

HC18P23xL

數據手冊

8 位 LCD 型 OTP 單片機

目錄

1 产品简述	5
1.1 特性	5
1.2 系统框图	7
1.3 引脚图	8
1.4 PAD 坐标信息	12
1.5 引脚说明	13
1.6 引脚电路	17
2 电性参数	18
2.1 极限参数	18
2.2 直流特性	18
2.3 交流特性	19
2.4 电气特性曲线图	20
3 中央处理器（CPU）	21
3.1 存储器	21
3.2 寻址模式	35
3.3 堆栈	36
4 复位	37
4.1 概述	37
4.2 上电复位	38
4.3 看门狗定时器复位	38
4.4 欠压复位	39
4.5 外部复位	40
5 系统时钟	43
5.1 概述	43
5.2 时钟框图	43
5.3 系统高频时钟	44
5.4 系统低频时钟	45
6 系统工作模式	47
6.1 概述	47
6.2 模式切换举例	48
6.3 高低频模式切换	49
6.4 唤醒时间	49
6.5 OSCCON 寄存器	50
7 中断	51
7.1 内核中断	52
7.2 外设中断	53
7.3 GIE 全局中断	56

7.4 中断保护	56
7.5 TIMER0 定时器中断	57
7.6 INT0 外部中断	58
7.7 PORT 电平变化中断	58
7.8 TIMER1 中断	60
7.9 TIMER2 定时器中断	61
7.10 AD 中断	62
7.11 CCP 中断	62
7.12 UART 中断	62
7.13 SPI 中断	62
7.14 PWM 中断	62
7.15 多中断操作	62
8 I/O 口	64
8.1 I/O 口模式	64
8.2 I/O 口上拉控制寄存器	65
8.3 I/O 口下拉控制寄存器	66
8.4 PORT 驱动控制寄存器	66
8.5 I/O 口数据寄存器	67
8.6 管脚配置寄存器	68
9 定时器/计数器	69
9.1 看门狗定时器	69
9.2 TIMER0 定时器/计数器	70
9.3 TIMER1 定时器/计数器	73
9.4 TIMER2 定时器	74
9.5 CCP 模块	76
10 PWM 模块	84
10.1 概述	84
10.2 PWM 相关寄存器	84
10.3 PWM0 模块	85
10.4 PWM1 模块	88
10.5 PWM2 模块	89
10.6 死区时间	90
11 模数转换 (ADC)	92
11.1 ADC 概述	92
11.2 A/D 寄存器	92
11.3 A/D 控制寄存器	93
11.4 AD 转换时间	94
11.5 ADC 使用	96
12 液晶显示驱动(LCD)	98
12.1 概述	98
12.2 LCD 相关寄存器	98

12.3 LCD 波形时序图.....	103
13 LED 驱动器.....	107
13.1 概述.....	107
13.2 LED 控制寄存器.....	107
13.3 LED 驱动波形.....	108
13.4 LED 亮度控制寄存器.....	108
13.5 说明.....	109
14 串行口通信.....	110
14.1 概述.....	110
14.2 UART 相关寄存器.....	110
14.3 工作方式.....	113
14.4 波特率发生器.....	117
14.5 多机通信.....	118
14.6 帧出错检测.....	119
15 串行外部设备接口 SPI.....	120
15.1 SPI 相关寄存器.....	120
15.2 SPI 信号描述.....	121
15.3 SPI 时钟速率.....	122
15.4 SPI 功能框图.....	122
15.5 SPI 工作模式.....	123
15.6 SPI 传送形式.....	124
15.7 SPI 出错检测.....	125
15.8 SPI 配置对照表.....	125
16 指令表.....	126
17 开发工具.....	127
17.1 OTP 烧录器 (PM18-4.0).....	127
17.2 HC-IDE.....	127
18 封装尺寸.....	128
18.1 SOP16.....	128
18.2 SOP20.....	128
18.3 SOP24.....	129
18.4 SSOP24.....	129
18.5 SOP28.....	130
18.6 LQFP48.....	130
19 芯片正印命名规则.....	131
19.1 芯片型号说明 (第一行).....	131
19.2 日期码规则 (第二行).....	131
19.3 生产批号 (第三行).....	131
20 修改记录.....	132

1 產品簡述

HC18P23xL是一顆採用高速低功耗CMOS工藝設計開發的8位元高性能精簡指令單片機，內部有8K×16位一次性程式設計ROM(OTP-ROM)，512×8位元的資料寄存器(RAM)，6組雙向I/O口、三個Timer計時器/計數器，兩個CCP模組，支援一路UART及SPI通信，一個8×32(4×32)的液晶顯示驅動模組及8×16LED驅動模組，一個16通道的12位模數轉換器，多個系統時鐘，四種系統工作模式以及多個中斷源。這款單片機可以廣泛應用於帶有顯示功能的遊戲搖杆、計時器、遙控器等產品。

1.1 特性

- CPU 特性
 - 36條高性能精簡指令
 - 8K×16位元的OTP程式記憶體
 - 512×8位的資料記憶體
 - 8級堆疊暫存器
 - 2T/4T時鐘模式
 - 立即、直接和兩組間接定址模式
 - 16位RDT查表
- I/O 口
 - 6組雙向I/O口：PORTA，PORTB，PORTC，PORTD，PORTE，PORTF
 - 最多48個雙向I/O口，其中有40個可用作液晶驅動口
 - 高灌/高拉電流能力，可直接驅動LED (IOH/IOL：16mA/25mA@ 4.4V/0.6V)
 - 最多47個可程式設計弱上拉/下拉口 (PA、PB、PC、PD、PE、PF)
 - 所有IO驅動電流可配置 (IOH/IOL：5mA/8mA@ 4.4V/0.6V)
 - PORTF大灌電流、與LCD COM口複用 (IOH/IOL：17mA/65mA@ 4.4V/0.6V)
 - 最多48個具有喚醒功能的電平變化中斷埠 (PA、PB、PC、PD、PE、PF)
- 三個 Timer 計時器/計數器
 - Timer0：帶有8位預分頻器的8位計時器/計數器
 - Timer1：帶有預分頻器的16位計時器/計數器
 - Timer2：帶有8位週期寄存器的8位計時器
- 兩個CCP模組
 - 16位捕捉、16位比較、最高12位PWM
- 3組6路可程式設計帶死區控制的固定相位PWM
 - 其中1組12bit、2組8bit
- 液晶顯示驅動
 - 最大驅動液晶點陣8×32(4×32)
 - 1/3、1/4、1/5、1/8 Duty，1/2、1/3、1/4Bias
- 8×16LED點陣驅動模組
 - LED點陣驅動模組
 - 最大可驅動8×16LED點陣
- BOR重定系統
 - BOR2.0V/2.4V/3.6V
- UART 模組
- SPI 模組
- 模數轉換器
 - 12位轉換解析度
 - 最多16個模擬輸入通道 (15個外部ADC輸入，1個內部1/4VDD檢測)
 - 內部參考電壓(VDD、4V、3V、2V)
 - 外部參考電壓
- 雙系統時鐘
 - 高頻系統時鐘
 - 高頻晶體振盪器：最高 20MHz
 - 內部 RC 振盪器：高達 32MHz
 - 低頻系統時鐘
 - 低頻晶體振盪器：32.768KHz
 - 低頻 RC 振盪器：32K(5V 典型值)
- 系統工作模式
 - 高頻模式
 - 低頻模式
 - 休眠模式
 - 綠色模式
- 中斷源

- 計時器中斷：Timer0、Timer1和Timer2
- INT0外部中斷
- 所有IO電平變化中斷
- CCP1/CCP2中斷
- ADC中斷
- UART中斷
- SPI中斷
- PWM中斷
- 復位
 - 上電復位（POR）
 - 外部復位（MCLR Reset）
 - 欠壓復位（BOR）
 - 看門狗計時器復位（WDT Reset）

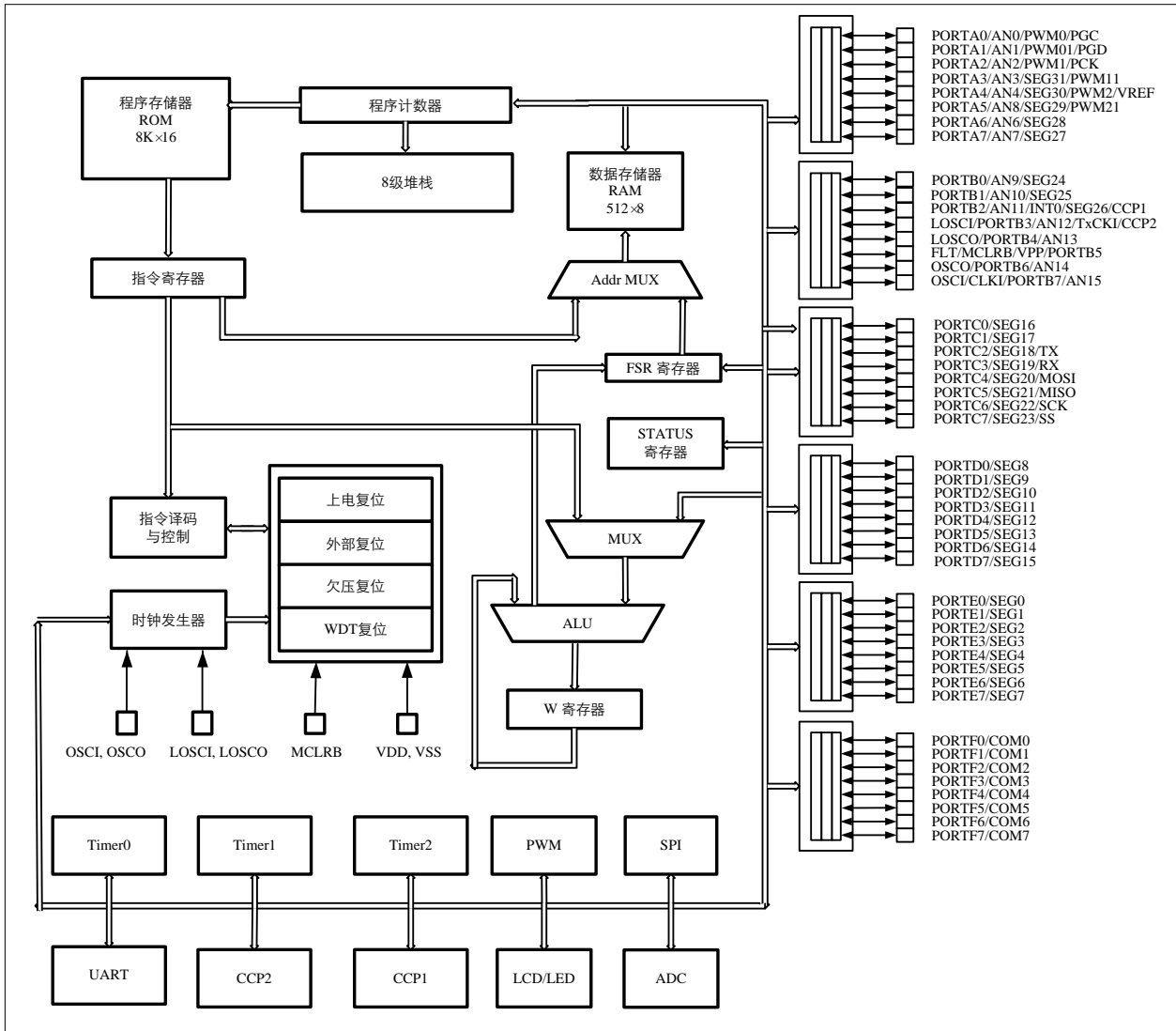
✓ 選型表

產品型號	ROM	RAM	堆疊	計時器	I/O	PWM	LCD	ADC	UART /SPI	PKG
HC18P232L	8K*16	512*8	8	3	14	6	4*6	9+1	1	SOP16
HC18P233L	8K*16	512*8	8	3	18	8	5*9	9+1	1	SOP20
HC18P233L	8K*16	512*8	8	3	22	6	6*10	10+1	1	SSOP24
HC18P233L	8K*16	512*8	8	3	22	6	6*10	10+1	1	SOP24
HC18P234L	8K*16	512*8	8	3	26	8	8*12	10+1	1	SOP28
HC18P235L	8K*16	512*8	8	3	46	8	4*30/ 8*30	13+1	1	LQFP48
HC18P235L	8K*16	512*8	8	3	48	8	4*32/ 8*32	15+1	1	Dice

使用注意事項：

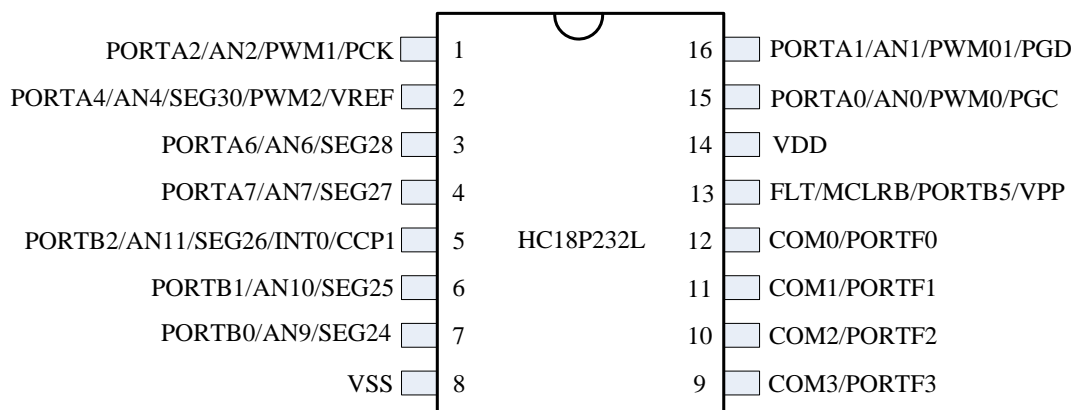
- 1、在系統對功耗要求較高時 ADC 推薦使用 2M 及以下採樣時鐘；
- 2、為提高 ADC 檢測精度，建議在 VDD 和 GND 之間並一個電容（104 電容即可）。
- 3、使用 LCD 時，相應的 COM 口及 SEG 口需禁止上拉電阻及下拉電阻功能，且不能外接 VDD 及 GND，防止 LCD 波形異常。
- 4、使用 LCD 時，需根據 LCD Pannel 尺寸選擇合適的偏置電阻，避免由於驅動不夠造成顯示效果不佳。
- 5、在使用 1/3Duty 時，COM3（PORTF3）埠為 NC，請勿使用。

1.2 系統框圖

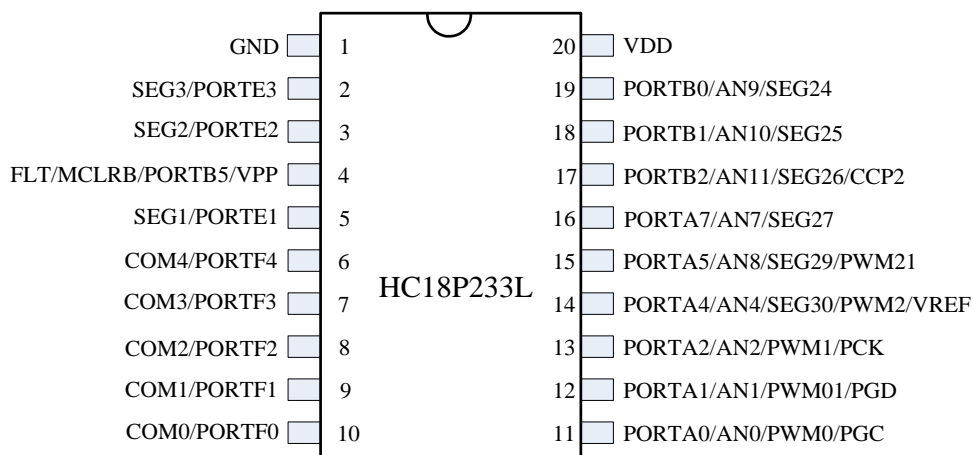


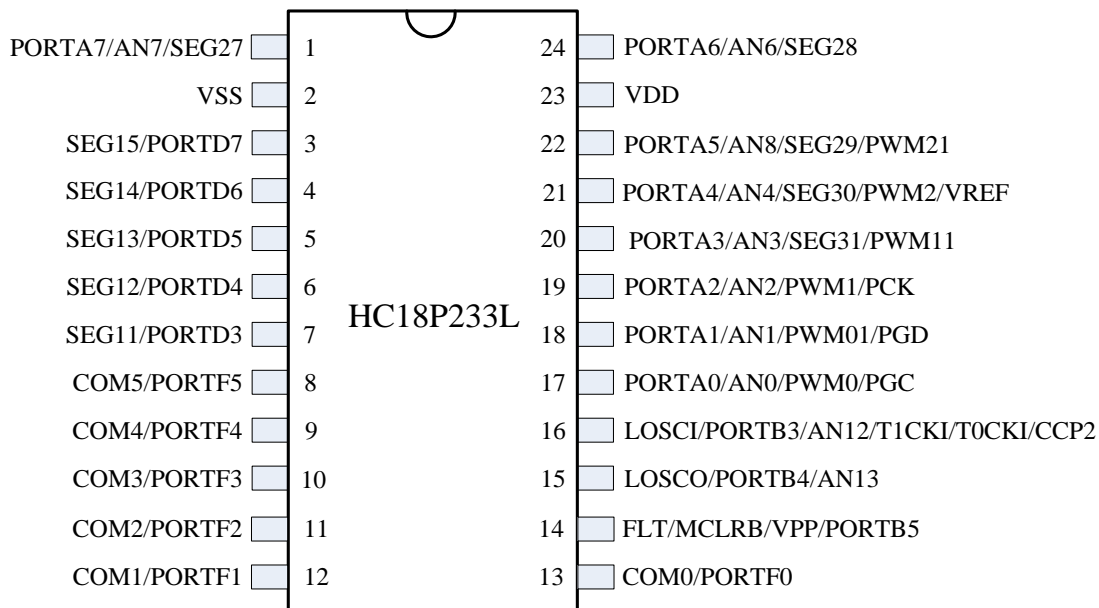
1.3 引腳圖

1.3.1 HC18P232L 引腳圖

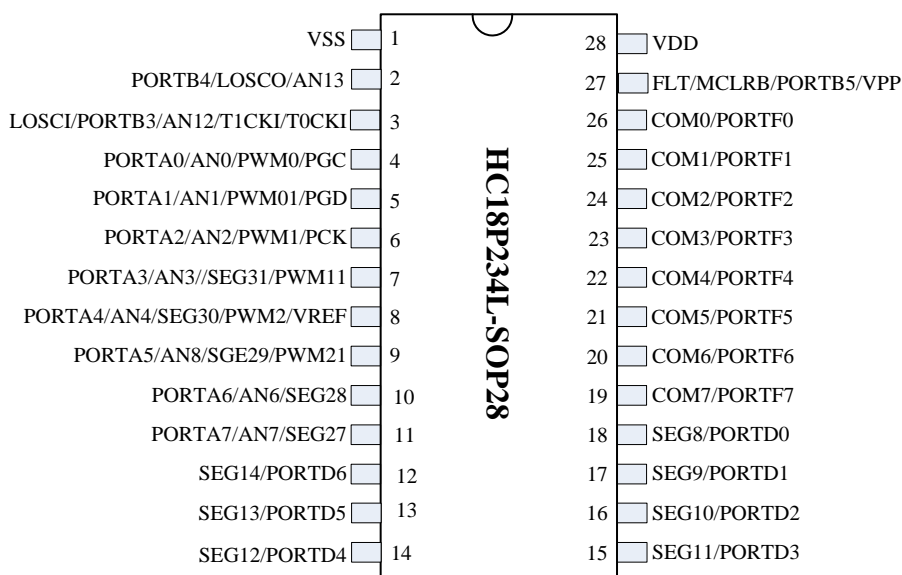


1.3.2 HC18P233L 引腳圖

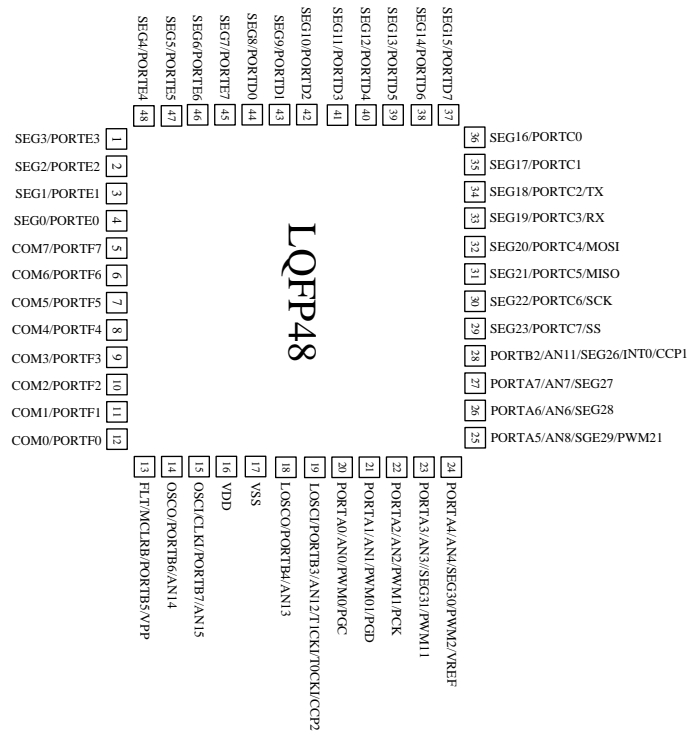




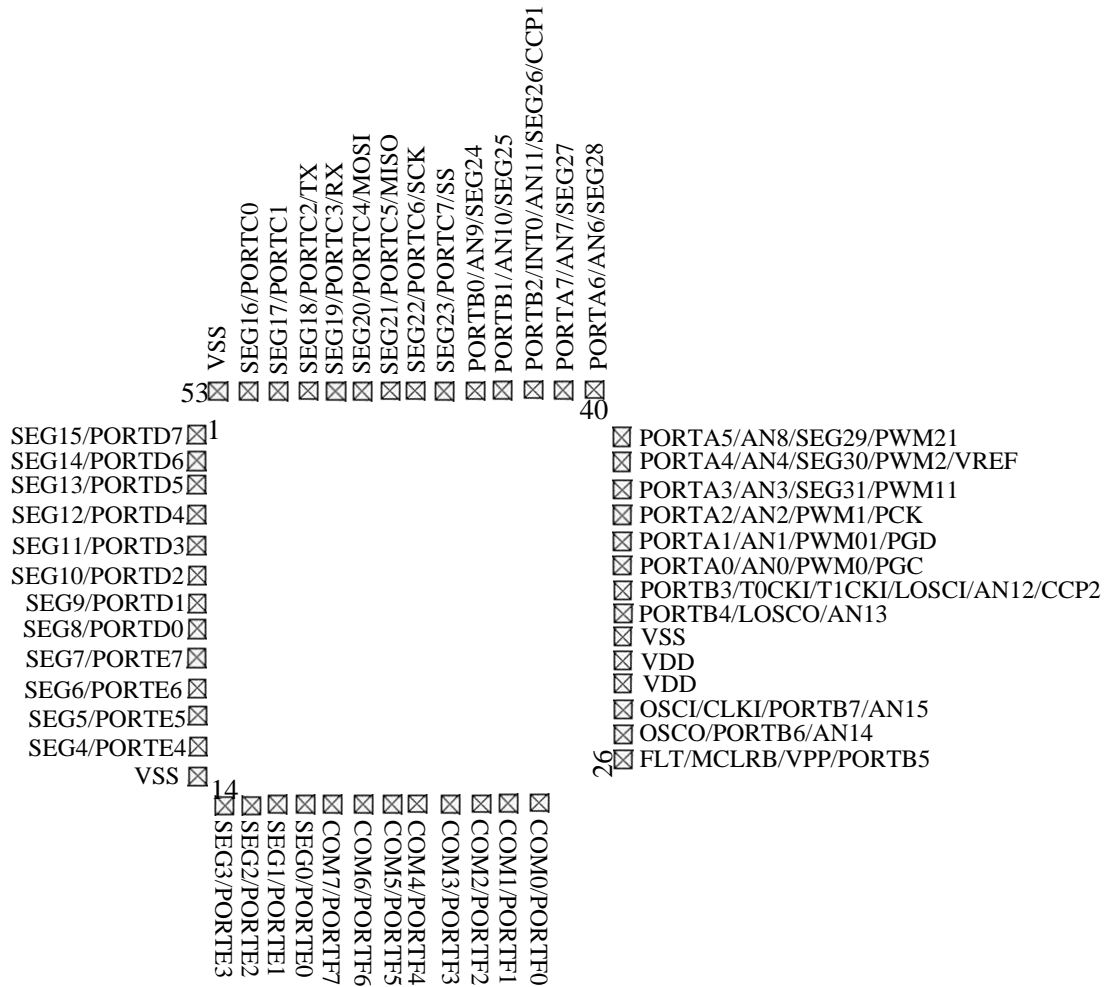
1.3.3 HC18P234L 引腳圖



1.3.4 HC18P235L 引腳圖



1.3.5 HC18P235L_DICE 引腳圖



1.4 PAD 座標信息

頂鋁厚度：0.85um					
Pad NO	Pin name	Coor-X	Coor-Y	PAD window	
1	pd[7]	-745.61	587.28	70	70
2	pd[6]	-745.61	492.5	70	70
3	pd[5]	-745.61	397.72	70	70
4	pd[4]	-745.61	302.94	70	70
5	pd[3]	-745.61	208.16	70	70
6	pd[2]	-745.61	113.38	70	70
7	pd[1]	-745.61	18.6	70	70
8	pd[0]	-745.61	-76.18	70	70
9	pe[7]	-745.61	-170.96	70	70
10	pe[6]	-745.61	-265.74	70	70
11	pe[5]	-745.61	-360.52	70	70
12	pe[4]	-745.61	-455.3	70	70
13	VSS	-733.61	-563.4	70	70
14	pe[3]	-625.22	-733.18	70	70
15	pe[2]	-530.44	-733.18	70	70
16	pe[1]	-435.66	-733.18	70	70
17	pe[0]	-340.88	-733.18	70	70
18	pf[7]	-224.485	-714.38	70	70
19	pf[6]	-110.605	-714.38	70	70
20	pf[5]	3.275	-714.38	70	70
21	pf[4]	117.155	-714.38	70	70
22	pf[3]	231.035	-714.38	70	70
23	pf[2]	344.915	-714.38	70	70
24	pf[1]	458.795	-714.38	70	70
25	pf[0]	572.675	-714.38	70	70
26	vpp	712.42	-651.78	70	70
27	pb[6]	726.71	-476.71	70	70
28	pb[7]	726.71	-381.93	70	70
29	VDD	714.39	-278.98	70	70
30	VDD	714.39	-183.7	70	70
31	VSS	733.39	-76.42	70	70
32	pb[4]	745.39	33.75	70	70
33	pb[3]	745.39	128.53	70	70
34	pa[0]	745.39	223.31	70	70
35	pa[1]	745.39	318.09	70	70
36	pa[2]	745.39	412.87	70	70
37	pa[3]	745.39	507.65	70	70
38	pa[4]	745.39	602.43	70	70
39	pa[5]	745.39	697.21	70	70
40	pa[6]	563.48	732.82	70	70

41	pa[7]	468.7	732.82	70	70
42	pb[2]	373.92	732.82	70	70
43	pb[1]	279.14	732.82	70	70
44	pb[0]	184.36	732.82	70	70
45	pc[7]	89.58	732.82	70	70
46	pc[6]	-5.2	732.82	70	70
47	pc[5]	-99.98	732.82	70	70
48	pc[4]	-194.76	732.82	70	70
49	pc[3]	-289.54	732.82	70	70
50	pc[2]	-384.32	732.82	70	70
51	pc[1]	-479.1	732.82	70	70
52	pc[0]	-573.88	732.82	70	70
53	VSS	-674.83	720.82	70	70

1.5 引腳說明

名稱	類型	說明
SEG3 PORTE3	AN I/O	SEG3輸出口 輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG2 PORTE2	AN I/O	SEG2輸出口 輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG1 PORTE1	AN I/O	SEG1輸出口 輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG0 PORTE0	AN I/O	SEG0輸出口 輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
COM7 PORTF7	AN I/O	COM7輸出口 輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
COM6 PORTF6	AN I/O	COM6輸出口 輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
COM5 PORTF5	AN I/O	COM5輸出口 輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
COM4 PORTF4	AN I/O	COM4輸出口 輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
COM3 PORTF3	AN I/O	COM3輸出口 輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
COM2 PORTF2	AN I/O	COM2輸出口 輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
COM1 PORTF1	AN I/O	COM1輸出口 輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
COM0 PORTF0	AN I/O	COM0輸出口 輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
MCLRB	I	復位輸入口，低電平有效
VPP	P	程式設計高壓電源輸入
PORTB5	I/O	輸入/輸出口，只能輸出0，埠電平變化中斷
FLT	I	PWM故障檢測輸入口

