十速

TM57ME20

DATA SHEET

Rev V2.0

AMENDMENT HISTORY

Version	Date	Description
V1.0	Aug, 2010	New release
V1.1	Dec, 2010	1. Add more description about /Borrow and /Digit Borrow in ALU and Working (W) Register section. 2. Add Internal RC mode description and figure in System Clock Oscillator section. 3. Modify the status affected of the NOP instruction.
V1.2	Feb, 2011	1. Modify IVCPD description in System Configuration Register (SYSCFG) section.
V1.3	May, 2011	1. Add operating voltage selection in System Configuration Register (SYSCFG) section. 2. Add 32 KHz operating current in Electrical Characteristics section.
V1.4	Oct, 2011	Modify the package type data.
V1.5	Dec, 2011	Add Ordering Information table in the Packaging Information section.
V1.6	Jan, 2012	1. Add the Electrical Characteristics specs in the Features section. 2. Add description in Reset section. 3. Merge the information about LVR Circuit Characteristics into DC Characteristics table.
V1.7	Jul, 2012	Modify document format.
V1.8	Apr, 2013	1. Modify Block Diagram. 2. Modify Packaging Information.
V1.9	Jul, 2013	1. Add supported EV board on ICE. 2. Modify pin assignment name. 3. Add pin summary.
V2.0	Aug, 2013	1. Modify 32-DIP/SOP pin assignment. 2. Modify Interrupt description. 3. Modify Ordering Information

CONTENTS

FEATURES

1. MTP: 2K x 14 bits MTP ROM (Support ISP uses 5 wires)
2. RAM: 96 x 8 bits
3. STACK: 8 Levels
4. I/O ports: Four Bit programmable I/O ports (Max. 29 pins)
5. Timer0/Counter: 8-bit timer/counter with divided by 1~256 pre-scale option, stop counting
6. Timer1: 16-bit auto-reloadable timer with divided by 1~256 pre-scale option
7. Timer2:
   - 15-bit Timer2 with divided by 2-bit pre-scale option
   - 15-bit Timer2 with 4 interrupt interval option

   Timer2 is used to idle mode wake-up timer or one simple 15-bit time base
8. Two 8+2 bits PWM channels capable of 1024 duty resolution and 256 period resolution
9. Two analog voltage comparators
10. PB0~PB7 individual pin low level wake up
11. Oscillation Sources
    - Fast Clock:
      - FXT (Fast Crystal): 1 MHz~12 MHz
      - FIRC (Fast Internal RC): 4 MHz
      - XRC (External R, External C): 10 KHz~3 MHz
    - Slow Clock:
      - SXT (Slow Crystal): 32768 Hz
      - XRC (External R, External C): 10 KHz~3 MHz
      - SIRC (Slow Internal RC): 138 KHz/35 KHz/8.5 KHz/2.1 KHz, @5V; 119 KHz/30 KHz/7.5 KHz/1.9 KHz, @3V
12. Power Saving Operation Mode
    - Fast Mode: Slow clock can be disabled or enabled
    - Slow Mode: Fast clock stops, CPU is running
    - Idle Mode: Slow clock is running, CPU stops, Timer2 is running
    - Stop Mode: All clocks stop, Wake-up Timer is disabled or enabled

13. Dual system clock

- FIRC + SIRC
- FIRC + SXT
- FIRC + XRC
- FXT + SIRC
- XRC + SIRC

14. Reset

- Power On Reset
- Watchdog Reset
- Low Voltage Reset
- External Pin Reset

15. 2-Level Low Voltage Reset: 2.2V/3.2V (Can be disabled)

16. Operation Voltage: Low Voltage Reset Level to 5.5V

- fosc = 4 MHz, 2.2V ~ 5.5V
- fosc = 8 MHz, 2.3V ~ 5.5V
- fosc = 12 MHz, 2.6V ~ 5.5V
- fosc = 16 MHz, 3.3V ~ 5.5V

17. Interrupts

- Two External Interrupt pins:
  - One pin is falling edge triggered
  - One pin is rising or falling edge triggered
- Timer0, Timer1, Timer2, Wake-up Timer Interrupt
- CP0, CP1 Interrupt

18. Watchdog Timer

- Clocked by built-in RC oscillator with 4 adjustable reset/interrupt time durations (111 ms/57 ms/28 ms/14 ms, @5V; 121 ms/61 ms/30 ms/15 ms, @3V)
- Watchdog timer can be disabled/enabled in stop mode

19. Support auto store/restore STATUS and W before/after interrupt routine

20. I/O Ports

- CMOS Output
- Pseudo-Open-Drain or Open-Drain Output
- Schmitt Trigger Input with/without pull-up resistor

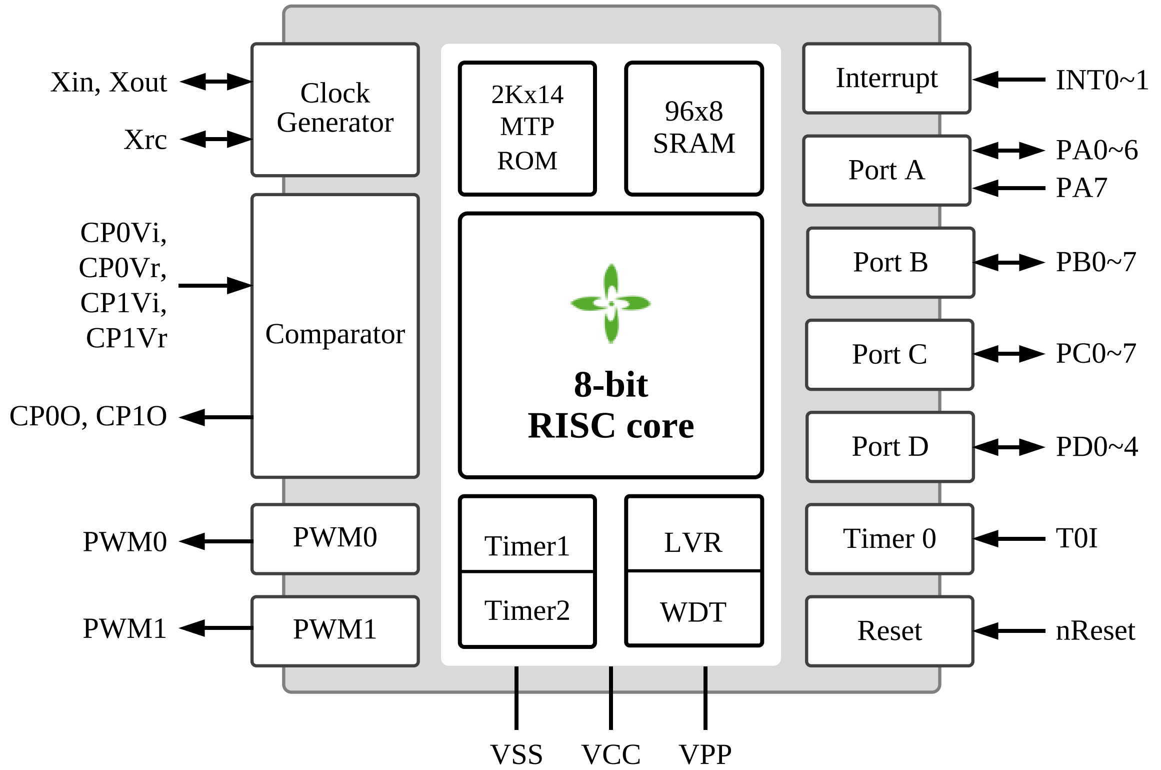
21. Instruction Set: 36 Instructions

22. Package Types: 28-DIP/SOP, 32-DIP/SOP

23. Supported EV board on ICE

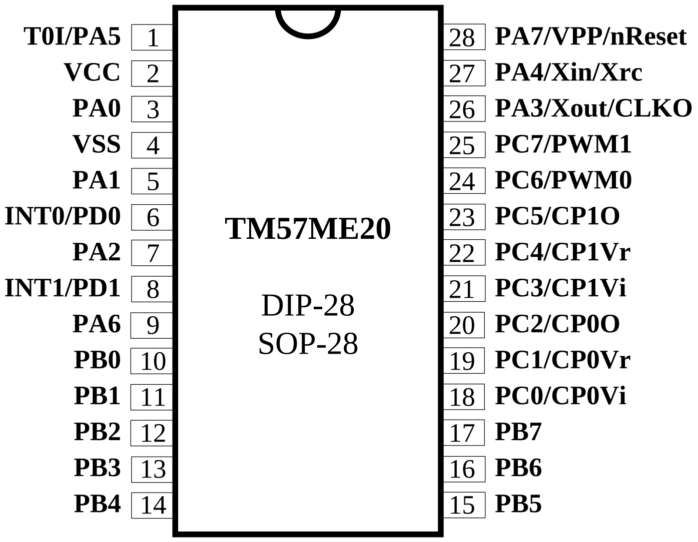
EV board: EV2788

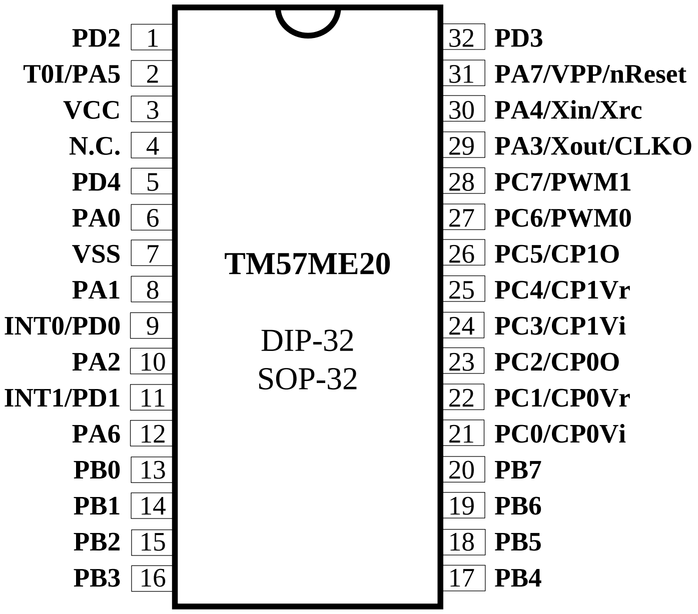
BLOCK DIAGRAM



TM57ME20 Block Diagram

PIN ASSIGNMENT





PIN DESCRIPTION

Name	In/Out	Pin Description
PA0–PA2	I/O	Bit-programmable I/O port for Schmitt-trigger input, CMOS push-pull output or "pseudo-open-drain" output. Pull-up resistors are assignable by software.
PA3–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.
PA7	I	Schmitt-trigger input
PB0–PB7	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.
PC0–PC7	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.
PD0–PD4	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.
nRESET	I	External active low reset
Xin, Xout	–	Crystal/Resonator oscillator connection for system clock
Xrc	–	External RC oscillator connection for system clock
CLKO	O	CPU Instruction clock output for external/internal RC mode
VCC, VSS	P	Power Voltage input pin and ground
VPP	I	PROM programming high voltage input
INT0–INT1	I	External interrupt input
CP0Vi, CP1Vi	I	Comparator voltage input
CP0Vr, CP1Vr	I	Comparator reference voltage input
CP0O, CP1O	O	Comparator output
PWM0–PWM1	O	PWM output
T0I	I	Clock input to Timer0

PIN SUMMARY

Pin Number		Pin Name	Type	GPIO					Function After Reset	Alternate Function			
				Input		Output							
32-SOP/DIP	28-SOP/DIP			Weak Pull-up	Ext. Interrupt	O.D	P.O.D	P.P		PWM	Touch Key	ADC	MISC
1	-	PD2	I/O	○		○		○	PD2				
2	1	PA5/T0I	I/O	○		○		○	PA5				T0I
3	2	VCC	P										
4	-	NC	-										
5	-	PD4	I/O	○		○		○	PD4				
6	3	PA0	I/O	○			○	○	PA0				
7	4	VSS	P										
8	5	PA1	I/O	○			○	○	PA1				
9	6	PD0/INT0	I/O	○	○	○		○	PD0				
10	7	PA2	I/O	○			○	○	PA2				
11	8	PD1/INT1	I/O	○	○	○		○	PD1				
12	9	PA6	I/O	○		○		○	PA6				
13	10	PB0	I/O	○		○		○	PB0				
14	11	PB1	I/O	○		○		○	PB1				
15	12	PB2	I/O	○		○		○	PB2				
16	13	PB3	I/O	○		○		○	PB3				
17	14	PB4	I/O	○		○		○	PB4				
18	15	PB5	I/O	○		○		○	PB5				
19	16	PB6	I/O	○		○		○	PB6				
20	17	PB7	I/O	○		○		○	PB7				
21	18	PC0/CP0Vi	I/O	○		○		○	PC0				CP0Vi
22	19	PC1/CP0Vr	I/O	○		○		○	PC1				CP0Vr
23	20	PC2/CP0O	I/O	○		○		○	PC2				CP0O
24	21	PC3/CP1Vi	I/O	○		○		○	PC3				CP0Vi
25	22	PC4/CP1Vr	I/O	○		○		○	PC4				CP1Vr
26	23	PC5/CP1O	I/O	○		○		○	PC5				CP1O
27	24	PC6/PWM0	I/O	○		○		○	PC6	○			
28	25	PC7/PWM1	I/O	○		○		○	PC7	○			
29	26	PA3/Xout/CLKO	I/O	○		○		○	PA3				CLKO
30	27	PA4/Xin/Xrc	I/O	○		○		○	PA4				
31	28	PA7/VPP/nReset	I	○					PA7				nReset
32	-	PD3	I/O	○		○		○	PD3				

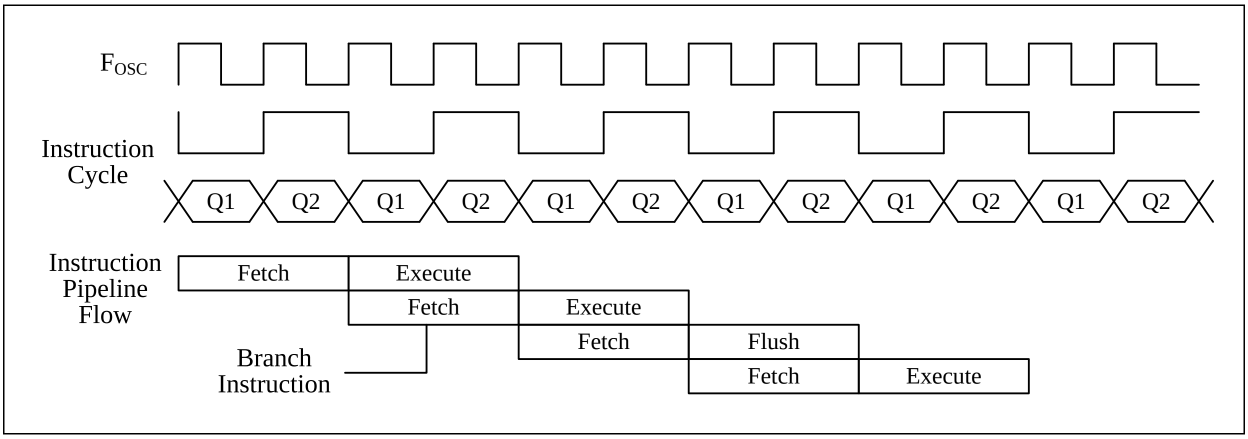
Symbol：P.P. = Push-Pull Output
P.O.D. = Pseudo Open Drain
O.D. = Open Drain

FUNCTIONAL DESCRIPTION

1. CPU Core

1.1 Clock Scheme and Instruction Cycle

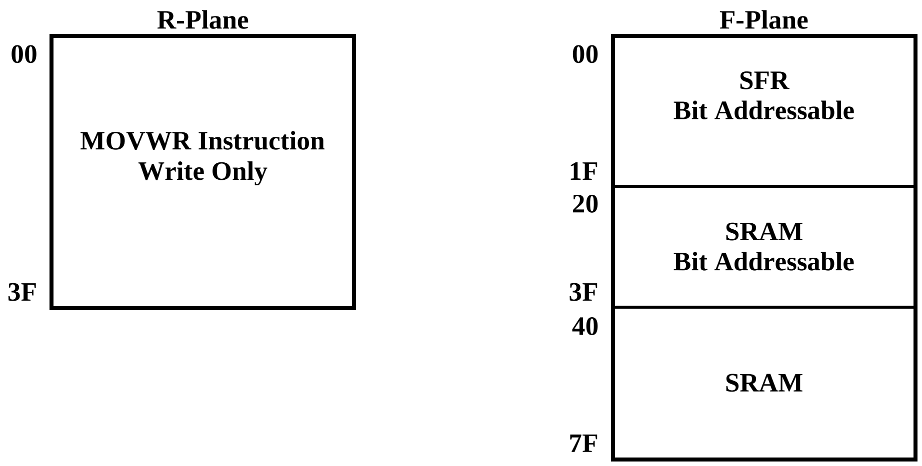
The system clock is internally divided by two to generate Q1 state and Q2 state for each instruction cycle. The Programming Counter (PC) is updated at Q1 and the instruction is fetched from program ROM and latched into the instruction register in Q2. It is then decoded and executed during the following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

There are two Data Memory Planes in CPU, R-Plane and F-Plane. The registers in R-Plane are write-only. The “MOVWR” instruction copy the W-register’s content to R-Plane registers by direct addressing mode.

