



十速

TM57MA45/MA46

DATA SHEET

Rev 0.92

tenx reserves the right to change or discontinue the manual and online documentation to this product herein to improve reliability, function or design without further notice. **tenx** does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others. **tenx** products are not designed, intended, or authorized for use in life support appliances, devices, or systems. If Buyer purchases or uses **tenx** products for any such unintended or unauthorized application, Buyer shall indemnify and hold **tenx** and its officers, employees, subsidiaries, affiliates and distributors harmless against all claims, cost, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use even if such claim alleges that **tenx** was negligent regarding the design or manufacture of the part.

AMENDMENT HISTORY

Version	Date	Description
0.90	Nov, 2017	New release.
0.91	Nov, 2017	1. MA45, MA46 維護方便規格書合併成一份 2. 電氣特性部份: 增加類比 IP (OP/CMP/CC/DPDMV)描述
0.92	Feb, 2018	修正 register 錯誤敘述

CONTENTS

AMENDMENT HISTORY	2
CONTENTS.....	3
FEATURES	5
SYSTEM BLOCK DIAGRAM.....	8
PIN ASSIGNMENT DIAGRAM.....	9
PIN DESCRIPTIONS	11
PIN SUMMARY.....	12
FUNCTION DESCRIPTION.....	13
1. CPU Core.....	13
1.1 Clock and Instruction Cycle	13
1.2 Program ROM (PROM)	14
1.3 Programming Counter (PC) and Stack.....	15
1.4 ALU and Working (W) Register	16
1.5 RAM Addressing Mode	17
1.6 STATUS Register (F-Plane 03H)	19
Example: Write immediate data into STATUS register.....	19
Example: Bit addressing set and clear STATUS register.....	19
Example: Determine the C flag by BTFSS instruction.	19
1.7 Interrupt.....	20
2. Chip Operation Mode	24
2.1 Reset	24
2.2 System Configuration Register (SYSCFG).....	25
2.3 Power-Down Mode	26
2.4 System Clock and Operation Mode Selection.....	27
3. Peripheral Functional Block	31
3.1 Watchdog Timer (WDT) / Wakeup Timer (WKT).....	31
3.2 8-bit Timer0.....	34
3.3 Timer1	40
3.4 PWM0: (11+2) bits PWM	46
3.5 PWM1: (11+2) bits PWM	53
3.6 Operational Amplifier and Comparator	59
3.7 Analog-to-Digital Convert	64
3.8 USB Charging Control	67
3.8.1 DPDMV (DM0/DP0)	67
3.8.2 DPDM6B (DM1/DM2 and DP1/DP2).....	69
3.8.3 CC1/CC2	71
3.9 System Clock.....	73
4. I/O Port	75
4.1 PA0~6.....	75
4.2 PA7.....	79
MEMORY MAP.....	84

1. F-Plane	84
2. R-Plane	90
INSTRUCTION SET	96
ELECTRICAL CHARACTERISTICS	109
1. Absolute Maximum Ratings ($T_A = 25^\circ\text{C}$)	109
2. DC Characteristics ($T_A = -25^\circ\text{C}$ to $+85^\circ\text{C}$, $V_{CC} = 5\text{V}$)	109
3. Clock Timing ($T_A = -25^\circ\text{C}$ to $+85^\circ\text{C}$, $V_{CC} = 5\text{V}$)	110
4. Reset Timing Characteristics ($T_A = -25^\circ\text{C}$ to $+85^\circ\text{C}$, $V_{CC} = 5\text{V}$)	110
5. ADC Electrical Characteristics ($T_A = -25^\circ\text{C}$ to $+85^\circ\text{C}$, $V_{CC} = 5\text{V}$)	110
6. OPA Electrical Characteristics ($T_A = -25^\circ\text{C}$ to $+85^\circ\text{C}$, $V_{CC} = 5\text{V}$)	111
7. Comparator Electrical Characteristics ($T_A = -25^\circ\text{C}$ to $+85^\circ\text{C}$, $V_{CC} = 5\text{V}$)	111
8. DPDMMV/DMDM6B Characteristics ($T_A = -25^\circ\text{C}$ to $+85^\circ\text{C}$, $V_{CC} = 5\text{V}$)	111
9. CC1/CC2 Characteristics ($T_A = -25^\circ\text{C}$ to $+85^\circ\text{C}$, $V_{CC} = 5\text{V}$)	112
10. LVR Circuit Characteristics ($T_A = -25^\circ\text{C}$ to $+85^\circ\text{C}$, $V_{CC} = 5\text{V}$)	112
11. Characteristic Graphs	112
PACKAGE INFORMATION	115
● DIP-20 (300mil) Package Dimension	116
● SOP-20 (300mil) Package Dimension	117
● Skinny DIP-24 (300mil) Package Dimension	118
● SOP-24 (300mil) Package Dimension	119
● Skinny DIP-28 (300mil) Package Dimension	120
● SOP-28 (300mil) Package Dimension	121

FEATURES

1. **ROM: 4K x 14 bits MTP (Multi Time Programmable ROM)**
2. **RAM: 184 x 8 bits**
3. **STACK: 8 levels**
4. **System Oscillation Sources(Fsys):**
 - Fast-Clock:
 - FIRC (Fast Internal RC): 12 MHz (can be trimmed)
 - FXT (Fast Crystal): 1~12 MHz
 - Slow-Clock:
 - SIRC (Slow Internal RC): 128 KHz @VCC=3V
 - SXT (Slow Crystal): 32768 Hz
5. **System Clock Prescaler**
 - System Oscillation Source can be divided by 8/ 4/ 2/ 1 as System Clock (Fsys)
6. **Dual System Clock:**
 - FIRC + SIRC
 - FIRC + SXT
 - FXT + SIRC
7. **Power Saving Operation Mode**
 - Fast Mode: CPU is running at fast clock. Slow Clock can be disabled or enabled
 - Slow Mode: CPU is running at slow clock. Fast Clock stops.
 - Stop Mode: Clocks stop, Wake-up Timer is disabled or enabled
8. **Reset Source**
 - Power On Reset / Watchdog Reset / Low Voltage Reset / External Pin Reset
9. **3-Level Low Voltage Reset: 2.9V/2.3V/2.0V**
10. **ISP (In-System Programming) uses only 4 wires (VCC, VSS, PA1, PA0)**
11. **2 Independent Timers**
 - Timer0
 - 8-bit timer divided by 1~256 pre-scaler option, Counter/ Interrupt / Stop function
 - Overflow and Toggle out
 - Timer1
 - 16-bit timer with two pre-scalers, Counter/ Interrupt / Stop / Clear & Hold /Set / Reload function
 - Overflow and Toggle out

12. Wake-up (WKT) Timer

- Clocked by build-in SIRC oscillator with 4 adjust Interrupt time interval
27ms/54ms/108ms/216ms @VCC=5V

13. Watchdog Timer

- Clocked by build-in SIRC oscillator with 4 adjust Interrupt time interval
108ms/216ms/864ms/1728ms @VCC=5V

14. 2 Independent PWMs

- PWM0:
 - 11+2 bits, period-adjustable / duty-adjustable / Clear & Hold
 - Clock Source: 108MHz (output of PLL) or system clock(Fsys)
 - Width differential output pair
 - Non-overlap durations adjustable
 - High Drive/Sink up to 180mA
- PWM1:
 - 11+2 bits, period-adjustable / duty-adjustable / Clear & Hold
 - Clock Source: 108MHz (output of PLL) system clock(Fsys)
 - Width differential output pair
 - Non-overlap durations adjustable

15. Interrupt

- Three external Interrupt pins
 - 1 pin is rising or falling edge wake-up triggered
 - 2 pins are falling edge wake-up triggered
- Timer0 / Timer1 /WKT(wake-up) Interrupts
- PWM0 / PWM1 Interrupt
- CMP comparator Interrupt

16. Operational Amplifier(OPA) and Comparator

- OPA offset voltage < 2mV@Vo=1.5V, Ta=25C, VCC=5V, VSS=0V
- With OPA offset calibration
- Both edge trigger selection for Comparator
- Comparator debounce time selection : 0/ 2/ 4/ 8 system clock period time

17. 12-bit Analog to Digital Convert(ADC):

- 13 external input channel from IO pin
- 1 channel from OPA output
- ADC reference voltage = LDOC (2.5V) for TM57MA45
- ADC reference voltage = VCC (5V) for TM57MA46

- 1 channel from internal reference voltage VBGO (1.25V).

18. I/O Ports: Maximum 22 programmable I/O pins

- Open-Drain Output
- CMOS Push-Pull Output
- Schmitt Trigger Input with pull-up resistor

19. High-Sink and High-Drive I/O for PWM0

- Up to 180mA current I/O : PA0, PD7 (SOP20/DIP20 not support)

20. USB Charging Control

- Support Quick Charge 2.0 specification
- USB battery charging specification revision 1.2 compatible

21. Instruction Set

- 39 instructions

22. Instruction Execution Time

- 2 oscillation clocks per instruction except branch

23. Table Read Instruction: 14-bit ROM data lookup table**24. Operation Voltage: Low Voltage Reset Voltage to 5.5V**

- Fsys = 12 MHz, 2.5V ~ 5.5V

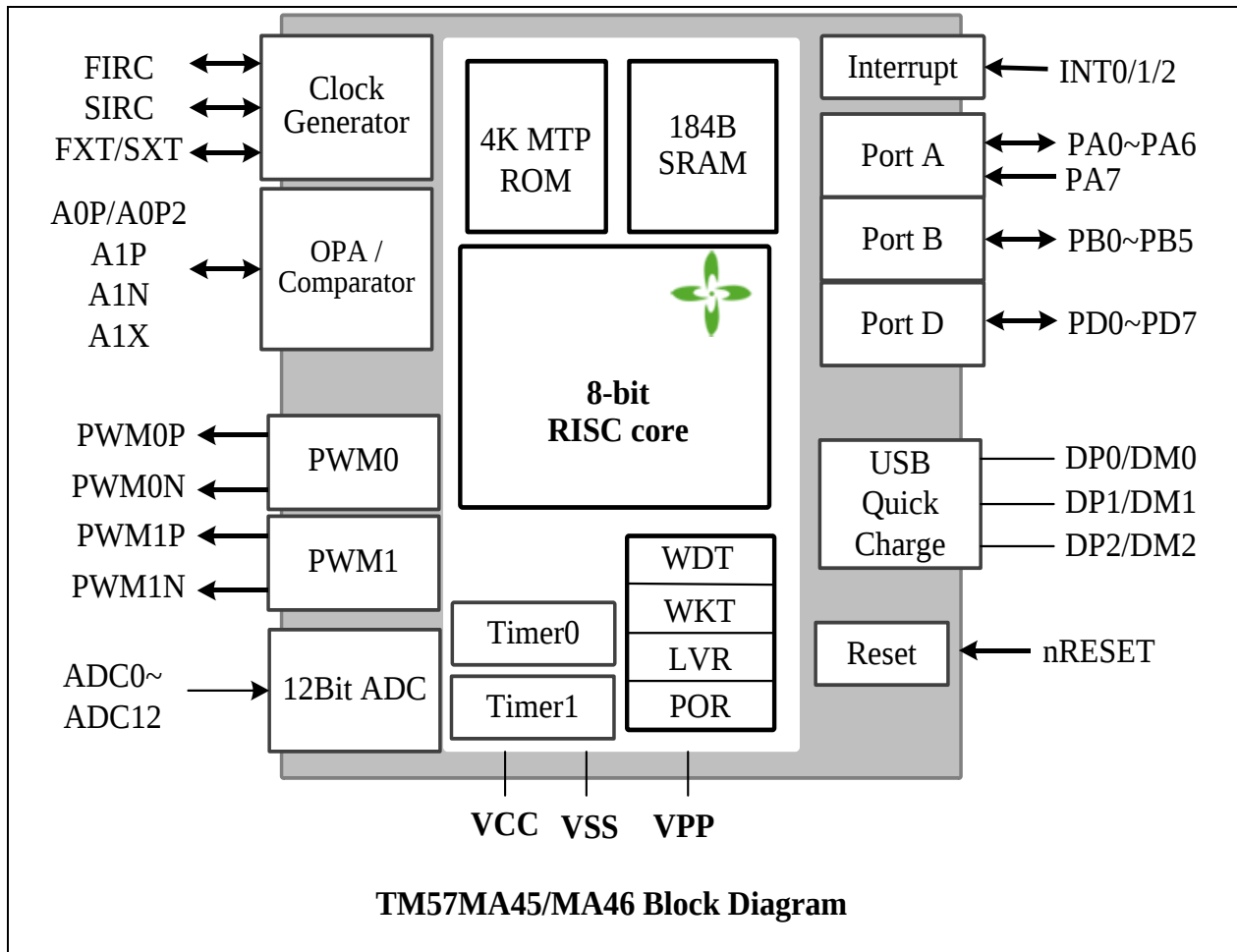
25. Package Type:

- SOP28/DIP28 (TM57MA45: ADC VREF = LDOC=2.5V)
- SOP28/DIP28 (TM57MA46: ADC VREF = VCC=5V)
- SOP24, DIP24 (TM57MA46: ADC VREF = VCC=5V)
- SOP20, DIP20 (TM57MA46: ADC VREF = VCC=5V)

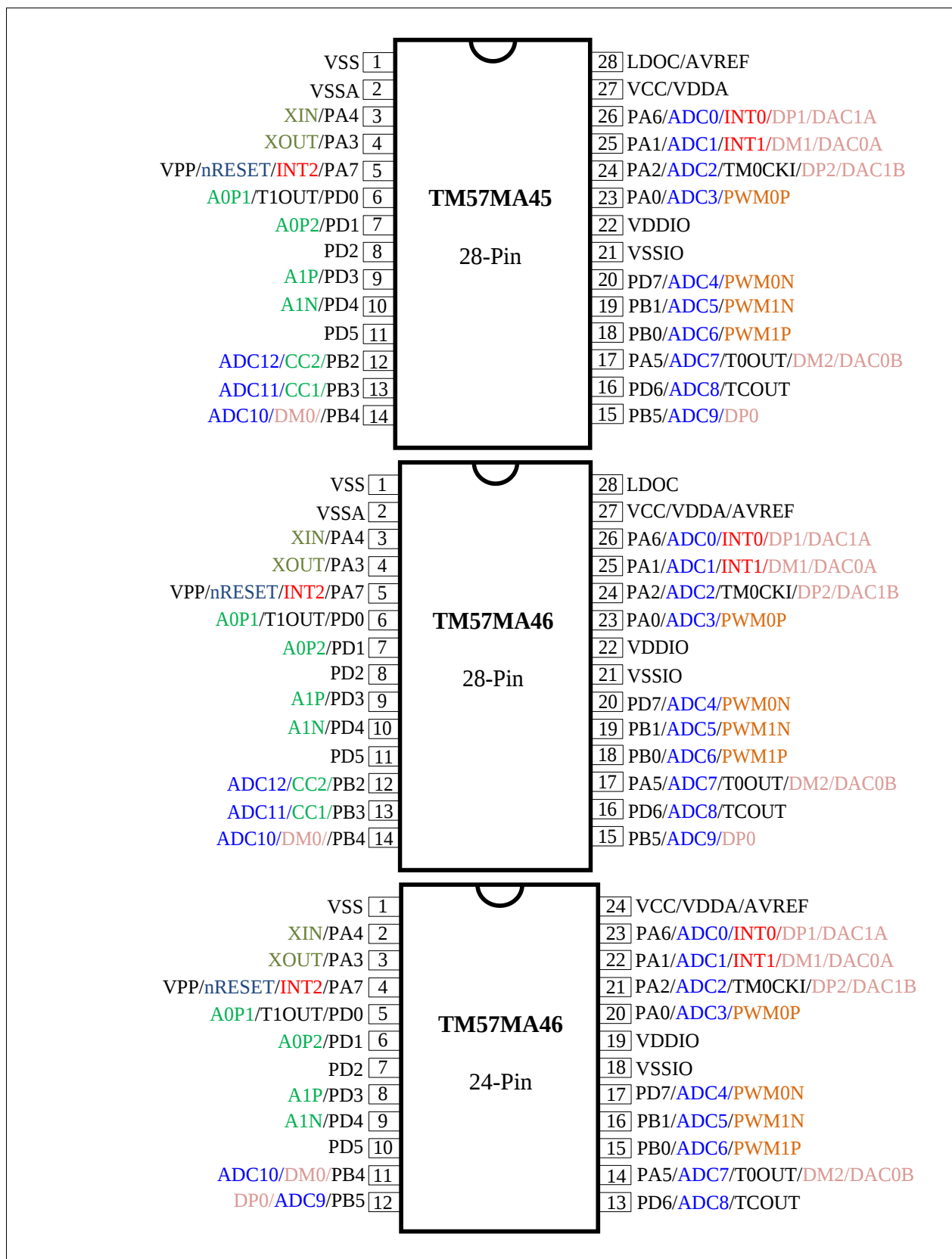
26. Support EV board on ICE

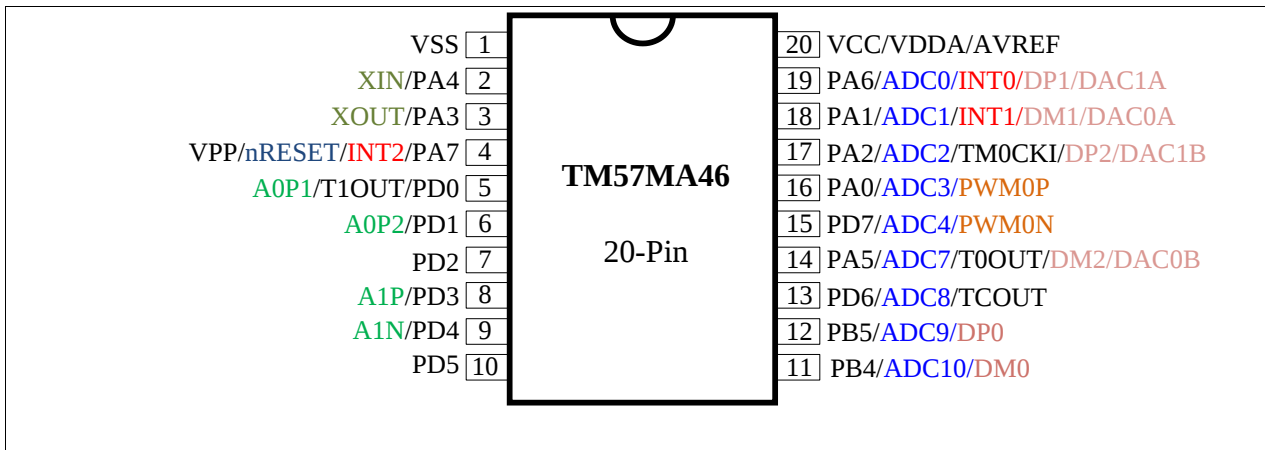
EV board: EV8214

SYSTEM BLOCK DIAGRAM



PIN ASSIGNMENT DIAGRAM





PIN DESCRIPTIONS

Name	In/Out	Pin Description
PA6-PA0	I/O	Bit-programmable I/O port for Schmitt-trigger input, CMOS push-pull output or open-drain output.
VPP/ nRESET/ PA7	I	External Program High Voltage/External active low reset/ Schmitt-trigger input
VCC,VDDA, VSS,VSSA	P	Power input pin and ground
VDDIO, VSSIO	P	Power input pin and ground for PWM0N/PWM0P
INT0~INT2	I	External interrupt input
XIN	I	Xtal input
XOUT	O	Xtal output
PB5-PB0	I/O	Bit-programmable I/O port for Schmitt-trigger input, CMOS push-pull output or open-drain output.
PD7-PD0	I/O	Bit-programmable I/O port for Schmitt-trigger input, CMOS push-pull output or open-drain output.
PWM0N PWM0P PWM1N PWM1P	O	PWM0/PWM1 outputs
ADC12~ADC0	I	Analog to Digital Convert input pin
TM0CKI	I	Timer0 counter mode input pin
TM0OUT	O	Timer0 overflow toggle output
TM1OUT	O	Timer1 overflow toggle output
CC1/CC2	I/O	USB Battery Charge CC1/CC2
DP0/DM0	I/O	USB Battery Charge DP0/DM0
DP1/DM1	I/O	USB Battery Charge DP1/DM1
DP2/DM2	I/O	USB Battery Charge DP2/DM2
DAC0A/DAC0B	O	DAC0 output
DAC1A/DAC1B	O	DAC1 output
A0P1, A0P2	I	Operational Amplifier input OPP
A1P,A1N	I	Comparator input
AVREF	P	ADC VREF (ADC Reference Voltage)=LDOC (for TM57MA45) ADC VREF (ADC Reference Voltage)=VCC (for TM57MA46)
LDOC	O	Low Drop Output(2.5V), when F13.3=0; need connect 1uF to Ground

Programming pins:

Normal mode: VCC / VSS / PA0 / PA1 / PA2 / PA3 / PA4 / PA7 (VPP)

ISP mode with high voltage: VCC / VSS / PA0 / PA1 / PA7 (VPP) -When using ISP (In-system Program) mode, the PCB needs to remove all components of PA0, PA1, PA7.

ISP mode without high voltage: VCC / VSS / PA0 / PA1 -When using ISP (In-system Program) mode with internal high voltage, the PCB needs to remove all components of PA0, PA1.

Pin Summary

Pin number			Pin Name	Type	GPIO				Function after reset	Alternate Function			
SOP28	SOP24	SOP20			Input		Output			PWM	OPA/CMP	ADC	MISC
					Weak Pull-up	Ext. Interrupt	O.D.	P.P.					
1			VSSA	P									
2	1	1	VSS	P									
3	2	2	XIN/PA4	I/O	O		O	O	PA4				
4	3	3	XOUT/PA3	I/O	O		O	O	PA3				
5	4	4	VPP/nRESET/INT2/PA7	I	O	O			PA7				
6	5	5	A0P1/TM1OUT/PD0	I/O	O		O	O	PD0		O		TM1OUT
7	6	6	A0P2/PD1	I/O	O		O	O	PD1		O		
8	7	7	PD2	I/O	O		O	O	PD2		O		
9	8	8	A1P/PD3	I/O	O		O	O	PD3		O		
10	9	9	A1N/PD4	I/O	O		O	O	PD4		O		
11	10	10	A1X/PD5	I/O	O		O	O	PD5		O		
12			CC2/ADC12/PB2	I/O	O		O	O	PB2			O	
13			CC1/ADC11/PB3	I/O	O		O	O	PB3			O	
14	11	11	DM0/ADC10/PB4	I/O	O		O	O	PB4			O	
15	12	12	DP0/ADC9/PB5	I/O	O		O	O	PB5			O	
16	13	13	TCOUT/ADC8/PD6	I/O	O		O	O	PD6			O	TCOUT
17	14	14	DM2/TM0OUT/ADC7/PA5	I/O	O		O	O	PA5			O	TM0OUT
18	15	15	PWM1P/ADC6/PB0	I/O	O		O	O	PB0	O		O	
19	16	16	PWM1N/ADC5/PB1	I/O	O		O	O	PB1	O		O	
20	17		PAM0N/ADC4/PD7	I/O	O		O	O	PD7	O		O	
21	18		VSSIO	P									
22	19		VDDIO	P									
23	20		PWM0P/ADC3/PA0	I/O	O		O	O	PA0	O		O	
24	21	17	DP2/TM0CKI/ADC2/PA2	I/O	O		O	O	PA2			O	TM0CKI
25	22	18	DM1/INT1/ADC1/PA1	I/O	O	O	O	O	PA1			O	
26	23	19	DP1/INT0/ADC0/PA6	I/O	O	O	O	O	PA6			O	
27	24	20	VCC/VDDA	P									
28			LDOC	P									

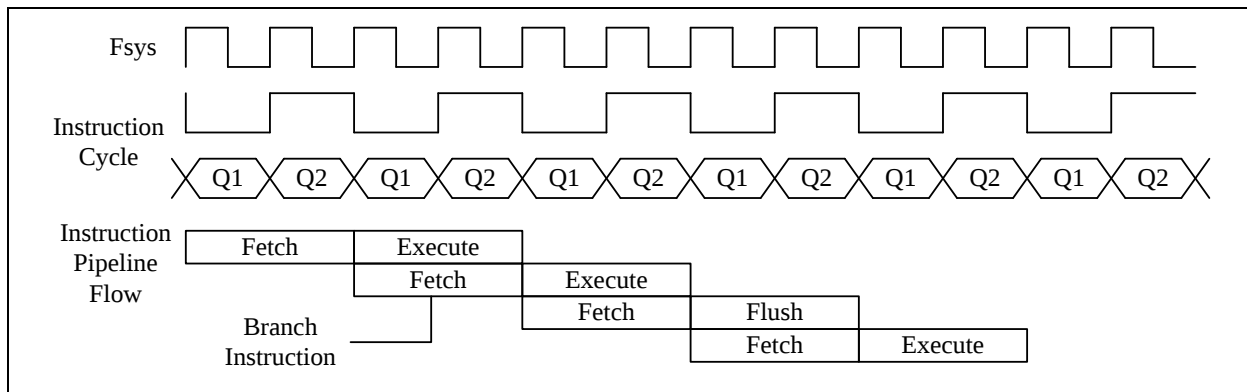
Symbol : O.D. = Open Drain
P.P. = Push-Pull Output

FUNCTION DESCRIPTION

1. CPU Core

1.1 Clock and Instruction Cycle

The system clock 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. Branch instructions take two cycles since the fetch instruction is ‘flushed’ from the pipeline, while the new instruction is being fetched and then executed.



Terminology definitions:

(1) **Fsys:** System clock. The main clock that drives the core logic and all peripherals. The clock source can be either Fast-clock or Slow-clock which can be set by register.

(2) **Instruction Cycle** = $F_{sys} / 2$

FIRC: Fast Internal RC oscillator

SIRC: Slow Internal RC oscillator

FXT: Fast external crystal oscillator

SXT: Slow external crystal oscillator

1.2 Program ROM (PROM)

The MTP Program ROM of this device is 1K words, with an extra INFO area to store the SYSCFG. The ROM can be written multi-times and can be read as long as the PROTECT bit of SYSCFG is not set. The SYSCFG can be read no matter PROTECT is set or cleared, but can be written only when PROTECT is not set or ROM is erased. That is, unprotect the PROTECT bit needs the erased ROM

Program Memory

000	Reset Vector
001	Interrupt Vector
009	
	USER CODE
FFF	SYSCFG (INFO area)

1.3 Programming Counter (PC) and Stack

The Programming Counter is 10-bit wide capable of addressing a 4K x 14 MTP 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 Vectors (from 001h to 009h) are provided for PC initialization and Interrupts. For CALL/GOTO instructions, PC loads the lower 10 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 [11:8] keeps unchanged. The STACK is 12-bit wide and 8-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.

For table lookup, the device offer the powerful table read instructions TABRL, TABRH to return the 14-bit ROM data into W register by setting DPRT={DPH, DPL} F-Plane registers.

◇ Example: to look up the FLASH ROM data located “TABLE1” and “TABLE2”.

```

        ORG      00h                ; Reset Vector
        GOTO     START

START:   ORG      60h

        MOVLW    00h
        MOVWF    INDEX              ; Set lookup table's address
LOOP:    MOVFW    INDEX              ; Move index value to W register
        CALL     TABLE1            ; To lookup data, W=55h
        .....
        GOTO     LOOP              ; Go to LOOP label
        .....
        MOVLW    (TABLE2>>8)&0xff
        MOVWF    DPH                ; DPH register (F18.1~0)
        MOVWF    (TABLE2)&0xff
        MOVWF    DPL                ; DPL register (F17.7~0)
        TABRL
        TABRH                        ; W=86h
        .....                      ; W=19h

TABLE1:  ADDWF    PCL, 1              ; Add the W with PCL, the result back in PCL
        RETLW    55h                ; W=55h when return
        RETLW    56h                ; W=56h when return
        RETLW    58h                ; W=58h when return
        .....
        ORG      f68h

TABLE2:  .DT 0x1986, 0x3719, 0x2983 ; 14-bit FLASH ROM data

```

1.4 ALU and Working (W) Register

The ALU is 8-bit 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.

Note: /Borrow represents inverted of Borrow register.

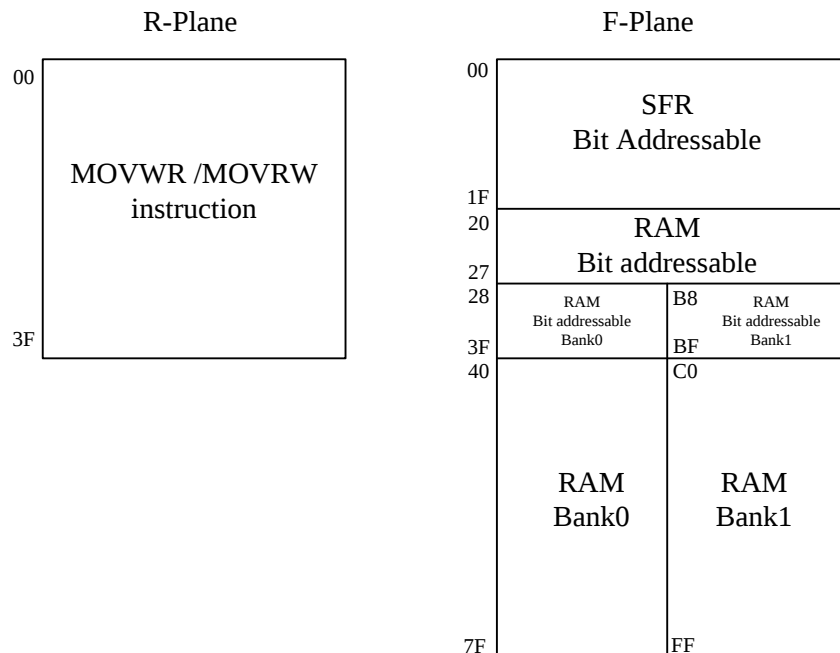
/Digit Borrow represents inverted of Digit Borrow register

1.5 RAM Addressing Mode

There are two Data Memory Planes in CPU, R-Plane and F-Plane. The F-Plane supports rich instructions operation, such as ADDWF, INCF, MOVWF, ..., while the R-Plane only supports MOVWR and MOVWRW instructions to exchange data between R-Plane and W-Register.

The R-plane can be indirect accessed via RSR register (F1C.7~0) and INDR (R00). The INDR register is not a physical register. Addressing INDR actually addresses the register whose address is contained in the RSR register (RSR is a pointer)

The lower locations of F-Plane are reserved for the SFR. Above the SFR is General Purpose Data Memory, implemented as a static RAM. F-Plane 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). The first half of F-Plane is bit-addressable, while the second half of F-Plane is not bit-addressable.



✧ Example: Write immediate data into R-Plane register.

```
MOVLW    AAh        ; Move immediate AAh into W register
MOVWR    05h        ; Move W register value into R-Plane location 05h data register
```

✧ Example: Move the immediate data 55h to W register and F-Plane location 20h

```
MOVLW    55h        ; Move immediate 55h into W register
MOVWF    20h        ; Move W register value into F-Plane location 20h data register
```

✧ Example: Move R-Plane location 0Bh data into W register

```
MOVWR    0Bh        ; To get a content of R-plane location 0BH and save in W.
```

◇ Example: Move F-Plane location 20h data into W register

MOVFW 20h ; Get a content of F-Plane location 20h and save into W.