OSCO	AN	高頻晶體振盪器輸出口
PORTB6	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
AN14	AN	AD通道14輸入口
OSCI	AN	高頻晶體振盪器輸入口
CLKI	I	外部時鐘輸入口
PORTB7	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
AN15	AN	AD通道15輸入口
VDD	P	電源輸入
VSS	P	電源地
PORTB4	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
LOSCO	AN	低頻晶體振盪器輸出口
AN13	AN	AD通道13輸入口
PORTB3	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
T0CKI	I	Timer0外部時鐘輸入口（施密特觸發器）
T1CKI	I	Timer1外部時鐘輸入口（施密特觸發器）
LOSCI	AN	低頻晶體振盪器輸入口
AN12	AN	AD通道12輸入口
CCP2	I/O	CCP2輸入/輸出口。
PORTA0	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
AN0	AN	AD通道0輸入口
PWM0	O	12位PWM0輸出口
PGC	I	程式設計時鐘輸入口
PORTA1	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
AN1	AN	AD通道1輸入口
PWM01	O	與PWM0有固定相位關係的12位PWM輸出
PGD	I/O	程式設計資料登錄/輸出口
PORTA2	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
AN2	AN	AD通道2輸入口
PWM1	O	8位PWM1輸出口
PCK	O	測試模式下內部RC輸出口
PORTA3	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
AN3	AN	AD通道3輸入口
SEG31	AN	SEG31輸出口
PWM11	O	與PWM1有固定相位關係的8位PWM輸出
PORTA4	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
AN4	AN	AD通道4輸入口
SEG30	AN	SEG30輸出口
PWM2	O	8位PWM2輸出口
VREF	AN	ADC參考電壓輸入口
PORTA5	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
AN8	AN	AD通道8輸入口
SEG29	AN	SEG29輸出口
PWM21	O	與PWM2有固定相位關係的8位PWM輸出
PORTA6	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG28	AN	SEG28輸出口
AN6	AN	AD通道6輸入口

PORTA7	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG27	AN	SEG27輸出口
AN7	AN	AD通道7輸入口
PORTB2	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG26	AN	SEG26輸出口
AN11	AN	AD通道11輸入口
CCP1	IO	CCP1輸入/輸出口。
PORTB1	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG25	AN	SEG25輸出口
AN10	AN	AD通道10輸入口
PORTB0	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG24	AN	SEG24輸出口
AN9	AN	AD通道9輸入口
SEG23	AN	SEG23輸出口
PORTC7	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SS	I/O	SPI從機片選口
SEG22	AN	SEG22輸出口
PORTC6	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SCK	I/O	SPI時鐘口
SEG21	AN	SEG21輸出口
PORTC5	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
MISO	I/O	SPI的資料口，主機的輸入和從機的輸出
SEG20	AN	SEG20輸出口
PORTC4	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
MOSI	I/O	SPI資料登錄口，主機的輸出和從機的輸入
SEG19	AN	SEG19輸出口
PORTC3	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
RX	I	UART接收資料埠
SEG18	AN	SEG18輸出口
PORTC2	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
TX	O	UART發送資料埠
SEG17	AN	SEG17輸出口
PORTC1	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG16	AN	SEG16輸出口
PORTC0	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG15	AN	SEG15輸出口
PORTD7	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG14	AN	SEG14輸出口
PORTD6	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG13	AN	SEG13輸出口
PORTD5	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG12	AN	SEG12輸出口
PORTD4	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG11	AN	SEG11輸出口
PORTD3	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG10	AN	SEG10輸出口

PORTD2	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG9	AN	SEG9輸出口
PORTD1	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG8	AN	SEG8輸出口
PORTD0	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG7	AN	SEG7輸出口
PORTE7	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG6	AN	SEG6輸出口
PORTE6	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG5	AN	SEG5輸出口
PORTE5	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷
SEG4	AN	SEG4輸出口
PORTE4	I/O	輸入/輸出口，帶可程式設計上下拉，埠電平變化中斷

注: I = 輸入 O = 輸出 I/O = 輸入/輸出 P = 電源 AN = 模擬輸入輸出

1.6 引腳電路

圖 1-1：PORTA[7:5][3:0]的等效電路

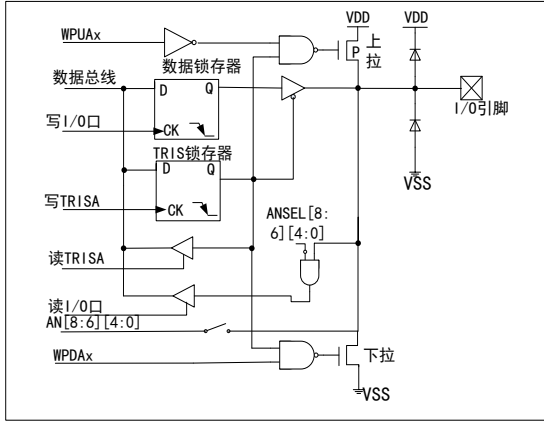


圖1-2：PORTA4的等效電路

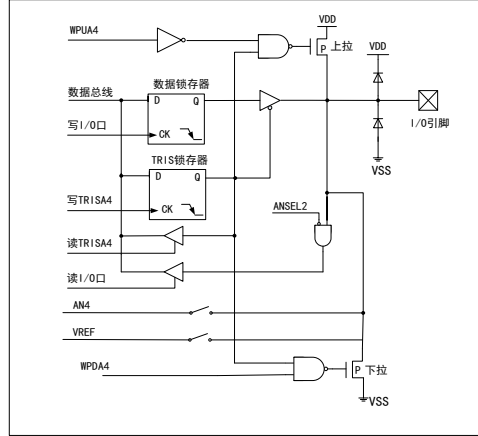


圖1-3：PORTB[7:6][4:0]的等效電路

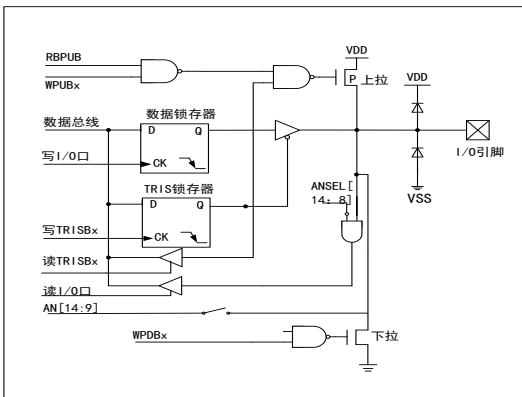


圖1-4：PORTB5口的等效電路

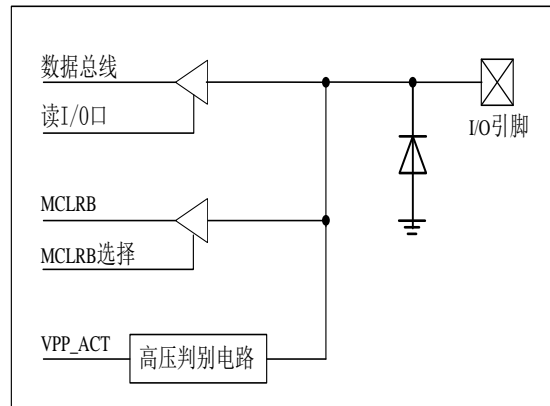


圖 1-5：PORTF[7:0]口的等效電路

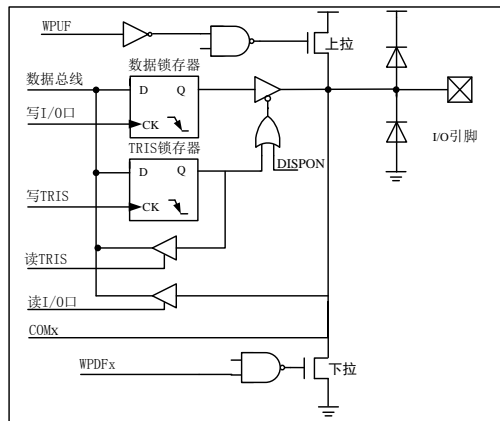
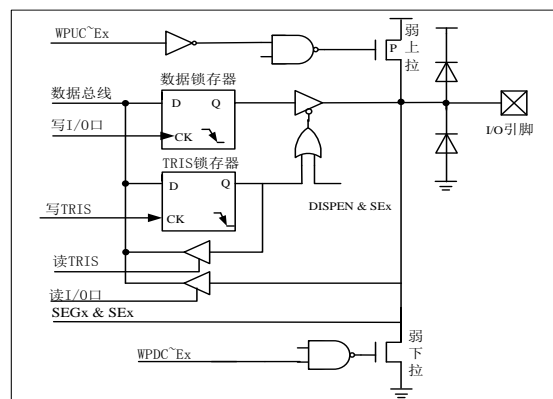


圖 1-6：PORTC、PORTD、PORTE 口等效電路



2 電性參數

2.1 極限參數

儲存溫度.....	-50 ℃~125 ℃
工作溫度.....	-40 ℃~85 ℃
電源供應電壓.....	VSS-0.3V~VSS+6.0V
埠輸入電壓.....	VSS-0.3V~VDD+0.3V

2.2 直流特性

符號	參數	測試條件		最小值	典型值	最大值	單位
		VDD	條件 (常溫 25 ℃)				
VDD	工作電壓	—	Fsys = 8MHz, 2T	1.8	—	5.5	V
		—	Fsys = 16MHz, 2T	2.6	—	5.5	V
IDD1	工作電流	3V	Fsys = 16MHz, 4T, LCD 關閉, 高頻模式, WDT 禁止, 無負載	—	1.70	—	mA
		5V		—	3.00	—	mA
IDD2	工作電流	3V	Fsys = 8MHz, 4T, LCD 關閉, 高頻模式, WDT 禁止, 無負載	—	1.20	—	mA
		5V		—	2.20	—	mA
IDD3	工作電流	3V	Fsys = 32KHz, 4T, LCD 關閉, 低頻模式, WDT 禁止, 無負載	—	3.0	—	μA
		5V		—	10.0	—	μA
IDD4	工作電流	3V	Fsys = 32KHz, 4T, LCD 關閉, 綠色模式, WDT 禁止, 無負載	—	2.0	—	μA
		5V		—	7.0	—	μA
Isb1	靜態電流	3V	LCD 關閉, 休眠模式, WDT 使能, 無負載	—	2.0	—	μA
		5V		—	7.0	—	μA
Isb2	靜態電流	3V	LCD 關閉, 休眠模式, 外部低頻晶振計時, 無負載	—	2.0	—	μA
		5V		—	5.2	—	μA
Isb3	靜態電流	3V	LCD 打開 (270K 偏置電阻), 休眠模式, WDT 禁止, 無負載	—	5	—	μA
		5V		—	10	—	μA
Isb3	靜態電流	3V	LCD 打開 (270K 偏置電阻), 休眠模式 WDT 禁止 全部點亮	—	14	—	μA
		5V		—	36	—	μA
Isb4	靜態電流	3V	LCD 關閉, 休眠模式, WDT 禁止, 無負載	—	—	1	μA
		5V		—	—	1	μA
VIL1	輸入低電平	—	輸入口	VSS	—	0.3VDD	V
VIH1	輸入高電平	—	輸入口	0.7VDD	—	VDD	V
VIL2	輸入低電平	—	施密特輸入口	VSS	—	0.2VDD	V
VIH2	輸入高電平	—	施密特輸入口	0.8VDD	—	VDD	V
VBOR1	低電壓重定 2.0V	—	—	—	2.0	—	V
VBOR2	低電壓重定 2.4V	—	—	—	2.4	—	V
VBOR3	低電壓重定 3.6V	—	—	—	3.6	—	V

VLVD0	低電壓標誌	—	—	—	2.4	—	V
VLVD1	低電壓標誌	—	—	—	3.6	—	V
IOL1	PORTF(LFEN=0)	3V	VOL=VSS+0.6V	—	30	—	mA
		5V		—	65	—	mA
IOH1	PORTF(LFEN=0)	3V	VOH=VDD-0.6V	—	11	—	mA
		5V		—	17	—	mA
IOL2	PORTF(LFEN=1)	3V	VOL=VSS+0.6V	—	9	—	mA
		5V		—	20	—	mA
IOH2	PORTF(LFEN=1)	3V	VOH=VDD-0.6V	—	4	—	mA
		5V		—	6.5	—	mA
IOL3	PORTx(LFEN=0)	3V	VOL=VSS+0.6V	—	18	—	mA
		5V		—	28	—	mA
IOH3	PORTx(LFEN=0)	3V	VOH=VDD-0.6V	—	9.0	—	mA
		5V		—	17	—	mA
IOL4	PORTx(LFEN=1)	3V	VOL=VSS+0.6V	—	5.0	—	mA
		5V		—	8.0	—	mA
IOH4	PORTx(LFEN=1)	3V	VOH=VDD-0.6V	—	3.5	—	mA
		5V		—	5.5	—	mA
RPH	內部上拉電阻	5V	可程式設計上拉電阻	—	100	—	kΩ
RPD	內部下拉電阻	5V	可程式設計下拉電阻	—	100	—	kΩ
VAD	ADC 輸入電壓	—	—	VSS	—	V _{REF}	V
DNL	差分非線性誤差	5V	AD 時鐘頻率 2MHz	—	±1	—	LSB
INL	積分非線性誤差	5V	AD 時鐘頻率 2MHz	—	±1	—	LSB
IADC	ADC 工作電流	3V	AD 時鐘頻率 2MHz	—	0.3	—	mA
		5V		—	0.5	—	mA

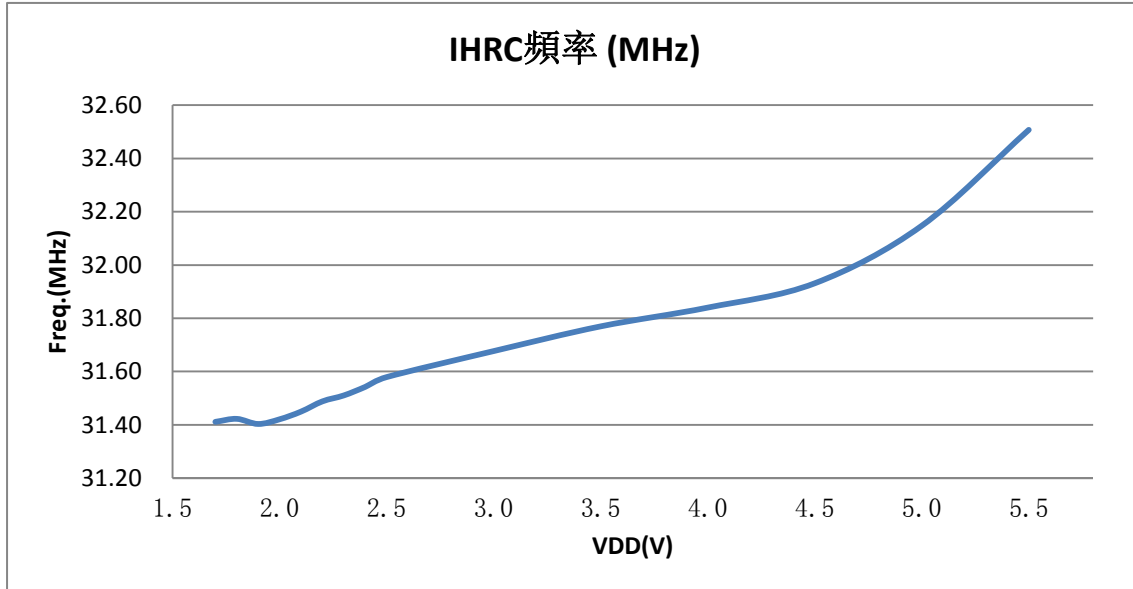
注：表格中 x = A、B、C、D、E 五組 PORT 口。

2.3 交流特性

符號	參數	測試條件		最小值	典型值	最大值	單位
		VDD	條件 (常溫 25℃)				
FSYS	系統時鐘	—	1.8V~5.5V	—	8	—	MHz
		—	2.6V~5.5V	—	16	—	MHz
FRCH	高頻內部 RC 振盪器	5V	—	—	32	—	MHz
FRCL	低頻內部 RC 振盪器	5V	—	—	32	—	KHz
FOSH	外部高頻晶振	—	2.6~5.5V	1	—	20	MHz
FOSL	外部低頻晶振	—	2.0~5.5V	14	32.768	—	KHz
TVDD	VDD 上升時間	5V	—	—	—	100	ms
TBOR	欠壓復位回應時間	5V	—	100	—	—	ns
TWDT	看門狗溢出時間	5V	使用預分頻 1:1	—	18	—	ms
			不使用預分頻器	—	72	—	ms

TMCLRB	復位脈衝時間	5V	—	200	—	—	μs
--------	--------	----	---	-----	---	---	----

2.4 電氣特性曲線圖



3 中央處理器 (CPU)

HC18P23xL CPU內核包括：

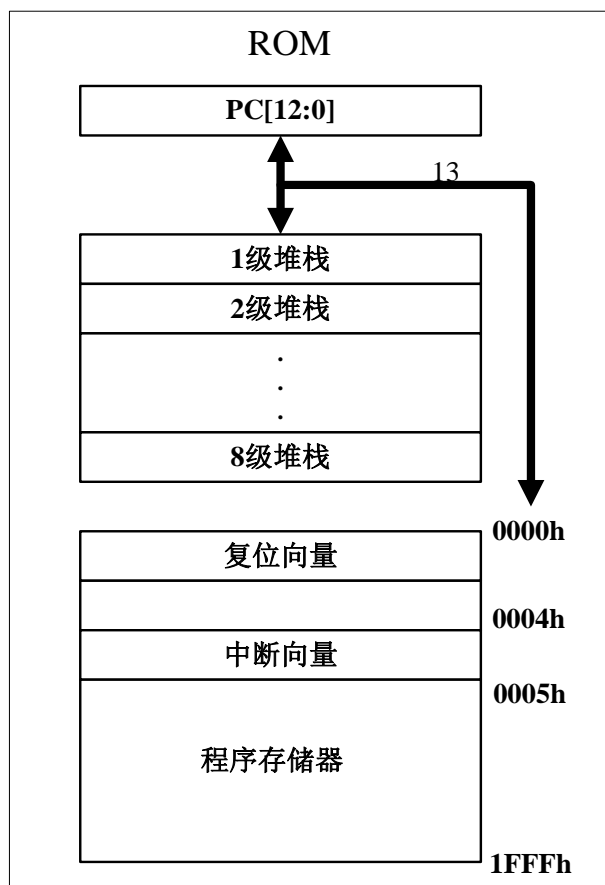
- 2T/4T 時鐘模式
- 8 級堆疊
- 程式記憶體
- 定址方式
- 資料記憶體

3.1 記憶體

3.1.1 程式記憶體 (OTP-ROM)

HC18P23xL具有8K×16位元的程式記憶體，下圖給出了程式記憶體的映射。訪問超出物理位址以外的單元時，會導致返回到位址最低單元。

復位向量是0000h，中斷向量是0004h。



3.1.1.1 復位向量 (0000h)

復位向量為0000h

- 上電復位 (POR=0，BOR=X，TO=1)

- 低電壓重定 (POR=1, BOR=0, TO=1)
- 看門狗復位 (POR=1, BOR=1, TO=0)
- 外部復位 (POR=1, BOR=1, TO=1)

發生上述任一種重定後，程式將從0000h處重新開始執行，系統寄存器也都將恢復為預設值。根據PCON寄存器中的POR，BOR標誌及STATUS 寄存器中的TO標誌位元的內容可以判斷系統重定方式。下面一段程式演示了如何定義ROM中的復位向量。

- 例：定義復位向量

```

        ORG          0000H      ;復位向量
        GOTO        MAIN      ;跳轉到使用者程式
        ...
        ORG          400H      ;使用者程式起始
MAIN:
        ...
        ...
        END            ;使用者程式結束
    
```

- 例：復位源判斷

```

        ORG          0000H
        GOTO        RST_JUGE
        ...
RST_JUGE:
        BCF         STATUS,RP0 ;Bank0
        BTFSS      PCON,POR
        GOTO        ISPOR      ;POR標誌為0，判定為上電復位
        BTFSS      PCON,BOR
        GOTO        ISBOR      ;POR=1，BOR=0，判定為低電壓重定
        BTFSS      STATUS,TO
        GOTO        ISWDTR     ;POR=1，BOR=1，TO=0，判定為WDT復位
EXT_RST:
        ...
        ...
        ...
        ;POR=1，BOR=1，TO=1，判定為外部復位
        ...
ISPOR:
        BSF         PCON,POR   ;上電復位處理常式
        ...
ISBOR:
        BSF         PCON,BOR   ;低電壓重定處理常式
        ...
ISWDTR:
        CLRWDT      ;TO標誌置1，WDT復位處理常式
        ...
        ;其他程式，注意處理Bank
    
```

3.1.1.2 中斷向量 (0004h)

中斷向量位址為0004h。一旦有中斷回應，程式計數器PC的當前值就會存入堆疊暫存器並跳轉到0004H 開始執行中斷服務程式。中斷服務副程式中需要對相應狀態寄存器進行適當的中斷點保護和恢復。下面的示例程式說明了如何編寫中斷服務程式。

- 例：中斷副程式的編寫

	BTFSS	STATUS,RP0	
Bank1_Inter:	GOTO	Bank0_Inter	;判斷進中斷前在Bank0還是Bank1
			;Bank1處理常式
	MOVWF	W_TEMP1	;保護W寄存器
	SWAPF	STATUS,W	
	MOVWF	STATUS_TEMP1	;保護STATUS寄存器
	MOVF	PCLATH,W	
	MOVWF	PCLATH_TEMP1	;保護PCLATH寄存器
	BCF	STATUS , RP0	
	BSF	Bank_Flag	
Bank0_Inter:	GOTO	Interrupt_Sev	
	MOVWF	W_TEMP0	
	SWAPF	STATUS,W	
	MOVWF	STATUS_TEMP0	
	MOVF	PCLATH,W	
	MOVWF	PCLATH_TEMP0	
	BCF	Bank_Flag	
	GOTO	Interrupt_Sev	
Interrupt_Sev:			
	BCF	STATUS , RP0	
	BTFSC	INTCON,INTF	
	GOTO	ISR_INT	;發生INT0中斷
	BTFSC	INTCON,T0IF	
	GOTO	ISR_T0	;發生T0中斷
	...		
	...		
Exit_Int:			
	BCF	STATUS , RP0	
	BTFSS	Bank_Flag	
	GOTO	Bank0_Exit	
Bank1_Exit:			
	BSF	STATUS , RP0	
	MOVF	PCLATH_TEMP1 ,W	
	MOVWF	PCLATH	;恢復PCLATH
	SWAPF	STATUS_TEMP1,W	
	MOVWF	STATUS	;恢復STATUS
	SWAPF	W_TEMP1,F	
	SWAPF	W_TEMP1,W	;恢復W
	RETFIE		;退出中斷
Bank0_Exit:			
	BCF	STATUS , RP0	

```

MOVF    PCLATH_TEMP0,W
MOVWF   PCLATH           ;恢復PCLATH
SWAPF   STATUS_TEMP0,W
MOVWF   STATUS           ;恢復STATUS
SWAPF   W_TEMP0,F
SWAPF   W_TEMP0,W       ;恢復W
RETFIE  ;退出中斷
    
```

對於編寫中斷服務程式，需要以下幾個要點需注意：

1. 中斷入口位址為 0x04，回應中斷後，程式自動跳轉到 0x04 開始執行；
2. 中斷服務程式需首先對相應的寄存器進行保護；
3. 保存系統寄存器時注意 Bank，如示例代碼中，分別定義一組 RAM 保存進入中斷前 Bank 的狀態；
4. 中斷服務副程式返回前對保護的寄存器進行恢復，注意恢復順序，對 W 必須使用 SWAPF；
5. 程式中使能兩個以上的中斷源時，程式需對發生中斷的中斷源進行判斷，從而執行相應的服務程式。
6. 需要軟體清零對應的中斷標誌；
7. RETFIE 指令將自動使能 GIE，請勿在中斷服務副程式中用其它指令使能 GIE，以免造成中斷回應混亂。

3.1.1.3 查表

方式一：

利用 ADDWF PCL, F 和 RETLW 指令實現資料表，因為以 PCL 為目的運算元的運算將改變程式指標(PC)值，其具體操作為 PC 的低 8 位為 ALU 的運算結果，PC 的高 5 位元將從 PC 高位緩衝器 PCLATH 中獲得。如下是資料表實現的一個例子。

➤ 例：數據查表

```

...
MOVLW   HIGH    TAB1    ;獲得資料表地址高8位（內部巨集指令）
MOVWF   PCLATH        ;表位址高位賦給PCLATH
MOVF    TABBUF,W      ;獲得表資料偏移量，調用前賦值。
CALL    TAB1          ;調用資料表
...

ORG     100H

TAB1 :

ADDWF   PCL,F         ;表頭運算
RETLW   DATA0_TAB1  ;W=0對應資料
RETLW   DATA1_TAB1  ;W=1對應資料
RETLW   DATA2_TAB1  ;W=2對應資料
...
RETLW   DATAFE_TAB1 ;W=0xFE對應資料
    
```


對於資料查表的程式設計，需注意：

1. 資料表寬度：8 位；
2. 資料表無法直接跨頁訪問，單頁可實現最大長度：255；
3. 當 PCL 與 W 的加運算有進位時，進位將被捨棄資料表溢出，將造成查表混亂；故表頭儘量放在資料頁前端，以免資料表溢出；
4. TABBUF 的值不得大於表長，否則將造成運行混亂。

➤ 例：跳轉表

跳轉表能夠實現多地址跳轉功能。由於 PCL 和 W 的值相加即可得到新的 PCL，同時 PCH 從 PCLATH 中載入，因此，可以通過對 PCL 加上不同的 W 值來實現多位址跳轉，可參考以下範例。

```

...
ORG          0100H
MOVLW       HIGH TAB2    ;獲得跳轉表位址高位（內部巨集指令）
MOVWF       PCLATH
MOVF        TABBUF,W
TAB2:       ADDWF        PCL,F
GOTO        LABEL0_TAB2 ;TABBUF =0,跳轉 LABEL0_TAB2
GOTO        LABEL1_TAB2 ;以下類推
GOTO        LABEL2_TAB2
GOTO        LABEL3_TAB2
    
```

注：

如上跳轉表，有 4 個跳轉分支，TABBUF 的合法範圍為 0x00~0x03。

方式二：

可以通過以下5個特殊功能寄存器對ROM區中的資料進行查找。

- PMCON
- PMDATL
- PMDATH
- PMADRL
- PMADRH

寄存器 PMADRH 指向 ROM 區資料位址的高位元組（bit8~bit15），寄存器 PMADRL 指向 ROM 區資料位址的低位元組（bit0~bit7）。將 PMCON 寄存器的 RDON 位置 1 啟動讀操作，使用兩條指令來讀數據，RDON 位置 1 後的二條指令被自動忽略，建議用戶 RDON 位置 1 後的兩條指令為 NOP。執行完讀操作後，所查找的資料保存在 PMDATH:PMDATL 寄存器。

➤ 例：查找ROM 位址為“TABLE”的值

```

BCF        STATUS,RP0    ;BANK0
MOVF       TABLE_ADDR_H,W
MOVWF     PMADRH        ;設置TABLE位址高位元組
MOVF       TABLE_ADDR_L,W
MOVWF     PMADRL        ;設置TABLE位址低位元組
    
```

```

        BSF          PMCON, RDON    ;開始讀
        NOP
        NOP          ;等待兩條指令
        MOVF        PMDATL, W
        MOVWF       TABLE _DATAL ;TABLE _DATAL= TABLE位址資料低位元
    組
        MOVF        PMDATH, W
        MOVWF       TABLE _DATAH ;TABLE _DATAH= TABLE位址資料高位元
    組
        ...
        ...
    TABLE:    DW          1234H    ;定義資料表（16 位元）資料。
               DW          F178H
               DW          2123H
    
```

9Eh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PMCON	-	-	-	-	-	-	-	RDON
R/W	-	-	-	-	-	-	-	R/W
POR的值	-	-	-	-	-	-	-	0

bit 0 **RDON**：讀控制位

0 =不啟動ROM記憶體讀操作

1 =啟動ROM讀操作（由硬體清零RDON；軟體只能將RDON位置1，但不能清零）

9Ah	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PMDATL	PMD7	PMD6	PMD5	PMD4	PMD3	PMD2	PMD1	PMD0
R/W	R/W	R/W	- R/W	R/W	R/W	R/W	R/W	R/W
POR的值	x	x	x	x	x	x	x	x

9Bh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PMDATH	PMD15	PMD14	PMD13	PMD12	PMD11	PMD10	PMD9	PMD8
R/W	- R/W	- R/W	- R/W	R/W	R/W	R/W	R/W	R/W
POR的值	x	x	x	x	x	x	x	x

PMDx[15:0]：ROM記憶體讀操作後， **PMADRH:PMADRL** 指向位址的資料

9Ch	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PMADRL	PMA7	PMA6	PMA5	PMA4	PMA3	PMA2	PMA1	PMA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	x	x	x	x	x	x	x	x

9Dh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PMADRH	PMA15	PMA14	PMA13	PMA12	PMA11	PMA10	PMA9	PMA8
R/W	- R/W	- R/W	- R/W	R/W	R/W	R/W	R/W	R/W
POR的值	x	x	x	x	x	x	x	x

PMAx[15:0] : ROM記憶體位址

3.1.2 晶片配置選擇表

HC18P23xL提供的配置字可以對多個系統模組做配置選擇，詳細配置選擇見下面表格。系統還提供了一個4×16bit的器件ID供使用者存儲校驗或其他代碼標識號。在程式運行過程中不能訪問這些存儲單元，但可在程式設計燒錄/校驗時對它們進行讀寫。