The lower locations of F-Plane are reserved for the 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.



1.3 Programming Counter (PC) and Stack

The Programming Counter is 11-bit wide capable of addressing a 2K 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 11 bits address from instruction word. For RET/RETI/RETLW instructions, PC retrieves its content from the top level STACK. For the other instructions updating PC [7:0], the PC [10:8] keeps unchanged. The STACK is 11-bit wide and 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.

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 STATUS Register

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.

<table>
<tr><th>STATUS</th><th>Bit 7</th><th>Bit 6</th><th>Bit 5</th><th>Bit 4</th><th>Bit 3</th><th>Bit 2</th><th>Bit 1</th><th>Bit 0</th></tr>
<tr><td>Reset Value</td><td>–</td><td>–</td><td>–</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>R/W</td><td>–</td><td>–</td><td>–</td><td>R</td><td>R</td><td>R/W</td><td>R/W</td><td>R/W</td></tr>
</table>

<table>
<tr><th>Bit</th><th colspan="2">Description</th></tr>
<tr><td>7-5</td><td colspan="2">Not Used</td></tr>
<tr><td>4</td><td colspan="2">TO: Time Out
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">PD: Power Down
0: after Power On Reset, LVR Reset, or CLRWDT instruction
1: after SLEEP instruction</td></tr>
<tr><td>2</td><td colspan="2">Z: 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">DC: Decimal Carry Flag or Decimal /Borrow Flag</td></tr>
<tr><td>ADD instruction</td><td>SUB instruction</td></tr>
<tr><td>1: a carry from the low nibble bits of the result occurs
0: no carry</td><td>1: no borrow
0: a borrow from the low nibble bits of the result occurs</td></tr>
<tr><td rowspan="3">0</td><td colspan="2">C: Carry Flag or /Borrow Flag</td></tr>
<tr><td>ADD instruction</td><td>SUB instruction</td></tr>
<tr><td>1: a carry occurs from the MSB
0: no carry</td><td>1: no borrow
0: a borrow occurs from the MSB</td></tr>
</table>

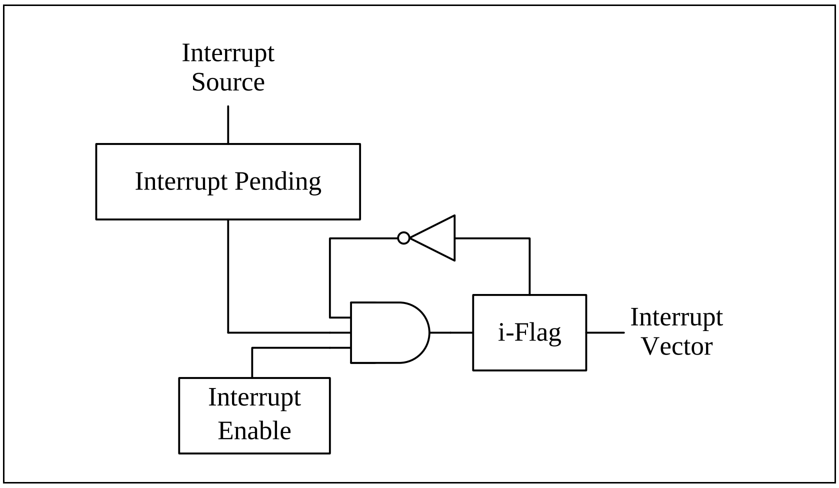
1.6 Interrupt

The TM57ME20 has 1 level, 1 vector and 8 interrupt sources. Each interrupt source has its own enable control bit. An interrupt event will set its individual pending flag; no matter its interrupt enable control bit is 0 or 1. Because TM57ME20 has only 1 vector, there is not an interrupt priority register. The interrupt priority is determined by F/W.

If the corresponding interrupt enable bit has been set (F-Plane 08h), 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.

The STATUS and W register can be automatically stored into the internal memory when interrupt and be restored when exit from interrupt. This functionality is optional and can be enabled or disabled via HWAUTO which in R-Plane 0BH Bit7.



2. Chip Operation Mode

2.1 Reset

The TM57ME20 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 clock source, LVR level and chip operation mode are selected by the SYSCFG register value. The Low Voltage Reset features static reset when supply voltage is below a threshold level. There are two threshold levels can be selected. The LVR's operation mode is defined by the SYSCFG register.

There are two voltage selections for the LVR threshold level, one is higher level which is suitable for application with V_{CC} is more than 3.3V, while another one is suitable for application with V_{CC} is less than 3.3V. See the following LVR Selection Table; user must also consider the lowest operating voltage of operating frequency.

LVR Selection Table:

LVR Threshold Level	Consider the operating voltage to choose LVR
LVR3.2	$5.5V > V_{CC} > 3.3V$
LVR2.2	V_{CC} is wide voltage range, more or less than 3.3V

The External Pin Reset and Watchdog Reset can be disabled or enabled by the SYSCFG register. These two resets also set all the control registers to their default reset values. The TO/PD flag is not affected by these resets.

2.2 System Configuration Register (SYSCFG)

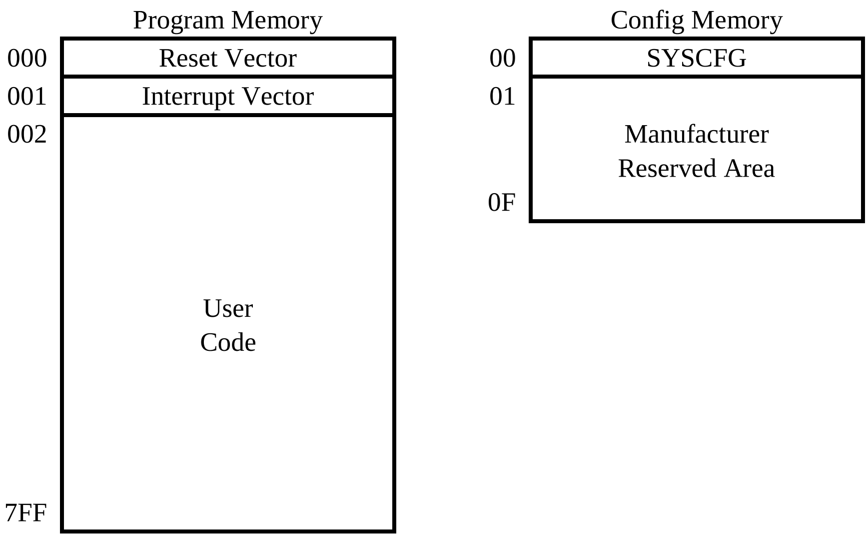
The System Configuration Register (SYSCFG) is located at MTP INFO area. The SYSCFG determines the option for initial condition of MCU. It is written by MTP Writer only. User can select clock source, LVR threshold voltage and chip operation mode by SYSCFG register. The 13th bit of SYSCFG is code protection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

Bit	**13~0**	
Default Value	**00_0000_0000_0000**	
Bit	Description	
13	**PROTECT**: Code Protection Selection	
	1	Code Protection
	0	No Protect
12	**ICVPD**: IVC*/LVR Mode Selection	
	1	IVC/LVR Auto OFF in Idle/Stop Mode
	0	IVC/LVR Always ON
11	**LVR**: LVR Threshold	
	1	LVR threshold is 2.2V, always enable
	0	LVR threshold is 3.2V, always enable
10	**LVRE**: LVR Enable	
	1	Enable LVR
	0	Disable LVR
9-8	**CLKS**: Clock Source Selection	
	11	Fast Crystal (1 MHz~12 MHz)
	10	Slow Crystal (32 KHz~1 MHz)
	01	Fast Internal RC (4 MHz)
	00	External RC
7	**XRESETE**: External Pin Reset Enable	
	1	Enable External Pin Reset
	0	Disable External Pin Reset
6	**WDTE**: WDT Reset Enable	
	1	Enable WDT Reset, Disable WKT Timer
	0	Disable WDT Reset, Enable WKT Timer
5	**3V/5V Selection**: Operating Voltage Selection	
	1	V_{CC} maximum operating voltage at 3.3V
	0	V_{CC} maximum operating voltage at 5.5V
4-0	**FIRCF**: Fast Internal RC Frequency Adjustment Control	

* IVC is the chip built-in 3.3V regulator for internal circuit.

2.3 MTP Program ROM

The MTP ROM of this device is 2K words, with an extra INFO area to store the SYSCFG. 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 not set or MTP ROM is erased. That is, un-protect the PROTECT bit needs the erased MTP ROM.

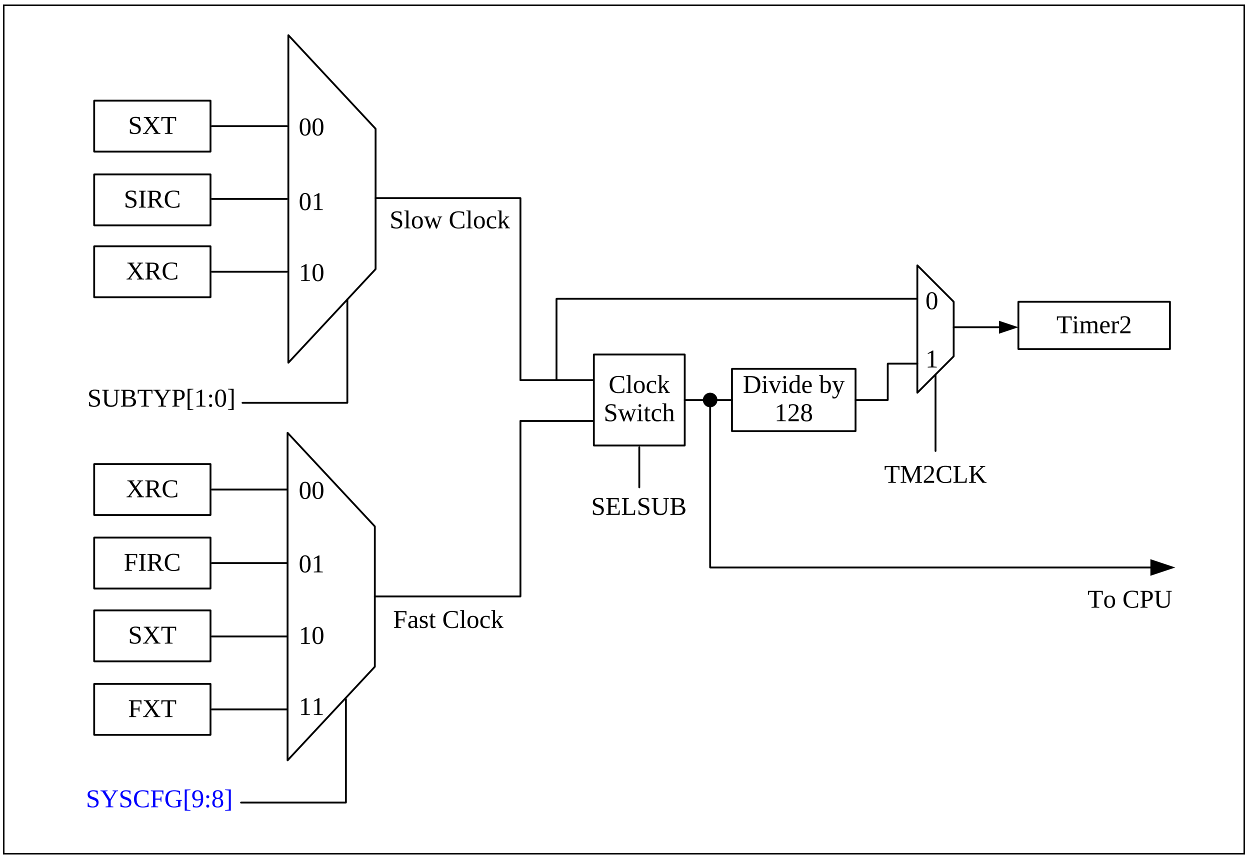


2.4 Power-Down Mode

The Power-down mode is activated by SLEEP instruction. During the Power-down mode, the system clock and peripherals stop to minimize power consumption, while the WDT/WKT Timer is working or not depends on F/W setting. The Power-down mode can be terminated by reset, or enabled interrupts (external pins and WKT interrupt) or PB0-7 pins low level wakeup. In the Power-down mode, user can enable or disable IVC according to the standby current requirement. Enabled IVC can provide the chip internal circuit more stable 3.3V power.

2.5 Dual System Clock

TM57ME20 is designed with dual-clock system. There are five kinds of clock source, FXT (Fast Crystal) Clock, SXT (Slow Crystal) Clock, XRC (External RC) Clock, SIRC (Slow Internal RC) Clock and FIRC (Fast Internal RC). Each clock source can be applied to CPU kernel as system clock. When in idle mode, only slow clock can be configured to keep oscillating to provide clock source to Timer2. Refer to the Figure as below.



Fast Mode:

After power on or reset, TM57ME20 enters fast mode. In fast mode, TM57ME20 can select FXT, XRC or FIRC as its CPU clock by SYSCFG bit9 and bit8 setting. Besides, firmware can also enable or disable the slow clock for the Timer2 system operating. In this mode, the program is executed using fast clock as CPU clock. The Timer0, PWM0, PWM1 blocks are also driven by fast clock. Timer2 can also be driven by fast clock by setting TM2CLK to “1”.

Slow Mode:

In slow mode, TM57ME20 can select SXT, XRC or SIRC as its CPU clock by R-Plane control register (SUBTYP). In this mode, the fast clock is stopped and slow clock is enabled for power saving. All peripheral blocks clock sources are slow clock in the slow mode.

Idle Mode:

If slow clock is enabled and TM2CLK=0 before executing the SLEEP instruction, the TM57ME20 enters the “Idle Mode”. In this mode, the slow clock will continue running to provide clock to Timer2 block. CPU stop fetching code and all blocks are stop except Timer2 related circuits.