◇ Example: Indirectly addressing mode with FSR/INDF register. (F-Plane 04h / 00h)

MOVLW 20h
MOVWF FSR ; Move immediate 20h into FSR register
MOVLW 55h
MOVWF INDF ; Use data pointer FSR write data into F-Plane location 20h
 ; (immediate data 20h into F-Plane 20h)

INCF FSR, 1 ; Increment the index address for the next address
MOVWF INDF ; Use data pointer FSR read a data from F-Plane location 21h
 ; (W register get data from F-Plane 21h)

1.6 STATUS Register (F-Plane 03H)

This register contains the arithmetic status of ALU and the Reset status. 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 are 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	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W
Bit	Description							
7	GB0: General Purpose Bit 0							
6	GB1: General Purpose Bit 1							
5	RAMBK: Ram Bank select 0: Bank 0 1: Bank 1							
4	TO: Time Out Flag 0: after Power On Reset, LVR Reset or CLRWDT/SLEEP instruction 1: WDT time out occurs							
3	PD: Power Down Flag 0: after Power On Reset, LVR Reset or CLRWDT instruction 1: after SLEEP instruction							
2	Z: Zero Flag 0: the result of a logic operation is not zero 1: the result of a logic operation is zero							
1	DC: Decimal Carry Flag or Decimal / Borrow Flag							
	ADD instruction				SUB instruction			
	0: no carry 1: a carry from the low nibble bits of the result occurs				0: a borrow from the low nibble bits of the result occurs 1: no borrow			
0	C: Carry Flag or /Borrow Flag							
	ADD instruction				SUB instruction			
	0: no carry 1: a carry occurs from the MSB				0: a borrow occurs from the MSB 1: no borrow			

✧ **Example: Write immediate data into STATUS register.**

```
MOVLW    00h
MOVWF    STATUS    ; Clear STATUS register
```

✧ **Example: Bit addressing set and clear STATUS register.**

```
BSF      STATUS, 0    ; Set C=1
BCF      STATUS, 0    ; Clear C=0;
```

✧ **Example: Determine the C flag by BTFSS instruction.**

```
BTFSS    STATUS, 0 ; Check the carry flag
GOTO     LABEL_1  ; If C=0, goto LABEL_1
GOTO     LABEL_2  ; If C=1, goto LABEL_2
```

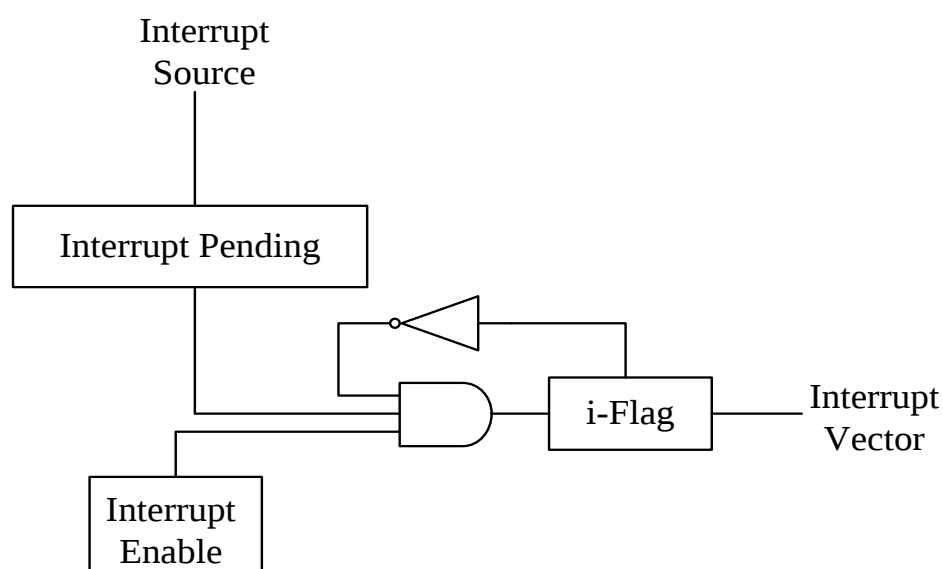
1.7 Interrupt

The TM57MA45/MA46 has 9 vectors and 9 interrupt sources. Each interrupt source has its own enable control bit. An interrupt event will set its individual interrupt flag.

Adr	Description
00	Reset Vector
01	Comparator Interrupt
02	PWM0 Interrupt
03	PWM1 Interrupt
04	External Pin0(PA6) Interrupt
05	External Pin1(PA1) Interrupt
06	External Pin2(PA7) Interrupt
07	Wakeup Timer Interrupt
08	Timer0 Interrupt
09	Timer1 Interrupt

If the corresponding interrupt enable bit has been set, 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 00n”(n=01h ~ 09h) instruction is inserted to CPU, and i-flag is set to prevent recursive interrupt nesting.

The i-Flag is clear 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 level triggered. F/W must clear the interrupt event register while serving the interrupt routine.



✧ Example: Setup INT1 (PA1) interrupt request and rising edge trigger.

```

ORG      000h                      ; Reset vector
GOTO     START                      ; Goto user program address
.....
ORG      01h                      ; Comparator interrupt vector
GOTO     CMP_SUBROUTINE            ; if comparator interrupt occurs
.....
ORG      05h                      ; INT1 interrupt vector
GOTO     INT1_SUBROUTINE           ;if INT1 interrupt occurs
.....

ORG      100h
START:
    MOVLW xxxx00xxB                ; Set INT1 (PA1) pin mode as 0
    MOVLW PAMODL
    BSF    PAD, 1                  ; Release INT1, set PA1 as input with Pull-up resistor
    MOVLW 10100100B                ; Set HWAUTO=1, INT1 interrupt trigger as rising edge
    MOVWR  R0D
    MOVLW 11111101B
    MOVWR  INTIF                   ; Clear INT1 interrupt request flag
    MOVLW 00000010
    MOVWF  INTIE                   ; Enable INT1 interrupt

MAIN:
    .....
    GOTO   MAIN
CMP_ SUBROUTINE:
    MOVLW 11111110b
    MOVWF  INTIF1                  ; clear Comparator interrupt request flag
    .....
    RETI

INT1_SUBROUTINE:
    MOVLW 11111101b
    MOVWF  INTIF                   ; clear INT1 interrupt request flag
    .....
    RETI
.....

```

F08	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIE	PWM0IE	PWM1IE	TM1IE	TM0IE	WKTIE	INT2IE	INT1IE	INT0IE
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

F08.7 PWM0IE: PWM0 interrupt enable
0: disable
1: enable

F08.6 PWM1IE: PWM1 interrupt enable
0: disable
1: enable

- F08.5 TM1IE: Timer1 interrupt enable
0: disable
1: enable
- F08.4 TM0IE: Timer0 interrupt enable
0: disable
1: enable
- F08.3 WKTIE: Wake up Timer interrupt enable
0: disable
1: enable
- F08.2 INT2IE: External Pin2 (PA7) interrupt enable
0: disable
1: enable
- F08.1 INT1IE: External Pin1 (PA1) interrupt enable
0: disable
1: enable
- F08.0 INT0IE: External Pin0 (PA6) interrupt enable
0: disable
1: enable

F09	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIF	PWM0IF	PWM1IF	TM1IF	TM0IF	WKTIF	INT2IF	INT1IF	INT0IF
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

- F09.7 PWM0IF: PWM0 interrupt flag
This bit is set by H/W while PWM0 period is completed, write 0 to this bit will clear this flag
- F09.6 PWM1IF: PWM1 interrupt flag
This bit is set by H/W while PWM1 period is completed, write 0 to this bit will clear this flag
- F09.5 TM1IF: Timer1 interrupt flag
This bit is set by H/W while Timer1 overflows; write 0 to this bit will clear this bit
- F09.4 TM0IF: Timer0 interrupt flag
This bit is set by H/W while Timer0 overflows; write 0 to this bit will clear this flag
- F09.3 WKTIF: Wake up Timer interrupts flag
This bit is set by H/W while WKT time out, write 0 to this bit will clear this flag
- F09.2 INT2IF: External Pin2 (PA7) interrupt event pending flag
This bit is set by H/W at INT2 pin's falling edge, write 0 to this bit will clear this flag
- F09.1 INT1IF: External Pin1 (PA1) interrupt event pending flag
This bit is set by H/W at INT1 pin's falling edge, write 0 to this bit will clear this flag
- F09.0 INT0IF: External Pin0 (PA6) interrupt event pending flag
This bit is set by H/W at INT0 pin's falling/rising edge, write 0 to this bit will clear this flag

F0A	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIE1	-	-	-	-	-	-	-	CMPIE
R/W	-	-	-	-	-	-	-	R/W
Reset	-	-	-	-	-	-	-	0

F0A.1 CMPIE: Comparator interrupt enable
0: disable
1: enable

F0B	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIF1	-	-	-	-	-	-	-	CMPIF
R/W	-	-	-	-	-	-	-	R/W
Reset	-	-	-	-	-	-	-	0

F0B.1 CMPIF: Comparator event pending flag
This bit is set by H/W while comparator interrupt occur, write 0 to this bit will clear this flag

R0D	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MR0D	INT1EDG	TM1CM	HWAUTO	-	-	ADCLKS		
R/W	R/W	R/W	R/W	-	-	R/W		
Reset	0	0	0	-	-	0	0	0

R0D.7 INT1EDG: External Pin1(PA1) trigger edge select
0: INT1 (PA1) pin falling edge to trigger interrupt event
1: INT1 (PA1) pin rising edge to trigger interrupt event

R0D.5 HWAUTO: H/W auto save/restore STATUS w/o TO, PD
0: H/W disables auto save/restore STATUS during Interrupt
1: H/W enables auto save/restore STATUS during Interrupt

2. Chip Operation Mode

2.1 Reset

The TM57MA45/MA46 can be RESET in four ways.

- Power-On-Reset
- Low Voltage Reset (LVR)
- External Pin Reset (PA7)
- Watchdog Reset (WDT)

After Power-On-Reset, all system and peripheral control registers are then set to their default hardware Reset values. The LVR level is selected by the SYSCFG register value.

The Low Voltage Reset features static reset when supply voltage is below a threshold level. There are three threshold level can be selected. The LVR's operation mode is defined by SYSCFG. See the following LVR Selection Table; user must also consider the lowest operation voltage of operating frequency.

LVR Selection Table

LVR level	Operating voltage
LVR29	5.5V > VCC > 3.3V or VCC is fixed at 5.0V
LVR23	5.5V > VCC > 2.7V
LVR22	VCC is wide voltage range

The External Pin reset can be disabled or enabled by the SYSCFG. These reset also clear all the control registers to their default reset value. The TO/PD flag are not affected by these resets.

✧ Example: Defining Reset Vector

```

ORG          000h
GOTO        START          ; Jump to user program address
.....
ORG          100h
START:
.....          ; 100h, the head of user program
.....
GOTO        START

```

2.2 System Configuration Register (SYSCFG)

The System Configuration Register (SYSCFG) is located at MTP INFO area address 0x001. The SYSCFG determines the option for initial condition of MCU. It is written by PROM Writer only. User can select LVR threshold voltage and chip operation mode by SYSCFG. The 13th bit of SYSCFG is code protection selection bit. If this bit is 1, the data in MTP ROM will be protected when user reads MTP ROM.

Bit	13~0	
Default Value	00_0000_xxxx_xxxx	
Bit	Description	
13	PROTECT: Code Protection Select	
	1	Enable
	0	Disable
12	XRSTE External Pin (PA7) Reset Enable	
	1	Enable
	0	Disable (PA7 as input pin)
11-10	LVR: Low Voltage Reset Mode	
	11	LVR level =2.2V, always enable
	10	Reserved
	01	LVR level =2.3V
	00	LVR level =2.9V
9-8	WDTE: WDT Reset Enable	
	11	WDT always enable
	10	WDT enable in FAST/SLOW mode, disable in STOP mode
	0x	WDT disable
7-0	Tenx Reserved	

2.3 Power-Down Mode

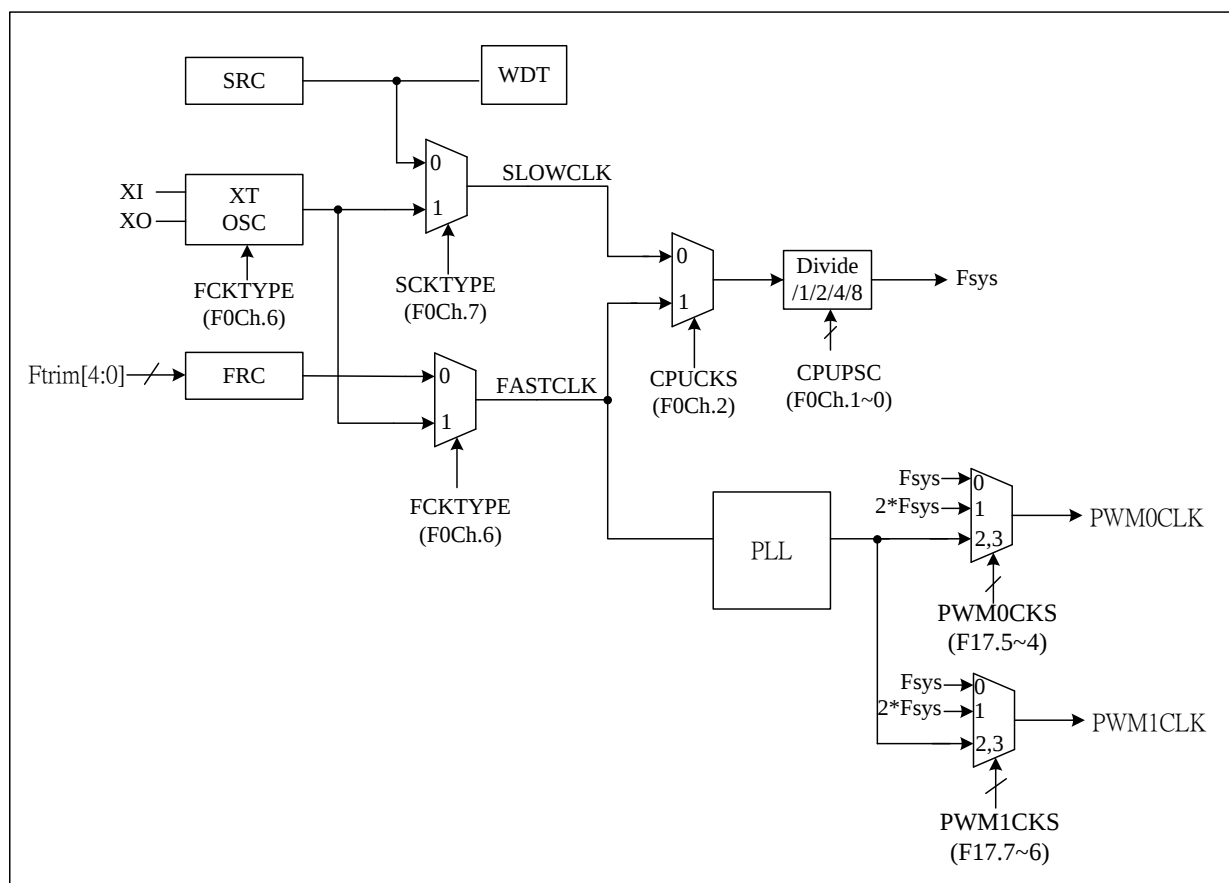
The Power-down mode is activated by SLEEP instruction. During the Power-down mode, the system clock and peripherals stop to minimize power consumption, whether the WDT/WKT Timer are working or not depend on F/W setting. The Power-down mode can be terminated by Reset, or enabled Interrupts (External pins, WKT)

R03	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWRDN	PWRDN							
R/W	W							
Reset	-	-	-	-	-	-	-	-

R03.7~0 PWRDN: Write this register to enter Power-down Mode

2.4 System Clock and Operation Mode Selection

TM57MA45/MA46 is designed with dual-clock system. There are four kinds of clock source, FIRC (Fast Internal RC), FXT (Fast Crystal, 1~12MHz), SIRC (Slow Internal RC) and SXT (Slow Crystal, 32KHz). Each clock source can be applied to CPU kernel as system clock. Refer to the Figure as below. There is a PLL module in TM57MA45/MA46. This PLL can pump the 12MHz FASTCLK to 108MHz clock for PWM.



Clock Scheme Block Diagram

TM57MA45/MA46 is operated indifferent modes: Fast Mode, Slow Mode and Stop mode.

Fast Mode

In this Mode, the program is executed using Fast-clock as CPU clock (Fsys). The TM57MA45/MA46 enters Fast mode by setting the CPUCKS (F0C.2) when it is in SLOW mode. If user want to change to Slow mode, Slow-clock should be enabled first (SLOWSTP=0), then switch to Slow-clock as CPU clock (CPUCKS=0), turn off Fast-clock (FASTSTP=1) in the end.

Slow Mode

After power-on or reset, device enter Slow mode, the system default clock is SIRC. In this mode, the Fast-clock can be stopped (by FASTSTP=1, for power saving) or run (by FASTSTP=0), and Slow-clock is enabled. The TM57MA45/MA46 enters Slow mode by clear the CPUCKS when it is in Fast mode. If user want to change to Fast mode, Fast-clock should be enabled first (FASTSTP=0), then switch to Fast-clock as CPU clock (CPUCKS=1).

STOP Mode

When TM57MA45/MA46 is running in Slow Mode, if Slow-clock and WKT/WDT are disabled before executing the SLEEP instruction, every block is turned off and the device enters the STOP mode. Once the device is in STOP mode, the only way to wakeup device is by external pin interrupt.

CPU Mode & Clock Functions Table

Mode	Oscillator	Fsys	Fast-clock	Slow-clock	TM0/TM1 ADC PWM0~1	Wakeup
Fast	FIRC/XTAL	Fast-clock	Run	Set by SLOWSTP	Run	-
Slow	SIRC/XTAL	Slow-clock	Set by FASTSTP	Run	Run	-
Stop	Stop	Stop	Stop	Stop	Stop	IO

FAST mode switches to SLOW mode

The source clock of Slow-clock is SIRC or SXT according to SCKTYPE (F0C.7). If SCKTYPE is set the clock source of Slow-clock is Slow Crystal (SXT), otherwise is Slow Internal RC (SIRC). The following steps are suggested to be executed by order when FAST mode transits to SLOW mode.

Select Slow-clock type(SXT: SCKRTPE=1, SIRC: SCKTYPE=0)

Switch system clock source to Slow-clock (CPUCKS=0)

Stop Fast-clock (FASTSTP=1)

Example: Switch operating mode from FAST mode to SLOW mode with SXT

BSF SCKTYPE ; select SXT as Slow-clock source

BCF CPUCKS ; switch system clock source to Slow-clock

BSF FASTSTP ; stop Fast-clock

SLOW mode switches to FAST mode

In this mode the system clock source can be FIRC or FXT according FCKTYPE (F0C.6). If FCKTYPE=0, the system clock is Fast Internal RC (FIRC), otherwise the clock source is Fast Crystal (FXT). The following steps are suggested to be executed by order when SLOW mode transits to FAST mode.

Enable Fast-clock (FASTSTP=0)

Switch system clock source to Fast-clock (CPUCKS=1)

Example: Switch operating mode from SLOW mode to FAST mode with FIRC

BCF FASTSTP ; enable Fast-clock

BCF FCKTYPE ; select FIRC as Fast-clock source

BSF CPUCKS ; switch system clock source to Fast-clock

STOP Mode Setting

The STOP mode can be configured by following setting in order:

Stop Slow-clock (SLOWSTP =1)

Stop WKT/WDT (WKTIE=0)

Shut down IVC/OPA, Comparator/LDOC for save power(PDIVC=1, PDANG=1, PDPLL=1, PDLDO=1) for power saving

Execute Sleep instruction

Example: Switch FAST/SLOW mode to STOP mode

BSF SLOWSTP ; disable Slow-clock

BCF WKTIE ; disable WDT/WKT

MOVLW 11x11xxxB ; shut down all analog power

MOVWF F13

MOVLW xx11xxxx ; power saving option

MOVWF F0D

SLEEP

IO setting notes in STOP mode:

Note: In STOP mode, PA3 and PA4 must be set as input mode with internal pull-up enable (IO mode 0) to avoid floating state when select FXT or SXT mode.

	Fast-clock	Slow-clock	PAD[4:3]	PAMODH[1:0]	PAMODL[7:6]
1	FIRC	SIRC	X	X	X
2	FIRC	SXT	11B	00B	00B
3	FXT	SIRC	11B	00B	00B

X: Don't care

F0C	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CLKCTL	SCKTYPE	FCKTYPE	-	SLOWSTP	FASTSTP	CPUCKS	CPUPSC	
R/W	R/W	R/W	-	R/W	R/W	R/W	R/W	
Reset	0	0	-	0	0	0	1	1

F0C.7 SCKTYPE: Slow-clock type

0: SIRC

1: SXT

F0C.6 FCKTYPE: Fast-clock type

0: FIRC

1: FXT

F0C.4 SLOWSTP: Slow-clock stop running bit

0: enable Slow-clock running

1: Stop Slow-clock running

F0C.3 FASTSTP: Fast-clock stop running bit

0: enable Fast-clock running

1: Stop Fast-clock running

F0C.2 CPUCKS: CPU clock source selection bit
 0: select Slow-clock as system clock
 1: select Fast-clock as system clock

F0C.1~0 CPUPSC: System clock source prescaler.
 00: divided by 8
 01: divided by 4
 10: divided by 2
 11: divided by 1

F0D	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MF0D	ADEOC	ADST	PWRSAPV		-	-	-	-
R/W	R/W	R/W	R/W		-	-	-	-
Reset	0	0	1	1	-	-	-	-

F0D.5 PWRSAPV[1]: LVR auto power save enable bit
 0: disable LVR auto power off in STOP mode
 1: enable LVR auto power off in STOP mode

F0D.4 PWRSAPV[0]: IVC auto power save enable bit
 0: disable IVC auto power off in STOP mode
 1: enable IVC auto power off in STOP mode

F13	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ANA_PWR	PDANG	PDPLL	-	PDIVC	PDLDO	PUMP	VCCFLT	CLKFLT
R/W	R/W	R/W	-	R/W	R/W	R/W	R/W	R/W
Reset	1	1	-	0	1	0	0	0

F13.7 PDANG: OPA/CMP power down enable bit
 0: power down disable
 1: power down enable

F13.6 PDPLL: PLL power down enable bit
 0: power down disable
 1: power down enable

F13.4 PDIVC: IVC power down enable bit
 0: power down disable
 1: power down enable

F13.3 PDLDO: LDO power down enable bit
 0: power down disable, LDOC pin output 2.5V
 1: power down enable

R03	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWRDN	PWRDN							
R/W	W							
Reset	-	-	-	-	-	-	-	-

R03.7~0 PWRDN: Write this register to enter Power Down mode

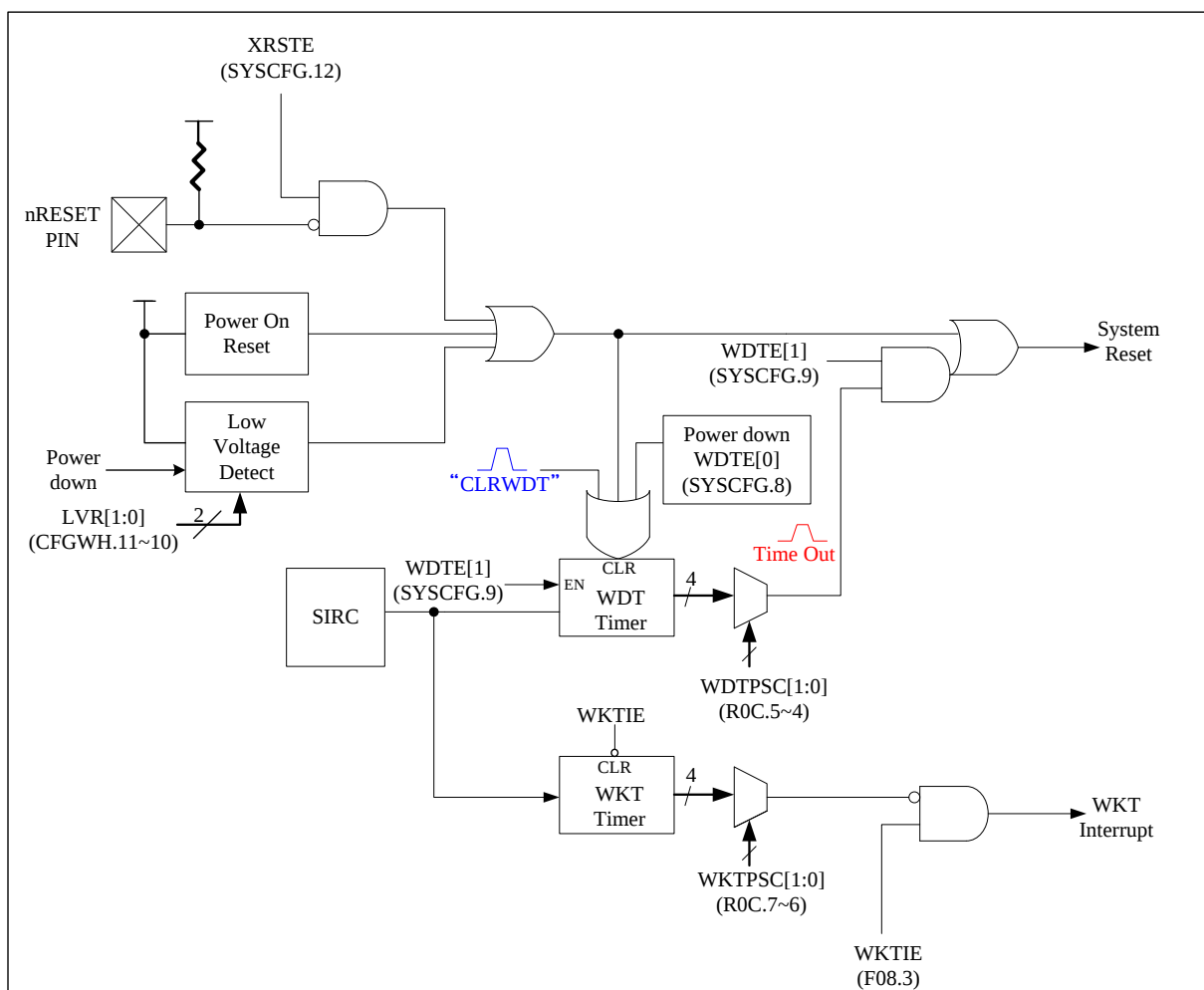
3. Peripheral Functional Block

3.1 Watchdog Timer (WDT) / Wakeup Timer (WKT)

The WDT and WKT share the same internal RC oscillator (SIRC) and have individual own counters. The overflow period of WDT/WKT can be selected by individual prescaler (WDTPSC [1:0], WKTPSC [1:0]).

If the Watchdog is enabled (SYSCFG [9], WDTE [1]=1), the WDT generates the chip reset signal when WDT overflows. Watchdog clear is controlled by CLRWDT instruction and moving any value into WDTCLR (R04) to clear watchdog timer.

The WKT timer is an interval timer. When WKT timer overflows time out, it will generate overflow time out flag “WKTIF” (F09.3). The WKT timer is cleared/stopped by WKTIE=0. Set WKTIE=1, the WKT timer generate WKT overflow time out interrupt and always count regardless at any CPU operating mode.



WDT/WKT Block Diagram

Example: Clear watchdog timer by CLRWDT instruction

MAIN:

```

...                               ; execute program
CLRWDT                           ; execute CLRWDT instruction
...
GOTO      MAIN

```

Example: Clear watchdog by write WDTCLR register

MAIN:

```

...                               ; execute program
MOVWR      WDTCLR                 ; write any value into WDTCLR register
...
GOTO      MAIN

```

Example: set WKT period and interrupt function

```

MOVLW      10xxxxxxB             ; R0C.7~6=2(WKTPSC)
MOVWR      MR0C
MOVLW      11110111B             ; clear WKT interrupt flag by using byte operation
                                   ; don't use bit operation "BCF WKTIF" to clear flag
MOVWF      INTIF                  ; write 11110111B data to F-Plane 09H
MOVLW      00001000B             ; enable WKT interrupt function
MOVWF      INTIE

```

F03	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STATUS	GB1	GB0	RAMBK	TO	PD	Z	DC	C
R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

F03.4 TO: WDT time out flag, read-only

0: after Power On Reset, LVR Reset or CLRWDT/SLEEP instructions

1: WDT time out occurs

F08	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIE	PWM0IE	PWM1IE	TM1IE	TM0IE	WKTIE	INT2IE	INT1IE	INT0IE
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

F08.3 WKTIE: Wake up Timer interrupt enable

0: disable

1: enable

F09	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIF	PWM0IF	PWM1IF	TM1IF	TM0IF	WKTIF	INT2IF	INT1IF	INT0IF
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

F09.3 WKTIF: Wake up Timer event pending flag

This bit is set by H/W while WKT time out, write 0 to this bit will clear this flag

R04	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WDTCLR	WDTCLR							
R/W	W							
Reset	-	-	-	-	-	-	-	-

R04.7~0 WDTCLR: Write this register to clear WDT

R0C	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MR0C	WKTPSC		WDTPSC		TM1PSC	TM0OE	TCOE	TM1OE
R/W	W		W		W	W	W	W
Reset	0		0		0	0	0	0

R0C.7~6 WKTPSC: WKT pre-scale select (the time IS NOT precise enough for accurate timing application)

00: 27ms
01: 54ms
10: 108ms
11: 216ms

R0C.5~4 WDTPSC: WDT pre-scale select (the time IS NOT precise enough for accurate timing application)

00: 108ms
01: 216ms
10: 864ms
11: 1728ms

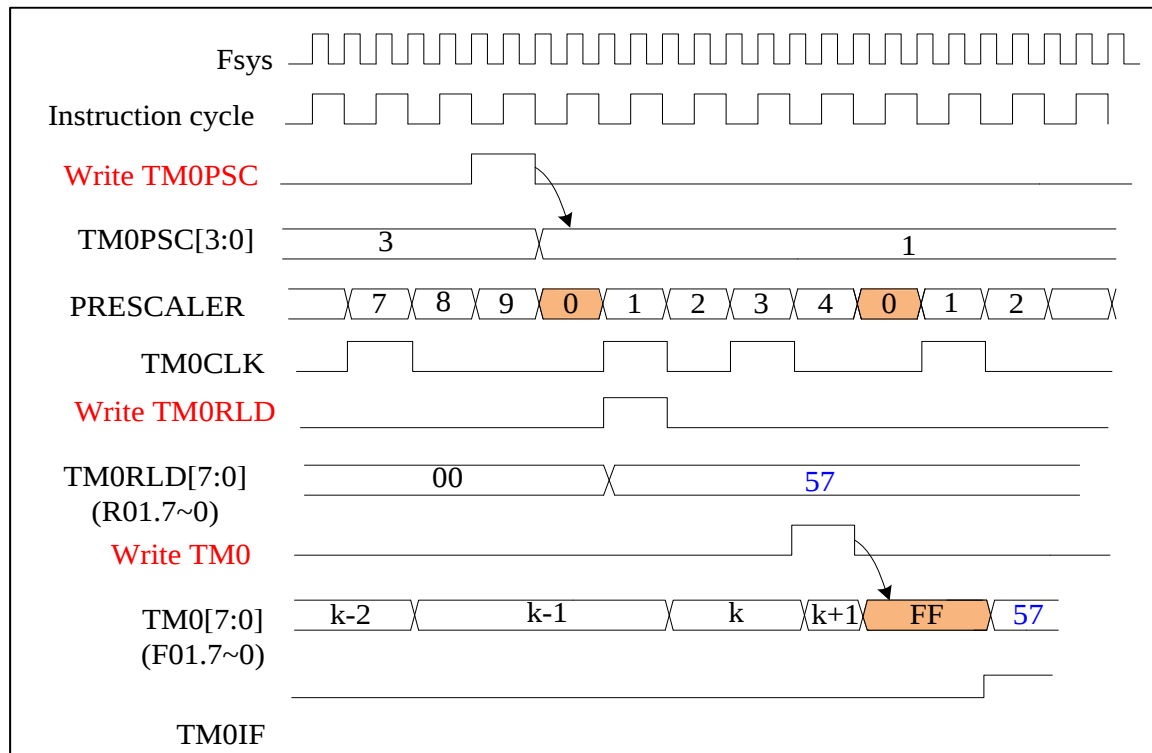


The Timer0 is an 8-bit wide register of F-Plane 01h (TM0). It can be read or written as any other registers of F-Plane. Besides, Timer0 increases itself periodically and automatically rolls over based on the pre-scaled clock source, which can be the instruction cycle or TM0CKI (PA2) rising/falling input. The Timer0's increasing rate is determined by the TM0PSC (R02.3~0). The Timer0 always generates interrupt flag TM0IF (F09.4) and also reload the new data from TM0RLD (R01.7~0) when it rolls over. It generates Timer0 interrupt if the TM0IE (F08.4) bit is set. Timer0 can be stopped counting if the TM0STP (F12.0) bit is set.



Timer Mode:

When the Timer0 prescaler (TM0PSC) is written, the internal 8-bit prescaler will be cleared to 0 to make the counting period correct at the first Timer0 count. TM0CLK is the internal signal that causes the Timer0 to increase by 1 at the end of TM0CLK. TM0WR is also the internal signal that indicates the Timer0 is directly written by instruction; meanwhile, the internal 8-bit prescaler will be cleared. When Timer0 counts from FFh to TM0RLD data, TM0IF (Timer0 Interrupt Flag) will be set to 1 and generate interrupt if TM0IE (Timer0 Interrupt Enable) is set. The following timing diagram describes the Timer0 works in pure timer mode.



