

十速

TM57PE40

规格书

Rev V1.3

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.

修改记录

版本	日期	描述
V1.0	Apr, 2013	新颁
V1.1	Jun, 2013	1. 增加支持的 EV board 于 ICE 2. 增加引脚摘要 3. 修改订购须知
V1.2	Aug, 2013	修改订购须知
V1.3	Sep, 2015	修改操作电压范围

CONTENTS

修改记录.....	2
基本功能.....	5
系统结构图.....	8
引脚分配图.....	9
管脚描述.....	10
管脚摘要.....	11
功能描述.....	12
1. CPU 的特色.....	12
1.1 时钟配置和指令周期.....	12
1.2 寻址模式.....	13
1.3 程序计数器（PC）和堆栈.....	14
1.4 ALU 和工作寄存器（W）.....	14
1.5 状态寄存器.....	15
1.6 中断.....	16
2. 芯片工作模式.....	17
2.1 复位.....	17
2.2 系统配置寄存器(SYSCFG).....	18
2.3 可分区编程.....	19
2.4 双系统时钟.....	20
2.5 双系统时钟的模式转换.....	22
3. 外围功能模块.....	25
3.1 看门狗（WDT）/唤醒（WKT）定时器.....	25
3.2 Timer0.....	26
3.3 Timer1.....	28
3.4 Timer0 和 Timer1 用于脉宽及周期捕获.....	30
3.5 Timer2: 15 位定时器.....	32
3.6 PWM0: 8 位 PWM.....	33
3.7 PWMA: (8+2) 位 PWM.....	35
3.8 模拟比较器.....	37
3.9 触摸按键.....	38
3.10 系统时钟振荡器.....	41
4 I/O 端口.....	42
4.1 PA0-2.....	42

4.2 PA3-6, PB0-7, PD0-7, PE0-4.....	43
4.3 PA7.....	44
内存功能图.....	45
F-Plane	45
R-Plane	47
指令集.....	50
电气特性.....	62
1. 绝对最大值.....	62
2. 直流 DC 特性	63
3. 时钟计时.....	65
4. 复位定时特性.....	65
5. LVR 线路特性	65
6. 比较器特性.....	65
7. 特性曲线图.....	66
封装说明.....	67
24 脚直插封装尺寸(300 mil).....	68
24 脚贴片封装尺寸(300 mil).....	69
28 脚直插封装尺寸(300 mil).....	70
28 脚直插封装尺寸(600 mil).....	71
28 脚贴片封装尺寸(300 mil).....	72
32 脚直插封装尺寸(600 mil).....	73
32 脚贴片封装尺寸(300 mil).....	74
32 脚贴片封装尺寸(450 mil).....	75

基本功能

1. **ROM: 4K x 14 位 OTP 或 2K x 14 位 TTP™ (ROM 可分两次分区编程)**
2. **RAM: 176 x 8 位**
3. **堆栈:6 级**
4. **振荡来源**
 - 快时钟
 - FXT (高速晶振): 1M~24 MHz
 - FIRC (内部高速 RC): 2/4/8/16 MHz
 - XRC (外部 RC): 10K~3 MHz
 - 慢时钟
 - SXT (低速晶振): 32768 Hz
 - XRC (外部 RC): 10K~3 MHz
 - SIRC (内部低速 RC): 168K/40K/9.8K/2.6 KHz @5V; 128K/30.3K/7.6K/2K @3V
 - TKRC (触摸键时钟): 128K/64K/16K/4 KHz @5V, 非触摸; 80K/40K/10K/2.5K @3V, 非触摸
5. **双系统时钟**
 - FIRC + SIRC
 - FIRC + SXT
 - FIRC + XRC
 - FIRC + TKRC
 - FXT + SIRC
 - FXT + TKRC
 - XRC + SIRC
 - XRC + TKRC
6. **省电操作模式**
 - 快时钟模式: 慢时钟可以禁止或使能
 - 慢时钟模式: 快时钟停止, CPU 持续运行
 - 空闲模式: 慢时钟持续运行, CPU 停止, Timer2 持续运行
 - 省电模式: 所有时钟停止, 唤醒定时器可以禁止或使能
7. **操作电压和速度: VDD=1.6V @4 MHz**

8. 3个独立定时器

- Timer0
 - 带有 1~256 预分频，计数器/中断/停止功能的 8 位定时器
 - 捕获：高占空比或低占空比（脉冲宽度测量）
 - 溢出和转换出
- Timer1
 - 带有 2 个预分频，计数器/中断/停止/清零和保留/设置/重载功能的 16 位定时器
 - 捕获：周期时间
 - 溢出和转换出
- Timer2
 - 带有 4 种中断时间间隔选项的 15 位定时器
 - 用作空闲模式唤醒定时器或普通的 15 位定时器
 - 时钟来源: SXT / XRC / SIRC / TKRC

9. 中断

- 三个外部中断引脚
 - 两个为下边沿触发引脚
 - 一个为上下边沿触发引脚
- Timer0 / Timer1 / Timer2 / WKT (唤醒) 中断
- 比较器输出变化中断

10. PB[7:0] 单独低电平触发引脚

11. 唤醒 (WKT)定时器

- 来自内部 RC 振荡模块的 4 组复位与唤醒定时
0.9 ms/1.8 ms/30 ms/120 ms @5V, 1 ms/2 ms/32 ms/128 ms @3V

12. 看门狗(WDT)定时器

- 来自内部 RC 振荡模块的 4 组复位与唤醒定时
100 ms/200 ms/800 ms/1600 ms @5V, 130 ms/280 ms/1100 ms/2200 ms @3V
- 看门狗定时器功能可在省电模式下被禁止/使能(WDTSLPSTP, R0Eh.5)

13. 2 个独立 PWMs

- PWM0:
 - 带有 1~8 预分频，周期可调/占空比可调/清零和保留/非反相或反相输出。
- PWMA:
 - 8+2 位，占空比可调的 PWM

14. 1 个模拟电压比较器

15. 15 信道触摸键，支持一个按键唤醒用于低电源消耗

16. 复位来源

- 上电复位 / 看门狗定时器复位 / 低电压复位 / 外部引脚复位

17. 低电压复位选项: LVR1.5V, LVR1.5V 于省电模式下停止, LVR2.3V, LVR3.2V

18. 工作电压: 设置复位电压至 5.5V

- fosc = 4 MHz, 1.7V ~ 5.5V
- fosc = 8 MHz, 1.8V ~ 5.8V
- fosc = 12 MHz, 2.1V ~ 5.5V
- fosc = 16 MHz, 3.1V ~ 5.5V
- fosc = 24 MHz, 4.0V ~ 5.5V

19. 操作温度范围: -40°C to +85°C

20. 指令表: 36 条指令

21. 指令执行时间

- 2 个振荡时钟于每个指令 (除了分支指令)

22. 输入/输出端口状态: 最高达 29 个可编程的 I/O 引脚

- 伪开漏式输出
- 开漏式输出
- CMOS 推挽输出
- 施密特输入或上拉输入

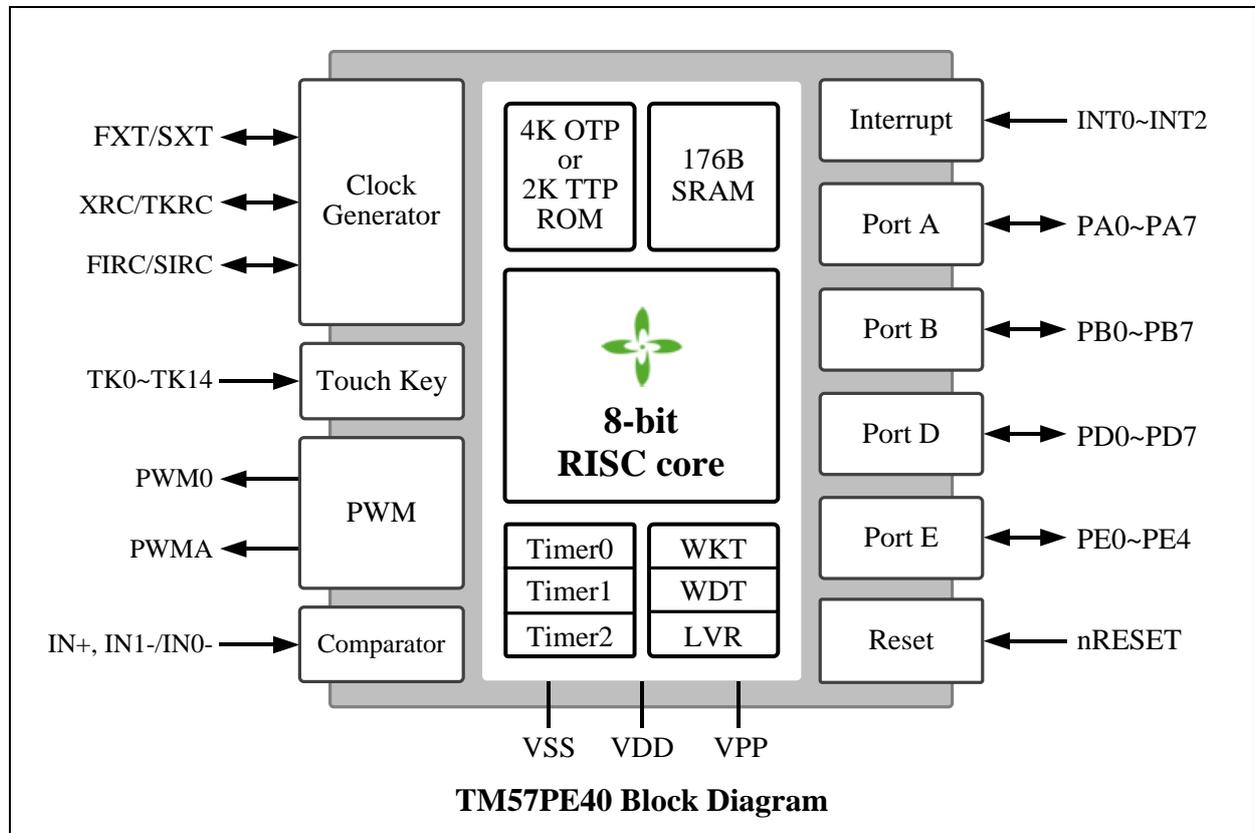
23. 封装类别:

- 24-pin DIP (300 mil), SOP (300 mil)
- 28-pin DIP (300/600 mil), SOP (300 mil)
- 32-pin DIP (600 mil), SOP (300/450 mil)

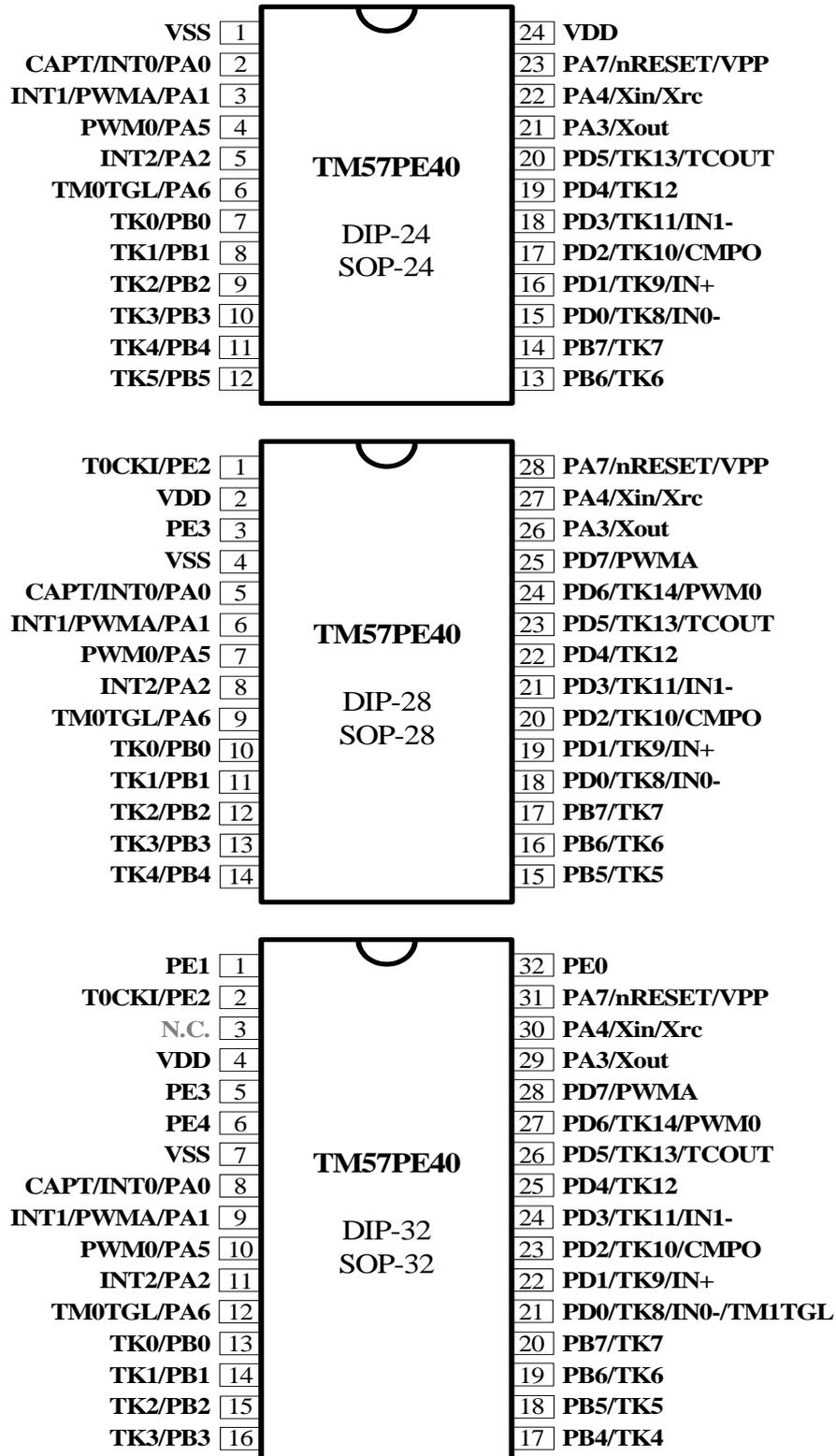
24. 支援的 EV board 于 ICE

EV board: EV2787

系统结构图



引脚分配图



管脚描述

名称	输入/输出	管脚描述
PA0-PA2	I/O	位编程 I/O 端口，可施密特触发输入，CMOS 推挽输出或者伪开漏式输出。上拉电阻由软件分配。
PA3-PA6	I/O	位编程 I/O 端口，可施密特触发输入，CMOS 推挽输出或者开漏式输出。上拉电阻由软件分配。
PA7	I	施密特上拉输入
PB0-PB7 PD0-PD7 PE0-PE4	I/O	位编程 I/O 端口，可施密特触发输入，CMOS 推挽输出或者开漏式输出。上拉电阻由软件分配。
nRESET	I	外部复位引脚（低电平触发）
Xin, Xout	-	供应系统时钟的晶振输入引脚
Xrc	-	供应系统时钟的外部 RC 振荡输入引脚
VDD, VSS	P	电源电压输入引脚和接地引脚
VPP	I	PROM 编程高电平输入引脚
INT0-INT2	I	外部中断输入引脚
PWM0/PWMA	O	PWM 输出引脚
TCOUT	O	除以 N 的输出指令周期时钟，N 为 1, 2, 4, 8。指令时钟频率是系统时钟频率除以 2 ($F_{cpuclk}/2$)。
T0CKI	I	Timer0 输入在计数模式
CAPT	I	Timer0/Timer1 捕获输入
TK14-TK0	I	触摸键输入
IN+	I	比较器正值输入
IN1-/IN0-	I	比较器负值输入，以 CMPINNS 位选择
TM0TGL	O	Timer0 溢出转换输出
TM1TGL	O	Timer1 溢出转换输出
CMPO	O	比较器输出

管脚摘要

管脚数量			管脚名称	类别	GPIO					复位后功能	替代功能			
32-SOP/DIP	28-SOP/DIP	24-SOP/DIP			输入		输出				PWM	Touch Key	ADC	MISC
					上拉	外部中断	O.D	P.O.D	P.P					
1	-	-	PE1	I/O	○		○		○					
2	1	-	T0CKI/PE2	I/O	○		○		○				T0CKI	
3	-	-	N.C.	-										
4	2	24	VDD	P										
5	3	-	PE3	I/O	○		○		○					
6	-	-	PE4	I/O	○		○		○					
7	4	1	VSS	P										
8	5	2	CAPT/INT0/PA0	I/O	○	○		○	○				CAPT	
9	6	3	INT1/PWMA/PA1	I/O	○	○		○	○	○				
10	7	4	PWM0/PA5	I/O	○		○		○	○				
11	8	5	INT2/PA2	I/O	○	○		○	○					
12	9	6	TM0TGL/PA6	I/O	○		○		○				TM0TGL	
13	10	7	TK0/PB0	I/O	○		○		○		○			
14	11	8	TK1/PB1	I/O	○		○		○		○			
15	12	9	TK2/PB2	I/O	○		○		○		○			
16	13	10	TK3/PB3	I/O	○		○		○		○			
17	14	11	PB4/TK4	I/O	○		○		○		○			
18	15	12	PB5/TK5	I/O	○		○		○		○			
19	16	13	PB6/TK6	I/O	○		○		○		○			
20	17	14	PB7/TK7	I/O	○		○		○		○			
21	18	15	PD0/TK8/IN0- /TM1TGL	I/O	○		○		○		○		TM1TGL/ CMP	
22	19	16	PD1/TK9/IN+	I/O	○		○		○		○		CMP	
23	20	17	PD2/TK10/CMPO	I/O	○		○		○		○		CMP	
24	21	18	PD3/TK11/IN1-	I/O	○		○		○		○		CMP	
25	22	19	PD4/TK12	I/O	○		○		○		○			
26	23	20	PD5/TK13/TCOUT	I/O	○		○		○		○		TCOUT	
27	24	-	PD6/TK14/PWM0	I/O	○		○		○	○	○			
28	25	-	PD7/PWMA	I/O	○		○		○	○				
29	26	21	PA3/Xout	I/O	○		○		○					
30	27	22	PA4/Xin/Xrc	I/O	○		○		○					
31	28	23	PA7/nRESET/VPP	I/O	○								nRESET	
32	-	-	PE0	I/O	○		○		○					

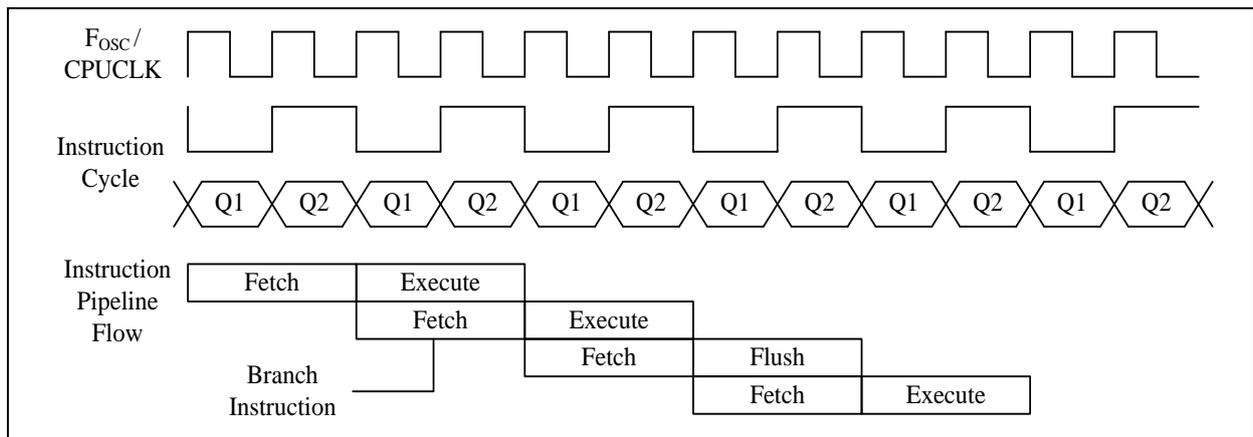
符号: P.P. = 推挽输出
 P.O.D. = 伪开漏式
 O.D. = 开漏式
 SYS = 透过 SYSCFG 位

功能描述

1. CPU 的特色

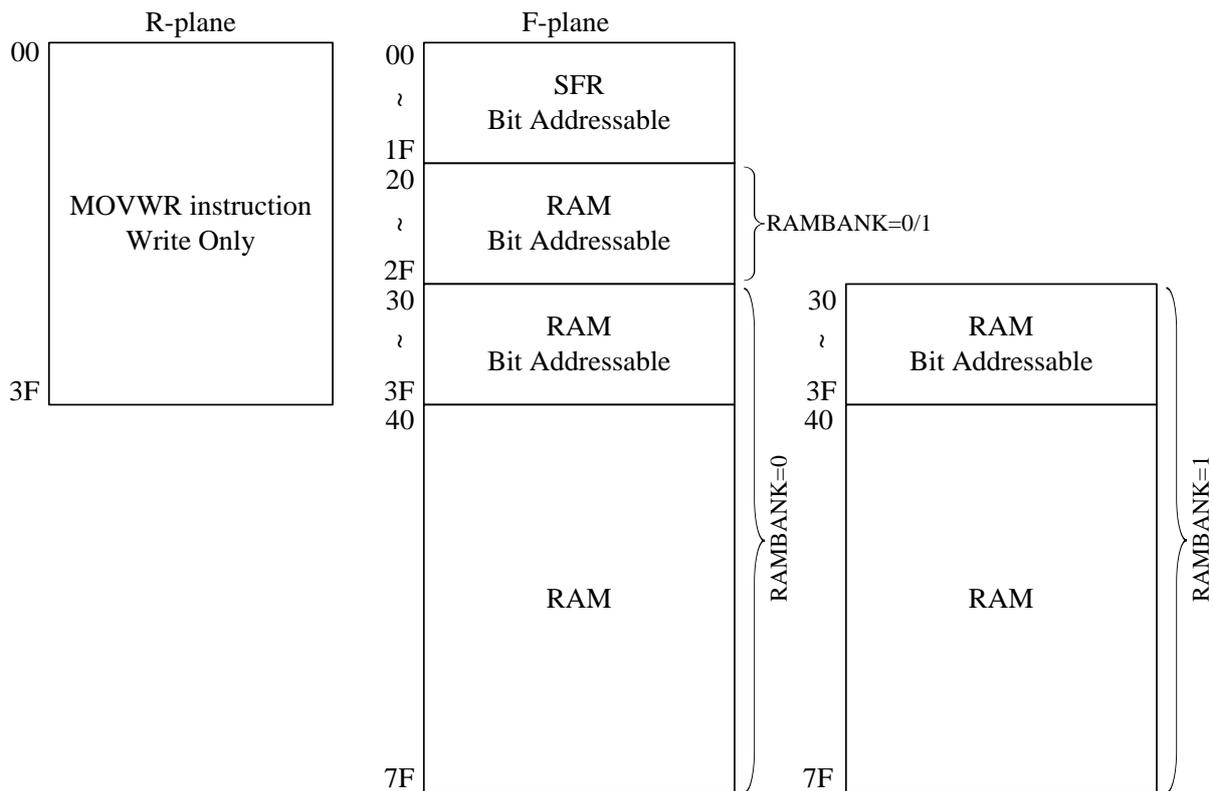
1.1 时钟配置和指令周期

系统时钟输入在内部被分成用于指令周期的 Q1 和 Q2 两个状态。程序计数器 (PC) 在 Q1 状态更新且指令从程序寄存器获得，在状态 Q2 锁存于指令寄存器。在接下来的 Q1-Q2 周期，程序被解码并执行。分支指令占用两个周期，因为从传输信道获得指令的同时新指令被获取并执行。



