十速

TM57M5610/15

DATA SHEET

Rev 0.93

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	May, 2018	New Release.
0.91	May, 2018	1. P7, 18, 43, 46, 48: Modify the SPEC of POR/LVR, update characteristics graph
0.92	Nov, 2018	1. P7, 19, 47 ,49: Modify the SPEC of POR/LVR, Update characteristics graph 2. P10: Update programming pins 3. P9, 52, 54: Add TSSOP20 package type 4. P49, 51: Update FRC/SRC characteristics graph
0.93	May, 2020	1. P10, P26, P28, P39, P45: Remove PB3~0 Open-drain funcion

CONTENTS

FAMILY OVERVIEW

P/N	Typ. V_{BAT}	ROM	RAM bytes	I/O (max.)	RFC	Timers	8-bit PWM	LCD S x C (max.)	LCD Bias	Time-piece current	Others
TM57M5610	3.0V	1Kx14 MTP	96	12	3-ch	8-bit x1 21-bit x1	x1	12 x 3 11 x 4	1/2	1uA	LVR LBD WDT
TM57M5615	1.5V									1.5uA	
TM57M5620	3.0V	2Kx14 MTP	176	16	4-ch	8-bit x2 21-bit x1	x2	31 x 4	1/3	2.5uA	LVR LBD WDT
TM57M5625	1.5V							29 x 4	1/2, 1/3	2.8uA	
TM57M5640	3.0V	4Kx14 MTP	336	20	4-ch	8-bit x2 21-bit x1	x2	45 x 4	1/3	2.5uA	LVR LBD WDT
TM57M5645	1.5V							43 x 4	1/2, 1/3	2.8uA	

Note: No matter V_{BAT}=3V or V_{BAT}=1.5V, the typical LCD bias voltage is:

1/2 bias: VL1=1.5V, V_{LCD}=3V

1/3 bias: VL1=1.0V, VL2=2.0V, V_{LCD}=3V

FEATURES

1. **Operating Voltage :**
   - ● V_{BAT}=LVR~3.6V (M5610)
   - ● V_{BAT}=LVR~1.8V (M5615)
2. **Timepiece Current (CPU Off, LCD On, 32K crystal oscillating) :**
   - ● M5610: 5uA @V_{DD}=3V, V_{BAT}=3V, without power saving
   - ● M5610: 1uA @V_{DD}=1.5V, V_{BAT}=3V, with power saving
   - ● M5615: 1.5uA @V_{DD}=1.5V, V_{BAT}=1.5V
3. **Program ROM: 1K x 14 bits MTP (Multi Time Programmable ROM)**
4. **RAM: 96 x 8 bits**
5. **STACK: 5 Levels**
6. **I/O ports: Maximum 12 programmable I/O pins**
   - ● Open-Drain Output
   - ● CMOS Push-Pull Output
   - ● Schmitt Trigger Input with pull-up resistor option
7. **System Oscillation Sources (Fsys) :**
   - ● Fast-clock
     - FIRC (Fast Internal RC) : 3.8MHz @V_{DD}=3V; 1.3MHz @V_{DD}=1.5V
   - ● Slow-clock
     - SIRC (Slow Internal RC) : 80KHz @V_{DD}=3V; 40KHz @V_{DD}=1.5V
     - SXT (Slow Crystal) : 32768 Hz
   - ● System Oscillation Sources can be divided by 1/2/4/8 as System Clock (Fsys)
   - ● Dual System Clock Switching between Fast-clock and Slow-clock
     - FIRC + SIRC
     - FIRC + SXT
8. **Power Saving Operation Mode**
   - ● FAST Mode: CPU running at Fast-clock
   - ● SLOW Mode: Fast-clock stop, CPU running at Slow-clock
   - ● IDLE Mode: Fast-clock and CPU stop; Slow-clock, Timer2 and LCD keep running
   - ● STOP Mode: All clocks stop
9. **Resistance to Frequency Converter (RFC)**
10. **Two Independent Timers**
    - ● Timer0 (TM0)
      - 8-bit timer with divided by 1~256 pre-scale option, reload/interrupt/stop function
      - Clock sources: Fsys or Slow-clock /16 (SIRC/SXT)

- Timer2 (T2)
  - 21-bit timer with 4 interrupt time period options (60s/1s/0.5s/0.125s)
  - Clock sources: Fsys /128 or Slow-clock (SIRC/SXT)
  - IDLE mode wake-up, if clock source is Slow-clock

11. Interrupts

- Three External Interrupt pins (INT0~INT2)
  - Rising or falling edge triggered interrupt
  - Wake-up CPU from IDLE/STOP mode
- Timer0/Timer2 Interrupts
- RFC overflow Interrupt

12. LCD Controller / Driver

- 1/3 or 1/4 Duty
- 4 COM x 11 SEG or 3 COM x 12 SEG
- 1/2 LCD Bias voltage, typical VL1=1.5V and V_{LCD}=3V

13. Watchdog Timer (WDT)

- Clocked by built-in RC oscillator with 2 adjustable reset times
  - 0.8s/0.4s @V_{DD}=3V
  - 1.6s/0.8s @V_{DD}=1.5V
- Watchdog timer is disabled in IDLE/STOP mode

14. 8-bit PWM for Buzzer / IR application

- Adjustable Period & Clock Pre-scale
- Clock source: Fast-clock or Slow-clock

15. Four types Reset

- Power On Reset :M5610 (2V@25°C , 2.4V@-40°C) , M5615 (1.29V@25°C , 1.40V@0°C)
- Watchdog Reset
- Low Voltage Reset: M5610=1.7V, M5615=1.25V (M5615 is recommended to disable LVR function by LVROFF after power on)
- External Pin Reset

16. Low Battery Detector (LBD) by BandGap Voltage Reference

- M5610: Detect V_{BAT} from 2.4V to 3.1V
- M5615: Detect V_{BAT} from 1.2V to 1.6V

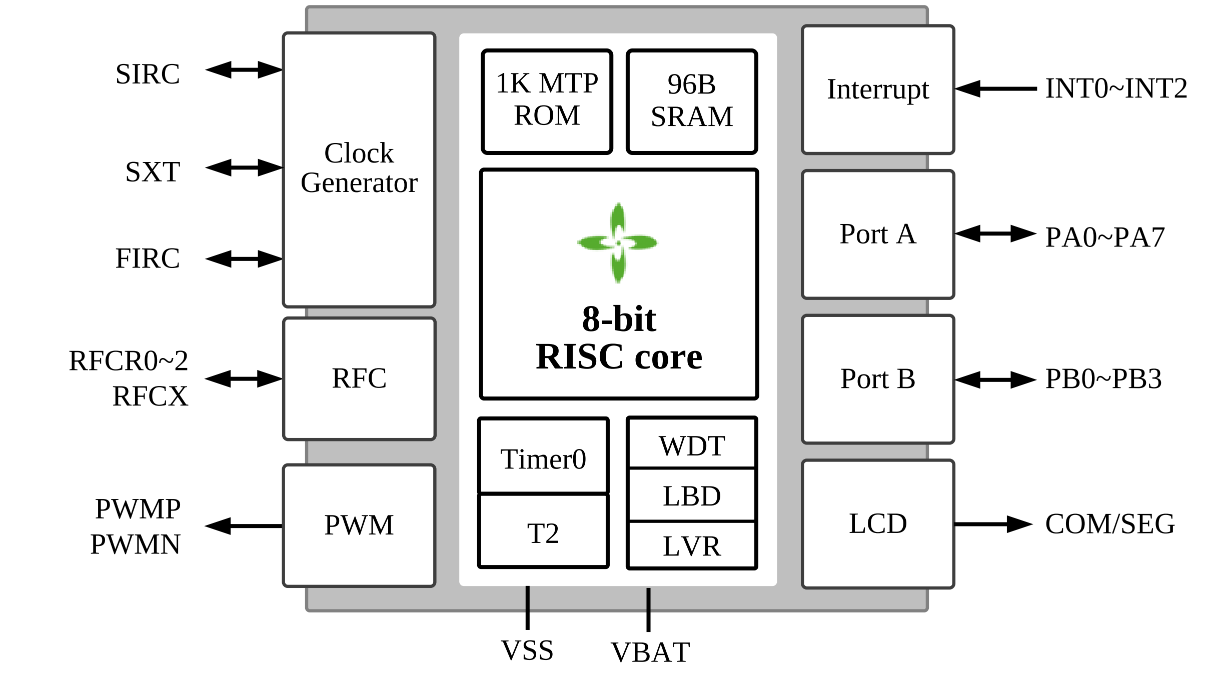
17. Operating Temperature Range :

- M5610: 40°C to + 85°C M5615: 0°C to + 85°C

18. Package Type : SSOP28/ TSSOP20 / dice-form

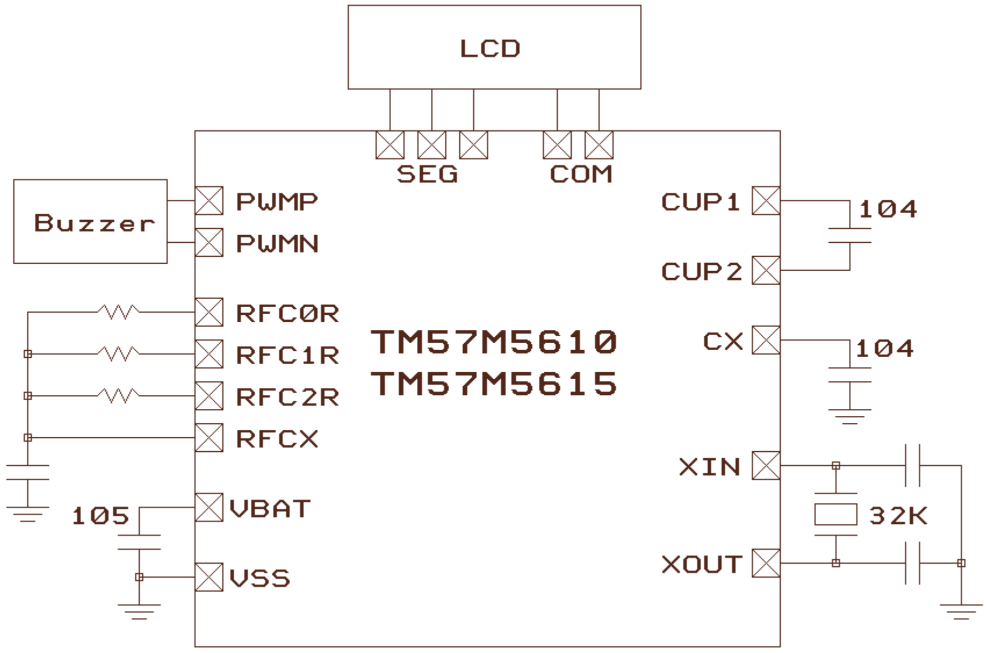
19. EV Board: EV8222

BLOCK DIAGRAM

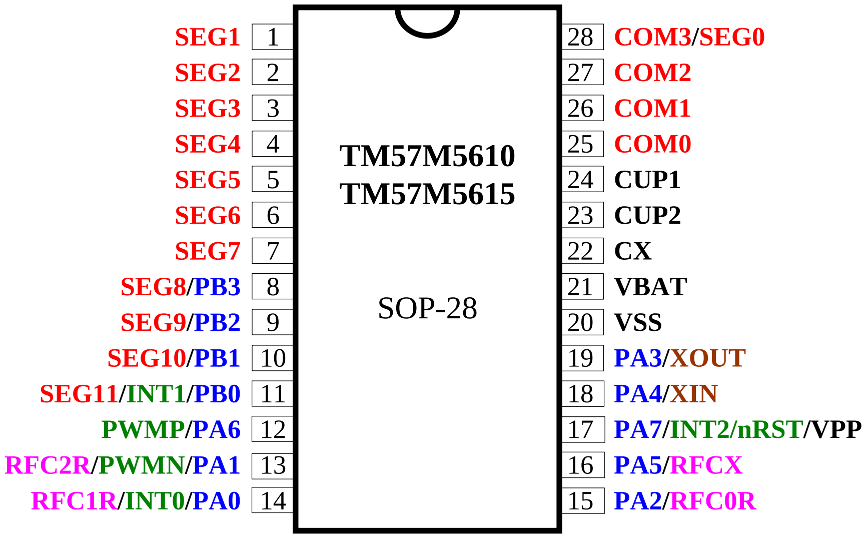


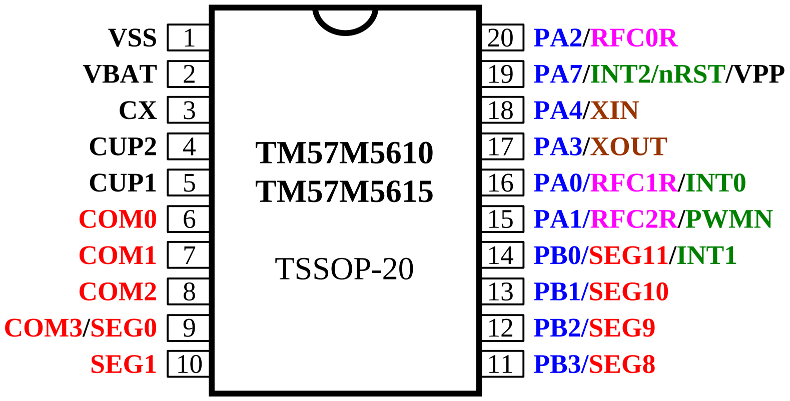
TM57M5610/5615 Block Diagram

APPLICATION CIRCUIT



PIN ASSIGNMENT





PIN DESCRIPTIONS

Name	In/Out	Pin Description
PA0-PA6	I/O	Bit-programmable I/O port for Schmitt-trigger input, “CMOS push-pull” output or “Open-Drain” output. Pull-up Resistors are assignable by software.
PB0-PB3	I/O	Bit-programmable I/O port for Schmitt-trigger input without pull-up, “CMOS push-pull”output
PA7	I/O	Bit-programmable I/O port for Schmitt-trigger input, or “Open-Drain” output. Pull-up resistors are assignable by software.
nRST	I	External active low reset with internal pull-high
INT0–INT2	I	External interrupt input
RFC0R~RFC2R	O	RFC resistor connection pin
RFCX	I	RFC clock input pin
COM0~COM3	O	LCD common output
SEG0~SEG11	O	LCD segment output
PWMP, PWMN	O	8-bit PWM0 output
CX, CUP1, CUP2	–	LCD bias capacitor connection pin
XIN, XOUT	–	Crystal / Resonator oscillator connection for system clock.
VPP	I	MTP programming high voltage input
VBAT, VSS	P	Power Voltage input pin and ground

Note: Programming pins are list below. It is better to remove the PCB components connected to these pins during In-Circuit-Programming.

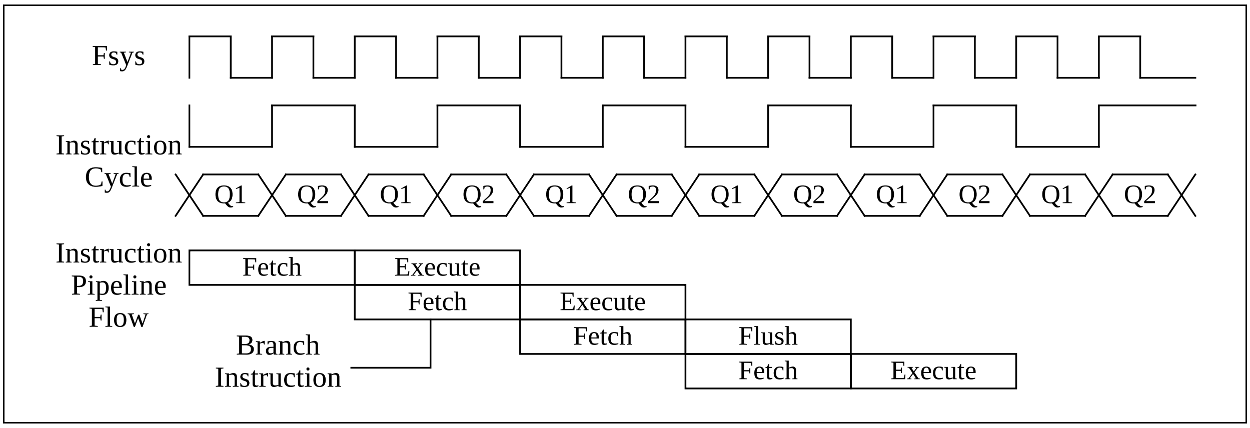
9 wire mode (M5615): VCC/VSS/PA0/PA1/PA2/PA3/PA4/PA7 (VPP)/CX
6 wire mode (M5615): VCC/VSS/PA0/PA1/PA7 (VPP) /CX
8 wire mode (M5610): VCC/VSS/PA0/PA1/PA2/PA3/PA4/PA7 (VPP)
5 wire mode (M5610): VCC/VSS/PA0/PA1/PA7 (VPP)

FUNCTIONAL DESCRIPTION

1. CPU Core

1.1 Clock Scheme 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.



1.2 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.