編譯選項	內容	功能說明
高頻系統時鐘選擇(OSCHM)	高頻晶體振盪器	高頻晶體振盪器 OSCI/OSCO 作為高頻晶體振盪器輸入/輸出口。
	內部 RC 振盪器	內部 16M RC 振盪器。
低頻系統時鐘選擇(OSCLM)	32K WDT 振盪器	內部 32K WDT 振盪器。
	計時器 1 振盪器	低頻晶體振盪器，32.768KHz，LOSCI/LOSCO 為低頻晶體振盪器輸入/輸出口。
高頻內部 RC 振盪器頻率選擇(ROSC)	16MHz	內部高頻 RC 振盪器頻率配置為 16MHz。
	8MHz	內部高頻 RC 振盪器頻率配置為 8MHz。
	4MHz	內部高頻 RC 振盪器頻率配置為 4MHz。
	2MHz	內部高頻 RC 振盪器頻率配置為 2MHz。
	1MHz	內部高頻 RC 振盪器頻率配置為 1MHz。
	500KHz	內部高頻 RC 振盪器頻率配置為 500KHz。
WDT 功能使能(WDTEN)	使能 WDT 功能	使能看門狗計時器功能。
	遮罩 WDT 功能	遮罩看門狗計時器功能。
外部復位使能(MCLREN)	使能外部復位	使能外部復位引腳。
	遮罩，做輸入	遮罩外部復位引腳，外部復位引腳做輸入功能。
加密使能位(CP0)	加密	使能使用者程式區 CODE 加密功能。
	不加密	遮罩使用者程式區 CODE 加密功能。
啟動時鐘選擇(SPDS)	高頻系統時鐘	系統選擇高頻系統時鐘作為啟動時鐘。
	低頻系統時鐘	系統選擇低頻系統時鐘作為啟動時鐘。
ADC 使能	打開 ADC 功能	使能 ADC 功能。
	禁止 ADC 功能	禁止 ADC 功能。
PORSEL 選擇	18ms	上電復位時間選擇為 18ms。
	4.5ms	上電復位時間選擇為 4.5ms。
SMTENB	IO 口施密特使能	使能 IO 口施密特功能。
	IO 口施密特禁止	禁止 IO 口施密特功能。
BORSEL	BOR3.6V	當系統電壓低於 3.6V 時，系統重定。
	BOR2.4V	當系統電壓低於 2.4V 時，系統重定。
	BOR2.0V	當系統電壓低於 2.0V 時，系統重定。
時鐘模式選擇(FCPUT)	4T	4T 模式，1 個運算速度有 4 個系統時鐘週期組成。
	2T	2T 模式，1 個運算速度有 2 個系統時鐘週期組成。

選擇晶片配置字注意事項：

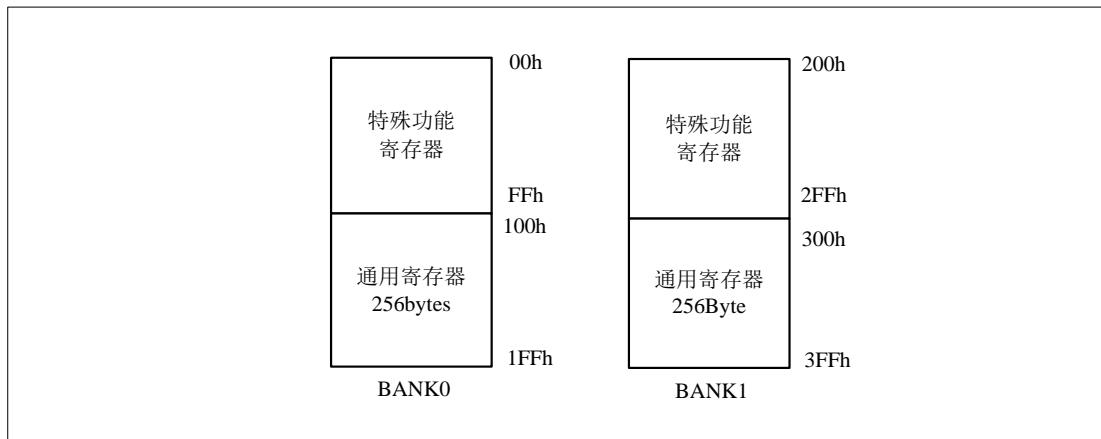
1. 在系統允許的情況下，盡量選用較低系統時鐘頻率，有利於降低系統功耗和提升系統電磁相容性；
2. 時鐘模式選擇為 2T 時，PWM 模組的最大解析度降低到 9 位；
3. 強干擾情況下，建議開啟 WDT 功能。

3.1.3 通用資料寄存器 (RAM)

HC18P23xL共有512個通用寄存器 (GPR) 和182個特殊功能寄存器 (SFR)，分在2個存儲區Bank0和Bank1，每個存儲區的低256個位址單元保留為特殊功能寄存器，RP0是存儲區的選擇位元。

CORE Register 00-09h&200-209h

00h&200h	INDF0	0 ^o	INDF0 ^o	間接尋址 0 寄存器 (不是实际存在的物理寄存器) ^o							
01h&201h	INDF1	1 ^o	INDF1 ^o	間接尋址 1 寄存器 (不是实际存在的物理寄存器) ^o							
02h&202h	PCL	2 ^o	PCL ^o	程序计数器 (PC) 低字节 ^o							
03h&203h	STATUS	3 ^o	STATUS ^o	^o	^o	RP0 ^o	TO ^o	PD ^o	Z ^o	DC ^o	C ^o
04h&204h	FSR0L	4 ^o	FSR0L ^o	間接尋址 0 地址低位指针 ^o							
05h&205h	FSR0H	5 ^o	FSR0H ^o	間接尋址 0 地址高位指针 ^o							
06h&206h	FSR1L	6 ^o	FSR1L ^o	間接尋址 1 地址低位指针 ^o							
07h&207h	FSR1H	7 ^o	FSR1H ^o	間接尋址 1 地址高位指针 ^o							
08h&208h	PCLATH	8 ^o	PCLATH ^o	^o	^o	^o	程序计数器高 5 位 ^o				
09h&209h	INTCON	9 ^o	INTCON ^o	GIE ^o	PEIE ^o	T0IE ^o	INTE ^o	RBIE ^o	T0IF ^o	INTF ^o	RBIF ^o



3.1.4 特殊功能寄存器 (SFR)

3.1.4.1 特殊功能寄存器列表

地址	名稱	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	復位初值
BANK0										
010h	TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111
011h	TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111
012h	TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111
013h	TRISD	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0	1111 1111
014h	TRISE	TRISE7	TRISE6	TRISE5	TRISE4	TRISE3	TRISE2	TRISE1	TRISE0	1111 1111
015h	TRISF	TRISF7	TRISF6	TRISF5	TRISF4	TRISF3	TRISF2	TRISF1	TRISF0	1111 1111
01Ch	PORTA	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0	0000 0000
01Dh	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	0000 0000
01Eh	PORTC	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	0000 0000
01Fh	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	0000 0000
020h	PORTE	PORTE7	PORTE6	PORTE5	PORTE4	PORTE3	PORTE2	PORTE1	PORTE0	0000 0000

地址	名稱	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	復位初值
021h	PORTF	PORTF7	PORTF6	PORTF5	PORTF4	PORTF3	PORTF2	PORTF1	PORTF0	0000 0000
028h	WPUA	WPUA7	WPUA6	WPUA5	WPUA4	WPUA3	WPUA2	WPUA1	WPUA0	1111 1111
029h	WPUB	WPUB7	WPUB6	WPUB5	WPUB4	WPUB3	WPUB2	WPUB1	WPUB0	1111 1111
02Ah	WPUC	WPUC7	WPUC6	WPUC5	WPUC4	WPUC3	WPUC2	WPUC1	WPUC0	1111 1111
02Bh	WPUD	WPUD7	WPUD6	WPUD5	WPUD4	WPUD3	WPUD2	WPUD1	WPUD0	1111 1111
02Ch	WPUE	WPUE7	WPUE6	WPUE5	WPUE4	WPUE3	WPUE2	WPUE1	WPUE0	1111 1111
02Dh	WPUF	WPUF7	WPUF6	WPUF5	WPUF4	WPUF3	WPUF2	WPUF1	WPUF0	1111 1111
034h	WPDA	WPDA7	WPDA6	-	WPDA4	WPDA3	WPDA2	WPDA1	WPDA0	11-1 1111
035h	WPDB	WPDB7	WPDB6	WPDB5	WPDB4	WPDB3	WPDB2	WPDB1	WPDB0	1111 1111
036h	WPDC	WPDC7	WPDC6	WPDC5	WPDC4	WPDC3	WPDC2	WPDC1	WPDC0	1111 1111
037h	WPDD	WPDD7	WPDD6	WPDD5	WPDD4	WPDD3	WPDD2	WPDD1	WPDD0	1111 1111
038h	WPDE	WPDE7	WPDE6	WPDE5	WPDE4	WPDE3	WPDE2	WPDE1	WPDE0	1111 1111
039h	WPDF	WPDF7	WPDF6	WPDF5	WPDF4	WPDF3	WPDF2	WPDF1	WPDF0	1111 1111
040h	IOCA	IOCA7	IOCA6	IOCA5	IOCA4	IOCA3	IOCA2	IOCA1	IOCA0	0000 0000
041h	IOCB	IOCB7	IOCB6	IOCB5	IOCB4	IOCB3	IOCB2	IOCB1	IOCB0	0000 0000
042h	IOCC	IOCC7	IOCC6	IOCC5	IOCC4	IOCC3	IOCC2	IOCC1	IOCC0	0000 0000
043h	IOCD	IOCD7	IOCD6	IOCD5	IOCD4	IOCD3	IOCD2	IOCD1	IOCD0	0000 0000
044h	IOCE	IOCE7	IOCE6	IOCE5	IOCE4	IOCE3	IOCE2	IOCE1	IOCE0	0000 0000
045h	IOCF	IOCF7	IOCF6	IOCF5	IOCF4	IOCF3	IOCF2	IOCF1	IOCF0	0000 0000
04Ch	PORCTR	-	-	-	CCPCT	SPPCT1	SPPCT0	UAPCT1	UAPCT0	---0 0000
04Dh	LAEN	LAEN7	LAEN6	LAEN5	LAEN4	LAEN3	LAEN2	LAEN1	LAEN0	0000 0000
04Eh	LBEN	LBEN7	LBEN6	LBEN5	LBEN4	LBEN3	LBEN2	LBEN1	LBEN0	0000 0000
04Fh	LCEN	LCEN7	LCEN6	LCEN5	LCEN4	LCEN3	LCEN2	LCEN1	LCEN0	0000 0000
050h	LDEN	LDEN7	LDEN6	LDEN5	LDEN4	LDEN3	LDEN2	LDEN1	LDEN0	0000 0000
051h	LEEN	LEEN7	LEEN6	LEEN5	LEEN4	LEEN3	LEEN2	LEEN1	LEEN0	0000 0000
052h	LFEN	LFEN7	LFEN6	LFEN5	LFEN4	LFEN3	LFEN2	LFEN1	LFEN0	0000 0000
054h	PIR1	-	ADIF	-	-	-	CCP1IF	T2IF	T1IF	-0-- -000
055h	PIR2	PWM2IF	PWM1IF	PWM0IF	-	RXIF	TXIF	SPIF	CCP2IF	000- 0000
056h	PIR3	-	-	-	RFIF	REIF	RDIF	RCIF	RAIF	---0 0000
058h	T1L	Timer1 計數寄存器低位元組								xxxx xxxx
059h	T1H	Timer1 計數寄存器高位元組								xxxx xxxx
05Ah	T1CON	T1CS1	T1CS0	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	-	T1ON	0000 00-0
05Bh	T0	Timer0 計數寄存器								xxxx xxxx
05Ch	T2	Timer2 計數寄存器								xxxx xxxx
05Dh	PR2	Timer 週期寄存器								0000 0000
05Eh	T2CON	-	T2CKPS3	T2CKPS2	T2CKPS1	T2CKPS0	T2ON	-	-	-000 00--
05Fh	PR1L	Timer1 週期寄存器低位元組								xxxx xxxx
060h	PR1CON	PWM1T1	PWM1T0	PWM2T1	PWM2T0	T1CKPS3	T1CKPS2	PWMPR1	PRIEN	0000 0000
070h	PIE1	-	ADIE	-	-	-	CCP1IE	T2IE	T1IE	-0-- -000

地址	名稱	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	復位初值
071h	PIE2	PWM2IE	PWM1IE	PWM0IE	-		UARTIE	SPIE	CCP2IE	000- -000
072h	PIE3	-	-	-	RFIE	REIE	RDIE	RCIE	RAIE	---0 0000
078h	OPTION	RBPUB	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	0000 0000
079h	PCON	LVD2EN	LVD1EN	-	WDTENS	LVD2F	LVD1F	POR	BOR	00-1 qqqq
07Ah	OSCCON	T0SCEN	-	-	-	-	-	HXEN	SCS	0--- --0q
080h	CCPR2L	CCP2 寄存器低位元組								xxxx xxxx
081h	CCPR2H	CCP2 寄存器高位元組								xxxx xxxx
082h	CCP2CON	-	-	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000
083h	CCPR1L	CCP1 寄存器低位元組								0000 0000
084h	CCPR1H	CCP1 寄存器高位元組								0000 0000
085h	CCP1CON	-	-	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000
08Ch	ANSELL	ANSEL7	ANSEL6	-	ANSEL4	ANSEL3	ANSEL2	ANSEL1	ANSEL0	11-1 1111
08Dh	ANSELH	ANSEL15	ANSEL14	ANSEL13	ANSEL12	ANSEL11	ANSEL10	ANSEL9	ANSEL8	1111 1111
092h	ADRESL	ADC 結果寄存器低位元組								0000 0000
093h	ADRESH	ADC 結果寄存器高位元組								0000 0000
094h	ADCON0	VHS1	VHS0	CHS3	CHS2	CHS1	CHS0	ADON	ADEN	0000 0000
095h	ADCON1	ADFM	ADCS2	ADCS1	ADCS0	-	-	-	ADREF	0000 ---0
096h	ADCLK	-	-	-	-	-	ADCLK2	ADCLK1	ADCLK0	---- -000
09Ah	PMDATL	程式記憶體讀數據寄存器的低位元組								0000 0000
09Bh	PMDATH	程式記憶體讀數據寄存器的高位元組								0000 0000
09Ch	PMADRL	程式記憶體讀位址寄存器的低位元組								0000 0000
09Dh	PMADRH	程式記憶體讀位址寄存器的高位元組								0000 0000
09Eh	PMCON	-	-	-	-	-	-	-	RDON	---- ---0
BANK1										
230h	SPSTAT	-	-	-	-	-	RXOV	MODF	WCOL	---- -000
231h	SPCTL	SSIG	SPEN	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	0000 0000
232h	SPDAT	SPI 資料寄存器								0000 0000
23Ah	BRT	串列傳輸速率發生器寄存器								0000 0000
23Bh	AUXR	-	UARTEN	UARTM0	BRTR	BRTX12	S1BRS	SMOD	SMOD0	-000 0000
23Ch	SCON	SM0/FE	SM1	SM2	REN	TR8	RB8	RXWK	-	0000 000-
23Dh	SBUF	串列口緩衝寄存器								0000 0000
23Eh	SADEN	從機位址寄存器								0000 -000
23Fh	SADDR	從機位址遮罩寄存器								00-0 0000
250h	PWM2DT	DT2.7	DT2.6	DT2.5	DT2.4	DT2.3	DT2.2	DT2.1	DT2.0	0000 0000
251h	PWM2D	PD2.7	PD2.6	PD2.5	PD2.4	PD2.3	PD2.2	PD2.1	PD2.0	0000 0000
252h	PWM2P	PP2.7	PP2.6	PP2.5	PP2.4	PP2.3	PP2.2	PP2.1	PP2.0	0000 0000
253h	PWM2C	-	-	-	-	PWM2S1	PWM2S0	CK21	CK20	---- 0000
254h	PWM1DT	DT1.7	DT1.6	DT1.5	DT1.4	DT1.3	DT1.2	DT1.1	DT1.0	0000 0000
255h	PWM1D	PD1.7	PD1.6	PD1.5	PD1.4	PD1.3	PD1.2	PD1.1	PD1.0	0000 0000

地址	名稱	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	復位初值
256h	PWM1P	PP1.7	PP1.6	PP1.5	PP1.4	PP1.3	PP1.2	PP1.1	PP1.0	0000 0000
257h	PWM1C	-	-	-	-	PWM1S1	PWM1S0	CK11	CK10	---- 0000
258h	PWM0DT	DT0.7	DT0.6	DT0.5	DT0.4	DT0.3	DT0.2	DT0.1	DT0.0	0000 0000
259h	PWM0DL	PD0.7	PD0.6	PD0.5	PD0.4	PD0.3	PD0.2	PD0.1	PD0.0	0000 0000
25Ah	PWM0DH	-	-	-	-	PD0.11	PD0.10	PD0.9	PD0.8	---- 0000
25Bh	PWM0PL	PP0.7	PP0.6	PP0.5	PP0.4	PP0.3	PP0.2	PP0.1	PP0.0	0000 0000
25Ch	PWM0PH	-	-	-	-	PP0.11	PP0.10	PP0.9	PP0.8	---- 0000
25Dh	PWM0C	-	-	FLTS	FLTC	PWM0S1	PWM0S0	CK01	CK00	--00 0000
25Eh	PWMEN	-	EFLT	EPWM21	EPWM11	EPWM01	EPWM2	EPWM1	EPWM0	-000 0000
25Fh	FLTM	-	-	FLT2M1	FLT2M0	FLT1M1	FLT1M0	FLT0M1	FLT0M0	--00 0000
2B0h	LCDCON	BIASCT1	BIASCT0	DUTCT	-	CS1	CS0	RLCD1	RLCD0	000- 0000
2B1h	LCDPS	-	-	-	-	LP3	LP2	LP1	LP0	---- 0000
2B2h	DISPCTR	-	-	-	-	-	SLPENB	DISPON	DISPCT	---- -000
2B3h	LCDLVD	-	-	-	-	LCDVD3	LCDVD2	LCDVD1	LCDVD0	---- 0000
2B6h	SEGSE0	SE7	SE6	SE5	SE4	SE3	SE2	SE1	SE0	0000 0000
2B7h	SEGSE1	SE15	SE14	SE13	SE12	SE11	SE10	SE9	SE8	0000 0000
2B8h	SEGSE2	SE23	SE22	SE21	SE20	SE19	SE18	SE17	SE16	0000 0000
2B9h	SEGSE3	SE31	SE30	SE29	SE28	SE27	SE26	SE25	SE24	0000 0000
2C0h	SEGDATA0	SEG0/C7	SEG0/C6	SEG0/C5	SEG0/C4	SEG0/C3	SEG0/C2	SEG0/C1	SEG0/C0	xxxx xxxx
2C1h	SEGDATA1	SEG1/C7	SEG1/C6	SEG1/C5	SEG1/C4	SEG1/C3	SEG1/C2	SEG1/C1	SEG1/C0	xxxx xxxx
2C2h	SEGDATA2	SEG2/C7	SEG2/C6	SEG2/C5	SEG2/C4	SEG2/C3	SEG2/C2	SEG2/C1	SEG2/C0	xxxx xxxx
2C3h	SEGDATA3	SEG3/C7	SEG3/C6	SEG3/C5	SEG3/C4	SEG3/C3	SEG3/C2	SEG3/C1	SEG3/C0	xxxx xxxx
2C4h	SEGDATA4	SEG4/C7	SEG4/C6	SEG4/C5	SEG4/C4	SEG4/C3	SEG4/C2	SEG4/C1	SEG4/C0	xxxx xxxx
2C5h	SEGDATA5	SEG5/C7	SEG5/C6	SEG5/C5	SEG5/C4	SEG5/C3	SEG5/C2	SEG5/C1	SEG5/C0	xxxx xxxx
2C6h	SEGDATA6	SEG6/C7	SEG6/C6	SEG6/C5	SEG6/C4	SEG6/C3	SEG6/C2	SEG6/C1	SEG6/C0	xxxx xxxx
2C7h	SEGDATA7	SEG7/C7	SEG7/C6	SEG7/C5	SEG7/C4	SEG7/C3	SEG7/C2	SEG7/C1	SEG7/C0	xxxx xxxx
2C8h	SEGDATA8	SEG8/C7	SEG8/C6	SEG8/C5	SEG8/C4	SEG8/C3	SEG8/C2	SEG8/C1	SEG8/C0	xxxx xxxx
2C9h	SEGDATA9	SEG9/C7	SEG9/C6	SEG9/C5	SEG9/C4	SEG9/C3	SEG9/C2	SEG9/C1	SEG9/C0	xxxx xxxx
2CAh	SEGDATA10	SEG10/C7	SEG10/C6	SEG10/C5	SEG10/C4	SEG10/C3	SEG10/C2	SEG10/C1	SEG10/C0	xxxx xxxx
2CBh	SEGDATA11	SEG11/C7	SEG11/C6	SEG11/C5	SEG11/C4	SEG11/C3	SEG11/C2	SEG11/C1	SEG11/C0	xxxx xxxx
2CCh	SEGDATA12	SEG12/C7	SEG12/C6	SEG12/C5	SEG12/C4	SEG12/C3	SEG12/C2	SEG12/C1	SEG12/C0	xxxx xxxx
2CDh	SEGDATA13	SEG13/C7	SEG13/C6	SEG13/C5	SEG13/C4	SEG13/C3	SEG13/C2	SEG13/C1	SEG13/C0	xxxx xxxx
2CEh	SEGDATA14	SEG14/C7	SEG14/C6	SEG14/C5	SEG14/C4	SEG14/C3	SEG14/C2	SEG14/C1	SEG14/C0	xxxx xxxx
2CFh	SEGDATA15	SEG15/C7	SEG15/C6	SEG15/C5	SEG15/C4	SEG15/C3	SEG15/C2	SEG15/C1	SEG15/C0	xxxx xxxx
2D0h	SEGDATA16	SEG16/C7	SEG16/C6	SEG16/C5	SEG16/C4	SEG16/C3	SEG16/C2	SEG16/C1	SEG16/C0	xxxx xxxx
2D1h	SEGDATA17	SEG17/C7	SEG17/C6	SEG17/C5	SEG17/C4	SEG17/C3	SEG17/C2	SEG17/C1	SEG17/C0	xxxx xxxx
2D2h	SEGDATA18	SEG18/C7	SEG18/C6	SEG18/C5	SEG18/C4	SEG18/C3	SEG18/C2	SEG18/C1	SEG18/C0	xxxx xxxx
2D3h	SEGDATA19	SEG19/C7	SEG19/C6	SEG19/C5	SEG19/C4	SEG19/C3	SEG19/C2	SEG19/C1	SEG19/C0	xxxx xxxx
2D4h	SEGDATA20	SEG20/C7	SEG20/C6	SEG20/C5	SEG20/C4	SEG20/C3	SEG20/C2	SEG20/C1	SEG20/C0	xxxx xxxx

地址	名稱	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	xxxx xxxx
2D5h	SEGDATA21	SEG21/C7	SEG21/C6	SEG21/C5	SEG21/C4	SEG21/C3	SEG21/C2	SEG21/C1	SEG21/C0	xxxx xxxx
2D6h	SEGDATA22	SEG22/C7	SEG22/C6	SEG22/C5	SEG22/C4	SEG22/C3	SEG22/C2	SEG22/C1	SEG22/C0	xxxx xxxx
2D7h	SEGDATA23	SEG23/C7	SEG23/C6	SEG23/C5	SEG23/C4	SEG23/C3	SEG23/C2	SEG23/C1	SEG23/C0	xxxx xxxx
2D8h	SEGDATA24	SEG24/C7	SEG24/C6	SEG24/C5	SEG24/C4	SEG24/C3	SEG24/C2	SEG24/C1	SEG24/C0	xxxx xxxx
2D9h	SEGDATA25	SEG25/C7	SEG25/C6	SEG25/C5	SEG25/C6	SEG25/C3	SEG25/C2	SEG25/C1	SEG25/C0	xxxx xxxx
2DAh	SEGDATA26	SEG26/C7	SEG26/C6	SEG26/C5	SEG26/C4	SEG26/C3	SEG26/C2	SEG26/C1	SEG26/C0	xxxx xxxx
2DBh	SEGDATA27	SEG27/C7	SEG27/C6	SEG27/C5	SEG27/C4	SEG27/C3	SEG27/C2	SEG27/C1	SEG27/C0	xxxx xxxx
2DCh	SEGDATA28	SEG28/C7	SEG28/C6	SEG28/C5	SEG28/C4	SEG28/C3	SEG28/C2	SEG28/C1	SEG28/C0	xxxx xxxx
2DDh	SEGDATA29	SEG29/C7	SEG29/C6	SEG29/C5	SEG29/C4	SEG29/C3	SEG29/C2	SEG29/C1	SEG29/C0	xxxx xxxx
2DEh	SEGDATA30	SEG30/C7	SEG30/C6	SEG30/C5	SEG30/C4	SEG30/C3	SEG30/C2	SEG30/C1	SEG30/C0	xxxx xxxx
2DFh	SEGDATA31	SEG31/C7	SEG31/C6	SEG31/C5	SEG31/C4	SEG31/C3	SEG31/C2	SEG31/C1	SEG31/C0	xxxx xxxx

注：x = 未知，u = 不變，q = 取值視條件而定，— = 未實現

3.1.4.2 累加器

8 位元資料寄存器W用來執行ALU與資料記憶體之間資料的傳送操作。如果操作結果為零（Z）或有進位產生（C或DC），程式狀態寄存器STATUS中相應位元會發生變化。

W 並不在RAM中，因此不可以用直接定址和間接定址模式對其進行讀寫。

➤ 例：W寄存器的讀寫操作

立即數寫入W寄存器操作：

```

MOVWL   H'0FF'      ;送十六進位數
MOVWL   D'10'       ;送十進位數字
MOVWL   B'11110000' ;送二進位數字
    
```

將W寄存器的資料寫入資料寄存器BUF中：

```
MOVWF   BUF
```

將資料寄存器BUF中的數讀入W寄存器：

```
MOVF   BUF,W
```

將W寄存器的資料與BUF中的資料加法運算後，結果存入BUF中：

```
ADDWF  BUF,F
```

3.1.4.3 INDFx寄存器

INDF0、INDF1 寄存器不是實際存在的寄存器，定址 INDF0、INDF1 將實現間接定址。

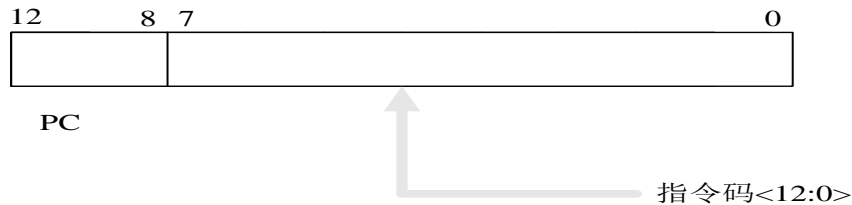
3.1.4.4 程式計數器

程式計數器（PC）為13位元寬，低位元組來自可讀寫的PCL寄存器，高位元組（PC[12:8]）不可讀寫，可通過PCLATH 寄存器間接寫入。

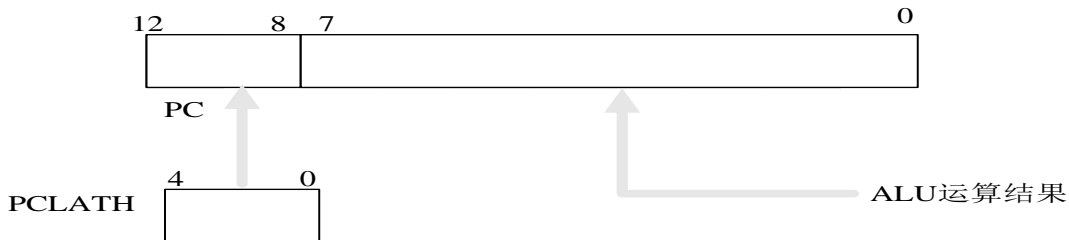
02h&202h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCL	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	0	0	0	0	0	0	0	0

08h&208h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCLATH	-	-	-	PCH12	PCH11	PCH10	PCH9	PCH8
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
POR的值	-	-	-	0	0	0	0	0

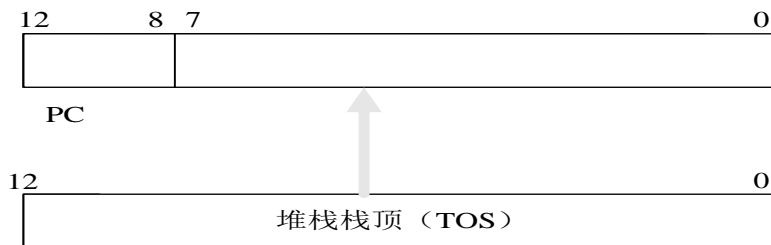
程式計數器順序指令運行時，每個運算速度程式自動加1，以下三種情況將使程式計數器重新裝載。分支指令（GOTO/CALL）：



以PCL作為目的運算元的指令：



副程式返回指令（RETURN/RETLW/RETFIE）：



3.1.4.5 STATUS寄存器

STATUS寄存器包含ALU的算術狀態、重定模式和寄存器的存儲區選擇位元。

03&203h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STATUS	-	-	RP0	TO	PD	Z	DC	C
R/W	-	-	R/W	R	R	R/W	R/W	R/W
POR的值	-	-	0	1	1	x	x	x

bit 5 **RP0**：BANK選擇位

1 = Bank1

0 = Bank0

bit 4 **TO**：超時位

1 = 上電、執行了CLRWDWT指令或SLEEP指令

0 = 發生了WDT溢出

bit 3 **PD**：掉電位

- 1 = 上電或執行了CLRWDWT指令
- 0 = 執行了SLEEP指令

bit 2 **Z**：結果為零位

- 1 = 算術或邏輯運算的結果為零
- 0 = 算術或邏輯運算的結果不為零

bit 1 **DC**：半進位/借位位

- 1 = 加法運算時低四位有進位/減法運算時沒有向高四位借位
- 0 = 加法運算時低四位沒有進位/減法運算時有向高四位借位

bit 0 **C**：進位/借位位

- 1 = 加法運算時有進位/減法運算時沒有借位發生/移位元後移出邏輯1
- 0 = 加法運算時沒有進位/減法運算時有借位發生/移位元後移出邏輯0

3.1.4.6 PCON寄存器

079h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCON	LVD2EN	LVD1EN	-	WDTENS	LVD2F	LVD1F	POR	BOR
R/W	R/W	R/W	-	R/W	R	R	R/W	R/W
POR的值	0	0	-	1	q	q	q	q