Timer0 Works in Timer mode

The equation of TM0OUT initial value is as following.

$$\text{TM0OUT output frequency} = \text{Instruction cycle} / \text{TM0PSC} / (256 - \text{TM0RLD})$$

Example:

Setup Timer0 to work in Timer mode and counting overflow toggle output to TM0OUT (PA5) pin configuration.

; setup Timer0 clock source and divider.

```
MOVLW 00000101B
```

```
MOVWR TM0CTL
```

; Timer0 clock source = Instruction cycle

; Divide by 32

; set Timer0 reload data

```
MOVLW 80h
```

```
MOVWR TM0RLD
```

; set timer0 reload data = 128

; set Timer0 timer.

```
BSF TM0STP
```

; disable Timer0 counting (default "0")

```
MOVLW 00H
```

```
MOVWF TM0
```

; clear Timer0 content

; setup TM0OUT pin function.

```
MOVLW 11010100B
```

```
MOVWR R0C ; enable Timer0 match toggle output to TM0OUT (PA5)
```

; enable TM0 timer and interrupt function.

```
MOVLW 11101111B ; clear Timer0 request interrupt flag
```

```
MOVWF INTIF
```

```
MOVLW 00010000 ; enable Timer0 interrupt function
```

```
MOVLW INTIE
```

```
BCF TM0STP ; enable Timer0 counting (default “0”)
```

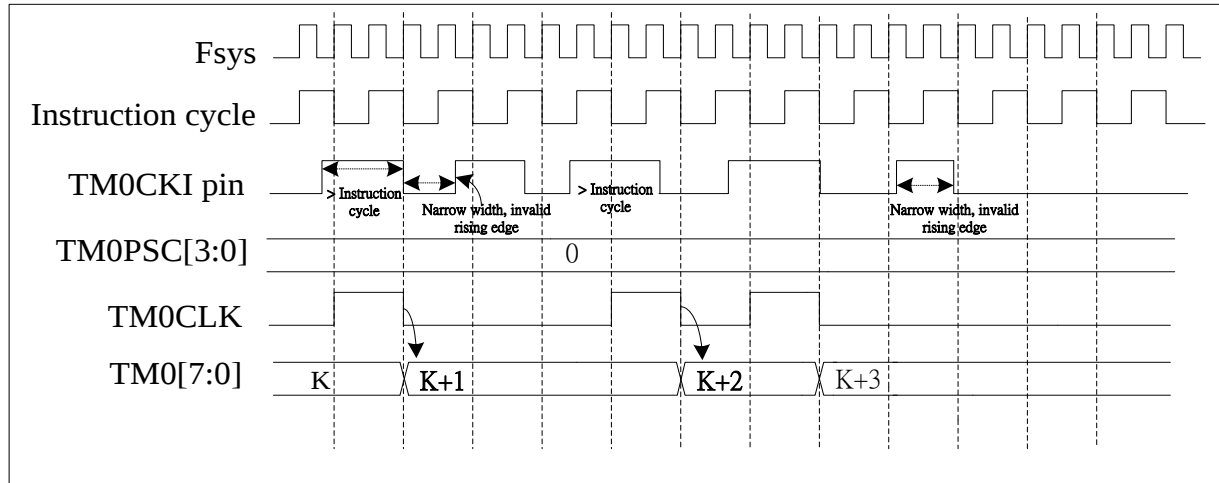
Example:

Timer0 clock source is $F_{sys}/2 = 6\text{MHz}$, Timer0 divided by 32

Timer0 interrupt frequency = $6\text{MHz} / 32 / (256 - 128) = 1464.84\text{Hz}$

Counter Mode:

If TM0CKS=1 then Timer0 counter source clock is from TM0CKI Pin. TM0CKI signal is synchronized by instruction cycle, which means the high/low time duration of TM0CKI must be longer than one instruction cycle time to guarantee each TM0CKI's change will be detected correctly by the synchronizer. The following timing diagram describe the Timer0 works in counter mode



Timer0 works in counter mode for TM0CKI (TM0EDG=0)

Example

Setup Timer0 to work in counter mode and clock source from TM0CKI pin (PA2) configuration.

; setup Timer0 clock source from TM0CKI pin (PA2) and divider.

```
MOVLW 00010000B
```

```
MOVWR TM0CTL      ; select Timer0 prescaler counting edge = rising edge
                  ; Timer0 clock source = TM0CKI pin (PA2)
                  ; divided by 1
```

; set Timer0 timer and stop Timer0 counting

```
BSF    TM0STP      ; disable Timer0 counting
```

```
MOVLW 00H
```

```
MOVWF  TM0         ; write 0 into Timer0 register of F-Plane
```

; set TM0 count and read Timer0 count

```
BCF    TM0STP      ; enable Timer0 counting
```

```
NOP
```

```
NOP
```

NOP

BSF TM0STP ; disable Timer0 counting (default “0”)

MOVFWD TM0 ; read Timer0 value

F01	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TM0	TM0							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

F01.7~0 TM0: Timer0 content

F08	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIE	PWM0IE	PWM1IE	TM1IE	TM0IE	WKTIE	INT2IE	INT1IE	INT0IE
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

F08.4 TM0IE: Timer0 interrupt enable

0: disable

1: enable

F09	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIF	PWM0IF	PWM1IF	TM1IF	TM0IF	WKTIF	INT2IF	INT1IF	INT0IF
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

F09.4 TM0IF: Timer0 interrupt flag

This bit is set by H/W while Timer0 overflows; write 0 to this bit will clear this flag

F12	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MF12	PWM0CLR	PWM1CLR	-	-	TM1SET	TM1CLR	TM1STP	TM0STP
R/W	R/W	R/W	-	-	R/W	R/W	R/W	R/W
Reset	0	0	-	-	0	0	0	0

F12.0 TM0STP: Timer0 counter stop

0: timer0 is counting

1: timer0 stops counting

R02	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TM0CTL	TM0CL	TM0CM	T0EDG	TM0CKS	TM0PSC			
R/W	W	W	W	W	W			
Reset	0	0	0	0	0	0	0	0

R02.7 TM0CL: Timer0 capture mode level

0: CAPT pin high level capture

1: CAPT pin low level capture

R02.6 TM0CM: Timer0 mode selection

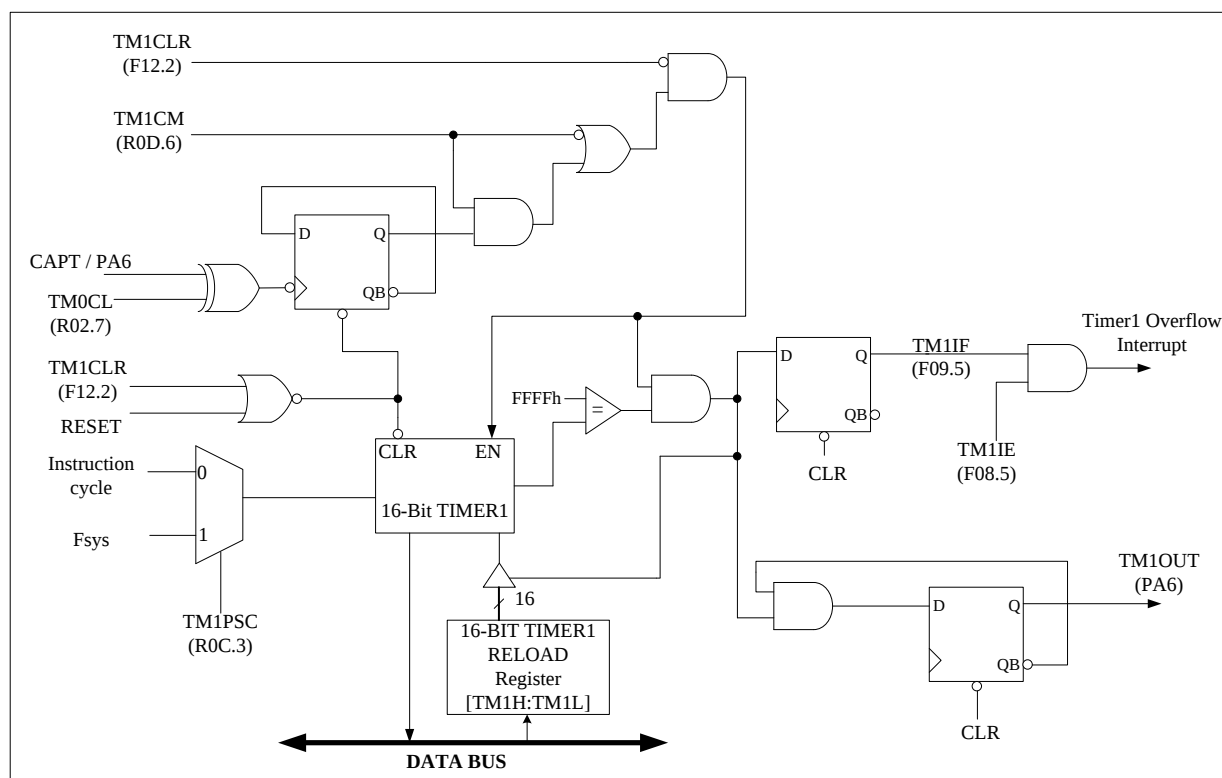
0: Timer/Counter mode, clock source from Instruction cycle ($F_{sys}/2$) or TM0CKI

1: Capture mode, CAPT pin level duration

- R02.5 TM0EDG: TM0CKI (PA2) edge selection for Timer0 prescaler count
0: TM0CKI (PA2) rising edge for Timer0 prescaler count
1: TM0CKI (PA2) falling edge for Timer0 prescaler count
- R02.4 TM0CKS: Timer0 clock source select
0: Instruction cycle ($F_{sys}/2$) as Timer0 prescaler clock
1: TM0CKI (PA2) as Timer0 prescaler clock
- R02.3~0 TM0PSC: Timer0 prescaler. Timer0 clock source
0000: divided by 1
0001: divided by 2
0010: divided by 4
0011: divided by 8
0100: divided by 16
0101: divided by 32
0110: divided by 64
0111: divided by 128
1xxx: divided by 256

3.3 Timer1

Timer1 is a 16-bit counter used as Capture/Timer mode with 16-bit auto-reload register. Timer1 can only be accessed by reading F-plane TM1H and TM1L. Writing TM1H and TM1L is actually writing to Timer1 reload registers. The clock sources of Timer1 are Fsys and Instruction cycle, selected by TM1PSC. Setting the bit TM1CLR will clear Timer1 and hold Timer1 on 0000h. Setting the TM1STP bit will stop Timer1 counting. TM1OUT is an output signal that toggles when Timer1 overflows.



TM1 Block Diagram

Note that writing to TM1H and TM1L is actually writing to Timer1 reload register, while reading TM1H and TM1L is actually reading the Timer1 counter itself. That is, Timer1 counter and Timer1 reload register share two addresses (14h, 15h) of F-plane.

Example:

Setup Timer1 work in Timer mode and counting overflow toggle output to TM1OUT(PA6) pin configuration.

; setup Timer1 clock source and divider

```

MOV LW 00001001B ; TM1OE=1 (enable TM1OUT)

MOV WR R0C ; TM1PSC=1 (Fsys as Timer1 clock source)

MOV LW 00000000B ; TM1CM=0(Timer1 as timer mode)

MOV WR R0D
    
```

; set Timer1 timer

BSF TM1STP ; stop Timer1 counting (default “0”)

BCF TM1SET

BSF TM1CLR ; clear Timer1 counter (default “0”)

MOVLW FFH

MOVWF TM1H ; write FFH into Timer1 counting high byte

MOVLW 00H

MOVWF TM1L ; write 00H into Timer1 counting low byte

; enable Timer1 timer and interrupt function

MOVLW 11011111B ; clear Timer1request interrupt flag

MOVWF INTIF

BSF TM1IE ; enable Timer1 interrupt function

BCF TM1SET

BCF TM1CLR

BCF TM1STP ; enable Timer1 counting (default “0”)

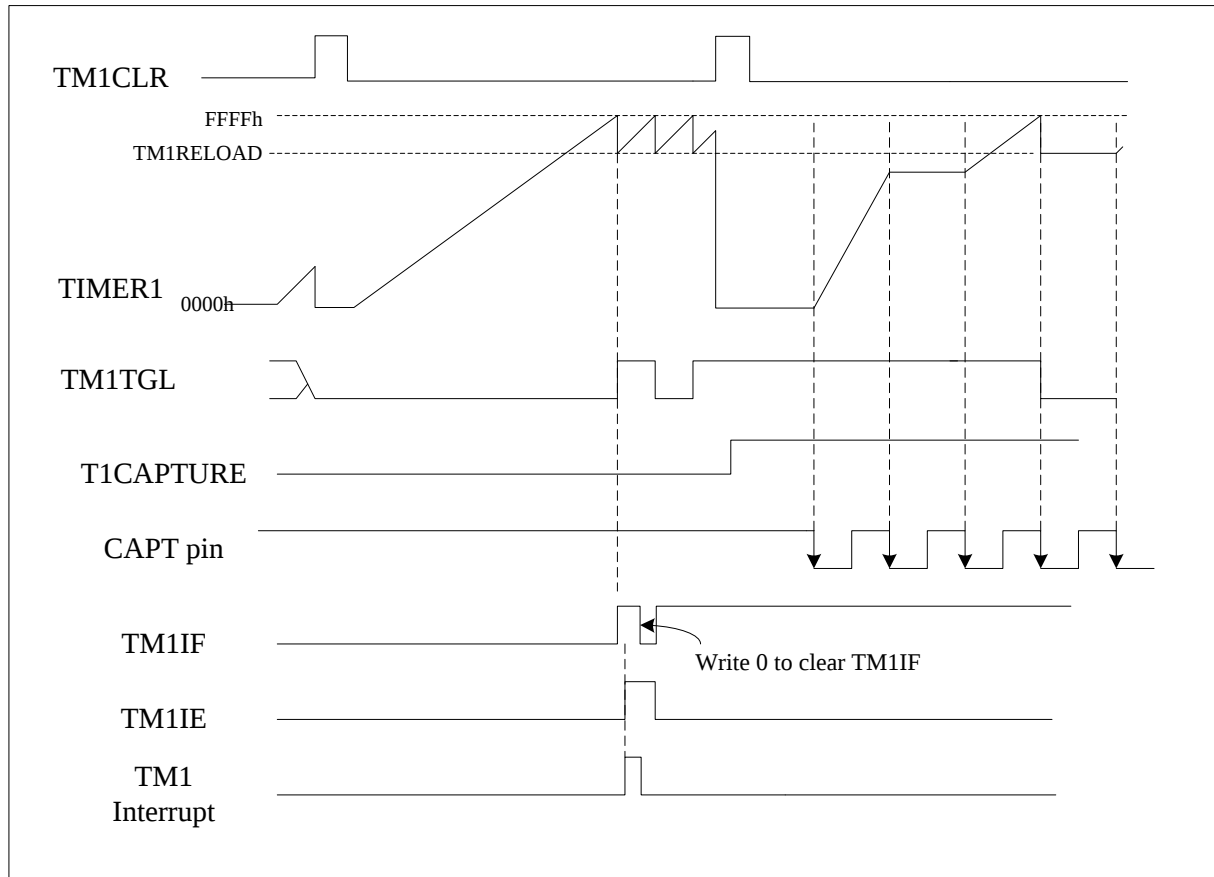
Example:

Timer1 clock source prescaler is $F_{sys} = 4\text{MHz}$, Timer1 MSB = FFH, TM1 LSB = 01H

TM1OUT output frequency = $2\text{MHz} / (FFFF - FF00) = 2\text{MHz} / 256 = 7.8\text{KHz}$

TM1OUT output time period = $1 / 7.8\text{KHz} = 128\mu\text{s}$

Timer1 can also works with Capture mode. When works in Capture mode, Timer1 will start counting when the TM1CLR bit is cleared and the first falling edge of CAPT pin (if TM0CL=0) is counting. When the 2nd falling edge of CAPT pin counting, Timer1 stop counting and hold the value. When the 3rd falling edge of CAPT pin is coming, the Timer1 continues counting. The following figure shows the detail timing diagram



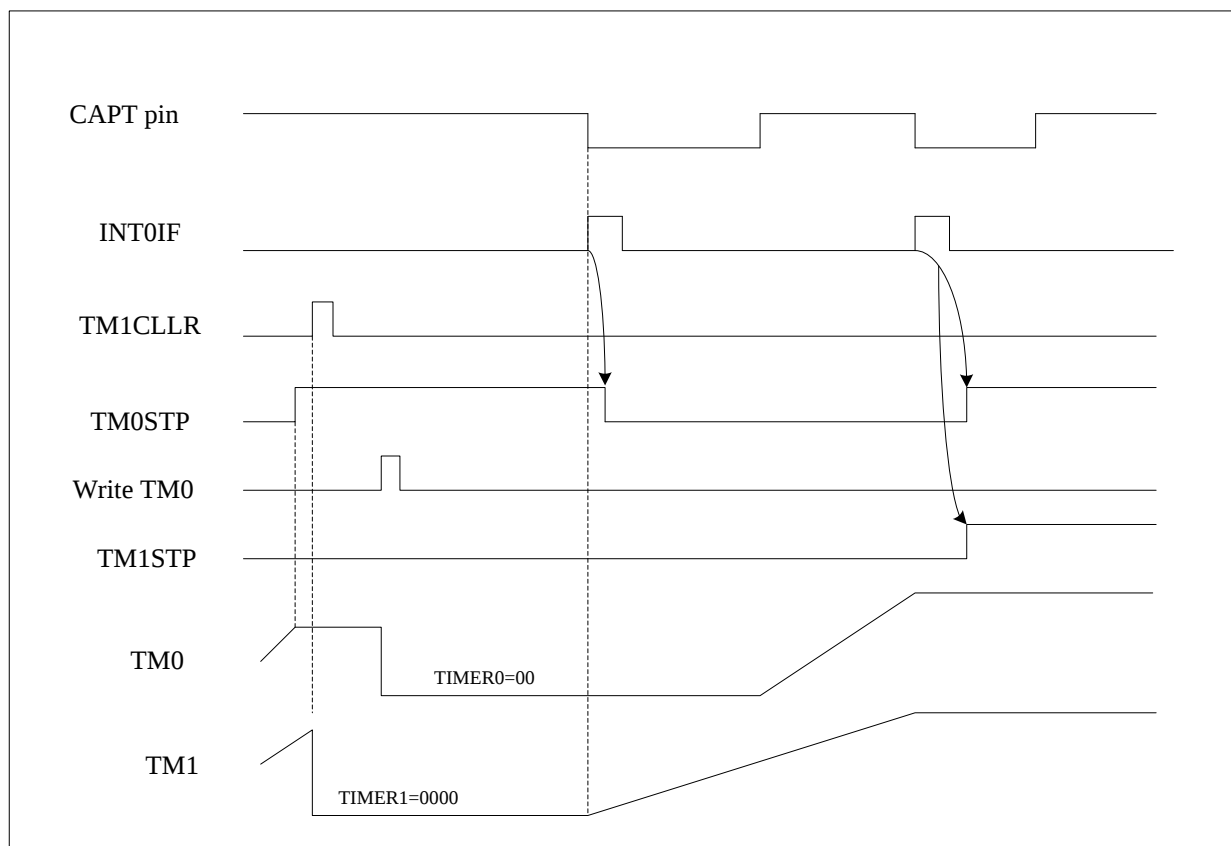
Timer1 works in Capture mode (TM0CL=0, implies CAPT falling edge)

Timer0 and Timer1 are used for Pulse Width and Period Capture

Timer0 and Timer1 can cooperate to measure the signal period and duty cycle time. The key is multi-function of PA6 (CAPT, INT0). Suppose that:

- TM0CKS=0, Timer0 prescaler increases per instruction cycle.
- TM0CM=1, TM1CM=1. Timer0 and Timer1 work in Capture mode
- PA6 pin (CAPT pin) interrupts every falling edge. TM0CL=0, Timer1 starts/holds in turn when PA6 pin (CAPT pin) falling edge is coming. Timer0 starts counting when PA6 pin (CAPT pin) is in logic '1' level, and holds the Timer0 value when PA6 pin (CAPT pin) is in logic '0' level.
- Timer1 is used to measure the signal period, Timer0 is used to measure the PA6 (CAPT pin) in logic '1' time (i.e. the duty cycle of the signal)

The following figure show how to use Timer0 and Timer1 to measure the PA6 (CAPT pin) signal's period and duty cycle (TM0CL=0)



Timer0 and Timer1 are used to measure the signal on CAPT pin

Follow the steps below to start measuring the CAPT pin's period and duty cycle.

Stop the Timer0 by firmware (TM0STP=1, Timer0 will be stopped and hold)

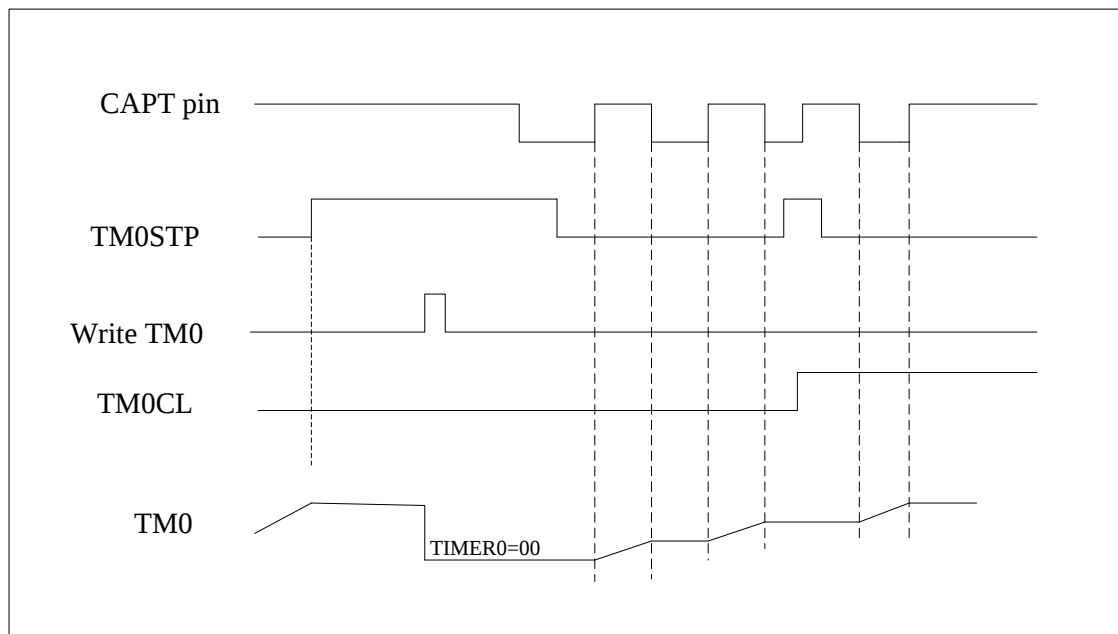
Clear Timer1 by firmware (TM1CLR=1)

Clear Timer0 by directly write 00h to Timer0 (Timer0 is still hold). Once CAPT pin falling edge is coming, the Timer1 starts counting; meanwhile the PA6 interrupt is generated and clears the TM0STP by firmware. Now the Timer0 is ready to count when CAPT pin goes high.

CAPT pin rising edge is coming, Timer0 starts counting until the CAPT pin return to 0 and holds the counting value. Timer1 also stops counting and holds the value.

PA6 interrupt is generated again, firmware stops Timer1 and Timer0 to read the period and duty cycle.

It is not necessary to use both Timer0 and Timer1. If only the duty cycle (CAPT high time) needs to be measured, there is no need to use Timer1 to measure the period. In such case, user can set TM0CM=1 and TM1CM=0. Timer0 is counting up only when CAPT pin is '1'. Note that internal prescaler will be kept to next Timer0 count, so it will not lose the counting accuracy.



Timer0 is used to measure the high (or low) time on CAPT pin

F08	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIE	PWM0IE	PWM1IE	TM1IE	TM0IE	WKTIE	INT2IE	INT1IE	INT0IE
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

F08.5 TM1IE: Timer1 interrupt enable
0: disable
1: enable

F09	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIF	PWM0IF	PWM1IF	TM1IF	TM0IF	WKTIF	INT2IF	INT1IF	INT0IF
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

F09.5 TM1IF: Timer1 interrupt flag
This bit is set by H/W while Timer1 overflows, write 0 to this bit will clear this flag

F14	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TM1L	TM1L							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

F14.7~0 TM1L: Timer1 counter low byte

Read TM1L will get the Timer1 counter low byte. Write TM1L will write the Timer1reload register low byte.

F15	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TM1H	TM1H							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

F15.7~0 TM1H: Timer1 counter high byte

Read TM1L will get the Timer1 counter high byte. Write TM1L will write the Timer1 reload register high byte.

F12	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MF12	PWM0CLR	PWM1CLR	-	-	TM1SET	TM1CLR	TM1STP	TM0STP
R/W	R/W	R/W	-	-	R/W	R/W	R/W	R/W
Reset	0	0	-	-	0	0	0	0

F12.3 TM1SET: Timer1 counter set to 'FFFF'h

0: release Timer1 set

1: set Timer1 to 'FFFF'

F12.2 TM1CLR: Timer1 counter clear

0: release Timer clear

1: clear Timer1 to '0000'h and hold

F12.1 TM1STP: Timer1 counter stop

0: Timer1 is counting

1: Timer1 stop counting

R0C	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MR0C	WKTTPSC		WDTPSC		TM1PSC	TM0OE	TCOE	TM1OE
R/W	W		W		W	W	W	W
Reset	0	0	0	0	0	0	0	0

R0C.3 TM1PSC: Timer1 prescaler

0: Instruction cycle (Fsys/2)

1: System clock (Fsys)

R0C.0 TM1OE: Timer1 overflow toggle output to PD0 enable bit

0: disable output TM1OUT

1: enable output TM1OUT

R0D	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MR0D	INT1EDG	TM1CM	HWAUTO	-	-	ADCLKS		
R/W	W	W	W	-	-	R/W		
Reset	0	0	0	-	-	0	0	0

R0D.6 TM1CM: Timer1 mode selection

0: Timer1 in Timer mode

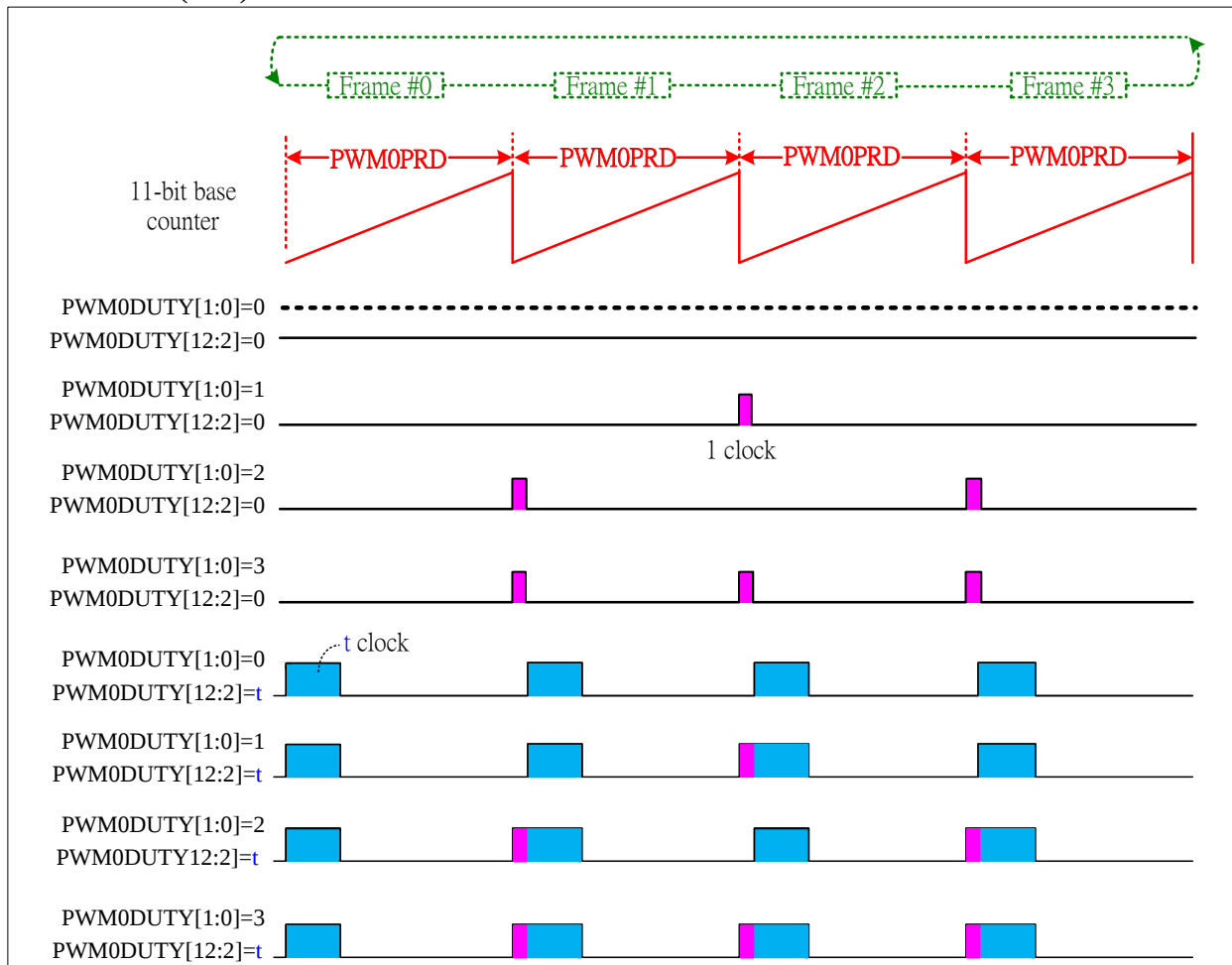
1: Timer1 in Capture mode to measure CAPT pin period time between successive rising or falling edges



The PWM0 can generate fixed frequency waveform with 8192 duty resolution based on System clock (Fsys) or PLL output 108MHz clock. A spread LSB technique allows PWM0 to run its frequency at “PWM0CLK divided by 256” instead of “PWM0CLK divided by 1024”, which means the PWM0 is 4 times faster than normal. The advantage of higher PWM0 frequency is that the post RC filter can transfer the PWM signal to more stable DC voltage level. The 11-bit PWM0 period register (PWM0PRD) is combined by PWM0PRDH (R10.7~0) and PWM0PRDL (R11.2~0). The PWM0 output signal reset to low level whenever the 11-bit base counter matches the 11-bit MSB of PWM0 duty register PWM0DUT1 (R14.7~0) and PWM0DUT2 (R15.7~5, 1~0). When the base counter rolls over, the 2-bit LSB of PWM0 duty register PWM0DUT2 (R15.1~0) decides whether to set the PWM0 output signal high immediately or set it high after one clock cycle delay. By setting PWM0HSHD (F17.3~0) 0FH, PWM0 can drive or sink about 180mA current when PD7 and PA0 in IO Mode2.



Comparator Interrupt can be used to disable the PWM0 output if the CMP2PWMOFF (R1B.0) flag is set to high. In this situation, PWM0N Pin will switch to normal IO Pin (PD7) and PWM0P Pin will switch to normal IO Pin (PA0).