1.3 Programming Counter (PC) and Stack

The Programming Counter is 10-bit wide capable of addressing a 1K 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 Vector (001h) are provided for PC initialization and Interrupt. For CALL / GOTO instructions, PC loads 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 [9:8] keeps unchanged. Therefore, the data of a lookup table must be located with the same PC [9:8].

The STACK is 10-bit wide and 5-level in depth. The CALL instruction and hardware interrupt will push STACK level in order. While the RET / RETI / RETLW instructions pop the STACK level in order.

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

◇Example: To look up the MTP data located "TABLE"

```
            ORG       000H           ; Reset Vector
            GOTO      START          ; Goto user program address
   START:
            MOVLW     00H
            MOVWF     INDEX          ; Set lookup table's address (INDEX)
   LOOP:
            MOVFW     INDEX          ; Move INDEX value to W register
            CALL      TABLE          ; To Lookup data (W = 55H when INDEX = 00H)
            …
            INCF      INDEX, 1       ; Increment the INDEX for next address
            …
            GOTO      LOOP           ; Goto LOOP label

            ORG       X00H           ; X = 1, 2, 3
   TABLE:
            ADDWF     PCL, 1         ; (Addr = X00H) 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
```

Note: The chip defines 256 ROM addresses as one page, so that ROM has four pages, 000H~0FFH, 100H~1FFH, 200H~2FFH, and 300H~3FFH. On the other words, PC[9:8] can be defined as page. A lookup table must be located at the same page to avoid getting wrong data. Thus, the lookup table has maximum 255 data for above example with starting a lookup table at X00H (X = 1, 2, 3). If a lookup table has fewer data, it needs not setting the starting address at X00H, but only confirms all lookup table data are located at the same page.

◇Example: To look up the MTP data located “TABLE” by TABRL and TABRH instructions

```
        ORG     000H                ; Reset Vector
        GOTO    START               ; Goto user program address
START:
        MOVLW   (TABLE >>8) & 0xff  ; Get high byte address of TABLE label
        MOVWF   DPH                 ; DPH (F17.1~0) = 02H
        MOVLW   (TABLE) & 0xff      ; Get low byte address of TABLE label
        MOVWF   DPL                 ; DPL (F04.7~0) = 80H
LOOP:
        TABRL                       ; W = 86H when DTPR = {DPH, DPL} = 0280H
        TABRH                       ; W = 19H when DTPR = {DPH, DPL} = 0280H
        …
        INCF    DPL, 1              ; Increment the DPL for next address
        …
        GOTO    LOOP                ; Goto LOOP label

        ORG     280H
TABLE:
        DT      0x1986              ; 14-bit ROM data
        DT      0x3719              ; 14-bit ROM data
```

F02	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCL	PCL							
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

F02.7~0 **PCL:** Low-byte of Program Counter (PC[7:0])

F0A	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCH	–	–	–	–	–	–	PCH	
R/W	–	–	–	–	–	–	R	R
Reset	–	–	–	–	–	–	0	0

F0A.1~0 **PCH:** 2 MSBs of Program Counter (PC[9:8])

F1D	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DPL	DPL							
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

F1D.7~0 **DPL:** Table read low address, data ROM pointer (DPTR[7:0])

F1E	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DPH	–	–	–	–	–	–	DPH	
R/W	–	–	–	–	–	–	R/W	R/W
Reset	–	–	–	–	–	–	0	0

F1E.1~0 **DPH:** 2 MSBs of Table read high address, data ROM pointer (DPTR[9:8])

1.4 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.

F03	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STATUS	GB2	GB1	GB0	TO	PD	Z	DC	C
Reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W

<table>
<tr><th>Bit</th><th colspan="2">Description</th></tr>
<tr><td>7</td><td colspan="2"><b>GB2</b>: General Purpose Bit 2</td></tr>
<tr><td>6</td><td colspan="2"><b>GB1</b>: General Purpose Bit 1</td></tr>
<tr><td>5</td><td colspan="2"><b>GB0:</b> General Purpose Bit 0</td></tr>
<tr><td>4</td><td colspan="2"><b>TO</b>: Time Out Flag
0: after Power On Reset, LVR Reset, or CLRWDT/SLEEP instruction
1: WDT time out occurs</td></tr>
<tr><td>3</td><td colspan="2"><b>PD</b>: Power Down Flag
0: after Power On Reset, LVR Reset, or CLRWDT instruction
1: after SLEEP instruction</td></tr>
<tr><td>2</td><td colspan="2"><b>Z</b>: Zero Flag
0: the result of a logic operation is not zero
1: the result of a logic operation is zero</td></tr>
<tr><td rowspan="3">1</td><td colspan="2"><b>DC</b>: Decimal Carry Flag or Decimal/Borrow Flag</td></tr>
<tr><td>ADD instruction</td><td>SUB instruction</td></tr>
<tr><td>0: no carry
1: a carry from the low nibble bits of the result occurs</td><td>0: a borrow from the low nibble bits of the result occurs
1: no borrow</td></tr>
<tr><td rowspan="3">0</td><td colspan="2"><b>C</b>: Carry Flag or/Borrow Flag</td></tr>
<tr><td>ADD instruction</td><td>SUB instruction</td></tr>
<tr><td>0: no carry
1: a carry occurs from the MSB</td><td>0: a borrow occurs from the MSB
1: no borrow</td></tr>
</table>

◇Example: Write immediate data into STATUS register

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

◇Example: Bit addressing set and clear STATUS register

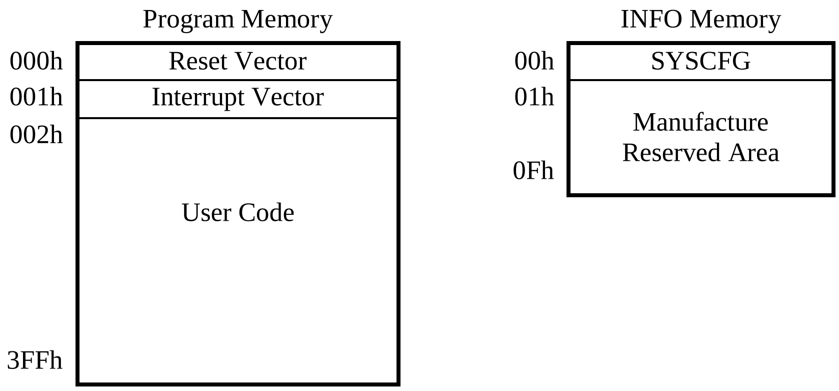
```
BSF       STATUS, 0           ; Set C = 1
BCF       STATUS, 0           ; Clear C = 0
```

◇Example: Determine the C flag by BTFSS instruction

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

2. Program ROM (MTP)

The MTP Program ROM of this device is 1K words, with an extra INFO area to store the SYSCFG and manufacture data. The MTP 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 cleared or MTP ROM is blank. That is, unprotect the PROTECT bit can be done only if the Program ROM area is blank. The tenx certified writer can do the above actions with the sophisticated software.



The System Configuration Register (SYSCFG) is located at MTP INFO area. The SYSCFG determines the option for initial condition of MCU. It is written by MTP Writer only. User can select chip operation mode by SYSCFG register.

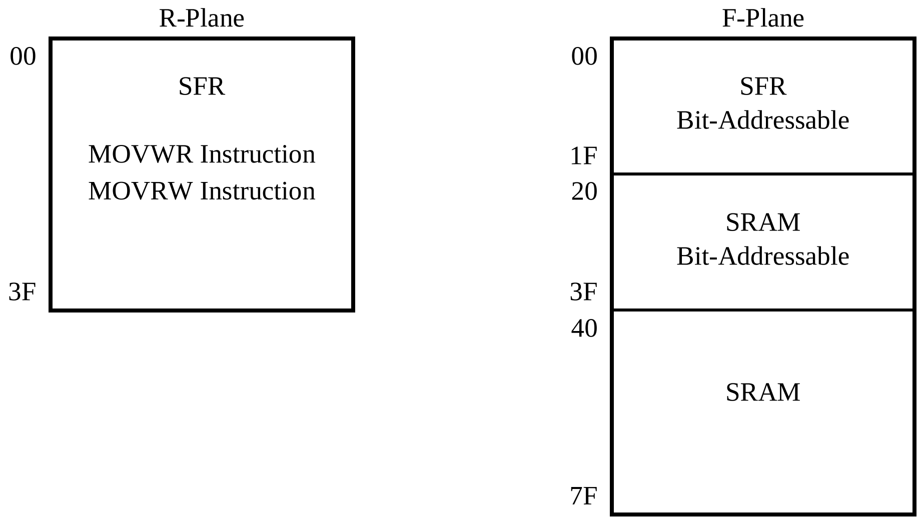
Bit	Description	
13	**PROTECT**: Code protection selection	
	1	Enable
	0	Disable
12	**XRSTE**: External Pin (PA7) Reset Enable	
	1	Enable
	0	Disable (PA7 as input I/O pin)
11~10	**LVR**: Low Voltage Reset	
9	**WDTE**: WDT Reset Enable	
	1	Enable
	0	Disable
8~0	Tenx Reserved	

3. Data Memory (RAM and SFR)

There are two Data Memory Planes in the chip, F-Plane and R-Plane.

The lower locations of F-Plane are reserved for Special-Function-Register (SFR). Above the SFR is General Purpose Data Memory, implemented as 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.

R-Plane can also be addressed directly or indirectly. Indirect Addressing is made by INDR register. 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 R-Plane is not bit-addressable and only supports the MOVWR, MOVRW byte operating instructions.



F-Plane	8/0	9/1	A/2	B/3	C/4	D/5	E/6	F/7
00h	INDF	TM0	PCL	STATUS	FSR	PAD	PBD	
08h	INTIE	INTIF	PCH	CLKCTL	MF0C	PWMDTY	LBDCTL	RFCTL
10h	LCDCTL	RFCNTH	RFCNTL					
18h					RSR	DPL	DPH	

R-Plane	8/0	9/1	A/2	B/3	C/4	D/5	E/6	F/7
00h	INDR	TM0RLD	TM0CTL	PWRDN	WDTCLR	PAMODH	PAMODL	PBMODL
08h	PWMCTL	PWMPRD	LVROFF					
20h	LCDRAM							

◇Example: Write immediate data into R-Plane register

```
MOVLW     AAH              ; Move immediate AAH into W register
MOVWR     05H              ; Move W value into R-Plane location 05H
```

◇Example: Move R-Plane location 20H data into W register

```
MOVRW     20H              ; To get a content of R-Plane location 20H to W
```

◇Example: Clear R-Plane by indirectly addressing mode

```
            MOVLW     20H              ; W = 20H
            MOVWF     RSR              ; Set R-Plane address to RSR register
  LOOP:
            MOVLW     00H
            MOVWR     INDR             ; Clear R-Plane 20H
```

◇Example: Clear F-Plane RAM data by indirectly addressing mode

```
            MOVLW     20H              ; W = 20H (SRAM start address)
            MOVWF     FSR              ; Set start address of user SRAM into FSR register
  LOOP:
            MOVLW     00H
            MOVWF     INDF             ; Clear user SRAM data
            INCF      FSR, 1           ; Increment the FSR for next address
            MOVLW     80H              ; W = 80H (SRAM end address)
            XORWF     FSR, 0           ; Check the FSR is end address of user SRAM?
            BTFSS     STATUS, Z        ; Check the Z flag
            GOTO      LOOP             ; If Z = 0, goto LOOP label
            …                          ; If Z = 1, exit LOOP
```

F00	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INDF	INDF							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	–	–	–	–	–	–	–

F00.7~0 **INDF:** Not a physical register, addressing INDF actually point to the F-Plane register whose address is contained in the FSR register

F04	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FSR	GB3	FSR						
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

F04.7 **GB3:** General purpose bit

F04.6~0 **FSR:** F-Plane file select register, indirect address mode pointer

F1C	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RSR	RSR							
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

F1C.7~0 **RSR:** R-Plane file select register, indirect address mode pointer

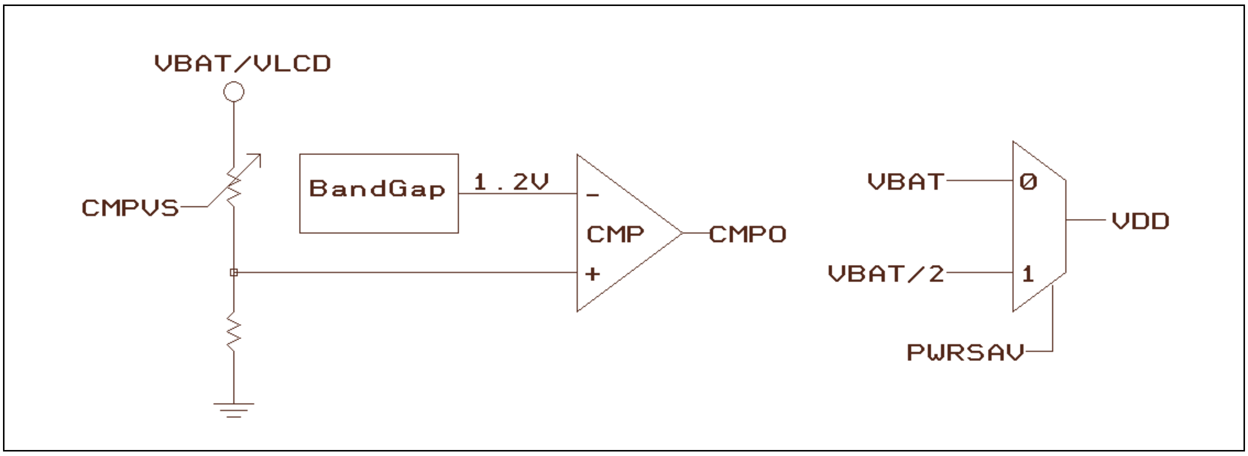
R00	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INDR	INDR							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	–	–	–	–	–	–	–

R00.7~0 **INDR:** Not a physical register, addressing INDR actually point to the R-Plane register whose address is contained in the RSR register

4. Power Management

V_{BAT} is the power supply for this chip. The typical condition is V_{BAT}=3V for M5610 and V_{BAT}=1.5V for M5615. V_{DD} is the internal voltage level for chip operation. User must keep 1.3V<V_{DD}<3.6V for the device's proper operation. The "PWRSAV" control bit can set V_{DD}=V_{BAT}/2 for M5610, but it's not valid for M5615.

The internal 1.2V BandGap module provides accuracy voltage reference for the Low Battery Detection (LBD) function. The V_{BAT} (M5610) or V_{LCD} (M5615) is divided by resistors to a certain level then compare to the BandGap voltage. Since M5615 compare V_{LCD} for LBD (V_{LCD}=V_{BAT}*2), user must turn on LCD and wait V_{LCD} stable before using this function. The BandGap and Comparator consume un-neglect current, so user should not use them too often. Because V_{BAT} voltage level changes very slowly, user can detect it once an hour or once a day to reduce current consumption.



F0E	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LBDCTL	CMPO	CMPVS			–	PWRSAV	PUMPCKS	–
R/W	R	R/W	R/W	R/W	–	R/W	R/W	–
Reset	–	0	0	0	–	0	0	–

F0E.7 **CMPO:** Compare result of BandGap voltage and V_{BAT}/V_{LCD} voltage divider. CMPO=1 means the V_{BAT}/V_{LCD} divider voltage is higher. ***Note:*** M5615 must turn on LCD for this function.

F0E.6~4 **CMPVS:** Select V_{BAT}/V_{LCD} resistor divider for Comparator input to compare with the 1.2V Bandgap reference voltage.
000: Comparator and Bandgap Disable
001: detect if M5610's V_{BAT}>2.4V; detect if M5615's V_{BAT}>1.20V
010: detect if M5610's V_{BAT}>2.5V; detect if M5615's V_{BAT}>1.25V
011: detect if M5610's V_{BAT}>2.6V; detect if M5615's V_{BAT}>1.30V
100: detect if M5610's V_{BAT}>2.7V; detect if M5615's V_{BAT}>1.35V
101: detect if M5610's V_{BAT}>2.8V; detect if M5615's V_{BAT}>1.40V
110: detect if M5610's V_{BAT}>2.9V; detect if M5615's V_{BAT}>1.45V
111: detect if M5610's V_{BAT}>3.0V; detect if M5615's V_{BAT}>1.50V

F0E.2 **PWRSAV:** Power saving control for M5610. ***Note:*** M5615 must keep PWRSAV=0
0: Disable, V_{DD}=V_{BAT}
1: Enable, V_{DD}=V_{BAT}/2

5. Reset

This device can be reset in four ways. The TO and PD flags at status register (STATUS) can indicate system reset status. The SYSCFG controls the Reset functionality.