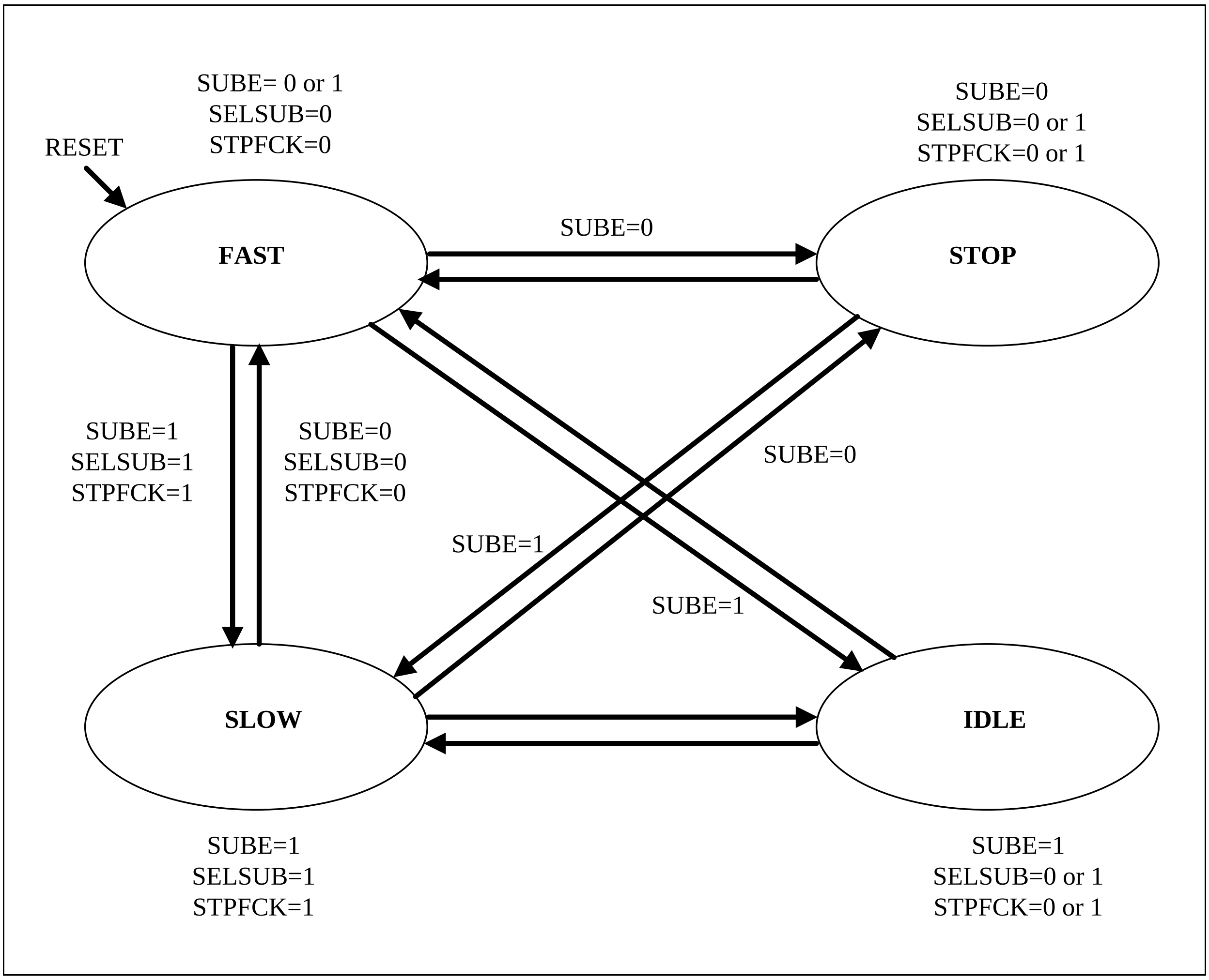
Stop Mode:

If slow clock is disabled before executing the SLEEP instruction, every block is turned off and the TM57ME20 enters the “Stop Mode”. Stop mode is similar to idle mode. The difference is all clock oscillators either fast or slow is powered-down and no clock is generated.

2.6 Dual System Clock Modes Transition

TM57ME20 is operated in one of four modes: Fast Mode, Slow Mode, Idle Mode, and Stop Mode.

Modes Transition Diagram:



Fast Mode transits to Slow Mode:

Fast mode can be chosen by SYSCFG [9:8] when equals to 11 (Fast Crystal), 00 (External RC), or 01 (Fast Internal RC). The following steps are suggested to be executed by order when fast mode transits to slow mode:

(1) Enable slow clock (SUBE=1)

(2) Switch to slow clock (SELSUB=1)

(3) Stop fast clock (STPFCK=1)

Slow Mode transits to Fast Mode:

Slow mode can be enabled by SUBE bit and SELSUB bit in CLKCTRL register. The following steps are suggested to be executed by order when slow mode transits to fast mode:

(1) Enable fast clock (STPFCK=0)

(2) Switch to fast clock (SELSUB=0)

(3) Stop slow clock (SUBE=0)

Note: Stop slow clock (SUBE=0) is optional. Slow clock can keep oscillating to provide Timer2 counter block in fast mode.

Idle Mode Setting:

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

(1) Enable slow clock (SUBE=1)

(2) Switch Timer2 clock source to slow clock (TM2CLK=0)

(3) Execute SLEEP instruction

Idle mode can be woken up by XINT, PBWAKP, Wake-up Timer, and Timer2 interrupt.

Stop Mode Setting:

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

(1) Stop slow clock (SUBE=0)

(2) Execute SLEEP instruction

Stop mode can be woken up by XINT, PBWAKP, and Wake-up Timer.

IO setting note in dual clock mode:

Note: In slow clock modes, PA3 and PA4 must be set as input pull-up mode when slow clock selects SXT or XRC mode. PA3 and PA4 IO setting list is as shown bellow.

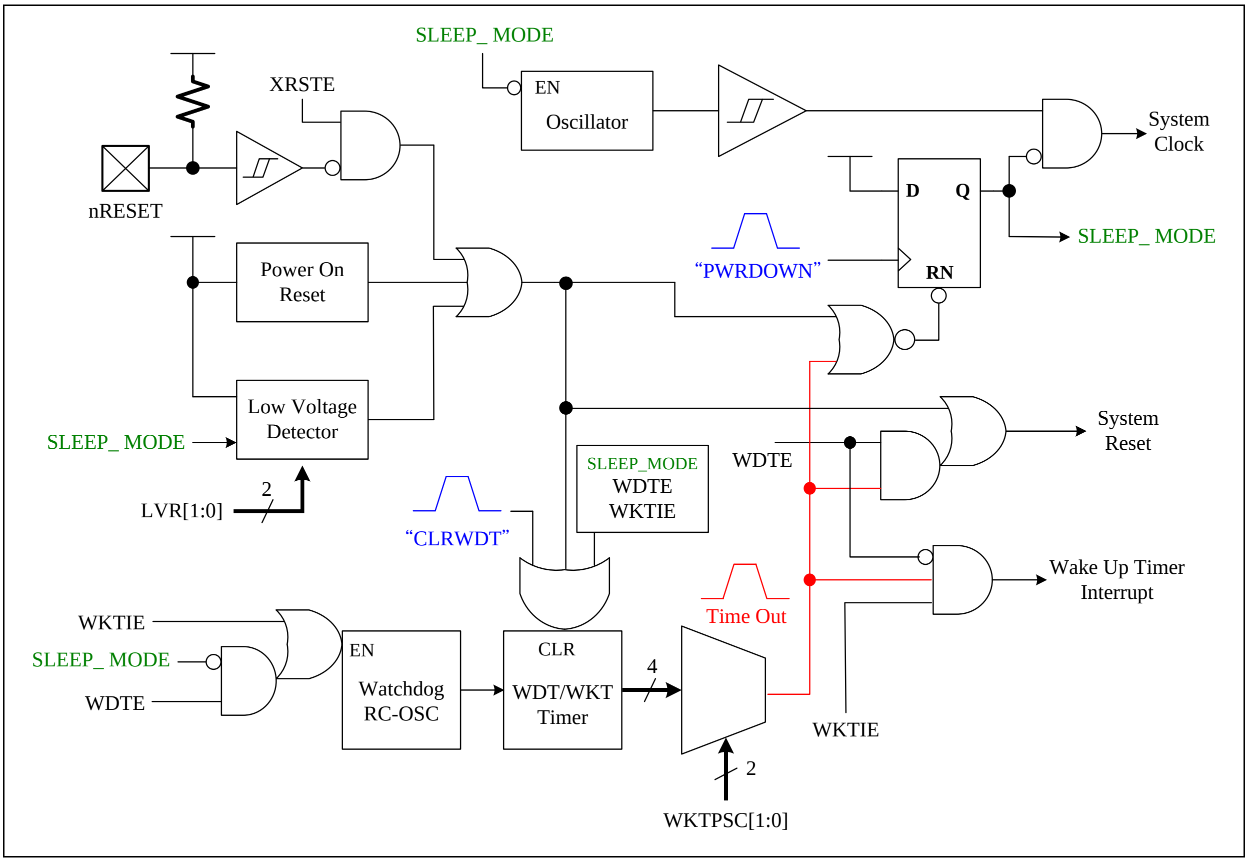
	Fast Clock	Slow Clock	PAD3	PAE3	nPAPU3	PAD4	PAE4	nPAPU4
1	FIRC	SIRC	※	※	※	※	※	※
2	FIRC	SXT	1	0	0	1	0	0
3	FIRC	XRC	※	※	※	1	0	0
4	FXT	SIRC	※	※	※	※	※	※
5	XRC	SIRC	※	※	※	※	※	※

※：Don’t care

3. Peripheral Functional Block

3.1 Watchdog (WDT) / Wakeup (WKT) Timer

The WDT and WKT share the same internal RC Timer. The overflow period of WDT/WKT can be selected from 14 ms to 121 ms. The WDT/WKT is cleared by the CLRWDT instruction. If the Watchdog Reset is enabled (WDTE=1), the WDT generates the chip reset signal, otherwise, the WKT only generates overflow time out interrupt. The WDT/WKT works in both normal mode and sleep mode. During sleep mode, user can further choose to enable or disable the WDT/WKT by "WKTIE". If WKTIE=0 in sleep mode (no matter WDTE is 1 or 0), the internal RC Timer stops for power saving. In other words, user keeps the WDT/WKT alive in sleep mode by setting WKTIE=1. If the WDTE=1 and WKTIE=0, WDT/WKT timer will be cleared and stopped to power saving in sleep mode. If the WDTE=1 and WKTIE=1, WDT/WKT timer keeps counting in sleep/normal mode. Refer to the following table and figure.

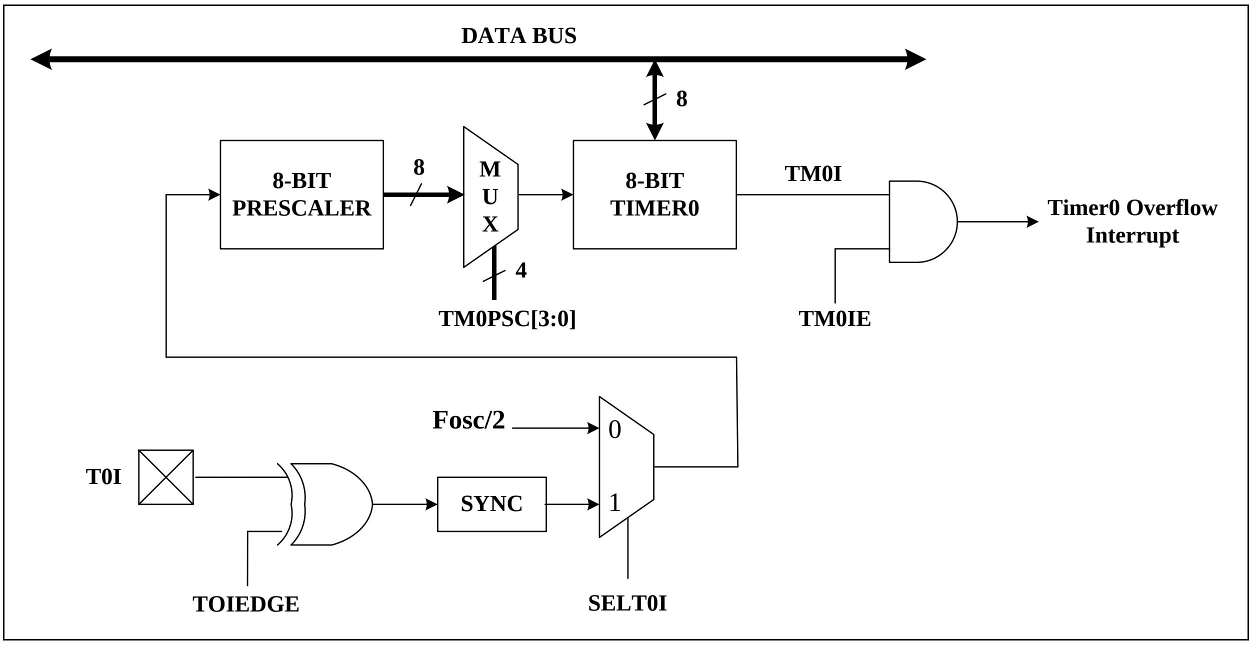


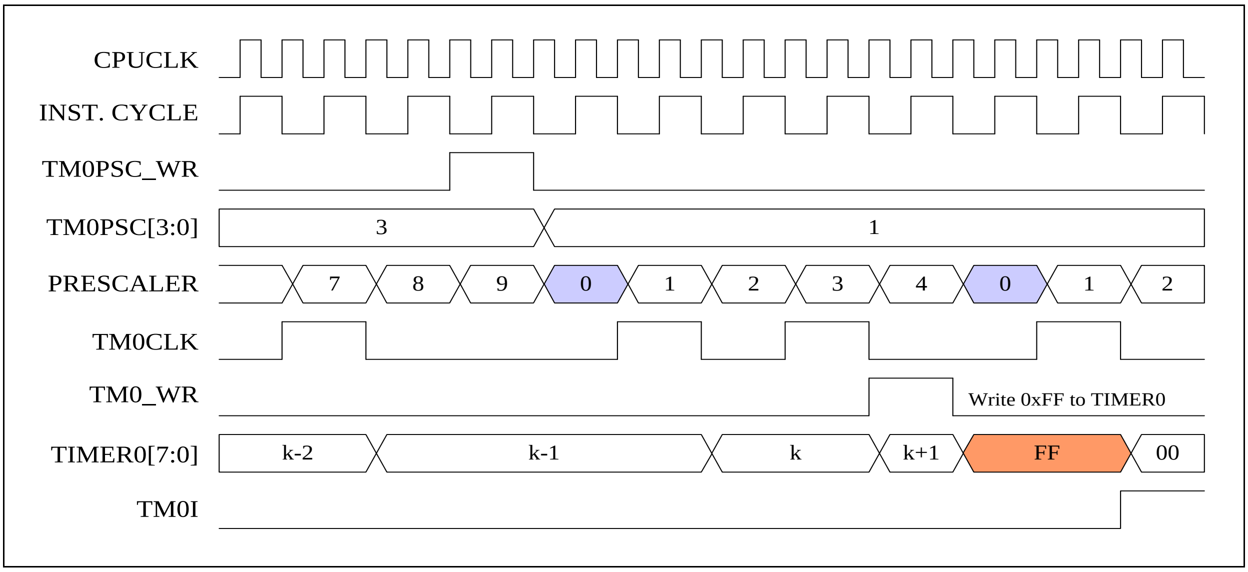
If the user program needs the MCU totally shuts down for power conservation in sleep mode, the following setting of control bits should be followed.

Mode	WDTE	WKTIE	Watchdog RC Oscillator
Normal Mode	0	0	Stop
	0	1	Run
	1	0	
	1	1	
Sleep Mode	0	0	Stop
	0	1	Run
	1	0	Stop
	1	1	Run

3.2 Timer0: 8-bit Timer/Counter with Pre-scale (PSC)

The Timer0 is an 8-bit wide register of F-Plane. It can be read or written as any other register 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 T0I input. The Timer0 increase rate is determined by "Timer0 Pre-Scale" (TM0PSC) register in R-Plane. The Timer0 can generate interrupt (TM0I) when it rolls over.