1.2 寻址模式

CPU 有两个数据储存器：R-Plane 和 F-Plane。R-Plane 里的寄存器只能写入。“MOVWR”指令通过直接寻址方式把 W 寄存器里的常数放在 R-Plane 寄存器中。F-Plane 的低位保留用于 SFR。SFR 寄存器以上的寄存器是一般目的的数据储存器，作为静态 RAM 使用。F-Plane 可被直接或间接寻址。通过 INDF 寄存器可间接寻址。INDF 寄存器不是物理寄存器。当对 INDF 进行存取时，它会根据 FSR 寄存器内的值作为地址，并指向该地址的寄存器。F-Plane 的前半部是可寻址之位，后半部是不可寻址之位。



1.3 程序计数器（PC）和堆栈

程序计数器是一个 12 位宽，可寻址 4K*14 的程序只读存储器。当一个程序指令被执行时，PC 里就存放着下一个将要被执行程序的地址。PC 值都会自动加 1，以下情况除外：复位向量（000h）和中断向量（001h）用来做 PC 初始化和中断。对于 CALL/GOTO 指令，PC 从指令中加载 12 位地址。对于 RET/RETI/RETLW 指令，PC 从堆栈顶取回其他地址值。对于更新 PC[7:0]的其他指令，PC[11:8]不变。堆栈有 12 位宽，深 6 级。CALL 指令和硬件中断将依顺序进入堆栈。RET/RETI/RETLW 指令将按顺序弹出堆栈。

1.4 ALU 和工作寄存器（W）

ALU 为 8 位算术逻辑单元，可进行加，减，移位和逻辑运算。在双操作数的指令中，一个操作数为 W 寄存器（W 寄存器为不可寻址寄存器，用于 ALU 运算），另一个操作数为文件寄存器或立即常数。在单操作数的指令中，操作数为 W 寄存器或文件寄存器。视指令执行情况，ALU 可能影响状态寄存器中的进位（C），半进位（DC），和零（Z）标志位。C 和 DC 标志在减法运算中用于借位和半借位标志。

备注：借位状态和借位值相反
半借位状态和半借位值相反

1.5 状态寄存器

状态寄存器包含 ALU 计算结果的状态和复位状态。状态寄存器和其他寄存器一样可以是任何寄存器的目的单元格。如果是影响 Z, DC, 或 C 标志位的指令以状态寄存器作为目的单元格, 那么就不可以写进这三个标志位了。这些位的置位或清零是来自 ALU 的逻辑运算值。所以, 只有 BCF, BSF, 和 MOVWF 指令被推荐选择状态寄存器用, 因为这些指令不影响标志值。

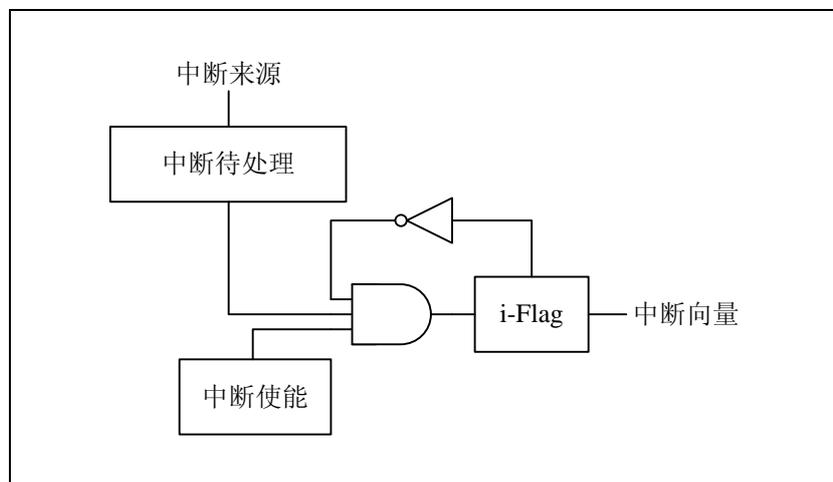
状态	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
复位值	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W
位	描述							
7	通用位							
6	通用位							
5	RAMBANK							
4	TO: 时间溢出 0: 上电复位后, LVR 复位或执行 CLRWDT/SLEEP 指令后 1: WDT 计时发生溢出							
3	PD: 省电模式标志 0: 上电复位后, LVR 复位或执行 CLRWDT 指令后 1: 执行 SLEEP 指令							
2	Z: 零标志位 0: 逻辑操作的结果不是 0 1: 逻辑操作的结果是 0							
1	DC: 辅助进位标志或借位标志							
	加法指令				减法指令			
	1: 低四位有进位 0: 无进位				1: 无借位 0: 低四位有借位			
0	C: 进位标志或借位标志							
	加法指令				减法指令			
	1: 高四位有进位 0: 无进位				1: 无借位 0: 高四位有借位			

1.6 中断

TM57PE40 有八个中断源共享的一个中断向量，因此没有中断优先级。每个中断源都有它自己得使能控制位，每个中断事件都可以触发它自己的中断标志位。因为 TM57PE40 没有中断优先权，所以 TM57PE40 中断优先处理取决于 F/W。

若相对的中断使能控制位已被置 1 (INTE)，它将触发 CPU 在最近执行指令周期的结尾处，接受并处理中断。同时，“CALL 001”指令被插进 CPU，并且 i-flag (中断标志位) 被置 1 来防止中断嵌套。

防止中断嵌套在执行“RETI”指令后即被清零。也就是说，在中断触发时至少一条指令在主程序中执行着。中断事件是边沿触发的。在处理中断程序后，F/W 必须清零中断事件的中断标志位。



2. 芯片工作模式

2.1 复位

可用以下四种方式复位：

- 上电复位
- 低电压复位（LVR）
- 通过 RESET 引脚来实现硬件复位
- 看门狗复位（WDT）

上电复位后，所有系统和外围控制寄存器都将恢复他们的默认硬件复位值。时钟源，LVR 和芯片工作模式在 SYSCFG 寄存器中设置。当电源电压低于阈值水平时，低电压复位功能静态重置。复位电压阈值有三组可以选择。LVR 的工作模式由 SYSCFG 寄存器定义。LVR32 适合 V_{DD} 大于 3.6V 的应用，LVR23 适合 V_{DD} 大于 2.7V 的应用，LVR15 适合 V_{DD} 大于 1.9V 的应用。若操作频率快于 16 MHz，建议选择 LVR 3.2V。

请参考以下 LVR 选择表；使用者必须考虑到操作频率的最低工作电压。

LVR 选项表：

LVR 阈值水平	考虑工作电压选择 LVR
LVR32	$5.5V > V_{DD} > 3.6V$
LVR23	V_{DD} 为较宽的 V_{DD} 电源电压， $3.6V > V_{DD} > 2.7V$
LVR15	V_{DD} 为较宽的 V_{DD} 电源电压， $2.7V > V_{DD} > 1.9V$

外部引脚复位和看门狗复位的禁止和启用由 SYSCFG 寄存器设置。这两种复位也将所有控制寄存器恢复到他们的默认复位值。而 TO/PD 标志不受这些复位影响。

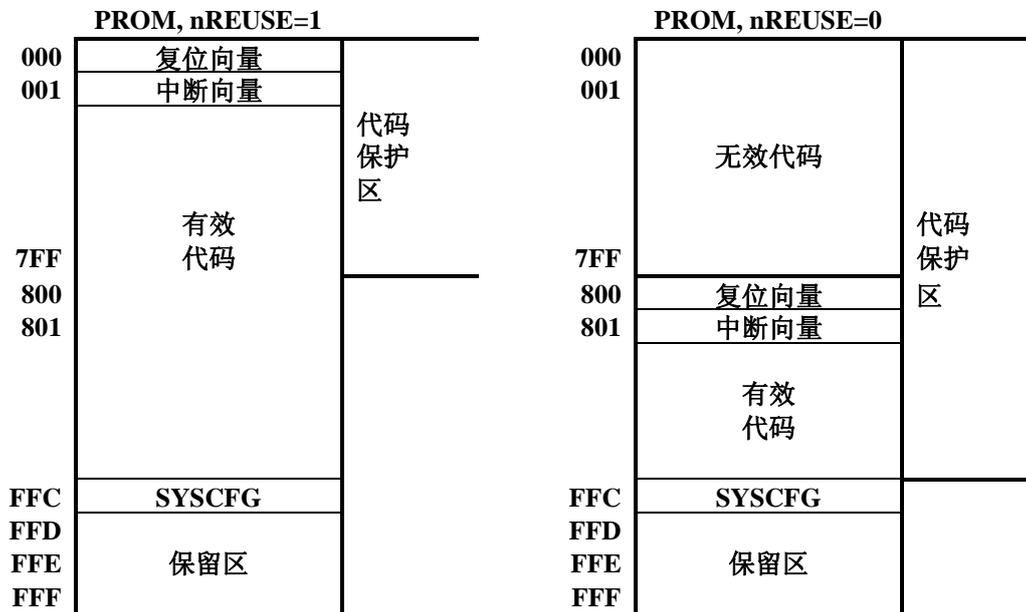
2.2 系统配置寄存器(SYSCFG)

系统配置寄存器（SYSCFG）位于 ROM 的地址是 FFCh。SYSCFG 的功能取决于 MCU 的初始化条件。该位只可被编程者写入。使用者可以通过写入 SYSCFG 值来选择时钟源，LVR 阈值电压和芯片工作模式。SYSCFG 的预设值为 14'b11_1111_111x_xxxx。SYSCFG 的第 13 位为代码保护选择位。若此位为 0，当使用者读取 PROM 时，PROM 的数据将被保护。

位	13~0	
预设值	11_1111_111x_xxxx	
位	描述	
13	nPROTECT: 代码保护选择位	
	1	不保护
	0	保护
12	nREUSE: 程序重新写入控制位	
	1	非重新写入
	0	重新写入
11-10	LVR: LV（低电压）复位模式	
	11	LVR 阈值为 1.5V, 总是处于使能状态
	10	LVR 阈值为 1.5V, 在省电模式处于禁止
	01	LVR 阈值为 2.3V, 总是处于使能状态
	00	LVR 阈值为 3.2V, 总是处于使能状态
9-8	CLKS: 快时钟源选择位	
	11	FXT (高速晶振) (1M~24 MHz)
	10	SXT (低速晶振, 32768 Hz)
	01	FIRC (高速内部 RC, 2/4/8/16 MHz by FIRCSEL[1:0])
	00	XRC (外部 RC)
7	XRESETE: 外部引脚复位选择位	
	1	外部引脚复位使能 (PA7 为复位引脚)
	0	外部引脚复位 禁止(PA7 为输入引脚)
6	WDTE: WDT 复位选择位	
	1	WDT 复位使能, WKT 定时器禁止
	0	WDT 复位禁止, WKT 定时器使能
5	测试模式, 保留值为“1”	
4-0	FIRCF: 高速内部 RC 频率调整控制位	

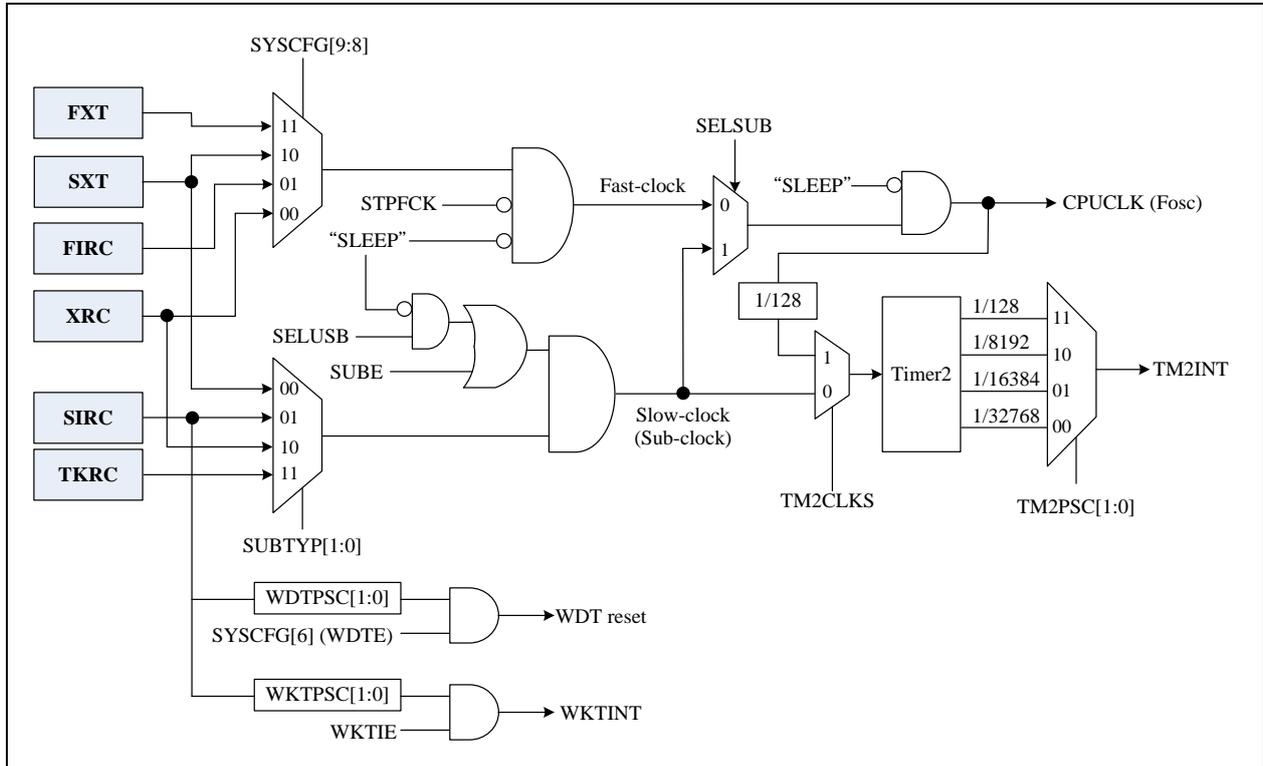
2.3 可分区编程

TM57PE40 程序存储器有 4K 字，比一些少于 2K 字的 F/W 程序来说要大很多。为了充分利用程序存储器，TM57PE40 允许编程者重复编程，故称为可两次可编程（TTP）ROM。当程序存储器的前一部分被无用的程序代码占用着，后一部分空间仍然是空白的，编程者就可在程序存储器的后半部分重写程序。在第二次编程的情况下，复位向量和中断向量会通过编译在程序的后半部分开头重新分配。编程者只需在 ICE 工具界面选择“重新写入(REUSE)”选项，然后编译器将目标代码放在相应的位置。也就是说，程序的复位值仍然在地址 000h，但汇编对象代码的复位值在 200h。在 SYSCFG 寄存器中，如果 nPROTECT=0 和 nREUSE=1，该代码保护区就在程序存储器的上半部分。这就允许在第二次编程时，编程者重新写入然后检验。把重新写入的程序代码放在程序存储器的后半部分以后，“nREUSE”写入“0”。此时，代码保护区域就变成除保留区的整个程序存储区了。



2.4 双系统时钟

TM57PE40 的设计是有双时钟系统。时钟来源有六种：FXT（高速晶振）时钟，SXT（低速晶振）时钟，XRC（外部 RC）时钟，SIRC（低速内部 RC）和 FIRC（高速内部 RC），它们都能作为 CPU 的时钟来源。在空闲模式时，只有慢时钟可被配置保持振荡提供时钟源为定时器 Timer2。请参考以下图。



时钟配置图

快时钟模式

上电或复位后 TM57PE40 将进入快时钟模式。在快时钟模式下，TM57PE40 可以通过 SYSCFG[9:8] 设置来选择 FXT, XRC, 或 FIRC 作为它的 CPU 时钟。另外，软体也可以禁止慢时钟作为 Timer2 的时钟源。

此时，程序以快时钟振荡作为 CPU 时钟（CPUCLK）执行。对于 Timer0, Timer1, PWM0, PWMA 的时钟源都来源于快时钟的预分频。Timer2 的时钟源也可以通过设置 TM2CLKS=1 和 SELSUB=0 来取得不同预分频。

慢时钟模式

在此模式下，TM57PE40 可以通过 R-Plane 控制寄存器（SUBTYP[1:0]）选择 SXT, XRC, SIRC 或 TKRC 作为 CPU 时钟源。在这种模式下，使能慢时钟禁止快时钟来节省耗电。所有外设（Timer0, Timer1, PWM0, PWMA, 等等...）的时钟源都来源于慢时钟。

空闲模式

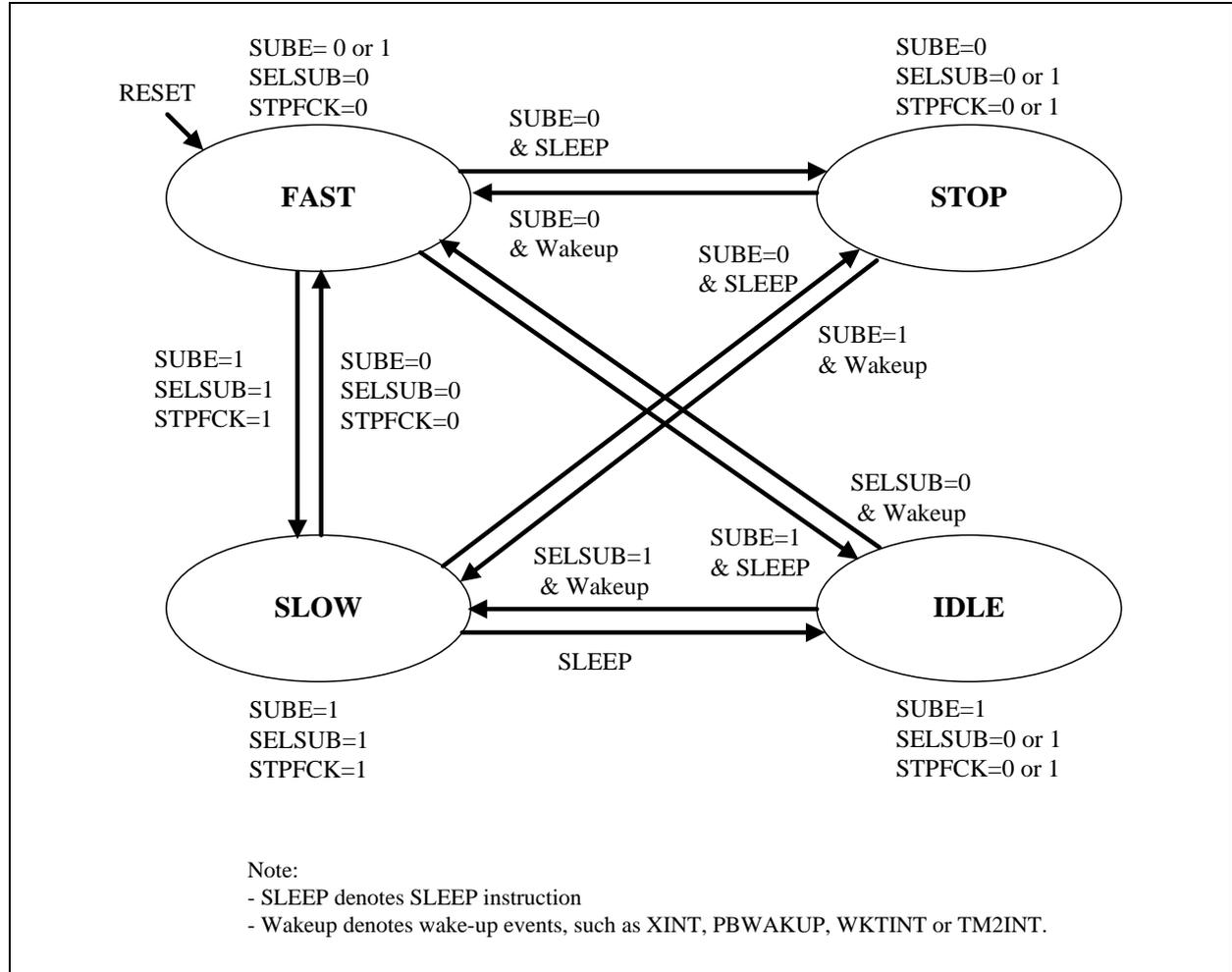
在执行 SLEEP 指令前把慢时钟使能，且 TM2CLKS=0, TM57PE40 进入空闲模式。在此模式，除慢时钟将持续保证 Timer2 模块的运行外，CPU 及其它所有电路都停止运行（如果 WDT/WKT 也被禁止）。

省电模式

在执行 SLEEP 指令前把慢时钟禁止，TM57PE40 将进入省电模式，所有模块都被禁止。省电模式与空闲模式是类似的。其差别在于省电模式下所有电路都停止运行（如果 WDT/WKT 也被禁止）。

2.5 双系统时钟的模式转换

TM57PE40 运作在四种模式之一：快时钟模式，慢时钟模式，空闲模式，和省电模式。



CPU 运作模块图

模式	振荡器	CPUCLK	快时钟	慢时钟	TM0/TM1	TM2	唤醒事件
快时钟	FIRC, FXT, SXT, XRC	快时钟	运行	以 STPFCK 设置	运行	运行	X
慢时钟	SXT, XRC, SIRC, TKRC	慢时钟	以 SUBE 设置	运行	运行	运行	X
空闲	SXT, XRC, SIRC, TKRC	停止	停止	运行	停止	运行	WKT/IO/PB/TM2
省电	SIRC ⁽¹⁾	停止	停止	停止	停止	停止	WKT/IO/PB

(1) 若 WDT 或 WKT 功能被使能

- **快时钟模式转换至慢时钟模式**

快时钟模式可以通过 SYSCFG [9:8] 设置为 11 (FXT)，00 (XRC)，或 01 (FIRC)。建议依序执行以下步骤将快时钟模式转换至慢时钟模式。

- (一) 使能慢时钟 (SUBE = 1)
- (二) 选择慢时钟作为 CPU 时钟源 (SELSUB = 1)
- (三) 停止快时钟 (STPFCK = 1)

- **慢时钟模式转换至快时钟模式**

慢时钟模式可以通过使能 CLKCTRL 的寄存器 SUBE 位和 SELSUB 位实现。建议依序执行以下步骤将慢时钟模式转换至快时钟模式。

- (一) 使能快时钟 (STPFCK = 0)
- (二) 选择快时钟作为 CPU 时钟源 (SELSUB = 0)
- (三) 停止慢时钟 (SUBE = 0)

注意：停止慢时钟 (SUBE = 0) 为非必要步骤。慢时钟可以持续振荡提供快时钟模式下的 Timer2 计数器模块。

- **空闲模式设置**

请依照以下设置配置空闲模式：

- (一) 使能慢时钟 (SUBE = 1)
- (二) 转换 Timer2 时钟来源至慢时钟 (TM2CLKS = 0)
- (三) 执行 SLEEP 指令

空闲模式可通过 XINT，PBWAKUP，WKT 定时器中断和 Timer2 中断唤醒。

- **省电模式设置**

请依照以下设置配置省电模式：

- (一) 终止慢时钟 (SUBE = 0)
- (二) 执行 SLEEP 指令

省电模式可通过 XINT，PBWAKUP 和 WKT 定时器中断唤醒。

PA3/PA4 IO 在双时钟模式设置注意事项

注意：在慢时钟模式，PA3 和 PA4 必须使能内部拉高。若慢时钟选择 SXT 或 XRC 模式，PA3 和 PA4 的 IO 设置列表如下

	快时钟	慢时钟	PAD3	PAE3	nPAPU3	PAD4	PAE4	nPAPU4
1	FIRC	SXT	1	0	0	1	0	0
2	FIRC	XRC	※	※	※	※	0	0
3	FXT	SIRC	1	0	0	1	0	0
4	XRC	SIRC	※	※	※	※	0	0