PWM0 Timing Diagram

Example: CPU running at Fast mode, Fsys = CPUCLK = FIRC 12MHz

;setup PWM0 clock prescaler

```

MOVLW    00000000B
MOVWF    MF17                ; PWM0 clock source = Fsys
MOVLW    11001100B          ; PWM0POE=1, PWM0NOE=1
MOVWR    R0B                 ; PWM0 prescaler /8

MOVLW    00000000B          ; PWM0 mode = 00
MOVWR    R0E                 ; PWM0DTS = 00

MOVLW    80H
MOVWR    PWM0PRDH            ; set PWM0PRDH = 80H
MOVLW    04H
MOVWR    PWM0PRDL            ; set PWM0PRDL = 04h
                                ; PWM0PRD = 100_0000_0100B

MOVLW    00000000B
MOVWR    PWM0DUT2            ; set PWM0DUT2 = 00H
    
```

```
MOVLW    80H
MOVWR    PWM0DUT1    ; set PWM0DUT1 = 80H
                        ; PWM0DUTY = 000_1000_0000_00B
BCF      PWM0CLR      ; enable PWM0 counting
```

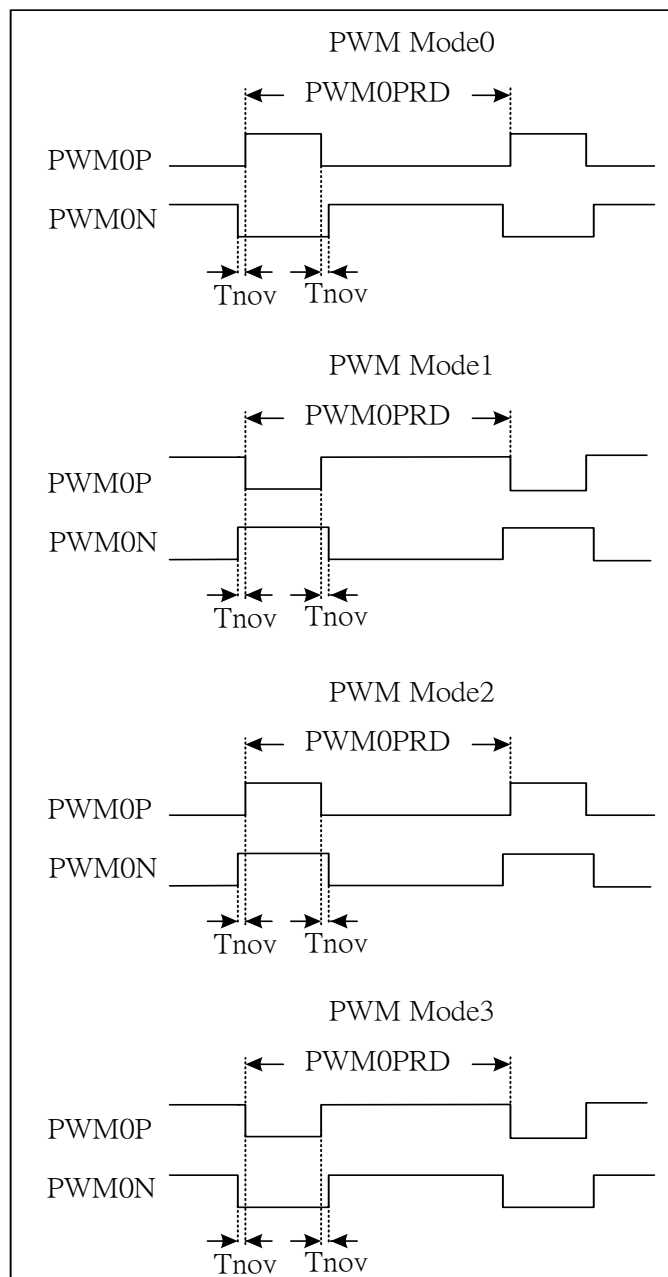
Example:

$F_{sys} = 12\text{MHz}$, $PWMPSC = 8$, $PWM0PRD = 404H$, $PWM0DUTY[12:2] = 080H$

$PWM0 \text{ output frequency} = 12\text{MHz} / 8 / PWM0PRD = 12\text{MHz} / 8 / 1028 = 1459.14\text{Hz}$

$PWM0P \text{ output duty} = 128:1028 = 12.45\%$.

PWM0 can be output via PWM0P and PWM0N with four different modes. The edges of the PWM0 pulse can be separated with 16 different time non-overlap clocks (PWM0 Prescale clocks) intervals which are selected by PWM0DT (R0E.3~0). The default output is PWM Mode0. The waveforms of four PWM output modes are shown below.



PWM0 Output Modes

F08	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIE	PWM0IE	PWM1IE	TM1IE	TM0IE	WKTIE	INT2IE	INT1IE	INT0IE
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

F08.7 PWM0IE: PWM0 interrupt enable

0: disable

1: enable

F09	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIF	PWM0IF	PWM1IF	TM1IF	TM0IF	WKTIF	INT2IF	INT1IF	INT0IF
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

F09.7 PWM0IF: PWM0 interrupt flag

This bit is set by H/W while PWM0 interrupt occurs, write 0 to this bit will clear this flag

R14	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM0DUT1	PWM0DUTY[9:2]							
R/W	R/W							
Reset	0							

R14.7~0 PWM0DUT1: PWM0DUTY[9:2]

R15	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM0DUT2	PWM0DUTY[12:10]			-			PWM0DUTY[1:0]	
R/W	R/W			-			R/W	
Reset	0	0	0	-	-	-	0	0

R15.4~0 PWM0DUT2: PWM0DUTY[12:10], PWM0DUTY[1:0]

F12	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MF12	PWM0CLR	PWM1CLR	-	-	TM1SET	TM1CLR	TM1STP	TM0STP
R/W	R/W	R/W	-	-	R/W	R/W	R/W	R/W
Reset	0	0	-	-	0	0	0	0

F12.7 PWM0CLR: PWM0 clear and hold

0: PWM0 is running

1: PWM0 is clear and hold

F17	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MF17	PWM1CKS		PWM0CKS		PD7HDRV	PD7HSNK	PA0HDRV	PA0HSNK
R/W	R/W		R/W		W	W	W	W
Reset	0		0		0	0	0	0

F17.5~4 PWM0CKS: PWM0 clock selection

00: Fsys as PWM0 clock source

01: 2*Fsys (2 times Fsys frequency) as PWM0 clock source

1x: PLLCKO 108MHz as PWM0 clock source

F17.3 PD7HRRV: Enable PWM0N (PD7) High Drive Current

0: disable PWM0N (PD7) High Drive Current

1: enable PWM0N (PD7) High Drive Current

F17.2 PD7HSNK: Enable PWM0N (PD7) High Sink Current

0: disable PWM0N (PD7) High Sink Current

1: enable PWM0N (PD7) High Sink Current

F17.1 PA0HRRV: Enable PWM0P (PA0) High Drive Current

0: disable PWM0P (PA0) High Drive Current

1: enable PWM0P (PA0) High Drive Current

F17.0 PA0HSNK: Enable PWM0P (PA0) High Sink Current

0: disable PWM0P (PA0) High Sink Current

1: enable PWM0P (PA0) High Sink Current

R0B	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMCTL	PWM0POE	PWM0NOE	PWM1POE	PWM1NOE	PWM0PSC		PWM1PSC	
R/W	R/W	R/W	R/W	R/W	R/W		R/W	
Reset	0	0	0	0	0	0	0	0

R0B.7 PWM0POE: PWM0P output enable
 0: disable PWM0P output to PA0
 1: enable PWM0P output to PA0

R0B.6 PWM0NOE: PWM0N output enable
 0: disable PWM0N output to PD7
 1: enable PWM0N output to PD7

R0B.3~2 PWM0PSC: PWM 0 clock source is divided by
 00: 1
 01: 2
 10: 4
 11: 8

R10	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM0PRDH	PWM0PRDH							
R/W	W							
Reset	0	0	0	0	0	0	0	0

R10.7~0 PWM0PRDH: PWM0 period data 8bit MSB

R11	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM0PRDL	PWM1PRDL			-	-	PWM0PRDL		
R/W	W			-	-	W		
Reset	0	0	0	-	-	0	0	0

R11.2~0 PWM0PRDL: PWM0 period data 3bit MSB

R0E	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MR0E	CMPTRIG		PWM0MODE		PWM0DT			
R/W	W		W		W			
Reset	0	0	0	0	0	0	0	0

R0E.5~4 PWM0MODE: PWM0P and PWM0N output mode
 00: Mode 0
 01: Mode 1
 10: Mode 2
 11: Mode 3

R0E.3~0 PWM0DT: PWM0 dead time selection
 0000: 0 pwm0 prescale clock (original PWM0)
 0001: 1 pwm0 prescale clock
 0010: 2 pwm0 prescale clock
 0011: 3 pwm0 prescale clock
 0100: 4 pwm0 prescale clock
 0101: 5 pwm0 prescale clock

0110: 6 pwm0 prescale clock
 0111: 7 pwm0 prescale clock
 1000: 8 pwm0 prescale clock
 1001: 9 pwm0 prescale clock
 1010: 10 pwm0 prescale clock
 1011: 11 pwm0 prescale clock
 1100: 12 pwm0 prescale clock
 1101: 13 pwm0 prescale clock
 1110: 14 pwm0 prescale clock
 1111: 16 pwm0 prescale clock

R06	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PAMODL	PA3MOD		PA2MOD		PA1MOD		PA0MOD	
R/W	W		W		W		W	
Reset	0	1	0	1	0	1	0	1

R06.1~0 PA0MOD:PA0 pin mode control
 00: Mode0
 01: Mode1
 10: Mode2
 11: Mode3

R09	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PDMODH	PD7MOD		PD6MOD		PD5MOD		PD4MOD	
R/W	W		W		W		W	
Reset	0	1	0	1	0	1	0	1

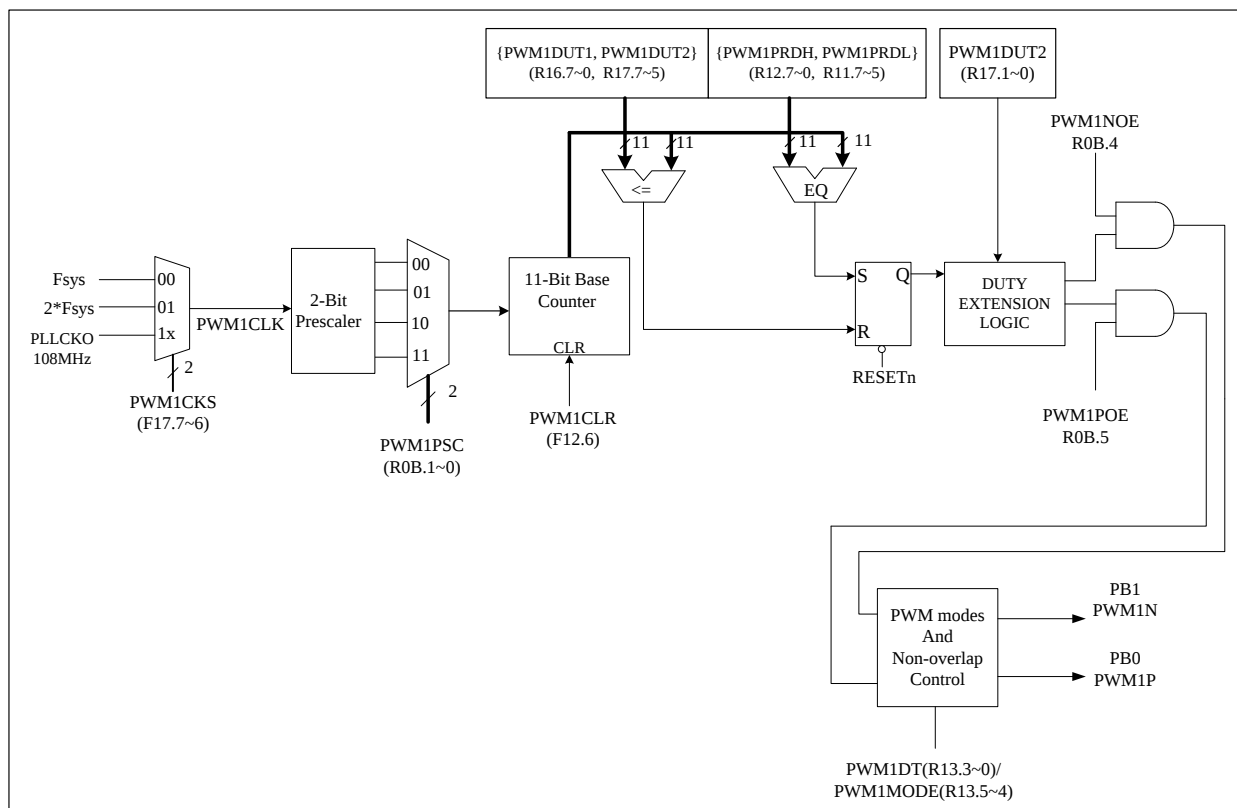
R09.7~6 PD7MOD:PD7 pin mode control
 00: Mode0
 01: Mode1
 10: Mode2
 11: Mode3

R1B	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CMPISSET	-	-	-	-	-	SOPPL	CMPINV	CMP2PWMOFF
R/W	-	-	-	-	-	W	W	W
Reset	-	-	-	-	-	0	0	0

R1B.0 CMP2PWMOFF: enable Comparator Interrupt to clear PWM output
 0: disable
 1: enable

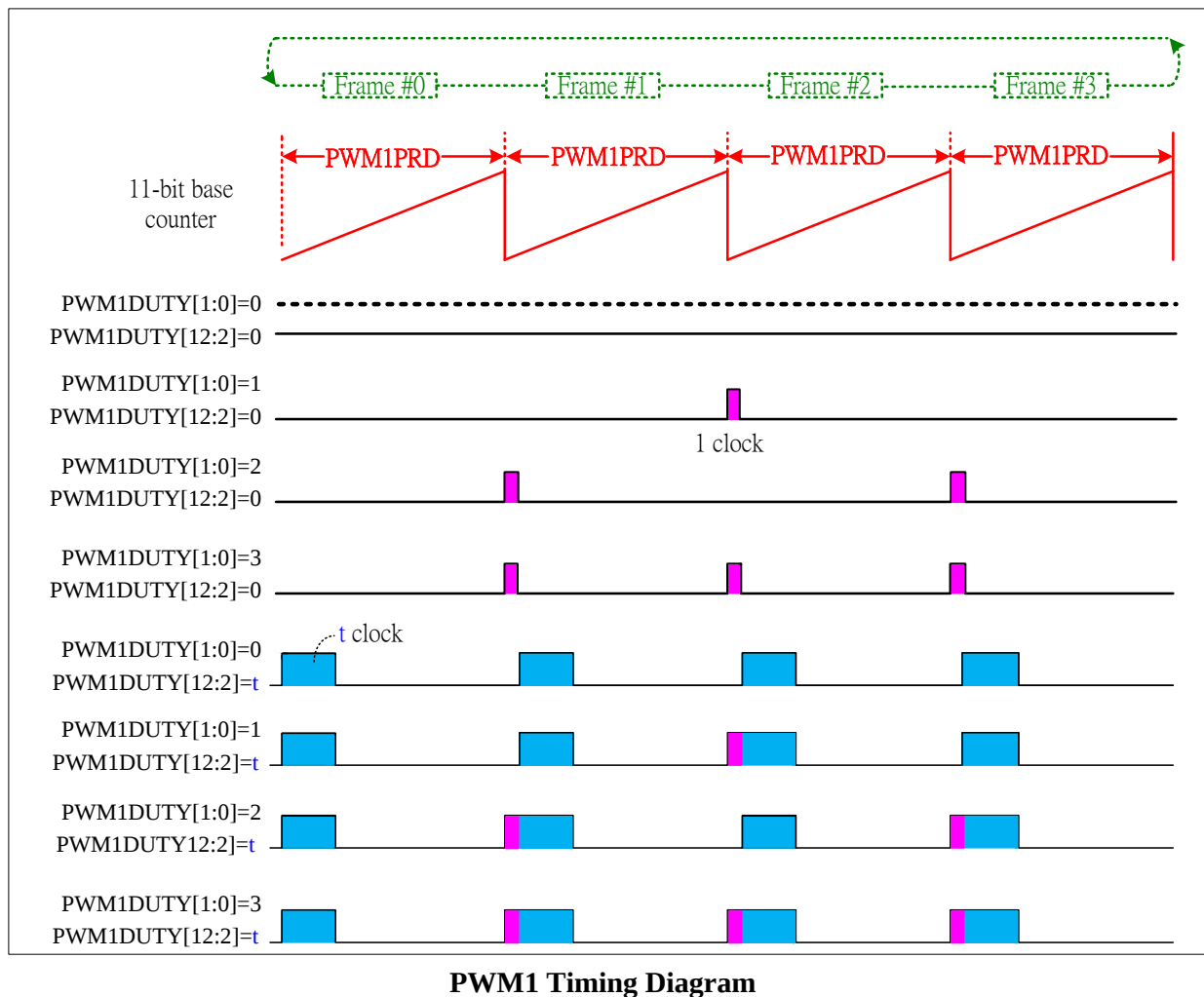
3.5 PWM1: (11+2) bits PWM

The PWM1 can generate fix frequency waveform with 8192 duty resolution based on System clock (Fsys) or PLL output 108MHz clock. A spread LSB technique allows PWM1 to run its frequency at “PWM1CLK divided by 256” instead of “PWM1CLK divided by 1024”, which means the PWM1 is 4 times faster than normal. The advantage of higher PWM1 frequency is that the post RC filter can transfer the PWM1 signal to more stable DC voltage level. The 11-bit PWM1 period register (PWM1PRD) is combined by PWM1PRDH (R12.7~0) and PWM1PRDL (R11.7~5). The PWM1 output signal reset to low level whenever the 11-bit base counter matches the 11-bit MSB of PWM1 duty register PWM1DUT1 (R16.7~0) and PWM1DUT2 (R17.7~5,1~0). When the base counter rolls over, the 2-bit LSB of PWM1 duty register PWM1DUT2 (R17.1~0) decides whether to set the PWM1 output signal high immediately or set it high after one clock cycle delay.



PWM1 Block Diagram

Comparator Interrupt can be used to disable the PWM1 output if the CMP2PWMOFF (R1B.0) flag is set to high. In this situation PWM1N Pin will switch to normal IO Pin (PB1) and PWM1P Pin will switch to normal IO Pin (PB0).



Example: CPU running at Fast mode, Fsys = CPUCLK = FIRC 12MHz

;setup PWM1 clock prescaler

```

MOVLW    00000000B
MOVWF    F17                ; PWM1 clock source = Fsys
MOVLW    00110011B         ; PWM1POE=1, PWM1NOE=1
MOVWR    R0B               ; PWM1 prescaler /8

MOVLW    00000000B         ; PWM1 mode = 00
MOVWR    R13               ; PWM1DTS = 00

MOVLW    80H
MOVWR    PWM1PRDH          ; set PWM1PRDH = 80H
MOVLW    80H
MOVWR    PWM1PRDL          ; set PWM1PRDL = 80h
                          ; PWM1PRD = 100_0000_0100B

MOVLW    00000000B         ; set PWM1DUT2 = 00H
MOVWR    PWM1DUT2
MOVLW    80H
MOVWR    PWM1DUT1          ; set PWM1DUT1 = 80H
                          ; PWM1DUTY = 000_1000_0000_00B
BCF      PWM1CLR           ; enable PWM1 counting
    
```

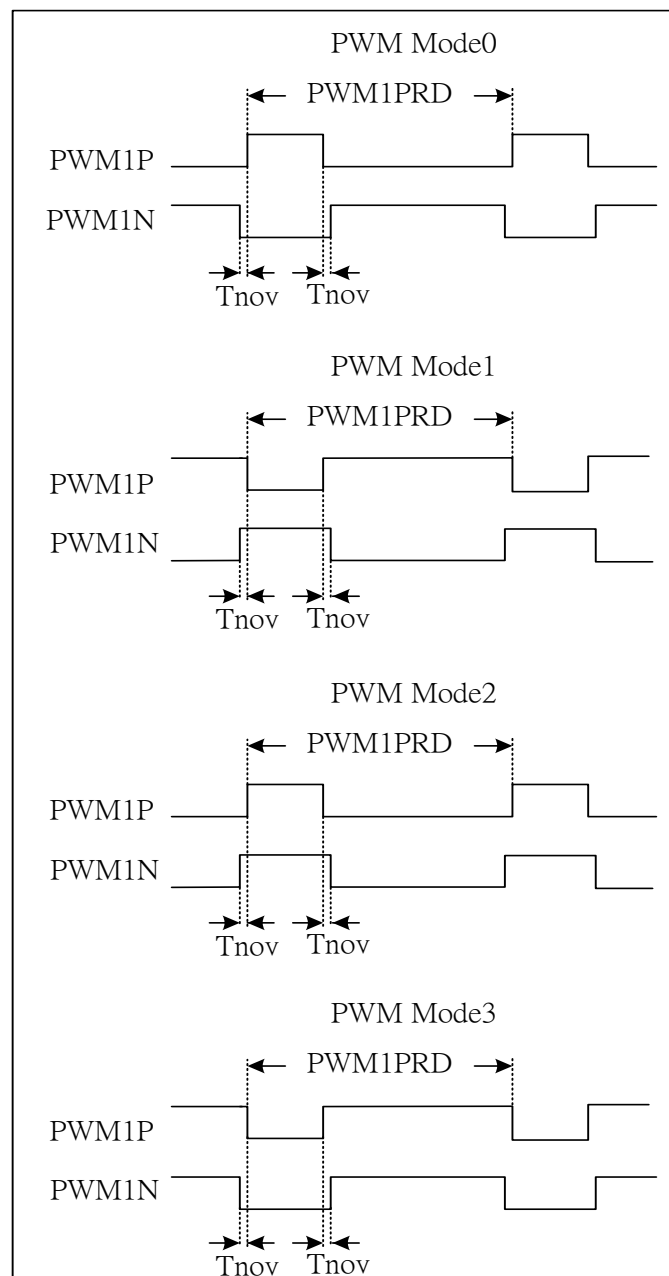
Example:

$F_{sys} = 12\text{MHz}$, $PWM1PSC = 8$, $PWM1PRD = 404H$, $PWM1DUTY[12:2] = 080H$

$PWM1$ output frequency = $12\text{MHz} / 8 / PWM1PRD = 12\text{MHz} / 8 / 1028 = 1459.14\text{Hz}$

$PWM1P$ output duty = $128:1028 = 12.45\%$.

$PWM1$ can be output via $PWM1P$ and $PWM1N$ with four different modes. The edges of the $PWM1$ pulse can be separated with 16 different time non-overlap clocks ($PWM1$ Prescale clocks) intervals which are selected by $PWM1DT$ (R13.3~0). The default output is PWM Mode0. The waveforms of four output modes are shown below.



PWM1 Output Modes

F08	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIE	PWM0IE	PWM1IE	TM1IE	TM0IE	WKTIE	INT2IE	INT1IE	INT0IE
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

F08.6 PWM1IE: PWM1 interrupt enable

0: disable

1: enable

F09	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIF	PWM0IF	PWM1IF	TM1IF	TM0IF	WKTIF	INT2IF	INT1IF	INT0IF
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

F09.6 PWM1IF: PWM1 interrupt flag

This bit is set by H/W while PWM1 interrupt occurs, write 0 to this bit will clear this flag

R16	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1DUT1	PWM1DUTY[9:2]							
R/W	R/W							
Reset	0							

R16.7~0 PWM1DUT1: PWM1DUTY[9:2]

R17	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM0DUT2	PWM1DUTY[12:10]			-	-	-	PWM1DUTY[1:0]	
R/W	R/W			-	-	-	R/W	
Reset	0	0	0	-	-	-	0	0

R17.4~0 PWM1DUT2: PWM1DUTY[12:10], PWM1DUTY[1:0]

F12	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MF12	PWM0CLR	PWM1CLR	-	-	TM1SET	TM1CLR	TM1STP	TM0STP
R/W	R/W	R/W	-	-	R/W	R/W	R/W	R/W
Reset	0	0	-	-	0	0	0	0

F12.6 PWM1CLR: PWM1 clear and hold

0: PWM1 is running

1: PWM1 is clear and hold

F17	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MF17	PWM1CKS		PWM0CKS		PD7HDRV	PD7HSNK	PA0HDRV	PA0HSNK
R/W	R/W		R/W		W	W	W	W
Reset	0		0		0	0	0	0

F17.7~6 PWM1CKS: PWM1 clock selection

00: Fsys as PWM1 clock source

01: 2*Fsys (2 times Fsys frequency) as PWM1 clock source

1x: PLLCKO 108MHz as PWM1 clock source

R0B	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMCTL	PWM0POE	PWM0NOE	PWM1POE	PWM1NOE	PWM0PSC		PWM1PSC	
R/W	R/W	R/W	R/W	R/W	R/W		R/W	
Reset	0	0	0	0	0	0	0	0

R0B.5 PWM1POE: PWM1P output enable
 0: disable PWM1P output to PB0
 1: enable PWM1P output to PB0

R0B.4 PWM1NOE: PWM1N output enable
 0: disable PWM1N output to PB1
 1: enable PWM1N output to PB1

R0B.1~0 PWM1PSC: PWM1 clock source is divided by
 00: 1
 01: 2
 10: 4
 11: 8

R12	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1PRDH	PWM1PRDH							
R/W	W							
Reset	0	0	0	0	0	0	0	0

R12.7~0 PWM1PRDH: PWM1 period data 8bit MSB

R11	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1PRDL	PWM1PRDL			-	-	PWM0PRDL		
R/W	W			-	-	W		
Reset	0	0	0	-	-	0	0	0

R11.7~5 PWM1PRDL: PWM1 period data 3bit MSB

R13	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MR13	CMPDBS		PWM1MODE		PWM1DT			
R/W	W		W		W			
Reset	0	0	0	0	0	0	0	0

R13.5~4 PWM1MODE: PWM1P and PWM1N output mode
 00: Mode 0
 01: Mode 1
 10: Mode 2
 11: Mode 3

R13.3~0 PWM1DT: PWM1Pdead time selection
 0000: 0 pwm1 prescale clock (original PWM1)
 0001: 1 pwm1 prescale clock
 0010: 2 pwm1 prescale clock
 0011: 3 pwm1 prescale clock
 0100: 4 pwm1 prescale clock
 0101: 5 pwm1 prescale clock
 0110: 6 pwm1 prescale clock
 0111: 7 pwm1 prescale clock
 1000: 8 pwm1 prescale clock

1001: 9 pwm1 prescale clock
 1010: 10 pwm1 prescale clock
 1011: 11 pwm1 prescale clock
 1100: 12 pwm1 prescale clock
 1101: 13 pwm1 prescale clock
 1110: 14 pwm1 prescale clock
 1111: 16 pwm1 prescale clock

R08	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PBMODL	PB3MOD		PB2MOD		PB1MOD		PB0MOD	
R/W	W		W		W		W	
Reset	0	1	0	1	0	1	0	1

R08.3~2 PB1MOD:PB1 pin mode control
 00: Mode0
 01: Mode1
 10: Mode2
 11: Mode3

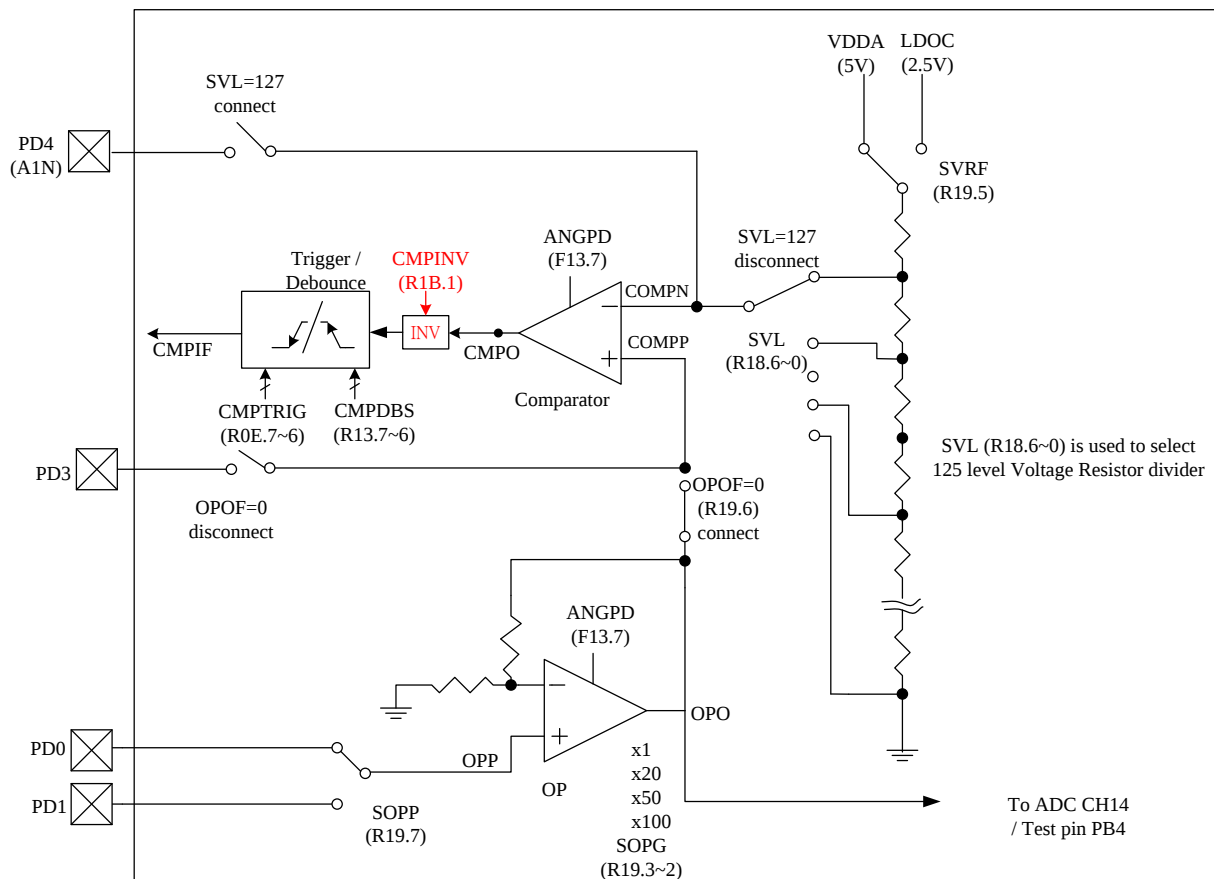
R08.1~0 PB0MOD:PB0 pin mode control
 00: Mode0
 01: Mode1
 10: Mode2
 11: Mode3

R1B	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CMPISSET	-	-	-	-	-	SOPPL	CMPINV	CMP2PWMOFF
R/W	-	-	-	-	-	W	W	W
Reset	-	-	-	-	-	0	0	0

R1B.0 CMP2PWMOFF: enable Comparator Interrupt to clear PWM output
 0: disable
 1: enable

3.6 Operational Amplifier and Comparator

There is an operational amplifier and a comparator in this device. By setting the ANGPD=1 (F13.7), the OPA and Comparator enter the power down mode. The SOPP (R19.7) register determined the OPA input signal (OPP) is PD0 or PD1. When SOPP=0 the OPP is PD0, otherwise OPP is PD1. The 4-level OP gain (1x/10x/50x/100x) is controlled by SOPG (R19.3~2). User can select the divider resistance voltage source come from VDDA (5V) or LDOC (2.5V) by setting SVRF (R19.5). The Comparator output (CMPO) is used to generate Interrupt (CMPIF). CMPTRIG (R0E.7~6) is used to determine which edge of CMPO can generate interrupt. And CMPDBS (R13.7~6) is used to determine the Comparator output debounce time. The Comparator's hysteresis is about 40mV. When CMPINV (R1B.1) is set to high, the CMPO will invert to the opposite signal.

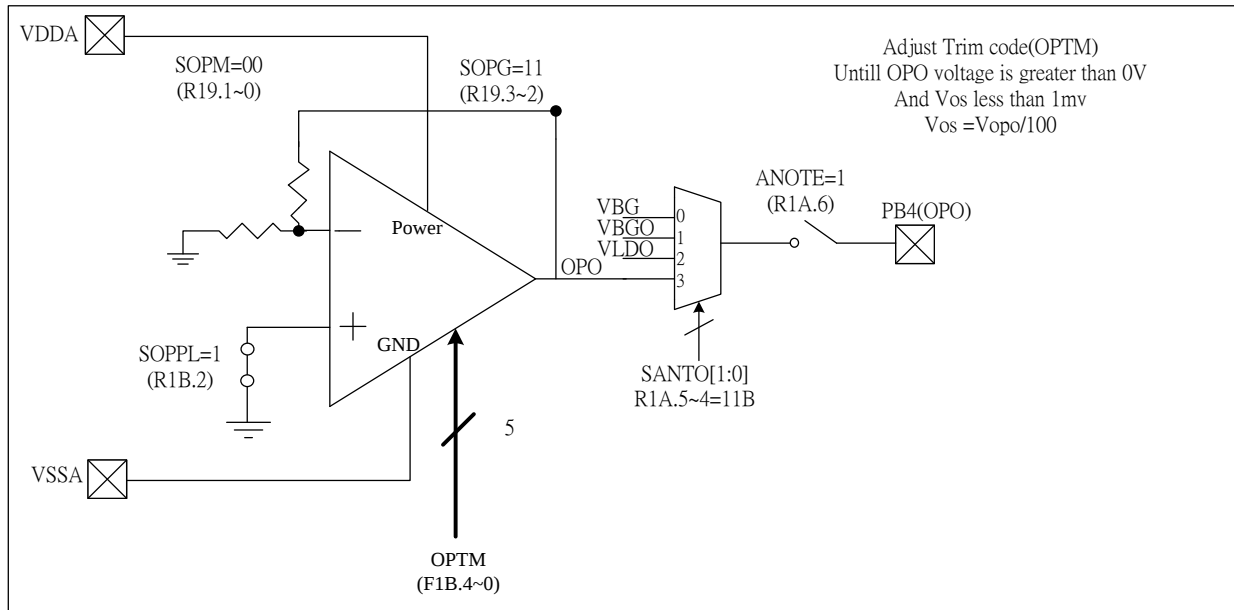


OPA / Comparators Block Diagram

In addition, the OPA support the build-in offset calibration mode by setting the SOPM (R19.1~0) value to 01B, SOTV (R19.4) to high, ANTOE (R1A.6) to high and SANTO (R1A.5~4) to 11B. In the OPA offset calibration mode, user has to change the OPTM (F1B.4~0) value from 00H to 1FH in turn and record the OPTM value when OPO output low into high. Similarity, changing the OPTM from 1FH to 00H in turn, we can get another value when OPO output high into low. According to the two values, the suitable OPTM can be choice for calibration. With I/O mode setting, the corresponding pins have to set as analog mode (Mode3). It can disable the pin logical input path for save power consumption.