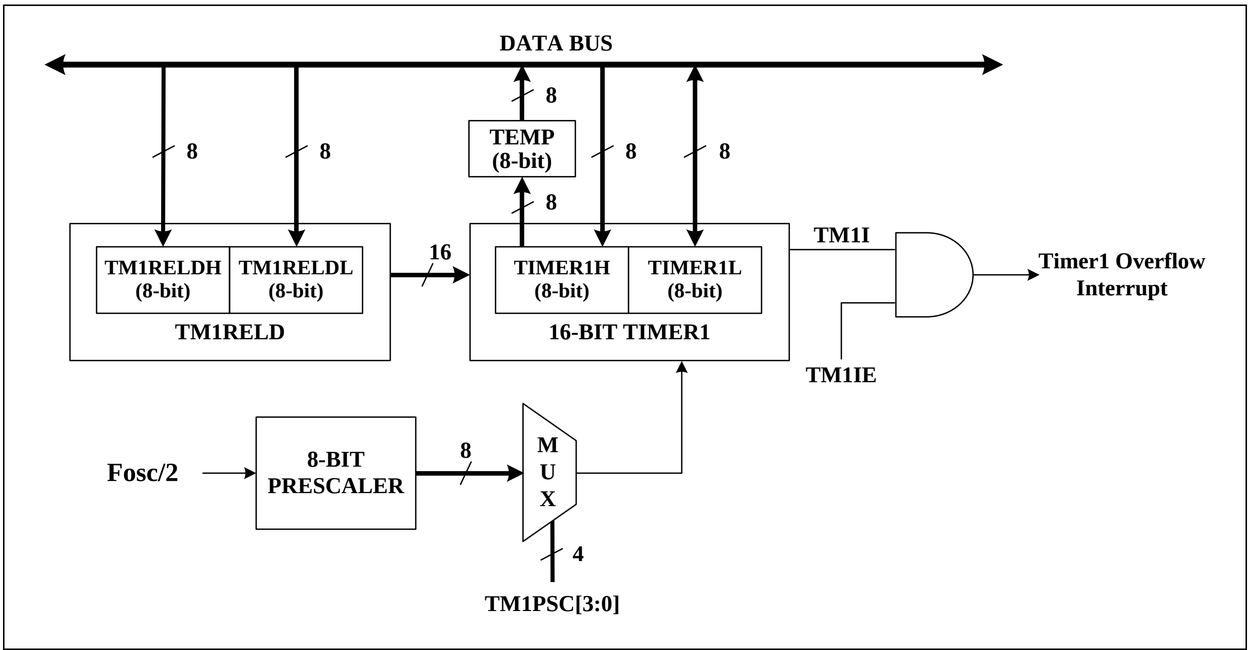


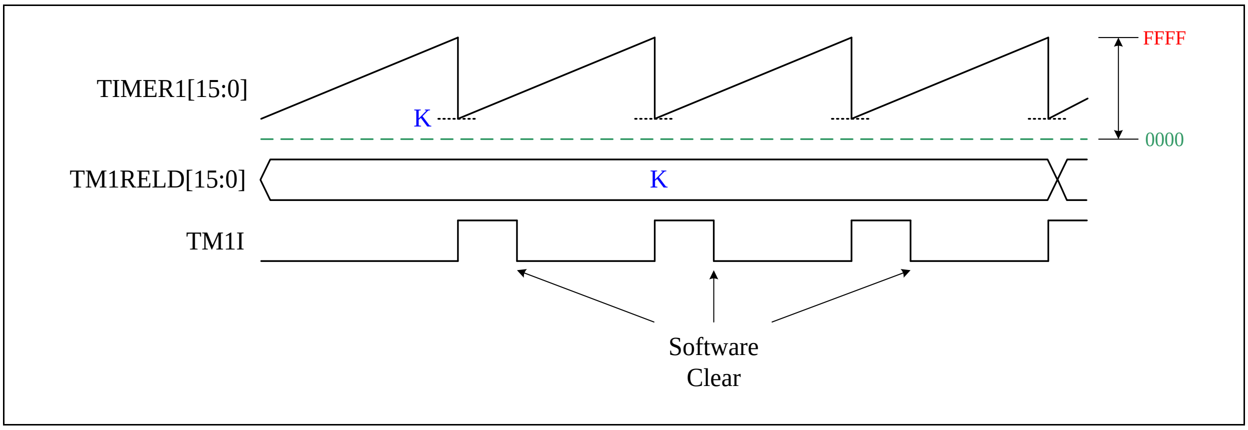


3.3 Timer1: 16-bit Timer with Pre-scale (PSC)

The Timer1 is a 16-bit wide register of F-Plane. It can be read or written as any other register of F-Plane. Besides, Timer1 increases itself periodically and automatically reloads a new "offset value" (TM1RELD) while it rolls over based on the pre-scaled instruction clock. The Timer1 increase rate is determined by "Timer1 Pre-Scale" (TM1PSC) register in R-Plane. The Timer1 can generate interrupt (TM1I) when it rolls over.

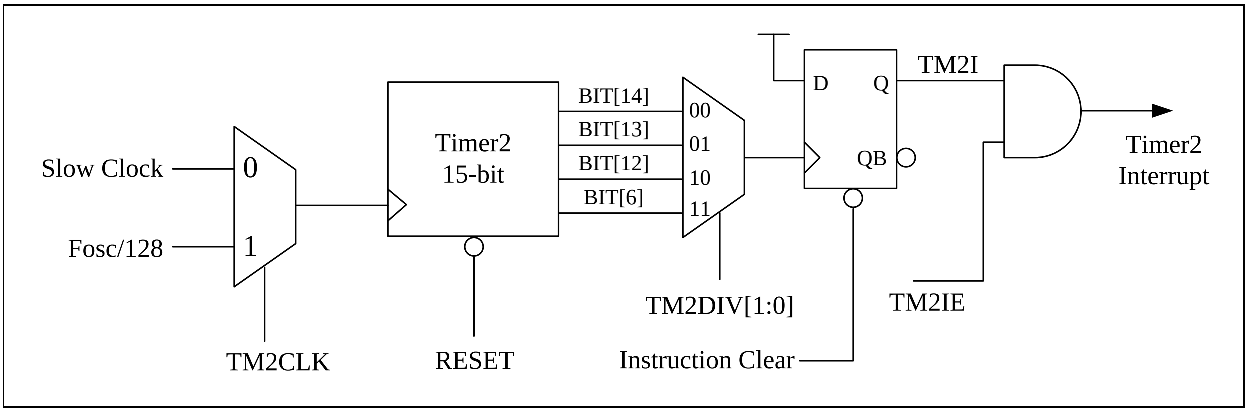
The Timer1 and TM1RELD are 16-bit registers that can be accessed via 8-bit data bus. The 16-bit register must be byte accessed using two read or write operations. There is a single 8-bit register for temporary storing of the high byte of Timer1 read. When the low byte of Timer1 register is read by the CPU, the high byte of Timer1 register is copied into the temporary register in the same clock cycle as the low byte is read. For Timer1 read, the low byte must be read before the high byte. Whatever high or low byte of a 16-bit register is written by the CPU, the value will be written into the register directly.





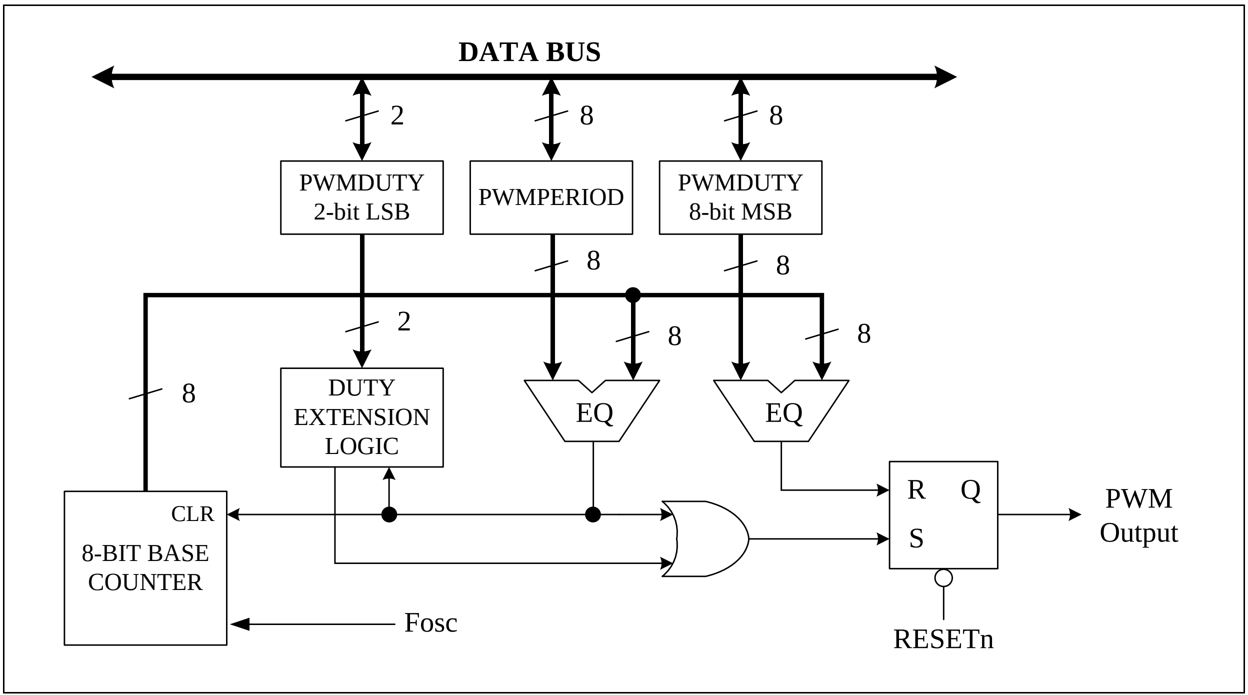
3.4 Timer2: 15-bit Timer with Pre-scale (PSC)

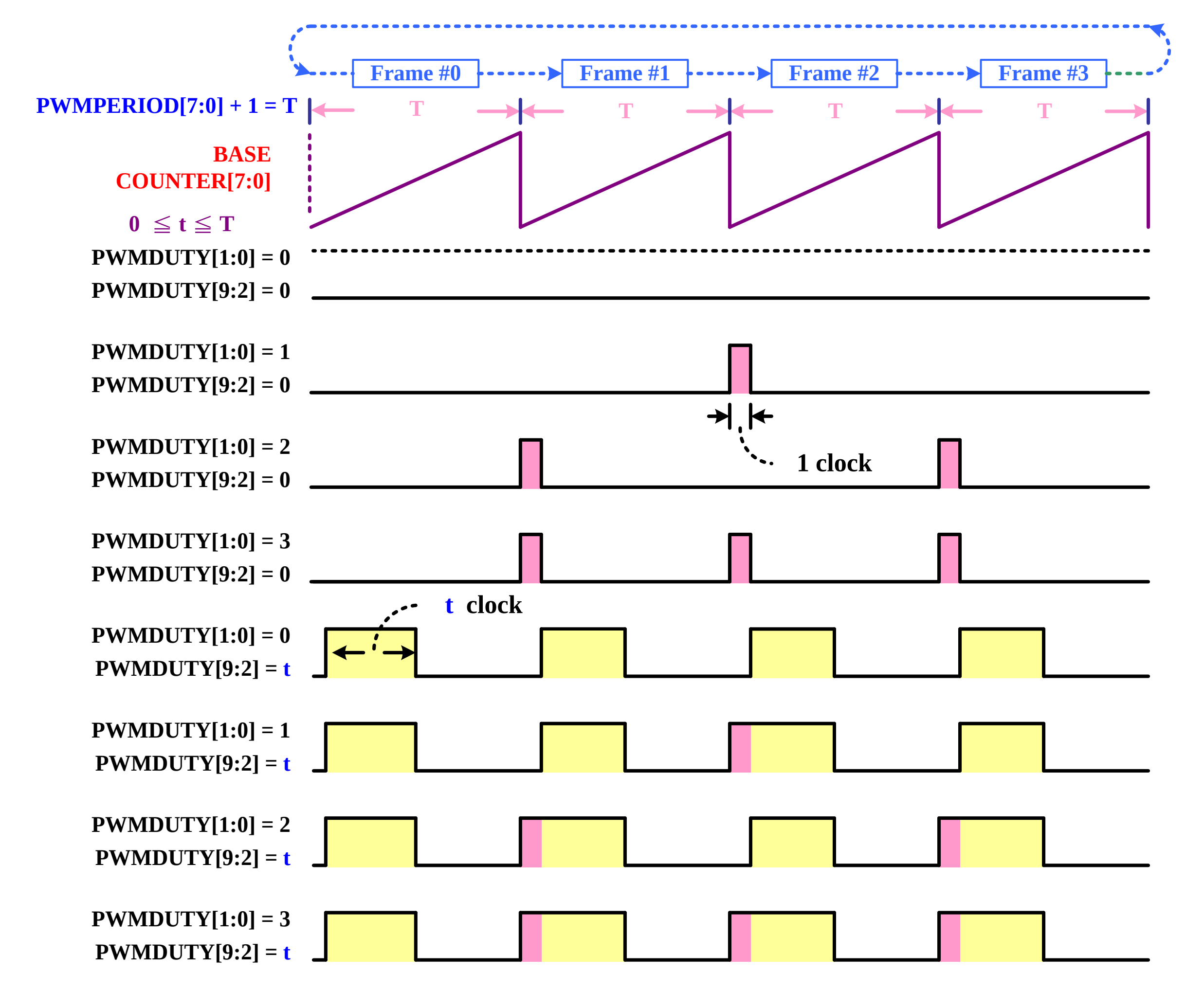
The Timer2 is a 15-bit counter and the clock sources are from either Fosc/128 or slow clock. It is used to generate time base interrupt and Timer2 counter block clock. The Timer2 content cannot be read by instructions. It generates interrupt flag (TM2I) with the clock divided by 32768, 16384, 8192, and 128, depends on TM2DIV register bits. Figure shows the block diagram of Timer2.



3.5 8+2 bits PWM

PWM0 and PWM1 have the same structure. The PWM supports period time and duty time adjustable. It also can generate fix frequency waveform with 1024 duty resolution based on system clock. A spread LSB technique allows PWM to run its frequency at "System Clock divided by 256" instead of "System Clock divided by 1024", which means the PWM is 4 times faster than normal. The advantage of higher PWM frequency is that the post RC filter can transform the PWM signal to more stable DC voltage level. The PWM output signal reset to low level whenever the 8-bit base counter matches the 8-bit MSB of PWM duty register (PWMDUTY). When the base counter rolls over, the 2-bit LSB of PWM duty register decides whether to set the PWM output signal high immediately or set it high after one clock cycle delay.

