	1	1	1:16	1:8
1	0	0	1:32	1:16
1	0	1	1:64	1:32
1	1	0	1:128	1:64
1	1	1	1:256	1:128

◆ **IORA** (0x125) : ***Port A I/O Direction Setting***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
IORA	IOA7	IOA6	IOA5	IOA4	IOA3	IOA2	IOA1	IOA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RESET	1	1	1	1	1	1	1	1

◆ **IORB** (0x126) : ***Port B I/O Direction Setting***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
IORB	IOB7	IOB6	IOB5	IOB4	IOB3	IOB2	IOB1	IOB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RESET	1	1	1	1	1	1	1	1

◆ **IORC** (0x127) : ***Port C I/O Direction Setting***

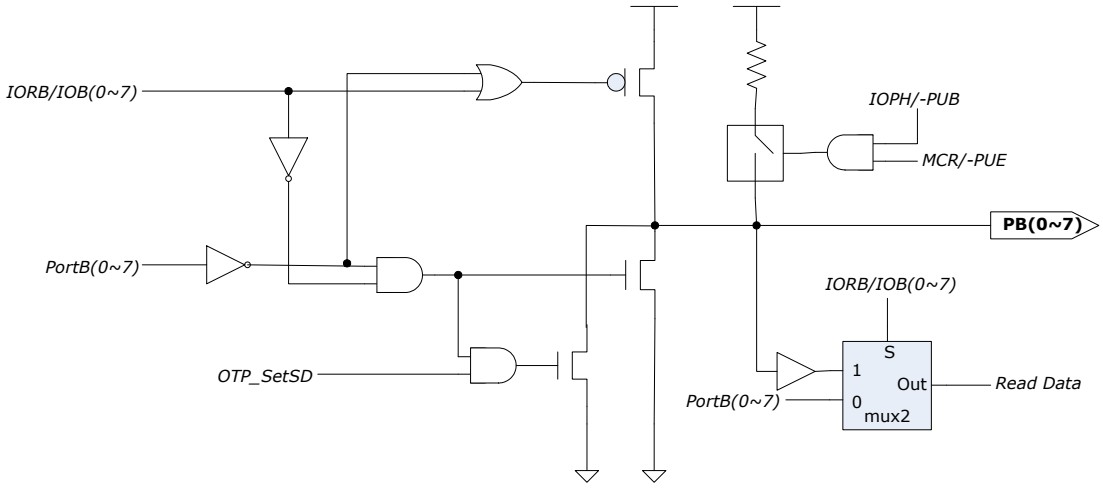
NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
IORC	-	-	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
RESET	-	-	1	1	1	1	1	1

IOx0~IOx7 : *I/O as a Input or Output*

IOx0~7=0 , as a Output.

IOx0~7=1 , as a Input.

Port B can be Wake Up (from High to Low) after Sleep if (/WUE)=0.



◆ **T1CON** (0x12C) : ***Timer 1 Control Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
T1CON	-	-	-	-	-	TR1S	TR1P1	TR1P0
R/W						R/W	R/W	R/W
RESET								

TR1S : ***Timer 1 Turn On Bit***

TR1S=0 , Timer 1 turn off.

TR1S=1 , Timer 1 turn on.

TR1P1,TR1P0 : ***Timer 1 Prescaler Rate***

TR1P1	TR1P0	TMR0 Rate
0	0	1:1
0	1	1:4
1	0	1:8
1	1	1:16

◆ **IOPH** (0x12D) : ***I/O Port Pull High Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
IOPH							/PUB	/PUA
R/W							R/W	R/W
RESET							1	1

/PUA~E : ***Each I/O Port Pull-High Enable Bit***

/PUA~B=0 , Enable (Connected with pull high Resistor).

/PUA~B=1 , Disable.

◆ **WDTS** (0x12E) : ***Watch Dog & Wake Up Control Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
WDTS	-	ODE	WDTE	SLP2E	-	-	-	/WUE
R/W	-	R/W	R/W	R/W	-	-	-	R/W
RESET								

ODE : ***Open Drain Control Bit for PC6 & PC7***

ODE=0 , PC6 & PC7 are Normal I/O Port.

ODE=1 , PC6 & PC7 are Open Drain Port.

WDTE : ***Watch Dog Timer Enable***

WDTE=0 , Disable the Watch Dog Timer.

WDTE=1 , Enable the Watch Dog Timer.

SLP2E : ***MCU Sleep Mode 2 Enable***

SLP2E=0 , Enter the MCU sleep mode 2.

SLP2E=1 , Set by MCU.

/WUE : ***I/O Port PIN Wake Up Enable***

/WUE=0 , Port B, Port C4,C5 & Port E can be wake –up
Wake Up Condition : As a Input Port & Input Signal from High to Low .

/WUE=1 , Disable PIN Wake Up Function.

I/O Port as a Wake Up PIN Setting Procedure:

Step 1: Setting as a Input PIN

Step 2: Pull High Enable .

Step 3: Enable Wake Up Function

Step 4: Enter to Sleep.

◆ **INTE** (0x12F) : ***Hardware Interrupt Control Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
INTE	-	-	-	-	TR1IE		EXIE	TR0IE
R/W					R/W		R/W	R/W
RESET					0		0	0

TR0IE : ***Timer 0 Interrupt Enable***

TR0IE=0 , Timer 0 Interrupt Disable .

TR0IE=1 , Timer 0 Interrupt Enable .

(Continue)

TR1IE : ***Timer 1 Interrupt Enable***

TR1IE=0 , Timer 1 Interrupt Disable .

TR1IE=1 , Timer 1 Interrupt Enable .

EXIE : ***External Interrupt Enable***

EXIE=0 , External Interrupt Disable .

EXIE=1 , External Interrupt Enable .

◆ **PWM1D** (0x130) : ***PWM1 Duty Cycle Data Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
PWM1D	P1D7	P1D6	P1D5	P1D4	P1D3	P1D2	P1D1	P1D0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W-	R/W-
RESET	0	0	0	0	0	0	0	0

P1D7~P1D0 : ***8 bits PWM1 Duty Cycle Data***

◆ **PWM1P** (0x132) : ***PWM1 Period Data Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
PWM1P	P1P7	P1P6	P1P5	P1P4	P1P3	P1P2	P1P1	P1P0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W-	R/W-
RESET	0	0	0	0	0	0	0	0

P1P7~P1P0 : ***8 bits PWM1 Period Data , PWM1 Signal Output from PE0 PIN.***

◆ **PWMC** (0x134) : ***PWM Control Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
PWMC	PW2E	PW1E	-	-	-	PWS2	PWS1	PWS1
R/W	R/W	R/W	-	-	-	R/W	R/W	R/W
RESET	0	0	-	-	-	0	0	0

PW1E : ***PW12 Enable Bit***

PW1E=0 , PWM1 Disable.

PW1E=1 , PWM1 Enable.

(Continue)

PWS2,PWS1,PWS0 : PWM Clock Prescaler *Rate*

PWS2	PWS1	PWS0	PWM Clock Rate
0	0	0	Fosc/2
0	0	1	Fosc/4
0	1	0	Fosc/8
0	1	1	Fosc/16
1	0	0	Fosc/32
1	0	1	Fosc/64
1	1	0	Fosc/128
1	1	1	Fosc/256

◆ **TMR1** (0x13E) : ***Timer 1 Value Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
TMR1	Timer 1 Value Register							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RESET								

◆ **TR1CV** (0x13F) : ***Timer 1 Comparator Value Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
TR1CV	Timer 1 Comparator Value Register							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RESET								

◆ **PAL** (0x14B) : ***Low Byte Address of The Data Latching***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
PAL	Data Latching Address Low Byte Part							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RESET								

◆ **PAH** (0x14C) : ***High Byte Address of The Data Latching***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
PAL	Data Latching Address High Byte Part							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

RESET								

◆ **INTF** (0x17F) : ***Interrupt Flag Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
INTF	-	-	-	-	TR1IF		EXIF	TR0IF
R/W					R/W		R/W	R/W
RESET					0		0	0

TR1IF : Timer 1 overflow flag.

TR1IF = 0 , Clear it by Software.

TR1IF = 1 , When Timer 1 overflow , this flag will be set to '1'.

SPIIF : SPI interrupt flag.

SPIIF = 0 , Clear it by Software.

SPIIF = 1 , Will be set when SPI interface interrupt occurred.

EXTIF : External interrupt flag.

EXTIF = 0 , Clear it by Software.

EXTIF = 1 , Will be set when the external interrupt occurred.

TR0IF : Timer 0 overflow flag.

TR0IF = 0 , Clear it by Software.

TR0IF = 1 , When Timer 0 overflow , this flag will be set to '1'.