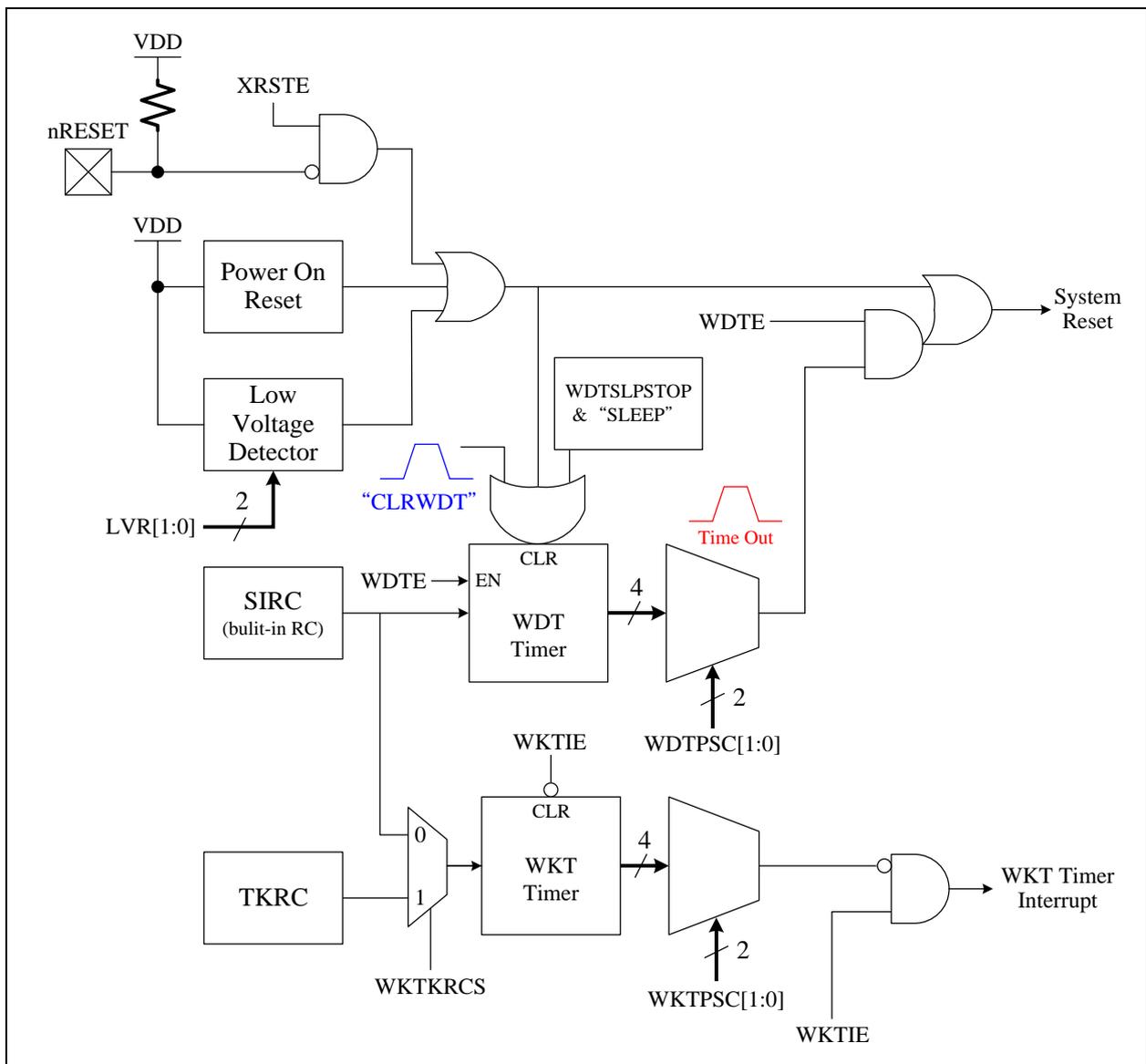
※：可忽略

3. 外围功能模块

3.1 看门狗 (WDT) /唤醒 (WKT) 定时器

WDT 和 WKT 是同一个内部 RC 振荡器 (SIRC)。WDT, WKT 的溢出周期可以单独选择 (WDTPSC[1:0], WKTPSC[1:0])。WDT 定时器以 CLRWDT 指令来清零。若看门狗被使能 (WDTE= 1), WDT 将产生芯片复位信号。设置 WDTSLPSTP (R0Eh.5)为 1 可以让 WDT 定时器在执行 SLEEP 指令后停止计数, 即 WDTSLPSTP =0, WDT 定时器将持续计数即使 SLEEP 指令已执行。

WKT 定时器为间隔定时器, WKT 失效将产生 WKT 中断标志 (WKTIF)。WKT 定时器可通过 WKTIE=0 清零/停止。当设置 WKTIE=1, WKT 定时器将计数不论在和种 CPU 运行模式。



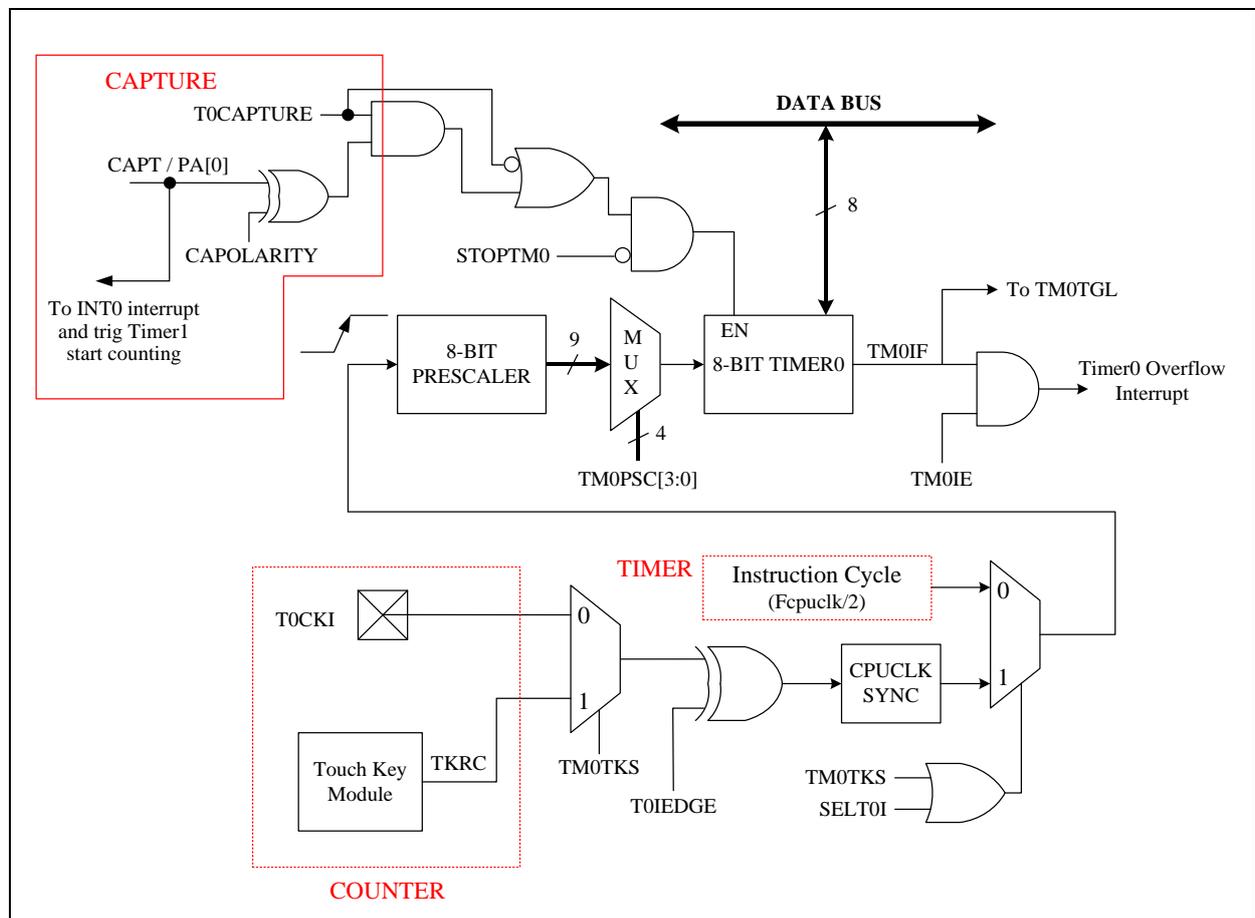
WDT/WKT 模块图

3.2 Timer0

Timer0 是位于 F-Plane 的 01h 的 8 位寄存器。如其他 F-Plane 寄存器一样可以读和写。另外，定时器在时钟预分频比率下周期性增加及自动溢出。这即是指令周期或 TOCKI (PE2) 上升/下降输入或触摸键振荡时钟 (TKRC) 上升/下降。Timer0 的增加比例由 R-Plane 中的 Timer0 预分频 (TM0PSC) 决定。Timer0 定时器溢出后可产生中断标志 (TM0IF) 若 Timer0 中断使能 (TM0IE) 被设置。Timer0 可停止计数若 STOPTM0 位被设置。TM0TGL 在 Timer0 溢出时为输出讯号触发。

Timer0 可被配置为捕获模式。若 TOCAPTURE 位被设为“1”，Timer0 于 CAPT 引脚 (即 PA0) 为高平(CAPOLARITY=0)或低平(CAPOLARITY=1)才会计数。

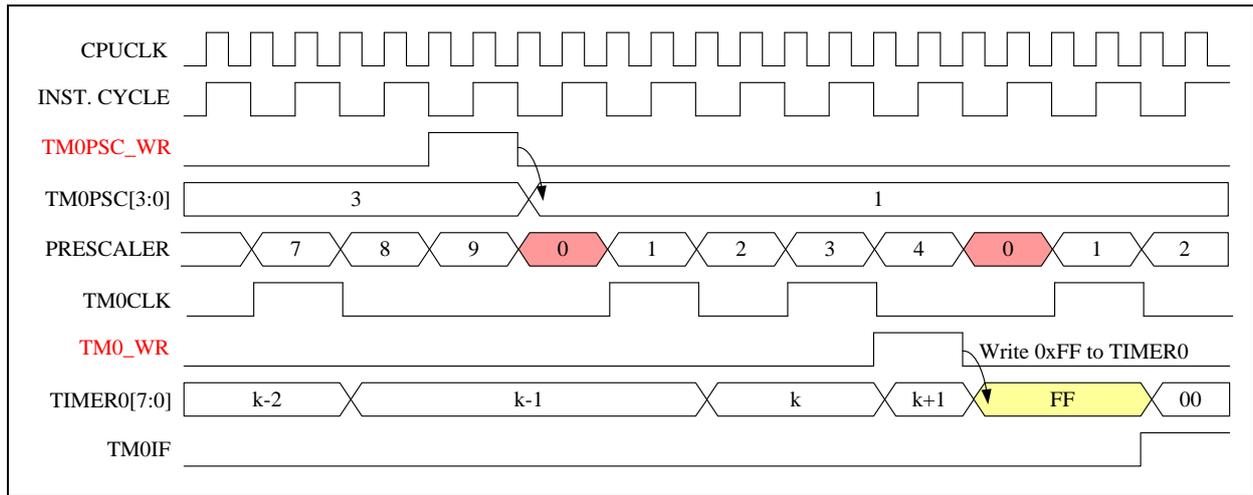
Timer0 也可被用于量测脉宽和周期捕获于 CAPT 引脚。此功能需要 Timer1 和 INTO 外部中断。软件控制细节将逐一说明。



Timer0 模块图

Timer0 运作在纯计时模式如下图所示：

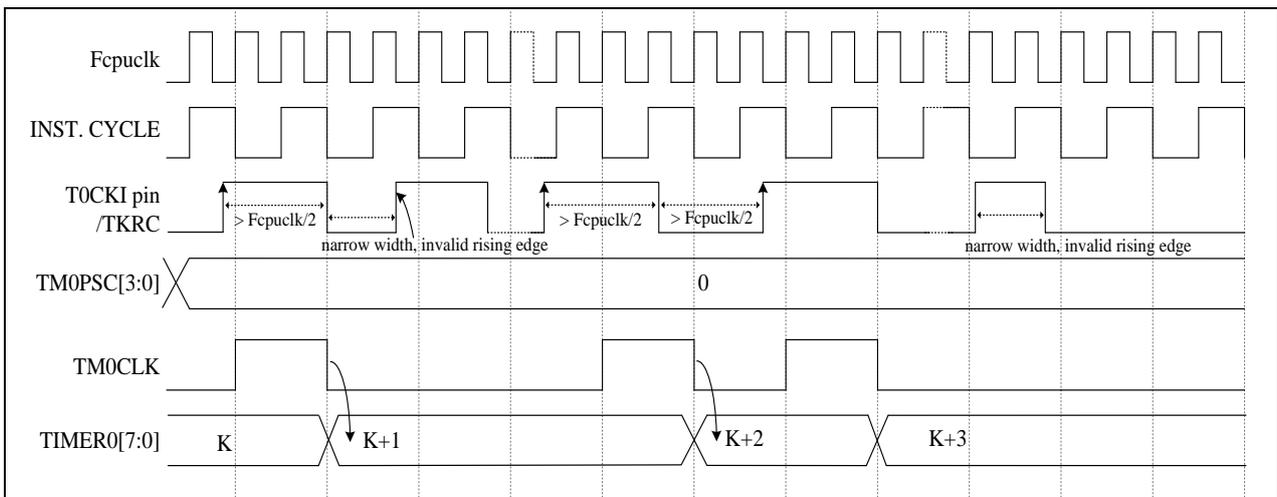
当 Timer0 预分频 (TM0PSC) 被写入，为了让第一次 Timer0 计数的周期正确，内部 8 位预分频将被清零至 0。TM0CLK 为内部讯号造成 Timer0 增加 1 在 TM0CLK 的最后。TM0WR 也是内部讯号表示 Timer0 是直接指令写入，同时，内部 8 位预分频将被清零。当 Timer0 从 FFh 数到 00h，TM0IF (Timer0 中断标志) 将被设为 1 并且若 TM0IE (Timer0 中断使能) 被设置，将会产生中断。



Timer0 在定时器模式下运作

以下时间图描述 Timer0 在计数模式下运行。

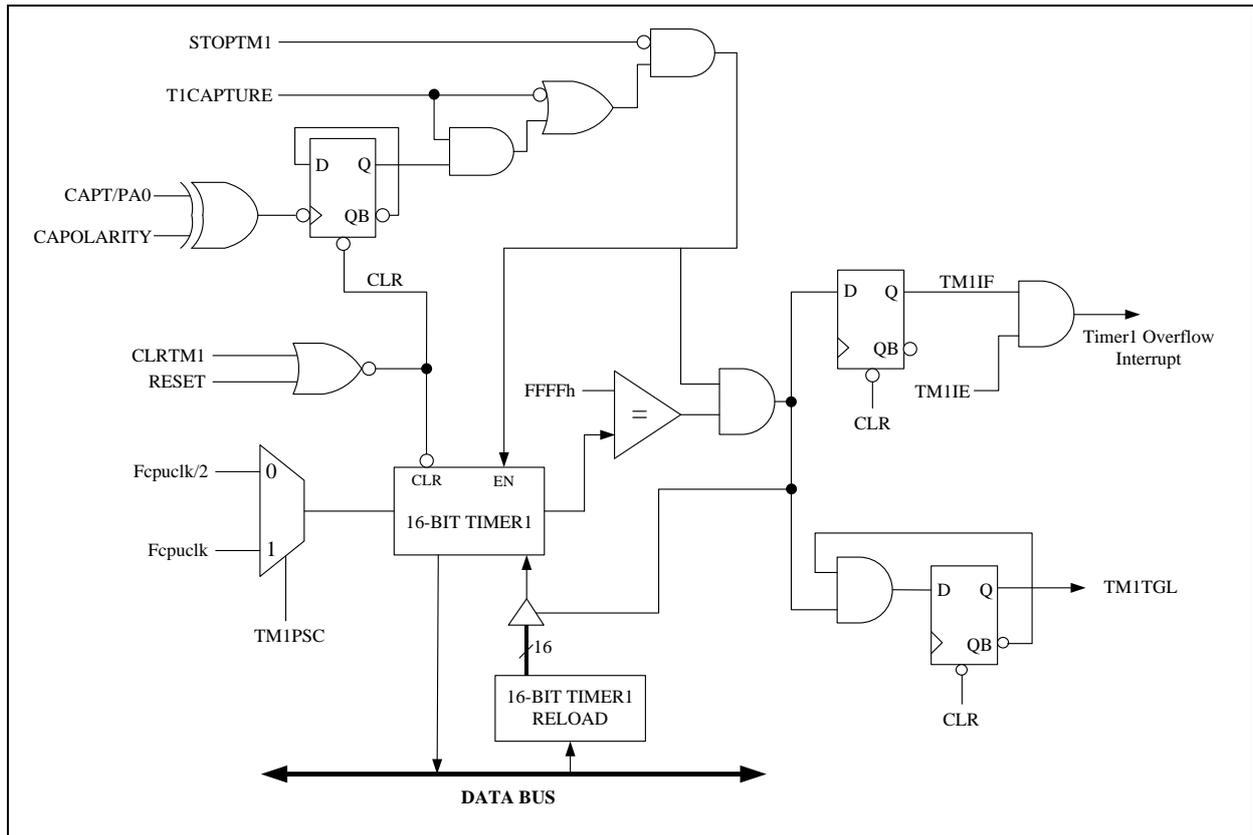
若 SELT0I=1，Timer0 的计数来源钟是从 TOCKI 引脚(TM0TKS=0)或触摸键模块(TM0TKS=1)。TOCKI 或 TKRC 讯号通过指令周期 ($F_{cpuclk}/2$) 做同步，意即，TOCKI/TKRC 的高/低时间区必须要大于一个指令周期时间确保 TOCKI /TKRC 的变化可被同步器正确侦测到。



Timer0 在计数器模式下于 TOCKI/TKRC (T0IEDGE=0)

3.3 Timer1

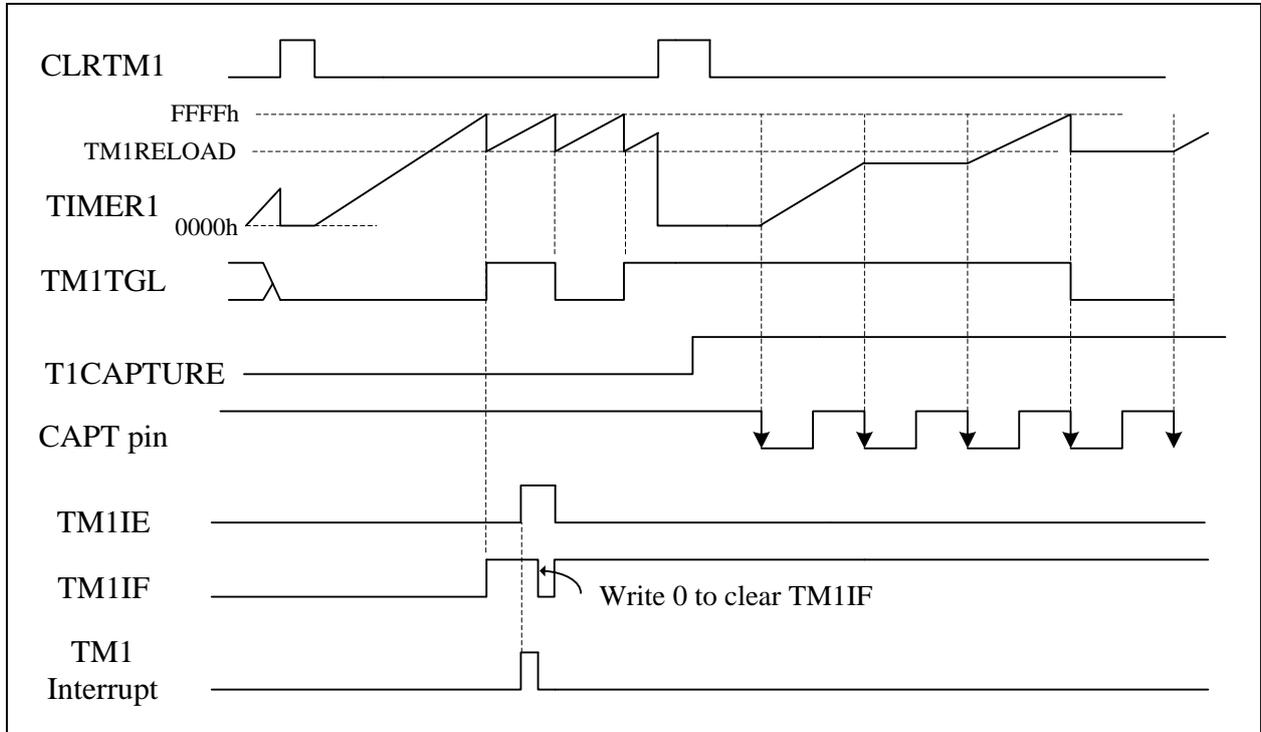
Timer1 是 16 位计数器以 16 位自动重载寄存器用于捕获/定时器模式。Timer1 只能通过读取 F-Plane TM1H 和 TM1L 存取。写入 TM1H 和 TM1L 实际上就是写入 Timer1 重载寄存器。Timer1 的时钟源为 Fcpuclk 和 Fcpuclk/2，通过 TM1PSC 选择。通过设置 CLR TM1 位将清零 Timer1 并且维持 Timer1 于 0000h。设置 STOPTM1 位将终止 Timer1 计数。TM1TGL 为输出讯号于 Timer1 溢出时被触发的。



Timer1 模块图

注意：写入 TM1H 和 TM1L 实际上就是写入 Timer1 重载寄存器，当读取 TM1H 和 TM1L 实际上是读取 Timer1 计数器本身。意即，Timer1 计数器和 Timer1 重载寄存器共享两个 F-Plane 地址(0ah, 0bh)。

Timer1 也可以运作于捕获模式。在捕获模式运作时，当 CLR_{TM1} 位被清零且 CAPT 引脚的第一次下降沿来临（若 CAP_{POLARITY}=0），Timer1 将开始计数。当 CAPT 引脚的第二次下降沿来临，Timer1 停止计数并且保留该值。当 CAPT 引脚的第三次下降沿来临，Timer1 将继续计数。以下图标显示计时详细情况。



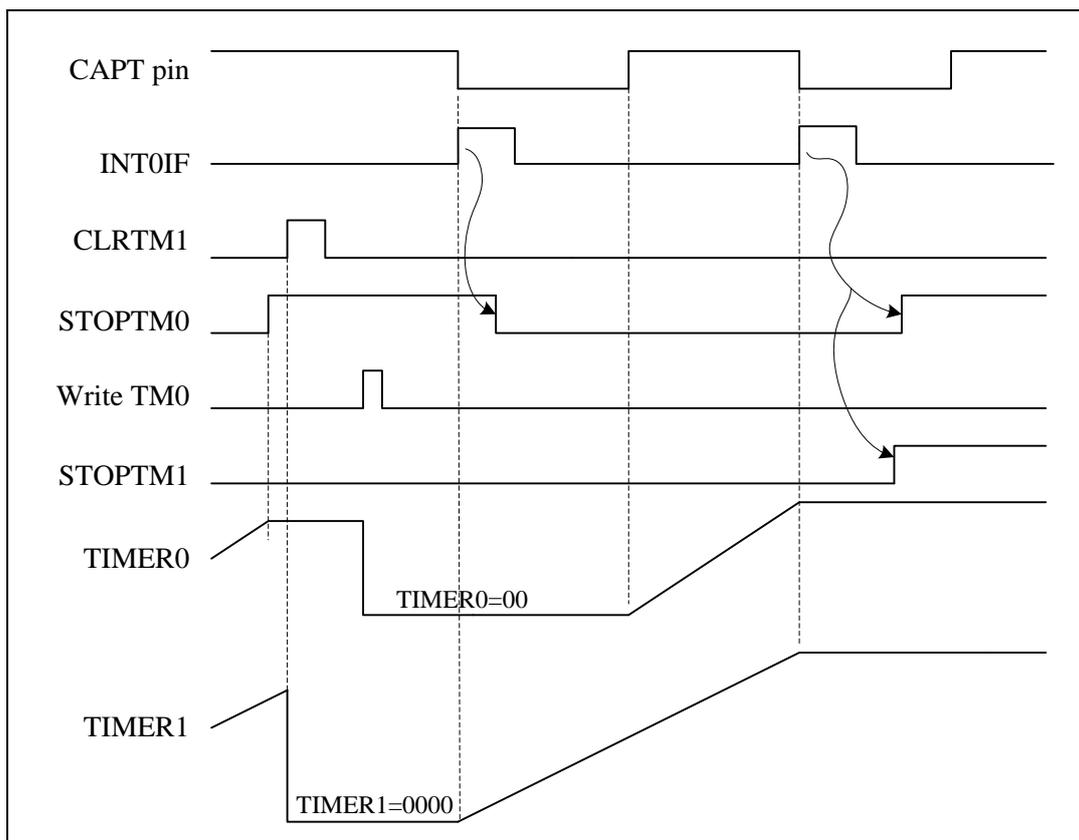
Timer1 在捕获模式下运作 (CAP_{POLARITY}=0, 意指 CAPT 下降沿)