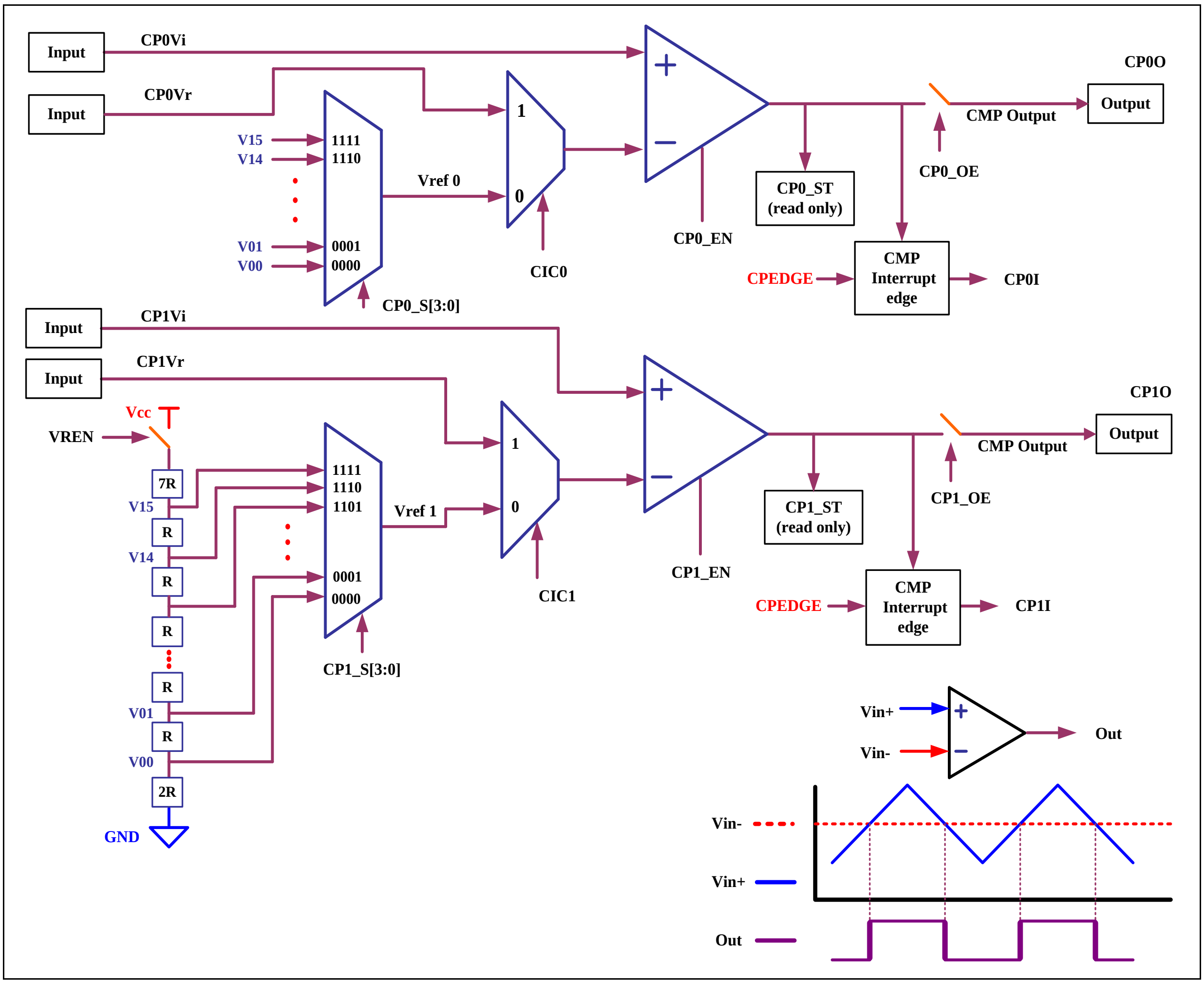



Frame #0
Frame #1
Frame #2
Frame #3
PWMPERIOD[7:0] + 1 = T
T
BASE COUNTER[7:0]
0 ≦ t ≦ T
PWMDUTY[1:0] = 0
PWMDUTY[9:2] = 0
PWMDUTY[1:0] = 1
PWMDUTY[9:2] = 0
1 clock
PWMDUTY[1:0] = 2
PWMDUTY[9:2] = 0
PWMDUTY[1:0] = 3
PWMDUTY[9:2] = 0
t clock
PWMDUTY[1:0] = 0
PWMDUTY[9:2] = t
PWMDUTY[1:0] = 1
PWMDUTY[9:2] = t
PWMDUTY[1:0] = 2
PWMDUTY[9:2] = t
PWMDUTY[1:0] = 3
PWMDUTY[9:2] = t

3.6 Analog Comparator

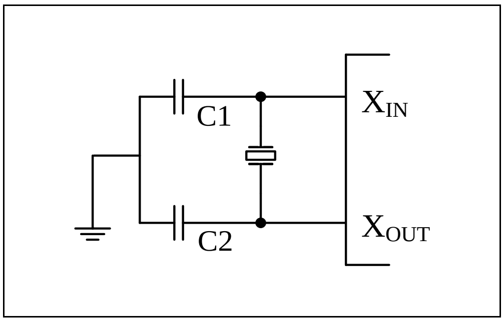
TM57ME20 has two analog comparators CMP0 and CMP1. They can be enabled by CPx_EN which is in F-Plane 12H Bit4~5. The analog comparators compare the input values on the positive pin CPxVi and negative pin CPxVr. When the voltage on positive pin is higher than the voltage on the negative pin, the analog comparators output CPxO is set. The output status CPx_ST can be read from F-Plane 14H Bit0~1, or output to pin by setting CPx_OE which is in F-Plane 12H Bit2~3. The analog comparator can generate interrupt (CPxI) when the output status changes. The user can select interrupt triggering on comparator output rise or fall. The analog comparators support internal reference voltage. To use internal reference voltage, enable VREN and clear CICx (default). The internal reference voltage provides the range of output voltage with 16 distinct levels. The range can be selected by CPx_S. A block diagram of the analog comparators is shown in below.

Note: A lower case "x" replaces the analog comparator number.

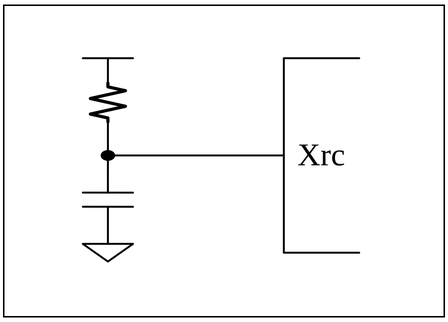


3.7 System Clock Oscillator

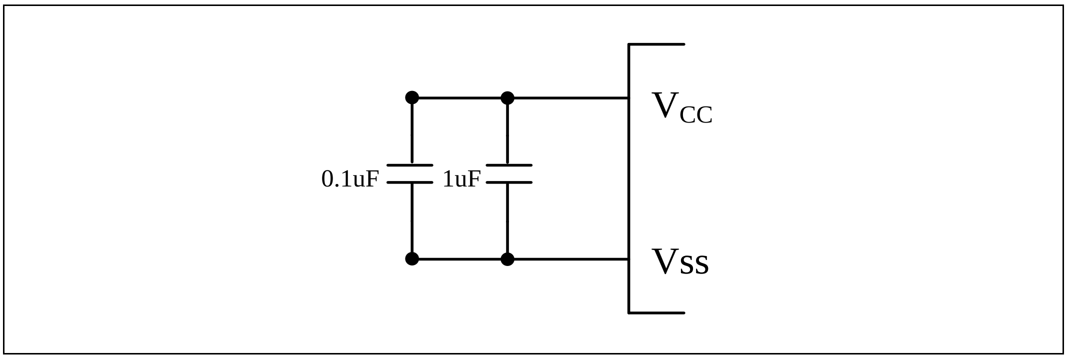
System clock can be operated in four different oscillation modes, which is selected by setting the CLKS in the SYSCFG register. In Slow/Fast Crystal mode, a crystal or ceramic resonator is connected to the Xin and Xout pins to establish oscillation. In external RC mode, the external resistor and capacitor determine the oscillation frequency. In the fast internal RC mode, the on-chip oscillator generates 4 MHz system clock. In this mode, PCB Layout may have strong effect on the stability of Internal Clock Oscillator. Since power noise degrades the performance of Internal Clock Oscillator, placing power supply bypass capacitors 1 uF and 0.1 uF very close to V_{CC}/V_{SS} pins improves the stability of clock and the overall system.



External Oscillator Circuit
(Crystal or Ceramic)



External RC Oscillator

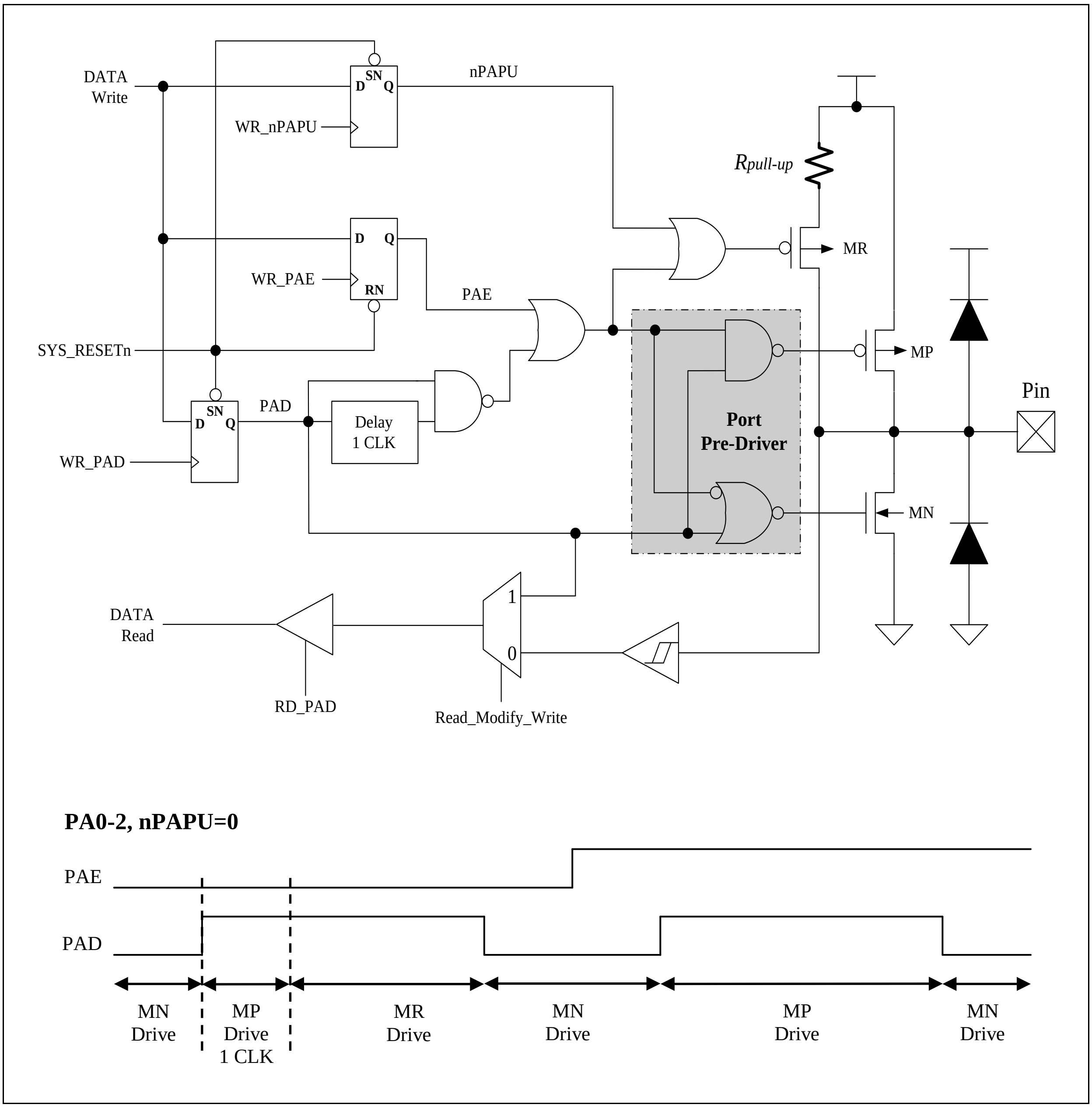


Internal RC Mode

4. I/O Port

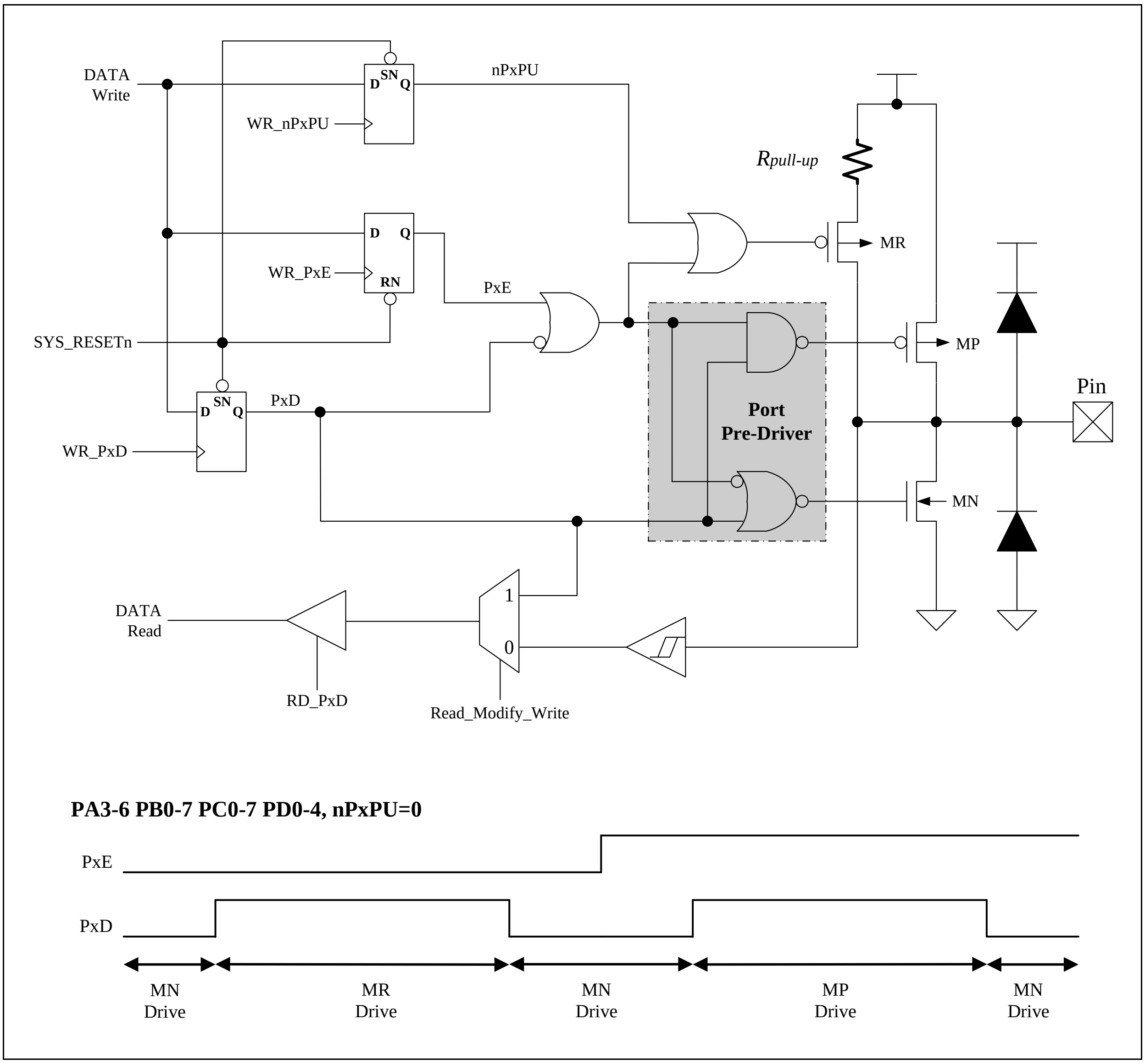
4.1 PA0-2

These pins can be used as Schmitt-trigger input, CMOS push-pull output or "pseudo-open-drain" output. The pull-up resistor is assignable to each pin by S/W setting. To use the pin in Schmitt-trigger input mode, S/W needs to set the PAE=0 and PAD=1. To use the pin in pseudo-open-drain mode, S/W sets the PAE=0.The benefit of pseudo-open-drain structure is that the output rise time can be much faster than pure open-drain structure. S/W sets PAE=1 to use the pin in CMOS push-pull output mode. Reading the pin data (PAD) has different meaning. In "Read-Modify-Write" instruction, CPU actually reads the output data register. In the other instructions, CPU reads the pin state. The so-called "Read-Modify-Write" instruction includes BSF, BCF and all instructions using F-Plane as destination.