3.0 Instruction Sets.

3.1 Overview

INSTRUCTION	DESCRIPTION	STATUS AFFECTED
NOP	No Operation	None
MCRW	A ---> CONT	None
MCRR	CONT ---> A	None
DAW	Decimal Adjust A	C
SLEEP	Go into Standby mode	T,P
WDTC	Clear Watchdog Timer	T,P
ENI	Enable Interrupt	None
DISI	Disable Interrupt	None
CLRW	Clear A	Z
CLR R	Clear R	Z
INT	PC+1 ---> Top Stack, Jump to 0002H	None
RETURN	Return from Subroutine	None
RETINT	Return from Interrupt	None
RETLW k	Return with literal in A	None
CALL k	Call Subroutine	None
JMP k	Jump to Address	None
MOV R,W	Move A to R	None
MOV W,R	Move R to A	Z
MOV R,R	Move R to R	Z
MOV W,k	Move literal to A	None
MOVC W,k	Move PAL, PAH ROM Data --> A	None
SWAPW R	Swap nibbles in R, ----> A	None
SWAP R	Swap nibbles in R, ----> R	None
BCR R,b	Bit Clear R	None
BSR R,b	Bit Set R	None
BTSC R,b	Bit Test R, Skip if Clear	None
BTSS R,b	Bit Test R, Skip if Set	None
RRCW R	Rotate Right R Through Carry ,----> A	C
RRC R	Rotate Right R Through Carry ,----> R	C
RLCW R	Rotate Left R Through Carry ,----> A	C

3.1 Overview (Continue)

INSTRUCTION	DESCRIPTION	STATUS AFFECTED
RLC R	Rotate Left R Through Carry ,----> R	C
AND W,R	A AND R ---> A	Z
AND R,W	A AND R ---> R	Z
AND W,k	A AND k ---> A	Z
OR W,R	A OR R ---> A	Z
OR R,W	A OR R ---> R	Z
OR W,k	A OR k ---> A	Z
XOR W,R	A XOR R ---> A	Z
XOR R,W	A XOR R ---> R	Z
XOR W,k	A XOR k ---> A	Z
INVW R	/R ---> A	Z
INV R	/R ---> R	Z
ADD W,R	A + R ---> A	Z,C,DC
ADD R,W	A + R ---> R	Z,C,DC
ADD W k	k+A ---> A	Z,C,DC
SUB W,R	R-A ---> A	Z,C,DC
SUB R,W	R-A ---> R	Z,C,DC
SUB W,k	k-A ---> A	Z,C,DC
INCW R	R+1 ---> A	Z
INC R	R+1 ---> R	Z
DECW R	R-1 ---> A	Z
DEC R	R-1 ---> R	Z
IRSZW R	R+1 ----> A, Skip if A=0	None
IRSZR R	R+1 ----> A, Skip if R=0	None
DRSZW R	R-1 ---> A, Skip if A=0	None
DRSZR R	R-1 ---> R, Skip if R=0	None

3.2 General Control Instructions

NOP	*No Operation*
Operation	NOP
Instruction Cycle	2
Affected Flag	None
Description	No any operation
Example	NOP

MCRW	*Writer W Register to MCR Register*
Operation	W --> MCR
Instruction Cycle	2
Affected Flag	None
Description	Copy W Register Data to MCR Register
Example	; before W = 10, MCR=15 MCRW ; after W=10, MCR=10

MCRR	*Writer MCR Register to W Register*
Operation	MCR --> W
Instruction Cycle	2
Affected Flag	None
Description	Copy MCR Register Data to W Register
Example	; before W = 10, MCR=15 MCRR ; after W=15, MCR=15

DAW	*Decimal Adjust*
Operation	If W[3:0]>9 or DC=1 Then W[3:0]+6 --> W[3:0] If W[7:4]>9 or C=1 Then W[7:4]+6 --> W[7:4]
Instruction Cycle	2
Affected Flag	C
Description	W Register Decimal Adjust & Place Result in W Register
Example	MOV W,@0x0E ; W=0x0EH DAW ; W=0x14H

SLEEP	*Sleep*
Operation	0--> WDT; Oscillator Stop
Instruction Cycle	2
Affected Flag	T,Z
Description	Clear Watch Dog Timer; then MCU Sleep
Example	SLEEP

WDTC	*Watch Dog Timer Clear*
Operation	0--> WDT;
Instruction Cycle	2
Affected Flag	T,P
Description	Clear Watch Dog Timer
Example	WDTC

ENI	*Enable Interrupt*
Operation	MCR/GIE Flag=0
Instruction Cycle	2
Affected Flag	None
Description	Enable Interrupt
Example	DISI

DISI	*Disable Interrupt*
Operation	MCR/GIE Flag=1
Instruction Cycle	2
Affected Flag	None
Description	Disable Interrupt
Example	DISI

CLR W	*Clear Working Register*
Operation	0 --> W
Instruction Cycle	2
Affected Flag	Z=1
Description	Clear Working Register Content --> W=0
Example	; before W=5AH CLR W ; after W=0H

CLR R	*Clear Register*
Operation	0 --> R
Instruction Cycle	2
Affected Flag	Z=1
Description	Clear Register Content --> R=0
Example	; before 0x10=5AH CLR 0x10 ; after 0x10=0H

INT	*Software Interrupt*
Operation	0001H --> PC, (PC+1) --> (Top Stack)
Instruction Cycle	4
Affected Flag	None
Description	Software Interrupt, go to 0001H Address(PC); Push the PC+1 to Top Stack
Example	ORG 0001H JMP SF_INT NOP SF_INT: NOP RETINT ; Feedback to LAB1 Main: NOP **INT** ; Software Interrupt ; go to 0001H Addr. LAB1 NOP ; ; Push LAB1 to Top Stack

3.3 Flow Control Instructions

RETURN	***Return from Subroutine.***
Operation	(Top Stack)--> PC,
Instruction Cycle	4
Affected Flag	None
Description	Return from Subroutine ; POP The Address from Top Stack
Example	SUB2: ; Subroutine 2. NOP NOP **RET** ; Return to LAB4 Addr. ; LAB3: NOP NOP CALL SUB2 ; go to SUB2 LAB4: NOP

RETINT	***Return from Interrupt.***
Operation	(Top Stack)--> PC, MCR/GIE Flag=1
Instruction Cycle	4
Affected Flag	None
Description	Return from Interrupt ; POP The Address from Top Stack
Example	ORG 0001H JMP SF_INT NOP SF_INT: NOP **RETINT** ; Feedback to LAB1 Main: NOP INT ; Software Interrupt ; go to 0001H Addr. LAB1 NOP ; ; Push LAB1 to Top Stack

RETLW	*Return with literal in W(Accumulator).*
Operation	(Top Stack)--> PC, MCR/GIE Flag=1
Instruction Cycle	4
Affected Flag	None
Description	Return from Interrupt ; POP The Address from Top Stack
Example	ORG 0001H JMP SF_INT NOP SF_INT: NOP **RETINT** ; Feedback to LAB1 Main: NOP INT ; Software Interrupt ; go to 0001H Addr. LAB1 NOP ; ; Push LAB1 to Top Stack

CALL k	*Call Subroutine*
Operation	K --> PC, (PC+1) --> (Top Stack)
Instruction Cycle	4
Affected Flag	None
Description	Call Subroutine, go to k Address(PC); Push the PC+1 to Top Stack
Example	**CALL** SUB1 ; go to SUB1 Addr. NOP NOP SUB1: ;

JMP k	*Jump to Assigned Addr.*
Operation	K --> PC,
Instruction Cycle	4
Affected Flag	None
Description	Go to k Address(PC);
Example	**JMP** LAB2 ; go to LAB2 Addr. NOP NOP LAB2: ;

3.4 Data Move Instructions

MOV R, W	*Write W Register to R Register*
Operation	W --> R
Instruction Cycle	2
Affected Flag	None
Description	Copy W Register Data to R Register
Example	NOP ; before W = 10, 0x20=0 MOV 0x20, W ; after W=10 , 0x20=10

MOV W, R	*Write R Register to W Register*
Operation	R --> W
Instruction Cycle	2
Affected Flag	Z
Description	Copy R Register Data to W Register
Example	NOP ; before W = 10, 0x25=15 MOV W, 0x25 ; after W=15 , 0x25=15

MOV R, R	*Writer R Register to itself*
Operation	R --> R
Instruction Cycle	2
Affected Flag	Z
Description	Copy R Register Data to R Register For check R Register is Zero or Not If Zero then Z flag=1

Example	; before 0x20=10
	..... ; 0x21=0
	MOV 0x20, 0x20 ; 0x20=10, Z=0
	MOV 0x21, 0x21 ; 0x21=0, Z=1

MOV W, k	*Writer a constant to W Register*
Operation	k --> W
Instruction Cycle	2
Affected Flag	None
Description	Copy W Register Data to R Register

Example	; before W = 10
	MOV W, @0x12 ; after W=12H
	MOV W, @01001010 ; W=4AH

MOVC	*Write ROM Data to W Register*
Operation	ROM Addr. [6:0]=PAL[7:1] ROM Addr. [11:7]=PAH[5:0] if PAL[0]='0'b then ROM[7:0]-->W if PAL[0]='1'b then ROM[15:8]-->W
Instruction Cycle	2
Affected Flag	None
Description	Read the ROM Data to W Register

Example	NOP ; Read Addr 100H ROM Data
	; ROM Addr.100H=0x1234
	
	MOV W, @0x00 ;
	MOV PAL, W
	MOV W, @0x02
	MOV PAH, W
	MOVC ;W=0x34 Read Low Byte
	MOV W, @0x01
	MOV PAL, W
	MOVC ;W=0x12 Read High Byte

SWAPW R	*Swap R; Place Result in W Register*
Operation	R[3:0] --> W[7:4], R[7:4] --> W[3:0]
Instruction Cycle	2
Affected Flag	None
Description	Swap R Register High Nibble and Low Nibble & Place Result in W Register
Example	; before W = 10, 0x10=5A SWAPW 0x10 ; after W=A5, 0x10=5A

SWAP R	*Swap R*
Operation	R[3:0] <--> R[7:4]
Instruction Cycle	2
Affected Flag	None
Description	Swap R Register High Nibble and Low Nibble & Place Result in R Register
Example	; before 0x10=5A SWAPW 0x10 ; after 0x10=A5