3.4 Timer0 和 Timer1 用于脉宽及周期捕获

Timer0 和 Timer1 可以合作量测讯号周期和占空周期时间。关键是多功能的 PA0 (CAPT, INT0)。
假设：

- SELT0I=0 且 TM0TKS=0, Timer0 预分频于每个指令周期而增加。
- T0CAPTURE=1, T1CAPTURE=1. Timer0 和 Timer1 于捕获模式运作。
- PA0 引脚 (CAPT 引脚) 中断每个下降沿。
- CAPOLARITY=0, 当 PA0 引脚 (CAPT 引脚)下降沿来临时, **Timer1** 开始/保留。**Timer0** 开始计数当 PA0 引脚 (CAPT 引脚)位于逻辑‘1’的位子, 且当 PA0 引脚 (CAPT 引脚)位于逻辑‘0’的位子将保留 Timer0 值。
- Timer1 用于量测讯号周期, Timer0 用于量测 PA0 (CAPT 引脚)位于逻辑 ‘1’时间(意指, 讯号的占空周期)。

以下图标显示如何使用 Timer0 和 Timer1 量测 PA0 (CAPT 引脚) 讯号的周期和占空周期 (CAPOLARITY=0)。

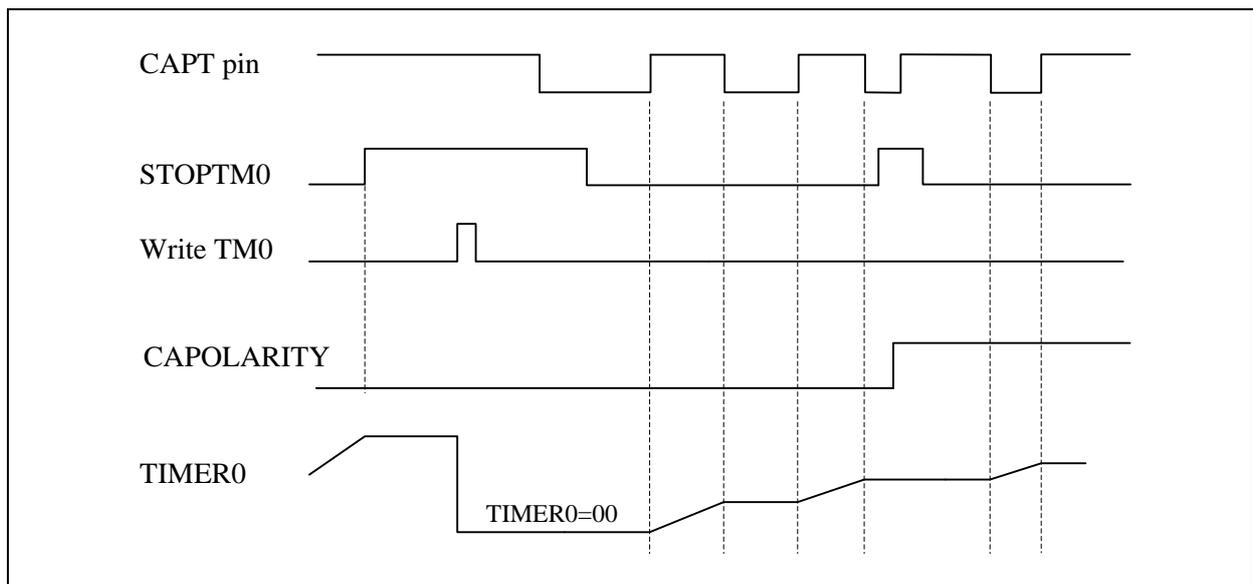


Timer0 和 Timer1 用于量测 CAPT 引脚的讯号

请遵循以下步骤开始量测 CAPT 引脚的周期和占空周期。

1. 通过 F/W 停止 Timer0 (STOPTM0=1, Timer0 将被停止和保留)
2. 通过 F/W 清零 Timer1 (CLRTM1=1)
3. 清零 Timer0 通过直接写入 00h 于 Timer0 (Timer0 仍然保留)
4. 一旦 CAPT 引脚下降沿来临, Timer1 开始计数, 同时 PA0 中断产生且 STOPTM0 被 F/W 清零。现在, Timer0 已准备好计数当 CAPT 引脚往上。
5. CAPT 引脚上升沿来临, Timer0 开始计数直到 CAPT 引脚回到 0 并且保留计数值。Timer1 也停止计数并且保留该值。
6. PA0 中断再次产生, F/W 停止 Timer1 和 Timer0 读取周期和占空周期。

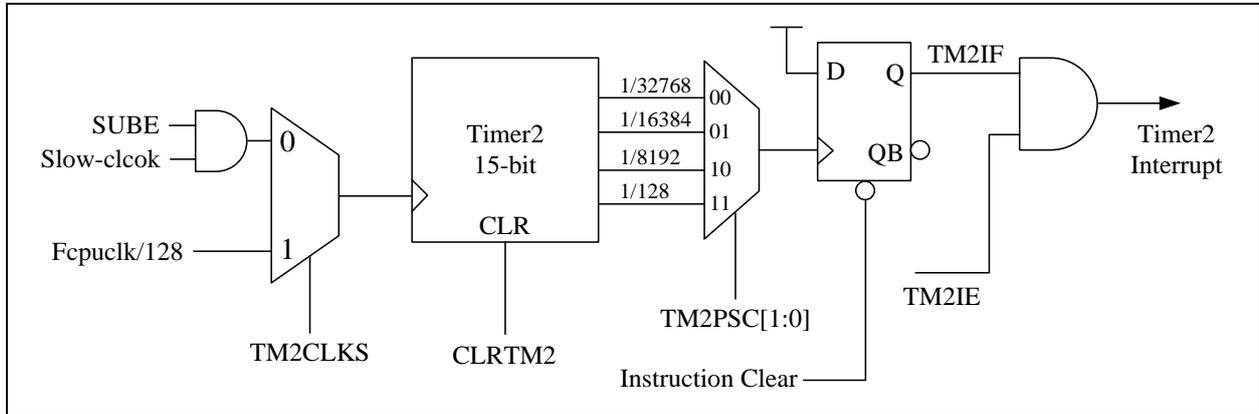
不需要同时使用 Timer0 和 Timer1。若占空周期 (CAPT 位于高时间) 需要被量测, 不需要使用 Timer1 量测周期。在此情况, 使用者可以设置 T0CAPTURE=1 和 T1CAPTURE=0。Timer0 只有当 CAPT 引脚为‘1’而计数。注意, 内部预分频将被保留到下次 Timer0 计数, 所以, 他不会失去计数的准确性。



Timer0 用于量测高 (或低)时间于 CAPT 引脚

3.5 Timer2: 15 位定时器

Timer2 是 15 位计数器且时钟源为 $F_{cpuclk}/128$ 或慢时钟。Timer2 用于产生固定的定时中断和计数器模块时钟。Timer2 的内容无法通过指令读取。Timer2 通过设置 TM2PSC[1:0] 寄存器来预分频 32768/16384/8192/128 产生中断标志 (TM2IF)。以下表示 Timer2 模块图。



Timer2 模块图

范例：

CPU 操作程序是由慢(SXT=32 KHz)转换到空闲模式并且每 0.5 sec 执行唤醒中断程序。

[CPU 运行在慢时钟模式, STPFCK=1, SUBE=1, SELSUB=1, WDTE=0]

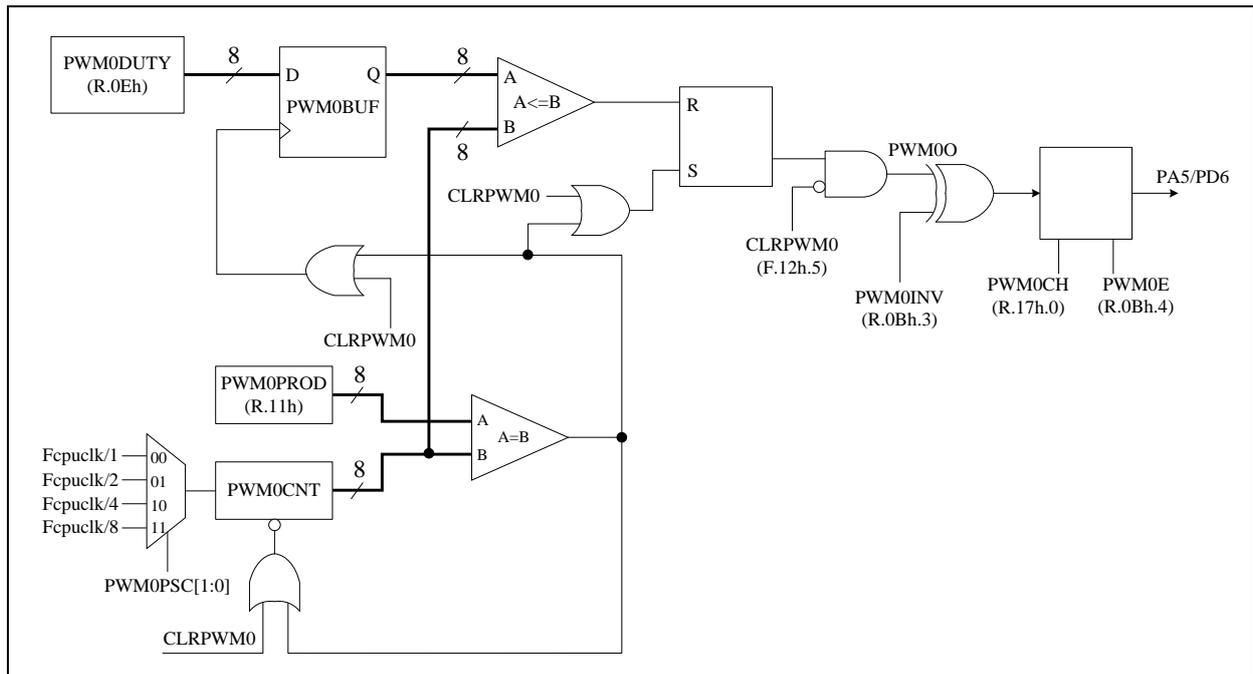
```

movlw    11000001    ; TM2CLKS=慢时钟(SXT=32 KHz), TM2PSC=div16384
bsf      TM2IE      ; Timer2 中断使能
bcf      CLRTM2     ; 勿清零 Timer2
sleep    ; 进入空闲模式, 等 Timer2 中断发生
nop
goto     $-2        ;
    
```

3.6 PWM0: 8 位 PWM

片内可产生一个 8 位 PWM 脉冲信号。其时钟源来自系统时钟除以 1, 2, 4, 8 预分频。PWM0 占空比可通过写入 PWM0DUTY 寄存器修改。对 PWM0DUTY 寄存器进行写入时不会立即改变 PWM0 输出占空比，而是要等下一周期开始时 PWM0BUF 才改变。

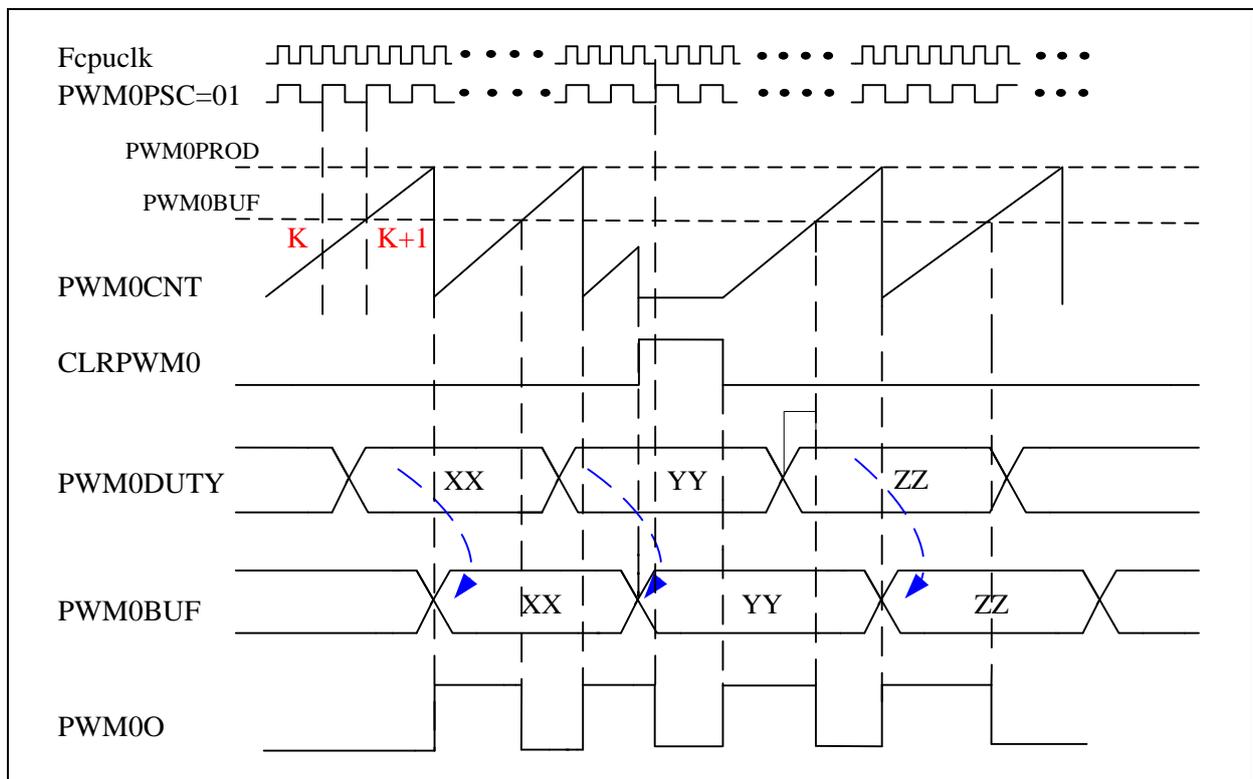
PWM0INV 可以反转 PWM0 输出。若 PWM0E 设为 1，PWM0 的其他输出管道选项可以是 PA5 或 PD6 由 PWM0CH 位控制。设置 CLRPWM0 将清零 PcPWM0 计数器并且加载 PWM0DUTY 至 PWM0BUF，CLRPWM0 位必须被清零，PWM0 计数器方能计数。以下图示表示 PWM0 模块图。



PWM0 模块图

图示中表示 PWM0 波形。当 CLRPWM0 位被设置为“1”时，PWM0 输出被清为 0 无论目前状态为何。一旦 CLRPWM0 位被清零，PWM0 输出将被设置为‘1’，开始 PWM 的新周期。当 PWM0CNT 大于或等于 PWM0BUF，PWM0 输出为 0。PWM0CNT 将持续计数直到等于 PWM0PRD，PWM0 输出将再次被设置为‘1’。

PWM0 周期可以通过写入周期值至 PWM0PROD 寄存器设置。注意，修改 PWM0PROD 将立即修改 PWM0PROD 的值，有别于 PWM0DUTY，还有 PWM0BUF 缓冲空间更新在目前周期的最后的占空比。程序设计者必须观察以下图留意目前时间修改 PWM0PRD。数位比较器将比较 PWM0CNT 计数器和 PWM0PROD，若设置 PWM0PROD 后，PWM0CNT 大于 PWM0PROD，错误的 PWM 长周期将产生，因为 PWM0CNT 必须计数直到溢出，然后维持计数 PWM0PROD 直到完成其周期。



PWM0 占空/周期计算如下：

- PWM0 输出占空= $PWM0DUTY / (PWM0PROD + 1)$
若 $PWM0DUTY = 80H$, $PWM0PROD = FFH$, PWM0 输出占空为 50%
- PWM0 输出频率= $(Fcpuclk) / (PWM0 \text{ 预分频}) / (PWM0PROD + 1)$

$PWM0PSC=2^b10$, PWM0 预分频= 4 ($Fcpuclk$ 除以 4)