- Power-On-Reset (POR)

  M5610 (2V@25°C , 2.4V@-40°C)

  M5615 (1.29V@25°C , 1.4V@0°C)

- Low Voltage Reset (LVR)

  M5610: 1.7V; M5615: 1.25V (Default enable, can be disable by FW setting)

  (M5615 is recommended to disable LVR function by LVROFF after power on)

- External Pin Reset (PA7)
- Watchdog Reset (WDT)

  Clocked by built-in RC oscillator

  0.8 second or 0.4 second @V_{DD}=3V; 1.6 second or 0.8 second @V_{DD}=1.5V

  Runs in Fast/Slow mode, stops in IDLE/STOP mode

F03	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STATUS	GB2	GB1	GB0	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

F03.4 **TO:** WDT Time Out Flag
0: after Power On Reset, LVR Reset, or CLRWDT / SLEEP instructions
1: WDT time out occurs

F03.3 **PD:** Power Down Flag
0: after Power On Reset, LVR Reset, or CLRWDT instruction
1: after SLEEP instruction

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 (=CLRWDT instruction)

R0A	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LVROFF	LVROFF							
R/W	W							
Reset	–	–	–	–	–	–	–	–

R0A.7~0 **LVROFF:** Write this register with 0x37 to force LVR disable
(We proposed M5615 to disable LVR after power on)

R08	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMCTL	PWMCKS	T2PSC		PWMPSC			PWMNOE	WDTPSC
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	0	0	0	0	0	0	0

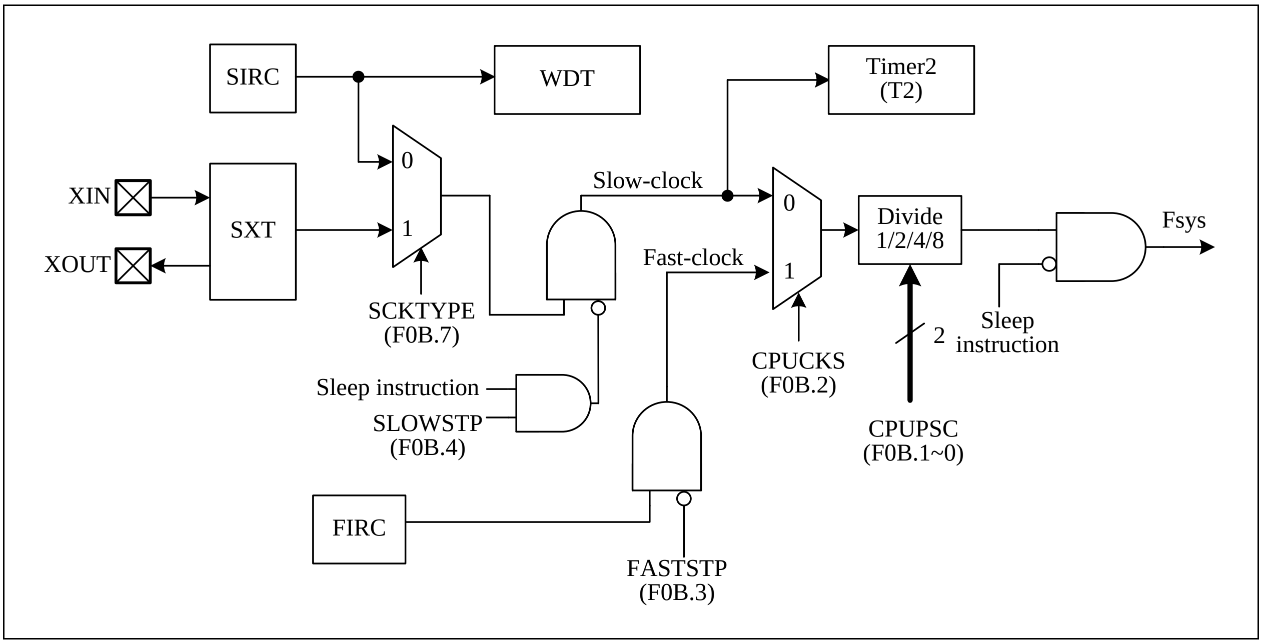
R08.1~0 **WDTPSC:** WDT timeout select
0: 0.8 second @V_{DD}=3V, 1.6 second @V_{DD}=1.5V
1: 0.4 second @V_{DD}=3V, 0.8 second @V_{DD}=1.5V

6. Clock Circuitry and Operation Mode

There are three kinds of system clock source.

- **SIRC** (Slow Internal RC, 80KHz @V_{DD}=3V, 40KHz @V_{DD}=1.5V)
- **SXT** (Slow Crystal, 32KHz)
- **FIRC** (Fast Internal RC, 3.8MHz @V_{DD}=3V, 1.3MHz @V_{DD}=1.5V)

The device is designed with dual-clock system. During runtime, user can directly switch the System clock between Fast-clock (FIRC) and Slow-clock (SIRC or SXT). It also can directly select a clock divider of 1, 2, 4, or 8. The CLKCTL (F0B) SFR controls the System clock operating. H/W automatically blocks the S/W abnormally setting for this register. S/W can only change the Slow-clock type in Fast mode. Never to write both FASTSTP=1 & CPUCKS=1. It is recommended to write this SFR bit by bit.



Clock Scheme Block Diagram

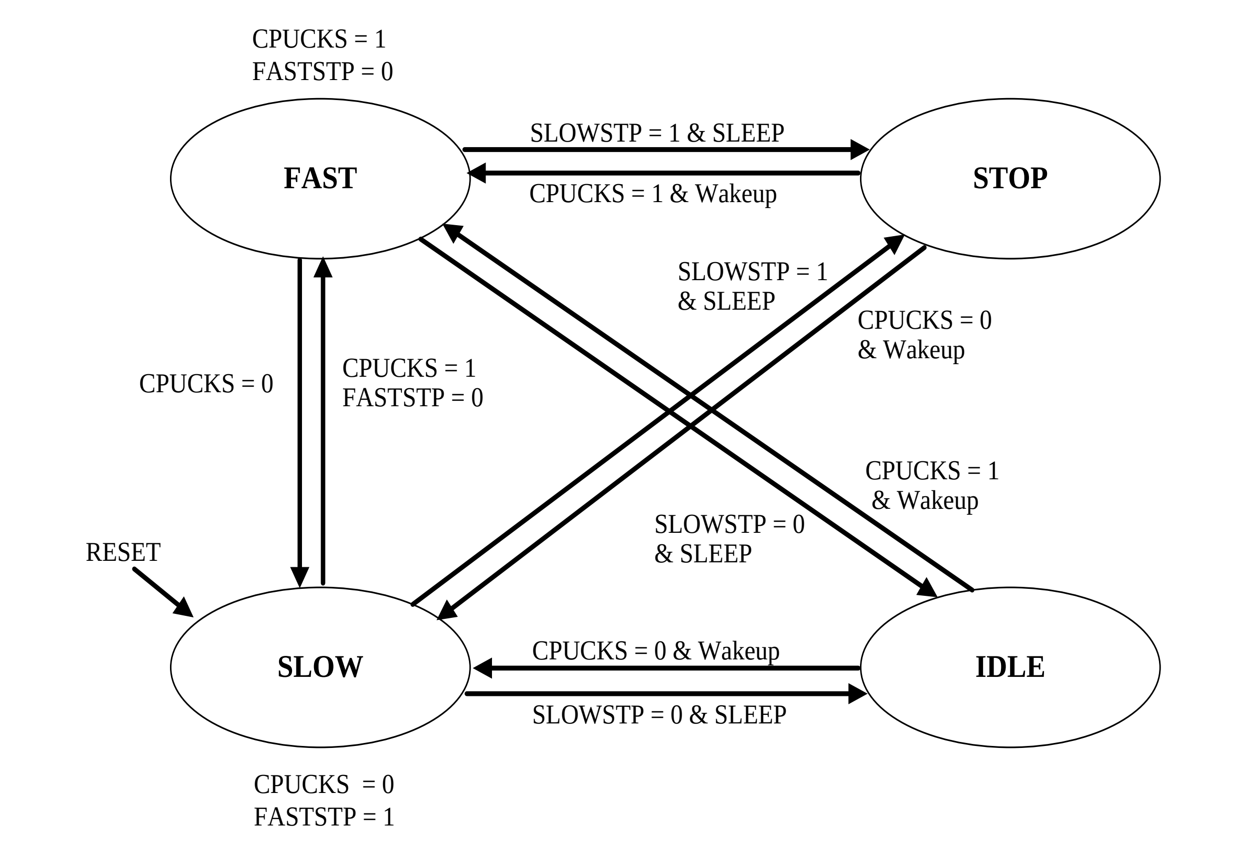
There are four operation modes for this device.

SLOW Mode: After power-on or reset, device enters SLOW mode. In this mode, the Fast-clock should be stopped (by FASTSTP=1, for power saving) and Slow-clock is enabled. The default Slow-clock is SIRC.

FAST Mode: In this mode, the program is executed using Fast-clock as CPU clock.

IDLE Mode: If Slow-clock is enabled (SLOWSTP=0) and T2CKS=0 before executing the SLEEP instruction, the CPU enters the IDLE mode. In this mode, the Slow-clock source keeps T2 block running. CPU stop fetching code and all blocks are stop except T2 related circuits. Idle mode is terminated by Reset or enabled Interrupts wake up.

STOP Mode: If Slow-clock is disabled before executing the SLEEP instruction, every block is turned off and the device enters the STOP mode after executing the SLEEP instruction. Stop Mode can be terminated by Reset or pin wake up.



Note:

- SLEEP denotes SLEEP instruction
- Wakeup denotes wake-up events, such as pin interrupt, pin wakeup or T2 Interrupt

CPU Operation Block Diagram

◇Example: Switch operating mode from SLOW mode to FAST mode

```
BCF     FASTSTP     ; Enable Fast-clock
BSF     CPUCKS      ; Switch system clock source to Fast-clock
```

◇Example: Switch operating mode from FAST mode to SLOW mode

```
BCF     SLOWSTP     ; Enable Slow-clock
BCF     CPUCKS      ; Switch system clock source to Slow-clock
BSF     FASTSTP     ; Stop Fast-clock
```

◇Example: Switch operating mode to IDLE mode

```
BCF     SLOWSTP     ; Enable Slow-clock
SLEEP               ; Enter IDLE mode
```

◇Example: Switch operating mode to STOP mode

```
BSF     SLOWSTP     ; Stop Slow-clock
SLEEP               ; Enter STOP mode
```

F0B	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CLKCTL	SCKTYPE	SXTGAIN		SLOWSTP	FASTSTP	CPUCKS	CPUPSC	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reset	0	1	1	0	1	0	1	1

F0B.7 **SCKTYPE:** Slow-clock type, this bit can be changed only in Fast mode (SELFCK=1).
0: SIRC
1: SXT, also set PA3 and PA4 as crystal oscillator pins.
Note: In SXT mode, user should set the PA3 and PA4 pins as Input with Pull-up (Mode 0).

F0B.6~5 **SXTGAIN:** 32768 SXT oscillator gain, 3=Highest gain, 0=Lowest gain. Higher gain can shorten the Crystal oscillation warm-up time. Lower gain can reduce oscillation current.

F0B.4 **SLOWSTP:** Slow-clock Stop control
0: Slow-clock run
1: Slow-clock stop

F0B.3 **FASTSTP:** Fast-clock Stop control, This bit can be changed only when CPUCKS=0
0: Fast-clock run
1: Fast-clock stop

F0B.2 **CPUCKS:** System clock (Fsys) selection, This bit can be changed only when FASTSTP=0
0: Slow-clock
1: Fast-clock

F0B.1~0 **CPUPSC:** System clock source prescaler.
00: divided by 8
01: divided by 4
10: divided by 2
11: divided by 1

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

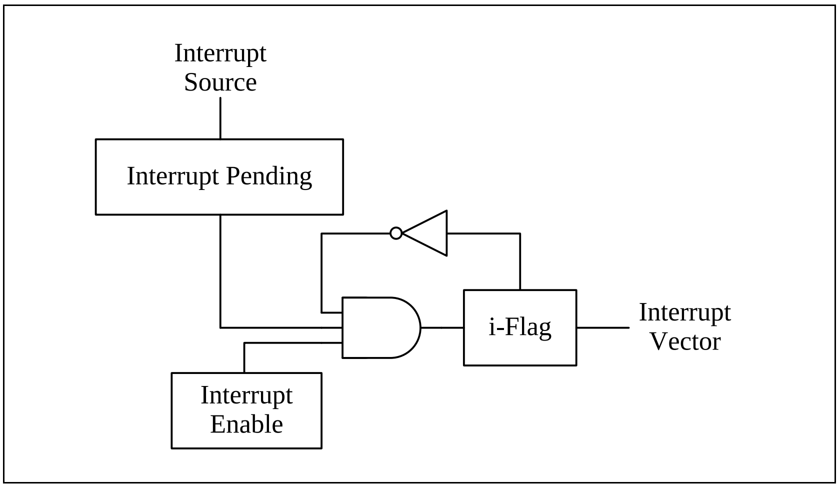
R03.7~0 **PWRDN:** Write this register (=SLEEP instruction) to enter IDLE or STOP Mode

7. Interrupt

This device has 1 level, 1 vector and 6 interrupt sources. Each interrupt source has its own enable control bit. An interrupt event will set its individual pending flag, no matter its enable control bit is 0 or 1.

If the corresponding interrupt enable bit has been set (INTIE), 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 001" instruction is inserted to CPU, and i-flag is set to prevent recursive interrupt nesting.

The i-flag is cleared in the instruction after the "RETI" instruction. That is, at least one instruction in main program is executed before service the pending interrupt. The interrupt event is level triggered. F/W must clear the interrupt event register while serving the interrupt routine.



F08	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIE	–	–	RFCIE	TM0IE	T2IE	INT2IE	INT1IE	INT0IE
R/W	–	–	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	–	0	0	0	0	0	0

F08.5 **RFCIE:** RFC interrupt enable
0: disable
1: enable

F08.4 **TM0IE:** Timer0 interrupt enable
0: disable
1: enable

F08.3 **T2IE:** Timer2 interrupt enable
0: disable
1: enable

F08.2 **INT2IE:** INT2 (PA7) interrupt enable
0: disable
1: enable

F08.1 **INT1IE:** INT1 (PB0) interrupt enable
0: disable
1: enable

F08.0 **INT0IE:** INT0 (PA0) interrupt enable
0: disable
1: enable

F09	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIF	–	–	RFCIF	TM0IF	T2IF	INT2IF	INT1IF	INT0IF
R/W	–	–	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	–	0	0	0	0	0	0

F09.5 **RFCIF:** RFC counter overflow interrupt event pending flag
Set by H/W while RFC counter overflow, clear by S/W writing 0xDF to INTIF

F09.4 **TM0IF:** Timer0 interrupt event pending flag
Set by H/W while Timer0 overflows, clear by S/W writing 0xEF to INTIF

F09.3 **T2IF:** Timer2 interrupt event pending flag
Set by H/W while Timer2 overflows, clear by S/W writing 0xF7 to INTIF

F09.2 **INT2IF:** INT2 (PA7) pin interrupt pending flag
Set by H/W at INT2 pin’s falling/rising edge, clear by S/W writing 0xFB to INTIF

F09.1 **INT1IF:** INT1 (PB0) pin interrupt pending flag
Set by H/W at INT1 pin’s falling/rising edge, clear by S/W writing 0xFD to INTIF

F09.0 **INT0IF:** INT0 (PA0) pin interrupt pending flag
Set by H/W at INT0 pin’s falling/rising edge, clear by S/W writing 0xFE to INTIF

F0C	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MF0C	T2CLR	T2CKS	TM0STP	–	–	INT2EDG	INT1EDG	INT0EDG
R/W	R/W	R/W	R/W	–	–	R/W	R/W	R/W
Reset	0	0	0	–	–	0	0	0

F0C.2 **INT2EDG:** INT2 pin (PA7) interrupt trigger edge select
0: falling edge to trigger
1: rising edge to trigger

F0C.1 **INT1EDG:** INT1 pin (PB0) interrupt trigger edge select
0: falling edge to trigger
1: rising edge to trigger

F0C.0 **INT0EDG:** INT0 pin (PA0) interrupt trigger edge select
0: falling edge to trigger
1: rising edge to trigger

8. I/O Port

I/O pins can be used as Schmitt-trigger input, CMOS push-pull output, or Open-drain output. The pull-up resistor is assignable to PA0~7 by S/W setting. To use the pin in Schmitt-trigger input mode, S/W needs to set the I/O pin to Mode0 or Mode1 and the corresponding port data PxD=1. Reading the pin data (PxD) has different meaning. In “Read-Modify-Write” instruction, CPU actually reads the output data register. In the others instructions, CPU reads the pin state. The so-called “Read-Modify-Write” instruction includes BSF, BCF and all instructions using F-Plane as destination.

The operations of four pin modes are listed as below. PA0~PA6 supports all 4 pin modes, PB0~PB3 supports Mode 1 (only Input without Pull-up, not include Open Drain function) and Mode2~3, PA7 only supports Mode 0~1.