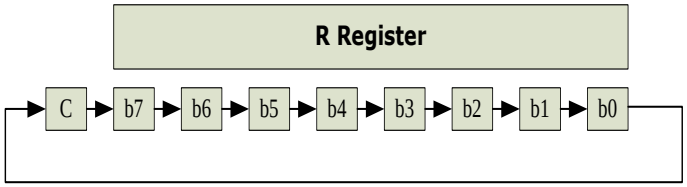
3.5 Bit Operate Instructions

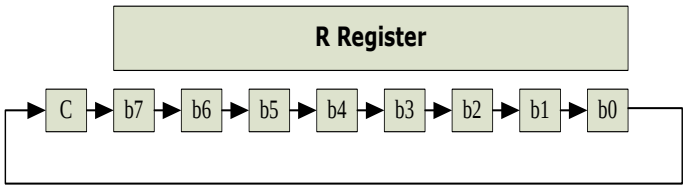
BCR R, b	*Clear Bit of Register*
Operation	0 --> R[b]
Instruction Cycle	2
Affected Flag	None
Description	Clear Assigned Bit of Register R[b]='0'b
Example	; before 0x10=5AH BCR 0x10,3 ; after 0x10=52H

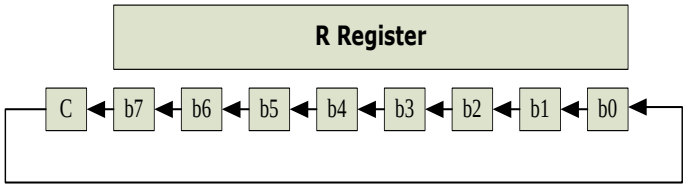
BSR R, b	*Set Bit of Register*
Operation	1 --> R[b]
Instruction Cycle	2
Affected Flag	None
Description	Set Assigned Bit of Register R[b]='1'b
Example	; before 0x10=5AH BSR 0x10,2 ; after 0x10=5EH

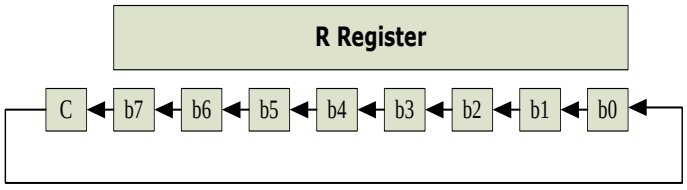
BTSC R, b	*Bit Test, Skip if Clear*
Operation	if R[b]=0 then Skip
Instruction Cycle	4
Affected Flag	None
Description	Bit Test; If this Bit is Clear, then Skip next instruction.
Example	; 0x20=01011010b ; 0x20 Bit2 ='0'b **BTSC** 0x20,2 ;Skip to NEXT2 NEXT1: JMP LAB1 NEXT2: JMP LAB2 LAB1: MOV W,@0AH LAB2: MOV W,@0BH ;

BTSS R, b	*Bit Test, Skip if Set*
Operation	if R[b]=1 then Skip
Instruction Cycle	4
Affected Flag	None
Description	Bit Test; If this Bit is Set, then Skip next instruction.
Example	; 0x20=01011010b ; 0x20 Bit3 ='1'b **BTSS** 0x20,1 ;Skip to NEXT2 NEXT1: JMP LAB1 NEXT2: JMP LAB2 LAB1: MOV W,@0AH LAB2: MOV W,@0BH ;

RRCW R	*Rotate Right R through Carry*
Operation	R[n]->W[n-1], R[0]->C, C--> W[7]
Instruction Cycle	2
Affected Flag	C
Description	 Place The Result in W Register
Example	; before W=0H , 0x10=5AH,C=1 RLCW 0x10 ; after W=ADH , 0x10=5AH,C=0

RRC R	*Rotate Right R through Carry*
Operation	R[n]->R[n-1], R[0]->C, C--> R[7]
Instruction Cycle	2
Affected Flag	C
Description	 Place The Result in R Register
Example	; before W=0H , 0x10=5AH,C=1 RLCW 0x10 ; after W=0H , 0x10=ADH,C=0

RLCW R	*Rotate Left R through Carry*
Operation	R[n]->W[n+1], R[7]->C, C--> W[0]
Instruction Cycle	2
Affected Flag	C
Description	 Place The Result in W Register
Example	; before W=0H , 0x10=5AH,C=1 RLCW 0x10 ; after W=B5H , 0x10=5AH,C=0

RLC R	*Rotate Left R through Carry*
Operation	R[n]->R[n+1], R[7]->C, C--> R[0]
Instruction Cycle	2
Affected Flag	C
Description	 Place The Result in R Register
Example	; before W=0H , 0x10=5AH,C=1 RLC 0x10 ; after W=0H , 0x10=B5H,C=0

3.6 Boolean Operate Instructions

AND W, R	*And logical operation*
Operation	W & R --> W
Instruction Cycle	2
Affected Flag	Z
Description	W Register & R Register & Place Result in W Register
Example	; before 0x10=5AH, W=0FH AND W,0x10 ; after 0x10=5AH, W=0AH

AND R, W	*And logical operation*
Operation	W and R --> R
Instruction Cycle	2
Affected Flag	Z
Description	W Register and R Register & Place Result in R Register
Example	; before 0x10=5AH, W=0FH AND 0x10, W ; after 0x10=0AH, W=0FH

AND W, k	*And logical operation*
Operation	W and k --> W
Instruction Cycle	2
Affected Flag	Z
Description	W Register and Constant k & Place Result in W Register
Example	; before W=A5H AND W, @0x0F ; after W=05H

OR W, R	*Inclusive OR logical operation*
Operation	W or R --> W
Instruction Cycle	2
Affected Flag	Z
Description	W Register or R Register & Place Result in W Register
Example	; before 0x10=5AH, W=0FH OR W,0x10 ; after 0x10=5AH, W=5FH

OR R, W	*Inclusive OR logical operation*
Operation	W or R --> R
Instruction Cycle	2
Affected Flag	Z
Description	W Register or R Register & Place Result in R Register
Example	; before 0x10=5AH, W=0FH OR 0x10, W ; after 0x10=5FH, W=0FH

OR W, k	*Inclusive OR logical operation*
Operation	W or k --> W
Instruction Cycle	2
Affected Flag	Z
Description	W Register or Constant k & Place Result in W Register
Example	; before W=A5H AND W, @0x0F ; after W=AFH

XOR W, R	*Exclusive OR logical operation*
Operation	W xor R --> W
Instruction Cycle	2
Affected Flag	Z
Description	W Register or R Register & Place Result in W Register
Example	; before 0x10=5AH, W=0FH XOR W,0x10 ; after 0x10=5AH, W=55H

XOR R, W	*Exclusive OR logical operation*
Operation	W xor R --> R
Instruction Cycle	2
Affected Flag	Z
Description	W Register xor R Register & Place Result in R Register
Example	; before 0x10=5AH, W=0FH XOR W,0x10 ; after 0x10=55H, W=0FH

XOR W, k	*Exclusive OR logical operation*
Operation	W xor k --> W
Instruction Cycle	2
Affected Flag	Z
Description	W Register xor Constant k & Place Result in W Register
Example	; before W=A5H AND W, @0x0F ; after W=AAH

3.7 Arithmetic Operate Instructions

INVW R	*Inverse Working Register*
Operation	/R --> W
Instruction Cycle	2
Affected Flag	Z
Description	Inverse Working Register Content W=/R
Example	; before W=0H ,0x10=5AH INVW 0x10 ; after W=A5H,0x10=5AH

INV R	*Inverse R Register*
Operation	/R --> R
Instruction Cycle	2
Affected Flag	Z
Description	Inverse Working Register Content W=/R
Example	; before W=0H , 0x10=5AH INV 0x10 ; after W=0H , 0x10=A5H

ADD W, R	*Add*
Operation	W+R --> W
Instruction Cycle	2
Affected Flag	Z,C,DC Z=1 ; when the Result =0 C=1 ; when the bit7 is Overflow DC=1; When the bit3 is Overflow
Description	W Register add R Register & Place Result in W Register

Example	; For R10 + R11--> W
	MOV W, 0x10 ; W=R10
	ADD W, 0x11 ; W=R10 +R11

ADD R, W	*Add*
Operation	W+R --> R
Instruction Cycle	2
Affected Flag	Z,C,DC Z=1 ; when the Result =0 C=1 ; when the bit7 is Overflow DC=1; When the bit3 is Overflow
Description	W Register add R Register & Place Result in R Register

Example	; For R10 + R11--> R11
	MOV W, 0x10 ; W=R10
	ADD 0x11, W ; R11=R10 +R11

ADD W, k	*Add*
Operation	W+k --> W
Instruction Cycle	2
Affected Flag	Z,C,DC Z=1 ; when the Result =0 C=1 ; when the bit7 is Overflow DC=1; When the bit3 is Overflow
Description	W Register add a constant & Place Result in W Register

Example	; For R10 + 15H --> W
	MOV W, 0x10 ; W=R10
	ADD W, @0x15 ; W=R10 +15H

SUB W, R	*Subtract*
Operation	R-W --> W
Instruction Cycle	2
Affected Flag	Z,C,DC Z=1 ; when the Result =0 ***C=0 ; when the bit7 is Overflow*** ***DC=0; When the bit3 is Overflow***
Description	R Register subtract W Register & Place Result in W Register

Example	; For R11- R10 --> W
	MOV W, 0x10 ; W=R10
	SUB W, 0x11 ; W=R11 – R10

SUB R, W	*Subtract*
Operation	R-W --> R
Instruction Cycle	2
Affected Flag	Z,C,DC Z=1 ; when the Result =0 ***C=0 ; when the bit7 is Overflow*** ***DC=0; When the bit3 is Overflow***
Description	R Register subtract W Register & Place Result in R Register

Example	; For R11- R10--> R11
	MOV W, 0x10 ; W=R10
	SUB 0x11, W ; R11=R11 -R10

SUB W, k	*Subtract*
Operation	k-W --> W
Instruction Cycle	2
Affected Flag	Z,C,DC Z=1 ; when the Result =0 ***C=0 ; when the bit7 is Overflow*** ***DC=0; When the bit3 is Overflow***
Description	A Constant subtract W Register & Place Result in W Register