注：- = 未實現，q = 取值視條件而定

bit 7 **LVD2EN**:3.6V低電壓檢測使能位元

- 1 = 使能3.6V低電壓檢測
- 0 = 禁止3.6V低電壓檢測

bit 6 **LVD1EN**:2.4V低電壓檢測使能位元

- 1 = 使能2.4V低電壓檢測
- 0 = 禁止2.4V低電壓檢測

bit 4 **WDTENS**:看門狗軟體使能位元（需要配置字中使能WDT，該位使能時才有效）

- 1 = 軟體使能看門狗
- 0 = 軟體禁止看門狗

bit 3 **LVD2F**:3.6V低電壓檢測標誌位元

- 1 = 電壓低於3.6V
- 0 = 電壓高於3.6V

bit 2 **LVD1F**:2.4V低電壓檢測標誌位元

- 1 = 電壓低於2.4V
- 0 = 電壓高於2.4V

bit 1 **POR**:上電重定模式位

- 1 = 非上電復位
- 0 = 上電重定（需要軟體置1）

bit 0 **BOR**:欠壓重定模式位

- 1 = 未發生欠壓復位
- 0 = 發生了欠壓重定（需要軟體置1）

注：

- 1、在晶片執行 SLEEP 指令前，軟體關閉看門狗計時器，可以節省休眠或綠色模式下晶片功耗；
- 2、LVD2EN 和 LVD1EN 檢測電平值僅作為設計參考，不能用作晶片工作電壓值得精確檢測。

3.2 定址模式

HC18P23xL 共有三種定址方式：即時定位、直接定址和間接定址模式

3.2.1 即時定位

立即數參與運算的定址方式

- 例：即時定位

```
ADDLW    06h           ; W 的內容加 6，結果放入 W
```

3.2.2 直接定址

寄存器參與運算的定址方式

- 例：直接定址

```
MOVWF    OPTION       ; W 的內容裝入 OPTION
```

3.2.3 間接定址

由指標 FSR (FSR_{xL}/FSR_{xH}) 指向的寄存器參與運算的定址方式。INDF_x 寄存器不是物理寄存器，對 INDF_x 寄存器操作可以實現間接定址

- 例：利用間接定址對 100h~1FFh，300h~3FFh 通用資料記憶體進行清零

```

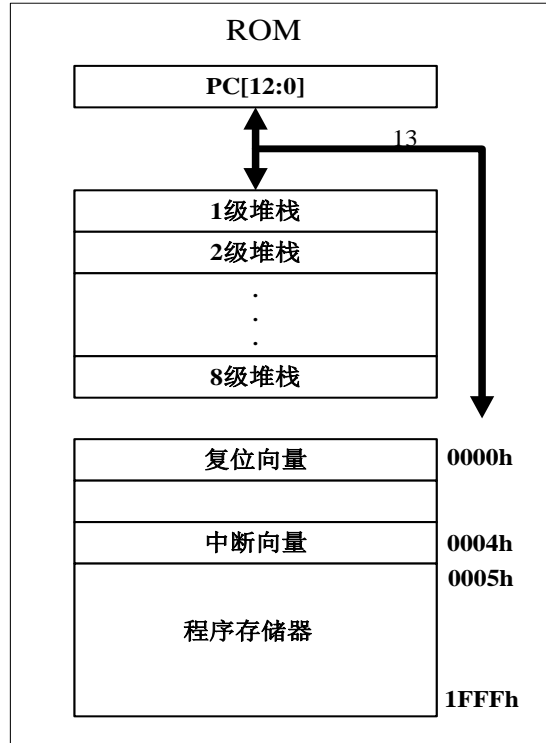
MOV LW    00h           ; 清零 0x100~0x1FF
MOV WF    FSR0L
MOV LW    0x01
MOV WF    FSR0H        ; FSR 指向 100h 地址
NEXTBYTE: CLRF          INDF0        ; 對 FSR 指向的資料記憶體清零
           INCFSZ       FSR,F        ; FSR0L+1，指向下一個位址
           ; 注意這裡的邊界值為欲操作 RAM 最大位址+1
           ; 利用間接定址，注意意外指向特殊寄存器的情况
           GOTO        NEXTBYTE     ; FSR0L 的值加一不溢出，迴圈清零下一個位址

MOV LW    00h           ; 清零 0x300~0x3FF
MOV WF    FSR1L        ; FSR 指向 300h 地址
MOV LW    0x03
MOV WF    FSR1H
NEXTBYTE_1: CLRF        INDF1        ; 對 FSR 指向的資料記憶體清零
            INCFSZ       FSR1L,F     ; FSR1L，指向下一個位址
            ; 注意這裡的邊界值為欲操作 RAM 最大位址+1
            ; 利用間接定址，注意意外指向特殊寄存器的情况
            GOTO        NEXTBYTE_1  ; FSR1L 的值加不溢出，迴圈清零下一個位址

CONTINUE: ...           ; 完成清零操作
    
```

3.3 堆疊

HC18P23xL 具有一個 8 級深度的硬體堆疊。當執行 CALL 指令或由於中斷導致程式跳轉時，PC 值會被壓入堆疊；當執行 RETURN、RETLW 或 RETFIE 指令時，PC 值從堆疊彈出。



注：

壓棧級數請勿超過 8 級，超過 8 級壓棧將導致堆疊溢位，溢出後堆疊指標迴圈，新的壓棧將覆蓋原堆疊內容。

4 復位

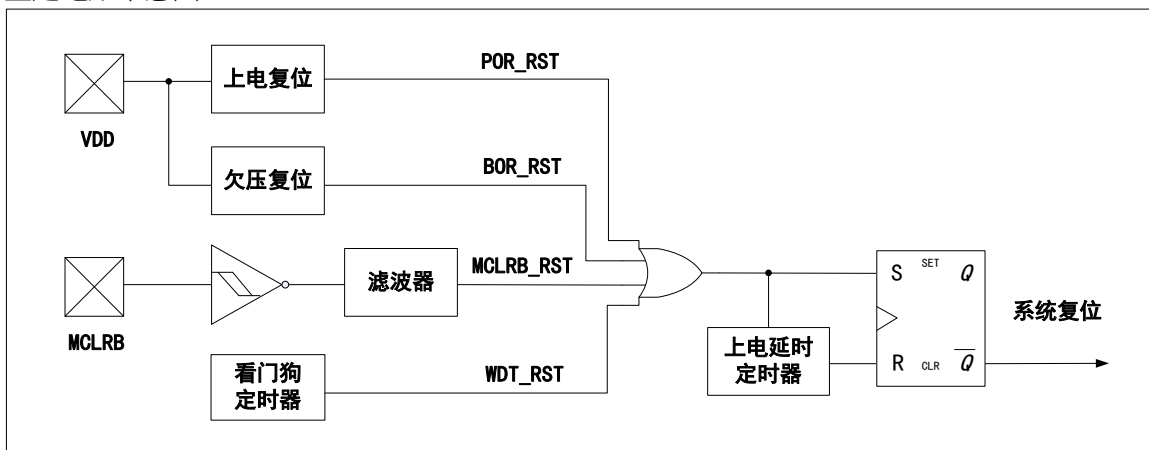
4.1 概述

HC18P23xL 共有四種重定方式：

- 上電復位 (POR)
- 外部復位 (MCLR Reset)
- 欠壓復位 (BOR)
- 看門狗計時器復位 (WDT Reset)

當上述任何一種重定產生時，系統進入重定模式，所有的特殊功能寄存器被初始化，程式停止運行，同時程式計數器 (PC) 清零。經過上電延時計時器延時後，系統結束重定模式，程式從 0000h 位址開始執行。STATUS 寄存器的 bit4 (TO 位) 及 PCON 寄存器的 bit0 (BOR 位)、bit1 (POR 位元) 顯示系統重定模式資訊，可通過 3 個標誌位元判斷重定來源，從而控制系統的運行路徑。

重定電路示意圖



特殊功能寄存器重定模式

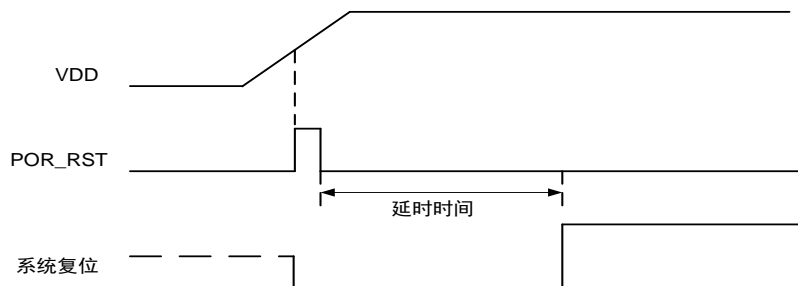
TO	POR	BOR	重定方式	說明
1	0	x	上電復位	電源上電。
u	u	0	欠壓復位	電源電壓低於BOR電壓點。
u	u	u	外部復位	外部復位管腳低電平。
0	u	u	看門狗計時器復位	運行模式下，看門狗計時器溢出。

重定方式	STATUS寄存器	PCON寄存器
上電復位	0001 1xxx	00-1 qq00
正常工作模式下的外部重定	0001 1xxx	00-1 qq0u
休眠模式下的外部重定	0001 0uuu	00-1 qquu
欠壓復位	0001 0uuu	00-1 qq00
看門狗計時器復位	0000 1uuu	00-1 qquu

注：u = 不變，x = 未知，- = 未使用，q = 取值視條件而定

4.2 上電復位

系統上電過程中，VDD 達到系統正常工作電壓之前，上電重定電路產生內部重定信號，可通過查詢 PCON 寄存器來判斷是否發生上電復位。VDD 最大上升時間 T_{VDD} 必須滿足規格要求。任何一種重定方式都需要一定的回應時間，系統提供完善的重定流程以保證重定動作的順利進行。對於不同類型的振盪器，完成重定所需要的時間也不同。因此，VDD 的上升速度和不同晶振的起振時間都不固定。RC 振盪器的起振時間最短，晶體振盪器的起振時間則較長。在使用者的使用過程中，應考慮系統對上電重定時間的要求。



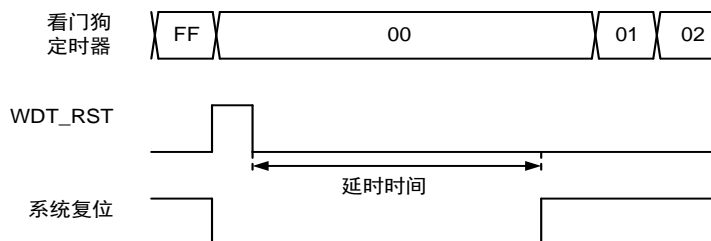
注：關於上電復位，請注意以下幾點：

- 1、 VDD 上電必須從 0V 開始，若 VDD 有殘留電壓，POR_RST 信號無法穩定產生；
- 2、 VDD 上電斜率必須滿足大於 500mV/ms，否則 POR_RST 信號可能無法產生；
- 3、 上電復位延時的重新延時時間在配置字裡選擇，兩個檔位（18ms/4.5ms）。

4.3 看門狗計時器復位

在高頻和低頻模式下，看門狗計時器溢出會產生WDT復位；在綠色和休眠模式下，看門狗計時器溢出將喚醒SLEEP並使其返回高頻或低頻模式，程式從SLEEP指令下一條開始執行。WDT計時器配置字和WDTENS都為1時，才能使能看門狗計時器。

看門狗復位示意圖：



注：關於看門狗復位使用時，請注意以下幾點：

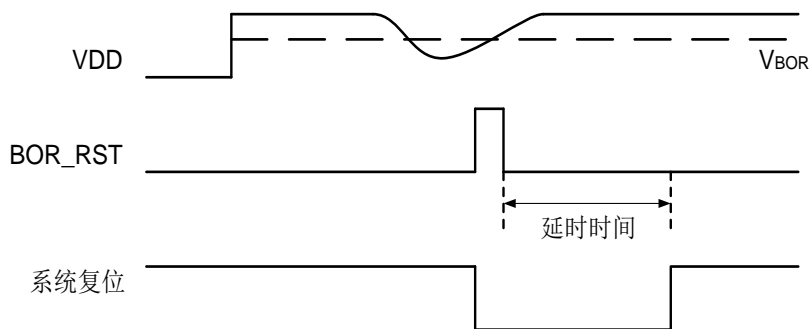
- 1、看門狗的使能邏輯：看門狗使能 = 看門狗配置字使能 & 看門狗軟體使能（WDTENS=1）；
- 2、不能在中斷程式中對看門狗進行清零，否則無法監控主程序跑飛情況；
- 3、程式中應該只在主程序中有一次清看門狗的動作，這種架構能夠最大限度的發揮看門狗的保護功能；
- 4、看門狗復位的延時時間為 2.2ms/1.1ms；
- 5、使用時注意：不論哪種方式重定後，看門狗軟體使能位元（WDTENS）的值为 1。

4.4 欠壓復位

4.4.1 欠壓復位的產生

當VDD電壓下降到 V_{BOR} 以下，且持續時間滿足，系統產生欠壓重定。

欠壓復位示意圖：



低電壓重定（BOR）是單片機內置的掉電重定保護裝置，當VDD 跌落並低於BOR 檢測電壓值時，BOR被觸發，系統重定。不同的單片機有不同的BOR 檢測電平。因此採用BOR 依賴於系統要求和環境狀況。如果電源跌落劇烈，遠低於BOR 觸發點，BOR 能夠起到保護作用，讓系統正常重定；如果電源電壓跌落不是很劇烈，僅僅是接近BOR 觸發點而造成的系統錯誤，則BOR 就不能起到保護作用讓系統重定。

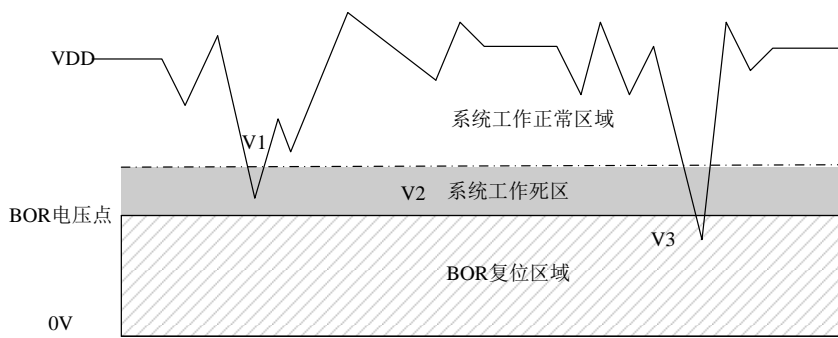
為避免電源較大的抖動，HC18P23xL採取必要的電源抖動處理電路或其他保護電路，防止電源抖動超過1.0V，導致晶片工作異常。

HC18P23xL通過配置字BOR編譯選項控制選擇低電壓檢測檔位元，請客戶在使用時根據情況選擇合適的BOR電壓。

BOR檔位：BOR3.6V/2.4V/2.0V

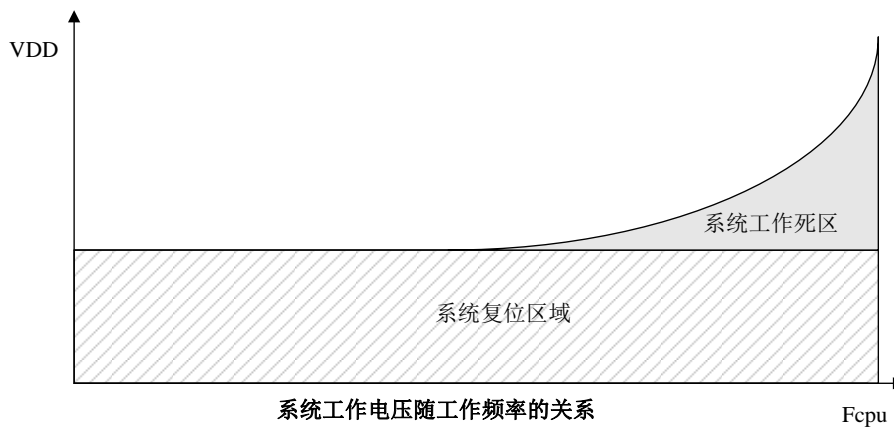
4.4.2 工作死區

電壓跌落可能會進入系統死區。系統死區意味著電源不能滿足系統的最小工作電壓要求。下圖是一個典型的掉電復位示意圖。圖中，VDD受到嚴重的干擾，電壓值降得非常低。虛線以上區域系統正常工作，在虛線以下的區域內，系統進入未知的工作狀態，這個區域稱作死區。當VDD跌至 V_1 時，系統仍處於正常狀態；當VDD跌至 V_2 時，系統進入死區，系統工作在死區時，可能導致程式的運行紊亂；當電壓跌至 V_3 ，且低於BOR電壓點，系統可正常重定，處於BOR電壓點的時間過短，系統仍無法正常產生欠壓重定信號，可能導致程式的運行紊亂。



4.4.3 工作死區與工作頻率的關係

工作死區電壓與工作速度相關，如下圖示意了死區與工作頻率的關係：



4.4.4 死區防護

對於死區防護，有以下幾點建議：

- 合理使用看門狗重定電路
- 降低系統的工作頻率
- 合理採用外部重定電路（電壓偏移重定電路、外部 IC 重定）

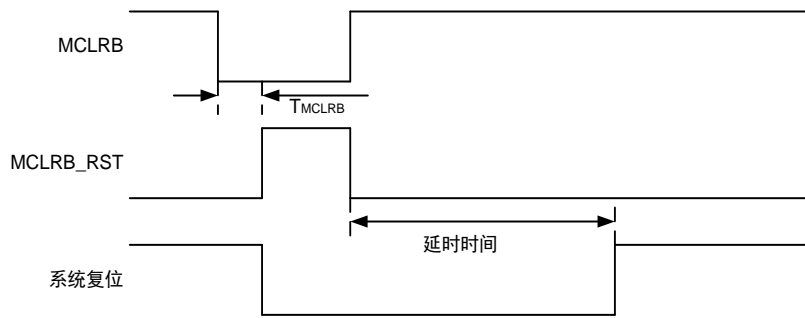
注：

二極體 RC 重定電路電壓偏移重定電路、外部 IC 重定防止系統進入死區。

4.5 外部復位

當外部復位埠 MCLR_B 輸入一個持續時間超過 T_{MCLR_B} 的低電平時，產生外部復位。MCLR_B 選擇配置字（編譯選項）為 1，MCLR_B 口為外部復位輸入口。

外部復位示意圖：



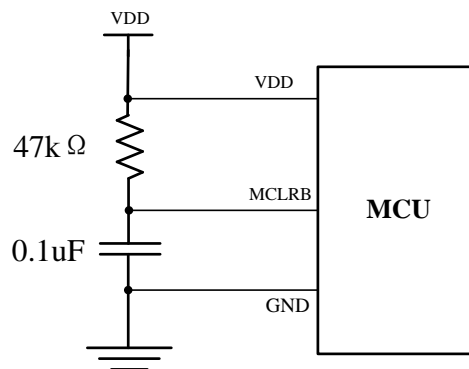
注：

T_{MCLR} 需大於 $200\mu s$ (典型值)；外部復位延時時間為 $2.2ms/1.1ms$ 。

4.5.1 外部 RC 重定電路

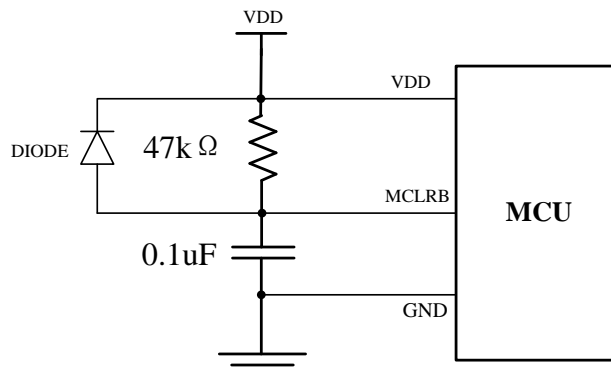
由電阻和電容組成的基本RC重定電路，它在系統上電的過程中能夠為復位引腳提供一個緩慢上升的重定信號。這個重定信號的上升速度低於VDD的上電速度，為系統提供合理重定時序，當復位引腳檢測到高電平時，系統重定結束，進入正常工作狀態。

如下圖：



4.5.2 二極體 RC 重定電路

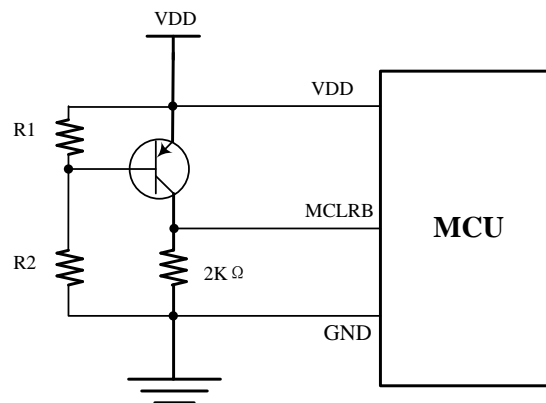
在基本RC重定電路上增加一個二極體 (DIODE)，對於電源異常情況，二極體正嚮導通使電容快速放電並與VDD保持一致，避免復位引腳持續高電平，系統無法正常重定。



4.5.3 電壓偏置重定電路

電壓偏置重定電路是一種簡單的電壓檢測重定電路，調整電壓檢測點，可以解決系統死區問題。電路中，R1和R2構成分壓電路，當R1和R2的分壓值高於三極管的開啟電壓時，三極管集電極輸出高電平，單片機正常工作；當R1和R2的分壓值低於三極管的開啟電壓時，集電極輸出低電平，MCU復位。

對於不同應用需求，選擇適當的分壓電阻。分壓電阻R1和R2在電路中要耗電，此處的功耗必須計入整個系統的功耗中。



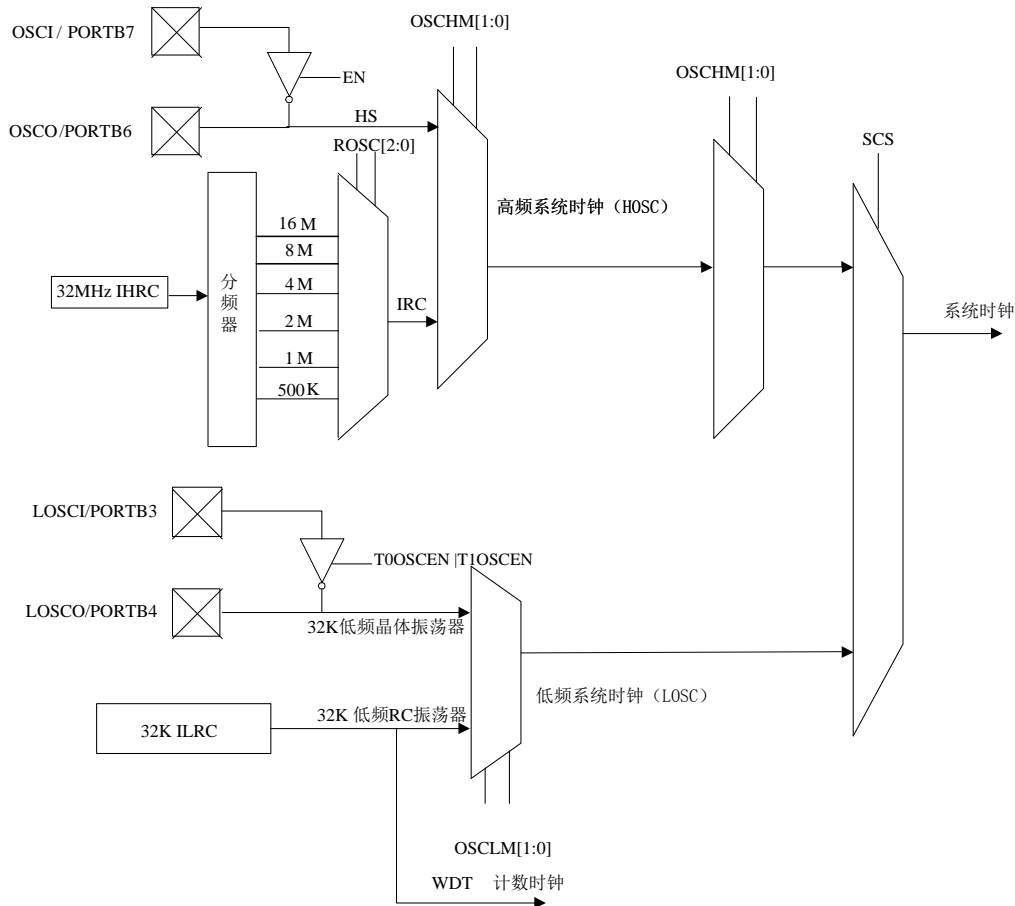
5 系統時鐘

5.1 概述

HC18P23xL內帶倍頻系統：高頻時鐘和低頻時鐘。高頻時鐘的時鐘源由高頻晶振或內部RC振盪電路（IRC 32MHz）提供。低頻時鐘的時鐘源則由低頻晶振或內部低速RC振盪電路（RC 32KHz@5V）提供。兩種時鐘都可作為系統時鐘源Fosc。OSCCON寄存器的SCS位元控制高頻時鐘和低頻時鐘之間切換。

- 高頻模式： $F_{cpu} = F_{sys} / N$ ，N = 2或4，時鐘模式選擇決定N的值。
- 低頻模式： $F_{cpu} = F_{sys} / N$ ，N = 2或4，時鐘模式選擇決定N的值。

5.2 時鐘框圖



- OSCHM[1:0]：高速系統時鐘選擇配置字
- OSCLM[1:0]：低速系統時鐘選擇配置字
- ROSC[2:0]：高速內部RC振盪器頻率選擇配置字
- Fosc：時鐘源頻率
- Fsys：系統時鐘頻率
- Fcpu：指令時鐘頻率

5.3 系統高頻時鐘

系統高頻時鐘有兩種選擇，通過OSCHM[1:0]高頻系統時鐘選擇配置字來控制。

高頻系統時鐘選擇配置字：

OSCHM[1:0]	說明
00	內部 RC 振盪器（IRC），OSCI/OSCO 作為普通 IO 口。
01	高頻晶體振盪器（HS），OSCI/OSCO 作為高頻晶體振盪器輸入/輸出口。 外部時鐘輸入，OSCI 作為外部時鐘輸入口，OSCO 作為外部時鐘輸出口。

5.3.1 內部高頻 RC 振盪器

配置字OSCHM[1:0]和ROSC[2:0]控制單片機的內置RC高速時鐘。OSCHM[1:0]若選擇“00”，則內置RC振盪器作為系統時鐘源，OSCI/OSCO作為通用I/O口。

內置RC高頻時鐘有16M/8M/4M/2M/1M /500K六種選擇。

高頻內部RC振盪器頻率選擇配置字

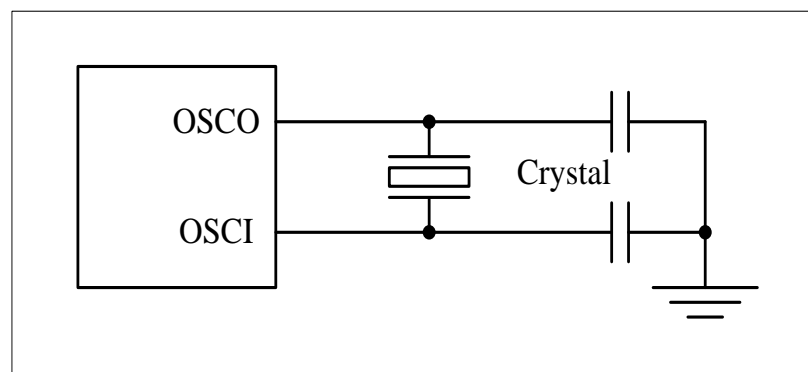
ROSC[2:0]	說明
110	內部RC振盪器頻率選擇16MHz
101	內部RC振盪器頻率選擇8MHz
100	內部RC振盪器頻率選擇4MHz
011	內部RC振盪器頻率選擇2MHz
010	內部RC振盪器頻率選擇1MHz
001	內部RC振盪器頻率選擇500KHz

5.3.2 外部高頻時鐘

外部高頻時鐘，由配置字 OSCHM 控制具體模式的選擇

- 高頻晶體振盪器：最高 20MHz

高頻晶體振盪器的頻率為1MHz~20MHz，推薦的典型值為4MHz、8MHz和16MHz，電容推薦值為20pF。



注：

OSCI 和 OSCO 引腳與振盪器和起振電容之間距離 10mm 以內。

5.4 系統低頻時鐘

高頻時鐘有兩種選擇，通過低頻時鐘選擇配置字來選擇。

- 低頻晶體振盪器： 32.768KHz
- 低頻 RC 振盪器： 32KHz (5V 典型值)