Pin Mode	Pin function	PxD SFR data	Pin State	Resistor Pull-up	Digital Input
Mode 0	Open Drain Output Low	0	Drive Low	N	N
	Input with Pull-up	1	Pull-High	Y	Y
Mode 1	Open Drain Output Low	0	Drive Low	N	N
	Input without Pull-up	1	Hi-Z	N	Y
Mode 2	CMOS Push-Pull Output	0	Drive Low	N	N
		1	Drive High	N	N
Mode 3	Alternative function, such as LCD, PWM and RFC	1	–	N	N

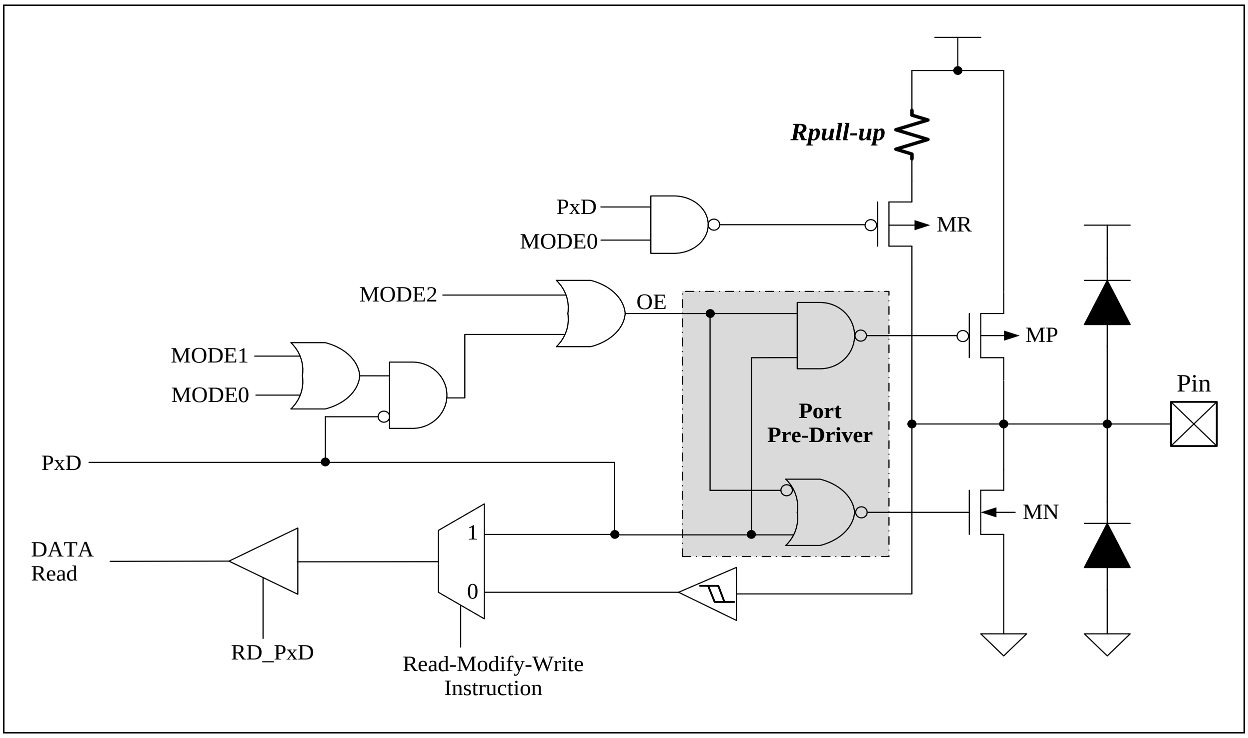
I/O Pin Function Table

Beside general purposed I/O port function, each pin may have one or more alternative functions.

Pin Name	Interrupt	LCD	RFC	Others	Mode3
PA0	INT0		RFC1R		RFC1R
PA1			RFC2R	PWMN	RFC2R
PA2			RFC0R		RFC0R
PA3				XOUT	
PA4				XIN	
PA5			RFCX		RFCX
PA6				PWMP	PWMP
PA7	INT2			nRST	
PB0	INT1	SEG11			SEG11
PB1		SEG10			SEG10
PB2		SEG9			SEG9
PB3		SEG8			SEG8

I/O Pin multi-function Table

Note: In SXT mode, user should set the PA3 and PA4 pins as Input with Pull-up (Mode 0).



I/O Port Structure

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

F05.7~0 **PAD:** PA7~PA0 data

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

R05.6 **PA7MOD:** PA7 pin mode
0: Mode0, open-drain I/O with internal pull-up
1: Mode1, open-drain I/O without internal pull-up

R05.5~4 **PA6MOD:** PA6 pin mode
00: Mode0, open-drain I/O with internal pull-up
01: Mode1, open-drain I/O without internal pull-up
10: Mode2, port data CMOS push-pull output
11: Mode3, PWMP CMOS push pull output

R05.3~2 **PA5MOD:** PA5 pin mode
00: Mode0, open-drain I/O with internal pull-up
01: Mode1, open-drain I/O without internal pull-up
10: Mode2, port data CMOS push-pull output
11: Mode3, RFCX input

R05.1~0 **PA4MOD:** PA4 pin mode
00: Mode0, open-drain I/O with internal pull-up
01: Mode1, open-drain I/O without internal pull-up
10: Mode2, port data CMOS push-pull output

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

R06.7~6 **PA3MOD:** PA3 pin mode
00: Mode0, open-drain I/O with internal pull-up
01: Mode1, open-drain I/O without internal pull-up
10: Mode2, port data CMOS push-pull output

R06.5~4 **PA2MOD:** PA2 pin mode
00: Mode0, open-drain I/O with internal pull-up
01: Mode1, open-drain I/O without internal pull-up
10: Mode2, port data CMOS push-pull output
11: Mode3, RFC0R output

R06.3~2 **PA1MOD:** PA1 pin mode
00: Mode0, open-drain I/O with internal pull-up
01: Mode1, open-drain I/O without internal pull-up
10: Mode2, port data CMOS push-pull output
11: Mode3, RFC2R output

R06.1~0 **PA0MOD:** PA0 pin mode
00: Mode0, open-drain I/O with internal pull-up
01: Mode1, open-drain I/O without internal pull-up
10: Mode2, port data CMOS push-pull output
11: Mode3, RFC1R output

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

F06.3~0 **PBD:** PB3~PB0 data

R07	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PBMODL	PB3MOD		PB2MOD		PB1MOD		PB0MOD	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	1	0	1	0	1	0	1

R07.7~6 **PB3MOD:** PB3 pin mode
0x: Mode1, input without internal pull-up
10: Mode2, port data CMOS push-pull output
11: Mode3, LCD SEG8 output

R07.5~4 **PB2MOD:** PB2 pin mode
0x: Mode1, input without internal pull-up
10: Mode2, port data CMOS push-pull output
11: Mode3, LCD SEG9 output

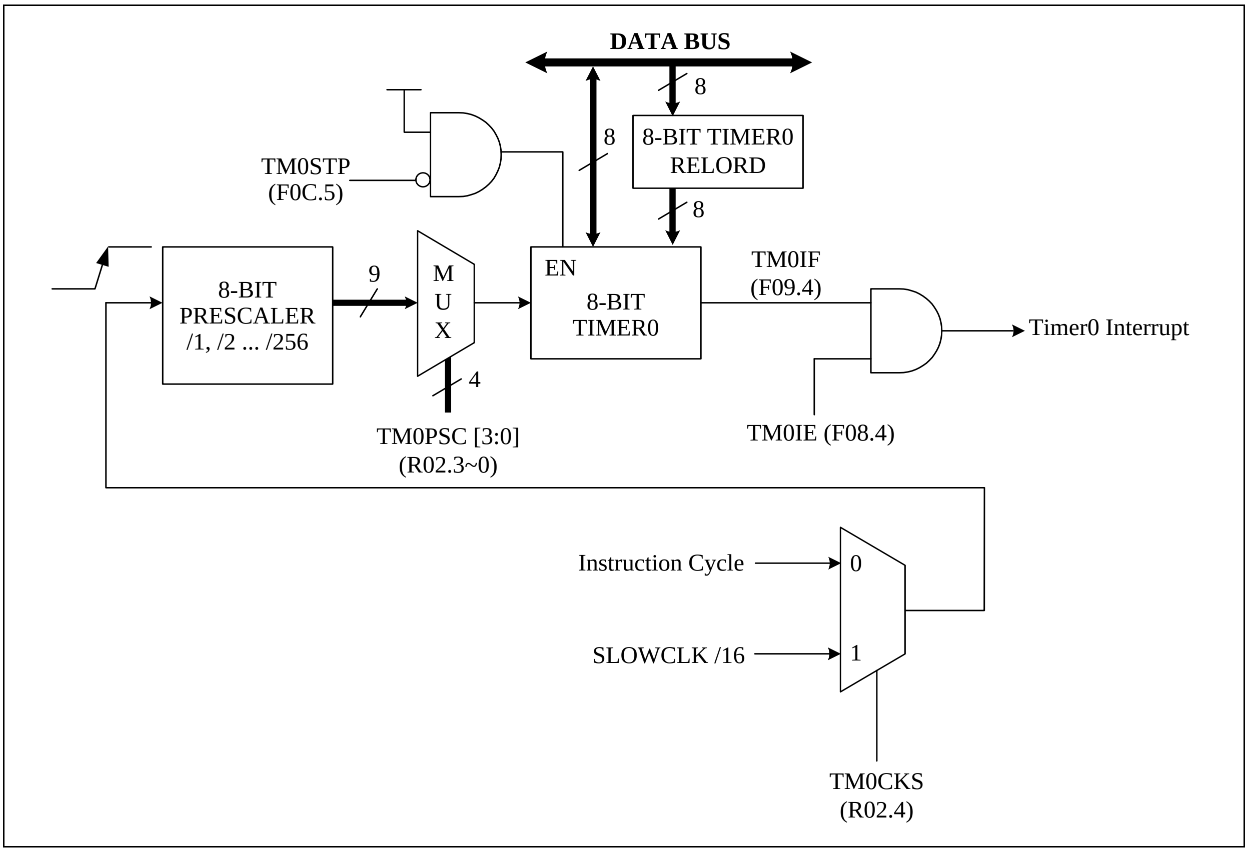
R07.3~2 **PB1MOD:** PB1 pin mode
0x: Mode1, input without internal pull-up
10: Mode2, port data CMOS push-pull output
11: Mode3, LCD SEG10 output

R07.1~0 **PB0MOD:** PB0 pin mode
0x: Mode1, input without internal pull-up
10: Mode2, port data CMOS push-pull output
11: Mode3, LCD SEG11 output

9. Timers

9.1 Timer0

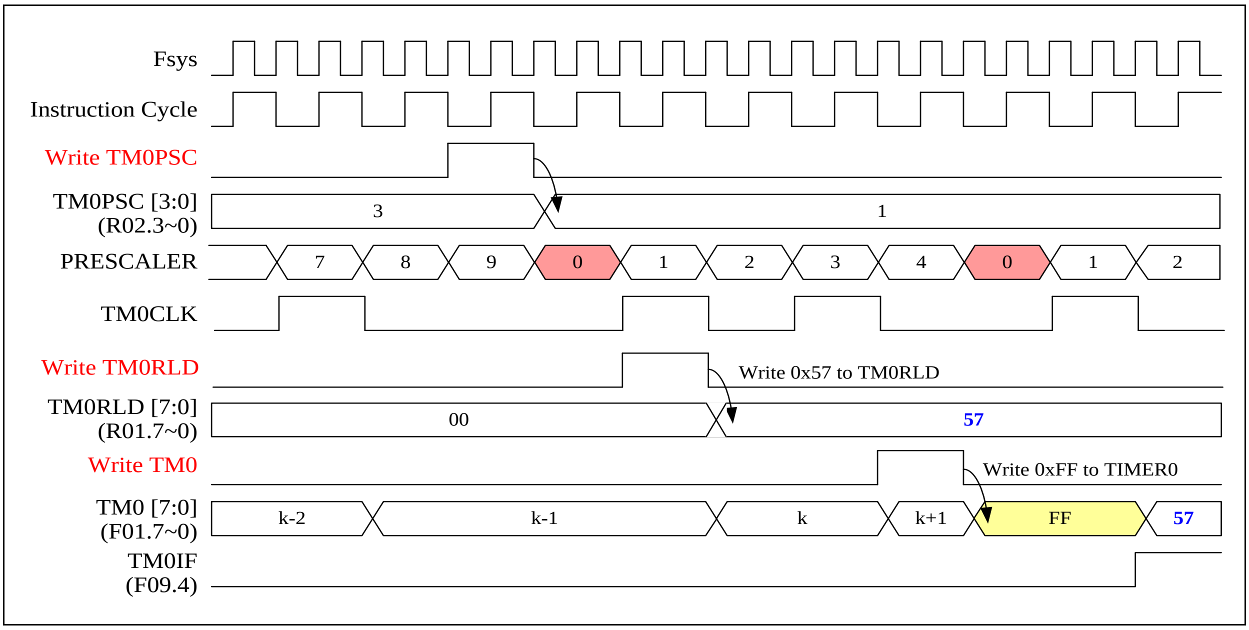
The Timer0 is an 8-bit wide register of F-Plane 01h (TM0). It can be read or written the same way as any other register of F-Plane. Besides, Timer0 increases itself according to the pre-scaled clock source, which comes from the instruction cycle or Slow-clock divided by 16. The Timer0 increase rate is determined by “Timer0 Pre-Scale” (TM0PSC). The Timer0 sets TM0IF flag and reloads itself with TM0RLD when Timer0’s count overflows. It generates Timer0 Interrupt if (TM0IE) is set. Timer0 can be stopped counting if the TM0STP bit is set.



Timer0 Block Diagram

The following timing diagram describes the Timer0 works in pure 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, TM0IF (Timer0 Interrupt Flag) will be set to 1 and generate interrupt if TM0IE (Timer0 Interrupt Enable) is set.



Timer0 works in Timer mode (TM0CKS=0)

The equation of TM0 interrupt time value is as following:

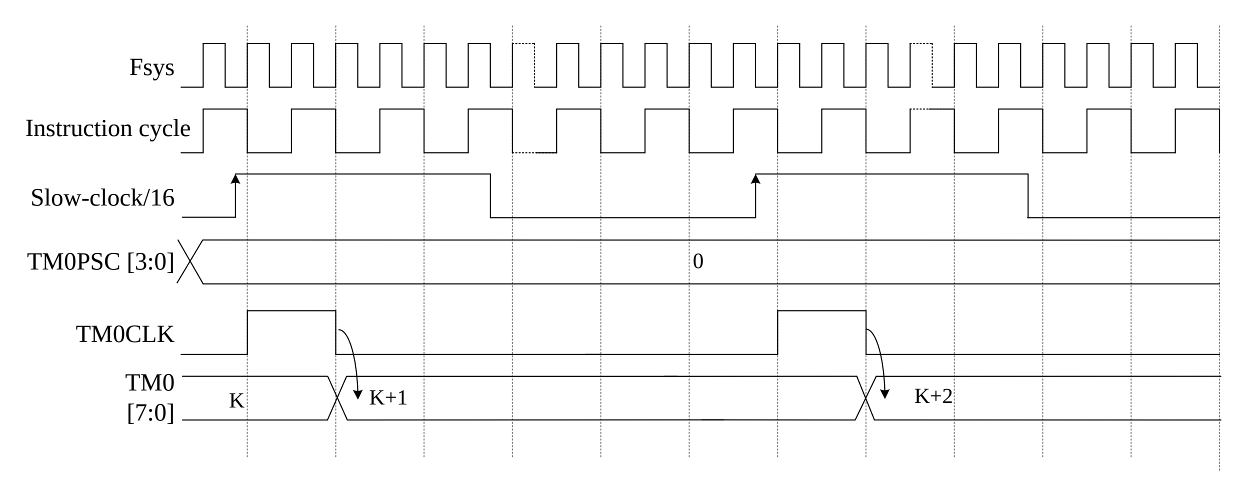
TM0 interrupt interval cycle time=Instruction cycle time/TM0PSC/ (256-TM0)

◇ Example: Setup TM0 work in Timer mode

```
; Setup TM0 clock source and divider
        MOVLW     00000101B     ; R02.4 = 0, Setup TM0 clock=Instruction cycle
        MOVWR     R02           ; R02.3~0=5 (TM0PSC)
                                ; TM0 clock prescaler=Instruction cycle divided by 32
; Set TM0 timer.
        BSF       TM0STP        ; Disable TM0 counting (Default "0").
        MOVLW     156
        MOVWF     TM0           ; Write 156 into TM0 register of F-Plane. (F01)
        MOVLW     124
        MOVWR     TM0RLD        ; Write 124 into TM0RLD register of R-Plane. (R01)

; Enable TM0 timer and interrupt function.
        MOVLW     11101111B     ; Clear TM0 request interrupt flag by byte operation
        MOVWF     INTIF         ; F-Plane 09H
        MOVLW     00010000B     ; Enable TM0 interrupt function
        MOVWR     INTIE         ; F-Plane 08H
        BCF       TM0STP        ; Enable TM0 counting (Default “0”).
```

If TM0CKS=1, Timer0's clock source is switched to "Slow-clock divided by 16", which is synchronized by instruction cycle. That means the Fsys must be faster than Slow-clock/4 for proper operation.