Example	; For 15H – R10 --> W
	MOV W, 0x10 ; W=R10
	ADD W, @0x15 ; W=15H – R10

INCW R	*Increment R*
Operation	R+1 --> W
Instruction Cycle	2
Affected Flag	Z
Description	R Register Content add 1 & Place Result in W Register
Example	; before 0x10=5AH, W=0 INCW 0x10 ; after 0x10=5AH, W=5BH

INC R	*Increment R*
Operation	R+1 --> R
Instruction Cycle	2
Affected Flag	Z
Description	R Register Content add 1 & Place Result in R Register
Example	; before 0x10=5AH INC 0x10 ; after 0x10=5BH

DECW R	*Decrement R*
Operation	R-1 --> W
Instruction Cycle	2
Affected Flag	Z
Description	R Register Content subtract1 & Place Result in W Register
Example	; before 0x10=5AH, W=0 DECW 0x10 ; after 0x10=5AH, W=59H

DEC R	*Decrement R*
Operation	R-1 --> R
Instruction Cycle	2
Affected Flag	Z
Description	R Register Content subtract 1 & Place Result in R Register
Example	; before 0x10=5AH DEC 0x10 ; after 0x10=59H

IRSZW R	*Increment R, Skip if Zero.*
Operation	(R+1)-->W , Skip if result = 0
Instruction Cycle	4
Affected Flag	None
Description	Decrement R & Place the Result in W; If (R+1)=0 then Skip next instruction.
Example	; before 0x20=FFH **IRSZW** 0x20 ;W=0H,Skip NEXT2 NEXT1: JMP LAB1 NEXT2: JMP LAB2 LAB1: MOV W,@0AH LAB2: MOV W,@0BH ;

IRSZR R	*Increment R, Skip if Zero.*
Operation	(R+1)-->R , Skip if result = 0
Instruction Cycle	4
Affected Flag	None
Description	Increment R & Place the Result in R; If (R+1)=0 then Skip next instruction.
Example	; before 0x20=FFH **IRSZR** 0x20 ;0x20=0H,Skip NEXT2 NEXT1: JMP LAB1 NEXT2: JMP LAB2 LAB1: MOV W,@0AH LAB2: MOV W,@0BH ;

DRSZW R	*Decrement R, Skip if Zero.*
Operation	(R-1)-->W , Skip if result = 0
Instruction Cycle	4
Affected Flag	None
Description	Decrement R & Place the Result in W; If (R-1)=0 then Skip next instruction.
Example	; before 0x20=01H **DRSZW** 0x20 ;W=0H,Skip NEXT2 NEXT1: JMP LAB1 NEXT2: JMP LAB2 LAB1: MOV W,@0AH LAB2: MOV W,@0BH ;

DRSZR R	***Decrement R, Skip if Zero.***
Operation	(R-1)-->R , Skip if result = 0
Instruction Cycle	4
Affected Flag	None
Description	Decrement R & Place the Result in R; If (R-1)=0 then Skip next instruction.
Example	; before 0x20=01H **DRSZR** 0x20 ;0x20=0H,Skip NEXT2 NEXT1: JMP LAB1 NEXT2: JMP LAB2 LAB1: MOV W,@0AH LAB2: MOV W,@0BH ;

4.0 Device Configuration Options.

4.1 Protection

YSM142 have protection option to protect the code in MCU OTP ROM. If the protection option had been armed, there is no way to read back the data from OTP ROM.

4.2 Warm-up time

While the MCU is booting up, it will delay for specific time to wait for system stable. User can set up the time according to the slowest device in whole system to prevent timing issue error.

4.3 Power Save Mode(Eco-Mode)

YSM142 can reduce the power consumption by setting up this option. But limitation are existed for specific Eco-Mode level.

The system frequency should lower than 12MHz for Eco-Mode level-1, and lower than 500KHz for Eco-Mode level-2.

4.4 LDO (Low Drop Regulator)

YSM142 had an LDO module which can made the chip core operate in more wide range of the voltage. But the power consumption will little higher if the LDO module was being enabled.

4.5 System clock source selection

YSM142 had 6 types of the system clock source:

1. XTH – High speed external crystal (Frequency > 1MHz)
2. XTL – Low Speed external crystal (Frequency < 1MHz)
3. IRC w/o OSCO – Internal RC mode, no any outgoing clocks signal.
4. IRC w OSCO – Internal RC mode with the clock out on OSCO.
5. Ext.RC w/o OSCO – External RC mode, no any outgoing clocks.
6. Ext.RC w OSCO – External RC mode with the clock out on OSCO.

Notes. OSCO frequency will be half from system frequency.

Ex. If Fosc=12MHz, the frequency from OSCO will be 6MHz.

4.6 Low Voltage Reset

YSM142 had an Low Voltage Reset module, which can self reset the MCU if the operating voltage is lower than specific voltage level. User can specific the reference voltage and the continuous time period. If the voltage was below the reference voltage and keep for an period of the user specific, MCU will reset to avoid the low voltage issue error.

4.7 General MCU Functions

There are the following general MCU function can be setting in device configuration option.

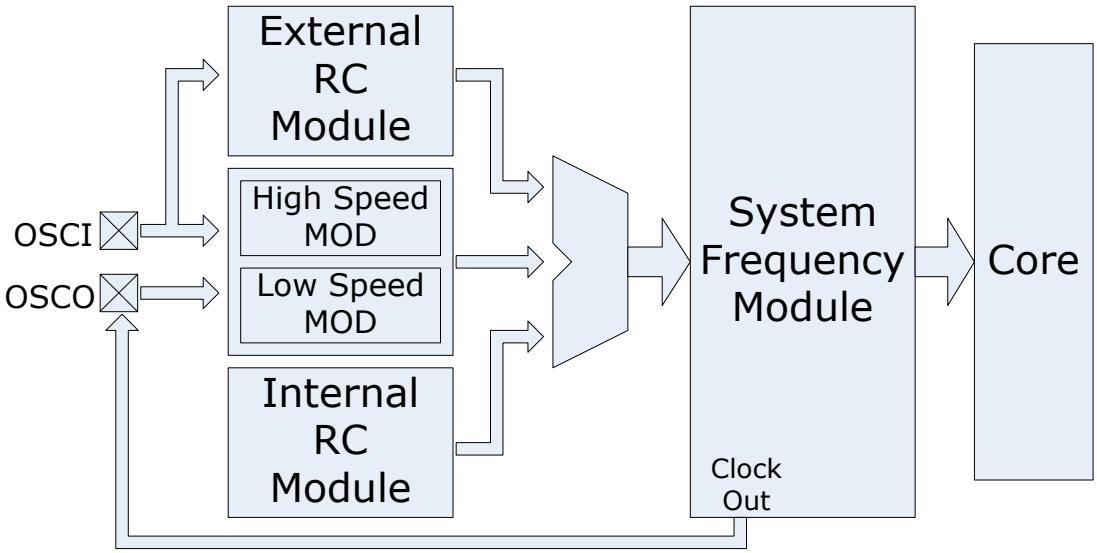
1. Watch Dog Timer : There is an internal RC clocked timer for watch dog function.

2. I/O Drive Enhance : I/O ports can enhance the driving ability by setting up this option, but the power consumption will rise higher.

3. Auto Pull Setting : YSM142 have internal Pull up/down resistor for I/O ports. But if the output ports with internal pull up/down resistor connected, it will have the current leaking. If the Auto Pull Setting had been set, MCU will disconnect the pull up/down resistor while the ports being set to output to avoid the current leaking.

5.0 System Clock.

5.1 System clock overview



5.2 Clock source type

YSM142 have 3 different types of the clock source:

1. External crystal
2. Internal RC
3. External RC

5.3 External Crystal

User can use the standard 2 pins crystal(XTAL) with the internal clocking circuit to generate the precise frequency. The frequency can reach up to 16MHz.

5.4 Internal RC

YSM142 had build in an RC frequency generator on chip. User can select this module to generate an frequency clock signal for core use. It's no need to add any additional device to generate the clock signal. User can setting up to export the generated clock signal out from OSCO pin or not by device configuration.

5.5 External RC

Except to the chip build-in RC frequency clock generator, user can connect to an RC circuit outside the chip from OSCI pin instead of the internal RC circuit. In this mode, user can adjust the external RC circuit to change the clock frequency. It's also can export the generated clock signal out from the OSCO pin(See Sector 4.6).

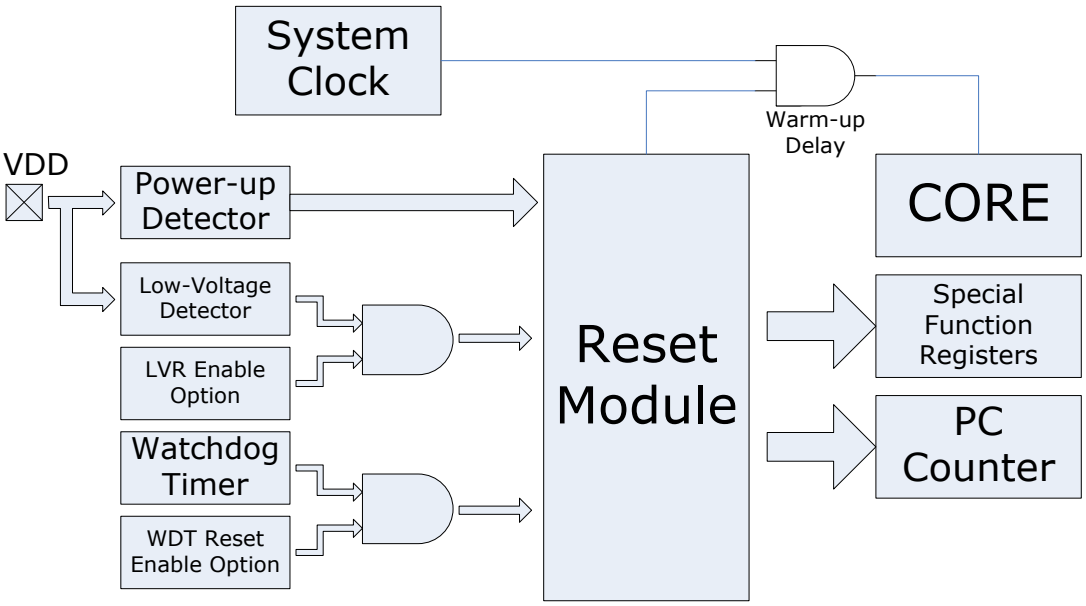
5.6 Clock signal export

YSM145 have an clock export module which can export an half frequency clock signal from the OSCO pin. This function was defined on the device configuration option.