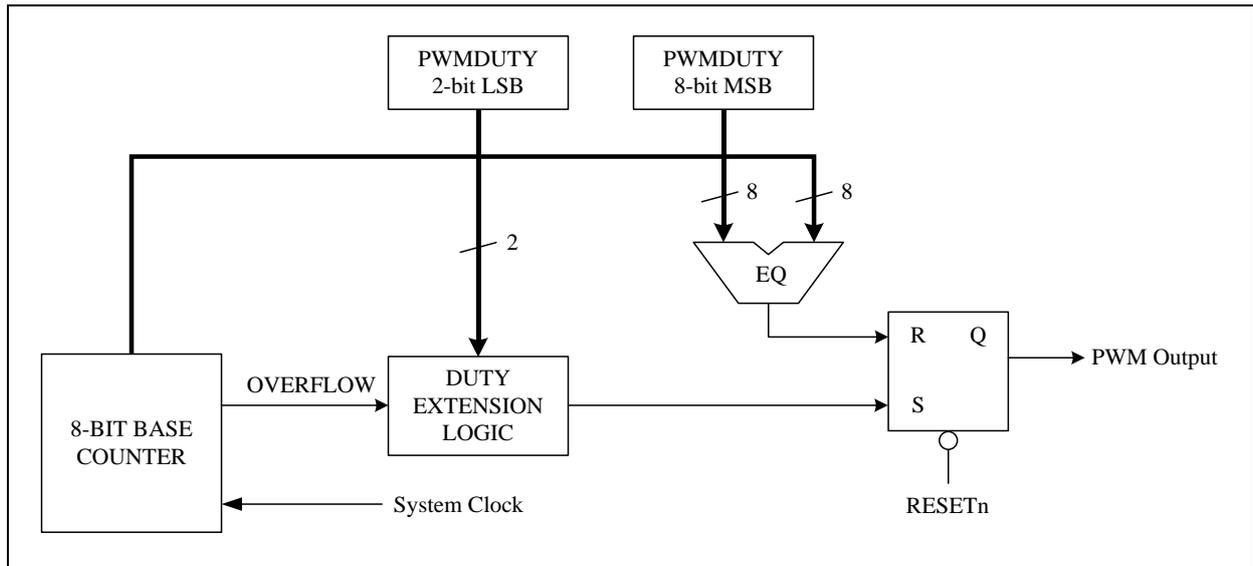
PWM0 周期数据($PWM0PROD$) = 39

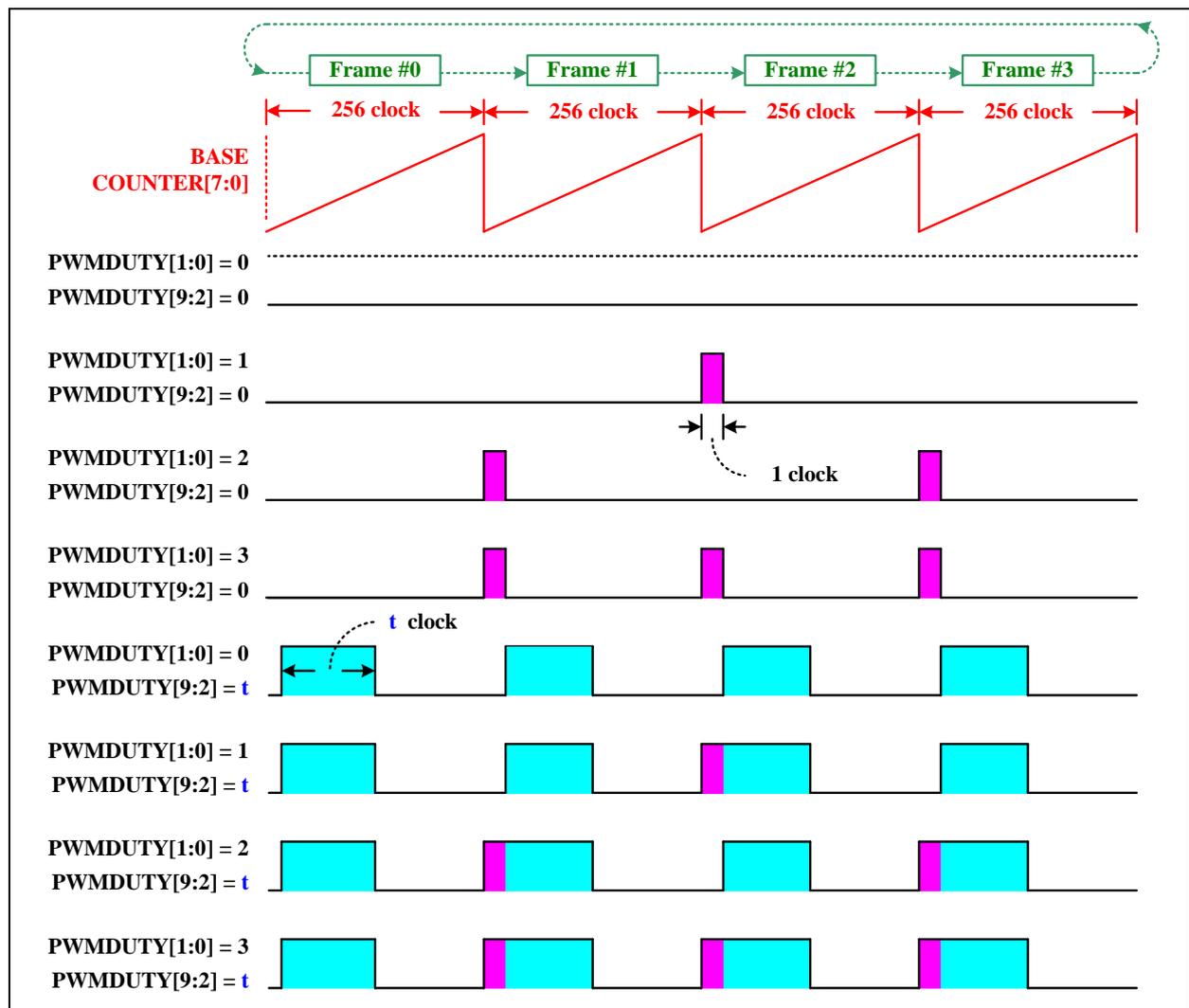
$Fcpuclk=4 \text{ MHz}$

PWM0 输出频率= $(4M)/(4)/(40) = 25000 \text{ Hz}$

3.7 PWMA: (8+2) 位 PWM

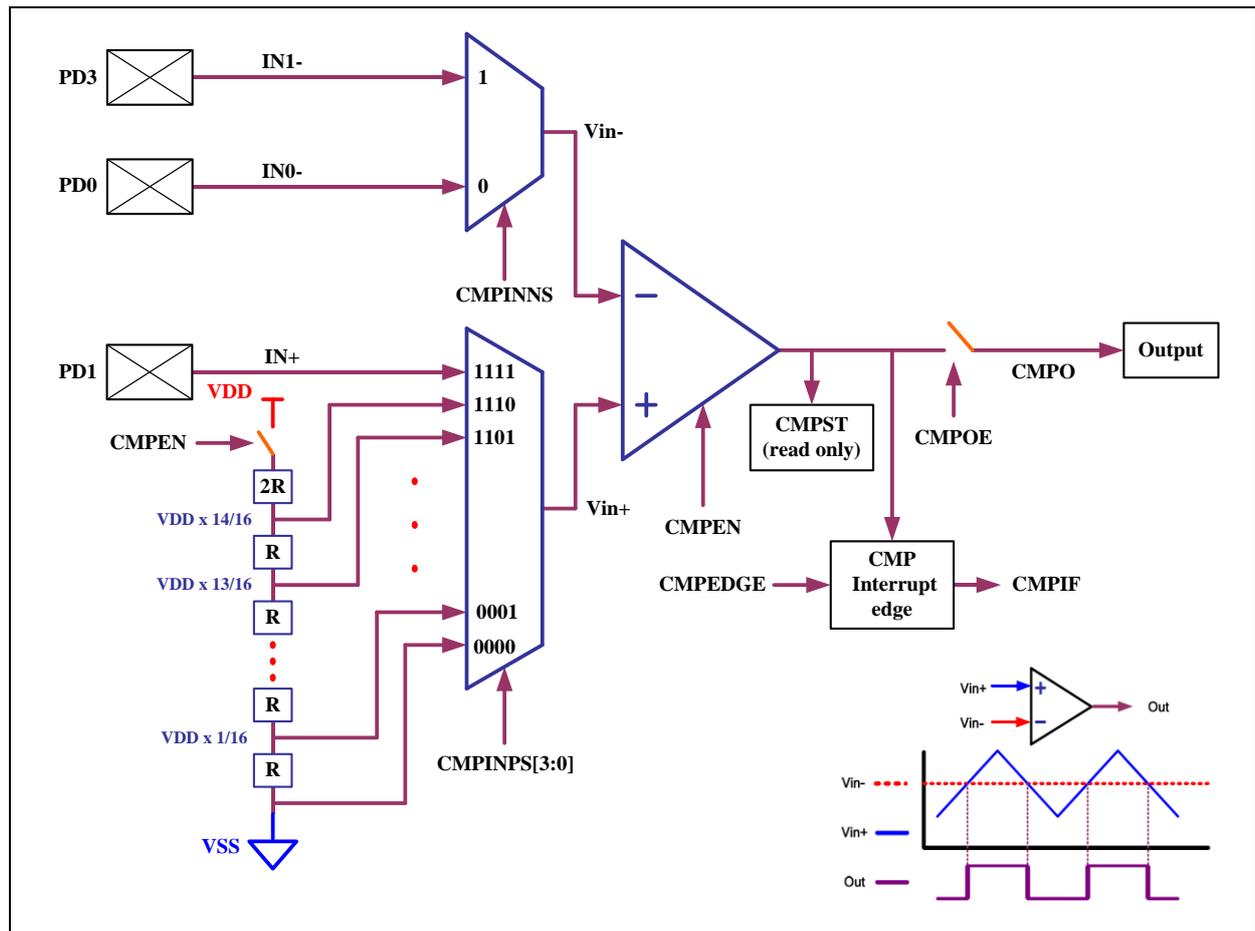
PWM 根据系统时钟(Fcpuclk)可以产生固定的 1024 占空比的波形。LSB 扩展技术使得 PWM 运行频率可以在『系统时钟除以 256』取代『系统时钟除以 1024』，意味着 PWM 速度是比正常的提高了 4 倍。PWM 比较高频率的优点是通过 RC 滤波器可以转换 PWM 讯号变成更稳定的直流电压值。当 8 位基数计数器符合 PWM 占空寄存器 PWMODUTY 的 8 位 MSB，PWM 输出讯号将复位至低电压。当计数器溢出后，PWM 占空寄存器的 2 位 LSB 决定是否设置 PWM 输出讯号立即变高或延迟一个时钟周期后才变高。





3.8 模拟比较器

TM57PE40 包含一个模拟比较器。此模拟比较器可通过在(R10h.7)的 CMPEN 使能。他将比较输入值位于正引脚 V_{in+} 和负引脚 V_{in-} 。当位于正引脚的电压高于负引脚的电压，模拟比较器输出 (CMPO)将被设置。CMPST 的输出状态可通过(F0Dh.0)读取或通过(R10h.5)的 CMPOE 输出至引脚。当输出状态改变，模拟比较器可以产生中断标志(CMPIF)。使用者可以选择中断触发于比较器输出上升或下降。负引脚的输入源可以通过 CMPINNS 选 IN0-或 IN1-。模拟比较器支持内部参考电压。内部参考电压提供输出电压范围以 15 不同的阶级。此范围可以通过 CMPINPS[3:0]选择。模拟比较器的模块图如下图所示。



比较器模块图

3.9 触摸按键

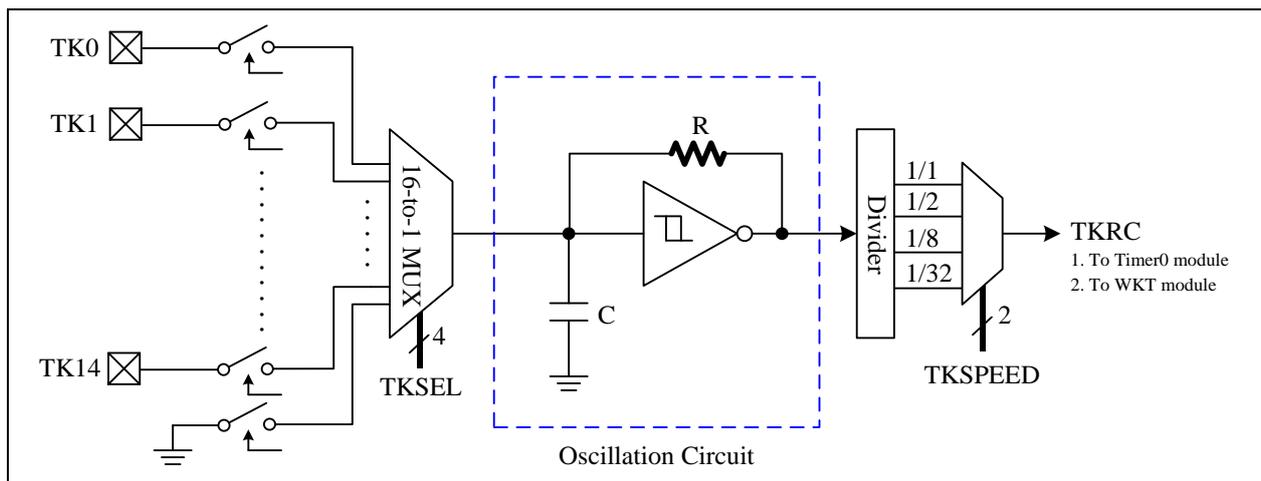
触摸按键将振荡时钟输出至 WKT 模块和 Timer0 模块。TKRC 至 Timer0 如同 TOI 输入。触摸按键模块包含一个 RC 振荡器，16-to-1 模拟输入选项，TKSPEED 控制位选择频率分频器输出。频率分频器将振荡时钟除以 1, 2, 8, 或 32。若 WKTKRCS 位为 1，被除的时钟将被送至 WKT 模块，而当 TMOTKS 位为 1，被除的时钟将被送至 Timer0 模块。Timer0 在上升或下降沿依据 TOIEDGE 位计数。

如果人类指尖靠近触摸板，C 的等效电容将增加，也就是说，振荡频率会降低。

基于上述论点，用户程序需要遵守：无论什么输入通道引起定时器 0 的计数值最小也只是在某一个固定的时间里，哪一个通道的按键被摸到或者手指接近引起的

要分清哪些通道计数的值是最低的，我们需要另一个定时器（定时器 1，定时器 2，或 WKT）来设置一个合适的间隔时间，因为定时器 0 计数不会溢出。在这个固定的间隔时间里，用户编程也就是触摸或接触按键来一个接一个地转换触摸键通道，并找到定时器 0 的最小计数值。

触摸按键振荡线路架构类似于 WKT，因此，若使用 WKT 模块计算触摸按键计数将得到最佳表现，独立的温度和电压系数。



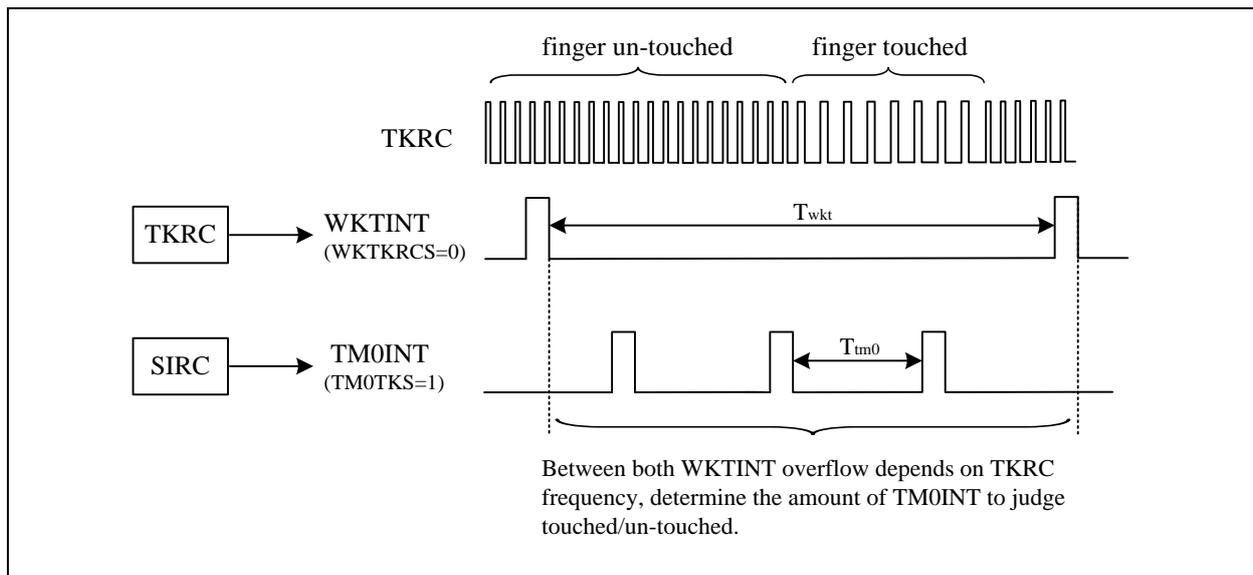
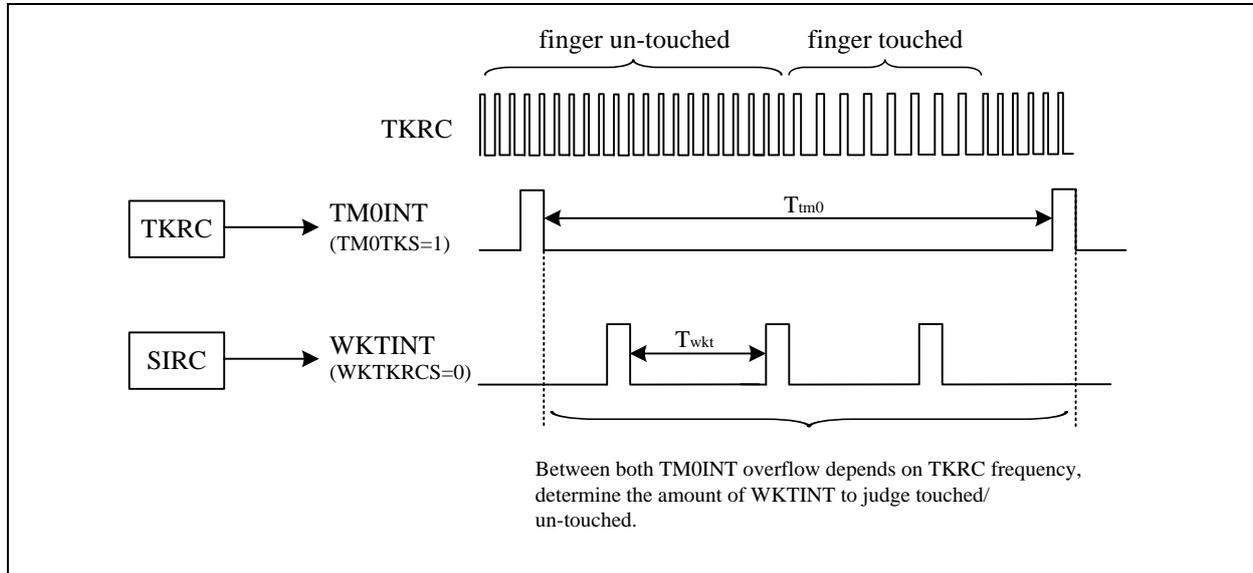
触摸按键振荡器的模块图

范例:

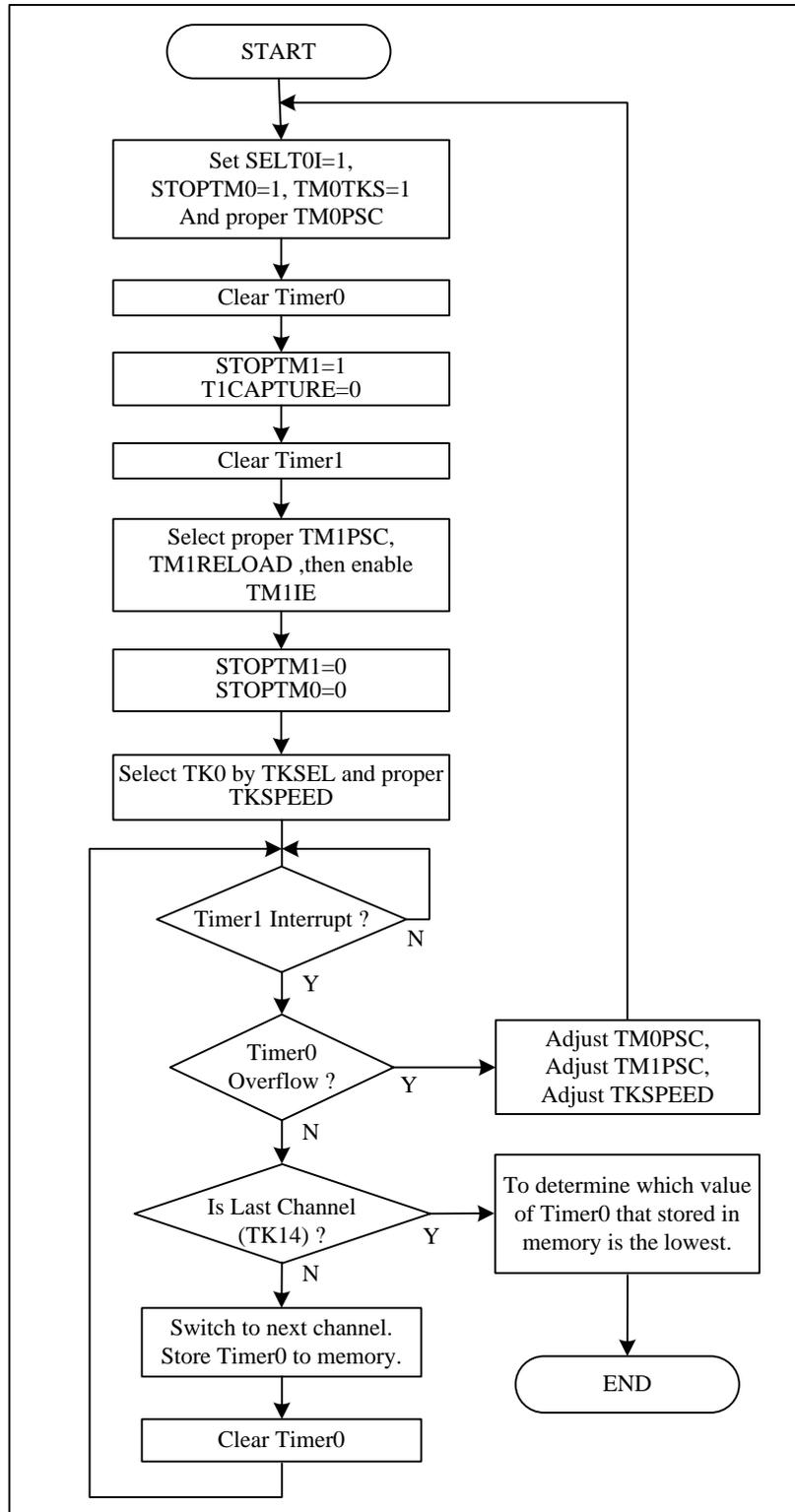
```

movlw    1000000b    ; 禁止 TK7(PB7)上拉电阻
movwr    nPBPU      ; R09h (R-Plane 地址 09h)
movlw    10000111b   ; WKTKRCS=1, WKT 定时器时钟源来自 TK7(PB7)引脚.
                    ; TMOTKS=0
                    ; TKSPEED=2'b00 (最高速度), TKSEL=4'b0111
movwr    TKCTL      ; R0Dh
movlw    01111111b   ; 禁止 PB7 数字输入。
movwr    PB_IE      ; R12h
    
```

3.9.1 触摸按键使用 Timer0 和 WKT 计数器



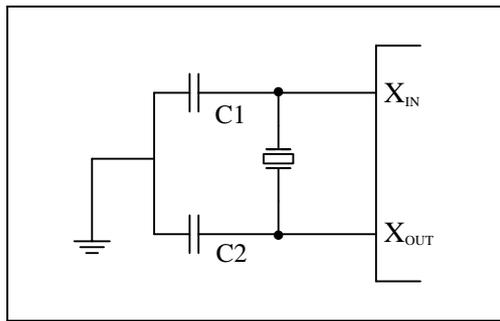
3.9.2 触摸按键使用 Timer0 和 Timer1 计数器



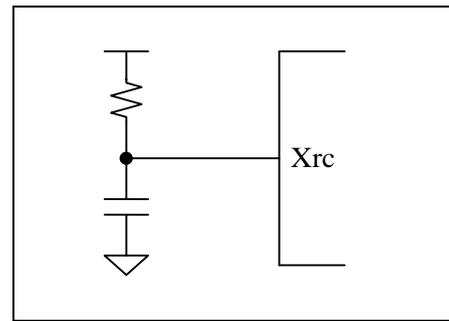
推荐的程序

3.10 系统时钟振荡器

系统时钟可以工作在四个不同的振荡模式中，可以通过 SYSCFG 寄存器中的 CLKS 位设置。在低速/高速晶振（SXT/FXT）模式，一个晶体或陶瓷谐振器连接到 X_{in} 和 X_{out} 引脚来做振荡器。在外部 RC（XRC）模式，外部电阻和电容决定振荡频率。在高速内部 RC（FIRC）模式，内置振荡器产生 16/8/4/2 MHz 振荡时钟，此时钟由寄存器 FIRCSEL[1:0] 位所控制。



外部振荡器线路
(晶体或陶瓷)

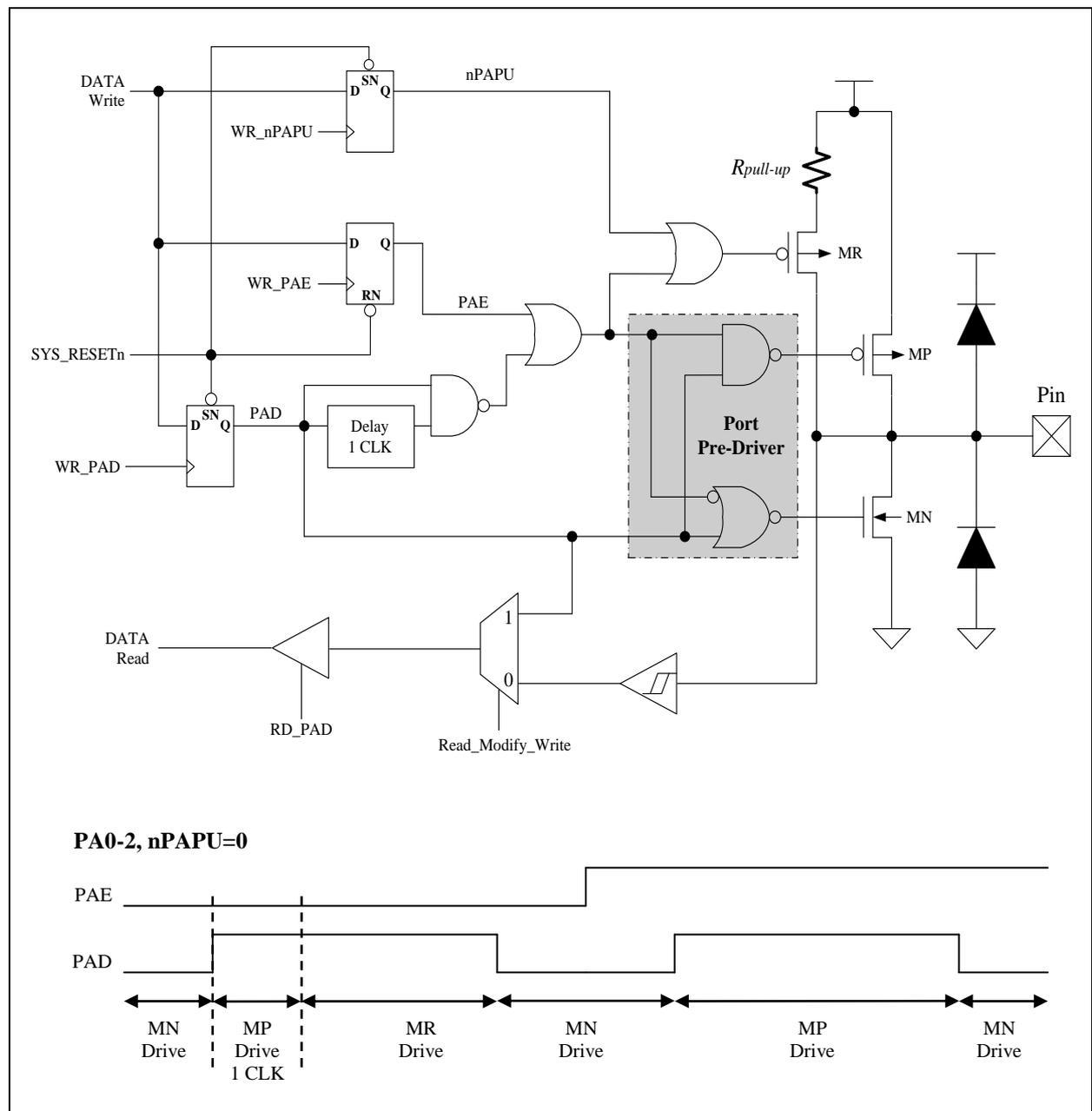


外部RC振荡器

4 I/O 端口

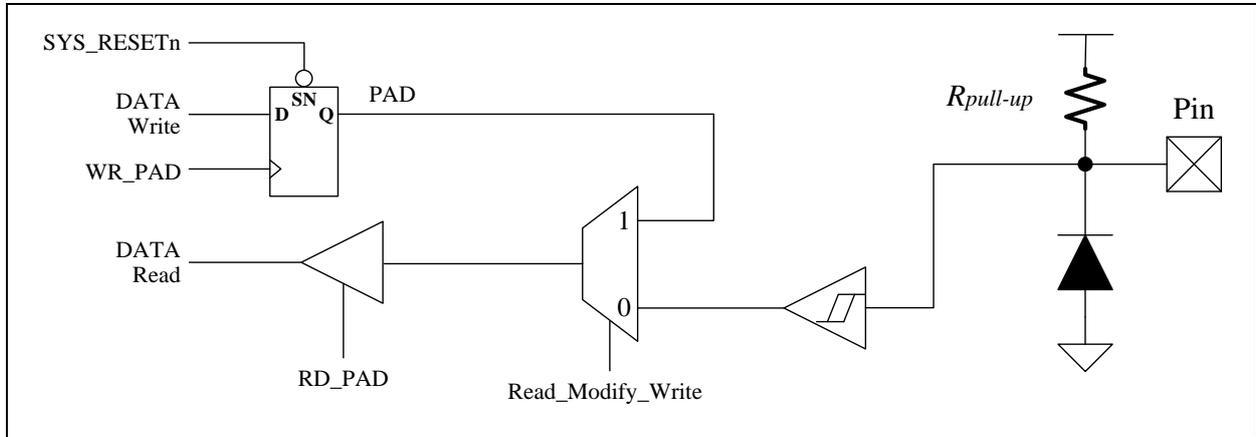
4.1 PA0-2

这 3 个引脚可被用做施密特触发输入，CMOS 推挽输出或『伪漏式』输出。分配给每个引脚的上拉电阻是由 S/W 设置。在施密特触发输入模式下，S/W 需设置 PAE=0 和 PAD=1。在『伪漏式』模式使用这些引脚，S/W 设置 PAE=0。伪漏式结构的优点是其输出上升时间可以大大快于纯漏式结构。S/W 设置 PAE=1 可使用 CMOS 推挽输出模式。读取数据引脚（PAD）有不同的意义。在“Read-Modify-Write”指令，CPU 其实读取输出数据寄存器。在其他指令，CPU 读取引脚状态。所谓“Read-Modify-Write”指令包含 BSF，BCF 和所有把 F-Plane 做目的地的指令。