TM0CKS=1, Timer0 clock source is Slow-clock/16

F01	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TM0	TM0							
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

F01 **TM0:** Timer0 data

F08	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIE	–	–	RFCIE	TM0IE	T2IE	INT2IE	INT1IE	INT0IE
R/W	–	–	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	–	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	–	–	RFCIF	TM0IF	T2IF	INT2IF	INT1IF	INT0IF
R/W	–	–	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	–	0	0	0	0	0	0

F09.4 **TM0IF:** Timer0 interrupt event pending flag
Set by H/W while Timer0 overflows, clear by S/W writing 0xEF to INTIF

F0C	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MF0C	T2CLR	T2CKS	TM0STP	–	–	INT2EDG	INT1EDG	INT0EDG
R/W	R/W	R/W	R/W	–	–	R/W	R/W	R/W
Reset	0	0	0	–	–	0	0	0

F0C.5 **TM0STP:** Timer0 counter stop
0: Timer0 is counting
1: Timer0 stop counting

R01	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TM0RLD	TM0RLD							
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

R01 **TM0RLD:** Timer0 Overflow Reload Data

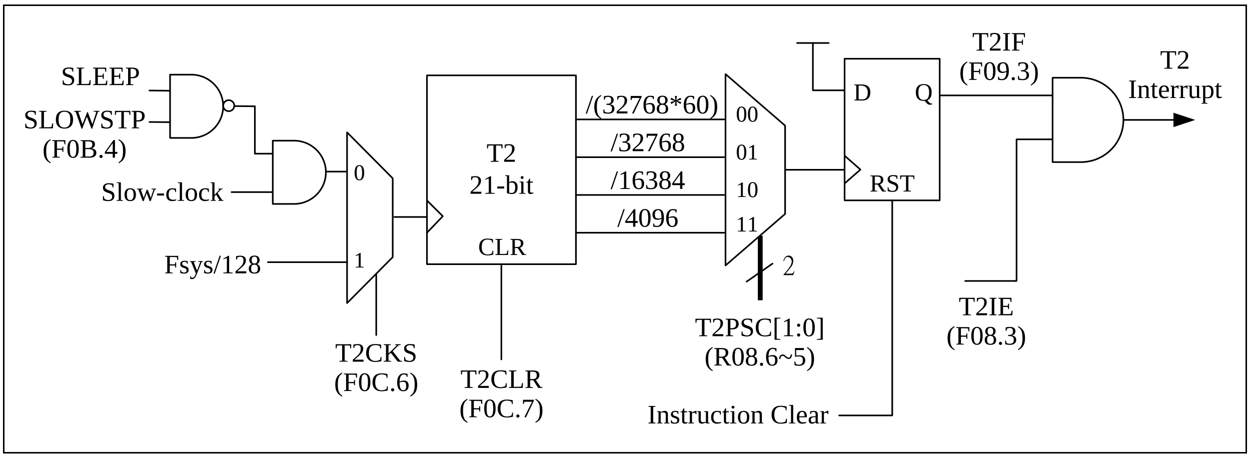
R02	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TM0CTL	–	–	–	TM0CKS	TM0PSC			
R/W	–	–	–	R/W	R/W	R/W	R/W	R/W
Reset	–	–	–	0	0	0	0	0

R02.4 **TM0CKS:** Timer0 clock source
0: Instruction cycle
1: Slow-clock /16

R02.3~0 **TM0PSC:** Timer0 prescaler, clock source divided by
0000: /1
0001: /2
0010: /4
0011: /8
0100: /16
0101: /32
0110: /64
0111: /128
1xxx: /256

9.2 Timer2

Timer2 (T2) is a 21-bit timer and the clock sources are from either Fsys/128 or Slow-clock. The clock source is used to generate time base interrupt and T2 module clock. It is selected by T2CKS (F0C.6). The T2's 21-bit content cannot be read by instructions. It generates interrupt flag T2IF (F09.3) with the clock divided by 32768*60, 32768, 16384, or 4096 depends on the T2PSC[1:0] (R08.6~5) bits. The following figure shows the block diagram of T2.



T2 Block Diagram

◇ Example: CPU is running at FAST mode, Fsys=Fast-clock=FIRC, Slow-clock source is SXT

```
; Setup T2 clock source and divider
            BCF         T2CKS           ; T2CKS=0, T2 clock source is Slow-clock
            MOVLW       00100000B
            MOVWR       R08             ; T2PSC=01b, divided by 32768
            BSF         T2CLR           ; T2CLR=1, clear T2 counter

; Enable T2 interrupt function
            MOVLW       11110111B
            MOVWF       INTIF           ; Clear T2 request interrupt flag
            BSF         T2IE            ; Enable T2 interrupt function
```

T2 clock source is Slow-clock = 32 KHz, T2 divided by 32768
T2 interrupt period =1 second

F08	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIE	–	–	RFCIE	TM0IE	T2IE	INT2IE	INT1IE	INT0IE
R/W	–	–	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	–	0	0	0	0	0	0

F08.3 **T2IE:** Timer2 interrupt enable
0: disable
1: enable

F09	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIF	–	–	RFCIF	TM0IF	T2IF	INT2IF	INT1IF	INT0IF
R/W	–	–	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	–	0	0	0	0	0	0

F09.3 **T2IF:** Timer2 interrupt event pending flag
Set by H/W while Timer2 overflows, clear by S/W writing 0xF7 to INTIF

F0C	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MF0C	T2CLR	T2CKS	TM0STP	–	–	INT2EDG	INT1EDG	INT0EDG
R/W	R/W	R/W	R/W	–	–	R/W	R/W	R/W
Reset	0	0	0	–	–	0	0	0

F0C.7 **T2CLR**: T2 counter clear
0: T2 is counting
1: T2 is cleared and TM2TGL=1, this bit is auto cleared by H/W

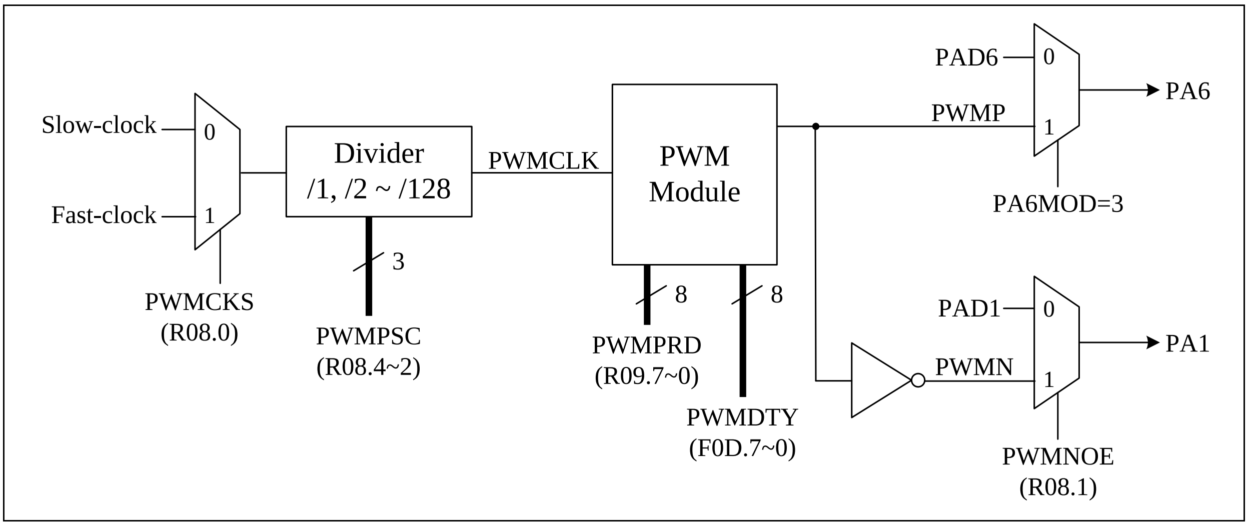
F0C.6 **T2CKS:** T2 clock source selection
0: Slow-clock
1: Fsys/128

R08	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMCTL	PWMCKS	T2PSC		PWMPSC			PWMNOE	WDTPSC
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	0	0	0	0	0	0	0

R08.6~5 **T2PSC:** T2 prescaler. T2 interrupt is T2 clock divided by
00: /(32768*60)
01: /32768
10: /16384
11: /4096

10. PWM

The PWM can select Fast-clock or Slow-clock as its clock source, with divided by 1~128 prescaler. The PWM period is adjustable by PWMPRD and its 256 step duty cycle is controlled by PWMDTY. The PWMP and PWMN are positive and negative CMOS output pairs to pins.



PWM Block Diagram

◇Example: Slow-clock = SXT 32768Hz

```
; Setup PWMP a 512Hz, 50% duty cycle output
            MOVLW     00011100B          ; PWMCKS=0, PWMPSC=111
            MOVWR     PWMCTL             ; PWMCLK=Slow-clock/1=32768Hz

            MOVLW     63
            MOVWR     PWMPRD             ; Set PWM period = 63 + 1 = 64

            MOVLW     32
            MOVWF     PWMDTY             ; Set PWM duty = 32

            MOVLW     00110000B          ; PA6MOD=3
            MOVWR     PAMODH             ; PWMP output to PA6 pin
```

PWM clock frequency = Slow-clock / PWMPSC = 32768Hz / 1 = 32768Hz
PWM output frequency = PWMCLK / (PWMPRD + 1) = 32768Hz / (63 + 1) = 512 Hz
PWM duty cycle = PWMDTY / (PWMPRD + 1) = 32 / (63 + 1) = 50%

R08	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMCTL	PWMCKS	T2PSC		PWMPSC			PWMNOE	WDTPSC
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	0	0	0	0	0	0	0

R08.7 **PWMCKS:** PWM clock source select
0: Slow-clock
1: Fast-clock

R08.4~2 **PWMPSC**: PWM clock prescaler
000: PWM clock is Slow/Fast clock divided by 128
001: PWM clock is Slow/Fast clock divided by 64
010: PWM clock is Slow/Fast clock divided by 32
011: PWM clock is Slow/Fast clock divided by 16
100: PWM clock is Slow/Fast clock divided by 8
101: PWM clock is Slow/Fast clock divided by 4
110: PWM clock is Slow/Fast clock divided by 2
111: PWM clock is Slow/Fast clock divided by 1

R08.1 **PWMNOE:** PWMN output to PA1 pin
0: disable
1: enable

F0D	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMDTY	PWMDTY							
R/W	R/W							
Reset	1	0	0	0	0	0	0	0

F0D.7~0 **PWMDTY**: PWM duty, 0=0 PWMCLK, 80h=128 PWMCLK, FFh=255 PWMCLK

R09	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMPRD	PWMPRD							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

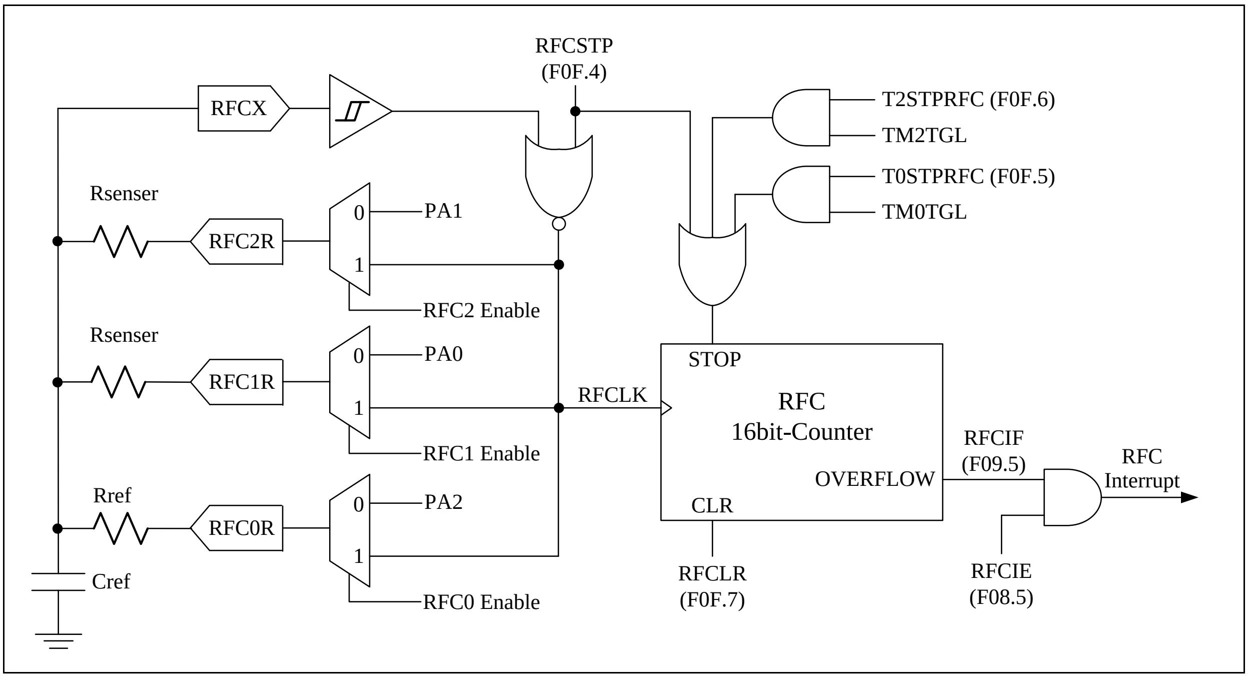
R09.7~0 **PWMPRD**: PWM period, FFh=256 PWMCLK, 7Fh=128 PWMCLK

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

R05.5~4 **PA6MOD:** PA6 pin mode
00: Mode0, open-drain I/O with internal pull-up
01: Mode1, open-drain I/O without internal pull-up
10: Mode2, port data CMOS push-pull output
11: Mode3, PWMP CMOS push pull output

11. Resistance to Frequency Converter (RFC)

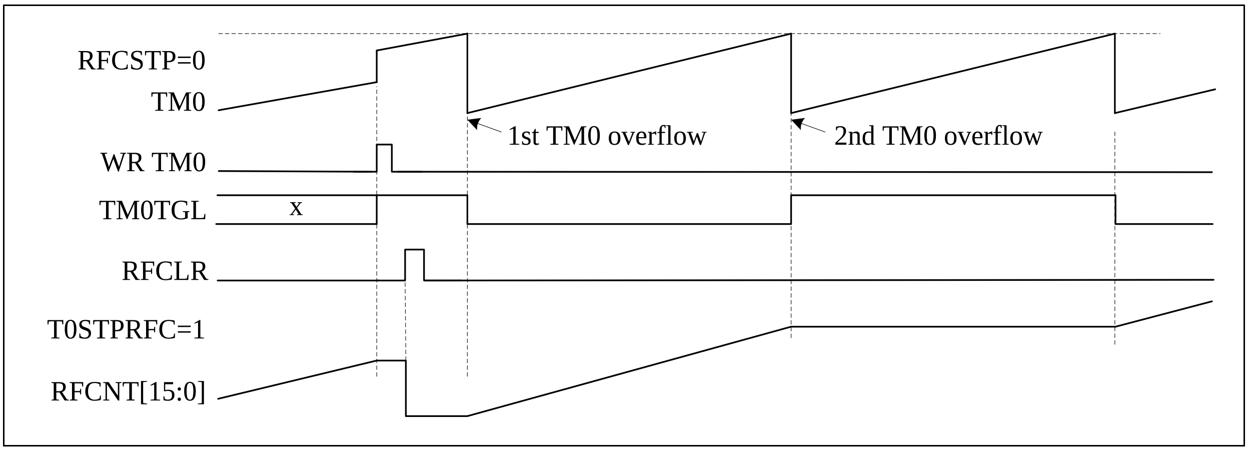
The RFC module contains RC oscillator and RFC counter. The RFC-clock comes from the oscillation circuitry built by RFCX pin and RFC0R, RFC1R or RFC2R pins.



RFC Block Diagram

The 16-bit RFC counter can stop by Timer0 or Timer2's overflow control. This function helps the RFC counter to count the RFC clock with more accuracy by H/W automatically start and stop. The steps of this usage are described below.

1. RFCSTP=0, T0STPRFC=1, T2STPRFC=0
2. Write Timer0 to setup a suitable overflow. Meanwhile, HW auto set TM0TGL=1 and RFCNT stops.
3. Clear RFCNT[15:0] by RFCLR=1
4. After the 1st Timer0 overflow, TM0TGL=0 and RFCNT start counting
5. Wait for the 2nd Timer0 overflow, TM0TGL=1 and RFCNT stops again, then read RFCNT data.