Ex. If the system clock frequency = 12MHz, Clock signal export from OSCO will be 6MHz clock signal.

6.0 Reset Module.

6.1 Overview



6.2 Power-On Reset

While the MCU had been power up, the MCU reset module will delay with the specific time period and reset the special function registers at the same time. The PC counter will also reset to address 0xFFF while the power on reset. User can adjust the power up delay period (warm-up time) to wait the slower device in the designed system to avoid the timing issue error.

6.3 Low Voltage Reset (L.V.R)

M142 had build-in an low voltage detector. If the operating voltage (VDD)'s level is keep below the specific reference voltage level for an specific period, low voltage detector will send an reset signal to the reset module to reset the MCU. This function is used to avoid the operating error from the low voltage issue. If the LVR reset had been executed, the PC counter will be reset to 0xFFF.

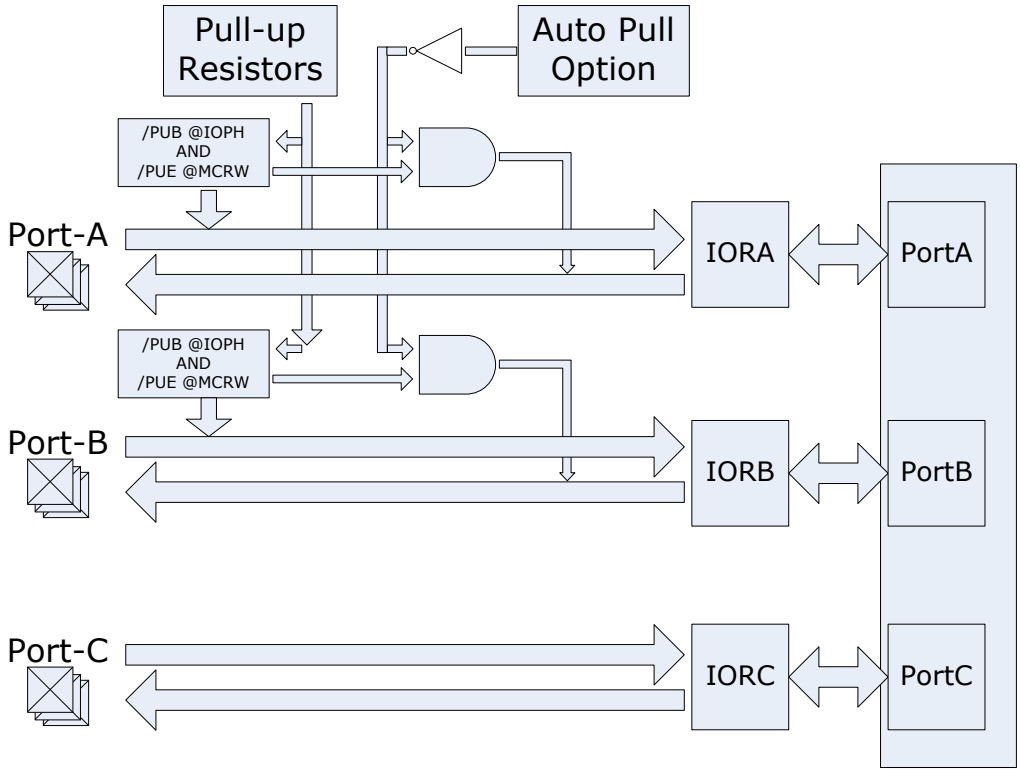
6.4 Watchdog Reset (WDT Reset)

YSM142 had a watchdog timer. If the watchdog timer had been enable and the watchdog timer overflow flag was being set, it will send an reset signal to reset module to reset the MCU. The PC counter will be reset to 0xFFF. The watchdog timer reset is used to avoid the MCU hang on crash or infinite looping status.

Notes: User should clear up the watch dog timer regularly to avoid the WDT reset.

7.0 I/O Ports Module.

7.1 Overview



7.2 Ports

M142 have 3 I/O ports, which are Port-A, Port-B and Port-C. Each port can be set to input or output. Except Port-C is 5bits wide, all the other ports are 8bits wide. All ports can connect with the internal pull resistors.

There are 3 general power relative ports(VDD / VDDL / VSS)

1 External reset port are available, low active.

1 External interrupt pin with internal pull up.

2 frequency ports (OSCI / OSCO)

1 Analog input port(Comparator port)

7.3 I/O Ports Direction Control

There are 3 ports in YSM142. Each one can be set to input or output by using the IORx special function register to control the port direction.

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
IORA	IOA7	IOA6	IOA5	IOA4	IOA3	IOA2	IOA1	IOA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RESET	1	1	1	1	1	1	1	1
IORB	IOB7	IOB6	IOB5	IOB4	IOB3	IOB2	IOB1	IOB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RESET	1	1	1	1	1	1	1	1
IORC	-	-		IOC4	IOC3	IOC2	IOC1	IOC0
R/W	-	-		R/W	R/W	R/W	R/W	R/W
RESET	-	-		1	1	1	1	1

IOx0~7=0 , as a Output. / IOx0~7=1 , as a Input.

7.4 I/O Ports Value Read/Write

User can read or output the port data via the PORTx special function registers.

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
PORTA	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RESET	0	0	0	0	0	0	0	0
PORTB	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RESET	0	0	0	0	0	0	0	0
PORTC				PC4	PC3	PC2	PC1	PC0
R/W				R/W	R/W	R/W	R/W	R/W
RESET				0	0	0	0	0

7.5 I/O Ports Pull resistors

◆ **MCRW** (0x122) : ***Main Control Register for Write***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MCRR	/PUE	GIE	-	-	PAB	PS2	PS1	PS0
R/W	R/W	R/W			R/W	R/W	R/W	R/W
RESET								

/PUE : ***I/O Port Pull-Up Resistor Enable*** .

/PUE=0 : I/O Connected with Pull-Up Resistor.

/PUE=1 : Pull-Resistor Disconnected.

◆ **IOPH** (0x12D) : ***I/O Port Pull High Register***

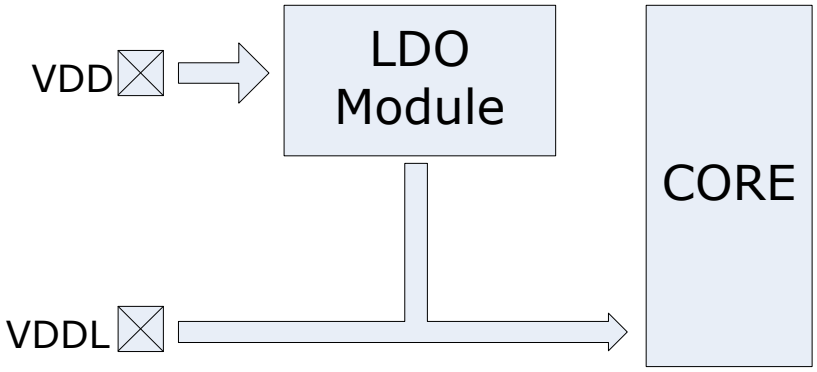
NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
IOPH	-	-	-	-	-	-	/PUB	/PUA
R/W	-	-	-	-	-	-	R/W	R/W
RESET								

/PUA~E : ***Each I/O Port Pull-High Enable Bit***

/PUA~E=0 , Enable (Connected with pull high Resistor).

/PUA~E=1 , Disable.

7.6 Power Relative Pins



VDD and VDDL should connect to an 0.1uF capacitor each.

7.7 External Interrupt Pin

User can trig the external interrupt by set low to the external interrupt pin with internal pull up.

8.0 Chip Status.

8.1 Overview

The status flags are being build in the YSM145's special function registers. User can know the status of the MCU by monitor the register.

8.2 Status Register

◆ **STATUS** (0x103) : ***Status Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
STATUS	GPR	-	-	/TO	/PD	Z	DC	C
R/W	R/W			R/W	R/W	R/W	R/W	R/W
RESET								

GPR : ***General Purpose Read/Write bit***

/TO : ***Time- Out Flag***

/TO=0 Watch dog Timer Overflow.

/TO=1 (a) Power On

(b) Execute WDTC or SLEEP Instruction.

/PD : ***Power Down Flag***

/PD=0 After SLEEP Instruction.

/PD=1 (a) Power On

(b) Execute WDTC Instruction.

Z : ***ALU Operation Zero Flag***

Z=0 Operation Result is not Zero.

Z=1 Operation Result is Zero.

DC : ***ALU Operation Half Carry /Borrow Flag***

DC=0 Half Carry/Borrow does not occur.

DC=1 Half Carry/Borrow occurred.