There are three kinds of Trim mode to calibrate the OP offset. Trim mode 3 can be used by user and Trim mode 1/2 is used for manufactory tester.

When user use Trim mode 3, F/W must set SOPPL(R1B.2)=1 to pull low the OPA input to ground, SOPM(R19.1~0)=00, SANTO(R1A.5~4)=11, ANOTE(R1A.6)=1, change OPTM(F1B.4~0) value until PB4(OPO) voltage is greater than 0V and Vos is less than 1mv. ($V_{os} = V_{opo}/100$). User can use Trim mode 3 and ADC CH14 and record the ADC output value (OPA_{OFF}). For normal OPA application, use normal mode combine ADC CH14 to get the OPO to ADC output value (OPA_{AD}). The most accuracy value of OPA combine ADC will be OPA_{AD} minus OPA_{OFF} .



OPA Trim Mode3

F13	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MF12	ANGPD	PLLPD	-	IVCPD	LDOPD	PUMP	VCCFLT	CLKFLT
R/W	R/W	R/W	-	R/W	R/W	R/W	R/W	R/W
Reset	1	1	-	0	0	0	0	0

F13.7 ANGPD: OPA and Comparator power down enable control

0: disable OPA and Comparator power down

1: enable OPA and Comparator power down

R18	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ANA_CTL1	PDBGOP	SVL						
R/W	R/W	W						
Reset	1	0						

R18.6~0 SVL: reference source (VDDA or LDOC) voltage level, total 125 levels

0: 1/125 voltage source

1: 2/125 voltage source

...

124: 125/125 voltage source

127: disconnect to COMPN

others: reserved

R19	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ANA_CTL2	SOPP	OPOF	SVRF	SOTV	SOPG		SOPM	
R/W	W	W	W	W	W		W	
Reset	0	0	0	0	1	1	0	0

R19.7 SOPP: select OPP input source
 0: OPP input source is PD0
 1: OPP input source is PD1

R19.6 OPOF: OPA output (OPO) connect to Comparator control
 0: OPO connect to COMPP
 1: OPO disconnect to COMPP

R19.5 SVRF: select Comparator reference voltage source
 0: VDDA
 1: LDOC

R19.4 SOTV: select reference voltage source for OPA trim offset
 0: AVSS
 1: LDOC

R19.3~2 SOPG: select OPA gain
 00: 1X
 01: 20X
 10: 50X
 11: 100X

R19.1~0 SOPM: select OPA working mode
 00: normal mode
 01: Trim 1 mode
 10: reserved
 11: reserved

R0E	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MR0E	CMPTRIG		PWM0MODE		PWM0DT			
R/W	W		W		W			
Reset	0	0	0	0	0	0	0	0

R0E.7~6 CMPTRIG: Comparator interrupts trigger direction
 00: rising edge trigger (When R1B.1=0)
 01: falling edge trigger (When R1B.1=0)
 10: both edge trigger (When R1B.1=0)
 11: high level trigger (When R1B.1=0)

R13	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MR13	CMPDBS		PWM1MODE		PWM1DT			
R/W	W		W		W			
Reset	0	0	0	0	0	0	0	0

R13.7~6 CMPDBS: Comparator output debounce time
 00: None
 01: 4 Fsys
 10: 8 Fsys
 11: 16 Fsys

R1A	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ANA_TST	-	ANTOE	SANTO		-	-	-	-
R/W	-	W	W		-	-	-	-
Reset	-	0	0	0	-	-	-	-

R1A.6 ANTOE: Analog signal to IO (PB4) enable
 0: disable analog signal to PB4
 1: enable analog signal to PB4

R1A.5~4 SANTO: select analog signal to output pad
 00: VBG
 01: VBGO
 10: LDOC
 11: OPO

F0A	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIE1	-	-	-	-	-	-	-	CMPIE
R/W	-	-	-	-	-	-	-	R/W
Reset	-	-	-	-	-	-	-	0

F0A.0 CMPIE: Comparator interrupt enable
 0: disable
 1: enable

F0B	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIF1	-	-	-	-	-	-	-	CMPIF
R/W	-	-	-	-	-	-	-	R/W
Reset	-	-	-	-	-	-	-	0

F0B.7 CMPIF: Comparator interrupt flag
 This bit is set by H/W while compare a new data, write 0 to this bit will clear this flag

R07	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PBMODH	-	-	-	-	PB5MOD		PB4MOD	
R/W	-	-	-	-	W		W	
Reset	-	-	-	-	0	1	0	1

R07.1~0 PB4MOD: PB4 Pin Mode Control
 00: Mode0
 01: Mode1
 10: Mode2
 11: Mode3, analog signal

F1B	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OPTM	-	-	-	OPTM				
R/W	-	-	-	R/W				
Reset	-	-	-	by SYSTEM setting				

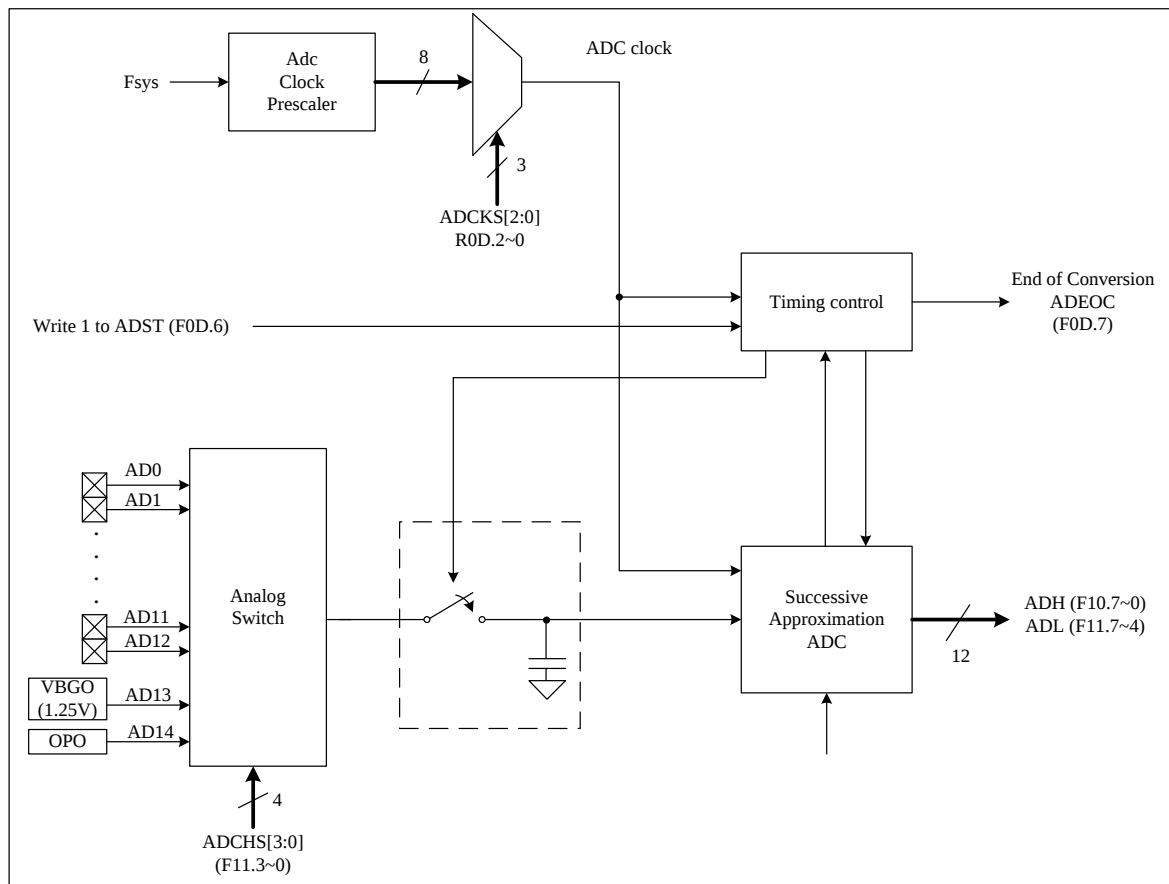
F1B.4~0 OPTM: OPA Trim value (When power on, this value is loaded from system config. 08H)

R1B	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CMPISSET	-	-	-	-	-	SOPPL	CMPINV	CMP2PWMOF
R/W	-	-	-	-	-	W	W	W
Reset	-	-	-	-	-	0	0	0

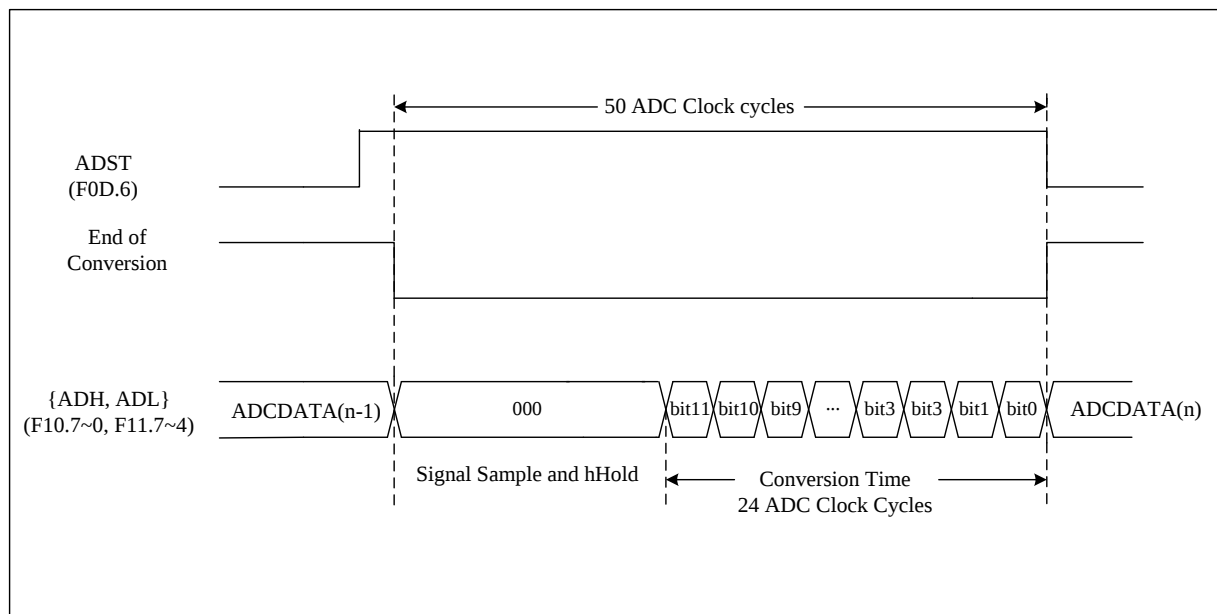
R1B.2 SOPPL: OPP input pull low enable
0: disable
1: enable

R1B.1 CMPINV: enable Comparator output (CMPO) invert to opposite signal
0: disable
1: enable

3.7 Analog-to-Digital Convert



The 12-bit ADC (Analog to Digital Convert) consists of a 15-channel analog input multiplexer, control register, clock generator, 12-bit successive approximation register and output data register. To use the ADC, user needs to set $ADCKS$ (R0D.2~0) to choose a proper ADC clock frequency. User then launches the ADC conversion by setting the $ADST$ (F0D.6) control bit. After end of conversion, H/W generates $ADEOC$ (F0D.7) flag and automatically clear $ADST$. User can poll $ADST$ or $ADEOC$ to know conversion status. The $PAMODH$ (R05.5~0), $PAMODL$ (R06.7~0), $PBMODH$ (R07.7~0), $PBMODL$ (R08.7~0) and $PDMODH$ (R09.7~0) control registers are used for ADC pin type setting. By setting those pin mode to Mode3 can disable the pin logical input path to save power consumption. User need to set $ADCHS$ (F11.3~0) to choose the input channel of ADC. One of them, $AD13$ is $VBGO$ (1.25V) input for ADC. For TM57MA45, the ADC $VREF$ input is $LDOC$ (2.5V). For TM57MA46, the ADC $VREF$ input is VCC (5V).



ADC Conversion Timing Diagram

Example:

TM57MA45: [CPU running at Fast mode, Fsys = FIRC 12MHz]
 ADC clock frequency = 750KHz, ADC channel = ADC5 (PB1),

3. Example:

```

MOVLW    xxxxx100B    ; Fsys = 12MHz
MOVWR    R0D          ; ADC clock prescaler /16

MOVLW    xxxx11xxB
MOVWR    PBMODL       ; set PB1 ADC pin type

MOVLW    xxxx0101B
MOVWF    F11          ; ADC channel select ADC5 (PB1)

BSF      ADST         ; ADC start conversion
  
```

WAIT_ADC:

```

    BTFSC    ADST      ; wait ADC conversion
    GOTO     WAIT_ADC
  
```

```

    MOVFW    ADH        ; read ADC data[11:4]
    MOVWF    ADC_MSB
    MOVFW    F11        ; read ADC data[3:0]
    ANDLW    F0H
    MOVWF    ADC_LSB
  
```

F0D	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MF0D	ADEOC	ADST	PWRSVAV		-	-	-	-
R/W	R	R/W	R/W		-	-	-	-
Reset	-	0	1	1	-	-	-	-

F0D.7 ADEOC: ADC end of conversion bit

F0D.6 ADST: ADC start conversion enable bit

F10	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADH	ADH							
R/W	R							
Reset	-	-	-	-	-	-	-	-

F10.7~0 ADH: ADC output data MSB [11:4]

F11	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MF11	ADL				ADCHS			
R/W	R				R/W			
Reset	-	-	-	-	0	0	0	0

F11.7~4 ADL: ADC output data LSB [3:0]

F11.3~0 ADCHS: ADC channel select

0000: ADC0 (PA6)	0001: ADC1 (PA1)	0010: ADC2 (PA2)
0011: ADC3 (PA0)	0100: ADC4 (PD7)	0101: ADC5 (PB1)
0110: ADC6 (PB0)	0111: ADC7 (PA5)	1000: ADC8 (PD6)
1001: ADC9 (PB5)	1010: ADC10 (PB4)	1011: ADC11 (PB3)
1100: ADC12 (PB2)	1101: ADC13 (VBGO)	1100: ADC14 (OPO)

R0D	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MR0D	INT1EDG	TM1CM	HWAUTO	-	-	ADCKS		
R/W	R/W	R/W	R/W	-	-	R/W		
Reset	0	0	0	-	-	0	0	0

R0D.2~0 ADCKS: ADC clock frequency selection

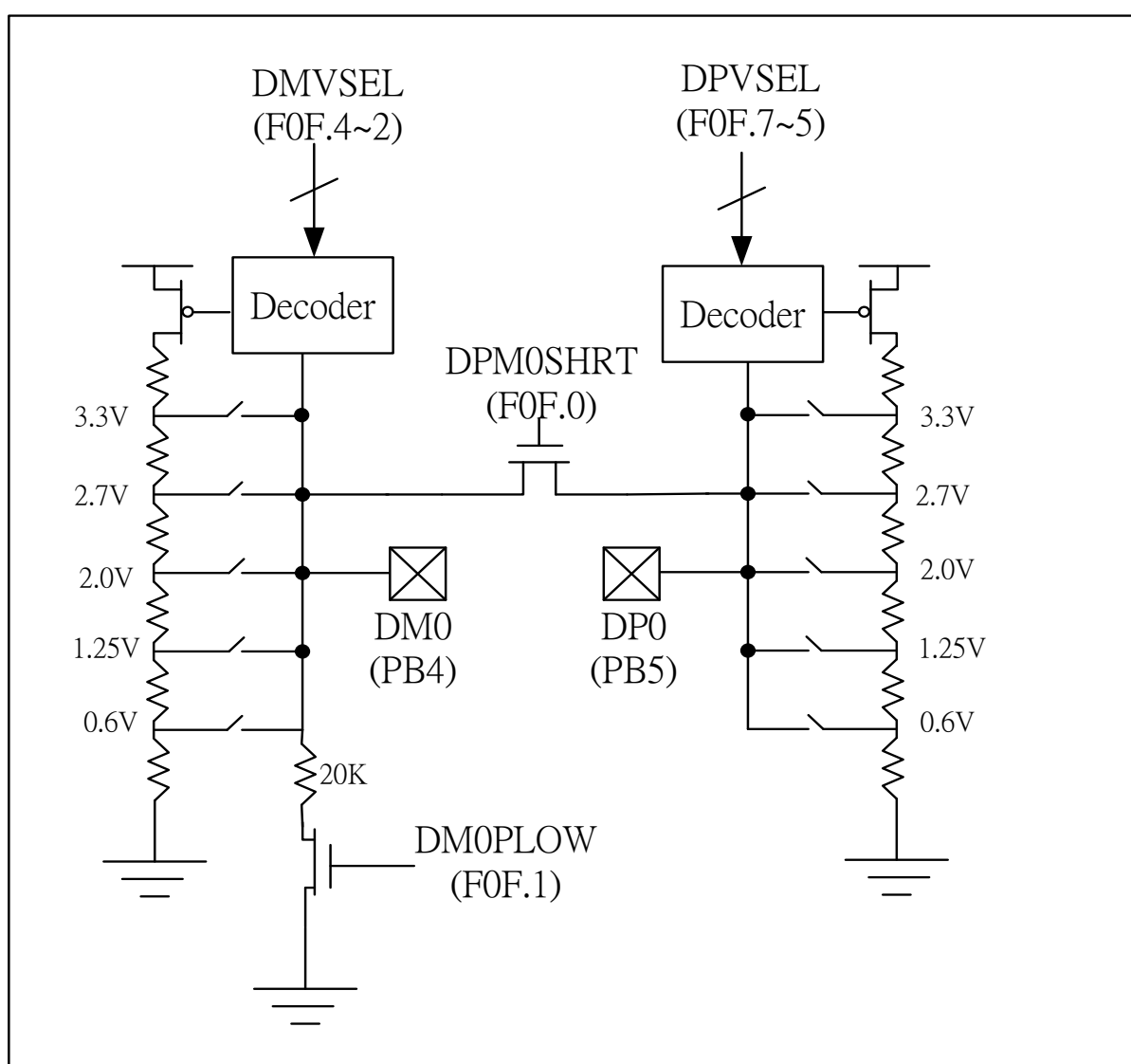
000: Fsys/256
001: Fsys/128
010: Fsys/64
011: Fsys/32
100: Fsys/16
101: Fsys/8
110: Fsys/4
111: Fsys/2

3.8 USB Charging Control

TM57MA45/MA46 have USB ports named DM0/DP0, DM1/DP1, DM2/DP2 and CC1/CC2 to implement the USB battery charge function. This device can support a high performance fast charging solution following Quick Charge 2.0 High Voltage Dedicated Charging Port (HVDCP) specification. With I/O mode setting, the corresponding pins have to set as analog mode (Mode3). It can disable the pin logical input path for save power consumption.

3.8.1 DPDMV (DM0/DP0)

The DPDMV is a voltage switch that can output different voltage 3.3V/2.7V/2.0V/1.25V/0.6V to DP0 and DM0 by setting DP0VSEL (F0F.7~5) and DM0VSEL (F0F.4~2). Beside, DP0 and DM0 can be as internal short by setting DPM0SHRT (F0F.0). The DM0 can assign a pull-low resistor by setting DM0PLOW (F0F.1) register. The DPDMV block diagram is shown as below.



DPDMV Block Diagram

F0F	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DPM0CTL	DP0VSEL			DM0VSEL			DM0PLOW	DPM0SHRT
R/W	R/W			R/W			R/W	R/W
Reset	1	1	1	1	1	1	0	0

F0F.7~5 DP0VSEL: DP0 (PB5) output voltage select

000: 0.6V
 001: 1.25V
 010: 2.0V
 011: 2.7V
 100: 3.3V
 101/110: float
 111: disable

F0F.4~2 DM0VSEL: DM0 (PB4) output voltage select

000: 0.6V
 001: 1.25V
 010: 2.0V
 011: 2.7V
 100: 3.3V
 101/110: float
 111/110: disable

F0F.1 DM0PLOW: DM0 pull low

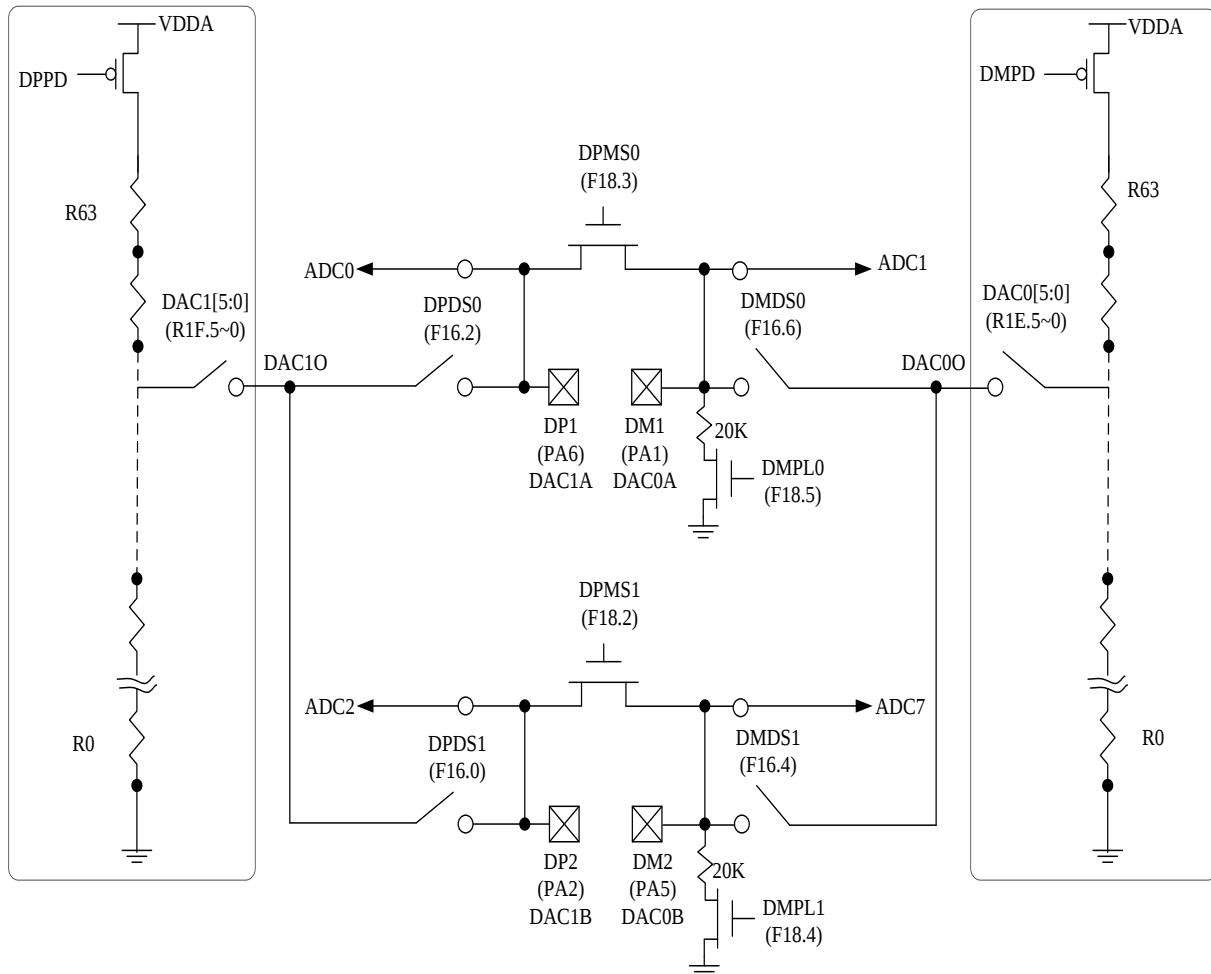
0: disable
 1: enable

F0F.0 DPM0SHRT: DP0/DM0 short inside

0: DP0/DM0 not short
 1: DP0/DM0 short together

3.8.2 DPDM6B (DM1/DM2 and DP1/DP2)

The DPDM6B module contains two 6-bit DAC which generate 64 level voltage to DM1/DM2 and DP1/DP2 by setting DAC0D (R1E.5~0) and DAC1D (R1F.5~0). The DM1, DM2, DP1 and DP2 are pin shared with normal I/O function, A/D function and D/A function determined by the DPDS0 (F16.2), DPDS1 (F16.0), DMDS0 (F16.6) and DMDS1 (F16.4) register. DP1 and DM1 can be as internal short by setting DPMS0 (F18.3). DP2 and DM2 can be as internal short by setting DPMS1 (F18.2). The DM1 can assign a pull-low resistor by setting DMPL0 (F18.5) and the DM2 can assign a pull-low resistor by setting DMPL1 (F18.4). These two DAC can be disabled individually by setting DPPD (F18.0) and DMPD (F18.1) to high. The DPDM6B block diagram is shown as below.



DPDM6B Block Diagram

F16	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DACCTL0	-	DMDS0	-	DMDS1	-	DPDS0	-	DPDS1
R/W	-	R/W	-	R/W	-	R/W	-	R/W
Reset	-	0	-	0	-	0	-	0

F16.6 DMDS0: DM1 signal switch

0: open

1: DAC0O/DM1 short

F16.4 DMDS1: DM2 signal switch

0: open

1: DAC0O/DM2 short

F16.2 DPDS0: DP1 signal switch

0: open

1: DAC1O/DP1 short

F16.0 DPDS1: DP2 signal switch

0: open

1: DAC1O/DP2 short

F18	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DACCTL1	-	-	DMPL0	DMPL1	DPMS0	DPMS1	DMPD	DPPD
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	-	0	0	0	0	1	1

F18.5 DMPL0: DM1 pull low enable bit

0: disable

1: enable

F18.4 DMPL1: DM2 pull low enable bit

0: disable

1: enable

F18.3 DPMS0: DP1/DM1 short enable bit

0: disable

1: enable

F18.2 DPMS1: DP2/DM2 short enable bit

0: disable

1: enable

F18.1 DMPD: DAC0 power down enable bit

0: disable

1: enable

F18.0 DPPD: DAC1 power down enable bit

0: disable

1: enable

R1E	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DAC0D	-	-	DAC0D					
R/W	-	-	R/W					
Reset	-	-	0	0	0	0	0	0

R1E.5~0 DAC0D: DAC0 voltage selects; total 64 level

00000: 1/64 VDDA

00001: 2/64 VDDA

00010: 3/64 VDDA

.....

11111: 64/64 VDDA

R1F	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DAC1D	-	-	DAC1D					
R/W	-	-	R/W					
Reset	-	-	0	0	0	0	0	0

R1F.5~0 DAC1D: DAC1 voltage selects; total 64 level

00000: 1/64 VDDA

00001: 2/64 VDDA

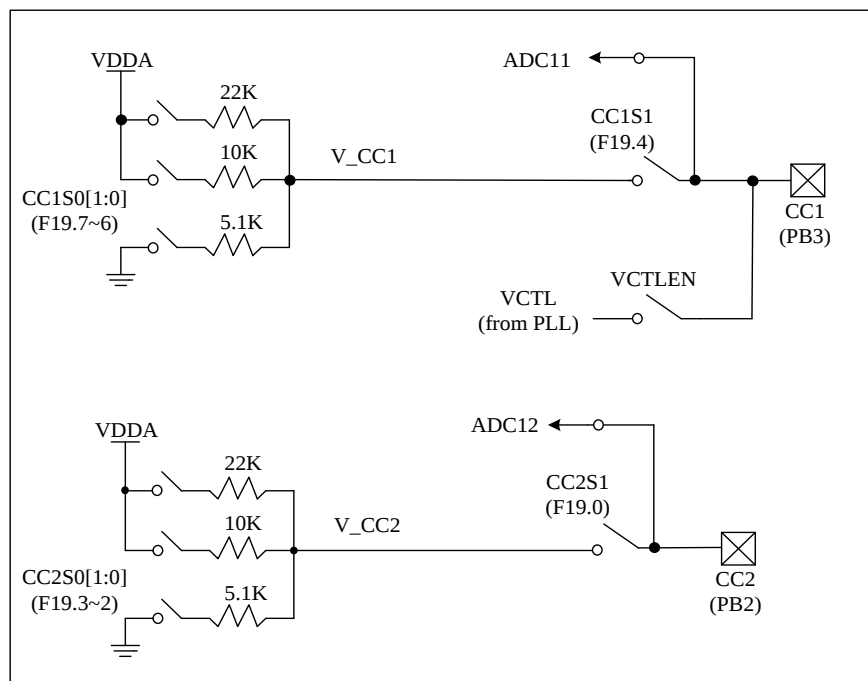
00010: 3/64 VDDA

.....

11111: 64/64 VDDA

3.8.3 CC1/CC2

CC1/CC2 ports are used for USB type C connect detection. The CC12 module can pull up the CC1/CC2 voltage level to VDDA or pull down to Ground by setting CC1S0 (F19.7~6)/CC2S0 (F19.3~2) register. The voltage of CC1/CC2 can be read by using ADC11/ADC12 when CC1S1 (F19.4)/CC2S1 (F19.0) is set to "1".



CC12 Block Diagram

F19	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CC12SEL	CC1S0		-	CC1S1	CC2S0		-	CC2S1
R/W	R/W		-	R/W	R/W		-	R/W
Reset	0	0	-	0	0	0	-	0

F19.7~6 CC1S0: CC1 voltage level select
 00: 5.1K Ohm resister to ground
 01: 10K Ohm resister to VDDA
 10: 22K Ohm resister to VDDA
 11: reserved

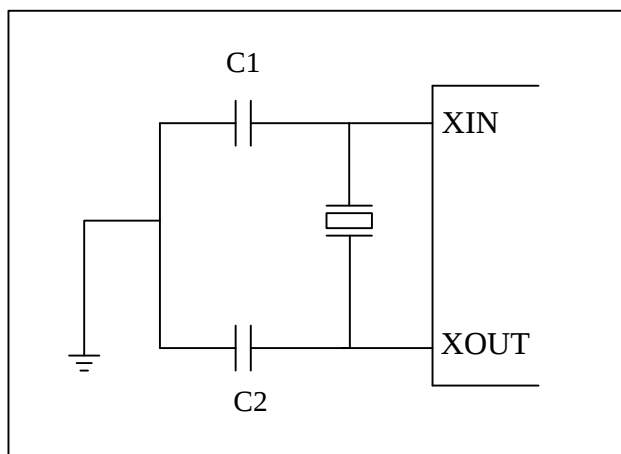
F19.4 CC1S1: CC1 voltage level switch
 0: open
 1: short

F19.3~2 CC2S0: CC2 voltage level select
 00: 5.1K Ohm resister to ground
 01: 10K Ohm resister to VDDA
 10: 22K Ohm resister to VDDA
 11: reserved

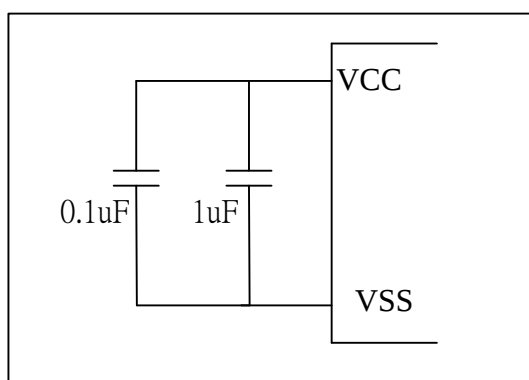
F19.0 CC2S1: CC2 voltage level switch
 0: open
 1: short

3.9 System Clock

System clock can be operated in four different oscillation modes. Four oscillation modes are FIRC, FXT, SIRC and SXT respectively. In Fast/Slow Crystal mode (FXT/SXT), a crystal or ceramic resonator is connected to the XIN and XOUT pins to establish oscillation. In the Fast Internal RC mode (FIRC), the on-chip oscillator generate 12 MHz system clock that can be trimmed by IRCF (F1F.4~0). Since power noise degrades the performance of Fast Internal Clock Oscillator, placing power supply bypass capacitor 1uF and 0.1uF very close to VCC/VSS pins to improve the stability of clock and the overall system.