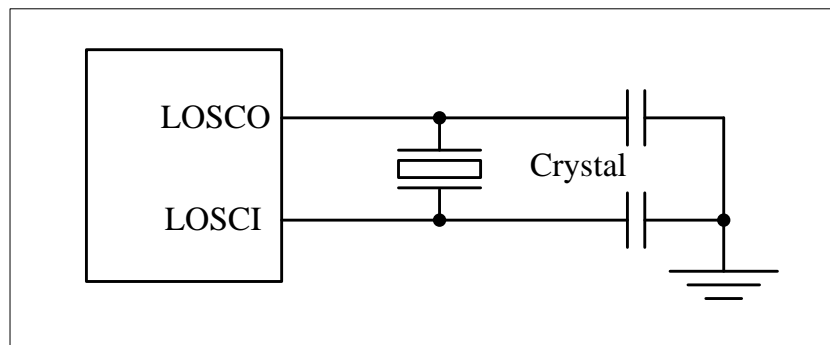
低頻系統時鐘選擇配置字

OSCLM[1:0]	說明
00	低頻 RC 振盪器，32KHz，LOSCI/LOSCO 作為輸入/輸出口
01	低頻晶體振盪器，32.768KHz，LOSCI/LOSCO 作為低頻晶體振盪器輸入/輸出口

5.4.1 低頻晶體振盪器

低頻晶體振盪器的頻率為32.768KHz，電容推薦值為20pF。

低頻晶體振盪器電路



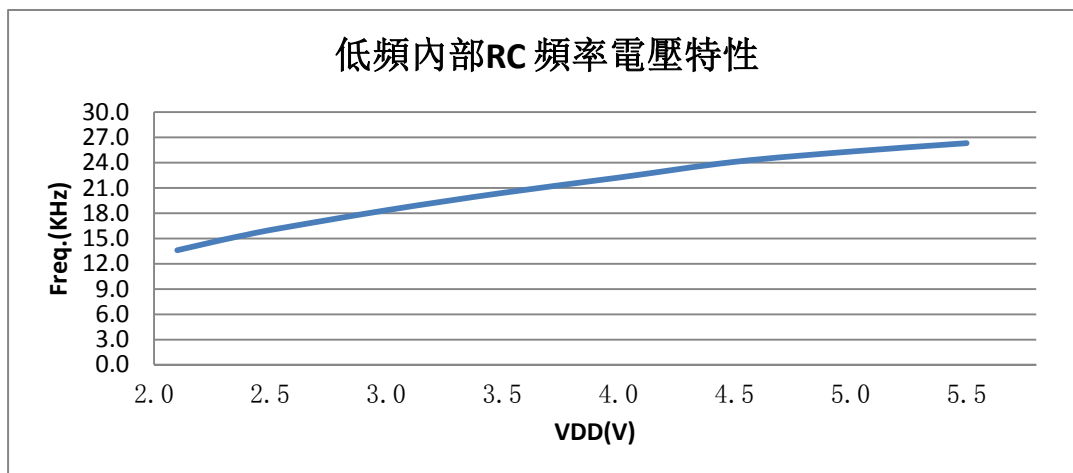
系統工作在綠色模式下，可以使能低頻晶體振盪器。

注：

外部高頻晶振接 OSCO、OSCI 埠，外部低頻晶振接 LOSCO、LOSCI 埠。

系統低頻時鐘源也可採用RC振盪電路。低頻內部RC振盪電路的輸出頻率受系統電壓和環境溫度的影響較大，通常為5V時輸出32KHz（典型值）。

輸出頻率與工作電壓之間的關係如下圖所示：



注：

低頻內部 RC 時鐘也用作看門狗計時器的時鐘。

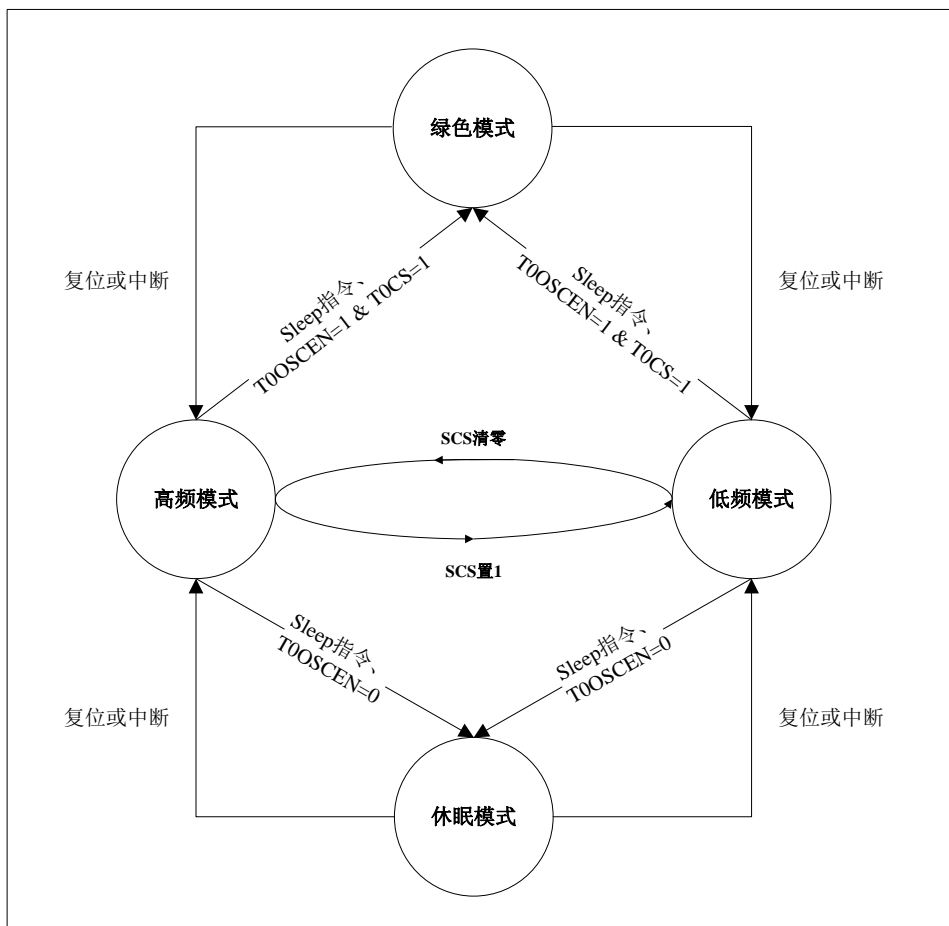
6 系統工作模式

6.1 概述

HC18P23xL可在以下四種工作模式之間進行切換：

- 高頻模式
- 低頻模式
- 休眠模式
- 綠色模式

系統重定後，工作於高頻模式還是低頻模式，由系統組態字決定。程式運行過程中，可以通過設置 SCS 位元使系統在高頻和低頻模式之間切換。



注：

1. 從休眠或綠色模式喚醒，中斷使能的情況則進入相應中斷，否則執行下一句；
2. 外部復位和 Timer2 中斷不能喚醒休眠或綠色模式；
3. 進入休眠或綠色模式前，關閉 WDT 可降低功耗。

各種模式下振盪器模組及Timer0/Timer1的工作狀態表

模組	高頻模式	低頻模式	綠色模式	休眠模式
高頻振盪器	運行	由HXEN決定	由HXEN決定	關閉
低頻振盪器	運行	運行	運行	關閉
Timer0	運行	運行	定時喚醒模式下運行	計數器模式下運行
Timer1	運行	運行	非同步定時喚醒模式下運行	非同步計數器模式下運行

6.2 模式切換舉例

- 例：高頻/低頻模式切換到睡眠模式。

```
BCF STATUS,RP0 ;Bank0
BCF OSCCON,T0SCEN
SLEEP
```

- 例：高頻模式切換到低頻模式。

```
BCF STATUS,RP0 ;Bank0
BSF OSCCON,SCS ;SCS = 1，系統進入低頻模式
```

- 例：從低頻模式切換到高頻模式。

```
BCF STATUS,RP0 ;Bank0
BCF OSCCON,SCS ;SCS = 0，系統進入高頻模式
```

- 例：從高頻/低頻模式切換到綠色模式

;T0計時器定時喚醒

```
BCF STATUS,RP0 ;Bank0
MOVLW 0x05
MOVWF OPTION
BSF OPTION,T0CS
BSF OSCCON,T0SCEN
BSF INTCON,T0IE ;使能T0計時器。
BSF INTCON,GIE
CLRF T0
SLEEP
```

- 例：從高頻/低頻模式切換到綠色模式。

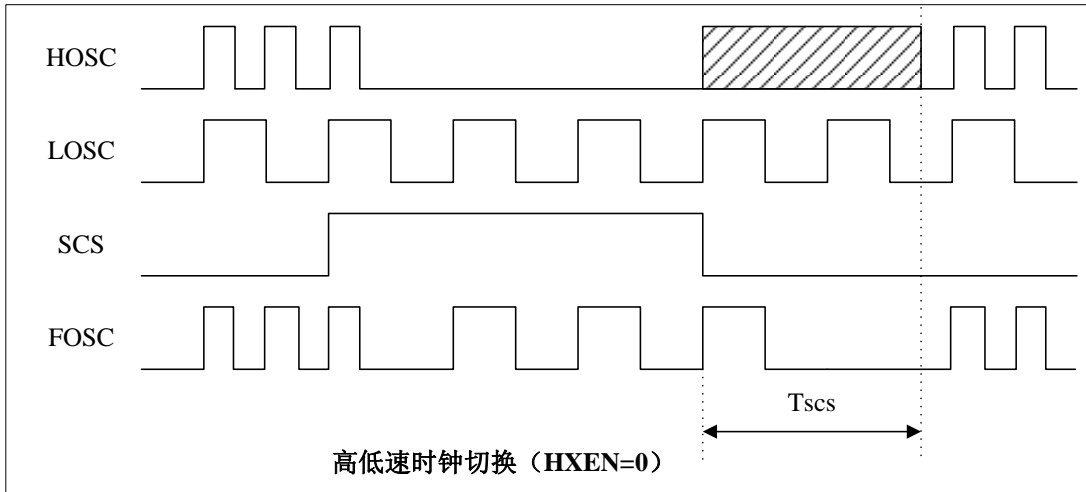
;T0計時器定時喚醒，OSCLM=01，低頻晶體振盪器為32.768KHz，定時喚醒時間為0.5s。

```
BCF STATUS,RP0 ;Bank0
MOVLW 0x05
MOVWF OPTION
BSF OPTION,T0CS
BSF OSCCON,T0SCEN
BCF INTCON,T0IF
BSF INTCON,T0IE ;使能T0計時器
BSF INTCON,GIE
CLRF T0
```

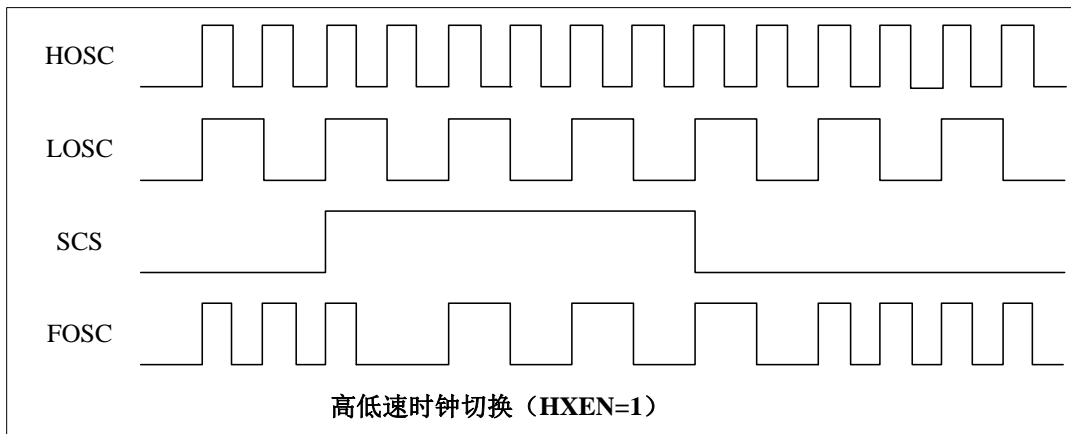
```

RTC_MODE:
    SLEEP
    BCF     STATUS,RP0           ;Bank0
    BCF     INTCON,T0IF         ;0.5s時間到
    ...
    GOTO    RTC_MODE
    
```

6.3 高低頻模式切換



高低頻切換時序：



時鐘切換時間 (Tscs) 計算：

$$T_{scs} = \text{高頻振盪器起振時間} + \text{高頻振盪器穩定時間}$$

不同類型高頻振盪器的穩定時間表

振盪器類型	高頻振盪器穩定時間
高頻晶體振盪器	1024 Clock
外部/內部 RC 振盪器	16 Clock

6.4 喚醒時間

系統進入休眠模式後，系統時鐘停止運行。外部中斷把系統從休眠模式下喚醒時，系統需要等待振

盪器起振計時器 (OST) 定時結束，以使振盪電路進入穩定工作狀態，等待的這一段時間稱為喚醒時間。喚醒時間結束後，系統進入高頻或低頻模式。

喚醒時間的計算如下：

$$\text{喚醒時間} = \text{起振時間} + \text{OST定時時間}$$

同類型振盪器OST定時時間表

振盪器類型	OST 定時時間
高/低頻晶體振盪器	1024 Clock
外/內部 RC 振盪器	16 Clock
低頻 RC 振盪器	4 Clock

注：系統進入綠色模式後，低頻時鐘正常運行。外部或內部中斷將系統從綠色模式中喚醒不需要喚醒時間。

6.5 OSCCON 寄存器

07Ah	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OSCCON	TOOSCEN	-	-	-	-	-	HXEN	SCS
R/W	R/W	-	-	-	-	-	R/W	R/W
POR的值	0	-	-	-	-	-	0	q

注：x = 未知，- = 未實現，q = 取值視條件而定

bit 7 TOOSCEN：低頻振盪器使能位

1 = 在低速或綠色模式下使能內部32K WDT振盪器

0 = 在低速或綠色模式下禁止內部32K WDT振盪器

bit 1 HXEN：高頻振盪器使能位

1 = 在低速或綠色模式下使能高頻振盪器

0 = 在低速或綠色模式下禁止高頻振盪器

bit 0 SCS：高低頻模式選擇位元

1 = 系統時鐘選擇為低頻系統時鐘

0 = 系統時鐘選擇為高頻系統時鐘

7 中斷

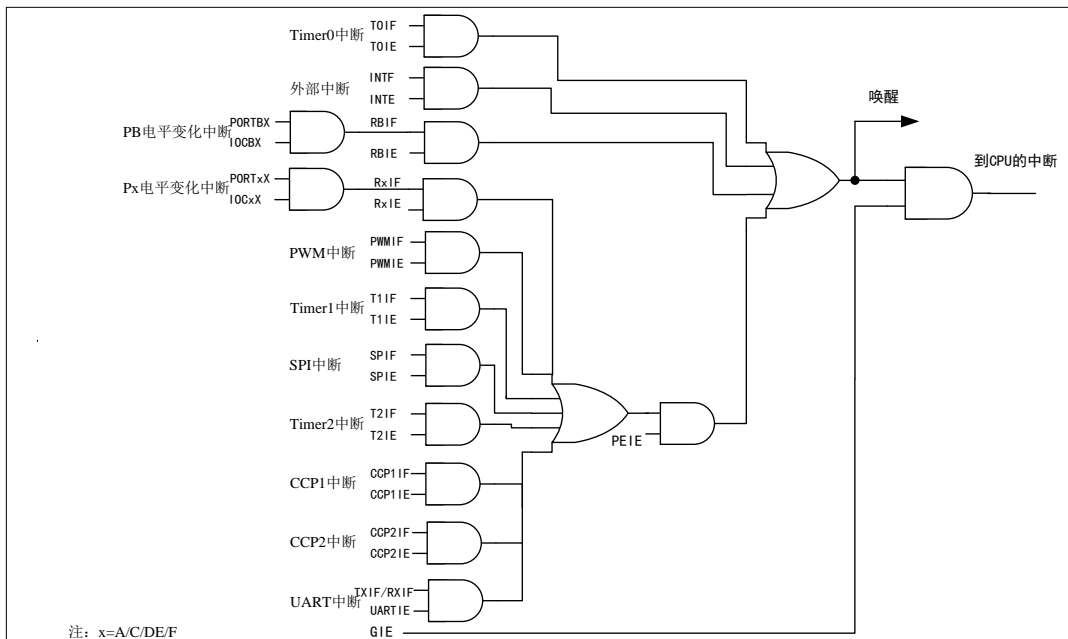
HC18P23xL中斷源：

- Timer0計時器中斷
- INT0外部中斷
- PORT口電平變化中斷
- Timer1計時器中斷
- Timer2計時器中斷
- CCP1中斷
- CCP2中斷
- AD中斷
- UART中斷
- SPI中斷
- PWM中斷

系統產生中斷時，程式計數器（PC）值壓入堆疊，程式跳轉至0004h，進入中斷服務程式。當程式運行到RETFIE指令時，系統退出中斷服務程式，程式計數器值出棧，系統執行PC+1位址對應的指令。

為避免誤進入中斷，在使能中斷和退出中斷服務程式之前，必須清除相應中斷標誌位元。在中斷服務程式中不需要軟體使能GIE，以免造成程式紊亂。

中斷示意圖



7.1 內核中斷

使能內核中斷必須將**GIE**和相應中斷的使能位置1，使能**PORTB**電平變化中斷還需要將相應埠配置為輸入並且**IOCB**的相應位置1。**INT0**外部中斷和**PORTB**電平變化中斷可以喚醒**SLEEP**，**Timer0**中斷在計數器模式和定時喚醒模式下可以喚醒**SLEEP**。

09h、209h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	0	0	0	0	0	0	0	x

注：x = 未知

bit 7 **GIE**：全域中斷使能位

1 = 使能所有未遮罩的中斷

0 = 禁止所有中斷

bit 5 **T0IE**：Timer0溢出中斷使能位

1 = 使能Timer0中斷

0 = 禁止Timer0中斷

bit 4 **INTE**：INT0外部中斷使能位

1 = 使能INT0外部中斷

0 = 禁止INT0外部中斷

bit 3 **RBIE**：PORTB電平變化中斷使能位

1 = 使能PORTB電平變化中斷

0 = 禁止PORTB電平變化中斷

bit 2 **T0IF**：Timer0溢出中斷標誌位元，Timer0計數寄存器在FFh至00h時產生溢出信號

1 = Timer0計數寄存器溢出（必須由軟體清零）

0 = Timer0計數寄存器未溢出

bit 1 **INTF**：INT0外部中斷標誌位元

1 = 發生INT0外部中斷（必須由軟體清零）

0 = 未發生INT0外部中斷

bit 0 **RBIF**：PORTB電平變化中斷標誌位元

1 = PORTB[7:0]中至少有一個口的電平狀態發生了改變（必須由軟體清零）

0 = PORTB[7:0]電平狀態沒有變化

078h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OPTION	-	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	-	1	1	1	1	1	1	1

bit 6 **INTEDG**：觸發INT0外部中斷的邊沿選擇位

1 = INT0引腳上升沿觸發中斷

0 = INT0引腳下降沿觸發中斷

7.2 外設中斷

使能外設中斷必須將**GIE**和**PEIE**置1，同時將相應中斷的使能位置1。Timer1門控事件中斷可以喚醒SLEEP，Timer1中斷在非同步計數器模式和非同步定時喚醒模式下可以喚醒SLEEP，CCPx中斷在捕捉模式下可以喚醒SLEEP。

09h、209h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	0	0	0	0	0	0	0	x

bit 7 **GIE**：全域中斷使能位

1 = 使能所有未遮罩的中斷

0 = 禁止所有中斷

bit 6 **PEIE**：外設中斷使能位

1 = 使能所有未遮罩的外設中斷

0 = 禁止所有外設中斷

070h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PIE1	RBPUB	ADIE	-	-	-	CCP1IE	T2IE	T1IE
R/W	R/W	R/W	-	-	-	R/W	R/W	R/W
POR的值	1	0	-	-	-	0	0	0

bit 6 **ADIE**：ADC中斷使能位

1 = 使能ADC中斷

0 = 禁止ADC中斷

bit 2 **CCP1IE**：CCP1中斷使能位

1 = 使能CCP1中斷

0 = 禁止CCP1中斷

bit 1 **T2IE**：Timer2計數寄存器與PR2匹配中斷使能位

1 = 使能Timer2匹配中斷

0 = 禁止Timer2匹配中斷

bit 0 **T1IE**：Timer1溢出中斷使能位

1 = 使能Timer1溢出中斷

0 = 禁止Timer1溢出中斷

071h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PIE2	PWM2IE	PWM1IE	PWM0IE	-	-	UARTIE	SPIE	CCP2IE
R/W	R/W	R/W	R/W	-	-	R/W	R/W	R/W
POR的值	0	0	0	-	-	0	0	0

bit [7:5] **PWMxIE**：PWM中斷使能位

1 = 使能PWM中斷

0 = 禁止PWM中斷

bit 2 **UARTIE**：UART中斷使能位

1 = 使能UART中斷

0 = 禁止UART中斷

bit 1 **SPIE** : SPI中斷使能位

1 = 使能SPI中斷

0 = 禁止SPI中斷

bit 0 **CCP2IE** : CCP2中斷使能位

1 = 使能CCP2中斷

0 = 禁止CCP2中斷

072h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PIE3	-	-	-	RFIE	REIE	RDIE	RCIE	RAIE
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
POR的值	-	-	-	0	0	0	0	0

Bit[4:0] **RxIE** : PORTx電平變化中斷使能位 (x=A/C/D/E/F)

1 = 使能PORTx電平變化中斷

0 = 禁止PORTx電平變化中斷

054h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PIR1	-	ADIF	-	-	-	CCP1IF	T2IF	T1IF
R/W	-	R/W	-	-	-	R/W	R/W	R/W
POR的值	-	0	-	-	-	0	0	0

055h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PIR2	PWM2IF	PWM1IF	PWM0IF	-	RXIF	TXIF	SPIF	CCP2IF
R/W	R/W	R/W	R/W	-	R/W	R/W	R/W	R/W
POR的值	0	0	0	-	0	0	0	0

054h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PIR3	-	-	-	RFIF	REIF	RDIF	RCIF	RAIF
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
POR的值	-	-	-	0	0	0	0	0

bit 6 **ADIF** : AD中斷標誌位元

1 = ADC轉換已完成 (必須由軟體清零)

0 = ADC轉換未完成或尚未開始

bit 2 **CCP1IF** : CCP1中斷標誌位元

捕捉模式 :

1 = 發生了捕捉事件 (必須用軟體清零)

0 = 未發生捕捉事件

比較模式 :

1 = 發生了比較事件 (必須用軟體清零)

0 = 未發生比較事件

PWM 模式 :

在此模式下未使用

- bit 1 **T2IF** : Timer2計數寄存器與PR2匹配中斷標誌位元
 1 = Timer2發生匹配 (必須用軟體清零)
 0 = Timer2未發生匹配
- bit 0 **T1IF** : Timer1溢出中斷標誌位元, Timer1計數寄存器在FFFFh至0000h時產生溢出信號
 1 = Timer1計數寄存器溢出 (必須由軟體清零)
 0 = Timer1計數寄存器未溢出
- bit [7:5] **PWMxIF** : PWM中斷標誌位元
 1 = PWM中斷產生中斷 (必須由軟體清零)
 0 = PWM中斷未產生中斷
- bit 3 **RXIF** : UART接收標誌位元
 1 = UART接收中斷產生中斷 (必須由軟體清零)
 0 = UART接收中斷未產生中斷
- bit 2 **TXIF** : UART發送中斷標誌位元
 1 = UART發送中斷產生中斷 (必須由軟體清零)
 0 = UART發送中斷未產生中斷
- bit 1 **SPIF** : SPI中斷標誌位元
 1 = SPI中斷產生中斷 (必須由軟體清零)
 0 = SPI中斷未產生中斷
- bit 0 **CCP2IF** : CCP2中斷標誌位元
 捕捉模式:
 1 = 發生了捕捉事件 (必須用軟體清零)
 0 = 未發生捕捉事件
 比較模式:
 1 = 發生了比較事件 (必須用軟體清零)
 0 = 未發生比較事件
 PWM 模式:
 在此模式下未使用
- bit [4:0] **RxIF** : PORTx電平變化中斷標誌位元(x=A/C/D/E/F)
 1 = PORTx[7:0]中至少有一個口的電平狀態發生了改變 (必須由軟體清零)
 0 = PORTx[7:0]電平狀態沒有變化

7.3 GIE 全域中斷

只有當全域中斷控制位GIE置“1”的時候程式才能回應插斷要求。一旦有中斷發生，程式計數器入棧，程式轉至中斷向量位址（ORG 0004H），堆疊層數加1。

例：設置全域中斷控制位（GIE）

```
BSF      INTCON,GIE      ;使能 GIE。
```

注：在所有中斷中，GIE 處於關閉狀態。

7.4 中斷保護

有插斷要求發生並被回應後，程式轉至0004H執行中斷服務程式。

中斷服務程式開始執行時，需保存W寄存器、STATUS寄存器、PCLATH寄存器的內容；結束中斷服務程式時，恢復PCLATH寄存器、STATUS寄存器、W寄存器的數值，注意順序。

注：

- 1.為了使保存系統寄存器的RAM，可進行特殊操作，保證保存相應的系統寄存器；
- 2.在退出中斷時，由於需要先恢復STATUS，再使用MOVWF指令恢復W，可能會改變STATUS，因此必須使用SWAPF指令恢復W。注意在中斷中共有兩句SWAPF指令。

➤ 例：對W、PCLATH 和STATUS 進行入棧保護。

```

ORG      0000H
GOTO    START
ORG      0004H
GOTO    INT_SERVICE
ORG      0010H

START:
...

INT_SERVICE:
MOVWF   W_TEMP           ;保存 W
SWAPF   STATUS,W
MOVWF   STATUS_TEMP      ;保存 STATUS
MOVF    PCLATH, W
MOVWF   PCLATH_TEMP     ;保存 PCLATH
CLRF    STATUS           ;切換到BANK0
...
MOVF    PCLATH_TEMP, W
MOVWF   PCLATH          ;恢復PCLATH
SWAPF   STATUS_TEMP,W
MOVWF   STATUS           ;恢復STATUS

```

```

SWAPF    W_TEMP,F
SWAPF    W_TEMP,W           ;恢復W
RETFIE                   ;退出中斷
...
END
    
```

7.5 Timer0 計時器中斷

T0 溢出時，無論 TOIE 處於何種狀態，TOIF 都會置 1。若 TOIE 和 TOIF 都置 1，且 GIE 使能，系統就會回應 TIMER0 的中斷；若 TOIE = 0，則無論 TOIF 是否置 1，系統都不會回應 TIMER0 中斷。

➤ 例：T0 插斷要求設置。

```

MOVLW    0xC5
BCF      STATUS,RP0           ;Bank0
MOVWF   OPTION                ;T0時鐘 = Fcpu / 64
MOVLW    0x40                 ;T0初始值 = 64H
BCF      STATUS,RP0
MOVWF   T0
BSF     INTCON,TOIE           ;置T0中斷使能標誌
BCF     INTCON,TOIF           ;清T0中斷標誌
BSF     INTCON,GIE            ;使能GIE
    
```

➤ 例：T0 中斷服務程式

```

ORG      0004H
GOTO     INT_SERVICE

INT_SERVICE:
MOVWF   W_TEMP                ;保存W
SWAPF   STATUS, W
MOVWF   STATUS_TEMP           ;保存STATUS
MOVF    PCLATH, W
MOVWF   PCLATH_TEMP           ;保存PCLATH
BCF     STATUS,RP0            ;BANK0
BTFSS   INTCON,TOIF           ;檢查是否有T0插斷要求標誌
GOTO     EXIT_INT              ;TOIF = 0，退出中斷

T0ISR:
BCF     INTCON,TOIF           ;清TOIF
MOVLW   0x40
MOVWF   T0                    ;重置T0值
...                                           ;T0中斷程式

EXIT_INT:
MOVF    PCLATH_TEMP, W
MOVWF   PCLATH                ;恢復PCLATH
SWAPF   STATUS_TEMP, W
MOVWF   STATUS                  ;恢復STATUS
SWAPF   W_TEMP, F
    
```

SWAPF W_TEMP,W ;恢復W
 RETFIE ;退出中斷

7.6 INTO 外部中斷

INT0 被觸發，則無論INTE 處於何種狀態，INTF 都會被置“1”。如果INTF=1 且INTE=1，**GIE**使能，系統回應該中斷;如果INTF=1 而INTE=0，系統並不會執行中斷服務。在處理多中斷時尤其需要注意。

078h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OPTION	RB PUB	INTE DG	T0CS	T0SE	PSA	PS2	PS1	PS0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	1	1	1	1	1	1	1	1

bit 6 **INTE DG**：觸發INT0外部中斷的邊沿選擇位

1 = INT0引腳上升沿觸發中斷

0 = INT0 引腳下降沿觸發中斷

➤ 例：INT0 插斷要求設置，電平觸發。

```

BCF        STATUS,RP0                ;Bank0
BSF        OPTION,INTEG              ;INT0 置為上升沿觸發
BCF        INTCON,INTF               ;INT0 插斷要求標誌清零
BSF        INTCON,INTE               ;使能INT0 中斷
BSF        INTCON,GIE                ;使能GIE
    
```