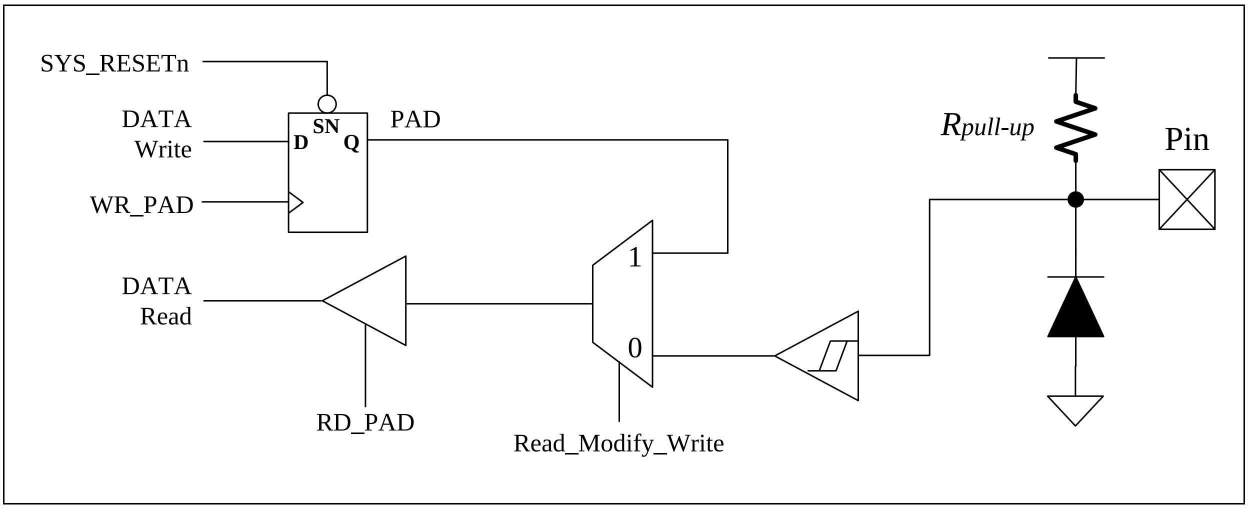
4.2 PA3-6, PB0-7, PC0-7, PD0-4

These pins are almost the same as PA0-2, except they do not support pseudo-open-drain mode. They can be used in pure open-drain mode, instead.



4.3 PA7

PA7 can be only used in Schmitt-trigger input mode. The pull-up resistor is always connected to this pin.



MEMORY MAP

F-Plane

Name	Address	R/W	Rst	Description
INDF	00.7~0	R/W	-	Not a physical register, addressing INDF actually point to the register whose address is contained in the FSR register
TIMER0	01.7~0	R/W	0	Timer0 content
PC	02.7~0	R/W	0	Programming Counter [7~0]
TO	03.4	R	0	WDT time out flag
PD	03.3	R	0	Sleep mode flag
ZFLAG	03.2	R/W	0	Zero flag
DCFLAG	03.1	R/W	0	Decimal Carry flag
CFLAG	03.0	R/W	0	Carry flag
GBIT1	04.7	R/W	0	General purpose bit
FSR	04.6~0	R/W	-	File Select Register, indirect address mode pointer
PAD7	05.7	R	-	PA7 pin state
PAD	05.6~0	R	-	Port A pin or "data register" state
		W	7F	Port A output data register
PBD	06.7~0	R	-	Port B pin or "data register" state
		W	FF	Port B output data register
PCD	07.7~0	R	-	Port C pin or "data register" state
		W	FF	Port C output data register
CP1IE	08.7	R/W	0	Comparator1 interrupt enable, 1=enable, 0=disable
CP0IE	08.6	R/W	0	Comparator0 interrupt enable, 1=enable, 0=disable
TM1IE	08.5	R/W	0	Timer1 interrupt enable, 1=enable, 0=disable
TM0IE	08.4	R/W	0	Timer0 interrupt enable, 1=enable, 0=disable
WKTIE	08.3	R/W	0	Wakeup Timer interrupt enable, 1=enable, 0=disable
TM2IE	08.2	R/W	0	Timer2 Interrupt enable, 1=enable, 0=disable
XINT1E	08.1	R/W	0	INT1 pin interrupt enable, 1=enable, 0=disable
XINT0E	08.0	R/W	0	INT0 pin interrupt enable, 1=enable, 0=disable
CP1I	09.7	R	-	Comparator1 interrupt event pending flag
		W	0	write 0: clear this flag; write 1: no action
CP0I	09.6	R	-	Comparator0 interrupt event pending flag
		W	0	write 0: clear this flag; write 1: no action
TM1I	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
TM0I	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
WKTI	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
TM2I	09.2	R	-	Timer2 interrupt event pending flag, set by H/W while Timer2 overflows
		W	0	write 0: clear this flag; write 1: no action
XINT1	09.1	R	-	INT1 interrupt event pending flag, set by H/W at INT1 pin's falling edge
		W	0	write 0: clear this flag; write 1: no action

Name	Address	R/W	Rst	Description
XINT0	09.0	R	-	INT0 interrupt event pending flag, set by H/W at INT0 pin's f/r edge
		W	0	write 0: clear this flag; write 1: no action
TIMER1	0a.7~0	R/W	0	Timer1 content 8-bit MSB
	0b.7~0	R/W	0	Timer1 content 8-bit LSB
PWM0DUTY	0c.7~0	R/W	0	PWM0 duty 8-bit MSB
	0d.7~6	R/W	0	PWM0 duty 2-bit LSB
PWM1DUTY	0e.7~0	R/W	0	PWM1 duty 8-bit MSB
	0f.7~6	R/W	0	PWM1 duty 2-bit LSB
PWMPERIOD	10.7~0	R/W	FF	PWM period
PDD	11.4~0	R	-	Port D pin or "data register" state
		W	1F	Port D output data register
CPEDGE	12.7	R/W	0	0: Comparator0/1 falling edge to trigger interrupt event 1: Comparator0/1 rising edge to trigger interrupt event
VREN	12.6	R/W	0	Internal reference voltage enable, 1: enable, 0: disable
CP1_EN	12.5	R/W	0	Comparator1 enable, 1: enable, 0: disable
CP0_EN	12.4	R/W	0	Comparator0 enable, 1: enable, 0: disable
CP1_OE	12.3	R/W	0	Comparator1 output enable, 1: enable, 0: disable
CP0_OE	12.2	R/W	0	Comparator0 output enable, 1: enable, 0: disable
CIC1	12.1	R/W	0	Comparator1 reference in selection 1: External reference voltage 0: Internal reference voltage
CIC0	12.0	R/W	0	Comparator0 reference in selection 1: External reference voltage 0: Internal reference voltage
CP1_S	13.7~4	R/W	0	Comparator1 internal reference voltage select 0000: V_{CC} * 2/24 0001: V_{CC} C * 3/24 ~ 1111: V_{CC} * 17/24
CP0_S	13.3~0	R/W	0	Comparator0 internal reference voltage select 0000: V_{CC} * 2/24 0001: V_{CC} C * 3/24 ~ 1111: V_{CC} * 17/24
CP1ST	14.1	R	-	Comparator1 output status
CP0ST	14.0	R	-	Comparator0 output status
SELSUB	15.7	R/W	0	Select slow clock as CPUCLK
STPFCK	15.6	R/W	0	Stop fast clock
SUBE	15.5	R/W	0	Slow clock enable
-	15.4~3	-	-	Reserved
CLRTM2	15.2	R/W	0	Write 1 to clear Timer2, auto cleared by H/W
STOPTM0	15.1	R/W	0	Stop Timer0 counting
-	15.0	-	-	Reserved
SRAM	20~7F	R/W	-	Internal RAM

R-Plane

Name	Address	R/W	Rst	Description
T0IEDGE	02.5	W	0	0: T0I rising edge to increase Timer0/PSC count 1: T0I falling edge to increase Timer0/PSC count
SELT0I	02.4	W	0	0: Timer0/PSC clock source is "Instruction Cycle" 1: Timer0/PSC clock source is T0I pin
TM0PSC	02.3~0	W	0	0000: Timer0 input clock divided by 1 0001: Timer0 input clock divided by 2 ~ 0111: Timer0 input clock divided by 128 1000: Timer0 input clock divided by 256
PWRDOWN	03	W	-	Write this register to enter Power-Down Mode
CLRWDT	04	W	-	Write this register to clear WDT/WKT
PAE	05.6~3	W	0	0: the pin is open-drain output or Schmitt-trigger input 1: the pin is CMOS push-pull output
	05.2~0	W	0	0: the pin is pseudo-open-drain output or Schmitt-trigger input 1: the pin is CMOS push-pull output
PBE	06.7~0	W	0	0: the pin is open-drain output or Schmitt-trigger input 1: the pin is CMOS push-pull output
PCE	07.7~0	W	0	0: the pin is open-drain output or Schmitt-trigger input 1: the pin is CMOS push-pull output
nPAPU	08.6~0	W	7F	0: the pin pull up resistor is enabled, except a. the pin's output data register (PAD) is 0 b. the pin's CMOS push-pull mode is chosen (PAE=1) c. the pin is working for Crystal or external RC oscillation 1: the pin pull up resistor is disabled
nPBPU	09.7~0	W	FF	0: the pin pull up resistor is enabled 1: the pin pull up resistor is disabled
nPCPU	0a.7~0	W	FF	0: the pin pull up resistor is enabled 1: the pin pull up resistor is disabled
HWAUTO	0b.7	W	0	Auto store/restore STATUS and W before/after interrupt routine
PWM0E	0b.6	W	0	0: disable PWM0 output to pin 1: enable PWM0 output to pin
PWM1E	0b.5	W	0	0: disable PWM1 output to pin 1: enable PWM1 output to pin
INT0EDGE	0b.4	W	0	0: INT0 pin falling edge to trigger interrupt event 1: INT0 pin rising edge to trigger interrupt event
CLK2PIN	0b.3	W	0	0: No Instruction Clock output to PA3 pin 1: Instruction Clock output to PA3 pin for external/internal RC mode
-	0b.2	-	-	Reserved

Name	Address	R/W	Rst	Description
WKTPSC	0b.1~0	W	11	WDT/WKT pre-scale option or SIRC frequency select WDT/WKT pre-scale option 00: WDT/WKT period is 14 ms, @5V; 15 ms, @3V 01: WDT/WKT period is 28 ms, @5V; 30 ms, @3V 10: WDT/WKT period is 57 ms, @5V; 61 ms, @3V 11: WDT/WKT period is 111 ms, @5V; 121 ms, @3V SIRC frequency select 00: SIRC Frequency is 138 KHz, @5V; 119 KHz, @3V 01: SIRC Frequency is 35 KHz, @5V; 30 KHz, @3V 10: SIRC Frequency is 8.5 KHz, @5V; 7.5 KHz, @3V 11: SIRC Frequency is 2.1 KHz, @5V; 1.9 KHz, @3V
TM1PSC	0c.3~0	W	0	0000: Timer1 input clock divided by 1 0001: Timer1 input clock divided by 2 ~ 0111: Timer1 input clock divided by 128 1000: Timer1 input clock divided by 256
TM1RELD	0d.7~0	W	0	Timer1 reloads offset value 8-bit MSB while it rolls over
	0e.7~0	W	0	Timer1 reloads offset value 8-bit LSB while it rolls over
PDE	10.4~0	W	0	0: the pin is open-drain output or Schmitt-trigger input 1: the pin is CMOS push-pull output
nPDPU	11.4~0	W	1F	0: the pin pull up resistor is enabled 1: the pin pull up resistor is disabled
PIE	12.3	W	1	PC4/CP1vr input type, 0: analog input, 1: digital input
	12.2	W	1	PC3/CP1vi input type, 0: analog input, 1: digital input
	12.1	W	1	PC1/CP0vr input type, 0: analog input, 1: digital input
	12.0	W	1	PC0/CP0vi input type, 0: analog input, 1: digital input
IVCTRL	13.1~0	W	0	Built-in regulator control in stop mode 00: V_{CC} > 4.5V 01: 4.5V > V_{CC} > 3.6V 10: 3.6V > V_{CC}
PBWKUP	14.7~0	W	0	Enable PB7~PB0 pin low level wake up
TM2CLK	15.4	W	0	Timer2 clock source 0: slow clock 1: CPUCLK/128
TM2DIV	15.3~2	W	0	Timer2 interrupt is Timer2 divide by – 0: 32768, 1: 16384, 2: 8192, 3: 128
SUBTYP	15.1~0	W	0	Slow clock type 0: SXT, 1: SIRC, 2: XRC

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
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					
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	00 0000 00rr rrrr	1	-	Move W to "r"
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		00 0000 0000 0100	1	TO, PD	Clear and enable 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		00 0000 0000 0011	1	TO, PD	Go into standby mode, Clock oscillation stops
XORLW	k	01 1111 kkkk kkkk	1	Z	XOR Literal "k" with W

ADDLW — Add Literal "k" and W

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

ADDWF — Add W and "f"

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

ANDLW — Logical AND Literal "k" with W

Syntax	ANDLW k
Operands	k : 00h ~ FFh
Operation	(W) ← (W) AND 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	(destination) ← (W) AND (f)
Status Affected	Z
OP-Code	00 0101 dfff ffff
Description	AND the W register with register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.
Cycle	1
Example	ANDWF FSR, 1 — B : W = 0x17, FSR = 0xC2 A : W = 0x17, FSR = 0x02

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

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

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

BTFSS	Test "b" bit of "f", skip if set(1)	
Syntax	BTFSS f [,b]	
Operands	f : 00h ~ 3Fh, b : 0 ~ 7	
Operation	Skip next instruction if (f.b) = 1	
Status Affected	-	
OP-Code	01 011b bbff ffff	
Description	If bit 'b' in register 'f' is 0, then the next instruction is executed. If bit 'b' in register 'f' is 1, then the next instruction is discarded, and a NOP is executed instead, making this a 2nd cycle instruction.	
Cycle	1 or 2	
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 : 000h ~ 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 <11:0>. 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 Z bit is set.	
Cycle	1	
Example	CLRW	B : W = 0x5A A : W = 0x00, Z = 1

CLRWDT	**Clear Watchdog Timer**	
Syntax	CLRWDT	
Operands	-	
Operation	WDT/WKT Timer ← 00h	
Status Affected	TO, PD	
OP-Code	00 0000 0000 0100	
Description	CLRWDT instruction clears the Watchdog/Wakeup Timer	
Cycle	1	
Example	CLRWDT	B : WDT counter = ? A : WDT counter = 0x00

COMF — Complement "f"

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

DECF — Decrement "f"

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

DECFSZ — Decrement "f", Skip if 0

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

GOTO — Unconditional Branch

Syntax	GOTO k	
Operands	k : 000h ~ FFFh	
Operation	PC.11~0 ← 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 <11:0>. GOTO is a two-cycle instruction.	
Cycle	2	
Example	LABEL1 GOTO SUB1	B : PC = LABEL1 A : PC = SUB1

INCF — Increment "f"

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

INCFSZ — Increment "f", Skip if 0

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

IORLW — Inclusive OR Literal with W

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

IORWF — Inclusive OR W with "f"