**External Oscillator Circuit
(Crystal or Ceramic)**



Fast Internal RC Mode

F1F	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IRCF	-	-	-	IRCF				
R/W	-	-	-	R/W				
Reset	-	-	-	by SYSTEM setting				

F1F.4~0 IRCF: FIRC frequency adjustment

There is a Phase Lock Loop (PLL) module in this device. The PLL is used to generate a higher frequency (108MHz) for PWM0/1. By setting the PDPLL (F13.6) to high, the PLL enter power down mode to save power. The reference clock input of PLL is Fast-clock (could be FIRC or FXT) according to FCKTYPE (F0C.6).

F13	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ANA_PWR	PDANG	PDPLL	-	PDIVC	PDLDO	PUMP	VCCFLT	CLKFLT
R/W	R/W	R/W	-	R/W	R/W	R/W	R/W	R/W
Reset	1	1	-	0	1	0	0	0

F13.6 PDPLL: PLL power down enable bit
 0: power down disable
 1: power down enable

4. I/O Port

4.1 PA0~6

The FE10 has total 22 I/O pins. These pins can be used as Schmitt-trigger input, CMOS push-pull output or Open-drain output except PA7. PA7 is used as Schmitt-trigger input. When SYSCFG[12] is set, PA7 is only used for external low active reset.

User can set each pin by their pin mode register PAMODH (R05), PAMODL (R06) for PA and PBMODH (R07), PBMODL (R08) for PB and PDMODH (R09), PDMODL (R0A) for PD. There are 4 kinds of IO modes Mode0, Mode1, Mode2 and Mode3 for each pin can be selected.

If the pin is used for Schmitt-trigger input, S/W must set the I/O pin as Mode0 or Mode1 and set the corresponding Port Data SFR to 1 to disable the pin's output driving circuit. Besides I/O port function, each pin has one or more alternative functions like ADC and OPA.

Mode	Pin function	PXn SFR data	Pin State	Register Pull-up	Digital Input
Mode0	Open Drain	0	Drive Low	N	N
		1	Pull-up	Y	Y
Mode1	Open Drain	0	Drive Low	N	N
		1	Hi-Z	N	Y
Mode2	CMOS Output	0	Drive Low	N	N
		1	Drive High	N	N
Mode3	Alternative Function, such as ADC, OPA	X	-	N	N

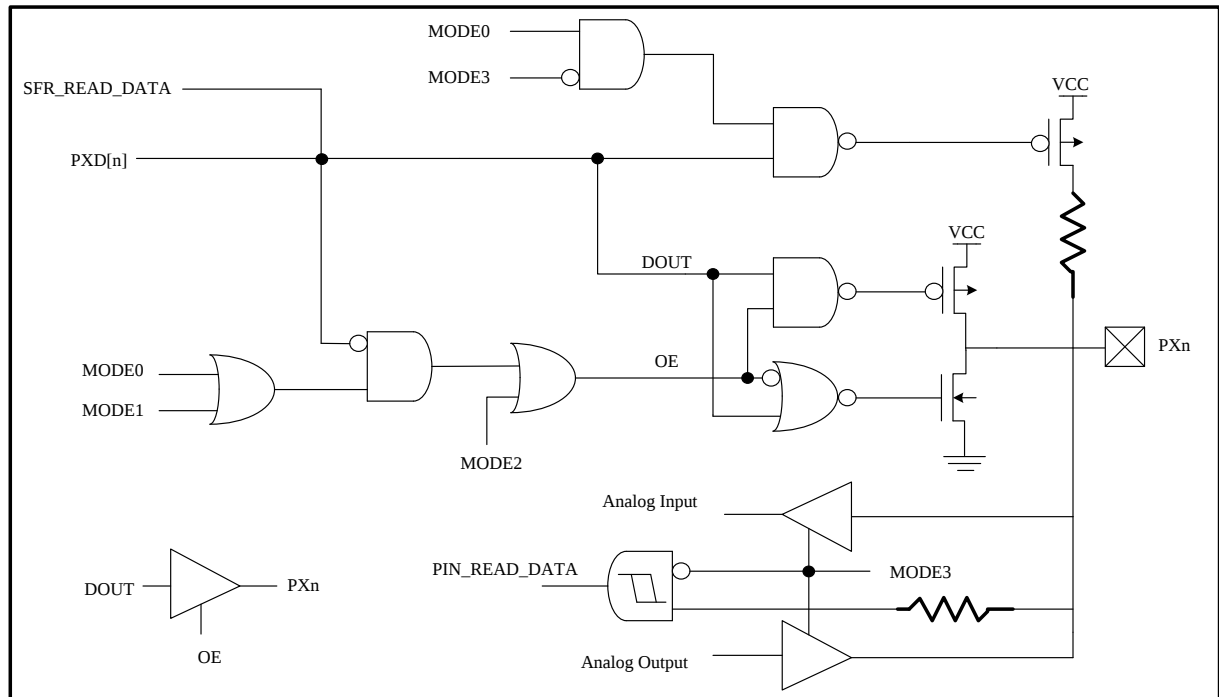
Pin Name	Function	INT	ADC	OPA	CMP	USB Charge	Mode3
PA0	PWM0P		ADC3				ADC3
PA1		INT1	ADC1			DM1	ADC1
PA2	TM0CKI		ADC2			DP2	ADC2
PA3							
PA4							
PA5	TM0OUT		ADC7			DM2	ADC7
PA6		INT0	ADC0			DP1	ADC0
PA7		INT2					
PB0	PWM1P		ADC6				ADC6
PB1	PWM1N		ADC5				ADC5
PB2			ADC12			CC2	ADC12
PB3			ADC11			CC1	ADC11
PB4			ADC10			DM0	ADC10
PB5			ADC9			DP0	ADC9
PD0	TM1OUT			A0P1			A0P1
PD1				A0P2			A0P2
PD2							
PD3					A1P		A1P
PD4					A1N		A1N
PD5					A1X		
PD6	TCOUT		ADC8				ADC8
PD7	PWM0N		ADC4				ADC4

The necessary SFR setting for pin's alternative function is list below.

Alternative Function	Mode	PXn SFR data	Pin State	Other necessary SFR setting
TM0CKI, INT0, INT1, INT2	0	1	Input with Pull-up	TM0CTL, INTIE
	1	1	Input	
TCOUT	0	X	TC Clock Open Drain Output with Pull-up	MR0C
	1	X	TC Clock Open Drain Output	
	2	X	TC Clock Output (CMOS Push-Pull)	
TM0OUT	0	X	TM0 Clock Open Drain Output with Pull-up	MR0C
	1	X	TM0 Clock Open Drain Output	
	2	X	TM0 Clock Output (CMOS Push-Pull)	
TM1OUT	0	X	TM1 Clock Open Drain Output with Pull-up	MR0C
	1	X	TM1 Clock Open Drain Output	
	2		TM1 Clock Output (CMOS Push-Pull)	
PWM0P, PWM0N PWM1P, PWM1N	0	X	PWM Open Drain Output with Pull-up	PWMCTL, MF17
	1	X	PWM Open Drain Output	
	2	X	PWM Output (CMOS Push-Pull)	
ADC0~ADC9	3	X	ADC Channel	MF11
DPn, DMn, CCn	3	X	USB Charge	DPM0CTL, DACCTL0, DACCTL1, CC12SEL, DAC0D, DAC1D
A0P1, A0P2	3	X	OPA analog input	ANA_CTL2

For tables above, a “CMOS Output” pin means it can sink and drive at least 4mA current. It is not recommended to use such as input function.

An “Open Drain” pin means it can sink at least 4mA current but only drive a small current (< 20uA). It can be used as input or output function and typically needs an external pull-up resistor.


IO Pin Structure

Example: Set PA0 as Schmitt-trigger input with pull-up (Mode0)

```
MOVLW    xxxxxxx1B
MOVWF    PAD
MOVLW    xxxxxx00B
MOVWR    PAMODL    ; Set PA0 as Schmitt-trigger input with pull-up
```

Example: Set PA0 as Schmitt-trigger input without pull-up (Mode1)

```
MOVLW    xxxxxxx1B
MOVWF    PAD
MOVLW    xxxxxx01B
MOVWR    PAMODL    ; Set PA0 as Schmitt-trigger input without pull-up
```

Example: Set PA0 as CMOS push-pull output mode and drive low (Mode2)

```
MOVLW    xxxxxxx0B
```

```
MOVWF    PAD
MOVLW    0x00000010B
MOVWR    PAMODL
```

Example PA0 as CMOS push-pull output mode and PWM0P output (Mode2)

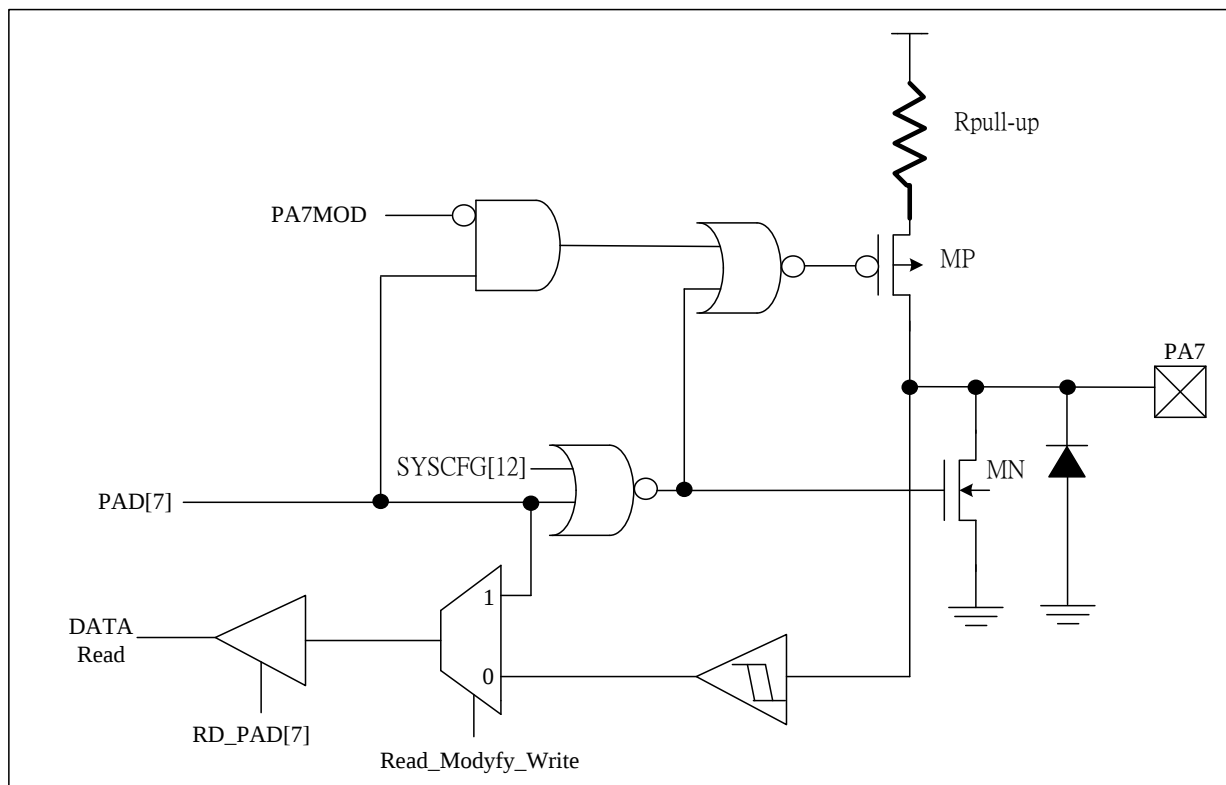
```
MOVLW    0x00000010B
MOVWR    PAMODL
MOVLW    0x00000001B
MOVWR    PWMCTL    ; set PWM0POE = 1
```

Example PA0 as ADC3 input (Mode3)

```
MOVLW    0x00000011B    ; set PA0 as ADC analog input
MOVWR    PAMODL
MOVLW    0x0000000011B    ; select channel ADC3
MOVWR    MF11
```

4.2 PA7

PA7 can be used in Schmitt-trigger input or open-drain output which is setting by the PAD [7] (F05.7) bit. When the PAD [7] bit is assigned as Schmitt-trigger input mode, otherwise is assigned as open-drain output mode and output low. The pull-up resistor is controlled by PA7MOD (R05.6) bit and the default value is disabled (i.e. PA7MOD=1) after system reset.



How to control PA7 status can be concluded as following list.

SYSCFG[12]	PAD[7]	PA7MOD	Pin State	Pull-up	MODE
0	0	0	Low	No	Open-drain output without pull-high
0	1	0	High	Yes	Input with pull-high
0	1	1	Hi-Z	No	Input without pull-high
1	1	0	High	Yes	Reset input with pull-high

✧ Example: Read state from PA7

Condition: SYSCFG[12] is set to "0". If SYSCFG[12] = "1", then PA7 is external reset pin function

```

BTFSS    PAD.7
GOTO     LOOP_A    ; if PA7=0
GOTO     LOOP_B    ; if PA7=1
    
```

F05	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PAD	PAD							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

F05.7 PAD[7]: PA7 data

F05.6~0 PAD[6:0] : PA6~PA0 data output register

0: output low

1: output high or Schmitt-trigger input mode

F06	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PBD	-	-	PBD					
R/W	-	-	R/W					
Reset	-	-	1	1	1	1	1	1

F06.5~0 PBD: PB5~PB0 data

0: output low

1: output high or Schmitt-trigger input mod

F07	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PDD	PDD							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

F07.7~0 PDD: PD7~PD0 data

0: output low

1: output high or Schmitt-trigger input mod

R05	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PAMODH	-	PA7MOD	PA6MOD		PA5MOD		PA4MOD	
R/W	-	W	W		W		W	
Reset	-	0	01		01		01	

R05.6 PA7MOD: PA7 I/O mode control

0: with pull-up (PAD [7] =1) ; 1: without pull-up

R05.5~4 PA6MOD: PA6 Pin mode control

00: Mode 0

01: Mode 1

10: Mode 2

11: Mode 3

R06.3~2 PA5MOD: PA5 Pin mode control

00: Mode 0

01: Mode 1

10: Mode 2

11: Mode 3

R06.1~0 PA4MOD: PA4 Pin mode control

00: Mode 0

01: Mode 1

10: Mode 2

11: Mode 3

R06	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PAMODL	PA3MOD		PA2MOD		PA1MOD		PA0MOD	
R/W	W		W		W		W	
Reset	01		01		01		01	

R06.7~6 PA3MOD: PA3 Pin mode control
 00: Mode 0
 01: Mode 1
 10: Mode 2
 11: Mode 3

R06.5~4 PA2MOD: PA2 Pin mode control
 00: Mode 0
 01: Mode 1
 10: Mode 2
 11: Mode 3

R06.3~2 PA1MOD: PA1 Pin mode control
 00: Mode 0
 01: Mode 1
 10: Mode 2
 11: Mode 3

R06.1~0 PA0MOD: PA0 Pin mode control
 00: Mode 0
 01: Mode 1
 10: Mode 2
 11: Mode 3

R07	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PBMODH	-	-	-	-	PB5MOD		PB4MOD	
R/W	-	-	-	-	W		W	
Reset	-	-	-	-	01		01	

R07.3~2 PB5MOD: PB5 Pin mode control
 00: Mode 0
 01: Mode 1
 10: Mode 2
 11: Mode 3

R07.1~0 PB4MOD: PB4 Pin mode control
 00: Mode 0
 01: Mode 1
 10: Mode 2
 11: Mode 3

R08	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PBMODL	PB3MOD		PB3MOD		PB1MOD		PB0MOD	
R/W	W		W		W		W	
Reset	01		01		01		01	

R08.7~6 PB3MOD: PB3 Pin mode control
 00: Mode 0
 01: Mode 1

10: Mode 2
11: Mode 3

R08.5~4 PB2MOD: PB2 Pin mode control
00: Mode 0
01: Mode 1
10: Mode 2
11: Mode 3

R08.3~2 PB1MOD: PB1 Pin mode control
00: Mode 0
01: Mode 1
10: Mode 2
11: Mode 3

R08.1~0 PB0MOD: PB0 Pin mode control
00: Mode 0
01: Mode 1
10: Mode 2
11: Mode 3

R09	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PDMODH	PD7MOD		PD6MOD		PD5MOD		PD4MOD	
R/W	W		W		W		W	
Reset	01		01		01		01	

R09.7~6 PD7MOD: PD7 Pin mode control
00: Mode 0
01: Mode 1
10: Mode 2
11: Mode 3

R09.5~4 PD6MOD: PD6 Pin mode control
00: Mode 0
01: Mode 1
10: Mode 2
11: Mode 3

R09.3~2 PD5MOD: PD5 Pin mode control
00: Mode 0
01: Mode 1
10: Mode 2
11: Mode 3

R09.1~0 PD4MOD: PD4 Pin mode control
00: Mode 0
01: Mode 1
10: Mode 2
11: Mode 3

R0A	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PDMODL	PD3MOD		PD2MOD		PD1MOD		PD0MOD	
R/W	W		W		W		W	
Reset	01		01		01		01	

R0A.7~6 PD3MOD: PD3 Pin mode control

00: Mode 0

01: Mode 1

10: Mode 2

11: Mode 3

R0A.5~4 PD2MOD: PD2 Pin mode control

00: Mode 0

01: Mode 1

10: Mode 2

11: Mode 3

R0A.3~2 PD1MOD: PD1 Pin mode control

00: Mode 0

01: Mode 1

10: Mode 2

11: Mode 3

R0A.1~0 PD0MOD: PD0 Pin mode control

00: Mode 0

01: Mode 1

10: Mode 2

11: Mode 3

MEMORY MAP

1. F-Plane

Name	Address	R/W	Rst	Description
(F00) INDF				Function related to RAM W/R
INDF	00.7~0	R/W	-	Not a physical register, addressing INDF actually point to the register whose address is contained in the SFR register
(F01) TM0				Function related to : Timer0
TM0	01.7~0	R/W	0	Timer0 content
(F02) PCL				Function related to : PROGRAM COUNT
PCL	02.7~0	R/W	0	Programming Counter LSB[7~0]
(F03) STATUS				Function related to : STATUS
GB1	03.7	R/W	0	General purpose bit 1
GB0	03.6	R/W	0	General purpose bit 0
RAMBK	03.5	R/W	0	Ram Bank Selection
TO	03.4	R	0	WDT timeout flag, cleared by PWRST, ‘SLEEP’ or ‘CLRWDT’ instruction
PD	03.3	R	0	Sleep mode flag, set by ‘SLEEP’ cleared by ‘CLRWDT’ instruction
Z	03.2	R/W	0	Zero flag
DC	03.1	R/W	0	Decimal carry flag
C	03.0	R/W	0	Carry flag
(F04) FSR				Function related to : RAM W/R
GB2	04.7	R/W	0	General purpose bit 2
FSR	04.6~0	R/W	-	File select register, indirect address mode pointer
(F05) PAD				Function related to : Port A
PAD	05.7	R	-	PA7 pin
		W	1	1: PA7 is Schmitt-trigger input mode
	05.6~0	R	-	Port A pin or “data register” state
		W	7F	Port A output data register
(F06) PBD				Function related to : Port B
PBD	06.5~0	R	-	Port B pin or “data register” state
		W	3F	Port B output data register
(F07) PDD				Function related to : Port D
PDD	07.7~0	R	-	Port D pin or “data register” state
		W	FF	Port D output data register
(F08) INTIE				Function related to : Interrupt Enable
PWM0IE	08.7	R/W	0	PWM0 Interrupt Enable 0: disable 1: enable
PWM1IE	08.6	R/W	0	PWM1 Interrupt Enable 0: disable 1: enable
TM1IE	08.5	R/W	0	Timer1 Interrupt Enable 0: disable 1: enable
TM0IE	08.4	R/W	0	Timer0 Interrupt Enable 0: disable 1: enable

Name	Address	R/W	Rst	Description
WKTIE	08.3	R/W	0	Wakeup Timer interrupt enable 0: disable 1: enable
INT2IE	08.2	R/W	0	INT2 (PA7) pin interrupt enable 0: disable 1: enable
INT1IE	08.1	R/W	0	INT1 (PA1) pin interrupt enable 0: disable 1: enable
INT0IE	08.0	R/W	0	INT0 (PA6) pin interrupt enable 0: disable 1: enable
(F09) INTIF				Function related to : Interrupt Flag
PWM0IF	09.7	R	-	PWM0 interrupt flag, set by H/W while PWM0 overflows
		W	0	write 0: clear this flag; write 1: no action
PWM1IF	09.6	R	-	PWM1 interrupt flag, set by H/W while PWM1 overflows
		W	0	write 0: clear this flag; write 1: no action
TM1IF	09.5	R	-	Timer1 interrupt event pending flag, set by H/W while Timer1 overflows
		W	0	write 0: clear this flag; write 1: no action
TM0IF	09.4	R	-	Timer0 interrupt event pending flag, set by H/W while Timer0 overflows
		W	0	write 0: clear this flag; write 1 : no action
WKTIF	09.3	R	-	WKT interrupt event pending flag, set by H/W while WKT time out
		W	0	write 0: clear this flag; write 1 : no action
INT2IF	09.2	R	-	INT2 (PA7) interrupt event pending flag, set by H/W at INT2 pin's falling edge
		W	0	write 0: clear this flag; write 1 : no action
INT1IF	09.1	R	-	INT1 (PA1) interrupt event pending flag, set by H/W at INT1 pin's falling/rising edge
		W	0	write 0: clear this flag; write 1 : no action
INT0IF	09.0	R	-	INT0 (PA6) interrupt event pending flag, set by H/W at INT0 pin's falling edge
		W	0	write 0: clear this flag; write 1 : no action
(F0A) INTIE1				Function related to : Interrupt Enable 1
CMPIE	0A.0	R/W	0	Comparator interrupt enable 0: disable 1: enable
(F0B) INTIF1				Function related to : Interrupt Flag 1
CMPIF	0B.0	R	-	Comparator interrupt event pending flag, set by H/W at Comparator compare a new value
		W	0	write 0: clear this flag; write 1 : no action
(F0C) CLKCTL				Function related to : Fsys
SCKTYPE	0C.7	R/W	0	Slow-clock Type selection 0: SIRC 1: SXT
FCKTYPE	0C.6	R/W	0	Fast-clock Type selection 0: FIRC 1: FXT

Name	Address	R/W	Rst	Description
SLOWSTP	0C.4	R/W	0	Slow-clock stop 0: Slow-clock is running 1: Slow-clock stops running in Power-down mode
FASTSTP	0C.3	R/W	0	Fast-clock stop 0: Fast-clock is running 1: Fast-clock stops running
CPUCKS	0C.2	R/W	0	System clock source select 0: slow-clock 1: Fast0-clock
CPUPSC	0C.1~0	R/W	11	System clock source prescaler. System clock source 00: divided by 8 01: divided by 4 10: divided by 2 11: divided by 1
(F0D) MF0D				Function related to : ADC / Power
ADEOC	0D.7	R	-	ADC end of conversion
ADST	0D.6	R/W	0	ADC start bit 1: ADC start conversion; 0: this bit is automatically clear by H/W end of conversion
PWRSV1	0D.5	R/W	1	Power save option bit 1 1: LVR auto power off in STOP mode
PWRSV0	0D.4	R/W	1	Power save option bit 0 1: IVC auto power off in STOP mode
(F0E) PCH				Function related to : PROGRAM COUNT
PCH	0E.7~4	R	-	Programming Counter MSB[11:8]
(F0F) DPM0CTL				Function related to : USB Battery Charge
DP0VSEL	0F.7~5	R/W	7	DP0 output voltage select; 7: disable, 0: 0.6V, 1: 1.2V, 2: 2.0V, 3: 2.7V, 4:3.3V, 5/6:reserved
DM0VSEL	0F.4~2	R/W	7	DM0 output voltage select; 7:disable, 0: 0.6V, 1: 1.2V, 2: 2.0V, 3: 2.7V, 4:3.3V, 5/6:reserved
DM0PLOW	0F.1	R/W	0	1: enable DM0 pull low, 0: disable DM0 pull low
DPM0SHRT	0F.0	R/W	0	1: DP0/DM0 short together, 0: DP0/DM0 not short
(F10) ADH				Function related to : ADC
ADH	10.7~0	R	-	ADC output data MSB[11~4]
(F11) MF11				Function related to : ADC
ADL	11.7~4	R	-	ADC output data LSB[3~0]
ADCHS	11.3~0	R/W	0	ADC channel select 0000: ADC0 (PA6) 0001: ADC1 (PA1) 0010: ADC2 (PA2) 0011: ADC3 (PA0) 0100: ADC4 (PD7) 0101: ADC5 (PB1) 0110: ADC6 (PB0) 0111: ADC7 (PA5) 1000: ADC8 (PD6) 1001: ADC9 (PB5) 1010: ADC10 (OPO) 1010: ADC11 (VBGO)

Name	Address	R/W	Rst	Description
(F12) MF12			Function related to : PWM0 / PWM1 / TM0 / TM1	
PWM0CLR	12.7	R/W	1	PWM0 clear and hold 0: PWM0 is running 1: PWM0 is clear and hold
PWM1CLR	12.6	R/W	1	PWM1 clear and hold 0: PWM1 is running 1: PWM1 is clear and hold
TM1SET	12.3	R/W	0	Timer1 set and hold 0: release Timer1 1: set Timer1 to ‘FFFFH’ and hold
TM1CLR	12.2	R/W	0	Timer1 clear 0: release Timer1 1: clear Timer1 to ‘0000H’ and hold
TM1STP	12.1	R/W	0	Timer1 counter stop 0: release 1: Stop Timer1 counting
TM0STP	12.0	R/W	0	Timer0 counter stop 0: release 1: Stop Timer0 counting
(F13) ANA_PWR			Function related to : Power saving / EFT / System Voltage	
PDANG	13.7	R/W	1	OPA / Comparator power down enable bit 0: OPA/Comparator running 1: OPA/Comparator power down
PDPLL	13.6	R/W	1	Internal PLL power down enable bit 0: PLL running 1: PLL power down
PDIVC	13.4	R/W	0	IVC power saving enable 0: disable 1: enable
PDLDO	13.3	R/W	1	Internal LDO power down enable bit 0: LDO running 1: LDO power down
PUMP	13.2	R/W	0	Internal PLL pump voltage oscillator 0: disable 1: enable
VCCFLT	13.1	R/W	0	VCC filter, enhance the chip’s power noise immunity 0: disable 1: enable
CLKFLT	13.0	R/W	0	Fsys clock filter for noise defecting 0: Fast 1: Slow
(F14) TM1L			Function related to : Timer1	
TM1L	14.7~0	R	-	Timer1 counter low byte
		W	0	Timer1 reload low byte
(F15) TM1H			Function related to : Timer1	
TM1H	15.7~0	R	-	Timer1 counter high byte
		W	0	Timer1 reload high byte

Name	Address	R/W	Rst	Description
(F16) DACCTL0				Function related to : DAC control register 0
DMDS0	16.6	R/W	0	DM1 signal switch; 0:open 1: DAC0O/DM1 short
DMDS1	16.4	R/W	0	DM2 signal switch; 0:open 1: DAC0O/DM2 short
DPDS0	16.2	R/W	0	DP1 signal switch; 0:open 1: DAC1O/DP1 short
DPDS1	16.0	R/W	0	DP2 signal switch; 0:open 1: DAC1O/DP2 short
(F17) MF17				Function related to : PWM0 / PD7 / PA0
PWM1CKS	17.7~6	R/W	0	PWM1 clock source select 00: Fsys 01: 2*Fsys(2 times Fsys frequency) 1x: PLLCKO 108MHz
PWM0CKS	17.5~4	R/W	0	PWM0 clock source select 00: Fsys 01: 2*Fsys(2 times Fsys frequency) 1x: PLLCKO 108MHz
PD7HDRV	17.3	R/W	0	PWM0N (PD7) high drive current enable bit 0: disable 1: enable
PD7HSNK	17.2	R/W	0	PWM0N (PD7) high sink current enable bit 0: disable 1: enable
PA0HDRV	17.1	R/W	0	PWM0P (PA0) high drive current enable bit 0: disable 1: enable
PA0HSNK	17.0	R/W	0	PWM0P (PA0) high sink current enable bit 0: disable 1: enable
(F18) DACCTL1				Function related to : DAC control register 1
DMPL0	18.5	R/W	0	DM1 pull low 0: disable 1: enable
DMPL1	18.4	R/W	0	DM2 pull low 0: disable 1: enable
DPMS0	18.3	R/W	0	DP1/DM1 short enable bit 0: disable 1: enable
DPMS1	18.2	R/W	0	DP2/DM2 short enable bit 0: disable 1: enable
DMPD	18.1	R/W	1	DM1/DM2 voltage source power down enable bit 0: disable 1: enable
DMPD	18.0	R/W	1	DM1/DM2 voltage source power down enable bit 0: disable 1: enable

Name	Address	R/W	Rst	Description
(F19) CC12SEL		Function related to : CC12 select register		
CC1S0	19.7~6	R/W	0	V_CC1 source select; 00 AVSS 5.1K, 01: AVDD 10K, 10: AVDD 22K
CC1S1	19.5~4	R/W	0	CC1 signal switch; 00 open, 01: V_CC1/CC1 short, 10: ADC11/CC1 short
CC2S0	19.3~2	R/W	0	V_CC2 source select; 00 AVSS 5.1K, 01: AVDD 10K, 10: AVDD 22K
CC2S1	19.1~0	R/W	0	CC2 signal switch; 00 open, 01: V_CC2/CC2 short, 10: ADC12/CC2 short
(F1A) VBTM		Function related to : System Band-gap voltage		
VBTM	1A.3~0	R/W	by SYS	Band-gap voltage adjustment
(F1B) OPTM		Function related to : OPA		
OPTM	1B.4~0	R/W	by SYS	OPA offset adjustment value
(F1C) RSR		Function related to : RAM W/R		
RSR	1C.7~0	R	-	R-Plane file select register, indirect address mode pointer
(F1D) DPL		Function related to : Table Read		
DPL	1D.7~0	R	0	Table read low address, data ROM pointer DPTR) low byte[7~0]
(F1E) DPH		Function related to : Table Read		
DPH	1E.3~0	R	0	Table read high address, data ROM pointer DPTR) high byte[11~8]
(F1F) IRCF		Function related to : Internal RC		
IRCF	1F.4~0	R	by SYS	FIRC frequency adjustment
User Data Memory				
SRAM	20~27	R/W	-	SRAM common area
	28~7F	R/W	-	SRAM Bank0 area (RAMBK=0, 88 bytes)
	28~7F	R/W	-	SRAM Bank1 area (RAMBK=1, 88 bytes)