4.3 PA7

PA7 只能用于施密特触发输入模式。内部上拉电阻总是保持连接。



内存功能图

F-Plane

名称	地址	R/W	复位值	描述
INDF	00.7~0	R/W	-	非物理寄存器，INDF 寻址实际上指向 FSR 寄存器内的地址的寄存器
TIMER0	01.7~0	R/W	0	Timer0 计数位
PC	02.7~0	R/W	0	程序计数器 [7~0]
GBIT0	03.7	R/W	0	通用位 0
GBIT1	03.6	R/W	0	一般寄存器位
RAMBANK	03.5	R/W	0	RAM Bank 选项
TO	03.4	R	0	WDT 溢出标志，可由 PWRST，‘SLEEP’或‘CLRWDT’指令清零
PD	03.3	R	0	SLEEP 模式标志，由‘SLEEP’设置，‘CLRWDT’指令清零
ZFLAG	03.2	R/W	0	零标志
DCFLAG	03.1	R/W	0	辅助进位或借位标志
CFLAG	03.0	R/W	0	进位标志位
GBIT2	04.7	R/W	0	一般寄存器位
FSR	04.6~0	R/W	-	文件选择寄存器，间接寻址指标
PAD7	05.7	R	-	PA7 引脚状态位
PAD	05.6~0	R	-	Port A 引脚或“数据寄存器”状态
		W	7F	Port A 输出数据寄存器
PBD	06.7~0	R	-	Port B 引脚或“数据寄存器”状态
		W	FF	Port B 输出数据寄存器
PDD	07.7~0	R	-	Port D 引脚或“数据寄存器”状态
		W	FF	Port D 输出数据寄存器
CMPIE	08.7	R/W	0	比较器中断使能，1=使能，0=禁止
TM2IE	08.6	R/W	0	Timer2 中断使能，1=使能，0=禁止
TM1IE	08.5	R/W	0	Timer1 中断使能，1=使能，0=禁止
TM0IE	08.4	R/W	0	Timer0 中断使能，1=使能，0=禁止
WKTIE	08.3	R/W	0	唤醒定时器中断使能，1=使能，0=禁止 设为 0 清零且禁止 WKT 定时器
XINT2E	08.2	R/W	0	INT2 引脚(PA2) 中断使能，1=使能，0=禁止
XINT1E	08.1	R/W	0	INT1 引脚(PA1) 中断使能，1=使能，0=禁止
XINT0E	08.0	R/W	0	INT0 引脚(PA0) 中断使能，1=使能，0=禁止
CMPIF	09.7	R	-	比较器中断事件延迟标志，通过 CMP 设置改变输出结果
		W	0	0: 清零此标志，1: 无效操作
TM2IF	09.6	R	-	Timer2 中断事件延迟标志，当 Timer2 溢出由 H/W 设置
		W	0	0: 清零此标志，1: 无效操作
TM1IF	09.5	R	-	Timer1 中断事件延迟标志，当 Timer1 溢出由 H/W 设置
		W	0	0: 清零此标志，1: 无效操作
TM0IF	09.4	R	-	Timer0 中断事件延迟标志，当 Timer0 溢出由 H/W 设置
		W	0	0: 清零此标志，1: 无效操作

名称	地址	R/W	复位值	描述
WKTIF	09.3	R	-	WKT 中断事件延迟标志, 当 WKT 溢出由 H/W 设置
		W	0	0: 清零此标志, 1: 无效操作
XINT2IF	09.2	R	-	INT2 中断事件延迟标志, 当 INT2 引脚下降沿由 H/W 设置
		W	0	0: 清零此标志, 1: 无效操作
XINT1IF	09.1	R	-	INT1 中断事件延迟标志, 当 INT1 引脚下降沿由 H/W 设置
		W	0	0: 清零此标志, 1: 无效操作
XINT0IF	09.0	R	-	INT0 中断事件延迟标志, 当 INT0 引脚下降/上升沿由 H/W 设置
		W	0	0: 清零此标志, 1: 无效操作
TM1L	0a.7~0	R	-	Timer1 计算低位
		W	0	Timer1 重载低位数据
TM1H	0b.7~0	R	-	Timer1 计算高位
		W	0	Timer1 重载高位数据
PWMADTH	0c.7~0	R/W	0	PWMA 占空 8 位 MSB
PWMADTL	0d.7~6	R/W	0	PWMA 占空 8 位 LSB
-	0d.5~1	-	-	保留
CMPST	0d.0	R	-	比较器比较结果输出
PWM0DUTY	0e.7~0	R/W	0	PWM0 占空 8 位
-	0f.7	-	-	保留
SIRCSEL	0f.6~5	R/W	3	SIRC 时钟选项 00:128K 01:32K 10:8K 11:2 KHz
STPFCK	0f.4	R/W	0	1: 停止快时钟振荡 0: 使能快时钟振荡
SELSUB	0f.3	R/W	0	1: 强迫快时钟振荡且设置慢时钟为 CPUCLK 0: 选择快时钟为 CPUCLK
SUBE	0f.2	R/W	0	若 SELSUB=1, 此 SUBE 位为无效, 慢时钟持续振荡 若 SELSUB=0, 设置 1 使能慢时钟振荡, 清为 0 停止慢时钟振荡
SUBTYP	0f.1~0	R/W	0	慢时钟类别 00:SXT 01:SIRC 10:XRC 11:TKRC
PED	10.4~0	R	-	Port E 引脚或“数据寄存器”状态
		W	1F	Port E 输出数据寄存器
-	11.7~0	-	-	保留
-	12.7	-	-	保留
KICKE	12.6	R/W	1	加速 SXT 暖机, 清为 0 在 SXT 振荡稳定后进入省电阶段
CLRPWM0	12.5	R/W	1	PWM0 清零和保留
CLRTM2	12.4	R/W	1	Timer2 清零和保留当此位为"1", 预设 Timer2 为无动作
SETTM1	12.3	R/W	0	Timer1 设置 FFFFh 并且当此位为"1"时则保留
CLRTM1	12.2	R/W	0	清零 Timer1 内容且当此位为"1"时则保留
STOPTM1	12.1	R/W	0	停止 Timer1 计数当此位为"1"时
STOPTM0	12.0	R/W	0	停止 Timer0 计数当此位为"1"时
SRAM	20~2f	R/W	-	RAM 公共区域(16 位)
	30~7F	R/W	-	RAM BANK0 区域(RAMBANK=0, 80 位)
	30~7F	R/W	-	RAM BANK1 区域(RAMBANK=1, 80 位)

R-Plane

名称	地址	R/W	复位值	描述
TM0PSC (R02h)				
CAPOLARITY	02.7	W	0	Timer0 捕获模式电位极性 0: 高电位捕获 1: 低电位捕获
TOCAPTURE	02.6	W	0	Timer0 模式 0: 定时器/计数器模式 (来源从 TM0PSC 输出或 TOCKI 或 TKRC) 1: 捕获模式(来源从 CAPT 引脚)
TOIEDGE	02.5	W	0	Timer0 预分频计数沿用于 TOCKI / TKRC 引脚 0: 上升沿提升 Timer0 预分频计数 1: 下降沿提升 Timer0 预分频计数
SELTOI	02.4	W	0	Timer0 预分频时钟源 0: 时钟源为 Fcpuclk/2 (指令周期) 1: 时钟源为 TOCKI 引脚 (PE2 引脚) 或 TKRC (参考 R-plane 0dh.6)
TM0PSC	02.3~0	W	0	Timer0 预分频. Timer0 预分频时钟源除以 0000: 1 0001: 2 0010: 4 0011: 8 0100: 16 0101: 32 0110: 64 0111: 128 1xxx: 256
PWRDOWN	03	W	-	写入此寄存器将进入终止/空闲模式 (即‘SLEEP’指令)
CLRWDWT	04	W	-	写入此寄存器将清零 WDT 定时器
PAE	05.6~3	W	0	个别位控制对应的引脚, 若位是: 0: 引脚为开漏式输出或施密特触发输入 1: 引脚为 CMOS 推挽输出
	05.2~0	W	0	个别位控制对应的引脚, 若位是: 0: 引脚为伪开漏式输出或施密特触发输入 1: 引脚为 CMOS 推挽输出
PBE	06.7~0	W	0	个别位控制对应的引脚, 若位是: 0: 引脚为开漏式输出或施密特触发输入 1: 引脚为 CMOS 推挽输出
PDE	07.7~0	W	0	个别位控制对应的引脚, 若位是: 0: 引脚为开漏式输出或施密特触发输入 1: 引脚为 CMOS 推挽输出
nPAPU	08.6~0	W	7F	个别位控制对应的引脚, 若位是: 0: 使能上拉电阻的引脚, 除了 a. 当引脚的输出数据寄存器 (PAD) 为 0 b. 当引脚的 CMOS 拉挽模式被选定 (PAE=1) c. 此引脚运行在 FXT/SXT/XRC/PWM/TM0TGL 输出 1: 引脚上拉电阻被禁止
nPBPU	09.7~0	W	FF	个别位控制对应的引脚, 若位是: 0: 使能上拉电阻的引脚, 除了 a. 当引脚的输出数据寄存器 (PBD) 为 0 b. 当引脚的 CMOS 拉挽模式被选定 (PBE=1) 1: 引脚上拉电阻被禁止

名称	地址	R/W	复位值	描述
nPDPU	0a.7~0	W	FF	个别位控制对应的引脚，若位是： 0: 使能上拉电阻的引脚，除了 a. 当引脚的输出数据寄存器（PDD）为 0 b. 当引脚的 CMOS 拉挽模式被选定（PDE=1） c. 此引脚运行在 PWM/CMPO/TM1TGL/TCOUT 输出 1: 引脚上拉电阻被禁止
WKTPSC (R0Bh)				
PWM0PSC	0b.7~6	W	0	PWM0 预分频 0:Fcpuclk 1:Fcpuclk/2 2:Fcpuclk/4 3:Fcpuclk/8
PWMAE	0b.5	W	0	PWMA 输出至 PA1/PD7 引脚使能，参考 R-plane 17h.1
PWM0E	0b.4	W	0	PWM0 输出至 PA5/PD6 引脚使能，参考 R-plane 17h.0
PWM0INV	0b.3	W	0	PWM0 负输出值至引脚 1:负值 0:正值
TM0OUT	0b.2	W	0	使能 Timer0 溢出触发输出至 PA6 引脚
WKTPSC	0b.1~0	W	3	WKT 周期 00=0.9 ms, 01=1.8 ms, 10=25 ms, 11=100 ms (VDD=5V) 00=1.0 ms, 01=2.0 ms, 10=32 ms, 11=128 ms (VDD=3V)
TM1PSC (R0Ch)				
TCOPSC	0c.7~6	W	0	CPUCLK 预分频 0:Fcpuclk/2 1:Fcpuclk/4 3:Fcpuclk/8 3:Fcpuclk/16
TCOE	0c.5	W	0	使能后预分频器 CPUCLK 输出至 PD5 引脚 (TCOUT)
-	0c.4	-	-	保留
INT1EDGE	0c.3	W	0	0: INT1 引脚下降沿触发中断事件 1: INT1 引脚上升沿触发中断事件
TM1OE	0c.2	W	0	使能 Timer1 溢出触发输出至 PD0 引脚 (TM1TGL)
T1CAPTURE	0c.1	W	0	Timer1 模式 0:Timer 模式 (来源自 TM1PSC 时钟输出) 1:捕获模式(来源自 CAPT 引脚) 量测 CAPT 引脚周期时间介于连续的上升沿和下降沿
TM1PSC	0c.0	W	0	Timer1 预分频. 1: Fcpuclk 0:Fcpuclk/2
TKCTL (R0Dh)				
WTKRCS	0d.7	W	0	WKT 定时器时钟选项 0:SIRC 1:TKRC
TM0TKS	0d.6	W	0	Timer0 时钟选项 0:T0CKI 引脚或 Fcpuclk/2 1:TKRC
TKSPEED	0d.5~4	W	0	触摸按键振荡频率选项, TKRC 分频器 0: 最高速触摸按键时钟(div1) 1:div2 2:div8 3: 最低速触摸按键时钟(div32)
TKSEL	0d.3~0	W	F	触摸按键信道选项 0000: TK0 0001: TK1 1110: TK14 1111: No channel
TM2PSC (R0Eh)				
WDT_PSC	0e.7~6	W	1	WDT 溢出. 0:112 ms 1:224 ms 2:896 ms 3:1800 ms
WDTSLPSTP	0e.5	W	0	1: WDT 停止计数当闲置/省电模式 0:WDT 总是计数 若 WDTE=0, 此 WDTSLPSTP 位为失效
TM2CLKS	0e.4	W	0	Timer2 时钟源. 1:Fcpuclk/128 0:慢时钟
FIRCSEL	0e.3~2	W	2	FIRC 时钟选项. 3:16 MHz 2:4 MHz 1:8 MHz 0:2 MHz

名称	地址	R/W	复位值	描述
TM2PSC	0e.1~0	W	0	Timer2 预分频. Timer2 时钟源除以- 0:32768 1:16384 2:8192 3:128
TESTREG	0f.7~0	W	0	维持 0, 勿写入
CMPCTL (R10h)				
CMPEM	10.7	W	0	比较器使能
CMPEGE	10.6	W	0	比较器中断沿. 0:下降沿 1:上升沿
CMPOE	10.5	W	0	比较器输出至 PD2 引脚使能
CMPIINS	10.4	W	0	比较器负值输入来源选项 1:IN1-(PD3) 0:IN0-(PD0)
CMPIINS	10.3~0	W	0	比较器正值输入来源选项 0000:Vref0 0001:Vref1 0010:Vref2 1110:Vref14 1111:IN+(PD1)
PWM0PROD	11.7~0	W	FF	PWM0 周期
PB_IE	12.7~0	W	FF	每位控制其对应引脚 0: 禁止 I/O 数字输入以达省电并且于模拟做多任务 1: 使能 I/O 数字输入
PD_IE	13.7~0	W	FF	每位控制其对应引脚 0: 禁止 I/O 数字输入以达省电并且于模拟做多任务 1: 使能 I/O 数字输入
PBWKUP	14.7~0	W	0	1: PB7~0 wake up enable 0:disable
PEE	15.4~0	W	0	每位控制其对应引脚, 若此位为 0: 引脚为开漏式输出或施密特触发输入 1: 引脚为 CMOS 推挽输出
nPEPU	16.4~0	W	FF	每位控制其对应引脚, 若此位为 0: 引脚上拉电阻使能, 除非 a. 引脚输出数据寄存器(PED)为 0 b. 引脚 CMOS 推挽模式被选定(PEE=1) 1: 引脚上拉电阻被禁止
PWMA0S (R17h)				
-	17.7~2	-	-	保留
PWMA_PD7	17.1	W	0	PWMA 输出信道选项 0:PA1 1:PD7
PWM0_PD6	17.0	W	0	PWM0 输出信道选项 0:PA5 1:PD6

指令集

每个指令都是一个 14 位字被分开成一个 Op-Code，他详细的阐述了指令类型，一个或更多的能进一步详解指令运行的操作数。指令在下面的表格中被分类为字节导向，位向导和文字操作列表。

对字节指令来说，“f”或“r”代表指示地址，“d”代表指示目的单元格。地址指示被用来指定程序存储器的哪位地址被指令使用。目的单元格指示出操作结果被放置的位置。如果“d”是 0，结果被放在 W 寄存器，否则结果被放在指令指示的地址中。

对位指示指令来说，“b”代表位指示，他用来选择受操作影响的位，而“f”代表地址指示。对于文字操作，立即数 k 代表文字或常数。

简记符号	描述
f	F-Plane 寄存器或RAM
r	R-Plane 寄存器或RAM
b	位地址
k	立即数k, 常数或标号
d	目的选择项。0 : 工作寄存器 1 : F-Plane寄存器或RAM
W	工作寄存器
Z	零标志
C	进位标志
DC	辅助进位标志
PC	程序计数器
TOS	堆栈顶
GIE	全局中断使能标志 (i-Flag)
[]	可选项
()	内容
.	比特项
B	之前
A	之后
←	赋值方向

助记符	操作码	指令周期	影响标志	说明	
字节数据寄存器操作指令					
<u>ADDWF</u>	f,d	00 0111 dfff ffff	1	C,DC,Z	W 和"f"相加→W 或"f"
<u>ANDWF</u>	f,d	00 0101 dfff ffff	1	Z	W 和"f"相与→W 或"f"
<u>CLRF</u>	f	00 0001 1fff ffff	1	Z	"f"清零
<u>CLRWF</u>		00 0001 0100 0000	1	Z	W 清零
<u>COMF</u>	f,d	00 1001 dfff ffff	1	Z	"f"反取→W 或"f"
<u>DECWF</u>	f,d	00 0011 dfff ffff	1	Z	"f"减一→W 或"f"
<u>DECFSZ</u>	f,d	00 1011 dfff ffff	1 or 2	-	"f"减一→W 或"f"，结果为 0 就间跳
<u>INCF</u>	f,d	00 1010 dfff ffff	1	Z	"f"加一→W 或"f"
<u>INCFSZ</u>	f,d	00 1111 dfff ffff	1 or 2	-	"f"加一→W 或"f"，结果为 0 就间跳
<u>IORWF</u>	f,d	00 0100 dfff ffff	1	Z	W 和"f"相或→W 或"f"
<u>MOVWF</u>	f	00 1000 0fff ffff	1	-	W ← "f"
<u>MOVWF</u>	f	00 0000 1fff ffff	1	-	"f" ← W
<u>MOVWR</u>	r	00 0000 00rr rrrr	1	-	"r" ← W
<u>RLF</u>	f,d	00 1101 dfff ffff	1	C	"f"带进位左移→W 或"f"
<u>RRF</u>	f,d	00 1100 dfff ffff	1	C	"f"带进位右移→W 或"f"
<u>SUBWF</u>	f,d	00 0010 dfff ffff	1	C,DC,Z	"f" - W → W 或"f"
<u>SWAPF</u>	f,d	00 1110 dfff ffff	1	-	"f"的高低半字节互换→W 或"f"
<u>TESTZ</u>	f	00 1000 1fff ffff	1	Z	测试"f"是否为 0 → W 或"f"
<u>XORWF</u>	f,d	00 0110 dfff ffff	1	Z	W 和"f"相导或→W 或"f"
位数据寄存器操作指令					
<u>BCF</u>	f,b	01 000b bbff ffff	1	-	"f"的"b"位清零
<u>BSF</u>	f,b	01 001b bbff ffff	1	-	"f"的"b"位置设 1
<u>BTFSC</u>	f,b	01 010b bbff ffff	1 or 2	-	"f"的"b"位若为 0，间跳
<u>BTFSS</u>	f,b	01 011b bbff ffff	1 or 2	-	"f"的"b"位若为 1，间跳
文字和控制指令					
<u>ADDLW</u>	k	01 1100 kkkk kkkk	1	C,DC,Z	立即数 k 和 W 相加
<u>ANDLW</u>	k	01 1011 kkkk kkkk	1	Z	立即数 k 和 W 相与
<u>CALL</u>	k	10 kkkk kkkk kkkk	2	-	调用子程序，子程序入口地址（标号）为 k
<u>CLRWDI</u>		00 0000 0000 0100	1	TO,PD	看门狗清零
<u>GOTO</u>	k	11 kkkk kkkk kkkk	2	-	跳转指令，目的地址（标号）为 k
<u>IORLW</u>	k	01 1010 kkkk kkkk	1	Z	立即数 k 和 W 相或
<u>MOVLW</u>	k	01 1001 kkkk kkkk	1	-	立即数 k → W
<u>NOP</u>		00 0000 0000 0000	1	-	空指令
<u>RET</u>		00 0000 0100 0000	2	-	从子程序返回
<u>RETI</u>		00 0000 0110 0000	2	-	从中断返回
<u>RETLW</u>	k	01 1000 kkkk kkkk	2	-	带立即数 k 返回，返回值在 W 中
<u>SLEEP</u>		00 0000 0000 0011	1	TO,PD	进入待机模式，时钟振荡停止
<u>XORLW</u>	k	01 1111 kkkk kkkk	1	Z	立即数 k 和 W 相导或

ADDLW 立即数立即数k和W相加

语法	ADDLW k	
操作数	k : 00h ~ FFh	
运算	$(W) \leftarrow (W) + k$	
影响的状态位	C, DC, Z	
OP-Code	01 1100 kkkk kkkk	
描述	W寄存器中的内容和8位'k'值相加，结果放在W寄存器中。	
周期	1	
范例	ADDLW 0x15	B : W = 0x10 A : W = 0x25

ADDWF W和'f'相加

语法	ADDWF f [,d]	
操作数	f : 00h ~ 7Fh, d : 0, 1	
运算	(目标寄存器: destination) $\leftarrow (W) + (f)$	
影响的状态位	C, DC, Z	
OP-Code	00 0111 dfff ffff	
描述	W寄存器的内容和寄存器的内容和'f'相加。若'd'为0，结果放在W中，若'd'为1，结果放在'f'中。	
周期	1	
范例	ADDWF FSR, 0	B : W = 0x17, FSR = 0xC2 A : W = 0xD9, FSR = 0xC2

ANDLW 立即数 k 和 W 逻辑与

语法	ANDLW k	
操作数	k : 00h ~ FFh	
运算	$(W) \leftarrow (W) \text{ AND } k$	
影响的状态位	Z	
OP-Code	01 1011 kkkk kkkk	
描述	W寄存器中的内容和8位'k'值相与运算。结果放在W寄存器中。	
周期	1	
范例	ANDLW 0x5F	B : W = 0xA3 A : W = 0x03

ANDWF W和'f'相逻辑与

语法	ANDWF f [,d]	
操作数	f : 00h ~ 7Fh, d : 0, 1	
运算	(destination) $\leftarrow (W) \text{ AND } (f)$	
影响的状态位	Z	
OP-Code	00 0101 dfff ffff	
描述	W寄存器中的内容和'f'寄存器相与。若'd'为0，结果放在W寄存器中。若'd'为1，结果放在'f'寄存器中。	
周期	1	
范例	ANDWF FSR, 1	B : W = 0x17, FSR = 0xC2 A : W = 0x17, FSR = 0x02

BCF "f"的"b"位清零

语法	BCF f [,b]	
操作数	f : 00h ~ 3Fh, b : 0 ~ 7	
运算	(f.b) ← 0	
影响的状态位	-	
OP-Code	01 000b bbff ffff	
描述	寄存器'f'的'b'位被清零	
周期	1	
范例	BCF FLAG_REG, 7	B : FLAG_REG = 0xC7 A : FLAG_REG = 0x47