➤ 例：INT0 中斷。

```

ORG        0004H ;
GOTO       INT_SERVICE
    
```

INT_SERVICE:

```

...                                    ;保存STATUS、W 和PCLATH
BTFSS      INTCON,INTF               ;檢測INTF
GOTO       EXIT_INT                  ;INTF= 0，退出中斷
BCF        INTCON,INTF               ;INTF清零
...                                    ;INT0 中斷服務程式
    
```

EXIT_INT:

```

...                                    ;恢復STATUS、W和PCLATH
RETFIE                                ;退出中斷
    
```

7.7 PORT 電平變化中斷

PORTx電平變化中斷時，則無論RBIE處於何種狀態，RxIF都會被置“1”。如果RxIF=1 且RxIE=1，**GIE**使能，系統回應該中斷;如果RxIF=1 而RxIE=0，系統並不會執行中斷服務。

電平變化中斷必須將PORTx埠設為輸入，並將寄存器IOCx對應位置“1”。

041h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IOCB	IOCB7	IOCB6	IOCB5	IOCB4	IOCB3	IOCB2	IOCB1	IOCB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	0	0	0	0	0	0	0	0

IOCB[7:0]：PORTB[7:0]電平變化中斷使能控制位。

0 = 該埠禁止電平變化中斷；

1 = 該埠使能電平變化中斷。

040h~045h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IOCx	IOCx7	IOCx6	IOCx5	IOCx4	IOCx3	IOCx2	IOCx1	IOCx0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	0	0	0	0	0	0	0	0

IOCx[7:0]：PORTx[7:0]電平變化中斷使能控制位。

0 = 該埠禁止電平變化中斷；

1 = 該埠使能電平變化中斷。

x = A、C、D、E、F

- 注：1.如要允許 PORTB 口電平變化中斷必須將 IOCB 的對應埠的位置 1；
 2.PORTB 電平變化中斷中，在清零 RBIF 之前必須執行 PORTB 埠讀操作；
 3.PORTA/C/D/E/F 需使能 PEIE。

➤ 例：PORTB1 電平變化插斷要求設置。

```

MOVLW    0x02
BCF      STATUS,RP0           ;BANK0
IORWF    TRISB,F             ;PORTB1 埠為輸入
MOVLW    0x02
IORWF    IOCB,F              ;使能PORTB1埠為電平變化中斷
MOVF     PORTB, W            ;讀PORTB 口
BCF      INTCON,RBIF         ;PROTB 插斷要求標誌清零
BSF      INTCON,RBIE         ;使能PROTB 中斷
BSF      INTCON,GIE          ;使能GIE
    
```

➤ 例：PORTB 中斷。

```

ORG      0004H
GOTO     INT_SERVICE
    
```

INT_SERVICE:

```

...           ;保存STATUS、W 和PCLATH
BCF      STATUS,RP0           ;BANK0
BTFSS    INTCON,RBIF         ;檢測RBIF
GOTO     EXIT_INT            ;RBIF = 0，退出中斷
MOVF     PORTB,W            ;讀PORTB 埠
BCF      INTCON,RBIF         ;RBIF 清零
...           ;PORTB 電平變化中斷服務程式
...
    
```

EXIT_INT:

```
... ;恢復STATUS、W 和PCLATH
RETFIE ;退出中斷
```

➤ 例：PORTB 中斷喚醒SLEEP。

```
BCF STATUS,RP0 ;BANK0
MOVLW 0x02
IORWF TRISB,F ;PORTB1 埠為輸入
MOVLW 0x02
IORWF IOCB,F ;使能PORTB1埠為電平變化中斷
MOVF PORTB,W ;讀PORTB口
BCF INTCON,RBIF ;PROTB插斷要求標誌清零
BSF INTCON,RBIE ;使能PROTB中斷
SLEEP
BCF INTCON,RBIE
MOVF PORTB,W ;讀PORTB埠
... ;其他程式
```

注：1.如要允許 PORTx 口電平變化中斷必須將 IOCx 的對應埠的位置 1；
2.PORTx 電平變化中斷中，在清零 R_xIF 之前必須執行 PORTx 埠讀操作。

7.8 Timer1 中斷

T1溢出時，無論T1IE 處於何種狀態，T1IF都會置“1”。若T1IE 和T1IF 都置“1”，且PEIE、GIE均使能，系統就會回應TIMER1的中斷;若T1IE = 0，則無論T1IF 是否置“1”，系統都不會回應TIMER1中斷。

➤ 例：TIMER1工作於非同步計數模式，並中斷喚醒SLEEP

```
MOVLW 0xA4
BCF STATUS,RP0 ;BANK0
MOVWF T1CON ;T1時鐘源為T1CKI;分頻比為1:4;非同步
計數器模式
MOVLW nnH
MOVWF T1H
MOVLW nnH
MOVWF T1L ;Timer1賦初值
MOVLW 0xC0
MOVWF INTCON ;使能外設中斷
BSF PIE1,T1IE
BCF STATUS,RP0
BCF PIR1,T1IF
BSF T1CON,T1ON
BCF OSCCON,T0OSCEN ;禁止低頻晶體振盪器
SLEEP ;進入SLEEP
```

```
T1INT_SERVICE:
    ...
    BCF     STATUS,RP0      ;保存STATUS、W 和PCLATH
    BTFSS   PIR1,T1IF      ;BANK0
    GOTO    EXIT_INT       ;檢測T1IF
    BCF     PIR1,T1IF      ;T1IF = 0，退出中斷
    ...
    ...                    ;T1IF 清零。
    ...                    ;TIMER1 中斷服務程式
    ...

EXIT_INT:
    ...
    RETFIE                  ;恢復STATUS、W和PCLATH
    ...                    ;退出中斷
```

7.9 Timer2 計時器中斷

當T2的值和PR2的值相同時，TIMER2中斷被觸發，則無論T2IE 處於何種狀態，T2IF 都會被置“1”。如果T2IF=1 且T2IE=1，且PEIE、GIE均使能，系統回應該中斷；如果T2IF=1 而T2IE=0，系統並不會執行中斷服務。

➤ 例：TIMER2 插斷要求設置

```
BCF     STATUS,RP0      ;BANK0
MOVLW   0xFF
MOVWF   PR2             ;設置T2 週期
MOVLW   0x04
MOVWF   T2CON          ;設置分頻比
CLRF    T2
BSF     PIE1,T2IE      ;使能TIMER2 中斷
BSF     INTCON,GIE
BSF     T2CON,T2ON    ;使能TIMER2
```

➤ 例：TIMER2 中斷。

```
ORG     0004H
GOTO    T2INT_SERVICE
```

```
T2INT_SERVICE:
    ...
    BCF     STATUS,RP0      ;保存STATUS、W 和PCLATH。
    BTFSS   PIR1,T2IF      ;BANK0
    GOTO    EXIT_INT       ;檢測T2IF
    BCF     PIR1,T2IF      ;T2IF = 0，退出中斷
    ...
    ...                    ;T2IF 清零
    ...                    ;TIMER2 中斷服務程式
    ...

EXIT_INT:
    ...
    ...                    ;恢復STATUS、W和PCLATH
```

7.10 AD 中斷

當 ADC 完成，ADON 被硬體清零，無論 ADIE 處於何種狀態，與此同時 ADIF 被置“1”。若 ADIE、ADIF 為“1”，且 PEIE、GIE 均使能，系統就會相應 ADC 中斷；若 ADIE = 0，則無論 ADIF 是否置“1”，系統都不會回應 ADC 中斷。

7.11 CCP 中斷

當發生 CCPx 中斷時，無論 CCPxIE 處於何種狀態，CCPxIF 被置“1”。若 CCPxIE、CCPxIF 為“1”，且 PEIE、GIE 均使能，系統就會相應 CCPx 中斷；若 CCPxIE = 0，則無論 CCPxIF 是否置“1”，系統都不會回應 CCPx 中斷。

7.12 UART 中斷

當 UART 接收、發送完成後，無論 UARTIE 處於何種狀態，TXIF、RXIF 都被置“1”。若 UARTIE、TXIF、RXIF 為“1”，且 PEIE、GIE 均使能，系統就會相應產生 UART 中斷。

7.13 SPI 中斷

當 SPI 接收、發送完成後，無論 SPIE 處於何種狀態，SPIF 都被置“1”。若 SPIE、SPIF 為“1”，且 PEIE、GIE 均使能，系統就會相應產生 SPI 中斷。

7.14 PWM 中斷

當 PWMx 週期計數器溢出，無論 PWMxIE 處於何種狀態，PWMxIF 都被置“1”。若 PWMxIE 為“1”，且 PEIE、GIE 均使能，系統就會相應產生 PWM 中斷。

7.15 多中斷操作

在同一時刻，系統中可能出現多個插斷要求。此時，使用者必須根據系統的要求對各中斷進行優先權的設置。插斷要求標誌 IF 由中斷事件觸發，當 IF 處於有效值“1”時，系統並不一定會回應該中斷。各中斷觸發事件如下表所示：

中斷	有效觸發
TOIF	TO溢出
INTF	由INTEDG控制

RBIF	PORTB電平變化
RxIF	其他PORT口電平變化中斷
T2IF	T2的值和PR2相同
TXIF/RXIF	UART發生發送接收事件
SPIF	SPI發生發送接收事件
PWMxIF	PWMx週期計數溢出中斷

多個中斷同時發生時，需要注意的是：首先，必須預先設定好各中斷的優先權；其次，利用IE和IF控制系統是否回應該中斷。在程式中，必須對中斷控制位元和插斷要求標誌進行檢測。

➤ 例：多中斷條件下檢測插斷要求。

```

ORG      0004H
GOTO    INT_SERVICE

INT_SERVICE:
...
;保存STATUS、W和PCLATH

INT0CHK:
...
;檢查是否有INT0插斷要求
;BANK0
BCF     STATUS,RP0
BTFSS  INTCON,INTE
;檢查是否使能INT0中斷
GOTO   INT0CHK
;跳到下一個中斷
BTFSC  INTCON,INTF
;檢查是否有INT0插斷要求
GOTO   INT0
;進入INT0 中斷

INTT0CHK:
...
;檢查是否有T0 插斷要求
;檢查是否使能T0中斷
BTFSS  INTCON,T0IE
GOTO   INTT2CHK
;跳到下一個中斷
BTFSC  INTCON,T0IF
;檢查是否有T0插斷要求
GOTO   INTT0
;進入T0 中斷

INTT2CHK:
...
;檢查是否有T2插斷要求
;BANK0
BCF     STATUS,RP0
BTFSS  PIE1,T2IE
;檢查是否使能T2中斷
GOTO   INTRBCHK
;跳到下一個中斷。
BCF     STATUS,RP0
;BANK0
BTFSC  PIR1,T2IF
;檢查是否有T2插斷要求
GOTO   INTT2
;進入T2中斷

INTRBCHK:
...
;檢查是否使能PB電平變化中斷
BTFSS  INTCON,RBIE
GOTO   INT_EXIT
;跳到中斷結束
BTFSC  INTCON,RBIF
;檢查是否有PB電平變化插斷要求
GOTO   INTRB
;進入PB電平變化中斷

INTT2:
BCF     PIR1,T2IF
...
;T2中斷處理常式
GOTO   INT_EXIT

INT_EXIT:
...
;恢復STATUS、W和PCLATH
RETFIE
;退出中斷
    
```


8 I/O口

HC18P23xL共有六組雙向埠：

- PORTA□
- PORTB□
- PORTC□
- PORTD□
- PORTE□
- PORTF□

8.1 I/O 口模式

010h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	1	1	1	1	1	1	1	1

011h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	1	1	1	1	1	1	1	1

012h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	1	1	1	1	1	1	1	1

013h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TRISD	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	1	1	1	1	1	1	1	1

014h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TRISE	TRISE7	TRISE6	TRISE5	TRISE4	TRISE3	TRISE2	TRISE1	TRISE0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	1	1	1	1	1	1	1	1

015h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TRISF	TRISF7	TRISF6	TRISF5	TRISF4	TRISF3	TRISF2	TRISF1	TRISF0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	1	1	1	1	1	1	1	1

TRISx[7:0]：PORTx[7:0]的輸入輸出控制位

1 = 輸入狀態

0 = 輸出狀態

注：1. PORTB5 可設置為開漏輸出。

08Ch	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ANSELL	ANSEL7	ANSEL6	-	ANSEL4	ANSEL3	ANSEL2	ANSEL1	ANSEL0
R/W	R/W	R/W	-	R/W	R/W	R/W	R/W	R/W
POR的值	1	1	-	1	1	1	1	1

08Dh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ANSELH	ANSEL15	ANSEL14	ANSEL13	ANSEL12	ANSEL11	ANSEL10	ANSEL9	ANSEL8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	1	1	1	1	1	1	1	1

ANSEL[15:0]：A/D引腳數模控制位

1：類比模式，作為類比信號口，僅可作為AD通道的模擬輸入

0：數位模式，作為數位輸入或輸出口

注：ANSEL 上電初始值為 B'xxx1 1111'，即作為模擬輸入。無論是否應用到 AD，均需要在上電後，對 IO 操作之前按需配置，否則 IO 口可能無法受控於對應的埠寄存器，狀態將不確定。

ANSEL[4:0]對應 AN4~AN0 (PA4~PA0)，ANSEL[7:6]對應 AN7、AN6 (PA7、PA6)；

ANSEL[8]對應 AN8 (即 PA5)；

ANSEL[13:9]對應 AN13~AN9 (PB4~PB0)，ANSEL[15:14]對應 AN15、AN14 (PB7、PB6)。

8.2 I/O 口上拉控制寄存器

028~02Dh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WPUx	WPUx7	WPUx6	WPUx5	WPUx4	WPUx3	WPUx2	WPUx1	WPUx0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	1	1	1	1	1	1	1	1

x = A、B、C、D、E、F

WPUx[7:0]：PORTx[7:0]的上拉使能位

1 = 上拉禁止

0 = 上拉使能

078h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OPTION	RBPUB	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	1	1	1	1	1	1	1	1

bit 7 **RBPUB**：PORTB上拉使能位

1 = PORTB上拉由WPUB決定

0 = 使能PORTB上拉(此時無論WPUB為何值PORTB都上拉)

注：1. 注意此處上拉控制寄存器邏輯，0 為使能，1 為禁止；

2. I/O 禁止懸空狀態，輸入狀態需設定內部上拉電阻。

8.3 I/O 口下拉控制寄存器

034h~039h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WPDx	WPDx7	WPDx6	WPDx5	WPDx4	WPDx3	WPDx2	WPDx1	WPDx0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	1	1	1	1	1	1	1	1

x = A、B、C、D、E、F

WPDx[7:0]：PORTx[7:0]的下拉使能位

1 = 下拉禁止

0 = 下拉使能

- 注：1. 注意此處下拉控制寄存器邏輯，0 為使能，1 為禁止；
 2. 當埠設置為輸出時，下拉無效；
 3. 當下拉打開時，上拉無效。

8.4 PORT 驅動控制寄存器

04Dh~051h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LxEN	LxEN7	LxEN6	LxEN5	LxEN4	LxEN3	LxEN2	LxEN1	LxEN0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	0	0	0	0	0	0	0	0

X=A/B/C/D/E/F

LxEN[7:0]：LxEN[7:0]PORT口驅動電流控制位

0 = 驅動電流不變

1 = 1/3源電流及漏電流

8.5 I/O 口資料寄存器

01Ch	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTA	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	x	x	x	x	x	x	x	x

01Dh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	x	x	x	x	x	x	x	x

01Eh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTC	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	x	x	x	x	x	x	x	x

01Fh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	x	x	x	x	x	x	x	x

020h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTE	PORTE7	PORTE6	PORTE5	PORTE4	PORTE3	PORTE2	PORTE1	PORTE0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	x	x	x	x	x	x	x	x

021h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTF	PORTF7	PORTF6	PORTF5	PORTF4	PORTF3	PORTF2	PORTF1	PORTF0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR的值	x	x	x	x	x	x	x	x

- 例：PORTA0做AD輸入，PORTA[4:1]做輸入，PORTB口輸出0xFF
- ```

..... ;其他初始化
BCF STATUS,RP0 ;Bank0
MOVLW B'00000001' ;PA0為類比信號口，其他為數位信號口
MOVWF ANSELL
CLRF ANSELH ;PB為數字口
MOVLW 0xFF
MOVWF TRISA ;PA口作為輸入口
CLRF PORTA ;清空PORTA

```

```

MOV LW 0x00
MOV WF TRISB ;設置PB口為輸出
MOV LW 0xFF
MOV WF PORTB ;PB口輸出高電平
TEST1 :
 ;下面訪問PORTA作為輸入檢測得到的資料
 BTFS PORTA,1 ;測試PORTA1輸入電平
 GOTO TEST1 ;PORTA1檢測到低電平
 GOTO TEST2 ;PORTA1檢測到高電平
TEST2 :
 ;其他程式

```

## 8.6 管腳配置寄存器

| 04Ch   | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3  | Bit 2  | Bit 1  | Bit 0  |
|--------|-------|-------|-------|-------|--------|--------|--------|--------|
| PORCTR | -     | -     | -     | CCPCT | SPPCT1 | SPPCT0 | UAPCT1 | UAPCT0 |
| R/W    | -     | -     | -     | R/W   | R/W    | R/W    | R/W    | R/W    |
| POR的值  | -     | -     | -     | 0     | 0      | 0      | 0      | 0      |

bit[1:0] UAPCT[1:0] UART 管腳配置位元

00 = RX 配置在 PORTC3，TX 配置在 PORTC2 (默認)

01 = RX 配置在 PORTF4，TX 配置在 PORTF5

10 = RX 配置在 PORTB0，TX 配置在 PORTB1

11 = RX 配置在 PORTD7，TX 配置在 PORTD6

bit[3:2] SPPCT[1:0] SPI 管腳配置位元

00 = SS/SCK/MISO/MOSI 配置在 PORTC7/ PORTC6/ PORTC5/ PORTC4 (默認)

01 = SS/SCK/MISO/MOSI 配置在 PORTF4/PORTF5/PORTF6/PORTF7

10 = SS/SCK/MISO/MOSI 配置在 PORTB0/ PORTB1/ PORTB2/ PORTB3

11 = SS/SCK/MISO/MOSI 配置在 PORTD7/ PORTD6/ PORTD5/ PORTD4

Bit[4]:CCPCT CCP 管腳配置為

0 = CCP1/CCP2 管腳配置在 PORTB2/PORTB3 (默認)

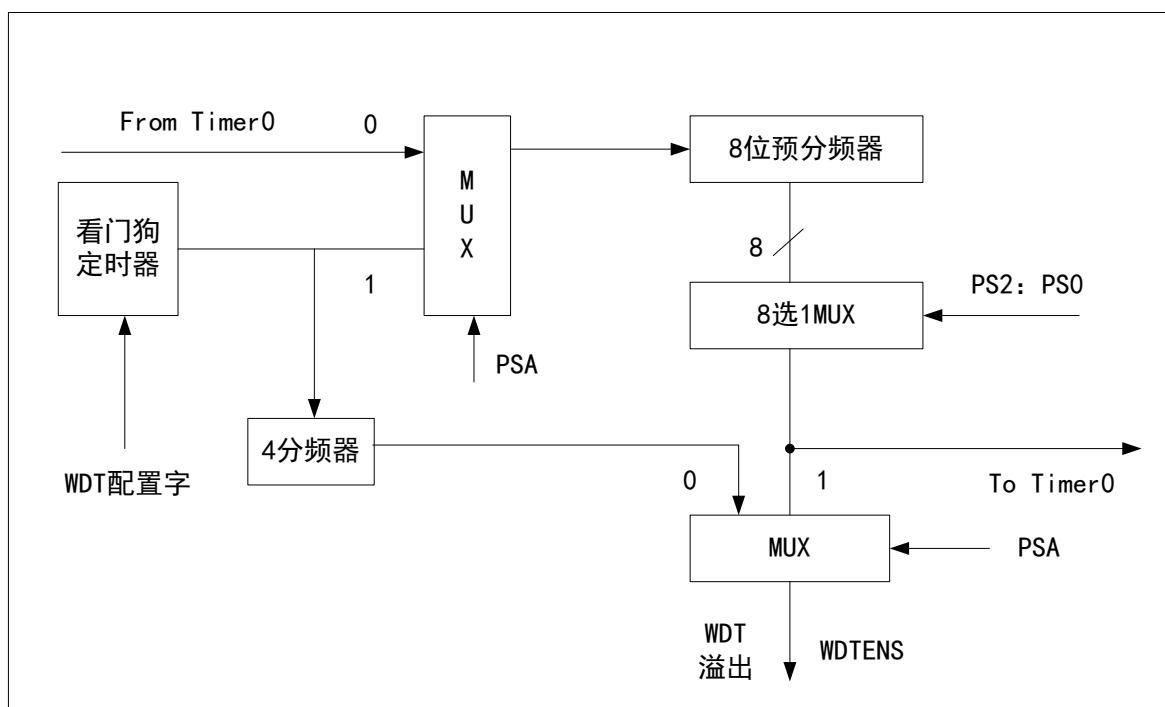
1 = CCP1/CCP2 管腳配置在 PORTA6/PORTA7

注：管腳複用功能優先順序：SPI > UART > CCP > IO。

# 9 計時器/計數器

## 9.1 看門狗計時器

HC18P23xL的看門狗計時器與Timer0計時器/計數器共用一個預分頻器。當PSA為0時，看門狗計時器每72ms（典型值）產生一個溢出信號；當PSA為1時，WDT溢出時間由預分頻器OPTION[2:0]設置決定，具體請參考9.2節 Timer0計時器/計數器。



| 079h  | Bit 7  | Bit 6  | Bit 5 | Bit 4  | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|--------|--------|-------|--------|-------|-------|-------|-------|
| PCON  | LVD2EN | LVD1EN | -     | WDTENS | LVD2F | LVD1F | POR   | BOR   |
| R/W   | R/W    | R/W    | -     | R/W    | R     | R     | R/W   | R/W   |
| POR的值 | 0      | 0      | -     | 1      | q     | q     | q     | q     |

bit 4 **WDTENS**：硬體看門狗軟體使能位元（需配置字使能看門狗，否則該位無效）

1 = 軟體使能硬體看門狗計時器

0 = 軟體遮罩硬體看門狗計時器

注：看門狗的使能邏輯 看門狗使能 = 晶片配置字使能（WDTEN） & 軟體使能（WDTENS）。

當系統處於休眠或綠色模式，看門狗計時器溢出將喚醒SLEEP並使其返回高頻或低頻模式，程式從SLEEP指令下一條開始執行。

注：

1. 對看門狗清零之前，檢查I/O口的狀態和RAM 的內容可增強程式的可靠性；
2. 不能在中斷中對看門狗清零，否則無法偵測到主程序跑飛的狀況；
3. 程式中應該只在主程序中有一次清看門狗的動作，這種架構能夠最大限度的發揮看門狗的保護功能。

➤ 例：看門狗在主程序中的應用

MAIN：

```

BCF STATUS,RP0 ; Bank 0
BSF PCON,WDTENS ;軟體使能WDT
... ;檢查IO狀態是否正確
... ;檢查RAM是否正確
GOTO ERR ;檢查IO/RAM出錯，進入出錯處理常式
CLRWDT
...
CALL SUB1
CALL SUB2
...
GOTO MAIN

```

➤ 例：在休眠狀態下，遮罩看門狗功能，可以節省系統功耗

```

...
BCF STATUS,RP0 ; Bank 0
BCF PCON,WDTENS ;軟體遮罩看門狗功能
BCF OSCCON,T0OSCEN ;禁止低頻晶體振盪器
SLEEP
BSF PCON,WDTENS ;喚醒後,重新使能看門狗功能
...

```

➤ 例：對看門狗計時器操作，看門狗計時器使能和清零

```

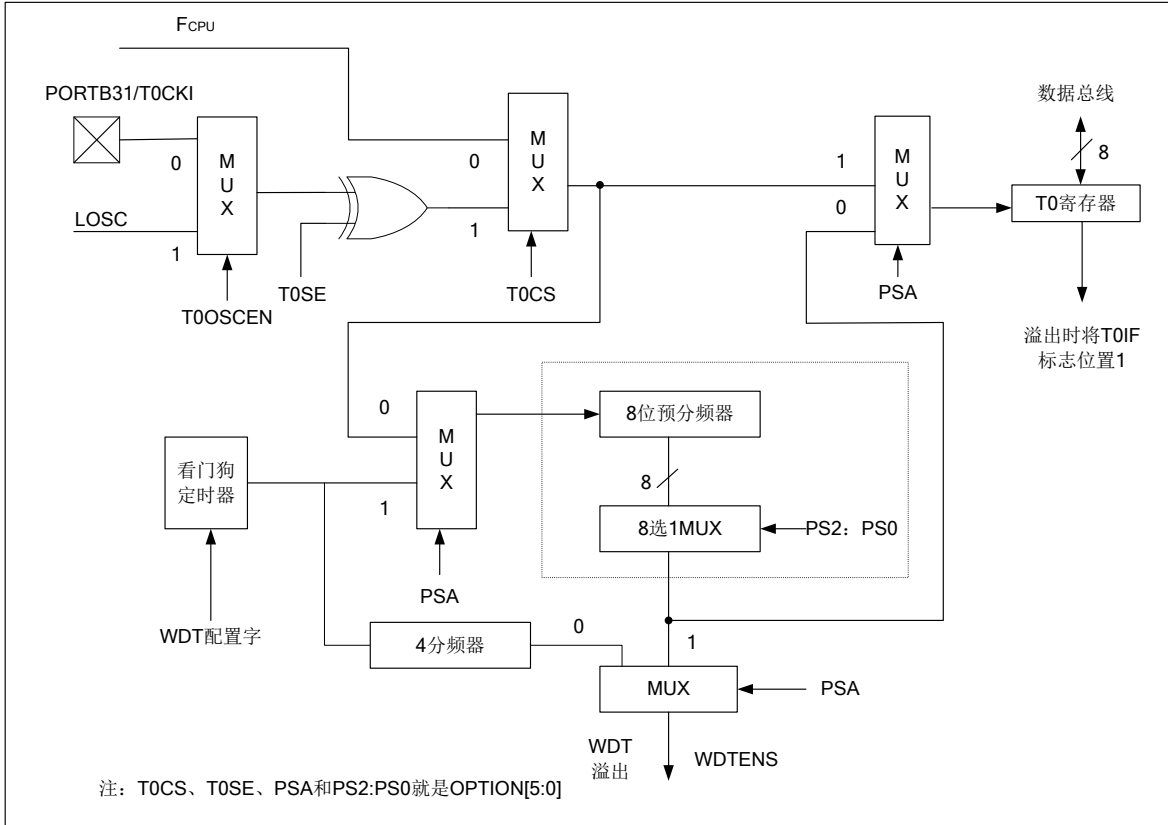
BCF STATUS,RP0 ; Bank 0
BSF PCON,WDTENS ;使能看門狗
CLRWDT
...