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

MOVFW — Move "f" to W

Syntax	MOVFW f	
Operands	f : 00h ~ 7Fh	
Operation	(W) ← (f)	
Status Affected	-	
OP-Code	00 1000 0fff ffff	
Description	The contents of register 'f' are moved to W register.	
Cycle	1	
Example	MOVFW FSR	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	00 0000 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

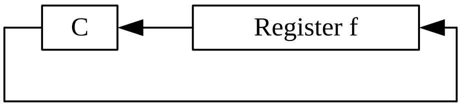
NOP	**No Operation**	
Syntax	NOP	
Operands	-	
Operation	No Operation	
Status Affected	-	
OP-Code	00 0000 0000 0000	
Description	No Operation	
Cycle	1	
Example	NOP	-

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	RET	A : PC = TOS

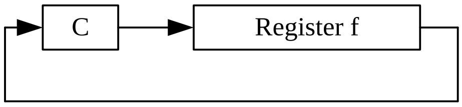
RETI	**Return from Interrupt**	
Syntax	RETI	
Operands	-	
Operation	PC ← TOS, GIE ← 1	
Status Affected	-	
OP-Code	00 0000 0110 0000	
Description	Return from Interrupt. Stack is POPed and Top-of-Stack (TOS) is loaded in to the PC. Interrupts are enabled. This is a two-cycle instruction.	
Cycle	2	
Example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

SLEEP — Go into standby mode, Clock oscillation stops

Syntax	SLEEP
Operands	-
Operation	-
Status Affected	TO, PD
OP-Code	00 0000 0000 0011
Description	Go into SLEEP mode with the oscillator stops.
Cycle	1
Example	SLEEP — -

SUBWF — Subtract W from "f"

Syntax	SUBWF f [,d]	
Operands	f : 00h ~ 7Fh, d : 0, 1	
Operation	(destination) ← (f) – (W)	
Status Affected	C, DC, Z	
OP-Code	00 0010 dfff ffff	
Description	Subtract (2's complement method) W register from register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.	
Cycle	1	
Example	SUBWF REG1, 1	B : REG1 = 0x03, W = 0x02, C = ?, Z = ? A : REG1 = 0x01, W = 0x02, C = 1, Z = 0
	SUBWF REG1, 1	B : REG1 = 0x02, W = 0x02, C = ?, Z = ? A : REG1 = 0x00, W = 0x02, C = 1, Z = 1
	SUBWF REG1, 1	B : REG1 = 0x01, W = 0x02, C = ?, Z = ? A : REG1 = 0xFF, W = 0x02, C = 0, Z = 0

SWAPF — Swap Nibbles in "f"

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

TESTZ — Test if "f" is zero

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

XORLW Exclusive OR Literal with W

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

XORWF Exclusive OR W with "f"

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

ELECTRICAL CHARACTERISTICS

1. Absolute Maximum Ratings (T_A = 25℃)

Parameter	Rating	Unit
Supply voltage	V_{SS} - 0.3 to V_{SS} + 6.5	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 +85	°C
Storage temperature	-65 to +150	

2. DC Characteristics (T_A = 25℃, V_{CC} = 2.0V to 5.5V)

Parameter	Sym	Conditions		Min	Typ	Max	Unit
Input High Voltage	V_{IH}	All Input, except PA7	V_{CC} = 5V	0.7V_{CC}	–	–	V
			V_{CC} = 3V	0.7V_{CC}	–	–	V
		PA7	V_{CC} = 5V	0.8V_{CC}	–	–	V
			V_{CC} = 3V	0.8V_{CC}	–	–	V
Input Low Voltage	V_{IL}	All Input, except PA7	V_{CC} = 5V	–	–	0.2V_{CC}	V
			V_{CC} = 3V	–	–	0.2V_{CC}	V
		PA7	V_{CC} = 5V	–	–	0.2V_{CC}	V
			V_{CC} = 3V	–	–	0.2V_{CC}	V
Output High Voltage	V_{OH}	All Output	V_{CC} = 5V, I_{OH} = 8 mA	4.6	–	–	V
			V_{CC} = 3V, I_{OH} = 4 mA	2.6	–	–	
Output Low Voltage	V_{OL}	All Output	V_{CC} = 5V, I_{OL} = 17 mA	–	–	0.5	V
			V_{CC} = 3V, I_{OL} = 7 mA	–	–	0.3	
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} = 0V	–	–	-1	μA
Power Supply Current	I_{DD}	Run 10 MHz, No Load	V_{CC} = 5.0V	–	3.1	–	mA
		Run 4 MHz, No Load	V_{CC} = 3.0V	–	1	–	
		Run 32 KHz, IVC disable	V_{CC} = 3.0V	–	27	–	μA
		Run 32 KHz, IVC enable	V_{CC} = 3.0V	–	100	–	
		Stop mode, IVC disable	V_{CC} = 5V	–	–	0.1	μA
			V_{CC} = 3V			0.1	
		Stop mode, IVC enable	V_{CC} = 5V	–	–	190	
			V_{CC} = 3V			80	
System Clock Frequency	F_{OSC}	F_{OSC}	V_{CC} = 3V	–	–	12	MHz
			V_{CC} = 2.2V			8	
			V_{CC} = 2.1V			4	
LVR Reference Voltage		V_{LVR}		–	2.2	–	V
				–	3.2	–	V
LVR Hysteresis Voltage		V_{HYST}		–	±0.1	–	V
Low Voltage Detection time		t_{LVR}		100	–	–	μs
Pull-Up Resistor	R_P		V_{CC} = 2.2V	–	150	–	kΩ
			V_{CC} = 2.1V		325		
		V_{IN} = 0 V PA7	V_{CC} = 5V	–	96	–	kΩ
			V_{CC} = 3V		92		

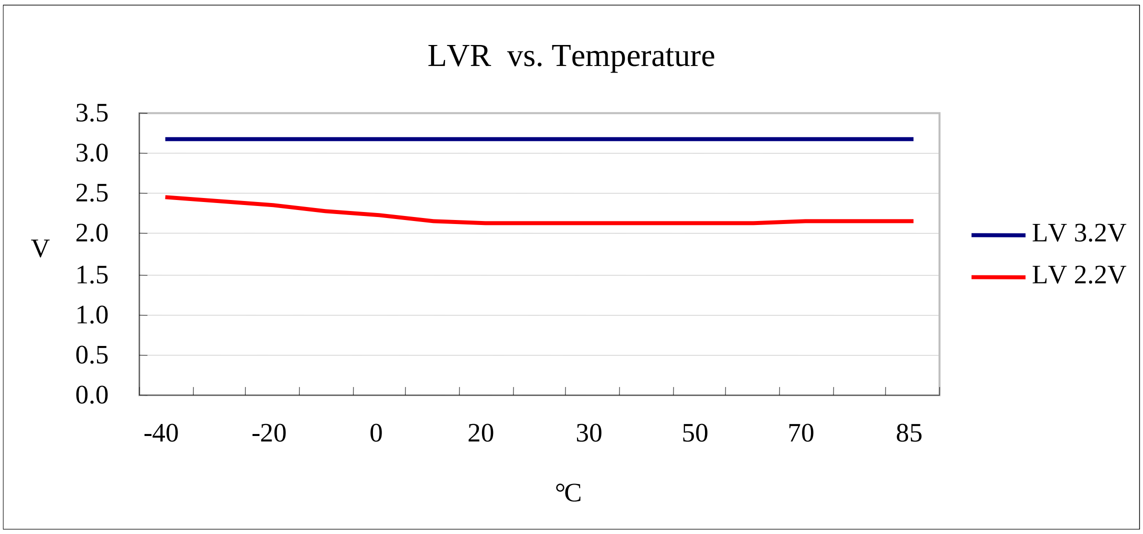
3. Clock Timing (T_A = -40°C to +85°C)

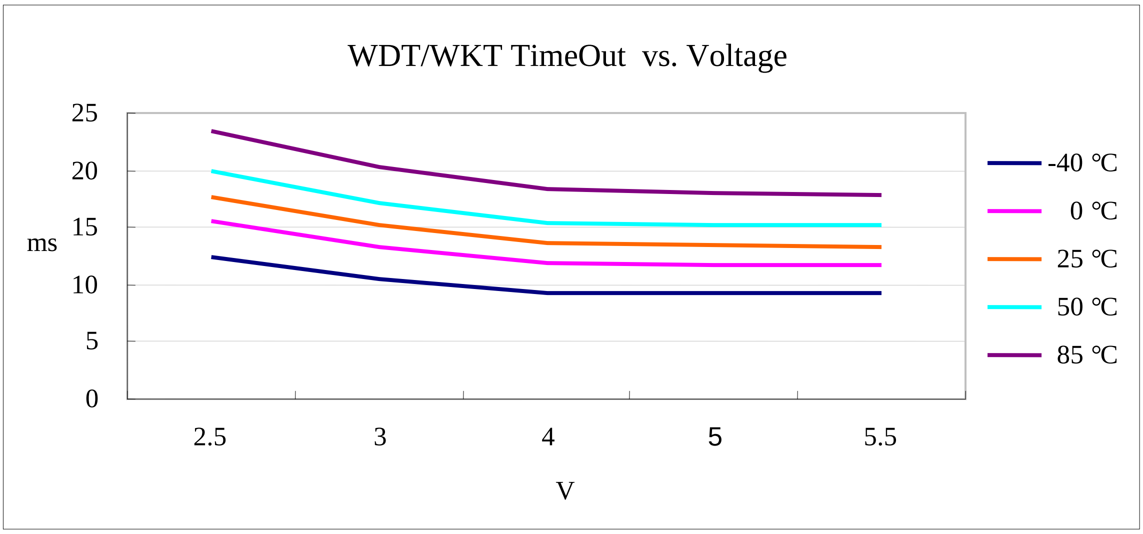
Parameter	Condition			Min	Typ	Max	Unit
External RC Frequency	V_{CC} = 3V	R = 4.7K	C = 20 pF	–	1.94	–	MHz
		R = 10K	C = 100 pF	–	0.69	–	
		R = 100K	C = 300 pF	–	0.04	–	
	V_{CC} = 5V	R = 4.7K	C = 20 pF	–	2.93	–	
		R = 10K	C = 100 pF	–	0.64	–	
		R = 100K	C = 300 pF	–	0.03	–	
Fast Internal RC Frequency	25°C, V_{CC} = 2.5 ~ 5.5V			3.85	4	4.15	
	-40°C ~ 85°C, V_{CC} = 2.5 ~ 5.5V			3.7	4	4.3	

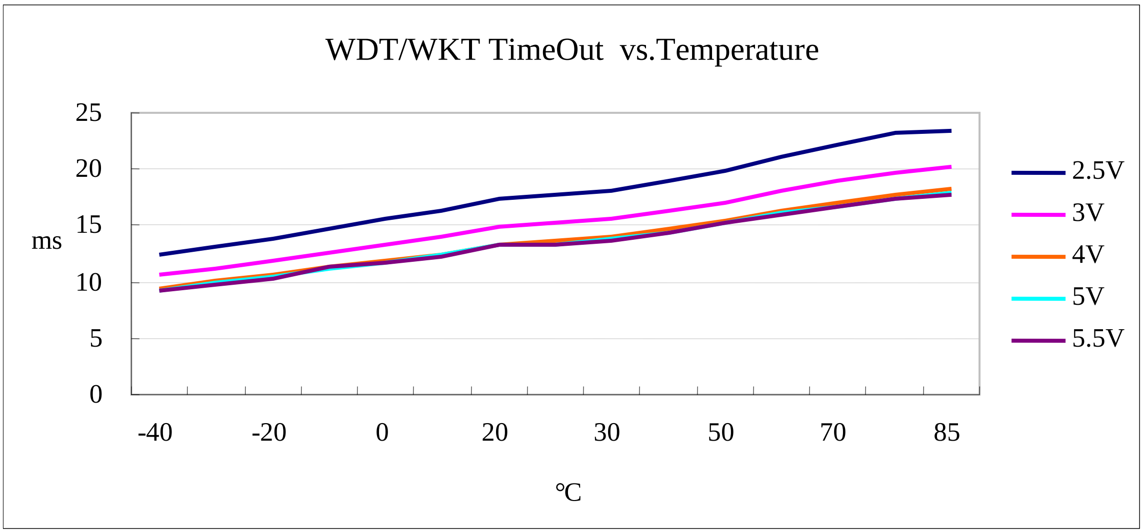
4. Reset Timing Characteristics (T_A = -40°C to +85°C, V_{CC} = 2.0V to 5.5V)

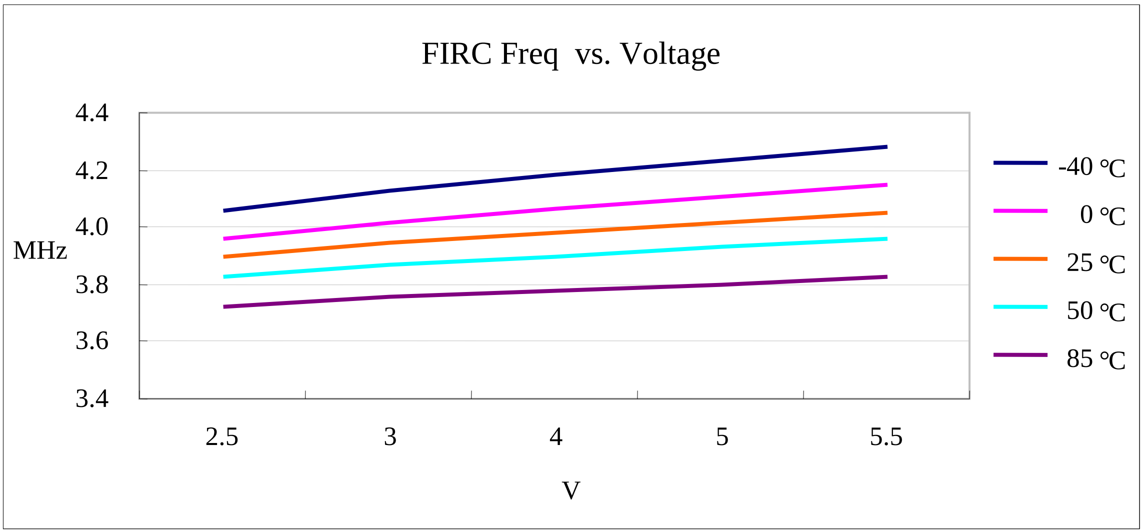
Parameter	Conditions	Min	Typ	Max	Unit
RESET Input Low width	Input V_{CC} = 5V ± 10 %	3	–	–	μs
WDT wakeup time	V_{CC} = 5V, WKTPSC = 11	–	114	–	ms
	V_{CC} = 3V, WKTPSC = 11	–	123	–	
CPU start up time	V_{CC} = 5V	–	3.5	–	ms

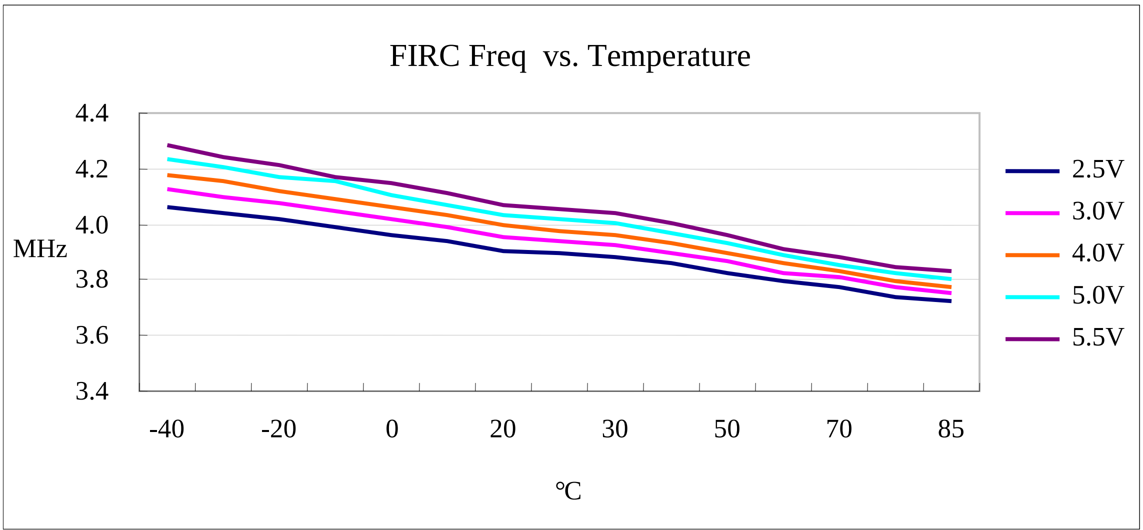
5. Characteristic Graphs






WDT/WKT TimeOut vs.Temperature
25
20
15
10
5
0
ms
-40
-20
0
20
30
50
70
85
℃
2.5V
3V
4V
5V
5.5V