2. R-Plane

Name	Address	R/W	Rst	Description
(R00) INDR				Function related to RAM W/R
INDR	00.7~0	R/W	-	Not a physical register, addressing INDR actually point to the register whose address is contained in the RSR register
(R01) TM0RLD				Function related to : Timer0
TM0RLD	01.7~0	R/W	-	Timer0 reload data
(R02) TM0CTL				Function related to : Timer0
TM0CL	02.7	R/W	0	Timer0 capture Mode level 0: High level capture 1: Low level capture
TM0CM	02.6	R/W	0	Timer0 Mode 0: Timer / Counter Mode Clock source from TM0PSC (set R02.3~0) TM0CKI (set R02.4) 1: Capture Mode Clock source from CAPT pin
TM0EDG	02.5	R/W	0	Timer0 prescaler counting edge for TM0CKI 0: rising edge 1: falling edge
TM0CKS	02.4	R/W	0	Timer0 prescaler clock source select 0: Fsys/2 1: TM0CKI pin (PA2)
TM0PSC	02.3~0	R/W	0	Timer0 prescaler. Timer0 prescaler clock source divided by 0000: divided by 1 0001: divided by 2 0010: divided by 4 0011: divided by 8 0100: divided by 16 0101: divided by 32 0110: divided by 64 0111: divided by 128 1xxx: divided by 256
(R03) PWRDN				Function related to : Power Down
PWRDN	03	W	-	Write this register to enter STOP mode (i.e. 'SLEEP' instruction)
(R04) WDTCLR				Function related to : WDT
WDTCLR	04	W	-	Write this register to clear WDT Timer (i.e. 'CLRWDT' instruction)
(R05) PAMODH				Function related to : Port A
PA7MOD	05.6	W	1	PA7 I/O mode control 0: with pull-up (PAD [7] =1) 1: without pull-up
PA6MOD	05.5~4	W	1	PA6~PA4 I/O mode control 00: Mode0
PA5MOD	05.3~2	W	1	01: Mode1
PA4MOD	05.1~0	W	1	10: Mode2 11: Mode3
(R06) PAMODL				Function related to : Port A
PA3MOD	06.7~6	W	1	PA3~PA0 I/O mode control
PA2MOD	06.5~4	W	1	00: Mode0
PA1MOD	06.3~2	W	1	01: Mode1
PA0MOD	06.1~0	W	1	10: Mode2 11: Mode3

Name	Address	R/W	Rst	Description
(R07) PBMODH				Function related to : Port B
PB5MOD	07.3~2	W	1	PB5~PB4 I/O mode control 00: Mode0 01: Mode1
PB4MOD	07.1~0	W	1	10: Mode2 11: Mode3
(R08) PBMODL				Function related to : Port B
PB3MOD	08.7~6	W	1	PB3~PB0 I/O mode control 00: Mode0 01: Mode1
PB2MOD	08.5~4	W	1	10: Mode2 11: Mode3
PB1MOD	08.3~2	W	1	
PB0MOD	08.1~0	W	1	
(R09) PDMODH				Function related to : Port D
PD7MOD	09.7~6	W	1	PD7~PD4 I/O mode control 00: Mode0 01: Mode1
PD6MOD	09.5~4	W	1	10: Mode2 11: Mode3
PD5MOD	09.3~2	W	1	
PD4MOD	09.1~0	W	1	
(R0A) PDMODL				Function related to : Port D
PD3MOD	0A.7~6	W	1	PD3~PD0 I/O mode control 00: Mode0 01: Mode1
PD2MOD	0A.5~4	W	1	10: Mode2 11: Mode3
PD1MOD	0A.3~2	W	1	
PD0MOD	0A.1~0	W	1	
(R0B) PWMCTL				Function related to : PWM
PWM0POE	0B.7	W	0	Enable PWM0P output to PA0 pin 0: disable 1: enable
PWM0NOE	0B.6	W	0	Enable PWM0N output to PD7 pin 0: disable 1: enable
PWM1POE	0B.5	W	0	Enable PWM1P output to PB0 pin 0: disable 1: enable
PWM1NOE	0B.4	W	0	Enable PWM1N output to PB1 pin 0: disable 1: enable
PWM0PSC	0B.3~2	W	0	PWM0 prescaler. PWM0 clock source 00: divided by 1 01: divided by 2 10: divided by 4 11: divided by 8
PWM1PSC	0B.1~0	W	0	PWM1 prescaler. PWM1 clock source 00: divided by 1 01: divided by 2 10: divided by 4 11: divided by 8
(R0C) MR0C				Function related to : WDT / WKT / Timer0 / Timer1
WKT PSC	0C.7~6	W	11	WKT period (@VCC=5V) 00: 27ms 01: 54ms 10: 108ms 11: 216ms

Name	Address	R/W	Rst	Description
WDTPSC	0C.5~4	W	01	WDT period (@VCC=5V) 00: 108ms 01: 216ms 10: 864ms 11: 1728ms
TM1CKS	0C.3	W	0	Timer1 clock source select 0: Fsys/2 1: Fsys
TM0OE	0C.2	W	0	Enable Timer0 overflow toggle output PA5 pin (TM0OUT) 0: disable 1: enable
TCOE	0C.1	W	0	Enable Instruction cycle (Fsys/2) output PD6 pin (TCOUT) 0: disable 1: enable
TM1OE	0C.0	W	0	Enable Timer1 overflow toggle output PD0 pin (TM1OUT) 0: disable 1: enable
(R0D) MR0D				Function related to : INT / Timer1 / PLL
INT1EDG	0D.7	W	0	INT1 (PA1) trigger edge select 0: INT1 (PA1) pin falling edge to trigger interrupt event 1: INT1 (PA1) pin rising edge to trigger interrupt event
TM1CM	0D.6	W	0	Timer1 Mode 0: Timer Mode (source from TM1PSC clock out) 1: Capture Mode (source from CAPT pin)
HWAUTO	0D.5	W	0	Enable H/W save/restore STATUS during Interrupt 0: disable 1: enable
ADCKS	0D.2~0	W	4	ADC clock frequency selection 000: Fsys / 256 001: Fsys / 128 010: Fsys / 64 011: Fsys / 32 100: Fsys / 16 101: Fsys / 8 110: Fsys / 4 111: Fsys / 2
(R0E) MR0E				Function related to : Comparator / PWM0
CMPTRIG	0E.7~6	W	1	Comparator interrupt trigger direction 00=Rising edge (when R1B.1=0) 01=Falling edge (when R1B.1=0) 10=Both edge (when R1B.1=0) 11=high level trigger (when R1B.1=0)
PWM0MODE	0E.5~4	W	0	PWM0 differential output mode 00: Mode0 01: Mode1 10: Mode2 11: Mode3

Name	Address	R/W	Rst	Description
PWM0DT	0E.3~0	W	0	PWM0 Dead Time (non-overlap time) select 0000: 0 pwm0 prescale clock (original PWM0) 0001: 1 pwm0 prescale clock 0010: 2 pwm0 prescale clock 0011: 3 pwm0 prescale clock 0100: 4 pwm0 prescale clock 0101: 5 pwm0 prescale clock 0110: 6 pwm0 prescale clock 0111: 7 pwm0 prescale clock 1000: 8 pwm0 prescale clock 1001: 9 pwm0 prescale clock 1010: 10 pwm0 prescale clock 1011: 11 pwm0 prescale clock 1100: 12 pwm0 prescale clock 1101: 13 pwm0 prescale clock 1110: 14 pwm0 prescale clock 1111: 16 pwm0 prescale clock
(R10) PWM0PRDH				Function related to : PWM0
PWM0PRDH	10.7~0	W	ff	PWM0 period data 8-bt MSB PWM0PRD[10:3]
(R11) PWMPRDL				Function related to : PWM0 / PWM1
PWM1PRDL	11.7~5	W	3	PWM1 period data 3-bt LSB PWM1PRD[2:0]
PWM0PRDL	11.2~0	W	3	PWM0 period data 3-bt LSB PWM0PRD[2:0]
(R12) PWM1PRDH				Function related to : PWM1
PWM1PRDH	12.7~0	W	ff	PWM1 period data 8-bt MSB PWM1PRD[10:3]
(R13) MR13				Function related to : Comparator / PWM1
CMPDBS	13.7~6	W	0	Comparator output debounce time 00=None 01=4*Fsys 10=8*Fsys 11=16*Fsys
PWM1MODE	13.5~4	W	0	PWM1 differential output mode 00: Mode0 01: Mode1 10: Mode2 11: Mode3
PWM1DT	13.3~0	W	0	PWM1 Dead Time (non-overlap time) select 0000: 0 pwm1 prescale clock (original PWM1) 0001: 1 pwm1 prescale clock 0010: 2 pwm1 prescale clock 0011: 3 pwm1 prescale clock 0100: 4 pwm1 prescale clock 0101: 5 pwm1 prescale clock 0110: 6 pwm1 prescale clock 0111: 7 pwm1 prescale clock 1000: 8 pwm1 prescale clock 1001: 9 pwm1 prescale clock 1010: 10 pwm1 prescale clock 1011: 11 pwm1 prescale clock 1100: 12 pwm1 prescale clock 1101: 13 pwm1 prescale clock 1110: 14 pwm1 prescale clock 1111: 16 pwm1 prescale clock

Name	Address	R/W	Rst	Description
(R14) PWM0DUT1				Function related to : PWM0
PWM0DUT1	14.7~0	R/W	0	PWM0DUTY [9:2]
(R15) PWM0DUT2				Function related to : PWM0
PWM0DUT2	15.7~5, 1~0	R/W	0	PWM0DUTY [12:10], PWM0DUTY [1:0]
(R16) PWM1DUT1				Function related to : PWM1
PWM1DUT1	16.7~0	R/W	0	PWM1DUTY [9:2]
(R17) PWM1DUT2				Function related to : PWM1
PWM1DUT2	17.7~5, 1~0	R/W	0	PWM1DUTY [12:10], PWM1DUTY [1:0]
(R18) ANA_CTL1				Function related to : Analog IP
PDBGOP	18.7	W	0	Power down VBGO to ADC enable 0: disable 1: enable
SVL	18.6~0	W	0	Select comparator reference voltage level Total 125 level; when SVL=127, comparator disconnect to 125 level reference voltage
(R19) ANA_CTL2				Function related to : Analog IP
SOPP	19.7	W	0	Select OPP input source 0: OPP input source is PD0 1: OPP input source is PD1
OPOF	19.6	W	0	OPA output (OPO) connect to Comparator control 0: OPO connect to COMPP 1: OPO disconnect to COMPP
SVRF	19.5	W	0	Comparator reference voltage source select 0: VDDA 1: LDOC
SOTV	19.4	W	0	select reference voltage source for OPA trim offset 0: AVSS 1: LDOC
SOPG	19.3~2	W	3	OPA gain select 00: 1X 01: 20X 10: 50X 11: 100X
SOPM	19.1~0	W	0	OPA working mode select 00: normal 01: Trim 1 10: Trim 2 11: Comp.
(R1A) ANA_CTL3				Function related to : Analog IP
VCLTEN	1A.7	W	0	Reserved for test; VCO control voltage switch
ANTOE	1A.6	W	0	Reserved for test; Analog signal output enable
SANTO	1A.5~4	W	0	Reserved for test; Analog signal output select 00: VBG 01: VBGO 10: LDOC 11: OPO
(R1B) CMPISSET				Function related to : Comparator
SOPPL	1B.2	W	0	1: enable OPP input pull low 0: disable
CMPINV	1B.1	W	0	1: enable Comparator output data invert (0 -> 1 or 1-> 0) 0: disable

Name	Address	R/W	Rst	Description
CMP2PWMOFF	1B.0	W	0	1: enable Comparator Interrupt to disable PWM output; 0: disable
(R1C) PBWKUP Function related to : Port B wakeup				
PBWKUP	1C.5	W	0	1: PB5 wakeup enable 0:disable
	1C.4	W	0	1: PB4 wakeup enable 0:disable
	1C.3	W	0	1: PB3 wakeup enable 0:disable
	1C.2	W	0	1: PB2 wakeup enable 0:disable
	1C.1	W	0	1: PB1 wakeup enable 0:disable
	1C.0	W	0	1: PB0 wakeup enable 0:disable
(R1E) DAC0D Function related to : DAC0 Data				
DAC0D	1E.7~0	W	0	DAC0 data register
(R1F) DAC1D Function related to : DAC1 data				
DAC1D	1F.7~0	W	0	DAC1 data register

Instruction Set

Each instruction is a 14-bit word divided into an OPCODE, which specifies 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 listed in the following table.

For byte-oriented instructions, “f” or “r” represents the address designator and “d” represents the 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 / Legend	Description
f	F-Plane Register File Address
r	R-Plane 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)
[]	Option Field
()	Contents
.	Bit Field
B	Before
A	After
←	Assign direction

Mnemonic		Op Code	Cycle	Flag Affect	Description
Byte-Oriented File Register Instruction					
<u>ADDWF</u>	f,d	00 0111 dfff ffff	1	C, DC, Z	Add W to f
<u>ANDWF</u>	f,d	00 0101 dfff ffff	1	Z	AND W to f
<u>CLRF</u>	f	00 0001 1fff ffff	1	Z	Clear f
<u>CLRW</u>		00 0001 0100 0000	1	Z	Clear W
<u>COMF</u>	f,d	00 1001 dfff ffff	1	Z	Invert F bit by bit
<u>DECF</u>	f,d	00 0011 dfff ffff	1	Z	Decrement of f
<u>DECFSZ</u>	f,d	00 1011 dfff ffff	1 or 2	-	Decrease f, skip if zero
<u>INCF</u>	f,d	00 1010 dfff ffff	1	Z	Increment of f
<u>INCFSZ</u>	f,d	00 1111 dfff ffff	1 or 2	-	Increase f, skip if zero
<u>IORWF</u>	f,d	00 0100 dfff ffff	1	Z	OR W to f
<u>MOVFW</u>	f	00 1000 0fff ffff	1	-	Move f to W
<u>MOVWF</u>	f	00 0000 1fff ffff	1	-	Move W to f
MOVRW	r	01 1111 rrrr rrrr	1	-	Move r to W
MOVWR	r	01 1110 rrrr rrrr	1	-	Move W to r
<u>RLF</u>	f,d	00 1101 dfff ffff	1	C	F rotate to left
<u>RRF</u>	f,d	00 1100 dfff ffff	1	C	F rotate to right
<u>SUBWF</u>	f,d	00 0010 dfff ffff	1	C, DC, Z	Subtract W from f
<u>SWAPF</u>	f,d	00 1110 dfff ffff	1	-	Swap high and low nibble of f
<u>TESTZ</u>	f	00 1000 1fff ffff	1	Z	Test f if zero
<u>XORWF</u>	f,d	00 0110 dfff ffff	1	Z	XOR W to f
Bit-Oriented File Register Instruction					
<u>BCF</u>	f,b	01 000b bbff ffff	1	-	Bit clear f
<u>BSF</u>	f,b	01 001b bbff ffff	1	-	Bit set f
<u>BTFSC</u>	f,b	01 010b bbff ffff	1 or 2	-	Bit test f, skip if clear
<u>BTFSS</u>	f,b	01 011b bbff ffff	1 or 2	-	Bit test f, skip if set
Literal and Control Instruction					
<u>ADDLW</u>	k	01 1100 kkkk kkkk	1	C, DC, Z	Add literal to W
<u>ANDLW</u>	k	01 1011 kkkk kkkk	1	Z	AND literal to W
<u>XORLW</u>	k	01 1101 kkkk kkkk	1	Z	XOR literal to W
<u>CALL</u>	k	10 kkkk kkkk kkkk	2	-	Subroutine call
<u>CLRWD_T</u>		01 1110 0000 0100	1	TO, PD	Clear watchdog timer
<u>GOTO</u>	k	11 kkkk kkkk kkkk	2	-	Unconditional branch
<u>IORLW</u>	k	01 1010 kkkk kkkk	1	Z	OR literal to W
<u>MOVLW</u>	k	01 1001 kkkk kkkk	1	-	Move literal to W
<u>NOP</u>		00 0000 0000 0000	1	-	No operation
<u>RET</u>		00 0000 0100 0000	2	-	Return from CALL
<u>RETI</u>		00 0000 0110 0000	2	-	Return from interrupt
<u>RETLW</u>	k	01 1000 kkkk kkkk	2	-	Return with literal to W
<u>SLEEP</u>		01 1110 0000 0011	1	TO, PD	Go into Power-down mode
<u>TABRL</u>		00 0000 0 1010000	2	-	Lookup ROM low data to W
<u>TABRH</u>		00 0000 0 1011000	2		Lookup ROM high data to W

ADDLW	Add Literal “k” and W	
Syntax	ADDLW k	
Operands	k : 00h ~ FFh	
Operation	$(W) \leftarrow (W) + k$	
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	$(\text{Destination}) \leftarrow (W) + (f)$	
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	$(W) \leftarrow (W) \text{ 'AND' } k$	
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	$(\text{Destination}) \leftarrow (W) \text{ 'AND' } (f)$	
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 ~ FFFh
Operation	Operation: TOS ← (PC)+ 1, PC.11~0 ← k
Status Affected	-
OP-Code	10 kkkk kkkk kkkk
Description	Call Subroutine. First, return address (PC+1) is pushed onto the stack. The 12-bit immediate address is loaded into PC bits <9:0>. 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
CLRWDWT	Clear Watchdog Timer
Syntax	CLRWDWT
Operands	-
Operation	WDT Timer ← 00h
Status Affected	TO,PD
OP-Code	00 0000 0000 0100
Description	CLRWDWT instruction enables and resets the Watchdog Timer.
Cycle	1
Example	CLRWDWT B : WDT counter = ? A : WDT counter = 0x00

COMF	Complement “f”
Syntax	COMF f [,d]
Operands	f : 00h ~ 7Fh, d : 0, 1
Operation	(destination) $\leftarrow$ ($\bar{f}$)
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) $\leftarrow$ (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) $\leftarrow$ (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 B : PC = LABEL1 GOTO LOOP A : CNT = CNT - 1 CONTINUE if CNT=0, PC = CONTINUE if CNT≠0, PC = LABEL1+1
GOTO	Unconditional Branch
Syntax	GOTO k
Operands	k : 00h ~ 3FFh
Operation	PC.9~0 $\leftarrow$ k
Status Affected	-
OP-Code	11 kkkk kkkk kkkk
Description	GOTO is an unconditional branch. The 12-bit immediate value is loaded into PC bits <9:0>. 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) $\leftarrow$ (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) $\leftarrow$ (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) $\leftarrow$ (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) $\leftarrow$ (W) OR (f)
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	MOVFw FSR, 0	B : FSR = 0xC2, W = ? A : FSR = 0xC2, W = 0xC2

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

MOVWR Move W to “r”

Syntax	MOVWR r	
Operands	r : 00h ~ 3Fh	
Operation	(r) ← (W)	
Status Affected	-	
OP-Code	01 1110 00rr rrrr	
Description	Move data from W register to register ‘r’.	
Cycle	1	
Example	MOVWR REG1	B : REG1 = 0xFF, W = 0x4F A : REG1 = 0x4F, W = 0x4F

MOVRW Move “r” to W

Syntax	MOVRW r	
Operands	r : 20h ~ FFh	
Operation	(W) ← (r)	
Status Affected	-	
OP-Code	01 1111 rrrr rrrr	
Description	Move data from register ‘r’ to W register.	
Cycle	1	
Example	MOVRW REG1	B : REG1 = 0x4F, W = ? A : REG1 = 0x4F, W = 0x4F

NOP No Operation

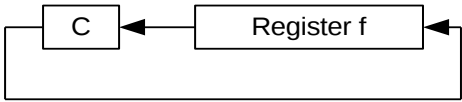
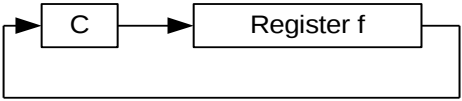
Syntax	NOP	
Operands	-	
Operation	No Operation	
Status Affected	-	
OP-Code	00 0000 0000 0000	
Description	No Operation	
Cycle	1	
Example	NOP	-

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

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 eight-bit 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
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

TABRH Return DPTR high byte to W

Syntax	TABRH
Operands	-
Operation	(W) ← ROM[DPTR] high byte content, Where DPTR = {DPH[9:8], DPL[7:0]}
Status Affected	-
OP-Code	00 0000 0101 1000
Description	The W register is loaded with high byte of ROM[DPTR]. This is a two-cycle instruction
Cycle	2

TABRL Return DPTR low byte to W

Syntax	TABRL
Operands	-
Operation	(W) ← ROM[DPTR] high byte content, Where DPTR = {DPH[9:8], DPL[7:0]}
Status Affected	-
OP-Code	00 0000 0101 0000
Description	The W register is loaded with high byte of ROM[DPTR]. This is a two-cycle instruction
Cycle	2

Example

```

:
:
MOVLW (TAB1 & 0xFF)
MOVWF DPL           ; where DPL is F-plane register
MOVLW (TAB1 >> 8) & 0xFF
MOVWF DPH           ; where DPH is F-plane register

TABRL               ; W=0x89
TABRH               ; W=0x37

ORG 0234H
TAB1:
.DT 0x3789, 0x2277

```

SLEEP Go into standby mode, Clock oscillation stops

Syntax	SLEEP
Operands	-
Operation	-
Status Affected	TO,PD
OP-Code	01 1110 0000 0011
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	$(W) \leftarrow (f) - (W)$	
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	$(\text{destination}, 7 \sim 4) \leftarrow (f, 3 \sim 0), (\text{destination}, 3 \sim 0) \leftarrow (f, 7 \sim 4)$	
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) $\leftarrow$ (W) XOR k
Status Affected	Z
OP-Code	01 1101 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) $\leftarrow$ (W) XOR (f)
Status Affected	Z
OP-Code	00 0110 dkkk kkkk
Description	Exclusive OR the content 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

Electrical Characteristics

1. Absolute Maximum Ratings ($T_A = 25^\circ\text{C}$)

Parameter	Rating	Unit
Supply voltage	$V_{SS} - 0.3$ to $V_{SS} + 5.5$	V
Input voltage	$V_{SS} - 0.3$ to $V_{CC} + 0.3$	
Output voltage	$V_{SS} - 0.3$ to $V_{CC} + 0.3$	
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 +105	$^\circ\text{C}$
Storage temperature	-65 to +150	

2. DC Characteristics ($T_A = -25^\circ\text{C}$ to $+85^\circ\text{C}$, $V_{CC} = 5\text{V}$)

Parameter	Symbol	Conditions		Min	Typ	Max	Unit
Input High Voltage	V_{IH}	All Input	$V_{CC} = 5\text{V}$	$0.8 V_{CC}$	—	V_{CC}	V
Input Low Voltage	V_{IL}	All Input	$V_{CC} = 5\text{V}$	0	—	$0.2 V_{CC}$	V
Output High Current	I_{OH}	All pin except PB0,PB1	$V_{CC} = 5\text{V}$, $V_{OH} = 4.5\text{V}$		8.5		mA
		PB0,PB1			16		
Output Low Current	I_{OL}	All pin except PB0,PB1	$V_{CC} = 5\text{V}$, $V_{OL} = 0.5\text{V}$		19		mA
		PB0,PB1			28		
Output High Current	I_{OH}	PA0, PD7	$V_{CC} = 5\text{V}$, $V_{OL} = 0.5\text{V}$ PA0HDRV=1 PD7HDRV=1		180		mA
Output Low Current	I_{OL}	PA0, PD7	$V_{CC} = 5\text{V}$, $V_{OL} = 0.5\text{V}$ PA0HSNK=1 PD7HSNK=1		200		mA
Input Leakage Current (pin high)	I_{ILH}	All Input	$V_{IN} = V_{CC}$	—	—	1	μA
Input Leakage Current (pin low)	I_{ILL}	All Input	$V_{IN} = 0\text{V}$	—	—	-1	μA
Output Leakage Current (pin high)	I_{OLH}	All Output	$V_{OUT} = V_{CC}$	—	—	2	μA
Output Leakage Current (pin low)	I_{OLL}	All Output	$V_{OUT} = 0\text{V}$	—	—	-2	μA
Power Supply Current	I_{CC}	Run 12 MHz	$V_{CC} = 5\text{V}$		3		mA
		Run 6 MHz	$V_{CC} = 5\text{V}$		2		mA
		Idle Mode	$V_{CC} = 5\text{V}$		14		μA

		Stop Mode	$V_{CC} = 5V$		0.1	1	μA
		Slow Mode SIRC 150KHz	$V_{CC} = 5V$		620		μA
Pull-Up Resistor	R_p	$V_{IN} = 0V$ PA6~PA0	$V_{CC} = 5V$		130		$k\Omega$
		$V_{IN} = 0V$ PA7	$V_{CC} = 5V$		60		
Bandgap Reference Voltage	VBG/ VBGO		$V_{CC}=5V$	-2%	1.25	+2%	V

3. Clock Timing ($T_A = -25^{\circ}C$ to $+85^{\circ}C$, $V_{CC} = 5V$)

Parameter	Condition	Min	Typ	Max	Unit
Fast Internal RC Frequency	$0^{\circ}C$ to $+85^{\circ}C$, $V_{CC} = 5V$	-3%	12.00	+3%	MHz
Slow Internal RC Frequency	$0^{\circ}C$ to $+85^{\circ}C$, $V_{CC} = 5V$	-25%	150	+25%	KHz

4. Reset Timing Characteristics ($T_A = -25^{\circ}C$ to $+85^{\circ}C$, $V_{CC} = 5V$)

Parameter	Conditions	Min	Typ	Max	Unit
Input High Voltage	—	$0.8 V_{CC}$	—	V_{CC}	V
Input Low Voltage	—	—	—	$0.2 V_{CC}$	V
RESET Input Low width	Input $V_{CC} = 5V \pm 10\%$	3	—	—	μs
WDT wakeup time	$V_{CC} = 5V$, WDT_PSC = 11				ms
CPU start up time	$V_{CC}=5V$		12		ms

5. ADC Electrical Characteristics ($T_A = -25^{\circ}C$ to $+85^{\circ}C$, $V_{CC} = 5V$)

Parameter	Conditions	Min	Typ	Max	Unit
Total Accuracy	$V_{CC} = 5V$	-	± 2.5	± 5	LSB
Integral Non-Linearity		-	± 3.2	± 5	LSB
Max Input Clock(f_{ADC})	-	-	-	1	MHz
Conversion Time	$f_{ADC}=1MHz$	-	50	-	μs
Input Voltage	TM57MA45	VSS	-	$0.95LDO_{C}$	V
	TM57MA46	VSS	-	$0.95V_{CC}$	

6. OPA Electrical Characteristics ($T_A = -25^{\circ}\text{C}$ to $+85^{\circ}\text{C}$, $V_{CC} = 5\text{V}$)

Symbol	Parameter	Test Conditions	Min	Typ	Max	Unit
VDDA	Power Supply		2.2	—	5.5	V
V _{icm}	Input Common Voltage		0	—	V _{CC} -1.2	V
V _{os2}	Input Offset Voltage	V _o =2V, after trim	-	-	2	mV
$\Delta V_{os}/\Delta T$	Temperature Coefficient of V _{os}	V _o =2V	-	-	5	$\mu\text{V}/^{\circ}\text{C}$
AVOL	Large Signal Voltage Gain	R _L =1M Ω , C _L =100pF V _i =0.1 to 4V, V _o =1 to 4V	-	120	-	dB
GWB	Gain Band Width Product	R _L =1M Ω , C _L =100pF	-	2.1	-	MHz
CMRR	Common Mode Rejection Ratio	V _o =2V	-	80	-	dB
PSRR	Power Supply Rejection Ratio	V _o =2V	-	80	-	dB
ICC	Supply Current Per Single Amplifier	A _v =1, V _o =2V; No Load	-	300	-	μA
SR	Slew Rate at Unity Gain	No Load	-	1.4	-	V/ μs
Φ_m	Phase Margin at Unity Gain	R _L =1M Ω , C _L =100pF	-	60	-	Degree
IOH	Output Source Current	V _{i+} - V _{i-} $\geq$ 10mV	-	18	-	mA
IOL	Output Sink Current	V _{i-} - V _{i+} $\geq$ 10mV	-	20	-	mA

7. Comparator Electrical Characteristics ($T_A = -25^{\circ}\text{C}$ to $+85^{\circ}\text{C}$, $V_{CC} = 5\text{V}$)

Symbol	Parameter	Test Conditions	Min	Typ	Max	Unit
	Operation Voltage		3	—	5.5	V
V _{in}	Input Voltage	-40 ~ 85 $^{\circ}\text{C}$ ($\pm 5\%$)	-5%	0.775* VDD	+5%	V
V _{os}	Input Offset	Without calibration	-5mV	-	+5mV	mV
T _{pd}	Analog Comparator Response Time	Analog Comparator Hysteresis disable and with 10mV overdrive	-	-	1	μs
V _{hys}	Hysteresis Width		55	70	85	mV
I _{comp}	Power Consumption		90		100	μA

8. DPDMV/DMDM6B Characteristics ($T_A = -25^{\circ}\text{C}$ to $+85^{\circ}\text{C}$, $V_{CC} = 5\text{V}$)

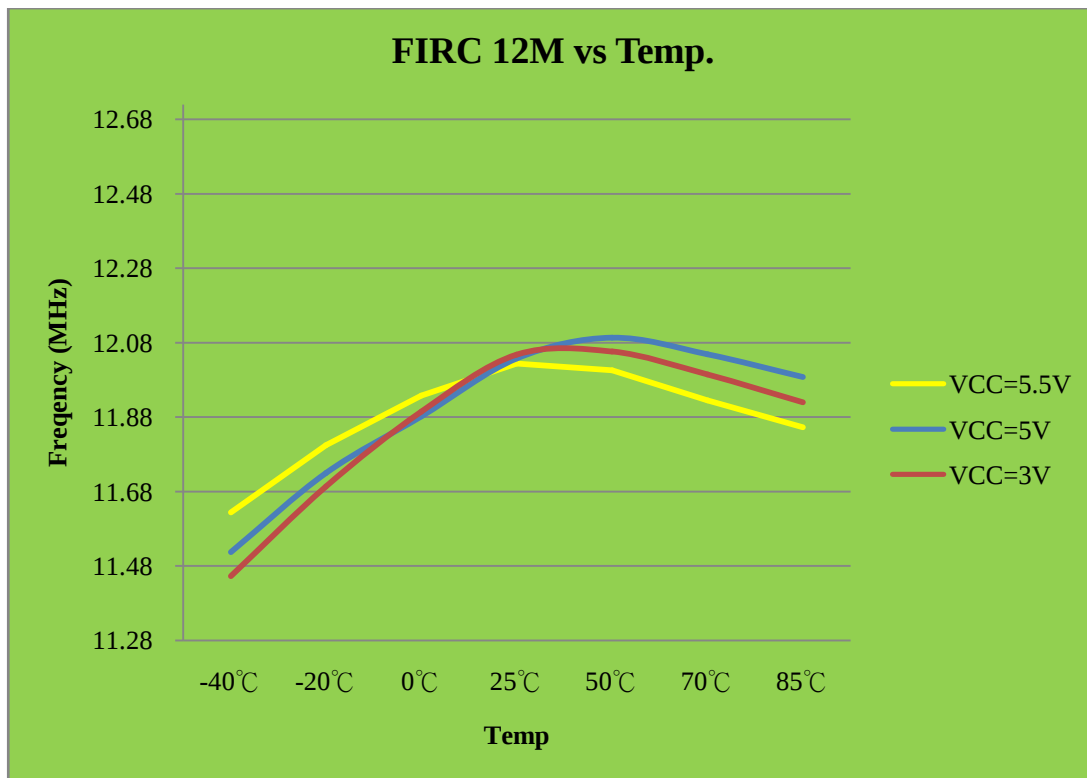
Symbol	Parameter	Test Conditions	Min	Typ	Max	Unit
I _{DP}	DP Current			100		μA
I _{DM}	DM Current			100		μA
R _{short}	DP and DM short switch resistance	I _{short} =200 μA	-	-	40	Ω
ΔV_{DP}	DP Voltage variation	I _{DP} =100 μA	-2	-	-3	mV
ΔV_{DM}	DM Voltage variation	I _{DM} =100 μA	-2	-	-3	mV
R _{sw}	DP and DM voltage level switch resistance	I _{sw} =200 μA	-	-	6	K Ω