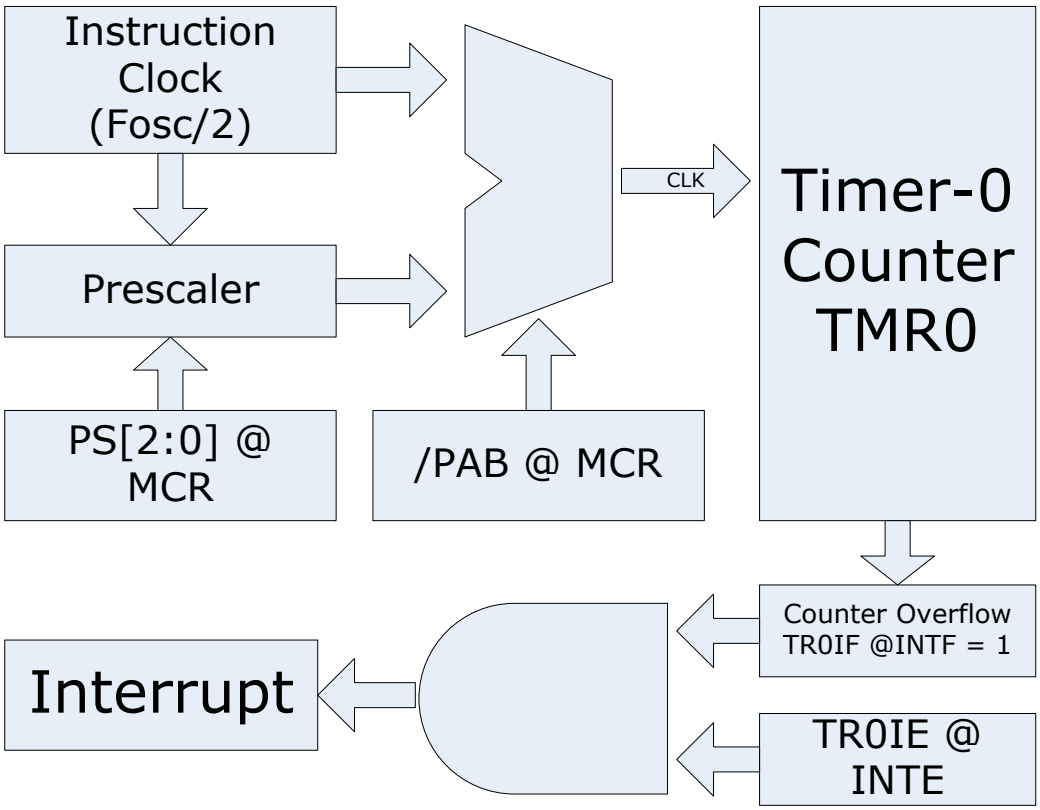
C : ***ALU Operation Carry /Borrow Flag***

C=0 Carry/Borrow does not occur.

C=1 Carry/Borrow occurred.

9.0 Timer0.

9.1 Overview



9.2 Registers

◆ **MCRW** (0x122) : ***Main Control Register for Write***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MCRR	/PUE	GIE	-	-	PAB	PS2	PS1	PS0
R/W	R/W	R/W			R/W	R/W	R/W	R/W
RESET	1	0			1	1	1	1

PAB : ***Prescaler bit Assignment .***

PAB=0 Assign to TMR0 (Timer 0).

PAB=1 Assign to WDT (Watch-Dog Timer).

PS2~PS0 : ***Prescaler bits.***

PS2	PS1	PS0	TMR0 Rate	WDT Rate
0	0	0	1:2	1:1
0	0	1	1:4	1:2
0	1	0	1:8	1:4
0	1	1	1:16	1:8
1	0	0	1:32	1:16
1	0	1	1:64	1:32
1	1	0	1:128	1:64
1	1	1	1:256	1:128

When The TMR0 from 0xff increasing to 0x00 , TR0IF will be set to '1', and if the TR0IE=1 , then the Program Counter will go to Address 0x001(Hardware Interrupt Address).

◆ **TMR0** (0x101) : ***8Bits Real Timer Clock/Counter***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
TMR0	8 bit Real Time Clock / Counter							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RESET	0	0	0	0	0	0	0	0

TMR0 : Increased By Instruction Cycle.

◆ **INTE** (0x12F) : ***Hardware Interrupt Control Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
INTE	-	-	-	-	TR1IE	-	EXIE	TR0IE
R/W	-	-	-	-	R/W	-	R/W	R/W
RESET	-	-	-	-	0	-	0	0

TR0IE : *Timer 0 Interrupt Enable*

TR0IE=0 , Timer 0 Interrupt Disable .

TR0IE=1 , Timer 0 Interrupt Enable .

◆ **INTF** (0x17F) : ***Interrupt Flag Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
INTF	-	-	-	-	TR1IF	-	EXIF	TR0IF
R/W	-	-	-	-	R/W	-	R/W	R/W
RESET	-	-	-	-	0	-	0	0

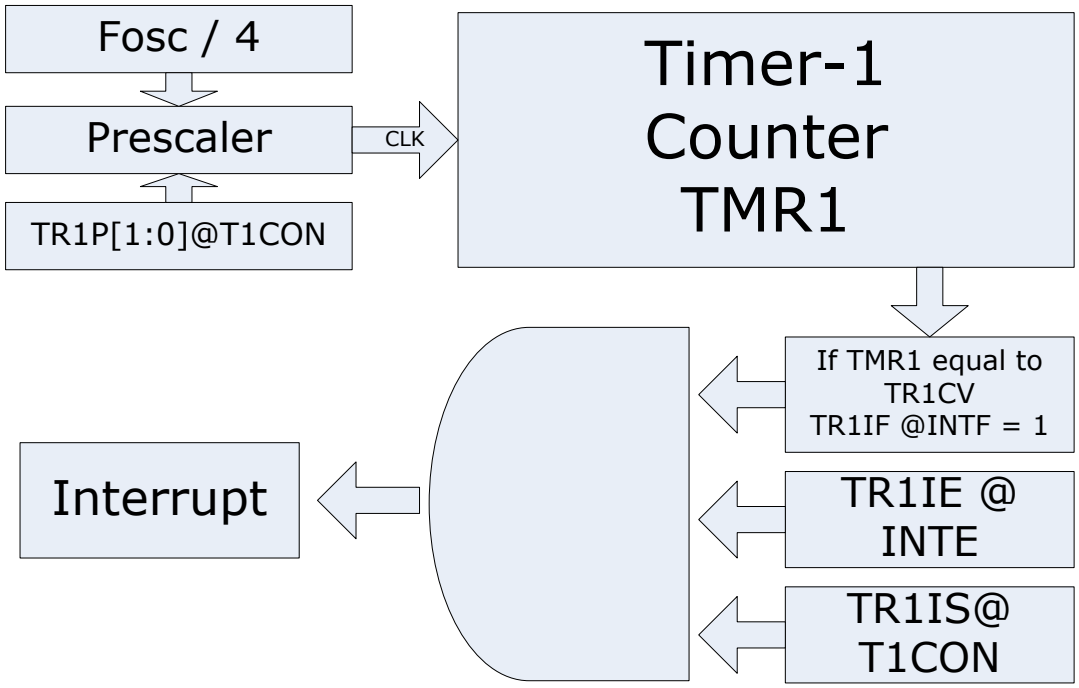
TR0IF : Timer 0 overflow flag.

TR0IF = 0 , Clear it by Software.

TR0IF = 1 , When Timer 0 overflow , this flag will be set to '1'.

10.0 Timer1.

10.1 Overview



10.2 Registers

Timer1 is an 8 bits timer. When set the TR1IE=1, TR1S=1 , the Timer1 will count from 0x00(TMR1).

If the Register TMR1 equal to the register TR1CV, the TR1IF will be set to 1 and the program counter will go to Address 0x001(Hardware Interrupt Address) .

◆ **INTE** (0x12F) : ***Hardware Interrupt Control Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
INTE	-	-	-	-	TR1IE	-	EXIE	TR0IE
R/W	-	-	-	-	R/W	-	R/W	R/W
RESET	-	-	-	-	0	-	0	0

TR1IE : *Timer 1 Interrupt Enable*

TR1IE=0 , Timer 1 Interrupt Disable .

TR1IE=1 , Timer 1 Interrupt Enable .

◆ **INTF** (0x17F) : ***Interrupt Flag Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
INTF	-	-	-	-	TR1IF	-	EXIF	TR0IF
R/W	-	-	-	-	R/W	-	R/W	R/W
RESET	-	-	-	-	0	-	0	0

TR1IF : Timer 1 overflow flag.

TR1IF = 0 , Clear it by Software.

TR1IF = 1 , When Timer 1 counter same to TR1CV , this flag will be set to '1'.

◆ **TMR1** (0x13E) : ***Timer 1 Value Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
TMR1	Timer 1 Value Register							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RESET								

◆ **TR1CV** (0x13F) : ***Timer 1 Comparator Value Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
TR1CV	Timer 1 Comparator Value Register							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RESET								

◆ **T1CON** (0x12C) : ***Timer 1 Control Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
T1CON	-	-	-	-	-	TR1S	TR1P1	TR1P0
R/W						R/W	R/W	R/W
RESET								

TR1S : ***Timer 1 Turn On Bit***

TR1S=0 , Timer 1 turn off.