TM0 control RFC

F0F	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RFCTL	RFCLR	T2STPRFC	T0STPRFC	RFCSTP	–	–	RFCHS	
R/W	R/W	R/W	R/W	R/W	–	–	R/W	R/W
Reset	1	0	0	1	–	–	0	0

F0F.7 **RFCLR:** clear RFC counter
0: RFC counter run
1: RFC counter clear

F0F.6 **T2STPRFC:** Timer2 overflow toggle signal (TM2TGL) to stop RFC counter
0: disable
1: enable

F0F.5 **T0STPRFC:** Timer0 overflow toggle signal (TM0TGL) to stop RFC counter
0: disable
1: enable

F0F.4 **RFCSTP:** S/W stop RFC counter and oscillator
0: RFC counter and oscillator run
1: RFC counter and oscillator stop

F0F.1~0 **RFCHS:** select RFC oscillator channel
00: RFC0R (PA2)
01: RFC1R (PA0)
10: RFC2R (PA1)

F11	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RFCNTH	RFCNTH							
R/W	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

F11.7~0 **RFCNTH:** RFC counter high byte, RFCNT[15:8]

F12	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RFCNTL	RFCNTL							
R/W	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

F12.7~0 **RFCNTL:** RFC counter low byte, RFCNT[7:0]

F08	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIE	–	–	RFCIE	TM0IE	T2IE	INT2IE	INT1IE	INT0IE
R/W	–	–	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	–	0	0	0	0	0	0

F08.5 **RFCIE:** RFC interrupt enable
0: disable
1: enable

F09	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTIF	–	–	RFCIF	TM0IF	T2IF	INT2IF	INT1IF	INT0IF
R/W	–	–	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	–	0	0	0	0	0	0

F09.5 **RFCIF:** RFC counter overflow interrupt event pending flag
Set by H/W while RFC counter overflow, clear by S/W writing 0xDF to INTIF

12. LCD Driver

The 1/2 Bias LCD Driver is capable of driving LCD panel with 4 COM x 11 SEG or 3 COM x 12 SEG. The typical LCD bias voltage is VL1=1.5V and V_{LCD}=3V, generated by chip internal pump circuit and CUP1, CUP2 and CX pins. For M5610, V_{LCD}=V_{BAT}. For M5615, V_{LCD}=2*V_{BAT}. The LCD clock source is Slow-clock. There are 4 LCD frame rate can be selected by SFR.

F10	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LCDCTL	LCDON	LCDFRM		LCDUTY	–	–	–	–
R/W	R/W	R/W	R/W	R/W	–	–	–	–
Reset	0	1	0	1	–	–	–	–

F10.7 **LCDON:** LCD driver enable
0: disable
1: enable

F10.6~5 **LCDFRM:** LCD frame rate, calculated by Slow-clock=32768Hz
00: 64Hz for 1/4 duty, 85Hz for 1/3 duty
01: 32Hz for 1/4 duty, 43Hz for 1/3 duty
10: 16Hz for 1/4 duty, 21Hz for 1/3 duty

F10.4 **LCDUTY:** LCD duty
0: 1/3 duty
1: 1/4 duty

R07	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PBMODL	PB3MOD		PB2MOD		PB1MOD		PB0MOD	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	1	0	1	0	1	0	1

R07.7~6 **PB3MOD:** PB3 pin mode
0x: Mode1, input without internal pull-up
10: Mode2, port data CMOS push-pull output
11: Mode3, LCD SEG8 output

R07.5~4 **PB2MOD:** PB2 pin mode
0x: Mode1, input without internal pull-up
10: Mode2, port data CMOS push-pull output
11: Mode3, LCD SEG9 output

R07.3~2 **PB1MOD:** PB1 pin mode
0x: Mode1, input without internal pull-up
10: Mode2, port data CMOS push-pull output
11: Mode3, LCD SEG10 output

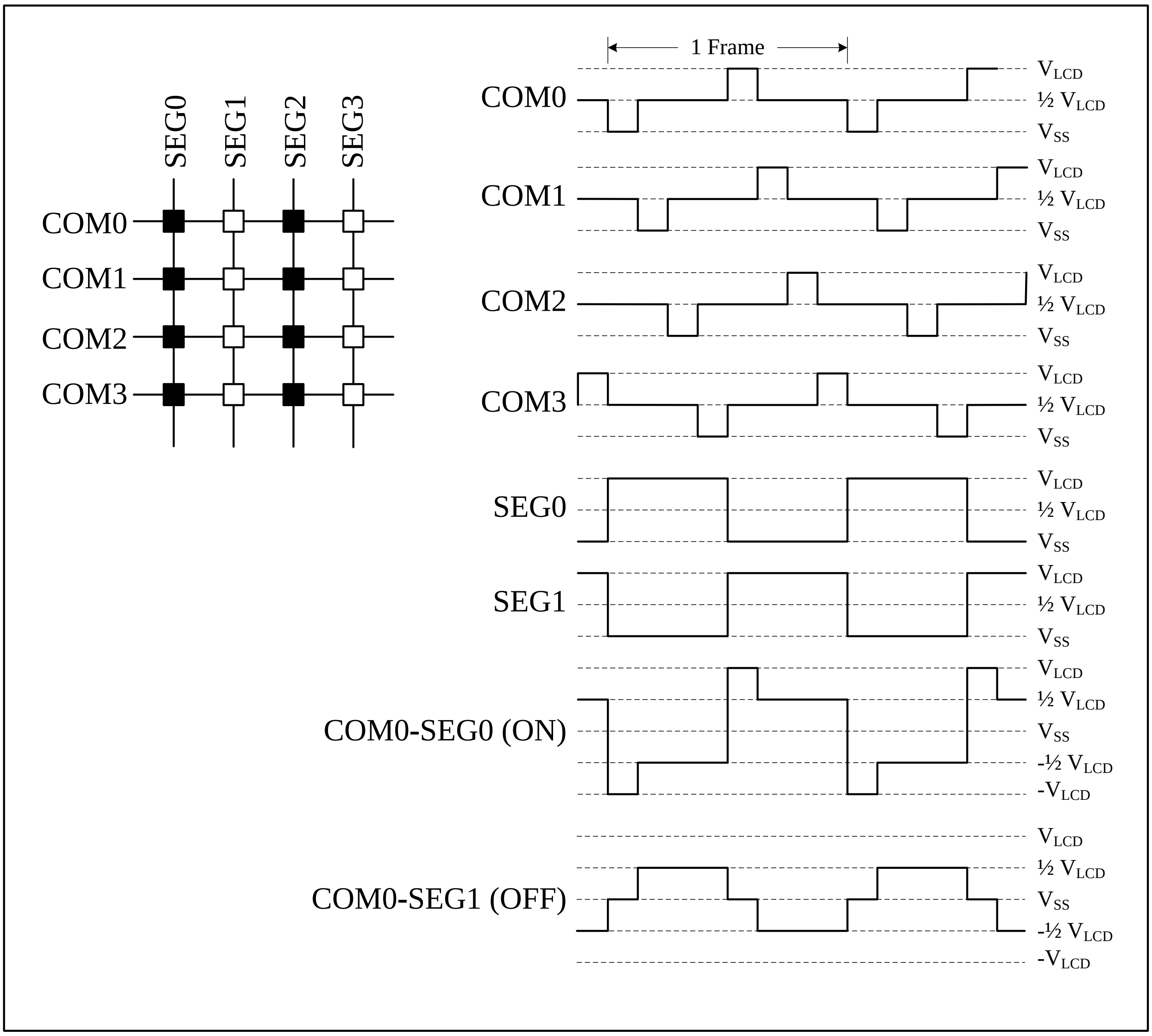
R07.1~0 **PB0MOD:** PB0 pin mode
0x: Mode1, input without internal pull-up
10: Mode2, port data CMOS push-pull output
11: Mode3, LCD SEG11 output

F0E	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LBDCTL	CMPO	CMPVS			–	PWRSAV	PUMPCKS	–
R/W	R	R/W	R/W	R/W	–	R/W	R/W	–
Reset	–	0	0	0	–	0	0	–

F0E.1 **PUMPCKS:** LCD pump clock select
0: Slow-clock / 4
1: Slow-clock / 8

	COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0
R-Plane	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
R20	SEG1	SEG1	SEG1	SEG1	SEG0	SEG0	SEG0	SEG0
R21	SEG3	SEG3	SEG3	SEG3	SEG2	SEG2	SEG2	SEG2
R22	SEG5	SEG5	SEG5	SEG5	SEG4	SEG4	SEG4	SEG4
R23	SEG7	SEG7	SEG7	SEG7	SEG6	SEG6	SEG6	SEG6
R24	SEG9	SEG9	SEG9	SEG9	SEG8	SEG8	SEG8	SEG8
R25	SEG11	SEG11	SEG11	SEG11	SEG10	SEG10	SEG10	SEG10

LCDRAM mapping



1/4 Duty, 1/2 Bias LCD driver

MEMORY MAP

F-Plane

Name	Address	R/W	Rst	Description
(F00) INDF				**Function related to: F-Plane R/W**
INDF	00.7~0	R/W	–	Not a physical register, addressing INDF actually point to the F-Plane register whose address is contained in the FSR register
(F01) TM0				**Function related to: Timer0**
TM0	01.7~0	R/W	0	Timer0 data
(F02) PCL				**Function related to: Program Counter**
PCL	02.7~0	R/W	0	Low-byte of Program Counter (PC[7~0])
(F03) STATUS				**Function related to: STATUS**
GB2	03.7	R/W	0	General purpose bit
GB1	03.6	R/W	0	General purpose bit
GB0	03.5	R/W	0	General purpose bit
TO	03.4	R	0	WDT timeout flag, set by WDT timeout; cleared by POR, LVR, ‘SLEEP’ or ‘CLRWDT’ instruction
PD	03.3	R	0	Power down flag, set by ‘SLEEP’ instruction; cleared by POR, LVR or ‘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: F-Plane R/W**
GB3	04.7	R/W	0	General purpose bit
FSR	04.6~0	R/W	0	F-Plane File Select Register, indirect address mode pointer
(F05) PAD				**Function related to: Port A**
PAD	05.7~0	R	FF	Port A pin or “data register” state
		W		Port A output data register
(F06) PBD				**Function related to: Port B**
PBD	06.3~0	R	F	Port B pin or “data register” state
		W		Port B output data register
(F08) INTIE				**Function related to: Interrupt Enable**
RFCIE	08.5	R/W	0	RFC interrupt enable 0: disable 1: enable
TM0IE	08.4	R/W	0	Timer0 interrupt enable 0: disable 1: enable
T2IE	08.3	R/W	0	Timer2 interrupt enable 0: disable 1: enable
INT2IE	08.2	R/W	0	INT2 pin (PA7) interrupt enable 0: disable 1: enable
INT1IE	08.1	R/W	0	INT1 pin (PB0) interrupt enable 1: enable 0: disable
INT0IE	08.0	R/W	0	INT0 pin (PA0) interrupt enable 0: disable 1: enable

Name	Address	R/W	Rst	Description
(F09) INTIF				**Function related to: Interrupt Flag**
RFCIF	09.5	R	0	RFC counter overflow interrupt event pending flag, Set by H/W while RFC counter overflows
		W		writing 0xDF to INTIF to clear this flag
TM0IF	09.4	R	0	Timer0 interrupt event pending flag, set by H/W while Timer0 overflows
		W		writing 0xEF to INTIF to clear this flag
T2IF	09.3	R	0	Timer2 interrupt event pending flag, set by H/W while WKT time out
		W		writing 0xF7 to INTIF to clear this flag
INT2IF	09.2	R	0	INT2 (PA7) interrupt event pending flag, set by H/W at INT2 pin's rising/falling edge
		W		writing 0xFB to INTIF to clear this flag
INT1IF	09.1	R	0	INT1 (PB0) interrupt event pending flag, set by H/W at INT1 pin's rising/falling edge
		W		writing 0xFD to INTIF to clear this flag
INT0IF	09.0	R	0	INT0 (PA0) interrupt event pending flag, set by H/W at INT0 pin's rising/falling edge
		W		writing 0xFE to INTIF to clear this flag
(F0A) PCH				**Function related to: PROGRAM COUNT**
PCH	0A.1~0	R	0	2 MSBs of Program Counter (PC[9:8])
(F0B) CLKCTL				**Function related to: system clock (Fsys)**
SCKTYPE	0B.7	R/W	0	Slow-clock Type 0: SIRC 1: SXT
SXTGAIN	0B.6~5	R/W	11	32768 SXT oscillator gain 0: lowest gain ... 3: highest gain
SLOWSTP	0B.4	R/W	0	Slow-clock Stop control 0: Slow-clock run 1: Slow-clock stop
FASTSTP	0B.3	R/W	1	Fast-clock Stop control 0: Fast-clock run 1: Fast-clock stop
CPUCKS	0B.2	R/W	0	System clock (Fsys) source selection 0: Slow-clock 1: Fast-clock
CPUPSC	0B.1~0	R/W	11	System clock source prescaler. Clock source is divided by 00: /8 01: /4 10: /2 11: /1
(F0C) MF0C				**Function related to: TM0/T2/Interrupt**
T2CLR	0C.7	R/W	0	T2 counter clear 0: T2 is counting 1: T2 is cleared and TM2TGL=1, this bit is auto cleared by H/W
T2CKS	0C.6	R/W	0	T2 clock source selection 0: Slow-clock 1: Fsys/128
TM0STP	0C.5	R/W	0	Timer0 counter stop 0: Timer0 running 1: Timer0 stop
INT2EDG	0C.2	R/W	0	INT2 pin (PA7) interrupt trigger edge select 0: falling edge 1: rising edge
INT1EDG	0C.1	R/W	0	INT1 pin (PB0) interrupt trigger edge select 0: falling edge 1: rising edge
INT0EDG	0C.0	R/W	0	INT0 pin (PA0) interrupt trigger edge select 0: falling edge 1: rising edge

Name	Address	R/W	Rst	Description
(F0D) PWMDTY				**Function related to: PWM**
PWMDTY	0D.7~0	R/W	80	PWM duty, 0=0 PWMCLK, 80h=128 PWMCLK, FFh=255 PWMCLK
(F0E) LBDCTL				**Function related to: LBD/LCD**
CMPO	0E.7	R	–	Compare result of BandGap voltage and V_{BAT}/V_{LCD} voltage divider.
CMPVS	0E.6~4	R/W	0	000: Comparator and Bandgap Disable 001: detect if M5610's V_{BAT}>2.4V; detect if M5615's V_{BAT}>1.20V 010: detect if M5610's V_{BAT}>2.5V; detect if M5615's V_{BAT}>1.25V 011: detect if M5610's V_{BAT}>2.6V; detect if M5615's V_{BAT}>1.30V 100: detect if M5610's V_{BAT}>2.7V; detect if M5615's V_{BAT}>1.35V 101: detect if M5610's V_{BAT}>2.8V; detect if M5615's V_{BAT}>1.40V 110: detect if M5610's V_{BAT}>2.9V; detect if M5615's V_{BAT}>1.45V 111: detect if M5610's V_{BAT}>3.0V; detect if M5615's V_{BAT}>1.50V
PWRSAV	0E.2	R/W	0	Power saving control for M5610. 0: Disable, V_{DD}=V_{BAT} 1: Enable, V_{DD}=V_{BAT}/2
PUMPCKS	0E.1	R/W	0	LCD pump clock select 0: Slow-clock / 4 1: Slow-clock / 8
(F0F) RFCTL				**Function related to: RFC**
RFCLR	0F.7	R/W	1	Clear RFC counter 0: RFC run 1: RFC clear
T2STPRFC	0F.6	R/W	0	Timer2 overflow toggle signal (TM2TGL) to stop RFC counter 0: disable 1: enable
T0STPRFC	0F.5	R/W	0	Timer0 overflow toggle signal (TM0TGL) to stop RFC counter 0: disable 1: enable
RFCSTP	0F.4	R/W	1	S/W stop RFC counter and oscillator 0: RFC run 1: RFC stop
RFCHS	0F.1~0	R/W	0	Select RFC oscillator channel 00: RFC0R (PA2) 01: RFC1R (PA0) 10: RFC2R (PA1)
(F10) LCDCTL				**Function related to: LCD**
LCDON	10.7	R/W	0	LCD driver enable 0: disable 1: enable
LCDFRM	10.6~5	R/W	10	LCD frame rate, calculated by Slow-clock=32768Hz 00: 64Hz for 1/4 duty, 85Hz for 1/3 duty 01: 32Hz for 1/4 duty, 43Hz for 1/3 duty 10: 16Hz for 1/4 duty, 21Hz for 1/3 duty
LCDUTY	10.4	R/W	1	LCD duty 0: 1/3 duty 1: 1/4 duty
(F11) RFCNTH				**Function related to: RFC**
RFCNTH	11.7~0	R	0	RFC counter high byte, RFCNT[15:8]
(F12) RFCNTL				**Function related to: RFC**
RFCNTL	12.7~0	R	0	RFC counter low byte, RFCNT[7:0]
(F1C) RSR				**Function related to: R-Plane R/W**
RSR	1C.7~0	R/W	0	R-Plane file select register, indirect address mode pointer
(F1D) DPL				**Function related to: Table Read**
DPL	1D.7~0	R/W	0	Table read low address, data ROM pointer (DPTR[7:0])
(F1E) DPH				**Function related to: Table Read**
DPH	1E.1~0	R/W	0	Table read high address, data ROM pointer (DPTR[9:8])
User Data RAM				
FRAM	20~7F	R/W	–	F-Plane RAM area (96 Bytes)