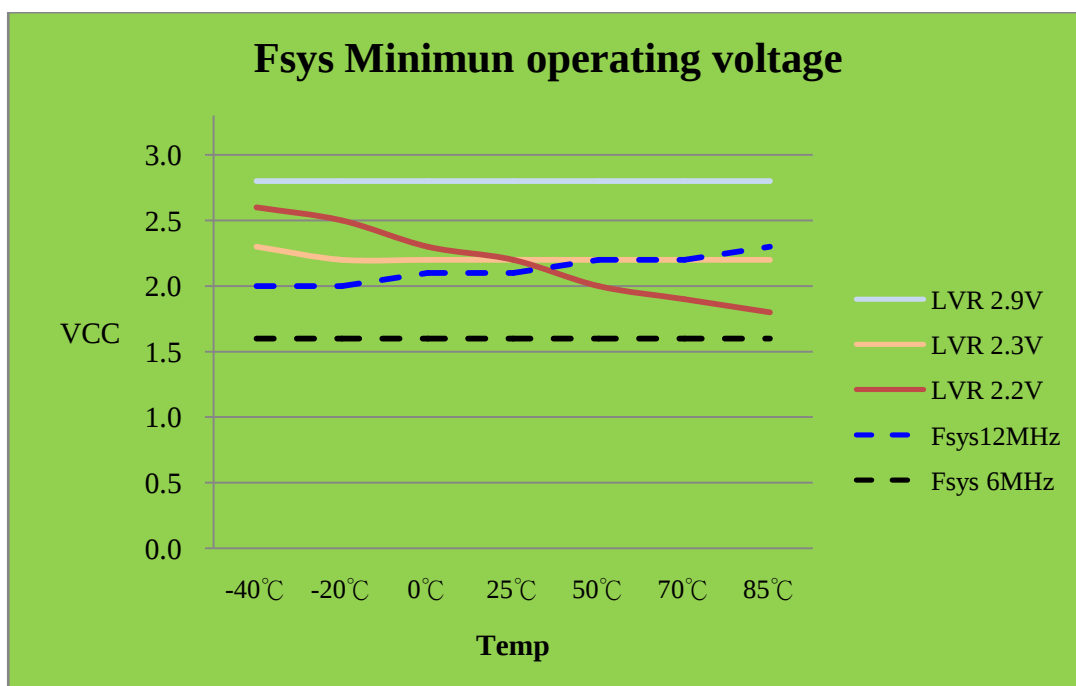
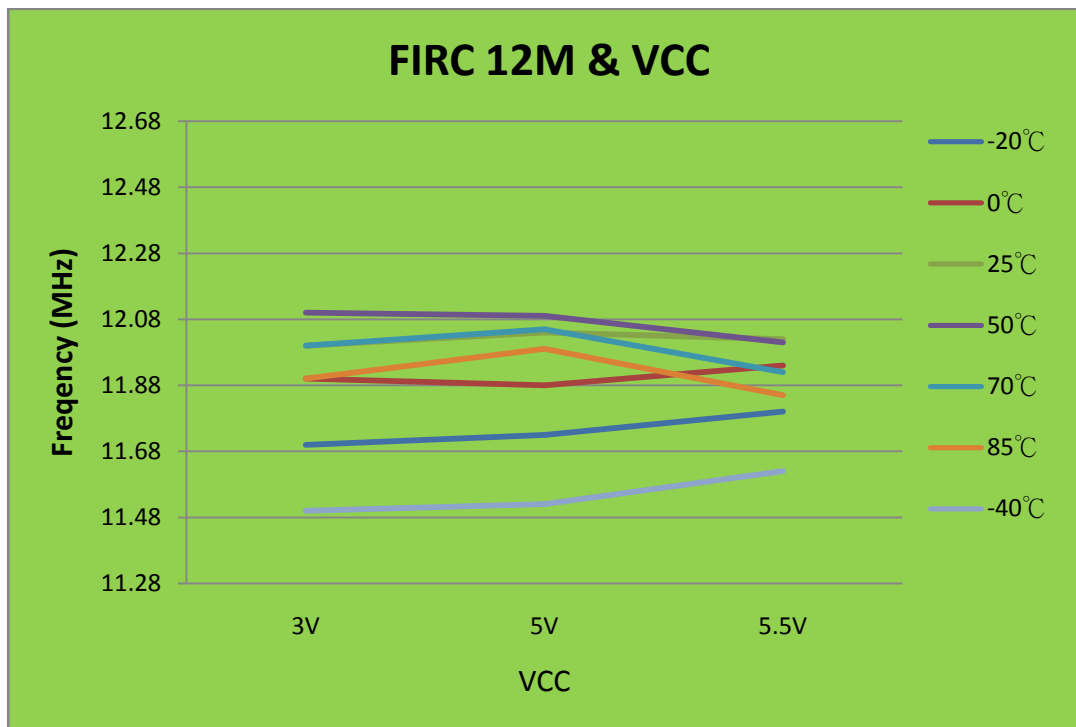
9. CC1/CC2 Characteristics ($T_A = -25^{\circ}\text{C}$ to $+85^{\circ}\text{C}$, $V_{CC} = 5\text{V}$)

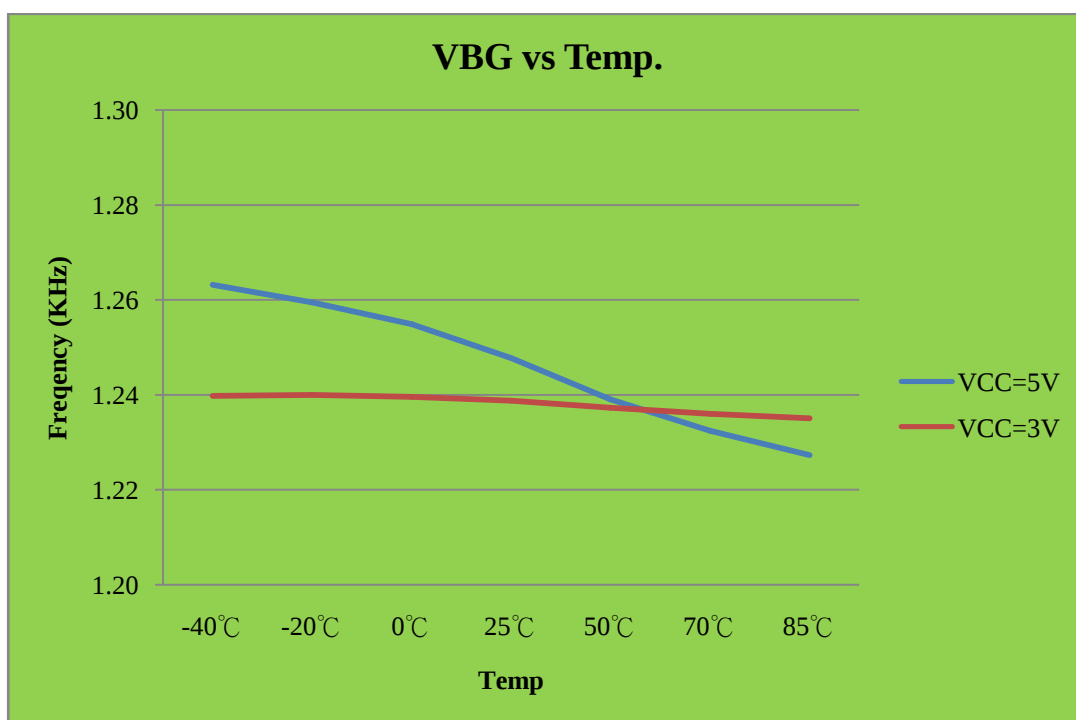
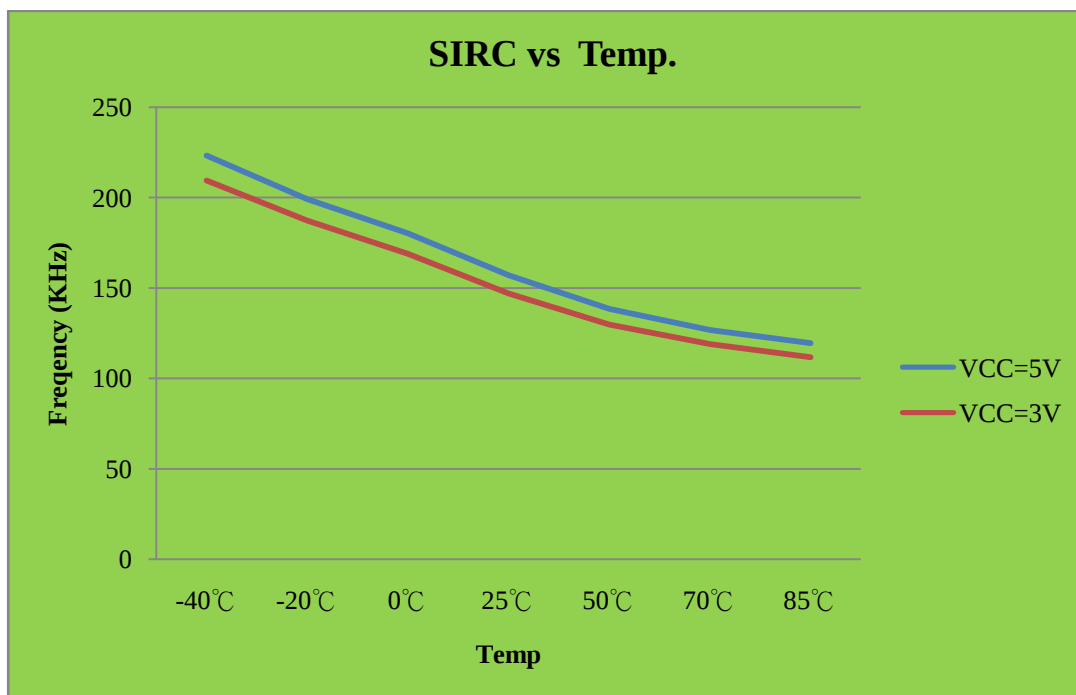
Symbol	Parameter	Test Conditions	Min	Typ	Max	Unit
R_{22k}	22K Ω resistance		-5%	21.3	+5%	Ω
R_{10k}	10K Ω resistance		-5%	10.8	+5%	Ω
$R_{5.1K}$	5.1K Ω resistance		-5%	5.5	+5%	Ω

10. LVR Circuit Characteristics ($T_A = -25^{\circ}\text{C}$ to $+85^{\circ}\text{C}$, $V_{CC} = 5\text{V}$)

Parameter	Symbol	Min	Typ	Max	Unit
LVR reference Voltage	V_{LVR}	-8%	2.2	+8%	V
		-3%	2.3	+3%	
		-3%	2.9	+3%	
LVR Hysteresis Voltage	V_{HYST}	—	± 0.2	—	V
Low Voltage Detection time	T_{LVR}	100	—	—	μs

11. Characteristic Graphs


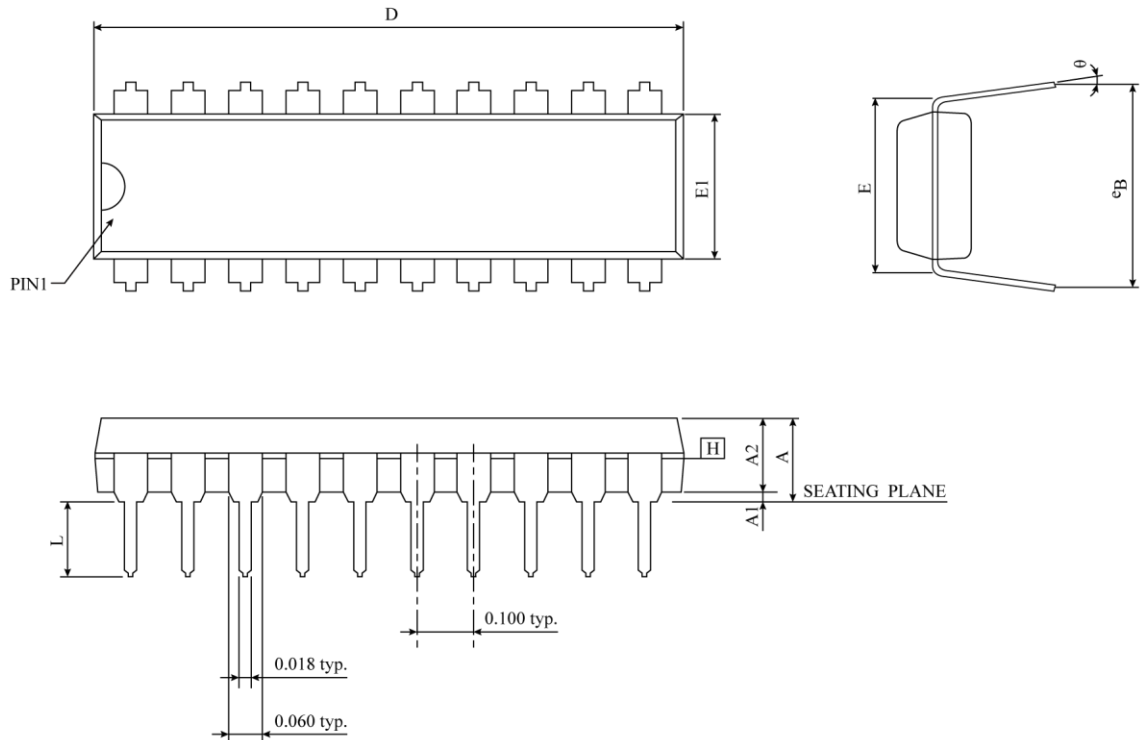




Package Information

The ordering information:

Ordering number	Package
TM57MA45/MA46-MTP	Wafer / Dice blank chip
TM57MA45/MA46-COD	Wafer / Dice with code
TM57MA45 -MTP-05	DIP 20-pin (300 mil)
TM57MA45 -MTP-12	SKINNY DIP 24-pin (300 mil)
TM57MA45 -MTP-13	SKINNY DIP 28-pin (300 mil)
TM57MA45 -MTP-21	SOP 20-pin (300 mil)
TM57MA45 -MTP-22	SOP 24-pin (300 mil)
TM57MA45-MTP-28	SSOP 24-pin (300 mil)
TM57MA45/MA46-MTP-23	SOP 28-pin (300 mil)

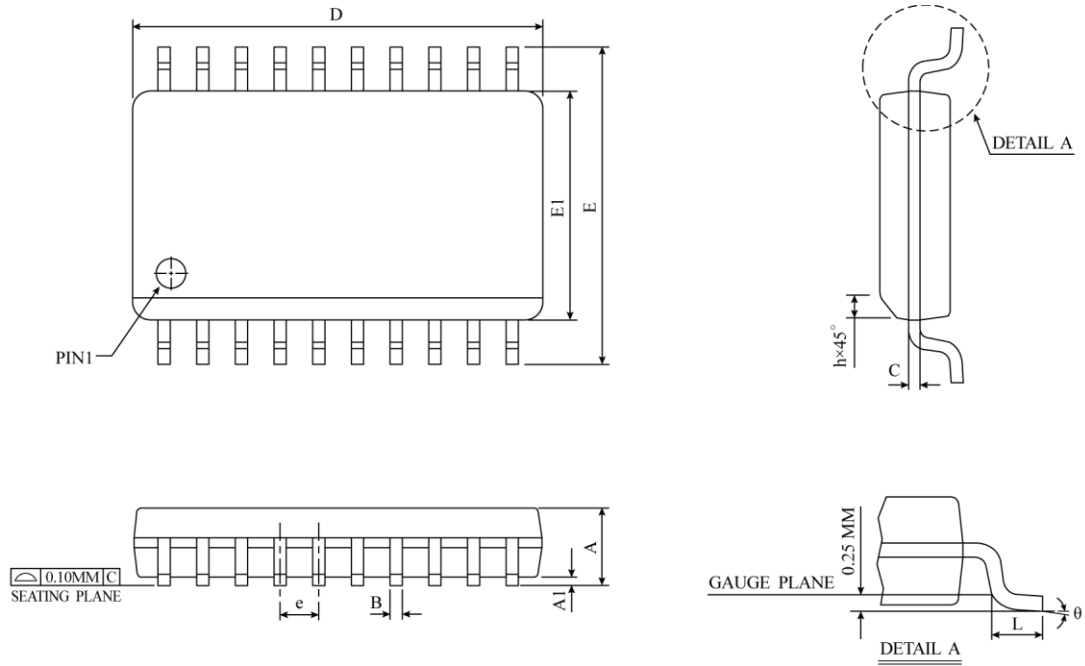
• DIP-20 (300mil) Package Dimension


SYMBOL	DIMENSION IN MM			DIMENSION IN INCH		
	MIN	NOM	MAX	MIN	NOM	MAX
A	-	-	4.445	-	-	0.175
A1	0.381	-	-	0.015	-	-
A2	3.175	3.302	3.429	0.125	0.130	0.135
D	25.705	26.061	26.416	1.012	1.026	1.040
E	7.620	7.747	7.874	0.300	0.305	0.310
E1	6.223	6.350	6.477	0.245	0.250	0.255
L	3.048	3.302	3.556	0.120	0.130	0.140
eB	8.509	9.017	9.525	0.335	0.355	0.375
θ	0°	7.5°	15°	0°	7.5°	15°
JEDEC	MS-001 (AD)					

NOTES :

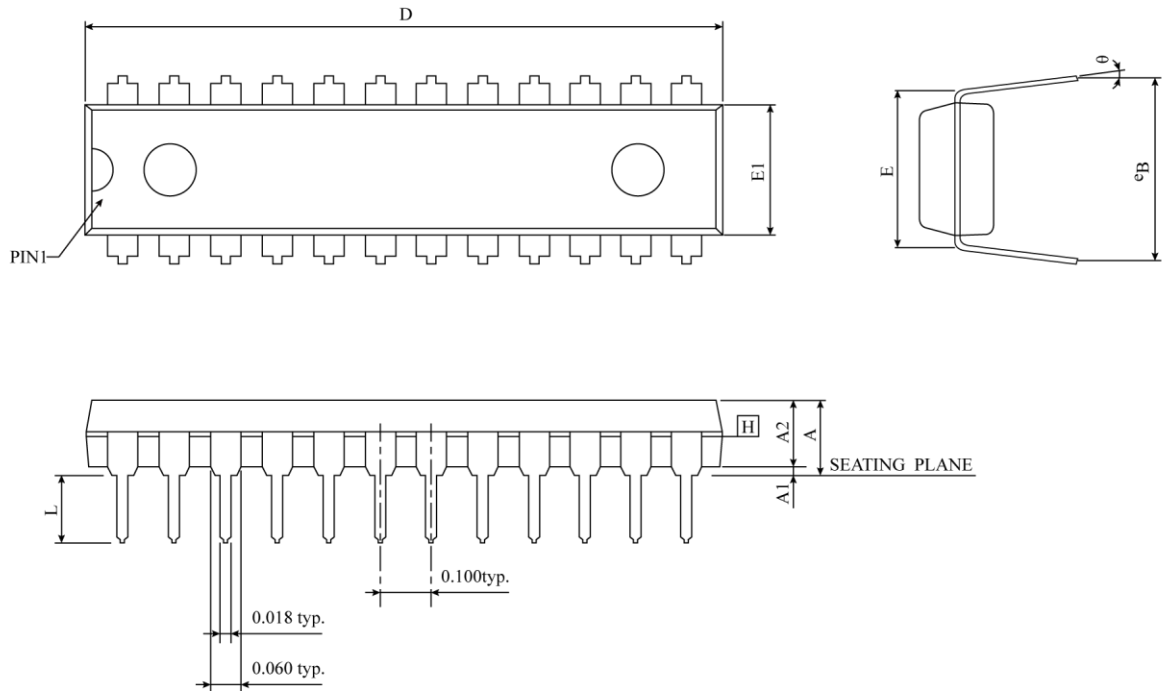
1. "D", "E1" DIMENSIONS DO NOT INCLUDE MOLD FLASH OR PROTRUSIONS. MOLD FLASH OR PROTRUSIONS SHALL NOT EXCEED .010 INCH.
2. eB IS MEASURED AT THE LEAD TIPS WITH THE LEADS UNCONSTRAINED.
3. POINTED OR ROUNDED LEAD TIPS ARE PREFERRED TO EASE INSERTION.
4. DISTANCE BETWEEN LEADS INCLUDING DAM BAR PROTRUSIONS TO BE .005 INCH MINIMUM.
5. DATUM PLANE  COINCIDENT WITH THE BOTTOM OF LEAD, WHERE LEAD EXITS BODY.

● SOP-20 (300mil) Package Dimension



SYMBOL	DIMENSION IN MM			DIMENSION IN INCH		
	MIN	NOM	MAX	MIN	NOM	MAX
A	2.35	2.50	2.65	0.0926	0.0985	0.1043
A1	0.10	0.20	0.30	0.0040	0.0079	0.0118
B	0.33	0.42	0.51	0.0130	0.0165	0.0200
C	0.23	0.28	0.32	0.0091	0.0108	0.0125
D	12.60	12.80	13.00	0.4961	0.5040	0.5118
E	10.00	10.33	10.65	0.3940	0.4425	0.4910
E1	7.40	7.50	7.60	0.2914	0.2953	0.2992
e	1.27 BSC			0.050 BSC		
h	0.25	0.50	0.75	0.0100	0.0195	0.0290
L	0.40	0.84	1.27	0.0160	0.0330	0.0500
θ	0°	4°	8°	0°	4°	8°
JEDEC	MS-013 (AC)					

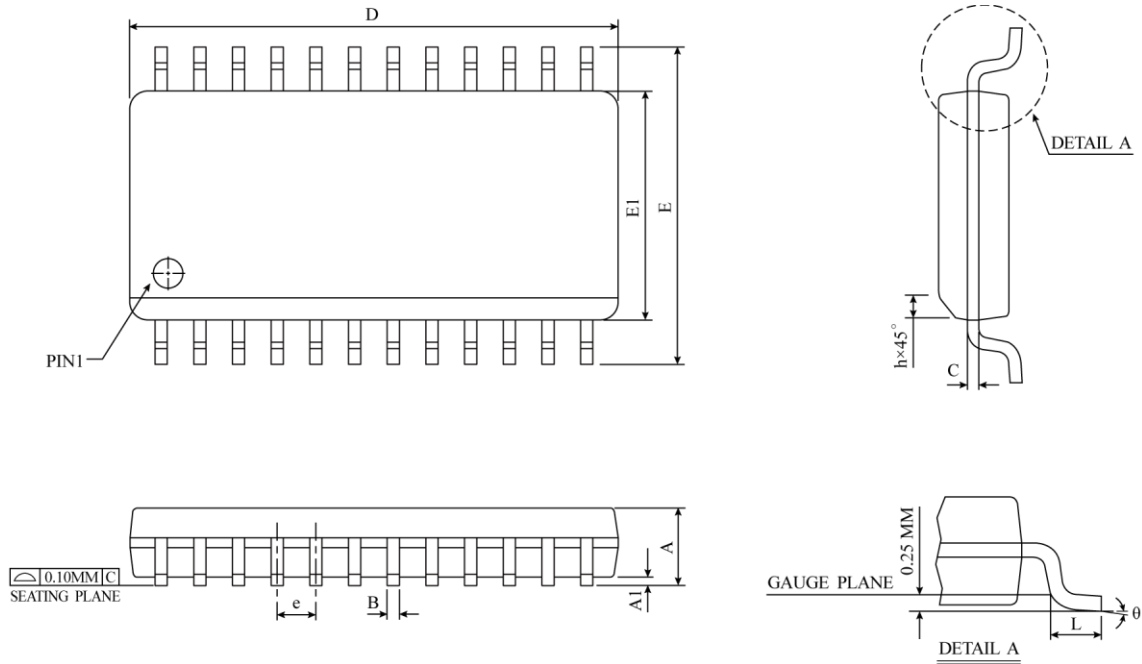
△ * NOTES : DIMENSION " D " DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS.
MOLD FLASH, PROTRUSIONS AND GATE BURRS SHALL
NOT EXCEED 0.15 MM (0.006 INCH) PER SIDE.

• Skinny DIP-24 (300mil) Package Dimension


SYMBOL	DIMENSION IN MM			DIMENSION IN INCH		
	MIN	NOM	MAX	MIN	NOM	MAX
A	-	-	5.334	-	-	0.210
A1	0.381	-	-	0.015	-	-
A2	3.175	3.302	3.429	0.125	0.130	0.135
D	31.242	31.877	32.512	1.230	1.255	1.280
E	7.620 BSC			0.300 BSC		
E1	6.426	6.553	6.680	0.253	0.258	0.263
L	2.921	3.366	3.810	0.115	0.133	0.150
eB	8.509	9.017	9.525	0.335	0.355	0.375
θ	0°	7.5°	15°	0°	7.5°	15°
JEDEC	MS-001 (AF)					

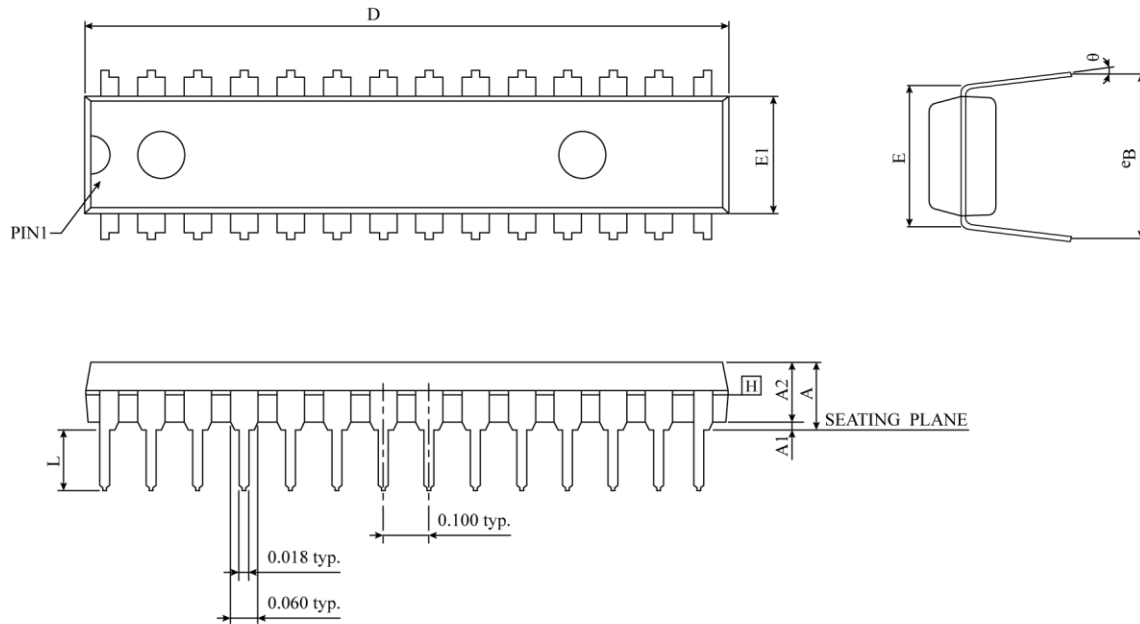
NOTES :

1. "D", "E1" DIMENSIONS DO NOT INCLUDE MOLD FLASH OR PROTRUSIONS. MOLD FLASH OR PROTRUSIONS SHALL NOT EXCEED .010 INCH.
2. eB IS MEASURED AT THE LEAD TIPS WITH THE LEADS UNCONSTRAINED.
3. POINTED OR ROUNDED LEAD TIPS ARE PREFERRED TO EASE INSERTION.
4. DISTANCE BETWEEN LEADS INCLUDING DAM BAR PROTRUSIONS TO BE .005 INCH MINIMUM.
5. DATUM PLANE $\square$ COINCIDENT WITH THE BOTTOM OF LEAD, WHERE LEAD EXITS BODY.

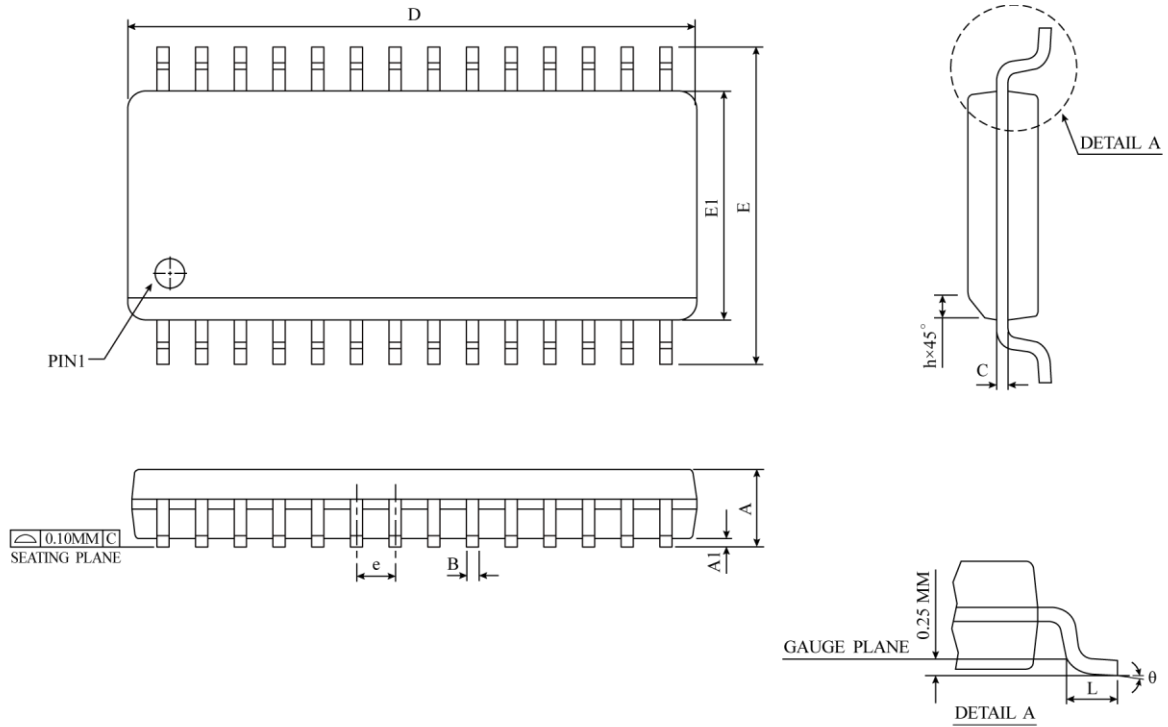
• SOP-24 (300mil) Package Dimension


SYMBOL	DIMENSION IN MM			DIMENSION IN INCH		
	MIN	NOM	MAX	MIN	NOM	MAX
A	2.35	2.50	2.65	0.0926	0.0985	0.1043
A1	0.10	0.20	0.30	0.0040	0.0079	0.0118
B	0.33	0.42	0.51	0.0130	0.0165	0.0200
C	0.23	0.28	0.32	0.0091	0.0108	0.0125
D	15.20	15.40	15.60	0.5985	0.6063	0.6141
E	10.00	10.33	10.65	0.3940	0.4425	0.4910
E1	7.40	7.50	7.60	0.2914	0.2953	0.2992
e	1.27 BSC			0.050 BSC		
h	0.25	0.50	0.75	0.0100	0.0195	0.0290
L	0.40	0.84	1.27	0.0160	0.0330	0.0500
θ	0°	4°	8°	0°	4°	8°
JEDEC	MS-013 (AD)					

△ * NOTES : DIMENSION "D" DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS.
MOLD FLASH, PROTRUSIONS AND GATE BURRS SHALL
NOT EXCEED 0.15 MM (0.006 INCH) PER SIDE.

• Skinny DIP-28 (300mil) Package Dimension


SYMBOL	DIMENSION IN MM			DIMENSION IN INCH		
	MIN	NOM	MAX	MIN	NOM	MAX
A	-	-	4.445	-	-	0.175
A1	0.381	-	-	0.015	-	-
A2	3.175	3.302	3.429	0.125	0.130	0.135
D	35.179	35.370	35.56	1.385	1.393	1.400
E	7.874 BSC			0.310 BSC		
E1	7.188	7.315	7.442	0.283	0.288	0.293
L	3.048	3.302	3.556	0.120	0.130	0.140
eB	8.382	8.954	9.525	0.330	0.353	0.375
θ	0°	7.5°	15°	0°	7.5°	15°
JEDEC	MS-015 (AH)					

• SOP-28 (300mil) Package Dimension


SYMBOL	DIMENSION IN MM			DIMENSION IN INCH		
	MIN	NOM	MAX	MIN	NOM	MAX
A	2.35	2.50	2.65	0.0926	0.0985	0.1043
A1	0.10	0.20	0.30	0.0040	0.0079	0.0118
B	0.33	0.42	0.51	0.0130	0.0165	0.0200
C	0.23	0.28	0.32	0.0091	0.0108	0.0125
D	17.70	17.90	18.10	0.6969	0.7047	0.7125
E	10.00	10.33	10.65	0.3940	0.4425	0.4910
E1	7.40	7.50	7.60	0.2914	0.2953	0.2992
e	1.27 BSC			0.050 BSC		
h	0.25	0.50	0.75	0.0100	0.0195	0.0290
L	0.40	0.84	1.27	0.0160	0.0330	0.0500
θ	0°	4°	8°	0°	4°	8°
JEDEC	MS-013 (AE)					

△ * NOTES : DIMENSION "D" DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS.
MOLD FLASH, PROTRUSIONS AND GATE BURRS SHALL
NOT EXCEED 0.15 MM (0.006 INCH) PER SIDE.