FIRC Freq vs. Voltage
4.4
4.2
4.0
3.8
3.6
3.4
MHz
2.5
3
4
5
5.5
V
-40 ℃
0 ℃
25 ℃
50 ℃
85 ℃

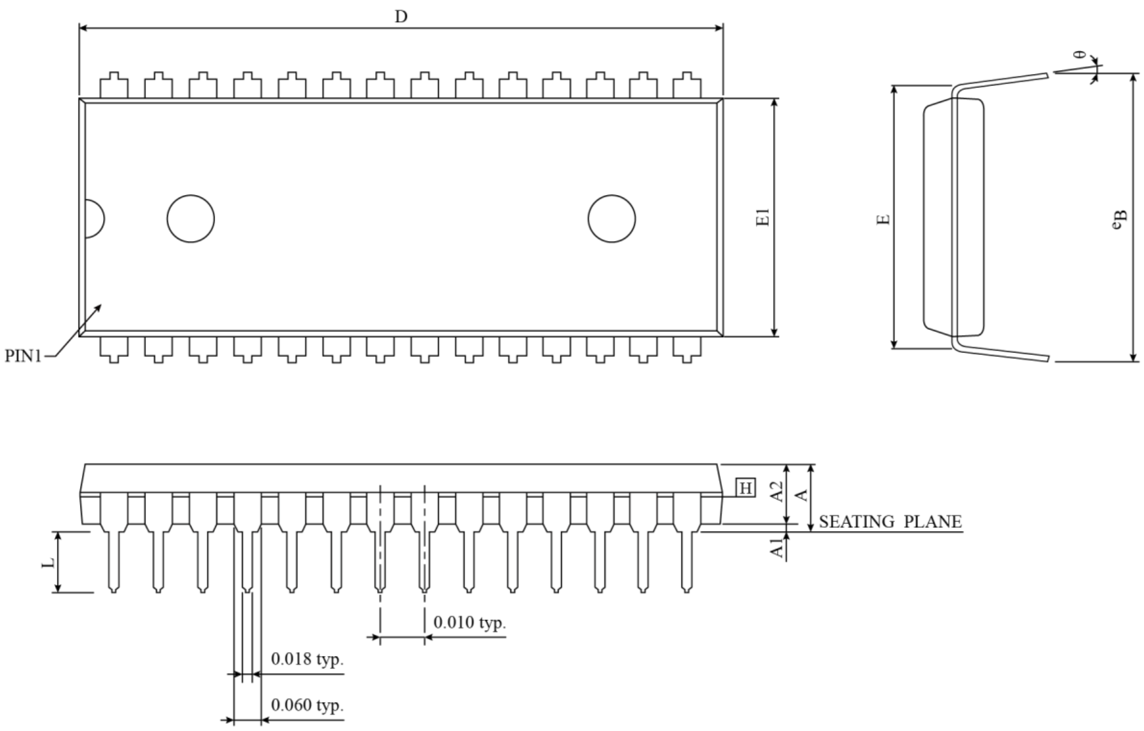

FIRC Freq vs. Temperature
4.4
4.2
4.0
3.8
3.6
3.4
MHz
-40
-20
0
20
30
50
70
85
℃
2.5V
3.0V
4.0V
5.0V
5.5V

PACKAGING INFORMATION

The ordering information:

Ordering number	Package
TM57ME20-MTP	Wafer / Dice blank chip
TM57ME20-COD	Wafer / Dice with code
TM57ME20-MTP-08	DIP 28-pin (600 mil)
TM57ME20-MTP-23	SOP 28-pin (300 mil)
TM57ME20-MTP-09	DIP 32-pin (600 mil)
TM57ME20-MTP-24	SOP 32-pin (300 mil)

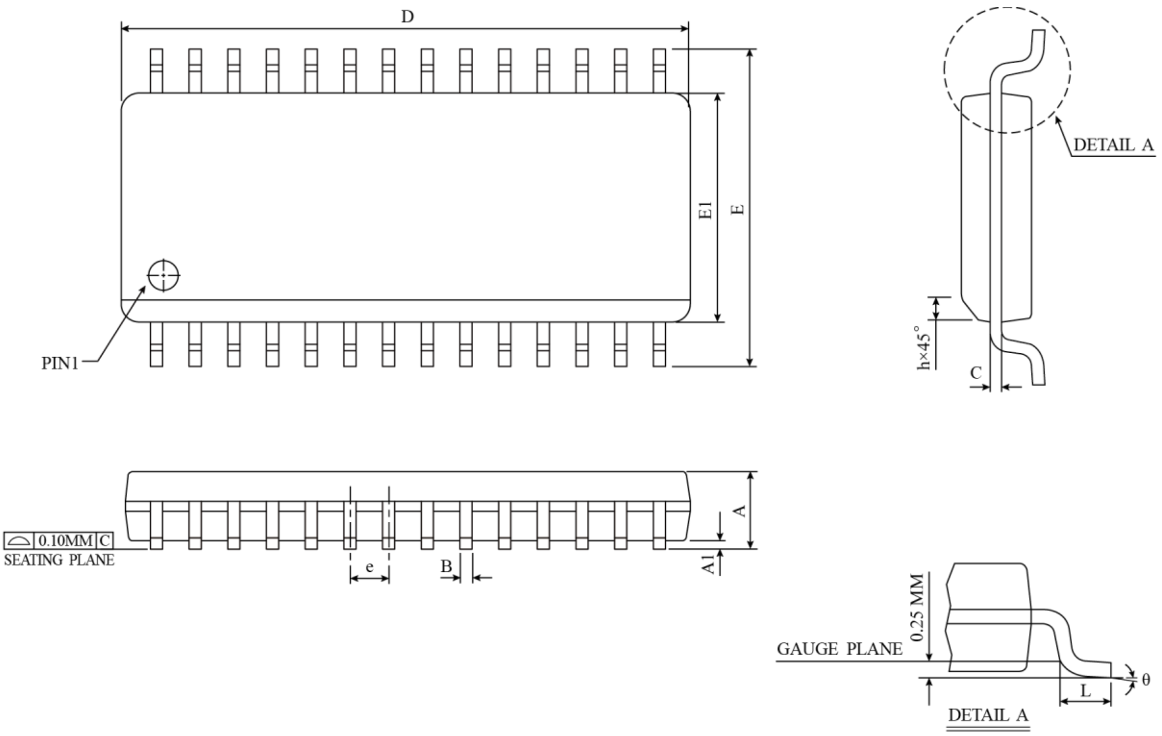
28-DIP (600mil) Package Dimension



SYMBOL	DIMENSION IN MM		DIMENSION IN INCH	
	MIN	MAX	MIN	MAX
A	-	5.588	-	0.220
A1	0.381	-	0.015	-
A2	3.810	4.064	0.150	0.160
D	36.957	37.338	1.455	1.470
E	15.240 BSC		0.600 BSC	
E1	13.716	13.970	0.540	0.550
L	2.921	5.080	0.115	0.200
^{e}B	16.002	17.018	0.630	0.670
θ	0°	15°	0°	15°
JEDEC	MS-011 (AB)			

NOTES：E1 DOES NOT INCLUDE MOLD FLASH.

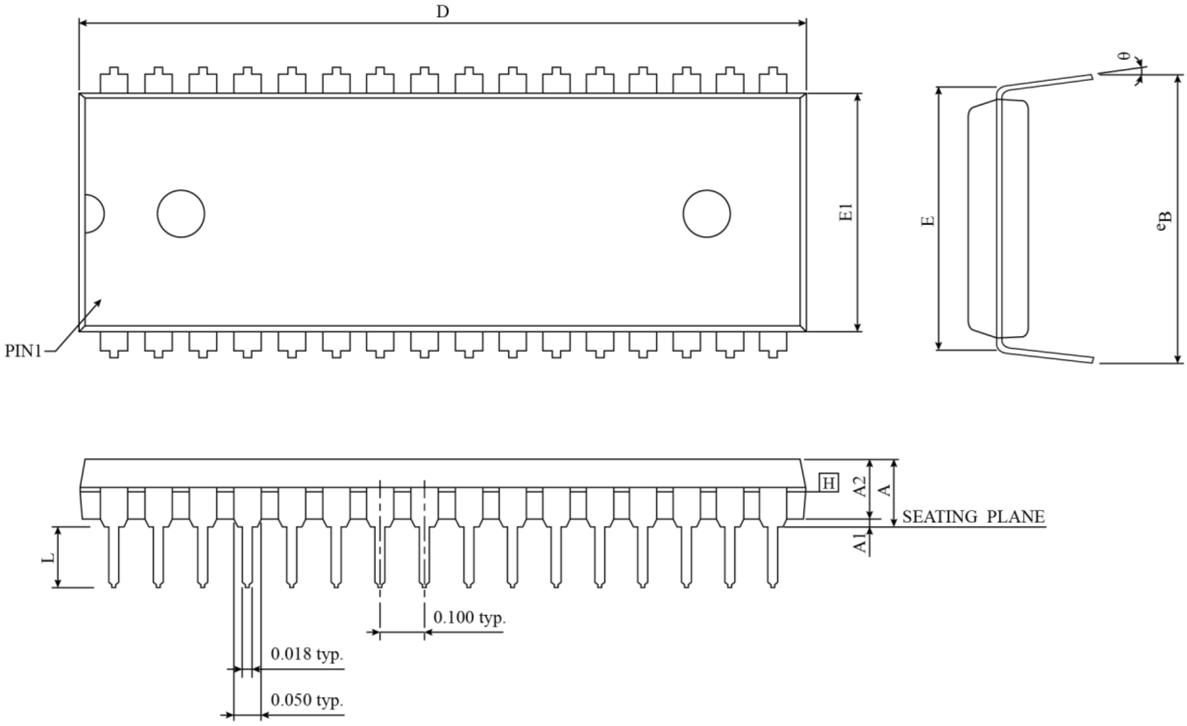
28-SOP Package Dimension



SYMBOL	DIMENSION IN MM		DIMENSION IN INCH	
	MIN	MAX	MIN	MAX
A	2.35	2.65	0.0926	0.1043
A1	0.10	0.30	0.0040	0.0118
B	0.33	0.51	0.013	0.020
C	0.23	0.32	0.0091	0.0125
D	17.70	18.10	0.6969	0.7125
E	10.00	10.65	0.394	0.491
E1	7.40	7.60	0.2914	0.2992
e	1.27 BSC		0.050 BSC	
h	0.25	0.75	0.010	0.029
L	0.40	1.27	0.016	0.050
θ	0°	8°	0°	8°
JEDEC	MS-013 (AE)			

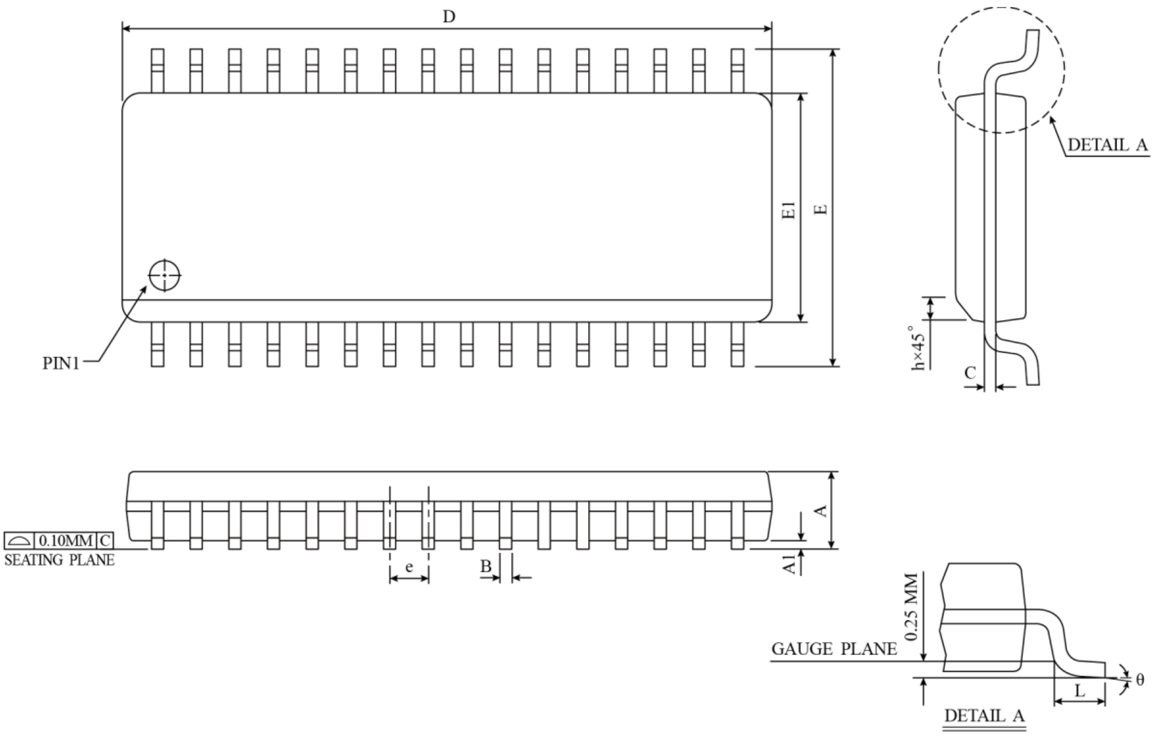
A * NOTES : DIMENSION " D " DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS.
MOLD FLASH, PROTRUSIONS AND GATE BURRS SHALL
NOT EXCEED 0.15 MM (0.006 INCH) PER SIDE.

32-DIP (600mil) Package Dimension



SYMBOL	DIMENSION IN MM		DIMENSION IN INCH	
	MIN	MAX	MIN	MAX
A	-	5.588	-	0.220
A1	0.381	-	0.015	-
A2	3.810	4.064	0.150	0.160
D	41.783	42.164	1.645	1.660
E	15.240 BSC		0.600 BSC	
E1	13.716	13.970	0.540	0.550
L	2.921	5.080	0.115	0.200
^{e}B	16.002	17.018	0.630	0.670
θ	0°	15°	0°	15°
JEDEC	MO-015 (AP)			

32-SOP (300mil) Package Dimension



SYMBOL	DIMENSION IN MM		DIMENSION IN INCH	
	MIN	MAX	MIN	MAX
A	2.35	2.65	0.0926	0.1043
A1	0.10	0.30	0.0040	0.0118
B	0.33	0.51	0.013	0.020
C	0.23	0.32	0.0091	0.0125
D	20.32	20.73	0.800	0.816
E	10.00	10.65	0.394	0.491
E1	7.40	7.60	0.2914	0.2992
e	1.27 BSC		0.050 BSC	
h	0.25	0.75	0.010	0.029
L	0.40	1.27	0.016	0.050
θ	0°	8°	0°	8°

A * 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.