BSF "f"的"b"位设置 1

语法	BSF f [,b]	
操作数	f : 00h ~ 3Fh, b : 0 ~ 7	
运算	(f.b) ← 1	
影响的状态位	-	
OP-Code	01 001b bbff ffff	
描述	寄存器'f'的'b'位被设置 1	
周期	1	
范例	BSF FLAG_REG, 7	B : FLAG_REG = 0x0A A : FLAG_REG = 0x8A

BTFSC "f"的"b"位若为 0, 间跳

语法	BTFSC f [,b]	
操作数	f : 00h ~ 3Fh, b : 0 ~ 7	
运算	Skip next instruction if (f.b) = 0	
影响的状态位	-	
OP-Code	01 010b bbff ffff	
描述	若寄存器'f'的'b'位为 1, 下一条指令被执行。否则下一条指令被放弃, 一个 NOP 指令被执行, 使这条指令要两个指令周期。	
周期	1 or 2	
范例	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 "f"的"b"位若为 1, 间跳

语法	BTFSS f [,b]	
操作数	f : 00h ~ 3Fh, b : 0 ~ 7	
运算	若(f.b) = 1, 忽略下一个指令	
影响的状态位	-	
OP-Code	01 011b bbff ffff	
描述	若寄存器'f'的'b'位为 0, 下一条指令被执行。否则下一条指令被放弃, 一个 NOP 指令被执行, 使这条指令要两个指令周期。	
周期	1 or 2	
范例	LABEL1 BTFSS FLAG, 1 TRUE GOTO SUB1 FALSE ...	B : PC = LABEL1 A : if FLAG.1 = 0, PC = TRUE if FLAG.1 = 1, PC = FALSE

COMF
"f"取反

语法	COMF f [,d]
操作数	f : 00h ~ 7Fh, d : 0, 1
运算	(destination) ← (\bar{f})
影响的状态位	Z
OP-Code	00 1001 dfff ffff
描述	寄存器'f'的值做补码运算后, 运算结果放在目标寄存器中。若"d"为 0, 结果放在 W 中。若"d"为 1, 结果放在'f'中。
周期	1
范例	COMF REG1, 0 B : REG1 = 0x13 A : REG1 = 0x13, W = 0xEC

DECF
"f"递减 1

语法	DECF f [,d]
操作数	f : 00h ~ 7Fh, d : 0, 1
运算	(destination) ← (f) - 1
影响的状态位	Z
OP-Code	00 0011 dfff ffff
描述	寄存器'f'的值递减 1。运算结果放在目标寄存器中。若"d"为 0, 结果放在 W 中。若"d"为 1, 结果放在'f'中。
周期	1
范例	DECF CNT, 1 B : CNT = 0x01, Z = 0 A : CNT = 0x00, Z = 1

DECFSZ
"f"递减 1, 结果为 0 则跳

语法	DECFSZ f [,d]
操作数	f : 00h ~ 7Fh, d : 0, 1
运算	(destination) ← (f) - 1, 若结果为 0 时, 忽略下一个指令
影响的状态位	-
OP-Code	00 1011 dfff ffff
描述	寄存器'f'的值递减 1。若'd'为 0, 结果放在 W 寄存器。若'd'为 1, 结果放在'f'寄存器中。若结果为 1, 下一个指令会继续执行。若结果为 0, 一个 NOP 指令会被取代执行, 此时会需要两个指令周期的时间。
周期	1 or 2
范例	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
强行跳转

语法	GOTO k	
操作数	k : 000h ~ FFFh	
运算	PC.11~0 ← k	
影响的状态位	-	
OP-Code	11 kkkk kkkk kkkk	
描述	GOTO 是无条件转移指令。12 位的立即值会被载入 PC <11:0>的位置中。GOTO 指令占用两个指令周期。	
周期	2	
范例	LABEL1 GOTO SUB1	B : PC = LABEL1 A : PC = SUB1

INCF
'f'递增

语法	INCF f [,d]	
操作数	f : 00h ~ 7Fh	
运算	(destination) ← (f) + 1	
影响的状态位	Z	
OP-Code	00 1010 dfff ffff	
描述	寄存器'f'的值递增 1。若'd'为 0，结果放在 W 寄存器。若'd'为 1，结果放在'f'寄存器中。	
周期	1	
范例	INCF CNT, 1	B : CNT = 0xFF, Z = 0 A : CNT = 0x00, Z = 1

INCFSZ
'f'递增，结果为 0 则跳

语法	INCFSZ f [,d]	
操作数	f : 00h ~ 7Fh, d : 0, 1	
运算	(destination) ← (f) + 1,若结果为 0 时，忽略下一个指令	
影响的状态位	-	
OP-Code	00 1111 dfff ffff	
描述	寄存器'f'的值递增 1。若'd'为 0，结果放在 W 寄存器。若'd'为 1，结果放在'f'寄存器中。若结果为 1，下一个指令会继续执行。若结果为 0，一个 NOP 指令会被取代执行，此时会需要两个指令周期的时间。	
周期	1 or 2	
范例	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	W 和立即数 k 逻辑或	
语法	IORLW k	
操作数	k : 00h ~ FFh	
运算	(W) ← (W) OR k	
影响的状态位	Z	
OP-Code	01 1010 kkkk kkkk	
描述	W 寄存器的值和'k'的值做 OR 运算。结果放在 W 寄存器中。	
周期	1	
范例	IORLW 0x35	B : W = 0x9A A : W = 0xBF, Z = 0
IORWF	W 和'f'逻辑或	
语法	IORWF f [,d]	
操作数	f : 00h ~ 7Fh, d : 0, 1	
运算	(destination) ← (W) OR k	
影响的状态位	Z	
OP-Code	00 0100 dfff ffff	
描述	包含 W 寄存器和'f'寄存器值相或。若'd'为 0, 结果放在 W 寄存器。若'd'为 1, 结果放在'f'寄存器。	
周期	1	
范例	IORWF RESULT, 0	B : RESULT = 0x13, W = 0x91 A : RESULT = 0x13, W = 0x93, Z = 0
MOVFW	'f'内容取出送到 W	
语法	MOVFW f	
操作数	f : 00h ~ 7Fh	
运算	(W) ← (f)	
影响的状态位	-	
OP-Code	00 1000 0fff ffff	
描述	把'f'寄存器中的值移到 W 寄存器中	
周期	1	
范例	MOVFW FSR	B : FSR = 0xC2, W = ? A : FSR = 0xC2, W = 0xC2
MOVLW	W 置立即数 k	
语法	MOVLW k	
操作数	k : 00h ~ FFh	
运算	(W) ← k	
影响的状态位	-	
OP-Code	01 1001 kkkk kkkk	
描述	8 位'k'载入 W 寄存器中。忽略的值汇集成 0。	
周期	1	
范例	MOVLW 0x5A	B : W = ? A : W = 0x5A

MOVWF	把 W 内容移动到"f"	
语法	MOVWF f	
操作数	f : 00h ~ 7Fh	
运算	(f) ← (W)	
影响的状态位	-	
OP-Code	00 0000 1fff ffff	
描述	W 寄存器值移到f寄存器中	
周期	1	
范例	MOVWF REG1	B : REG1 = 0xFF, W = 0x4F A : REG1 = 0x4F, W = 0x4F
MOVWR	把 W 内容移动到"r"	
语法	MOVWR r	
操作数	r : 00h ~ 3Fh	
运算	(r) ← (W)	
影响的状态位	-	
OP-Code	00 0000 00rr rrrr	
描述	W 寄存器值移到r寄存器中	
周期	1	
范例	MOVWR REG1	B : REG1 = 0xFF, W = 0x4F A : REG1 = 0x4F, W = 0x4F
NOP	空操作	
语法	NOP	
操作数	-	
运算	空操作	
影响的状态位	-	
OP-Code	00 0000 0000 0000	
描述	空操作	
周期	1	
范例	NOP	-
RET	子程序返回	
语法	RET	
操作数	-	
运算	PC ← TOS	
影响的状态位	-	
OP-Code	00 0000 0100 0000	
描述	从子程序返回。堆栈顶端的值会被弹出放在 PC。这是两个周期的指令。	
周期	2	
范例	RET	A : PC = TOS

SWAPF	"f"高低 4 位内容互换
语法	SWAPF f [,d]
操作数	f : 00h ~ 7Fh, d : 0, 1
运算	(destination,7~4) ← (f.3~0), (destination.3~0) ← (f.7~4)
影响的状态位	-
OP-Code	00 1110 dfff ffff
描述	寄存器'f'的高低半位值互换。运算结果放在目标寄存器中。若"d"为 0, 结果放在 W 中。若"d"为 1, 结果放在'f'中。
周期	1
范例	SWAPF REG1, 0 B : REG1 = 0xA5 A : REG1 = 0xA5, W = 0x5A

TESTZ	检测"f"是否为零
语法	TESTZ f
操作数	f : 00h ~ 7Fh
运算	Set Z flag if (f) is 0
影响的状态位	Z
OP-Code	00 1000 1fff ffff
描述	若寄存器 f 的内容是 0, 零标志被设置为 1。
周期	1
范例	TESTZ REG1 B : REG1 = 0, Z = ? A : REG1 = 0, Z = 1

XORLW	W 和立即数 k 逻辑导或
语法	XORLW k
操作数	k : 00h ~ FFh
运算	(W) ← (W) XOR k
影响的状态位	Z
OP-Code	01 1111 kkkk kkkk
描述	'k'的值和 W 寄存器的值做 XOR 运算。运算结果放进 W 寄存器中。
周期	1
范例	XORLW 0xAF B : W = 0xB5 A : W = 0x1A

XORWF	W 和"f"逻辑导或
语法	XORWF f [,d]
操作数	f : 00h ~ 7Fh, d : 0, 1
运算	(destination) ← (W) XOR (f)
影响的状态位	Z
OP-Code	00 0110 dfff ffff
描述	'f'寄存器的值和 W 寄存器的值做 XOR 运算。运算结果存于目标寄存器中。若"d"为 0, 结果放在 W 中。若"d"为 1, 结果放在'f'中。
周期	1
范例	XORWF REG, 1 B : REG = 0xAF, W = 0xB5 A : REG = 0x1A, W = 0xB5

电气特性

1. 绝对最大值 ($T_A = 25^\circ\text{C}$)

参数	范围	单位
电源电压	$V_{SS} - 0.3$ to $V_{SS} + 6.5$	V
输入电压	$V_{SS} - 0.3$ to $V_{DD} + 0.3$	
输出电压	$V_{SS} - 0.3$ to $V_{DD} + 0.3$	
每个引脚的输出拉电流	-25	mA
所有引脚的输出拉电流	-80	
每个引脚的输出灌电流	+30	
所有引脚的输出灌电流	+150	
最大工作电压	5.5	V
工作温度	-40 to +85	°C
储存温度	-65 to +150	

2. 直流 DC 特性 ($T_A = 25^\circ\text{C}$, $V_{DD} = 2.0\text{ V to } 5.5\text{ V}$)

参数	代号	条件	最小	典型值	最大	单位	
操作电压	V_{DD}	快时钟模式, 25°C , $F_{OSC} = 24\text{ MHz}$	4.0	–	5.5	V	
		快时钟模式, 25°C , $F_{OSC} = 16\text{ MHz}$	3.1	–	5.5		
		快时钟模式, 25°C , $F_{OSC} = 12\text{ MHz}$	2.1	–	5.5		
		快时钟模式, 25°C , $F_{OSC} = 8\text{ MHz}$	1.8	–	5.5		
		快时钟模式, 25°C , $F_{OSC} = 4\text{ MHz}$	1.7	–	5.5		
		慢时钟模式, 25°C , $F_{OSC} < 3\text{ MHz}$	1.5	–	5.5		
输入高电平	V_{IH}	所有输入, 除了 PA7	$V_{DD} = 5\text{ V}$	$0.6V_{DD}$	–	–	V
			$V_{DD} = 3\text{ V}$	$0.6V_{DD}$	–	–	V
		PA7	$V_{DD} = 5\text{ V}$	$0.7V_{DD}$	–	–	V
			$V_{DD} = 3\text{ V}$	$0.7V_{DD}$	–	–	V
输入低电平	V_{IL}	所有输入, 除了 PA7	$V_{DD} = 5\text{ V}$	–	–	$0.2V_{DD}$	V
			$V_{DD} = 3\text{ V}$	–	–	$0.2V_{DD}$	V
		PA7	$V_{DD} = 5\text{ V}$	–	–	$0.2V_{DD}$	V
			$V_{DD} = 3\text{ V}$	–	–	$0.2V_{DD}$	V
输出高电平	V_{OH}	所有输出	$V_{DD} = 5\text{ V}$, $I_{OH} = 8\text{ mA}$	4.4	–	–	V
			$V_{DD} = 3\text{ V}$, $I_{OH} = 4\text{ mA}$	2.6	–	–	V
输出低电平	V_{OL}	所有输出	$V_{DD} = 5\text{ V}$, $I_{OL} = 20\text{ mA}$	–	–	0.5	V
			$V_{DD} = 3\text{ V}$, $I_{OL} = 10\text{ mA}$	–	–	0.3	V
输入漏电流(高引脚)	I_{ILH}	所有输入	$V_{IN} = V_{DD}$	–	–	1	μA
输入漏电流(低引脚)	I_{ILL}	所有输入	$V_{IN} = 0\text{ V}$	–	–	-1	μA
电源电流	I_{DD}	快时钟模式, LVR 使能, WDT 使能, 无负载	$V_{DD} = 5\text{ V}$, FXT = 16 MHz	–	7	–	mA
			$V_{DD} = 3\text{ V}$, FXT = 16 MHz	–	3	–	
			$V_{DD} = 5\text{ V}$, FXT = 8 MHz	–	4	–	
			$V_{DD} = 3\text{ V}$, FXT = 8 MHz	–	2	–	
			$V_{DD} = 5\text{ V}$, FXT = 4 MHz	–	2.5	–	
			$V_{DD} = 3\text{ V}$, FXT = 4 MHz	–	1	–	
			$V_{DD} = 5\text{ V}$, FIRC = 16 MHz	–	6	–	
			$V_{DD} = 5\text{ V}$, FIRC = 8 MHz	–	3.2	–	
			$V_{DD} = 3\text{ V}$, FIRC = 8 MHz	–	1.5	–	
			$V_{DD} = 5\text{ V}$, FIRC = 4 MHz	–	2	–	
			$V_{DD} = 3\text{ V}$, FIRC = 4 MHz	–	1	–	

参数	代号	条件	最小	典型值	最大	单位	
电源电流	I_{DD}	慢时钟模式, LVR 使能, WKT PSC=11	$V_{DD} = 5 V,$ $SXT = 32 KHz$	-	60	-	μA
			$V_{DD} = 3 V,$ $SXT = 32 KHz$	-	10	-	
			$V_{DD} = 5 V,$ $SIRC=2KHz$	-	20	-	
			$V_{DD} = 3 V,$ $SIRC=2KHz$	-	6	-	
		空闲模式, LVR 使能	$V_{DD} = 5 V,$ $SXT=32 KHz$	-	35	-	
			$V_{DD} = 3 V,$ $SXT=32 KHz$ $TM2PSC=2'b00$	-	3	-	
			$V_{DD} = 5 V,$ $SIRC=32KHz$	-	34	-	
			$V_{DD} = 3 V,$ $SIRC=32KHz$	-	3	-	
		空闲模式, LVR 禁止	$V_{DD} = 5 V,$ $SXT = 32 KHz$	-	34	-	
			$V_{DD} = 3 V,$ $SXT=32 KHz$ $TM2PSC=2'b00$	-	2	-	
			$V_{DD} = 5 V,$ $SIRC=32KHz$	-	33	-	
			$V_{DD} = 3 V,$ $SIRC=32KHz$	-	2	-	
		省电模式, LVR 使能	$V_{DD} = 5 V$	-	1	2	
			$V_{DD} = 3 V$	-	0.5	1	
省电模式, LVR 禁止	$V_{DD} = 5 V$	-	-	0.1			
	$V_{DD} = 3 V$	-	-	0.1			
上拉电阻	R_P	$V_{IN} = 0 V$ Ports A/B/D/E	$V_{DD} = 5 V$	-	60	-	$K\Omega$
			$V_{DD} = 3 V$	-	120	-	
		$V_{IN} = 0 V$ PA7	$V_{DD} = 5 V$	-	55	-	$K\Omega$
			$V_{DD} = 3 V$	-	55	-	

3. 时钟计时 ($T_A = -40^{\circ}\text{C}$ to $+85^{\circ}\text{C}$)

参数	条件		最小	典型值	最大	单位
外部 RC 频率	$V_{DD} = 3\text{V}$	R = 4.7K C = 20 pF	-	3.3	-	MHz
		R = 10K C = 100 pF	-	0.8	-	
		R = 100K C = 300 pF	-	0.04	-	
	$V_{DD} = 5\text{V}$	R = 4.7K C = 20 pF	-	3.6	-	
		R = 10K C = 100 pF	-	0.7	-	
		R = 100K C = 300 pF	-	0.03	-	
FIRC 频率	25°C, $V_{DD} = 3 \sim 5.5\text{V}$ ($\pm 3\%$)		7.75	8	8.25	
	25°C, $V_{DD} = 2.6 \sim 3\text{V}$ ($\pm 5\%$)		7.6	8	8.4	
	$-40^{\circ}\text{C} \sim 85^{\circ}\text{C}$, $V_{DD} = 2.6 \sim 5.5\text{V}$		7.5	8	8.5	

 4. 复位定时特性 ($T_A = 25^{\circ}\text{C}$)

参数	条件	最小	典型值	最大	单位
复位输入低脉宽时间	Input $V_{DD} = 5\text{V} \pm 10\%$	3	-	-	μs
WDT 时间	$V_{DD} = 5\text{V}$, WDTPSC = 00	-	100	-	ms
	$V_{DD} = 3\text{V}$, WDTPSC = 00	-	130	-	ms
WKT 时间	$V_{DD} = 5\text{V}$, WKTPSC = 00	-	0.9	-	ms
	$V_{DD} = 3\text{V}$, WKTPSC = 00	-	1.0	-	ms
CPU 启动时间	$V_{DD} = 5\text{V}$	-	14	-	ms

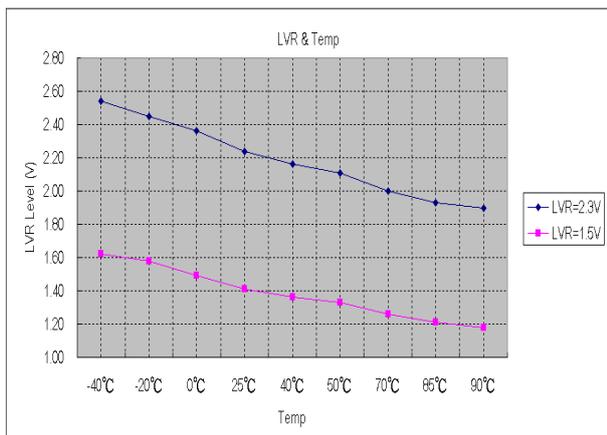
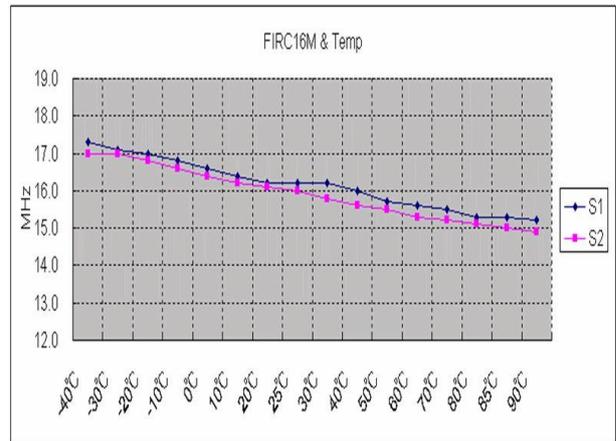
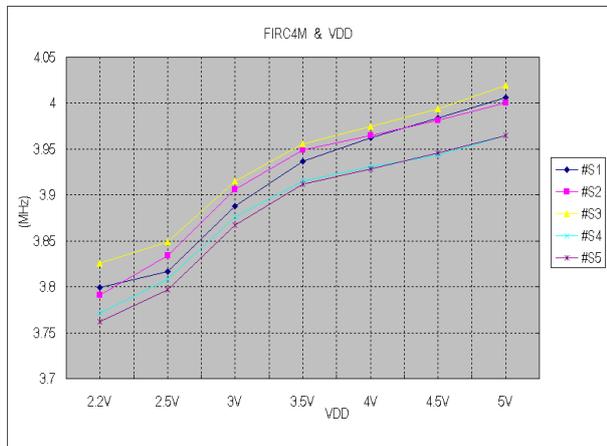
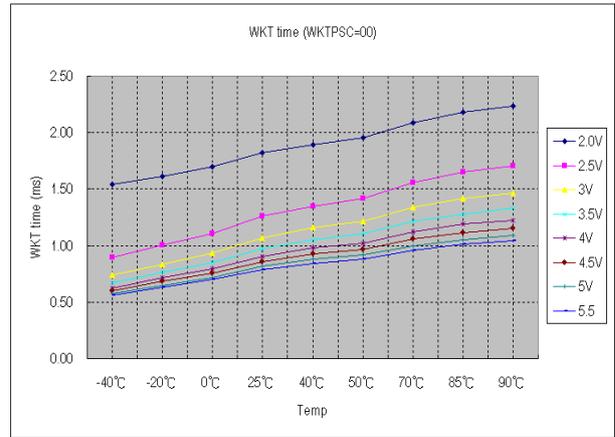
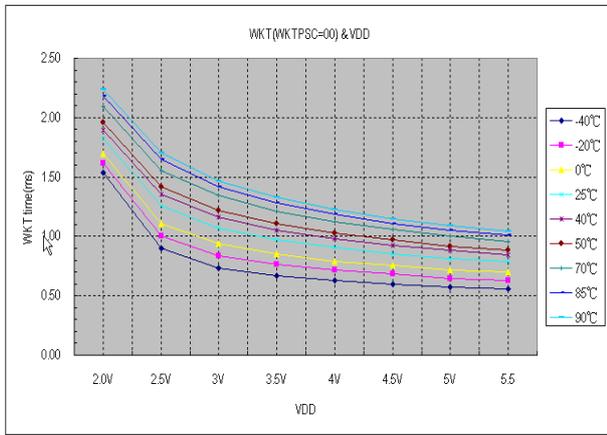
 5. LVR 线路特性 ($T_A = 25^{\circ}\text{C}$)

参数	代号	最小	典型值	最大	单位
LVR 滞后电压	V_{LVR}	-	1.5	-	V
		-	2.3	-	
		-	3.2	-	
低电压复位迟滞电压	V_{HYST}	-	± 0.1	-	V
低电压检测时间	t_{LVR}	100	-	-	μs

 6. 比较器特性 ($T_A = 25^{\circ}\text{C}$)

参数	代号	条件	最小	典型值	最大	单位
CMP 操作电流	I_{CMP}	$V_{DD} = 5\text{V}$	-	40		μA
CMP 输入偏移	V_{OFFSET}	$V_{in} = V_{SS}$	-50	-	+50	mV
CMP 常见伏范围	V_{CCVR}	$V_{DD} = 5\text{V}$	$V_{SS} - 0.3$	-	$V_{DD} - 0.3$	V
内部 V_{ref} 电压	V_{CIVREF}	CMPINPS=0~14	$V_{SS} + 0.1$		$V_{DD} - 0.6$	V