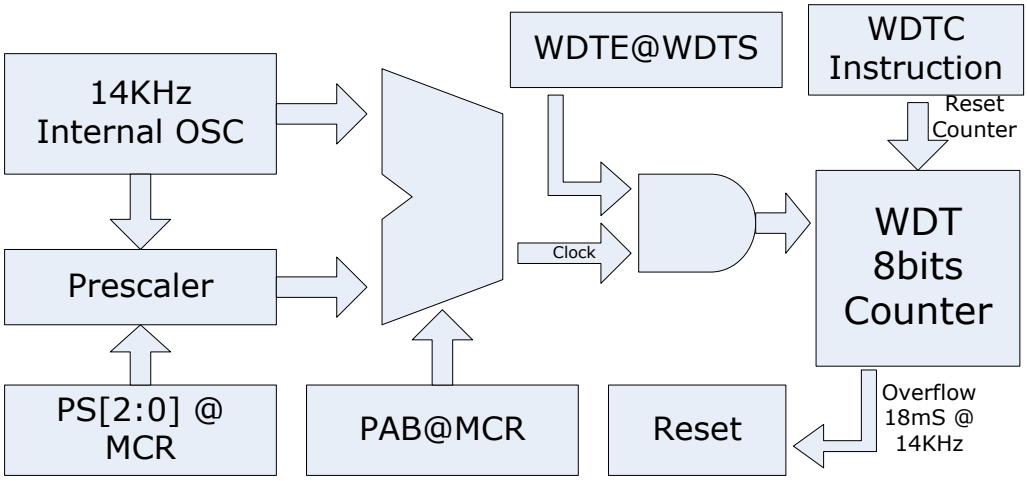
TR1S=1 , Timer 1 turn on.

TR1P1,TR1P0 : ***Timer 1 Prescaler Rate***

TR1P1	TR1P0	TMR0 Rate
0	0	1:1
0	1	1:4
1	0	1:8
1	1	1:16

11.0 Watchdog Timer.

11.1 Overview



11.2 Registers

◆ **MCRW** (0x122) : ***Main Control Register for Write***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MCRW	/PUE	GIE	-	-	PAB	PS2	PS1	PS0
R/W	R/W	R/W			R/W	R/W	R/W	R/W
RESET	1	0			1	1	1	1

PAB : ***Prescaler bit Assignment .***

PAB=0 Assign to TMR0 (Timer 0).

PAB=1 Assign to WDT (Watch-Dog Timer).

PS2~PS0 : ***Prescaler bits.***

PS2	PS1	PS0	TMR0 Rate	WDT Rate
0	0	0	1:2	1:1
0	0	1	1:4	1:2
0	1	0	1:8	1:4
0	1	1	1:16	1:8
1	0	0	1:32	1:16
1	0	1	1:64	1:32
1	1	0	1:128	1:64
1	1	1	1:256	1:128

◆ **WDTS** (0x12E) : ***Watch Dog & Wake Up Control Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
WDTS	-	ODE	WDTE	SLP2E	-	-	-	/WUE
R/W	-	R/W	R/W	R/W	-	-	-	R/W
RESET								

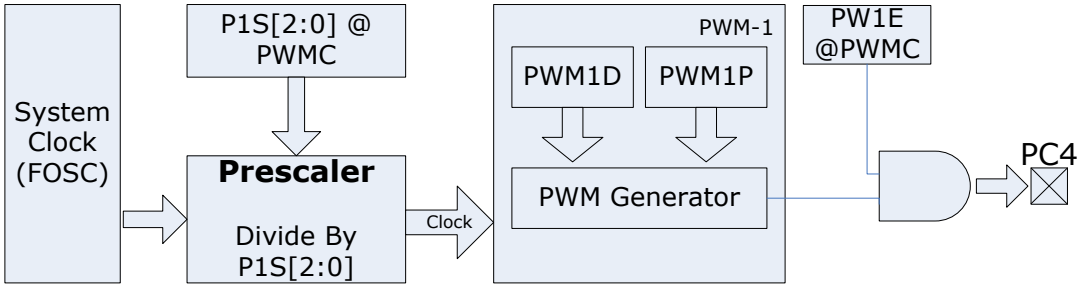
/WDTE : Watchdog timer Enable

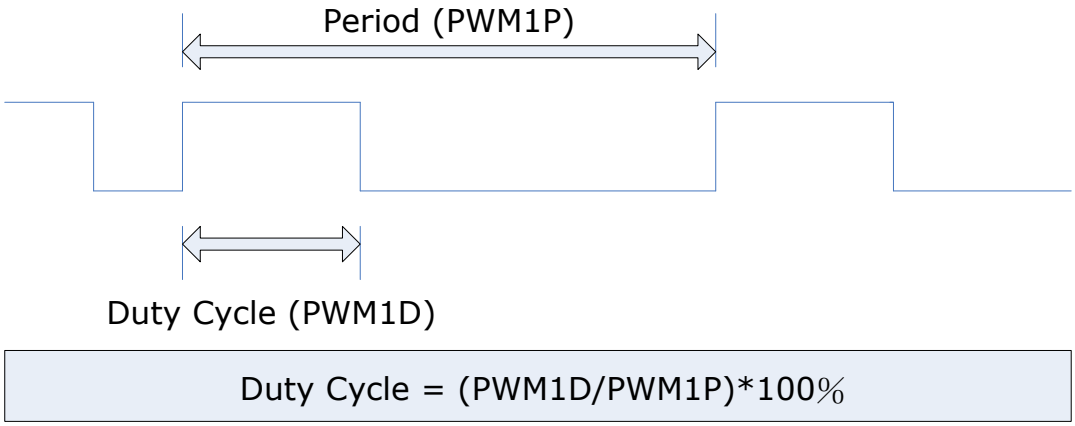
/WDTE=0 , Watchdog timer disable .

/WDTE=1 , Watchdog timer enable.

12.0 PWM – Pulse Width Modulator.

12.1 Overview





12.2 Registers

◆ **PWMC** (0x134) : ***PWM Control Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
DACR		PW1E	-	-	-	PWS2	PWS1	PWS0
R/W		R/W	-	-	-	R/W	R/W	R/W
RESET		0	-	-	-	0	0	0

PW1E : *PWM1 Enable Bit*

PW1E=0 , PWM1 Disable.

PW1E=1 , PWM1 Enable.

PWS2,PWS1,PWS0 : *PWM Clock Prescaler Rate*

PWS2	PWS1	PWS0	PWM Clock Rate
0	0	0	Fosc/2
0	0	1	Fosc/4
0	1	0	Fosc/8
0	1	1	Fosc/16
1	0	0	Fosc/32
1	0	1	Fosc/64
1	1	0	Fosc/128
1	1	1	Fosc/256

◆ **PWM1D** (0x130) : ***PWM1 Duty Cycle Data Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
PWM1D	P1D7	P1D6	P1D5	P1D4	P1D3	P1D2	P1D1	P1D0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W-	R/W-
RESET	0	0	0	0	0	0	0	0

P1D7~P1D0 : *8 bits PWM1 Duty Cycle Data*

◆ **PWM1P** (0x132) : ***PWM1 Period Data Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
PWM1P	P1P7	P1P6	P1P5	P1P4	P1P3	P1P2	P1P1	P1P0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W-	R/W-
RESET	0	0	0	0	0	0	0	0

P1P7~P1P0 : ***8 bits PWM1 Period Data***

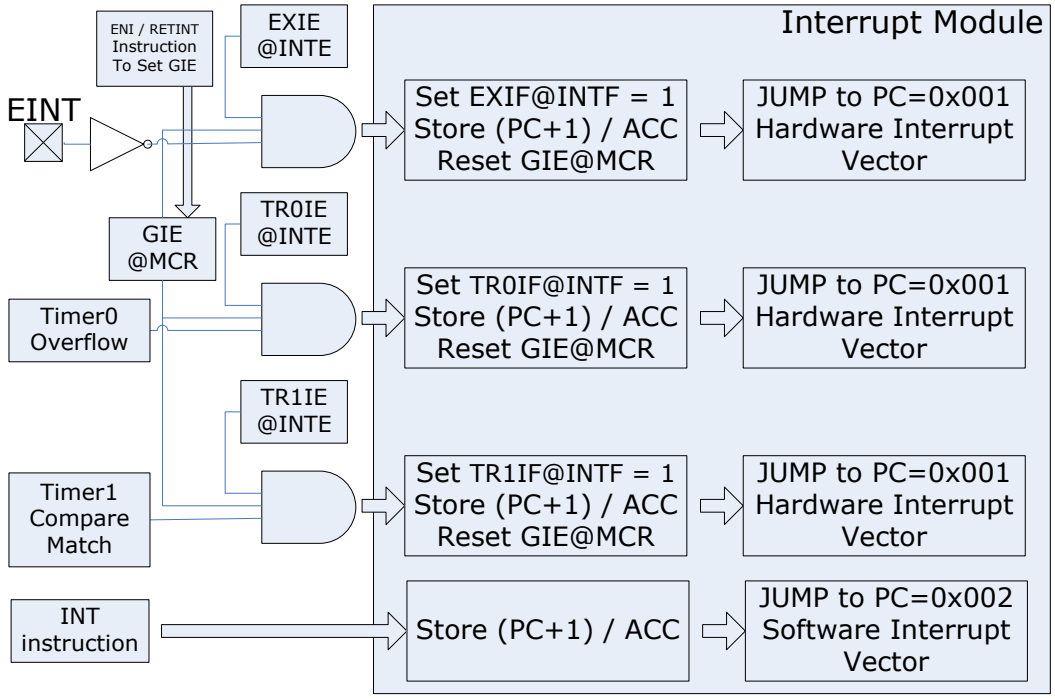
Note: PWM1 Signal Output from PC4 PIN

15.3 Code Setting Flow

```
MOV A,      @0X7F     ;Setup the PWM duty cycle register
MOV PWM1D,  A
MOV A,      @0XFF     ;Setup the PWM period data register
MOV PWM1P,  A
MOV A,      @0x02
MOV PWMC,   A         ; Set Prescaler to Fosc/8
BSR PWMC,   6         ; Active the PWM-1 module
;Now the PWM-1 start to output the PWM signal.
```

13.0 Interrupts.

13.1 Overview



13.2 Registers

- **MCRW** (0x122) : ***Main Control Register for Write***
- **MCRR** (0x124) : ***Main Control Register for Read***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MCRR	/PUE	GIE	-	-	PAB	PS2	PS1	PS0
R/W	R/W	R/W			R/W	R/W	R/W	R/W
RESET								

(Continue)

GIE : ***Global Interrupt Enable.***

GIE=0 Disable all interrupts. Execute DISI Instruction or accept Interrupt then GIE will be set to '0'. Reject any other Interrupt.