R-Plane

Name	Address	R/W	Rst	Description
(R00) INDR				**Function related to: R-Plane R/W**
INDR	00.7~0	R/W	–	Not a physical register, addressing INDR actually point to the R-Plane register whose address is contained in the RSR register
(R01) TM0RLD				**Function related to: TM0**
TM0RLD	01.7~0	R/W	0	Timer0 reload Data
(R02) TM0CTL				**Function related to: TM0**
TM0CKS	02.4	R/W	0	Timer0 clock source 0: Instruction cycle 1: Slow-clock /16
TM0PSC	02.3~0	R/W	0	Timer0 clock source prescaler. Clock source is divided by 0000: /1 0101: /32 0001: /2 0110: /64 0010: /4 0111: /128 0011: /8 1xxx: /256 0100: /16
(R03) PWRDN				**Function related to: Power Down**
PWRDN	03	W	–	Write this register (=SLEEP instruction) to enter IDLE or STOP Mode
(R04) WDTCLR				**Function related to: WDT**
WDTCLR	04	W	–	Write this register to clear WDT (=CLRWDT instruction)
(R05) PAMODH				**Function related to: Port A**
PA7MOD	05.6	R/W	0	0: Mode0, PA7 is open-drain I/O with internal pull-up 1: Mode1, PA7 is open-drain I/O without internal pull-up
PA6MOD	05.5~4	R/W	01	00: Mode0, PA6 is open-drain I/O with internal pull-up 01: Mode1, PA6 is open-drain I/O without internal pull-up 10: Mode2, PA6 is CMOS push-pull output 11: Mode3, PA6 is PWMP CMOS push pull output
PA5MOD	05.3~2	R/W	01	00: Mode0, PA5 is open-drain I/O with internal pull-up 01: Mode1, PA5 is open-drain I/O without internal pull-up 10: Mode2, PA5 is CMOS push-pull output 11: Mode3, PA5 is RFCX input
PA4MOD	05.1~0	R/W	01	00: Mode0, PA4 is open-drain I/O with internal pull-up 01: Mode1, PA4 is open-drain I/O without internal pull-up 10: Mode2, PA4 is CMOS push-pull output
(R06) PAMODL				**Function related to: Port A**
PA3MOD	06.7~6	R/W	01	00: Mode0, PA3 is open-drain I/O with internal pull-up 01: Mode1, PA3 is open-drain I/O without internal pull-up 10: Mode2, PA3 is CMOS push-pull output
PA2MOD	06.5~4	R/W	01	00: Mode0, PA2 is open-drain I/O with internal pull-up 01: Mode1, PA2 is open-drain I/O without internal pull-up 10: Mode2, PA2 is CMOS push-pull output 11: Mode3, PA2 is RFC0R output
PA1MOD	06.3~2	R/W	01	00: Mode0, PA1 is open-drain I/O with internal pull-up 01: Mode1, PA1 is open-drain I/O without internal pull-up 10: Mode2, PA1 is CMOS push-pull output 11: Mode3, PA1 is RFC2R output
PA0MOD	06.1~0	R/W	01	00: Mode0, PA0 is open-drain I/O with internal pull-up 01: Mode1, PA0 is open-drain I/O without internal pull-up 10: Mode2, PA0 is CMOS push-pull output 11: Mode3, PA0 is RFC1R output

Name	Address	R/W	Rst	Description
(R07) PBMODL				**Function related to: Port B**
PB3MOD	07.7~6	R/W	01	0x: Mode1, PB3 is input without internal pull-up 10: Mode2, PB3 is CMOS push-pull output 11: Mode3, PB3 is LCD SEG8 output
PB2MOD	07.5~4	R/W	01	0x: Mode1, PB2 is input without internal pull-up 10: Mode2, PB2 is CMOS push-pull output 11: Mode3, PB2 is LCD SEG9 output
PB1MOD	07.3~2	R/W	01	0x: Mode1, PB1 is input without internal pull-up 10: Mode2, PB1 is CMOS push-pull output 11: Mode3, PB1 is LCD SEG10 output
PB0MOD	07.1~0	R/W	01	0x: Mode1, PB0 is input without internal pull-up 10: Mode2, PB0 is CMOS push-pull output 11: Mode3, PB0 is LCD SEG11 output
(R08) PWMCTL				**Function related to: PWM/T2/WDT**
PWMCKS	08.7	R/W	1	PWM clock source select 0: Slow-clock 1: Fast-clock
T2PSC	08.6~5	R/W	0	T2 prescaler. T2 interrupt is T2 clock divided by 00: /(32768*60) 01: /32768 10: /16384 11: /4096
PWMPSC	08.4~2	R/W	0	PWM clock prescaler 000: PWM clock is Slow/Fast clock divided by 128 001: PWM clock is Slow/Fast clock divided by 64 010: PWM clock is Slow/Fast clock divided by 32 011: PWM clock is Slow/Fast clock divided by 16 100: PWM clock is Slow/Fast clock divided by 8 101: PWM clock is Slow/Fast clock divided by 4 110: PWM clock is Slow/Fast clock divided by 2 111: PWM clock is Slow/Fast clock divided by 1
PWMNOE	08.1	R/W	0	PWMN output to PA1 pin 0: disable 1: enable
WDTPSC	08.0	R/W	0	WDT timeout select 0: 0.8 second @V_{DD}=3V, 1.6 second @V_{DD}=1.5V 1: 0.4 second @V_{DD}=3V, 0.8 second @V_{DD}=1.5V
(R09) PWMPRD				**Function related to: PWM**
PWMPRD	09.7~0	R/W	FF	PWM period, FFh=256 PWMCLK, 7Fh=128 PWMCLK
(R0A) LVROFF				**Function related to: LVR**
LVROFF	0A.7~0	W	–	Write this register with 0x37 to force LVR disable (Suggestion : M5615 disable LVR after power on)
(R0A) LCDRAM				**Function related to: LCD**
LCDRAM	20~25	R/W	–	LCD RAM

INSTRUCTION SET

Each instruction is a 14-bit word divided into an Op Code, 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 list 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 or/Borrow Flag
DC	Decimal Carry Flag or Decimal/Borrow 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					
ADDWF	f, d	00 0111 dfff ffff	1	C, DC, Z	Add W and "f"
ANDWF	f, d	00 0101 dfff ffff	1	Z	AND W with "f"
CLRF	F	00 0001 1fff ffff	1	Z	Clear "f"
CLRW		00 0001 0100 0000	1	Z	Clear W
COMF	f, d	00 1001 dfff ffff	1	Z	Complement "f"
DECF	f, d	00 0011 dfff ffff	1	Z	Decrement "f"
DECFSZ	f, d	00 1011 dfff ffff	1 or 2	-	Decrement "f", skip if zero
INCF	f, d	00 1010 dfff ffff	1	Z	Increment "f"
INCFSZ	f, d	00 1111 dfff ffff	1 or 2	-	Increment "f", skip if zero
IORWF	f, d	00 0100 dfff ffff	1	Z	OR W with "f"
MOVFW	f	00 1000 0fff ffff	1	-	Move "f" to W
MOVWF	f	00 0000 1fff ffff	1	-	Move W to "f"
MOVWR	r	01 1110 00rr rrrr	1	-	Move W to "r"
MOVRW	r	01 1111 00rr rrrr	1	-	Move "r" to W
RLF	f, d	00 1101 dfff ffff	1	C	Rotate left "f" through carry
RRF	f, d	00 1100 dfff ffff	1	C	Rotate right "f" through carry
SUBWF	f, d	00 0010 dfff ffff	1	C, DC, Z	Subtract W from "f"
SWAPF	f, d	00 1110 dfff ffff	1	-	Swap nibbles in "f"
TESTZ	f	00 1000 1fff ffff	1	Z	Test if "f" is zero
XORWF	f, d	00 0110 dfff ffff	1	Z	XOR W with "f"
Bit-Oriented File Register Instruction					
BCF	f, b	01 000b bbff ffff	1	-	Clear "b" bit of "f"
BSF	f, b	01 001b bbff ffff	1	-	Set "b" bit of "f"
BTFSC	f, b	01 010b bbff ffff	1 or 2	-	Test "b" bit of "f", skip if clear
BTFSS	f, b	01 011b bbff ffff	1 or 2	-	Test "b" bit of "f", skip if set
Literal and Control Instruction					
ADDLW	k	01 1100 kkkk kkkk	1	C, DC, Z	Add Literal "k" and W
ANDLW	k	01 1011 kkkk kkkk	1	Z	AND Literal "k" with W
CALL	k	10 kkkk kkkk kkkk	2	-	Call subroutine "k"
CLRWDT		01 1110 0000 0100	1	TO, PD	Clear Watch Dog Timer
GOTO	k	11 kkkk kkkk kkkk	2	-	Jump to branch "k"
IORLW	k	01 1010 kkkk kkkk	1	Z	OR Literal "k" with W
MOVLW	k	01 1001 kkkk kkkK	1	-	Move Literal "k" to W
NOP		00 0000 0000 0000	1	-	No operation
RET		00 0000 0100 0000	2	-	Return from subroutine
RETI		00 0000 0110 0000	2	-	Return from interrupt
RETLW	k	01 1000 kkkk kkkK	2	-	Return with Literal in W
SLEEP		01 1110 0000 0011	1	TO, PD	Go into Power-down mode, Clock oscillation stops
TABRH		00 0000 0101 1000	2	-	Lookup ROM high data to W
TABRL		00 0000 0101 0000	2	-	Lookup ROM low data to W
XORLW	k	01 1101 kkkk kkkk	1	Z	XOR Literal "k" with W

ELECTRICAL CHARACTERISTICS

1. Absolute Maximum Ratings (T_A =25°C)

Parameter	Rating		Unit
Supply voltage	V_{SS} -0.3 to V_{SS} +3.6		V
Input voltage	V_{SS} -0.3 to V_{BAT} +0.3		
Output voltage	V_{SS} -0.3 to V_{BAT} +0.3		
Output current high per 1 pin / all pins	–20 / –50		mA
Output current low per 1 pin / all pins	+30 / +100		
Maximum operating voltage	3.6		V
Operating temperature	M5610	-40 to +85	°C
	M5615	0 to +85	
Storage temperature	-65 to +150		
Minimum power on voltage	M5610	2V @25°C 2.4V @-40°C	V
	M5615	1.29V @25°C 1.4V @0°C	

2. DC Characteristics (T_A =25°C)

Parameter	Sym	Conditions		Min	Typ	Max	Unit
Input High Voltage	V_{IH}	M5610: V_{BAT}=3V M5615: V_{BAT}=1.5V		0.7V_{BAT}	–	–	V
Input Low Voltage	V_{IL}			–	–	0.2V_{BAT}	
I/O Source/Sink Current	I_{OH}	V_{OH}=2.7V	M5610 V_{BAT}=3V	–	5	–	mA
	I_{OL}	V_{OL}=0.3V		–	15	–	
I/O Source/Sink Current	I_{OH}	V_{OH}=1.3V	M5615 V_{BAT}=1.5V	–	1.2	–	
	I_{OL}	V_{OL}=0.2V		–	4.5	–	
Input leakage current (pin high)	I_{ILH}	all Input	V_{IN}=V_{BAT}	–	–	1	uA
Input leakage current (pin low)	I_{ILL}		Vin=0V	–	–	–1	
Power Supply Current	I_{BAT}	FRC, 3.8MHz	M5610 V_{BAT}=3V V_{DD}=3V	–	360	–	uA
		SRC, 80KHz		–	13	–	
		SXT, 32KHz			12		
		FRC, 1.3MHz	M5610 V_{BAT}=3V V_{DD}=1.5V	–	45	–	
		SRC, 40KHz		–	3	–	
		SXT, 32KHz		–	3	–	
		FRC, 1.3MHz	M5615 V_{BAT}=1.5V	–	60	–	
		SRC, 40KHz		–	3	–	
		SXT, 32KHz		–	3	–	
Timepiece Current CPU Off, LCD On, 32K Crystal oscillating	I_{BAT}	M5610, V_{BAT}=3V, V_{DD}=3V			5		uA
		M5610, V_{BAT}=3V, V_{DD}=1.5V		–	1	–	
		M5615, V_{BAT}=1.5V		–	1.5	–	
Pull-Up Resistor	R_{PU}	M5610, V_{BAT}=3V		–	50	–	KΩ
		M5615, V_{BAT}=1.5V		–	190	–	

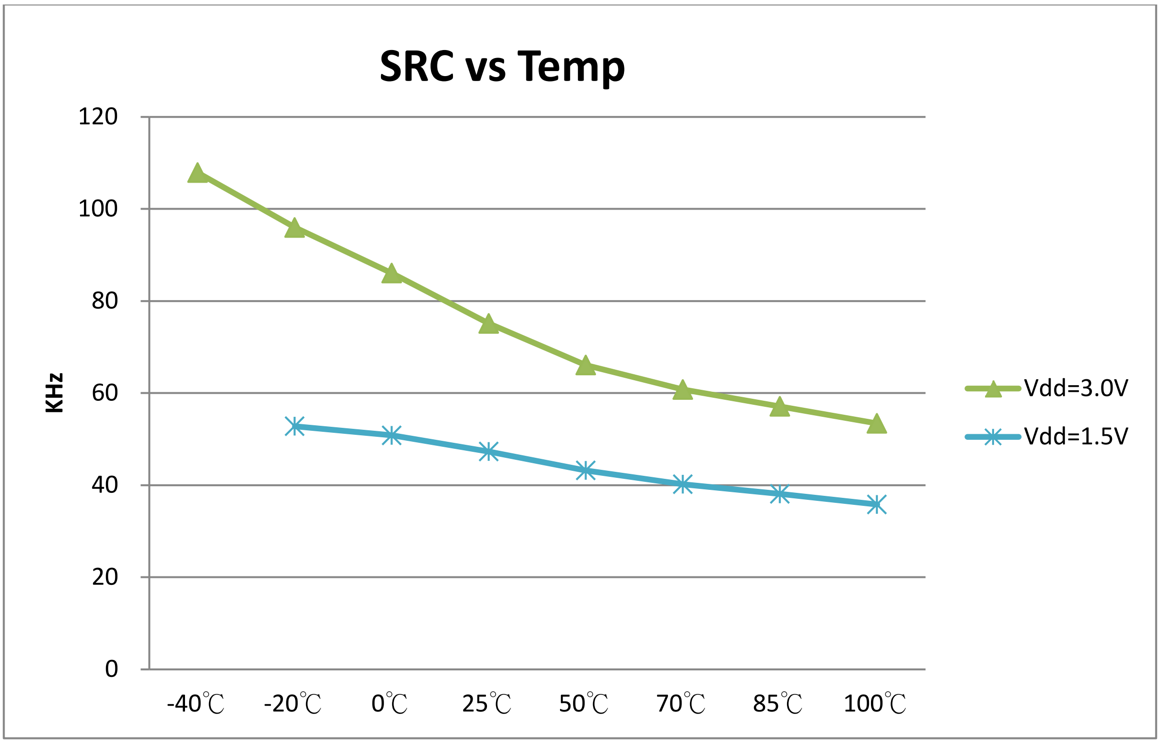
3. Clock Timing (T$_A$ = -40°C to +85°C)

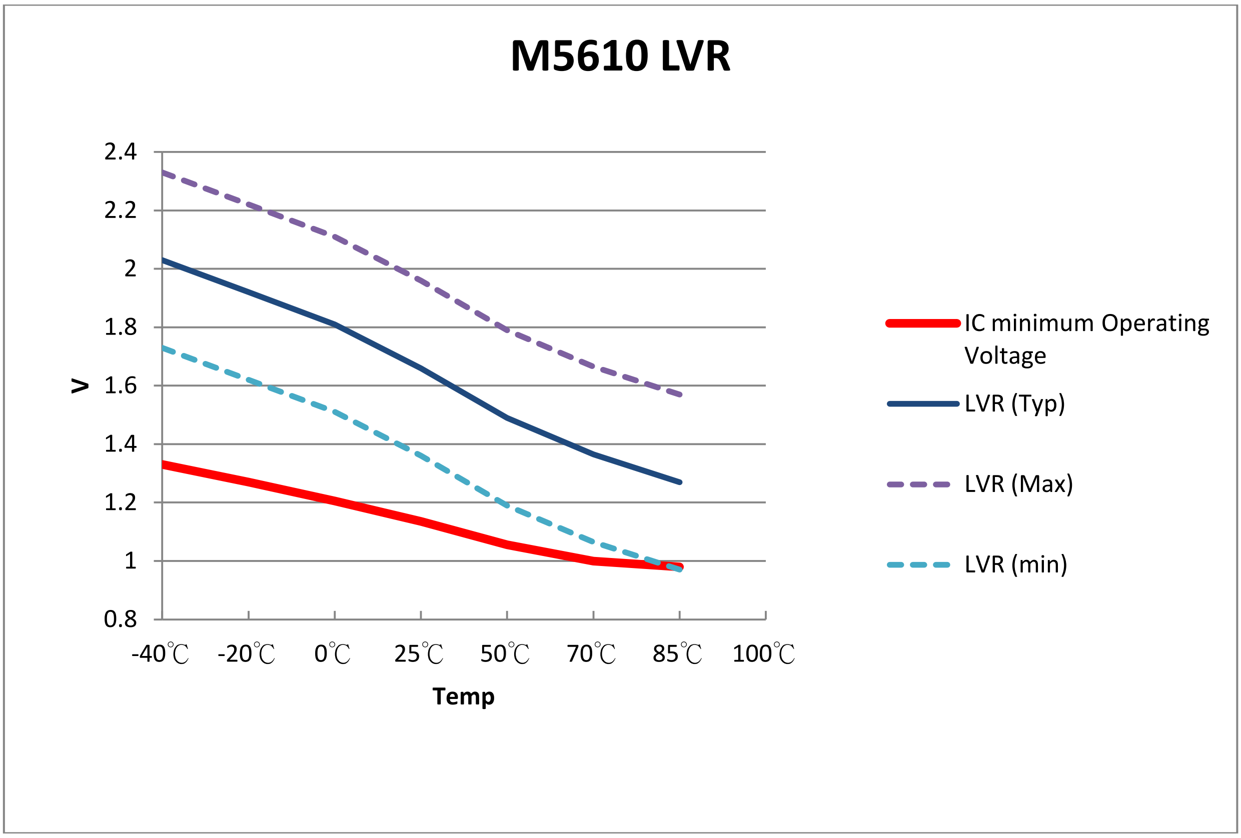
Parameter	Sym	Conditions	Min	Typ	Max	Unit
FRC Clock Frequency	F_{FRC}	V_{DD}=3V	–	3.8	–	MHz
		V_{DD}=1.5V	–	1.3	–	
SRC Clock Frequency	F_{SRC}	V_{DD}=3V	–	80	–	KHz
		V_{DD}=1.5V	–	40	–	