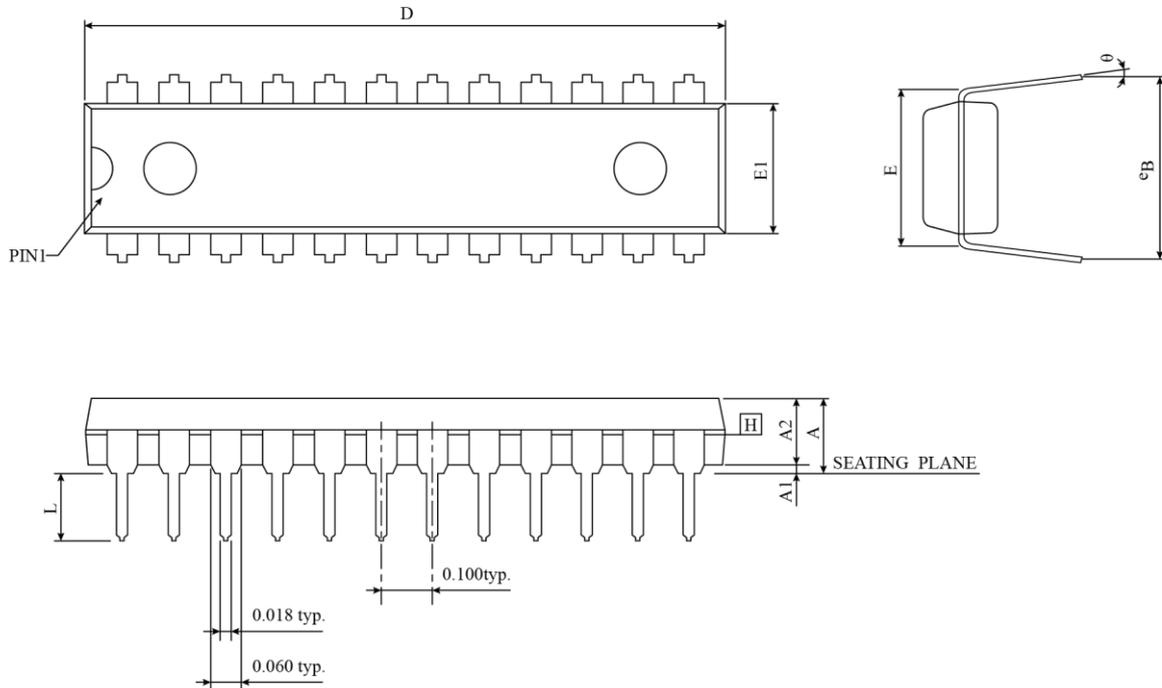
7. 特性曲线图



封装说明

订购须知:

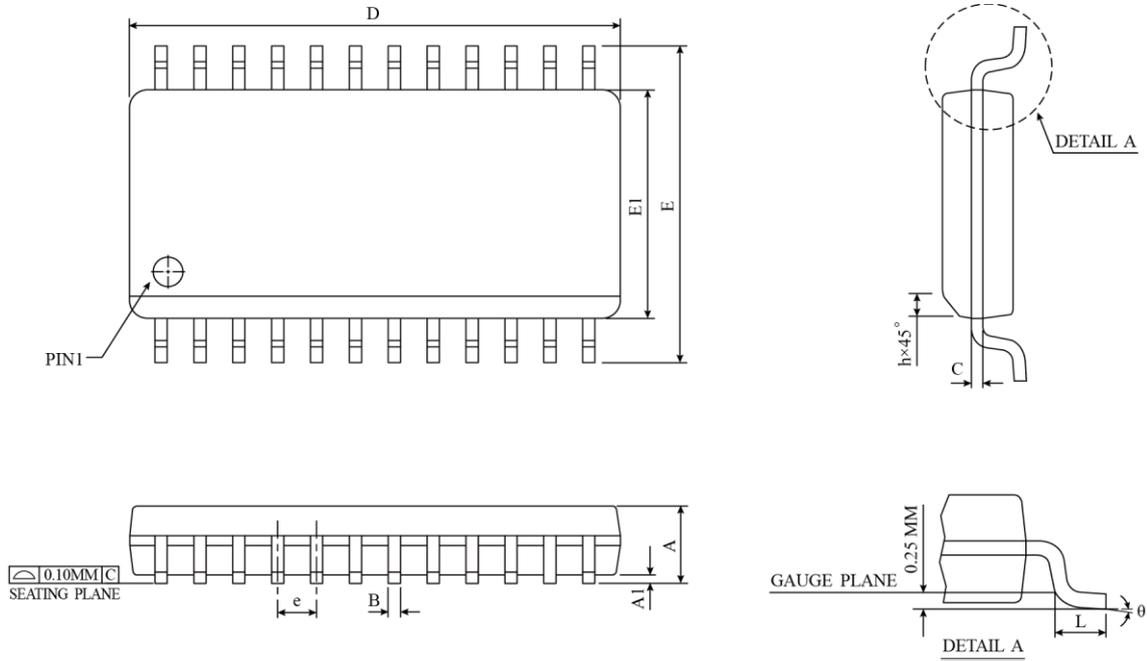
订购代码	封装
TM57PE40-OTP	Wafer / Dice 空片
TM57PE40-COD	Wafer / Dice 代烧 code
TM57PE40-OTP-12	DIP 24-pin (300 mil)
TM57PE40-OTP-22	SOP 24-pin (300 mil)
TM57PE40-OTP-13	DIP 28-pin (300 mil)
TM57PE40-OTP-08	DIP 28-pin (600 mil)
TM57PE40-OTP-23	SOP 28-pin (300 mil)
TM57PE40-OTP-09	DIP 32-pin (600 mil)
TM57PE40-OTP-24	SOP 32-pin (300 mil)
TM57PE40-OTP-A3	SOP 32-pin (450 mil)

24 脚直插封装尺寸(300 mil)


SYMBOL	DIMENSION IN MM		DIMENSION IN INCH	
	MIN	MAX	MIN	MAX
A	-	5.334	-	0.210
A1	0.381	-	0.015	-
A2	3.175	3.429	0.125	0.135
D	31.242	32.512	1.230	1.280
E	7.620 BSC		0.300 BSC	
E1	6.426	6.680	0.253	0.263
L	2.921	3.810	0.115	0.150
eB	8.509	9.525	0.335	0.375
theta	0°	15°	0°	15°
JEDEC	MS-001 (AF)			

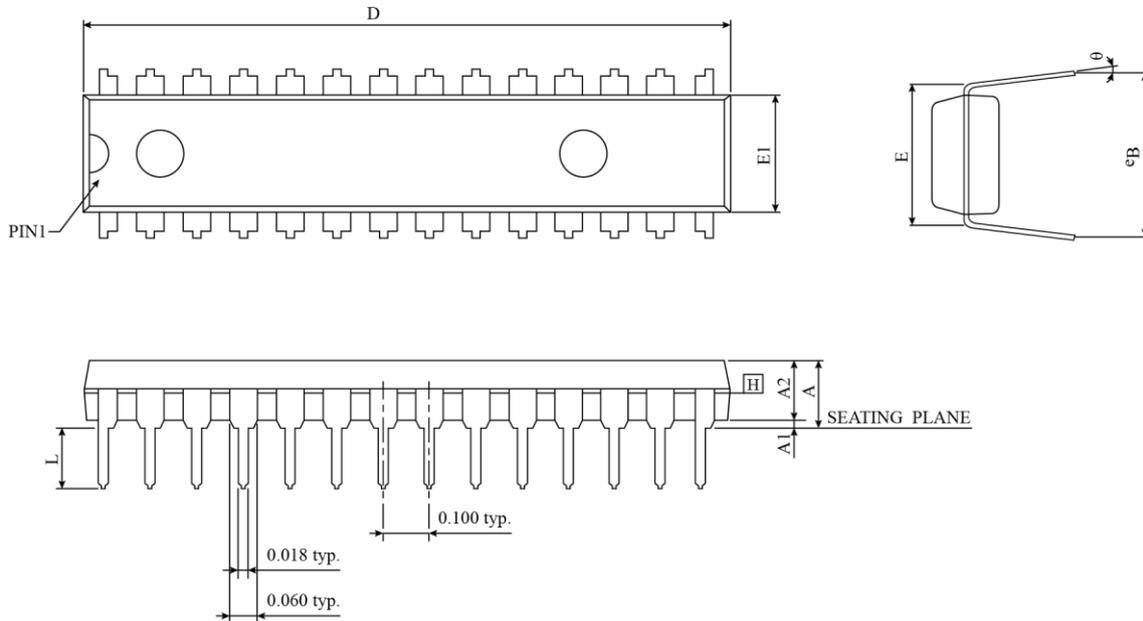
NOTES :

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

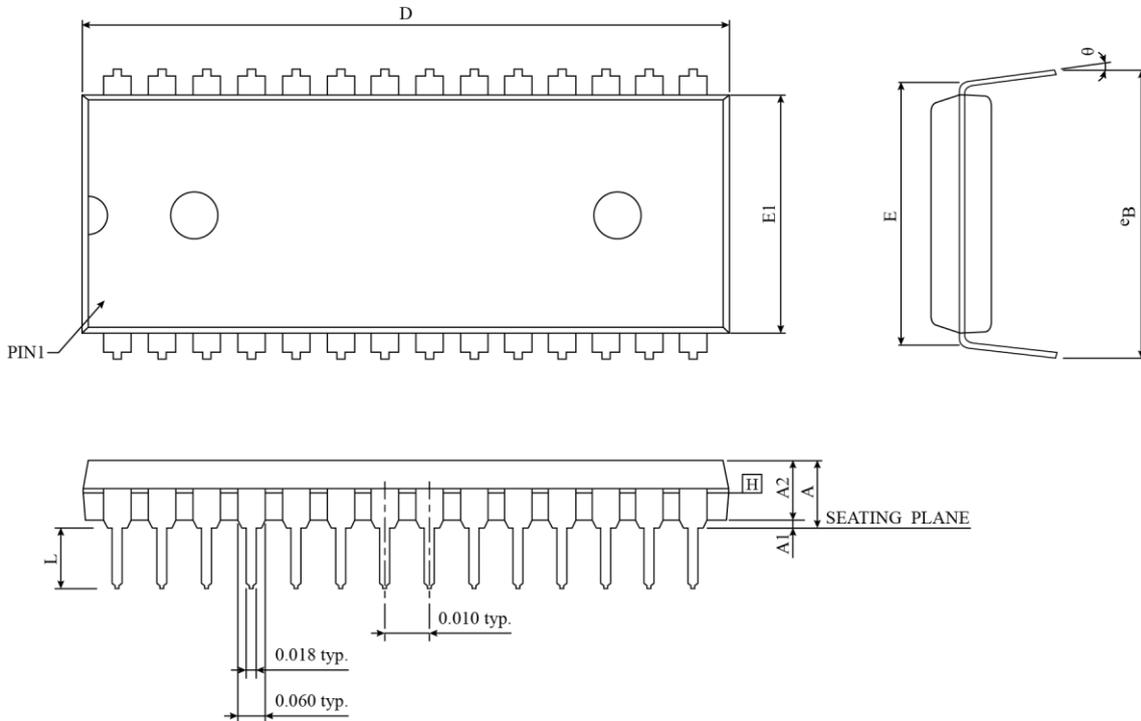
24 脚贴片封装尺寸(300 mil)


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	15.20	15.60	0.5985	0.6141
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 (AD)			

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

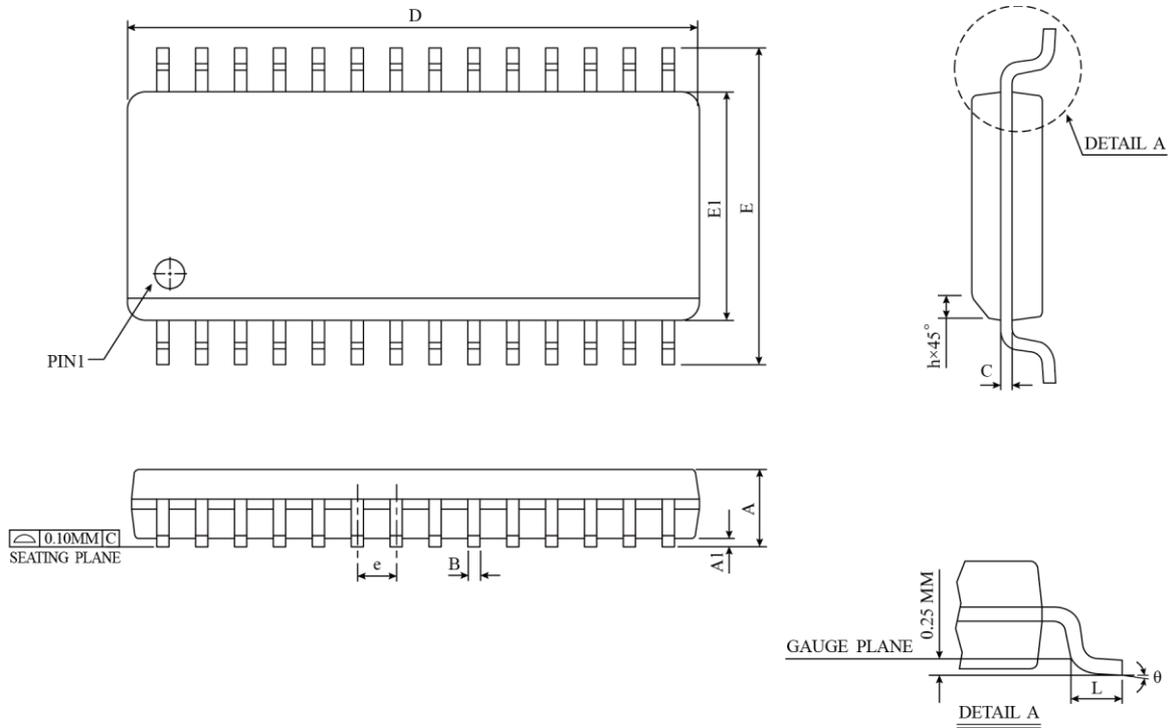
28 脚直插封装尺寸(300 mil)


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

28脚直插封装尺寸(600 mil)


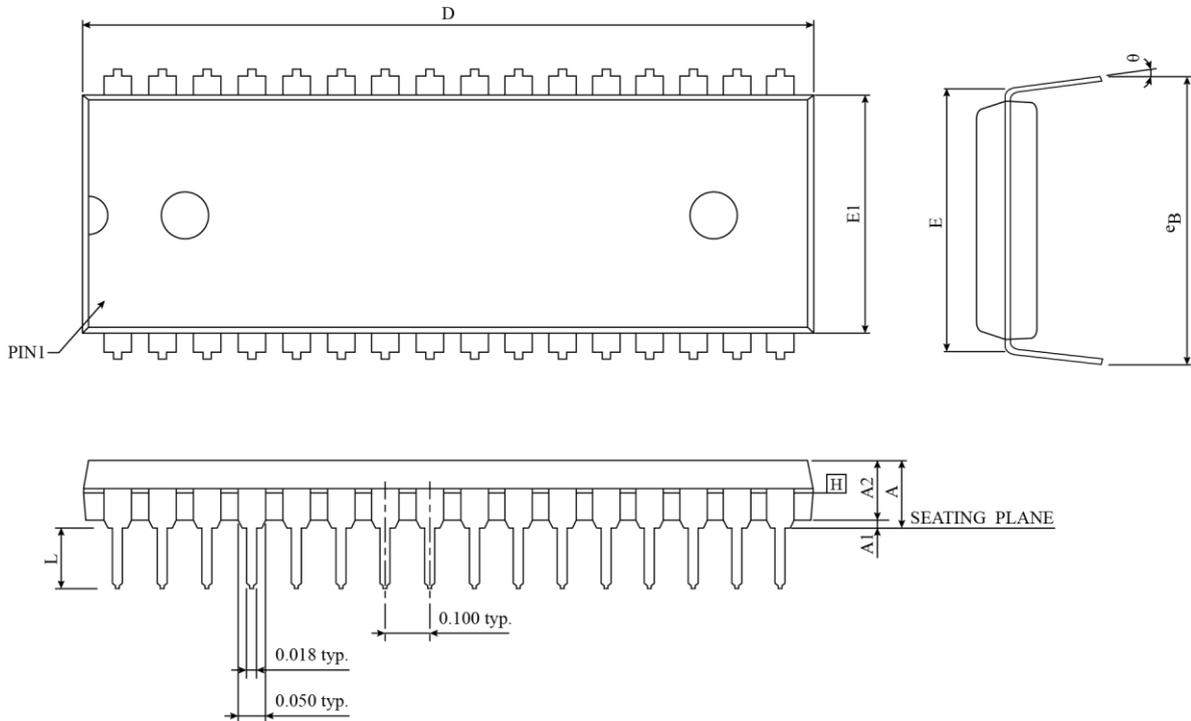
SYMBOL	DIMENSION IN MM		DIMENSION IN INCH	
	MIN	MAX	MIN	MAX
A	-	5.588	-	0.220
A1	0.381	-	0.015	-
A2	3.810	4.064	0.150	0.160
D	36.957	37.338	1.455	1.470
E	15.240 BSC		0.600 BSC	
E1	13.716	13.970	0.540	0.550
L	2.921	5.080	0.115	0.200
eB	16.002	17.018	0.630	0.670
θ	0°	15°	0°	15°
JEDEC	MS-011 (AB)			

NOTES : E1 DOES NOT INCLUDE MOLD FLASH.

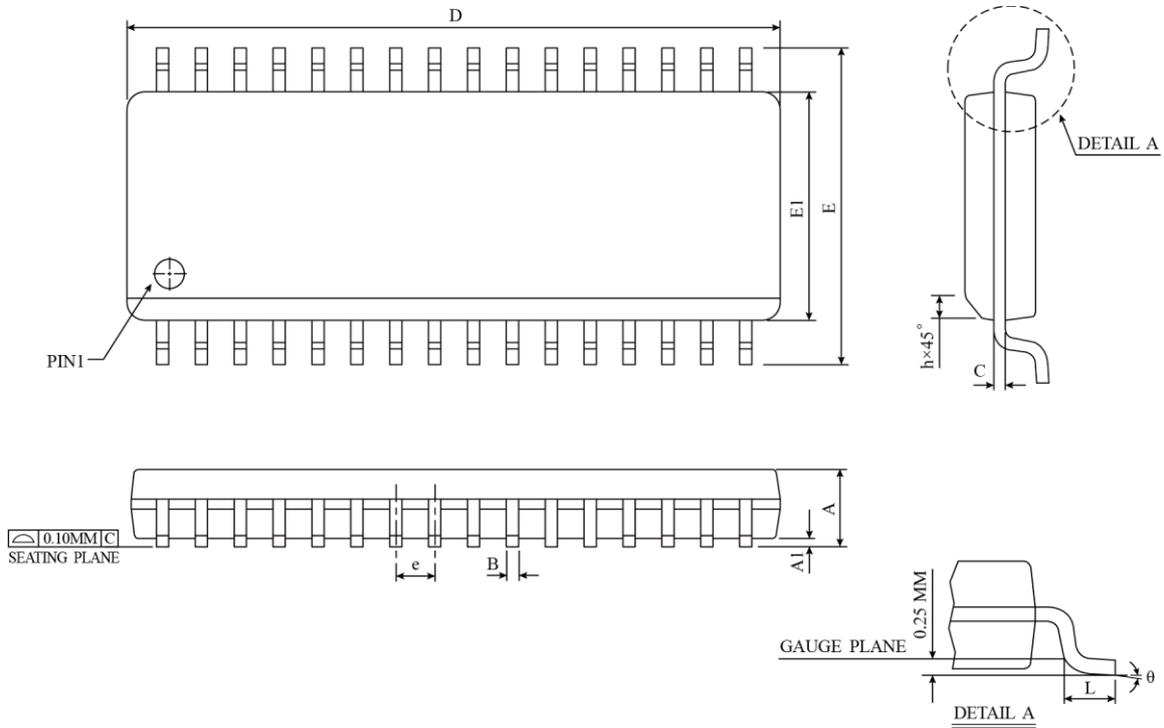
28 脚贴片封装尺寸(300 mil)


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)			

△ *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.

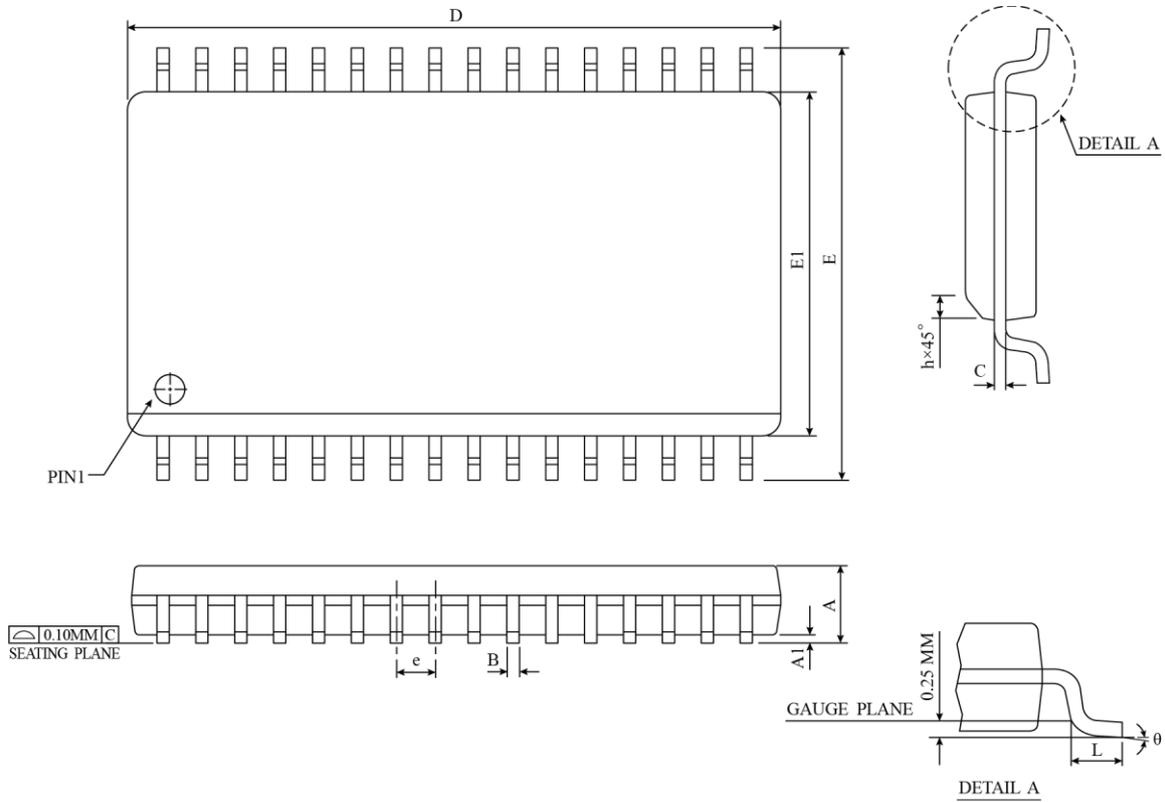
32 脚直插封装尺寸(600 mil)


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
eB	16.002	17.018	0.630	0.670
θ	0°	15°	0°	15°
JEDEC	MO-015 (AP)			

32 脚贴片封装尺寸(300 mil)


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°

△ *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 脚贴片封装尺寸(450 mil)


SYMBOL	DIMENSION IN MM		DIMENSION IN INCH	
	MIN	MAX	MIN	MAX
A	-	3.048	-	0.120
A1	0.051	0.305	0.002	0.012
B	0.41 TYP		0.016 TYP	
D	20.295	20.701	0.799	0.815
E	13.792	14.452	0.543	0.569
E1	11.176	11.430	0.440	0.450
e	1.27 TYP		0.050 TYP	
L	0.406	1.270	0.016	0.050
θ	0°	8°	0°	8°
JEDEC	M0-099 (AB)			

▲ *NOTES : DIMENSIONS "D" DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS.
 MOLD FLASH, PROTRUSIONS AND GATE BURRS SHALL NOT EXCEED .15mm (.006 in) PER SIDE.
 DIMENSIONS "E1" DOES NOT INCLUDE INTER-LEAD FLASH, OR PROTRUSIONS.
 INTER-LEAD FLASH AND PROTRUSIONS SHALL NOT EXCEED .25mm (.010 in) PER SIDE.
 DIMENSION B DOES NOT INCLUDE DAMBAR PROTRUSION.