```

## 9.2 Timer0 計時器/計數器

Timer0 計時器/計數器模組具有如下功能：

- 8 位可程式設計計時器
- 外部事件計數器
- 綠色模式定時喚醒



| 07Ah   | Bit 7   | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|---------|-------|-------|-------|-------|-------|-------|-------|
| OSCCON | T0OSCEN | -     | -     | -     | -     | -     | HXEN  | SCS   |
| R/W    | R/W     | -     | -     | -     | -     | -     | R/W   | R/W   |
| POR的值  | 0       | -     | -     | -     | -     | -     | 0     | q     |

| 078h   | Bit 7 | Bit 6  | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|-------|--------|-------|-------|-------|-------|-------|-------|
| OPTION | RBPUB | INTEDG | T0CS  | T0SE  | PSA   | PS2   | PS1   | PS0   |
| R/W    | R/W   | R/W    | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR的值  | 1     | 1      | 1     | 1     | 1     | 1     | 1     | 1     |

看門狗計時器與Timer0計時器/計數器共用一個預分頻器，當PSA=1預分頻器分配給WDT時，Timer0在所選中時鐘源的每個週期遞增；當PSA=0預分頻器分配給Timer0時，Timer0根據PS[2:0]值選擇的預分頻時鐘遞增。

Timer0的預分頻器不可定址，當預分頻器分配給Timer0時，對Timer0計數寄存器的寫操作可以對預分頻器清零。

### 1、Timer0預分頻比選擇

| PS[2:0] | Timer0預分頻比 | WDT預分頻比 | WDT溢出時間（典型值） |
|---------|------------|---------|--------------|
| 000     | 1 : 2      | 1 : 1   | 18ms         |
| 001     | 1 : 4      | 1 : 2   | 36 ms        |
| 010     | 1 : 8      | 1 : 4   | 72ms         |
| 011     | 1 : 16     | 1 : 8   | 144ms        |
| 100     | 1 : 32     | 1 : 16  | 288ms        |



|     |         |         |        |
|-----|---------|---------|--------|
| 101 | 1 : 64  | 1 : 32  | 576ms  |
| 110 | 1 : 128 | 1 : 64  | 1.152s |
| 111 | 1 : 256 | 1 : 128 | 2.304s |

## 2、Timer0 工作模式選擇

| T0CS | T0OSCEN | T0SE | Timer0工作狀態                                      |
|------|---------|------|-------------------------------------------------|
| 0    | x       | x    | 計時器模式，計數時鐘 Fcpu，<br>休眠和綠色模式下停止                  |
| 1    | 0       | 0    | 計數器模式，計數時鐘 T0CKI，上升沿計數<br>休眠模式下工作，溢出中斷可喚醒 SLEEP |
| 1    | 0       | 1    | 計數器模式，計數時鐘 T0CKI，下降沿計數<br>休眠模式下工作，溢出中斷可喚醒 SLEEP |
| 1    | 1       | 0    | 定時喚醒模式，計數時鐘LOSC，上升沿計數<br>綠色模式下工作，溢出中斷可喚醒SLEEP   |
| 1    | 1       | 1    | 定時喚醒模式，計數時鐘LOSC，下降沿計數<br>綠色模式下工作，溢出中斷可喚醒SLEEP   |

### 注：

Timer0 工作模式的選擇需符合上表描述，選擇除上表以外情況可能會造成程式運行混亂，請謹慎操作。

- 例：Timer0工作於計時器模式，計數時鐘為Fcpu，T0計滿到FF後溢出進入中斷

```

MOVLW 0x01
BCF STATUS,RP0 ;Bank0
MOVWF OPTION ;計時器模式，分頻比為1:4
MOVLW 0x00
MOVWF T0 ;T0賦初值
BSF INTCON,T0IE
BCF INTCON,T0IF
BSF INTCON,GIE

```

TOINT\_SERVICE:

```

... ;保存STATUS、W 和PCLATH
BCF STATUS,RP0 ;Bank0
BTFSS INTCON,T0IF ;檢測T0IF
GOTO EXIT_INT ;T0IF = 0，退出中斷
BCF INTCON,T0IF ;T0IF 清零
... ;TIMER0 中斷服務程式

```

EXIT\_INT:

```

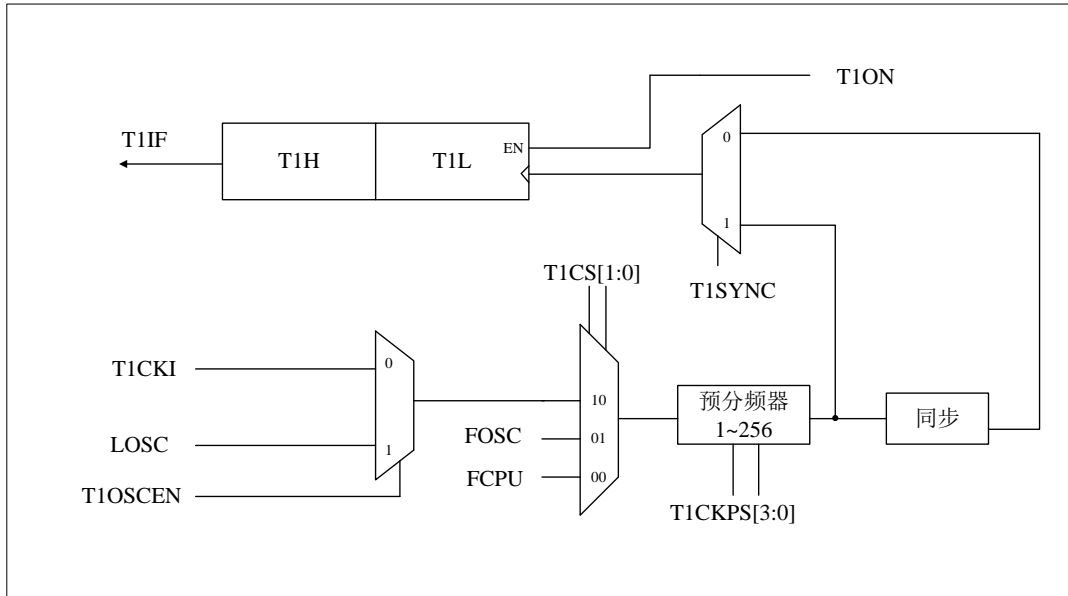
... ;恢復STATUS、W和PCLATH
RETFIE ;退出中斷

```

### 9.3 Timer1 計時器/計數器

Timer1 計時器/計數器模組具有如下功能：

- 16 位可程式設計計時器
- 外部事件計數器，可程式設計選擇同步、非同步功能
- 綠色模式定時喚醒



#### 9.3.1 Timer1 控制寄存器

|       |       |       |         |         |         |        |       |       |
|-------|-------|-------|---------|---------|---------|--------|-------|-------|
| 05Ah  | Bit 7 | Bit 6 | Bit 5   | Bit 4   | Bit 3   | Bit 2  | Bit 1 | Bit 0 |
| T1CON | T1CS1 | T1CS0 | T1CKPS1 | T1CKPS0 | T1OSCEN | T1SYNC | -     | T1ON  |
| R/W   | R/W   | R/W   | R/W     | R/W     | R/W     | R/W    | -     | R/W   |
| POR的值 | 0     | 0     | 0       | 0       | 0       | 0      | -     | 0     |

Timer1時鐘源選擇

| T1CS1 | T1CS0 | T1OSCEN | 時鐘源                      |
|-------|-------|---------|--------------------------|
| 0     | 0     | x       | 指令時鐘 (F <sub>cpu</sub> ) |
| 0     | 1     | x       | 系統時鐘 (F <sub>sys</sub> ) |
| 1     | 0     | 0       | TICK1引腳上的外部時鐘            |
| 1     | 0     | 1       | 低頻系統時鐘                   |

注：

Timer1 時鐘源的選擇需符合上表描述，選擇除上表以外情況會造成程式運行不正常，請謹慎操作。

Timer1輸入時鐘預分頻比選擇

| T1CKPS[1:0] | Timer1 預分頻比 |
|-------------|-------------|
| 00          | 1 : 1       |
| 01          | 1 : 2       |
| 10          | 1 : 4       |

|    |       |
|----|-------|
| 11 | 1 : 8 |
|----|-------|

Timer1的預分頻器不可定址，可以通過對Timer1計數寄存器寫操作將預分頻器清零。

Timer1工作模式選擇

| T1ON | T1CS[1:0] | T1OSCEN | T1SYNC | Timer1工作模式                     |
|------|-----------|---------|--------|--------------------------------|
| 1    | 00        | x       | x      | 計時器模式，休眠和綠色模式下停止               |
| 1    | 01        | x       | x      | 計時器模式，休眠和綠色模式下停止               |
| 1    | 10        | 0       | 0      | 同步計數器模式，休眠模式下停止                |
| 1    | 10        | 0       | 1      | 非同步計數器模式，休眠模式下工作，溢出中斷可喚醒SLEEP  |
| 1    | 10        | 1       | 0      | 同步定時喚醒模式，綠色模式下停止，溢出中斷不能喚醒SLEEP |
| 1    | 10        | 1       | 1      | 非同步定時喚醒模式，綠色模式下工作，溢出中斷可喚醒SLEEP |

注：

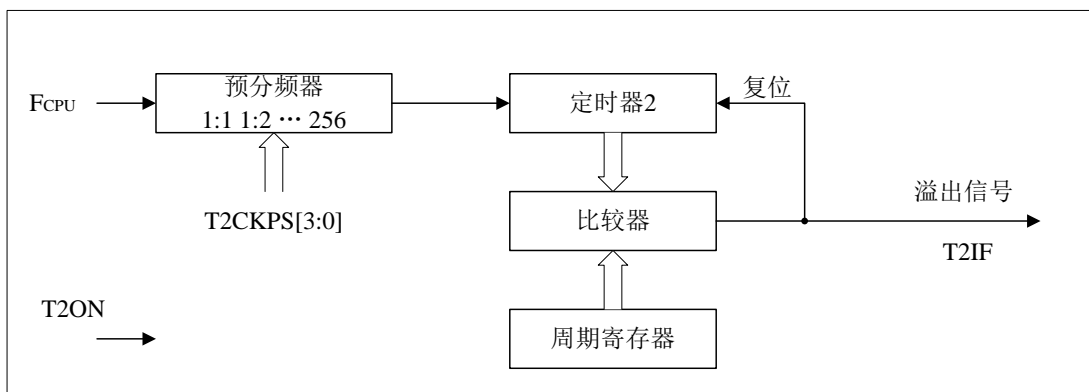
1、T1 為 16 位計時器，在溢出中斷重新賦值時應先 T1H，後 T1L，避免 T1L 在操作中的進位被覆蓋；清空時則應先 T1L 後 T1H，避免 T1L 進位意外進入 T1H 造成清空失敗；

2、Timer1 工作於同步計數器模式和同步定時喚醒模式時，不能喚醒 SLEEP 或綠色模式；

3、Timer1 工作模式的選擇需符合上表描述，選擇除上表以外情況可能會造成程式運行混亂，請謹慎操作。

## 9.4 Timer2 計時器

Timer2定時器具有8位預分頻器和8位週期寄存器( PR2)，Timer2計時器的輸入時鐘為指令時鐘FCPU，輸入時鐘通過預分頻器產生Timer2計數時鐘，當計數到與週期寄存器（PR2）的值相同時，在下一運算速度產生Timer2溢出信號，可根據實際需要選擇不同的預分頻比及設置週期寄存器的值，產生不同溢出時間。



## 9.4.1 Timer2 計時器

| 05Eh  | Bit 7 | Bit 6   | Bit 5   | Bit 4   | Bit 3   | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|---------|---------|---------|---------|-------|-------|-------|
| T2CON | -     | T2CKPS3 | T2CKPS2 | T2CKPS1 | T2CKPS0 | T2ON  | -     | -     |
| R/W   | -     | R/W     | R/W     | R/W     | R/W     | R/W   | -     | -     |
| POR的值 | -     | 0       | 0       | 0       | 0       | 0     | -     | -     |

bit 2 **T2ON** : Timer2模組使能位元

1 = 使能Timer2模組

0 = 禁止Timer2模組

Timer2具有一個8位可程式設計預分頻器，關閉Timer2模組和對Timer2計數寄存器或T2CON寄存器寫操作都將對預分頻器清零。

| T2CKPS[3:0] | Timer2 預分頻比 |
|-------------|-------------|
| 0000        | 1 : 1       |
| 0001        | 1 : 2       |
| 0010        | 1 : 4       |
| 0011        | 1 : 8       |
| 0100        | 1 : 16      |
| 0101        | 1 : 32      |
| 0110        | 1 : 64      |
| 0111        | 1 : 128     |
| 1xxx        | 1 : 256     |

## 9.4.2 Timer2 計數寄存器

| 05Ch  | Bit 7       | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------------|-------|-------|-------|-------|-------|-------|-------|
| T2    | Timer2計數寄存器 |       |       |       |       |       |       |       |
| R/W   | R/W         | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR的值 | 0           | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

## 9.4.3 Timer2 週期寄存器

| 05Dh  | Bit 7       | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------------|-------|-------|-------|-------|-------|-------|-------|
| PR2   | Timer2週期寄存器 |       |       |       |       |       |       |       |
| R/W   | R/W         | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR的值 | 1           | 1     | 1     | 1     | 1     | 1     | 1     | 1     |

Timer2 計時器的輸入時鐘為指令時鐘 Fcpu，輸入時鐘通過預分頻器產生 Timer2 計數信號，當計數到與週期寄存器（PR2）的值相同時產生 Timer2 溢出信號。

Timer2 溢出時間 = (PR2 + 1) × 預分頻比/Fcpu。

## 9.5 CCP 模組

HC18P23xL具有2個獨立的CCP模組CCP1和CCP2，每個CCP模組具有三種模式：

- 捕捉
- 比較
- PWM

CCP模組的時基由Timer1和Timer2提供。

### 1、CCP模組的時基

| CCP模式 | 時鐘源           |
|-------|---------------|
| 捕捉    | Timer1        |
| 比較    | Timer1        |
| PWM   | Timer2/Timer1 |

### 2、CCPxCON寄存器

| 082h、085h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3  | Bit 2  | Bit 1  | Bit 0  |
|-----------|-------|-------|-------|-------|--------|--------|--------|--------|
| CCPxCON   | -     | -     | DCxB1 | DCxB0 | CCPxM3 | CCPxM2 | CCPxM1 | CCPxM0 |
| R/W       | -     | -     | R/W   | R/W   | R/W    | R/W    | R/W    | R/W    |
| POR的值     | -     | -     | 0     | 0     | 0      | 0      | 0      | 0      |

bit 5~4 **DCxB[1:0]**：PWM占空比最低有效位

捕捉模式：未使用

比較模式：未使用

PWM模式：PWM占空比的低2位，高8位是CCPRxL寄存器

bit 3~0 **CCPxM[3:0]**：CCPx模式選擇位元

0000 = 捕捉/比較/PWM關閉（重定CCP模組）

0001 = 未使用（保留）

0010 = 比較模式，匹配時輸出翻轉電平（PIRx寄存器的CCPxIF位置1）

0011 = 未使用（保留）

0100 = 捕捉模式，每個下降沿

0101 = 捕捉模式，每個上升沿

0110 = 捕捉模式，每4個上升沿

0111 = 捕捉模式，每16個上升沿

1000 = 比較模式，匹配時輸出高電平（PIRx寄存器的CCPxIF位置1）

1001 = 比較模式，匹配時輸出低電平（PIRx寄存器的CCPxIF位置1）

1010 = 比較模式，匹配時僅產生軟體插斷（PIRx寄存器的CCPxIF位置1，CCPx引腳不受影響）

1011 = 比較模式，觸發特殊事件（PIRx寄存器的CCPxIF位置1，Timer1計數寄存器復位，CCPx引腳不受影響。）

11xx = PWM模式

### 9.5.1 捕捉模式

在輸入捕捉模式，適合用於測量引腳輸入週期性方波信號的週期、頻率和占空比等，也適合用於測量引腳輸入的非週期性矩形方波脈衝信號的寬度、到達時刻或消失時刻等參數。

當CCPx模組工作于捕捉模式時，一旦有下列事件在引腳CCPx上發生，CCPRx寄存器立即捕捉下這

一時刻的TMR1計數值：

- 每個下降沿
- 每個上升沿
- 每 4 個上升沿
- 每 16 個上升沿

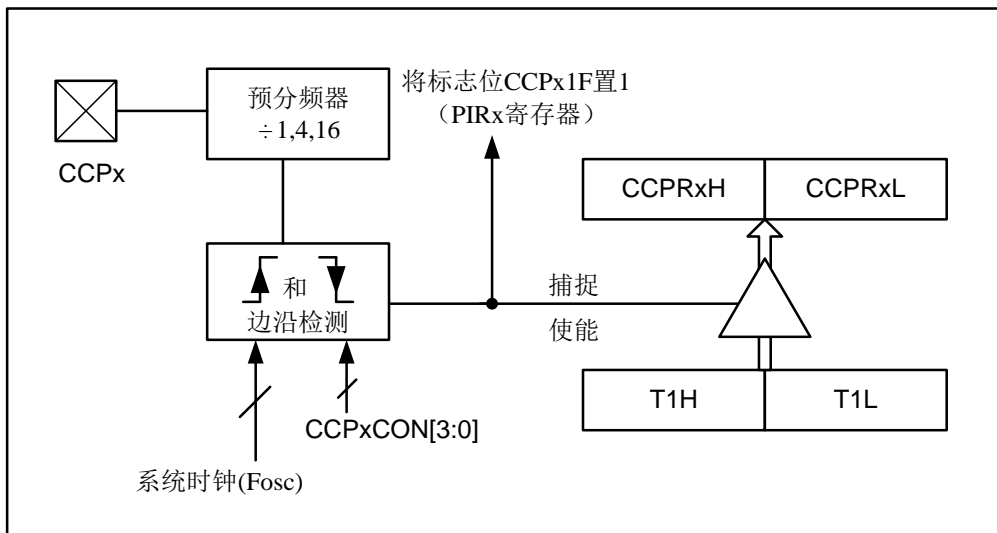
CCPxCON寄存器的CCPxM[3:0]設置的是預分頻器，關閉CCP模組或者CCP模組不在捕捉模式，預分頻計數器將會被清零。為避免錯誤中斷，可在改變預分頻比前通過清零 CCPxCON寄存器來關閉CCP模組。

在捕捉模式下，Timer1必須運行在計時器模式或同步計數器模式。

**使用注意：**

- 1、在捕捉模式下，CCPx引腳必須由相應的方向控制器設定為輸入方式；
- 2、CCPxCON寄存器的CCPxM[3:0]設置的是預分頻器，關閉CCP模組或者CCP模組不在捕捉模式，預分頻計數器將會被清零。為避免錯誤中斷，可在改變預分頻比前通過清零 CCPxCON寄存器來關閉CCP模組。如果需要中途改變預分頻器的分頻比，建議使用以下程式片段：
 

```
BCF STATUS,RP0 ;BANK0
CLRf CCPxCON ;關閉CCPx模組
MOVLW NEW_CAPT_PS ;選取新的分頻比（1:1、1:4、1:16）
MOVWF CCPxCON ;賦予CCPxCON寄存器，並打開CCPx模組
```
- 3、當一個捕捉事件發生後，硬體自動將CCPx的中斷標誌位元CCPxIF置1，表示產生了一次CCPx捕捉中斷。CCPxIF位元必須用軟體重新清零。當CCPRx寄存器中的值還未被程式讀取，而又發生了另一個新的捕捉事件時，原先的值將被新的值覆蓋掉；
- 4、在捕捉模式下，Timer1必須運行在計時器模式或同步計數器模式。



CCPx引腳上發生變化時，CCPRxH:CCPRxL捕捉Timer1計數寄存器的16位元值，PIRx寄存器的中斷標誌位元CCPxIF被置1。如果在CCPRxH和CCPRxL寄存器的值被讀出之前又發生另一次捕捉，那麼原來的捕捉值會被新捕捉值覆蓋。

### 9.5.2 比較模式

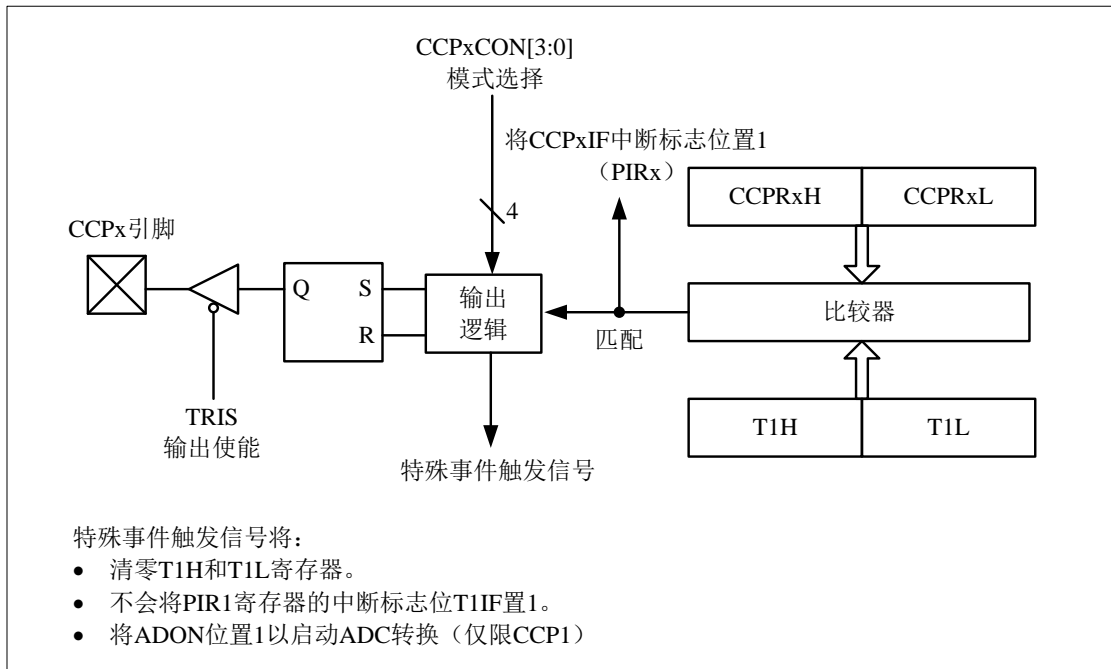
輸出比較模式，適用於從引腳上輸出不同寬度的矩形正脈衝、負脈衝、延時驅動信號、可控矽驅動信號、步進電機驅動信號等。

在比較模式下，CCPRxH:CCPRxL寄存器對成為了Timer1的週期寄存器，一旦Timer1計數寄存器對與CCPRxH和CCPRxL寄存器對發生匹配，Timer1計數寄存器對在Timer1時鐘的下一個上升沿重定，CCPx模組根據CCPxM[3:0] 控制位元的配置進行相應操作：

- CCPx 引腳輸出翻轉電平
  - CCPx 引腳輸出高電平
  - CCPx 引腳輸出低電平
  - 僅產生軟體插斷
  - 產生特殊事件觸發信號
- 所有比較模式都能產生CCP中斷。

**使用注意：**

- 1、當選擇產生特殊事件觸發信號時，如果ADC被使能，則啟動一次ADC轉換（僅限於CCP1）。在此模式下CCPx模組不會對CCPx引腳進行控制；
- 2、在比較模式下，CCPx引腳必須由相應寄存器設定為輸出模式，以便作為比較器的輸出端使用；
- 3、應該注意的是，如果對控制寄存器CCPxCON進行重新賦值，將會迫使CCPx引腳輸出一個默認的低電平，而這並非是正常的比較輸出結果；
- 4、在比較模式下，Timer1必須運行在計時器模式或同步計數器模式下。



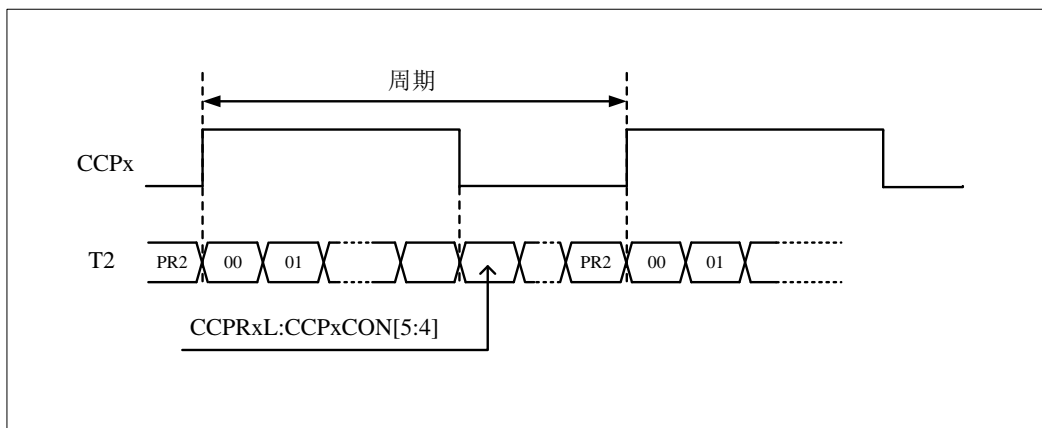
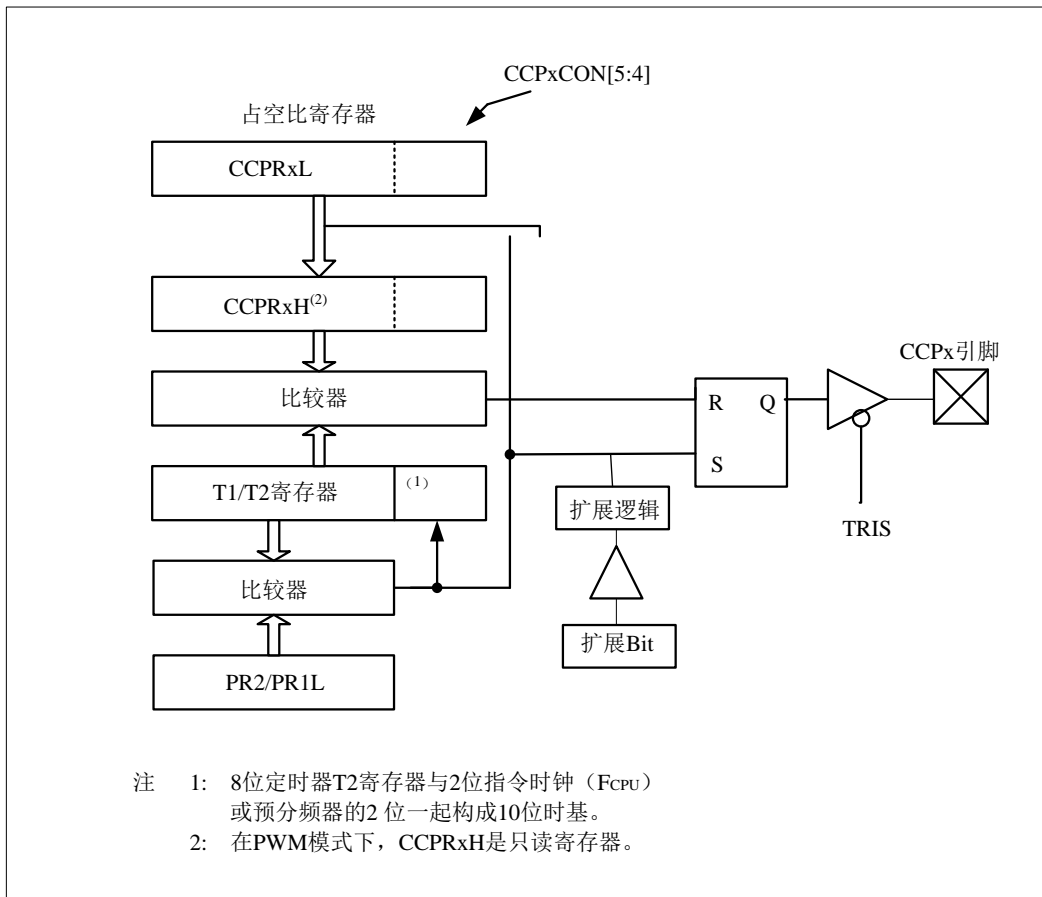
### 9.5.3 PWM 模式

脈寬調製，PWM(pulse width modulation)輸出工作模式，適用於從引腳上輸出脈衝寬度隨時可調的PWM信號。例如，實現直流電動機調速、簡易D/A轉換器、步進電機的變頻控制等。

9.5.3.1 PWM不選擇擴展

一、PWM時鐘源為Timer2

在PWM模式下，當Timer2計數寄存器中的值與PR2寄存器中的值發生匹配時，在下一個計數時鐘Timer2計數寄存器被清零，CCPx引腳被置1（如果PWM占空比為0%，CCPx引腳將不會被置1），PWM占空比值從CCPRxL鎖存到CCPRxH（在Timer2計數寄存器中的值與PR2寄存器中的值發生匹配前，占空比值不會被鎖存到CCPRxH中）。



✓ 以下公式中，當晶片配置為 4T 模式時，n=1;當晶片配置為 2T 模式時，n=1/2



PWM 週期：

$$\text{PWM 周期} = [(\text{PR2}) + 1] \cdot 4 \cdot n \cdot \text{Tsys}$$

(Timer2 預分頻值)

注：  $\text{Tsys} = 1/\text{Fsys}$

PWM脈衝寬度：

$$\text{脈衝寬度} = (\text{CCPRxL:CCPxCON}[5:4]) \cdot n \cdot \text{Tsys} \cdot (\text{Timer2 預分頻值})$$

注：  $\text{Tsys} = 1/\text{Fsys}$

如果脈衝寬度值比週期長，則指定的PWM引腳將保持不變。

PWM占空比：

$$\text{占空比} = \frac{(\text{CCPRxL:CCPxCON}\langle 5:4 \rangle)}{4(\text{PR2} + 1)}$$

當時鐘模式選擇配置字選擇為2T時，PWM占空比由CCPRxL寄存器和CCPxCON寄存器的DCxB[1]位決定。

PWM解析度：

$$\text{分辨率} = \frac{\text{Log}[4(\text{PR2} + 1)]}{\text{Log}(2)} \text{ 位}$$

解析度是PR2寄存器值的函數，當PR2為255時PWM最大解析度為10位元（時鐘模式選擇配置字選擇為2T時，PWM最大解析度為9位）。

#### 關於 PWM 模組的使用，需注意

1. 當  $\text{PWM:T2CON}[1:0] = 0$  時，PWM 輸出波形恒為低電平，占空比為 0%；當  $\text{PWM:T2CON}[1:0] > 4(\text{PR2}+1)$ ，PWM 輸出波形恒為高電平，占空比為 100%；
2. 當晶片配置為 2T 模式且 Timer2 的預分頻比為 1:1 時，因需保持 PWM0 (T2CON 的 BIT0 位) = 0，所以以上公式計算出來的解析度需減 1。

PWM使用：

1. 設置相關TRISB位為1，禁止CCPx引腳輸出
2. 設置PWM週期，輸入PR2寄存器值
3. 設置CCPxCON寄存器，將CCP模組配置為PWM模式
4. 設置PWM占空比，輸入CCPRxL寄存器和CCPxCON[5:4]寄存器值
5. 配置和啟動Timer2
  - 將PIR1寄存器的T2IF中斷標誌位元清零
  - 設置T2CON寄存器的T2CKPS位，選擇Timer2預分頻
  - 將T2CON寄存器的T2ON置1，使能Timer2

6. 設置PWM輸出

- 等待Timer2溢出，PIR1寄存器的T2IF位置1
- 將TRISB2或TRISB3(TRISA6或TRISA7)位清零，讓CCPx引腳輸出

7.如何關閉PWM輸出

- 將TRISB2或TRISB3(TRISA6或TRISA7)位置1，設引腳輸入
- 設置CCPxCON寄存器的CCPxCON[3:0]設為0000，關PWM功能，CCPx用作I/O口
- 根據需要,設PB2或PB3(PA6或PA7)輸出為高電平或者低電平

➤ 例：配置PWM輸出38K驅動方波，採用4M/4T模式。

```

ORG 0000H ;復位向量
GOTO MAIN
ORG 0004H ;中斷
.....
RETFIE