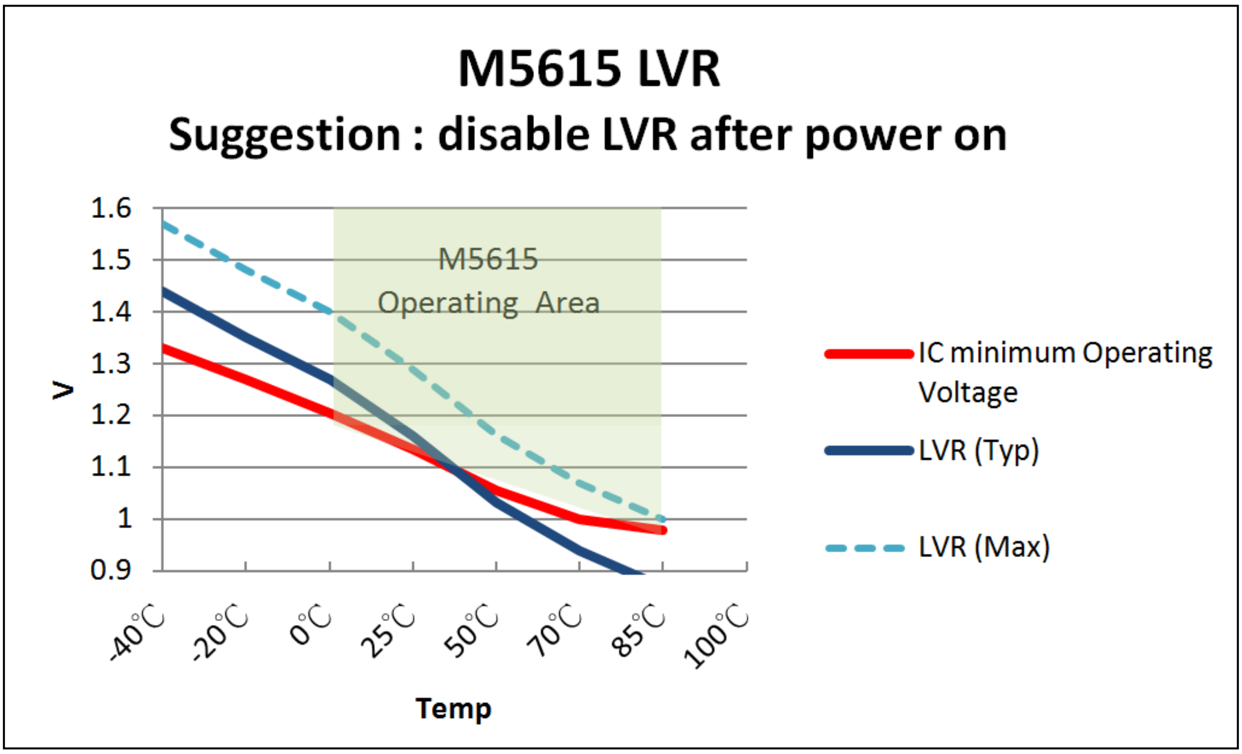
4. BandGap Reference Voltage

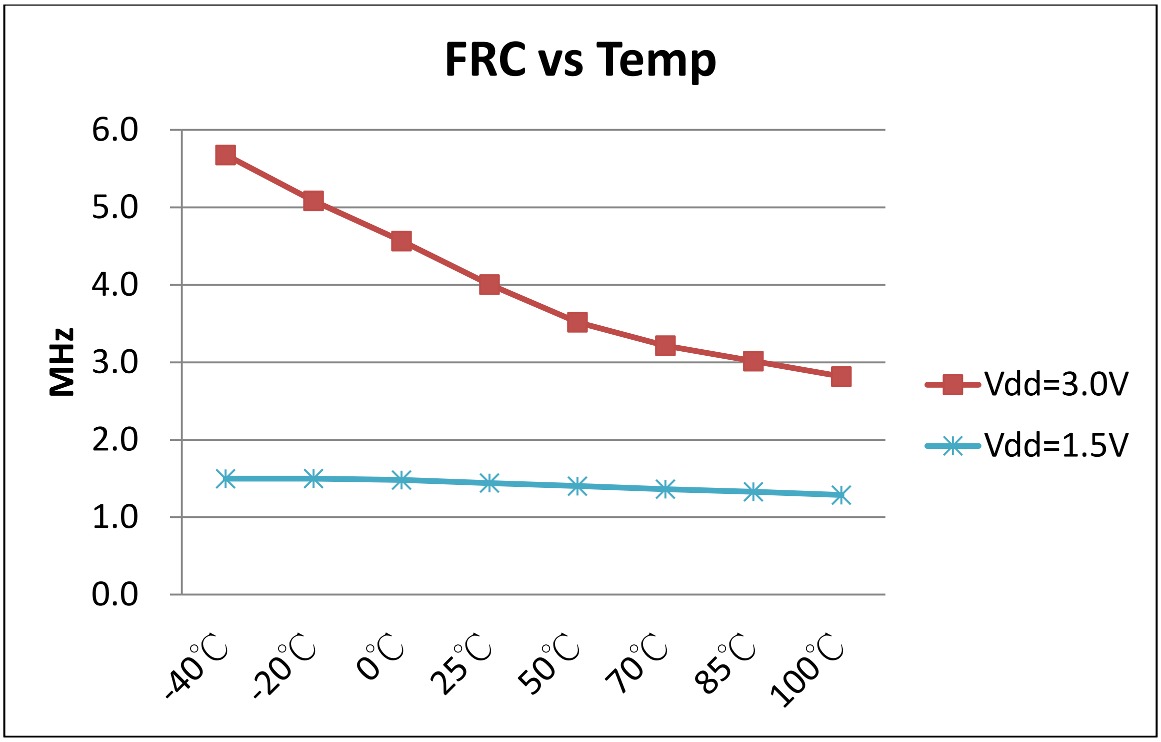
Parameter	Sym	Conditions	Min	Typ	Max	Unit
BandGap Voltage	V_{BG}	V_{BAT}=3V, 25°C	1.14	1.2	1.26	V
		V_{BAT}=3V, –40°C~85°C	1.12	1.2	1.28	

5. Characteristic Graphs




M5610 LVR
V
2.4
2.2
2
1.8
1.6
1.4
1.2
1
0.8
-40℃
-20℃
0℃
25℃
50℃
70℃
85℃
100℃
Temp
IC minimum Operating Voltage
LVR (Typ)
LVR (Max)
LVR (min)


M5615 LVR
Suggestion : disable LVR after power on
V
1.6
1.5
1.4
1.3
1.2
1.1
1
0.9
M5615
Operating Area
-40℃
-20℃
0℃
25℃
50℃
70℃
85℃
100℃
Temp
IC minimum Operating Voltage
LVR (Typ)
LVR (Max)


FRC vs Temp
MHz
6.0
5.0
4.0
3.0
2.0
1.0
0.0
-40℃
-20℃
0℃
25℃
50℃
70℃
85℃
100℃
Vdd=3.0V
Vdd=1.5V

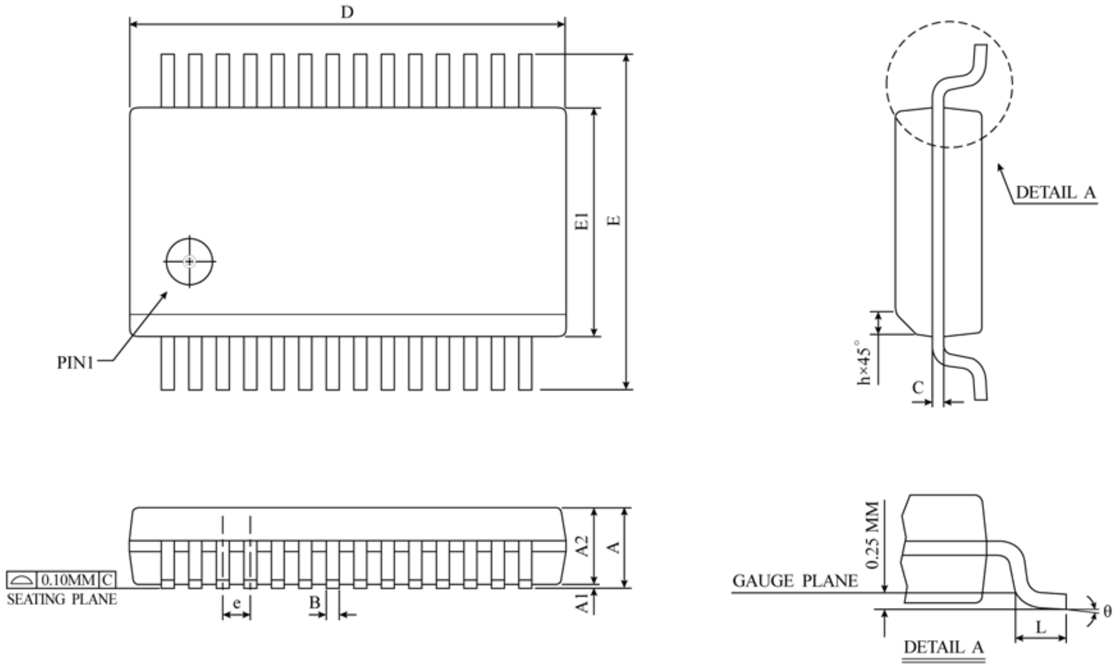
PACKAGING INFORMATION

Please note that the package information provided is for reference only. Since this information is frequently updated, users can contact Sales to consult the latest package information and stocks.

The ordering information:

Ordering number	Package
TM57M5610-MTP	Wafer/Dice blank chip
TM57M5610-COD	Wafer/Dice with code
TM57M5610-MTP-29	SSOP 28-pin (150 mil)
TM57M5610-MTP-46	TSSOP 20-pin (173 mil)
TM57M5615-MTP	Wafer/Dice blank chip
TM57M5615-COD	Wafer/Dice with code
TM57M5615-MTP-29	SSOP 28-pin (150 mil)
TM57M5615-MTP-46	TSSOP 20-pin (173 mil)

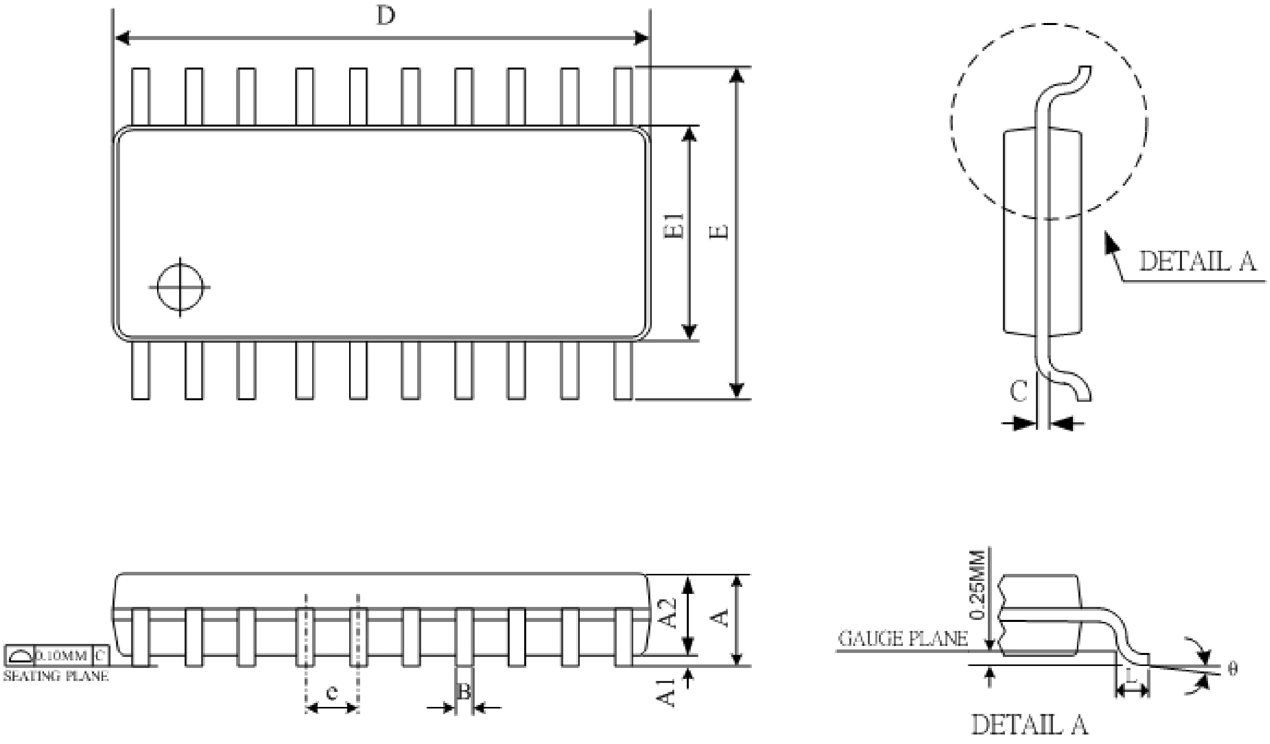
SSOP-28 (150mil) Package Dimension



SYMBOL	DIMENSION IN MM			DIMENSION IN INCH		
	MIN	NOM	MAX	MIN	NOM	MAX
A	1.50	1.65	1.80	0.06	0.06	0.07
A1	0.102	0.176	0.249	0.004	0.007	0.010
A2	1.40	1.475	1.55	0.06	0.06	0.06
B	0.20	0.25	0.30	0.01	0.01	0.01
C	0.2TYP			0.008TYP		
e	0.635TYP			0.025TYP		
D	9.804	9.881	9.957	0.386	0.389	0.392
E	5.842	6.020	6.198	0.230	0.237	0.244
E1	3.86	3.929	3.998	0.152	0.155	0.157
L	0.406	0.648	0.889	0.016	0.026	0.035
θ	0°	4°	8°	0°	4°	8°
JEDEC	M0-137(AF)					

⚠B *NOTES：DIMENSION “D” DOES NOT INCLUDE MOLD PROTRUSIONS OR GATE BURRS.
MOLD PROTRUSIONS AND GATE BURRS SHALL NOT EXCEED 0.006 INCH PER SIDE.

TSSOP-20 (173mil) Package Dimension



SYMBOL	DIMENSION IN MM			DIMENSION IN INCH		
	MIN	NOM	MAX	MIN	NOM	MAX
A	-	-	1.2	-	-	0.047
A1	0.05	0.10	0.15	0.002	0.004	0.006
A2	0.8	0.93	1.05	0.031	0.036	0.041
B	0.19	-	0.3	0.007	-	0.012
D	6.4	6.5	6.6	0.252	0.256	0.260
E	6.25	6.4	6.55	0.246	0.252	0.258
E1	4.3	4.4	4.5	0.169	0.173	0.177
e	0.65 BSC			0.026 BSC		
L	0.45	0.60	0.75	0.018	0.024	0.030
θ	0 °		8 °	0 °		8 °
JEDEC	MO-153 AC REV.F					

Notes：

1.DIMENSION "D" DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS. MOLD FLASH, PROTRUSIONS OR GATE BURRS SHALL NOT EXCEED 0.15 PER SIDE.

2.DIMENSION "E1" DOES NOT INCLUDE INTERLEAD FLASH OR PROTRUSION. INTERLEAD FLASH OR PROTRUSION SHALL NOT EXCEED 0.25 PER SIDE.

3.DIMENSION "B" DOES NOT INCLUDE DAMBAR PROTRUSION.ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.08MM TOTAL IN EXCESS OF THE "B" DIMENSION AT MAXIMUM METERIAL CONDITION. DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OF THE FOOT. MINIMUM SPACE BETWEEN PROTRUSION AND ADJACENT LEAD IS 0.07MM.