GIE=1 Global Enable all the Interrupt (Enable or not depend on others Register Enable Bit.
Execute ENI or RETI will be set to '1'. then accept others Interrupt.

◆ **INTE** (0x12F) : ***Hardware Interrupt Control Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
INTE	-	-	-	-	TR1IE		EXIE	TR0IE
R/W	-	-	-	-	R/W		R/W	R/W
RESET	-	-	-	-	0		0	0

TR1IE : *Timer 1 Interrupt Enable*

TR1IE=0 , Timer 1 Interrupt Disable .

TR1IE=1 , Timer 1 Interrupt Enable .

TR0IE : *Timer 0 Interrupt Enable*

TR0IE=0 , Timer 0 Interrupt Disable .

TR0IE=1 , Timer 0 Interrupt Enable .

EXIE : *External Interrupt Enable*

EXIE=0 , External Interrupt Disable .

EXIE=1 , External Interrupt Enable .

◆ **INTF** (0x17F) : ***Interrupt Flag Register***

NAME	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
INTF	-	-	-	-	TR1IF		EXIF	TR0IF
R/W	-	-	-	-	R/W		R/W	R/W
RESET	-	-	-	-	0		0	0

TR1IF : ***Timer 1 Overflow Flag***

TR1IF=0 , Clear it by Software.

TR1IF=1 , If the TMR1 is equal to TR1CV.

TR0IF : ***Timer 0 Overflow Flag***

TR0IF=0 , Clear it by Software.

TR0IF=1 , If the TMR0 is overflow .

EXIE : ***External Interrupt Enable***

EXIE=0 , External Interrupt Disable .

EXIE=1 , External Interrupt Enable .

13.3 Instructions

ENI	*Enable Interrupt*
Operation	MCR/GIE Flag=0
Instruction Cycle	2
Affected Flag	None
Description	Enable Interrupt
Example	DISI

DISI	*Disable Interrupt*
Operation	MCR/GIE Flag=1
Instruction Cycle	2
Affected Flag	None
Description	Disable Interrupt
Example	DISI

INT	***Software Interrupt***
Operation	0001H --> PC, (PC+1) --> (Top Stack)
Instruction Cycle	4
Affected Flag	None
Description	Software Interrupt, go to 0001H Address(PC); Push the PC+1 to Top Stack
Example	ORG 0001H JMP SF_INT NOP SF_INT: NOP RETINT ; Feedback to LAB1 Main: NOP **INT** ; Software Interrupt ; go to 0001H Addr. LAB1 NOP ; ; Push LAB1 to Top Stack

RETINT	***Return from Interrupt.***
Operation	(Top Stack)--> PC, MCR/GIE Flag=1
Instruction Cycle	4
Affected Flag	None
Description	Return from Interrupt ; POP The Address from Top Stack
Example	ORG 0001H JMP SF_INT NOP SF_INT: NOP **RETINT** ; Feedback to LAB1 Main: NOP INT ; Software Interrupt ; go to 0001H Addr. LAB1 NOP ; ; Push LAB1 to Top Stack

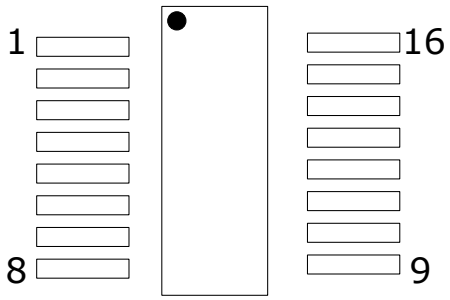
14.0 Electrical Characteristics.

14.1 DC Characteristics

Symbol	Parameter	Condition	Min	Typ	Max	Unit
VOP	Operating Voltage	LDO ON	2.2		5.0	V
IOP	Operating current	VDD=3V / LDO Off		2		mA
ISB	Standby current	VDD=3V / LDO Off		1		uA
FXTL	XTAL frequency	VDD=3V / LDO Off			16	MHz
FIRC	Internal RC Frequency	VDD=3V / LDO Off			16	MHz
IIH	Internal Pull-Low current	VDD=3V / LDO Off		30		uA
IIL	Internal Pull-High current	VDD=3V / LDO Off		40		uA
IOH	Output High current	VDD=3V / LDO Off		7		mA
IOL	Output low current	VDD=3V / LDO Off		14		mA

Appendix.1 Package Information.

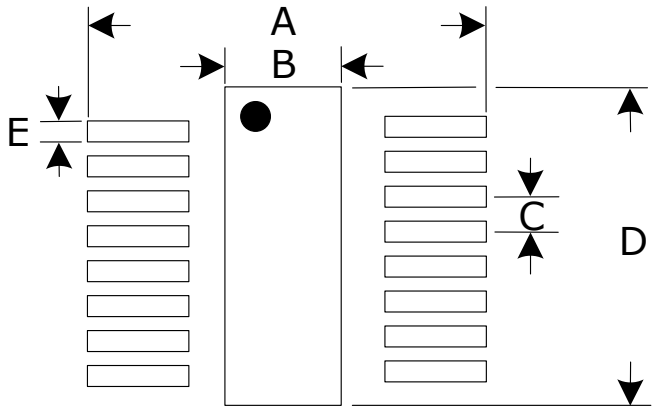
A1.1 SOP16



Pin Definition

Pin	NAME	Pin	NAME	Pin	NAME
1	PA2	7	PB2	13	VDD
2	PA3	8	PB3	14	VDDL
3	RSTN	9	PB4	15	PC1/OSCO
4	VSS	10	PB5	16	PC0/OSCI
5	EINT/PB0	11	PB6		
6	PB1	12	PB7		

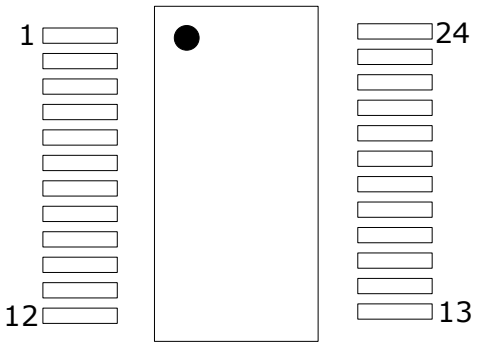
Package Dimension



Symbol	Min	Typ.	Max.
A		7.8 BSC	
B		5.3 BSC	
C		1.27 BSC	
D		10.8 MAX	
E		0.43 BSC	

All dimensions shown in MM.

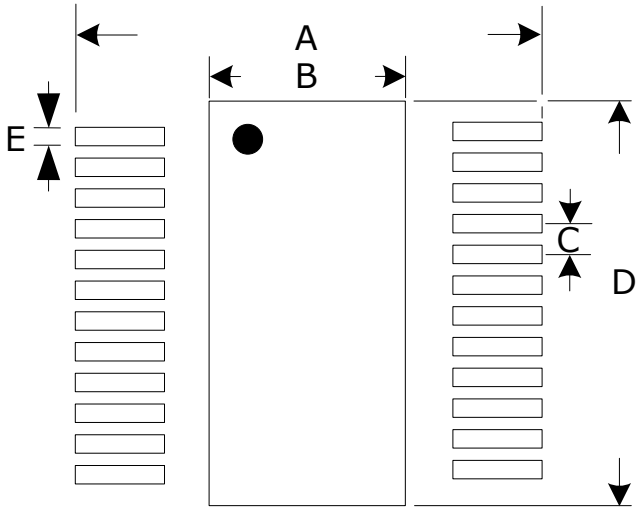
A1.2 SOP24



Pin Definition

Pin	NAME	Pin	NAME	Pin	NAME
1	PA2	9	PB2	17	PB7
2	PA3	10	PB3	18	VDD
3	PA4	11	PB4	19	VDDL
4	RSTN	12	PB5	20	PC2
5	VSS	13	PA6	21	PC0/OSCI
6	PB0/EINT	14	PC4	22	PA0
7	PC3	15	PA7	23	PA1
8	PB1	16	PB6	24	PA5

Package Dimension



Symbol	Min	Typ.	Max.
A	10.008	10.325	10.643
B	7.391	7.493	7.595
C		1.27 BSC	
D	15.215	15.405	15.596
E		0.406 BSC	

All dimensions shown in MM.

Appendix.2 Programming Information.

A2.1 Programming Pins

Name	Write Pin	Writer-48	Writer-20	Notes
PB7	CS	13	19	
PB4	DIO1	38	2	
PB5	DIO2	39	3	
PB6	SCK	12	18	
-	RST	11	4	
VDD	VDD	14	20	
VDDL	VDD			
VSS	VSS	11	4	
RSTN	VSS			
PB0/EINT	VPP	37	1	

Notes.

Writer-48 : The DIP48 slot on the writer.

Writer-20 : The Wide-20 pins slot on the writer.

Appendix.3 Support Information.

A If you need any support or suggestions, there are support method which you can contact with:

E-Mail :
mcu.support@ystek.com.tw
Telephone :
+886-3-5739389(Taiwan) call for MCU.FAE Support.
Websites :
http://www.ystek.com.tw