```

MAIN:

```

BCF STATUS,RP0 ;BANK0
BCF PCON,WDTENS ;DISABLE WDT
MOVLW 0x00
MOVWF PORCTR ;設置埠映射為PB2
CLRF OPTION ;使用高頻時鐘
BSF TRISB,2 ;CCPx口設為輸入

MOVLW D'25'
MOVWF PR2 ;設置PR2為26µs

MOVLW H'0D'
MOVWF CCPR1L ;占空比高8位
MOVLW B'00001100' ;選PWM模組，填寫占空比低兩位
MOVWF CCP1CON ;脈衝寬度13µs

BCF PIR1,T2IF
CLRF T2CON ;分頻比1：1
BSF T2CON,T2ON ;開T2
BSF PIE1,T2IE

BTFSS PIR1,T2IF ;等待T2溢出
GOTO $-1
BCF PIR1,T2IF ;清除中斷標誌
BCF PIE1,T2IE
BCF TRISB,2 ;配置完成，PB2將持續輸出38K

```

二、PWM2時鐘源為Timer1

|      |       |       |       |       |       |       |       |       |
|------|-------|-------|-------|-------|-------|-------|-------|-------|
| 05Fh | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------|-------|-------|-------|-------|-------|-------|-------|-------|

|      |                 |     |     |     |     |     |     |     |
|------|-----------------|-----|-----|-----|-----|-----|-----|-----|
| PR1L | Timer1週期寄存器低位元組 |     |     |     |     |     |     |     |
| R/W  | R/W             | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| POR值 | 0               | 0   | 0   | 0   | 0   | 0   | 0   | 0   |

|        |       |       |       |       |        |        |       |       |
|--------|-------|-------|-------|-------|--------|--------|-------|-------|
| 060h   | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3  | Bit 2  | Bit 1 | Bit 0 |
| PR1CON | PWM1T | PWM1T | PWM2T | PWM2T | T1CKPS | T1CKPS | PWMPR | PR1EN |
|        | 1     | 0     | 1     | 0     | 3      | 2      | 1     | N     |
| R/W    | R/W   | R/W   | R/W   | R/W   | R/W    | R/W    | R/W   | R/W   |
| POR值   | 0     | 0     | 0     | 0     | 0      | 0      | 0     | 0     |

Bit0 **PR1EN** : Timer1可設週期使能位

1= Timer1 為可設週期的8位Timer

0= Timer1 為16位Timer

Bit1 **PWMPR1** : Timer1提供PWM時鐘源使能位元

1=CCP2的PWM的時鐘源由8位元可設週期的Timer1提供（當PR1EN有效時，該位可用）

0=CCP2的PWM的時鐘源由Timer2提供

Bit[3-2] **T1CKPS[3-2]**:8位可設週期Timer1的分頻比控制位的高兩位

| T1CKPS[3:0] | Timer1 預分頻比 |
|-------------|-------------|
| 0000        | 1 : 1       |
| 0001        | 1 : 2       |
| 0010        | 1 : 4       |
| 0011        | 1 : 8       |
| 0100        | 1 : 16      |
| 0101        | 1 : 32      |
| 0110        | 1 : 64      |
| 0111        | 1 : 128     |
| 1xxx        | 1 : 256     |

#### 使用注意：

將寄存器PR1CON[bit1~bit0]置1，使PWM2的時鐘源選擇位元可設週期的8位Timer1，其他使用操作可參考由Timer2提供時鐘源的PWM2的使用。

#### 9.5.3.2 PWM選擇擴展

當PWM選擇擴展週期時，4 個調製週期為一個輸出週期，此時PWM最高可擴展到12位。也就是說當選擇擴展PWM模式時，PWM將為4個波形一個週期，此時不論你在哪一個波形輸出時改變PWM的資料記憶體都將在下一個週期才生效。

擴展週期控制寄存器PR1CON

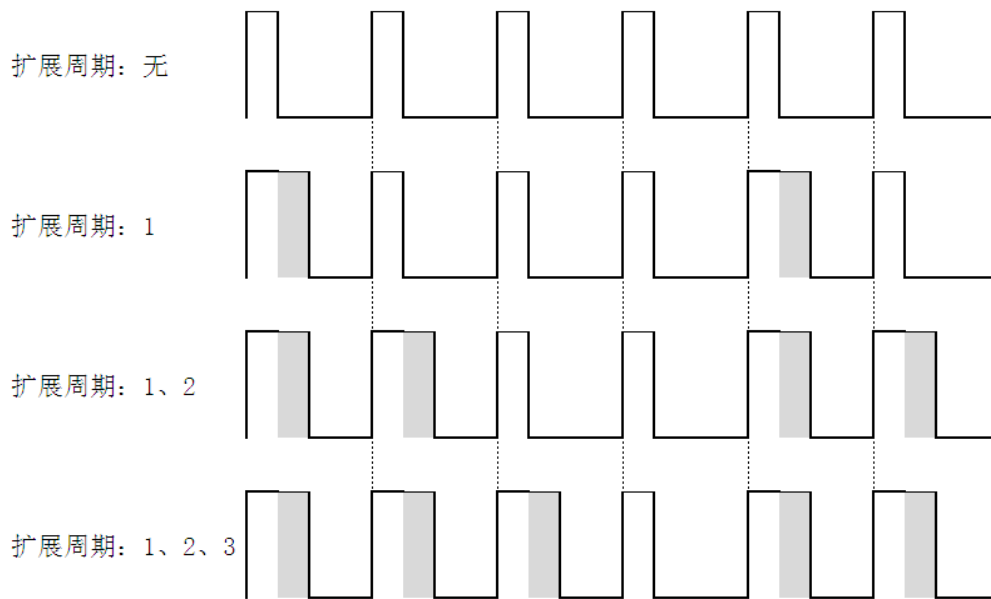
| 060h   | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3  | Bit 2  | Bit 1 | Bit 0 |
|--------|-------|-------|-------|-------|--------|--------|-------|-------|
| PR1CON | PWM1T | PWM1T | PWM2T | PWM2T | T1CKPS | T1CKPS | PWMPR | PR1EN |
| N      | 1     | 0     | 1     | 0     | 3      | 2      | 1     | N     |
| R/W    | R/W   | R/W   | R/W   | R/W   | R/W    | R/W    | R/W   | R/W   |
| POR值   | 0     | 0     | 0     | 0     | 0      | 0      | 0     | 0     |

Bit[5-4] PWM2T[1-0] : PWM2的擴展週期選擇位

- 00 : 無擴展週期
- 01 : 擴展週期為1
- 10 : 擴展週期為1、2
- 11 : 擴展週期為1、2、3

Bit[7-6] PWM1T[1-0] : PWM1的擴展週期選擇位

- 00 : 無擴展週期
- 01 : 擴展週期為1
- 10 : 擴展週期為1、2
- 11 : 擴展週期為1、2、3



# 10 PWM模組

## 10.1 概述

- 3 組帶死區互補 PWM 或 3 路獨立 PWM 輸出
- 提供每個 PWM 週期溢出中斷，但中斷共用同一向量入口
- 輸出極性可選擇
- 提供出錯偵測功能可緊急關閉 PWM 輸出
- PWM 工作時鐘源可設定時鐘分頻比
- PWM 可做計時器/計數器使用

HC18P23xL 集成了一個 12 位元 PWM 模組 PWM0 和兩個 8 位 PWM1/PWM2，三個模組各有一個計數器，PWM0 的計數器由 EPWM0 或 EPWM01 來控制，只要它們中的任何一個使能，都可以啟動計數器，計數器的時鐘源通過 PWM0C 控制寄存器裡的 CK0 來選擇。

如果 EPWM0 使能，而沒有通過功能引腳映射寄存器進行 PWM0 的映射，這樣也不會從晶片管腳上輸出 PWM0，這時候 PWM0 的計數器可以當一個計時器來使用，當計數器溢出時，如果中斷允許也會產生 PWM 中斷。

如果 EFLT 置 1，PWM0/PWM1/PWM2 輸出和其互補輸出可由 FLT 引腳輸入信號變化自動關閉。一旦檢測到 FLT 引腳輸入有效電平，PWM 輸出會立即關閉，但 PWM 內部計數器仍在繼續運行，這樣方便在 FLT 引腳錯誤去除後繼續 PWM 輸出。在 FLT 輸入信號有效期間，FLTS 位元無法清除。只有當 FLT 輸入信號消失後，才能軟體清除 FLTS 狀態位元，此時 PWM 恢復正常輸出。

PWM 計時器也為 PWM0/1/2 提供 3 個中斷源，在每個 PWM 週期都會產生中斷。它們有不同的控制位元和標誌位元。這樣用戶可以實現每個 PWM 週期中更改下一次迴圈的週期或占空比。

## 10.2 PWM 相關寄存器

| 25Eh  | Bit 7 | Bit 6 | Bit 5  | Bit 4  | Bit 3  | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|--------|--------|--------|-------|-------|-------|
| PWMEN | -     | EFLT  | EPWM21 | EPWM11 | EPWM01 | EPWM2 | EPWM1 | EPWM0 |
| R/W   | -     | R/W   | R/W    | R/W    | R/W    | R/W   | R/W   | R/W   |
| POR的值 | -     | 0     | 0      | 0      | 0      | 0     | 0     | 0     |

bit 6 **EFLT**：FLT引腳配置位元

1 = PWM故障檢測輸入引腳

0 = 普通I/O口或其他功能

bit 5 **EPWM21**：PWM21輸出控制位

1 = PWM21輸出允許

0 = PWM21 輸出禁止，用作 I/O 功能

bit 4 **EPWM11**：PWM11輸出控制位

1 = PWM11輸出允許

0 = PWM11輸出禁止，用作I/O功能

bit 3 **EPWM01**：PWM01輸出控制位

1 = PWM01輸出允許

0 = PWM01 輸出禁止，用作 I/O 功能

bit 2 **EPWM2**：PWM2輸出控制位

- 1 = PWM2輸出允許
- 0 = PWM2輸出禁止，用作I/O功能

bit 1 **EPWM1**：PWM1輸出控制位

- 1 = PWM1輸出允許
- 0 = PWM1輸出禁止，用作I/O功能

bit 0 **EPWM0**：PWM0輸出控制位

- 1 = PWM0輸出允許
- 0 = PWM0輸出禁止，用作I/O功能

當PWMMEN清零後，PWM輸出立即關閉。

**FLT** 埠主要用於檢測異常信號，快速關閉 PWM 輸出，**FLT** 探測到故障後，由硬體執行使 PWM 輸出關閉，所以當故障發生後，它可以快速響應，使得 PWM 輸出無效以保護連接 PWM 的大功率器件。當使能 **FLT** 引腳故障檢測功能後，埠禁止任何上下拉電阻功能。

如果 **EFLT** 位清零，則表示 **FLT** 埠對 PWM 計時器輸出控制無效。

| 25Fh  | Bit 7 | Bit 6 | Bit 5  | Bit 4  | Bit 3  | Bit 2  | Bit 1  | Bit 0  |
|-------|-------|-------|--------|--------|--------|--------|--------|--------|
| FLTM  | -     | -     | FLT2M1 | FLT2M1 | FLT1M1 | FLT1M1 | FLT0M1 | FLT0M1 |
| R/W   | -     | -     | R/W    | R/W    | R/W    | R/W    | R/W    | R/W    |
| POR的值 | -     | -     | 0      | 0      | 0      | 0      | 0      | 0      |

輸出故障時，PWM 口輸出狀態：

**FLT<sub>x</sub>M[1:0]**：PWM<sub>x</sub> 口 **FLT** 故障時後，埠輸出狀態

- 00 = PWM<sub>x</sub> 輸出高電平
- 01 = PWM<sub>x0</sub> 輸出高電平，PWM<sub>x1</sub> 輸出低電平
- 10 = PWM<sub>x0</sub> 輸出低電平，PWM<sub>x1</sub> 輸出高電平
- 11 = PWM<sub>x</sub> 輸出低電平

## 10.3 PWM0 模組

### 10.3.1 PWM0 控制寄存器

HC18P23xL 包含一個 12 位 PWM 模組。PWM 模組可以產生週期和占空比分別可調的脈寬調製波形。PWMC 寄存器用於控制 PWM 模組的時鐘，PWMPH/L 寄存器用於控制 PWM 輸出波形的週期。PWMDH/L 寄存器用於控制 PWM 模組輸出波形的占空比。

在 PWM 輸出允許器件可以修改這三個寄存器，但在下一個 PWM 週期修改才會起作用。

| 25Dh  | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3  | Bit 2  | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|--------|--------|-------|-------|
| PWM0C | -     | -     | FLTS  | FLTC  | PWM0S1 | PWM0S0 | CK01  | CK00  |
| R/W   | -     | -     | R/W   | R/W   | R/W    | R/W    | R/W   | R/W   |
| POR的值 | -     | -     | 0     | 0     | 0      | 0      | 0     | 0     |

bit 5 **FLTS**：FLT狀態位元

- 1 = PWM輸出關閉，硬體置1
- 0 = PWM 正常狀態，軟體清零

bit 4 **FLTC**：FLT引腳配置位元

1 = FLT為高電平時，PWM輸出關閉

0 = FLT為低電平時，PWM輸出關閉

bit [3:2] **PWM0S[1:0]**：PWM0和PWM01 占空比輸出模式選擇位元

11 = PWM0和PWM01均為低有效

10 = PWM0為低有效，PWM01為高有效

01 = PWM0為高有效，PWM01為低有效

00 = PWM0和PWM01均為高有效

Bit [1:0] **CK0[1:0]**：PWM1輸出控制位

11 = Fosc/16

10 = Fosc/8

01 = Fosc/4

00 = Fosc/2

注：Fosc = 32MHz

注意：

1、PWM0C寄存器中的FLTS和FLTC位控制PWM0/1/2計時器，而寄存器中的PWM0S,CK0[1:0]只能影響12位PWM計時器。

2、PWM輸出關閉時，PWM0/1/2和PWM01/11/21輸出固定電平：

PWMxS[1:0] = 00，PWMx和PWMx1均輸出固定低電平；

PWMxS[1:0] = 01，PWMx輸出固定低電平，PWMx1輸出固定高電平；

PWMxS[1:0] = 10，PWMx輸出固定高電平，PWMx1輸出固定低電平；

PWMxS[1:0] = 11，PWMx和PWMx1均輸出固定高電平；

其中x=0/1

3、一旦檢測到FLT引腳輸入有效電平，PWM輸出會立即關閉，但PWM內部計數器仍在繼續運行。這樣方便在FLT引腳錯誤去除後繼續PWM輸出。

4、在FLT輸入信號有效期間，FLTS位無法清除。只有當FLT輸入信號消失後，才能軟體清除FLTS狀態位元，此時PWM恢復正常輸出。

### 10.3.2 PWM0 週期寄存器

12 位 PWM 週期控制寄存器的高 4 位

| 25Ch   | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3  | Bit 2  | Bit 1 | Bit 0 |
|--------|-------|-------|-------|-------|--------|--------|-------|-------|
| PWM0PH | -     | -     | -     | -     | PP0.11 | PP0.10 | PP0.9 | PP0.8 |
| R/W    | -     | -     | -     | -     | R/W    | R/W    | R/W   | R/W   |
| POR的值  | -     | -     | -     | -     | 0      | 0      | 0     | 0     |

12 位 PWM 週期控制寄存器的低 8 位

| 25Bh   | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|-------|-------|-------|-------|-------|-------|-------|-------|
| PWM0PL | PP0.7 | PP0.6 | PP0.5 | PP0.4 | PP0.3 | PP0.2 | PP0.1 | PP0.0 |
| R/W    | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR的值  | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

PWM0輸出週期=[ PWM0PH:PWM0PL] ×PWM時鐘源

當[PWM0PH:PWM0PL]=0x00時：

如果PWM0S[1:0] =00，不管PWM占空比為多少，PWM0輸出低電平，PWM01輸出低電平。

如果PWM0S[1:0] =01，不管PWM占空比為多少，PWM0輸出低電平，PWM01輸出高電平。

如果PWM0S[1:0] =10，不管PWM占空比為多少，PWM0輸出高電平，PWM01輸出低電平。

如果PWM0S[1:0] =11，不管PWM占空比為多少，PWM0輸出高電平，PWM01輸出高電平。

### 10.3.3 PWM0 占空比寄存器

12 位 PWM 脈寬控制寄存器的高 4 位

| 25Ah   | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3  | Bit 2  | Bit 1 | Bit 0 |
|--------|-------|-------|-------|-------|--------|--------|-------|-------|
| PWM0DH | -     | -     | -     | -     | PD0.11 | PD0.10 | PD0.9 | PD0.8 |
| R/W    | -     | -     | -     | -     | R/W    | R/W    | R/W   | R/W   |
| POR的值  | -     | -     | -     | -     | 0      | 0      | 0     | 0     |

12 位 PWM 脈寬控制寄存器的低 8 位

| 259h   | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|-------|-------|-------|-------|-------|-------|-------|-------|
| PWM0DL | PD0.7 | PD0.6 | PD0.5 | PD0.4 | PD0.3 | PD0.2 | PD0.1 | PD0.0 |
| R/W    | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR的值  | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

PWM輸出占空比= [PWM0DH:PWM0DL] X PWM時鐘源

當[PWM0PH:PWM0PL]≤[PWM0DLH:PWM0DL]，

如果PWM0S[1:0] =00，PWM0輸出高電平，PWM01輸出低電平

如果PWM0S[1:0] =01，PWM0輸出低電平，PWM01輸出低電平

如果PWM0S[1:0] =10，PWM0輸出高電平，PWM01輸出高電平

如果PWM0S[1:0] =11，PWM0輸出低電平，PWM01輸出高電平

使用注意事項：

- 1、選擇PWM模組時鐘源。
- 2、設置PWM週期/占空比，先設置低位元，再設置高位。注意，即使高位數值不變，也要重寫一次，否則，低位的修改無效。
- 3、通過設置PWM控制寄存器(PWM0C)的PWM0Sx位選擇PWM輸出模式（高電平/低電平有效）。
- 4、通過設置PWM控制寄存器(PWMEN)的EPWM0或EPWM01來允許PWM上橋或下橋輸出。
- 5、如果PWM週期或者占空比需要改變，操作流程如同步驟2和步驟3說明。修改後的重載計數器的值在下一個週期開始有效。
- 6、注意不要對PWM關鍵寄存器進行誤操作。



## 10.4 PWM1 模組

### 10.4.1 PWM1 控制寄存器

| 257h  | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3  | Bit 2  | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|--------|--------|-------|-------|
| PWM1C | -     | -     | -     | -     | PWM1S1 | PWM1S0 | CK11  | CK10  |
| R/W   | -     | -     | -     | -     | R/W    | R/W    | R/W   | R/W   |
| POR的值 | -     | -     | -     | -     | 0      | 0      | 0     | 0     |

bit [3:2] **PWM1S[1:0]** : PWM1和PWM11占空比輸出模式選擇位元

- 11 = PWM1和PWM11均為低有效
- 10 = PWM1為低有效，PWM11為高有效
- 01 = PWM1為高有效，PWM11為低有效
- 00 = PWM1和PWM11均為高有效

Bit [1:0] **CK1[1:0]** : PWM1輸出控制位

- 11 = Fosc/16
  - 10 = Fosc/8
  - 01 = Fosc/4
  - 00 = Fosc/2
- 注：Fosc = 32MHz

### 10.4.2 PWM1 週期寄存器

週期控制寄存器

| 256h  | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| PWM1P | PP1.7 | PP1.6 | PP1.5 | PP1.4 | PP1.3 | PP1.2 | PP1.1 | PP1.0 |
| R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR的值 | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

PWM1輸出週期=[PWM1P] X PWM時鐘源

當[PWM1P]=0x00時：

- 如果PWM1S[1:0] =00，不管PWM占空比為多少，PWM1輸出低電平，PWM11輸出低電平。
- 如果PWM1S[1:0] =01，不管PWM占空比為多少，PWM1輸出低電平，PWM11輸出高電平。
- 如果PWM1S[1:0] =10，不管PWM占空比為多少，PWM1輸出高電平，PWM11輸出低電平。
- 如果PWM1S[1:0] =11，不管PWM占空比為多少，PWM1輸出高電平，PWM11輸出高電平。

### 10.4.3 PWM1 占空比寄存器

脈寬控制寄存器

| 255h  | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| PWM1D | PD1.7 | PD1.6 | PD1.5 | PD1.4 | PD1.3 | PD1.2 | PD1.1 | PD1.0 |
| R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR的值 | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

PWM輸出占空比= [PWM0D] X PWM時鐘源

當[PWM1P]≤[PWM1D]，

- 如果PWM1S[1:0] =00，PWM1輸出高電平，PWM11輸出低電平

如果PWM1S[1:0] = 01，PWM1輸出低電平，PWM11輸出低電平  
 如果PWM1S[1:0] = 10，PWM1輸出高電平，PWM11輸出高電平  
 如果PWM1S[1:0] = 11，PWM1輸出低電平，PWM11輸出高電平

使用注意事項：

- 1、選擇PWM模組時鐘源。
- 2、設置PWM週期/占空比，先設置低位元，再設置高位。注意，即使高位數值不變，也要重寫一次，否則，低位的修改無效。
- 3、通過設置PWM控制寄存器(PWM1C)的PWM1Sx位選擇PWM輸出模式（高電平/低電平有效）。
- 4、通過設置PWM控制寄存器(PWMEN)的EPWM1或EPWM01來允許PWM上橋或下橋輸出。
- 5、如果PWM週期或者占空比需要改變，操作流程如同步驟2和步驟3說明。修改後的重載計數器的值在下一個週期開始有效。
- 6、注意不要對PWM關鍵寄存器進行誤操作。

## 10.5 PWM2 模組

### 10.5.1 PWM2 控制寄存器

| 253h  | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3  | Bit 2  | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|--------|--------|-------|-------|
| PWM2C | -     | -     | -     | -     | PWM2S1 | PWM2S0 | CK21  | CK20  |
| R/W   | -     | -     | -     | -     | R/W    | R/W    | R/W   | R/W   |
| POR的值 | -     | -     | -     | -     | 0      | 0      | 0     | 0     |

bit [3:2] **PWM2S[1:0]**：PWM2和PWM21占空比輸出模式選擇位元

11 = PWM1和PWM11均為低有效  
 10 = PWM2為低有效，PWM21為高有效  
 01 = PWM2為高有效，PWM21為低有效  
 00 = PWM2和PWM21均為高有效

bit [1:0] **CK2[1:0]**：PWM2輸出控制位

11 = Fosc/16  
 10 = Fosc/8  
 01 = Fosc/4  
 00 = Fosc/2

注：Fosc = RC32MHz

### 10.5.2 PWM2 週期寄存器

週期控制寄存器

| 252h  | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| PWM2P | PP2.7 | PP2.6 | PP2.5 | PP2.4 | PP2.3 | PP2.2 | PP2.1 | PP2.0 |
| R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR的值 | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

PWM2輸出週期=[PWM2P] × PWM時鐘源

當[PWM2P]=0x00時：

- 如果PWM2S[1:0] =00，不管PWM占空比為多少，PWM2輸出低電平，PWM21輸出低電平。
- 如果PWM2S[1:0] =01，不管PWM占空比為多少，PWM2輸出低電平，PWM21輸出高電平。
- 如果PWM2S[1:0] =10，不管PWM占空比為多少，PWM2輸出高電平，PWM21輸出低電平。
- 如果PWM2S[1:0] =11，不管PWM占空比為多少，PWM2輸出高電平，PWM21輸出高電平。

### 10.5.3 PWM2 占空比寄存器

脈寬控制寄存器

| 251h  | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| PWM2D | PD2.7 | PD2.6 | PD2.5 | PD2.4 | PD2.3 | PD2.2 | PD2.1 | PD2.0 |
| R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR的值 | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

PWM2輸出占空比= [PWM0D] × PWM時鐘源

當[PWM2P]≤[PWM2D]，

- 如果PWM2S[1:0] =00，PWM2輸出高電平，PWM21輸出低電平
- 如果PWM2S[1:0] =01，PWM2輸出低電平，PWM21輸出低電平
- 如果PWM2S[1:0] =10，PWM2輸出高電平，PWM21輸出高電平
- 如果PWM2S[1:0] =11，PWM2輸出低電平，PWM21輸出高電平

使用注意事項：

- 1、選擇PWM模組時鐘源。
- 2、設置PWM週期/占空比，先設置低位元，再設置高位。注意，即使高位數值不變，也要重寫一次，否則，低位的修改無效。
- 3、通過設置PWM控制寄存器(PWM2C)的PWM2Sx位選擇PWM輸出模式（高電平/低電平有效）。
- 4、通過設置PWM控制寄存器(PWMEN)的EPWM2或EPWM21來允許PWM上橋或下橋輸出。
- 5、如果PWM週期或者占空比需要改變，操作流程如同步驟2和步驟3說明。修改後的重載計數器的值在下一個週期開始有效。
- 6、注意不要對PWM關鍵寄存器進行誤操作。

### 10.6 死區時間

一般的，當沒有插入死區時間時，PWM01/11/21 輸出波形與 PWM0/1/2 輸出波形成固定相位關係。當 PWM 控制寄存器中 EPWM01/11/21 位置 1 時，PWM01/11/21 的輸出波形硬體自動產生。

注意：

- 1、當 PWM0/1/2 被禁止，如果 PWM01/11/21 被允許，則 PWM01/11/21 仍然會有信號輸出。
- 2、如果 EFLT 置位，當 FLT 埠有效時，PWM01/11/21 和 PWM0/1/2 都輸出固定電平：  
 PWMxS[1:0] =00，PWMx和PWMx1均輸出固定低電平；  
 PWMxS[1:0] =01，PWMx輸出固定低電平，PWMx1輸出固定高電平；

PWMxS[1:0]=10，PWMx輸出固定高電平，PWMx1輸出固定低電平；

PWMxS[1:0]=11，PWMx和PWMx1均輸出固定高電平；

其中x=0/1

HC18P23xL PWM 提供死區時間控制功能。通過寫 PWM0/1/2 死區時間控制寄存器，在 PWM0/1/2 和 PWM0/1/2 之間產生死區時間。PWM0/1/2 和 PWM0/1/2 的週期相同。

注意：

- 1、修改死區時間前，應禁止 PWM 輸出。
- 2、在 PWMn 和 PWMn1 為互補波形輸出時，為了產生死區時間，請確保（PWMn 占空比寄存器值 x 分頻係數（2,4,8,16）>PWMn1 的死區時間（n=0, 1,2）控制。否則，PWM0/1/2/1 當 PWMnS[1:0]=10 時輸出高電平，PWMnS[1:0]=01 時輸出低電平。
- 3、PWMDT 寄存器用於控制死區時間，它的時基為振盪器時鐘，而週期和占空比的時基由 TnCKn[1:0]（n=0,1,2）控制，最小為 2 個振盪器時鐘。

#### PWM0 死區時間控制寄存器

| 258h   | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|-------|-------|-------|-------|-------|-------|-------|-------|
| PWM0DT | DT07  | DT06  | DT05  | DT04  | DT03  | DT02  | DT01  | DT00  |
| R/W    | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR的值  | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

#### PWM1 死區時間控制寄存器

| 254h   | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|-------|-------|-------|-------|-------|-------|-------|-------|
| PWM1DT | DT17  | DT16  | DT15  | DT14  | DT13  | DT12  | DT11  | DT10  |
| R/W    | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR的值  | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

#### PWM2 死區時間控制寄存器

| 250h   | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|-------|-------|-------|-------|-------|-------|-------|-------|
| PWM2DT | DT27  | DT26  | DT25  | DT24  | DT23  | DT22  | DT21  | DT20  |
| R/W    | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR的值  | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

PWMn（n=0,1,2）的死區時間為 PWMnDT× T<sub>osc</sub>

注：T<sub>osc</sub> = 1/F<sub>osc</sub> = 1/32MHz

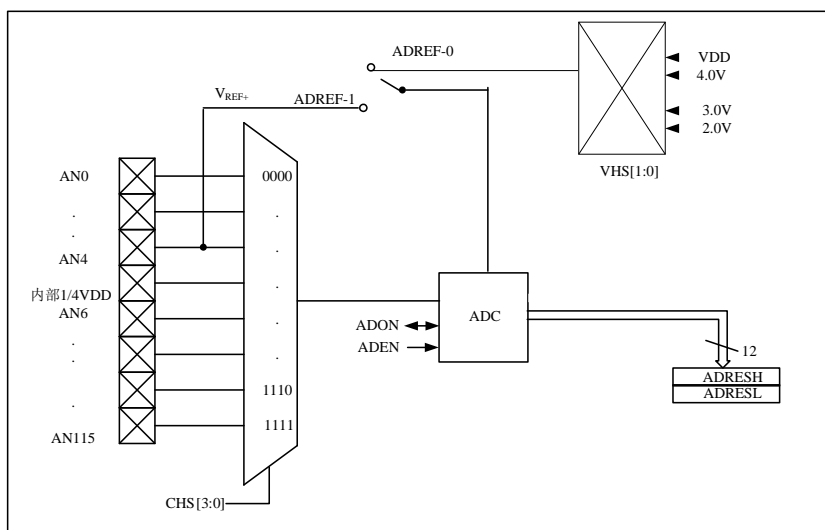
使用說明：當使用 PWM 模組時，系統時鐘必須選擇為內部 RC 模式。

# 11 模數轉換 (ADC)

## 11.1 ADC 概述

HC18P23xL具有一個12位轉換解析度的模數轉換器，共有15個外部類比輸入通道，1個內部電池檢測通道。

ADC的等效電路：



## 11.2 A/D 寄存器

| 08Ch   | Bit 7  | Bit 6  | Bit 5 | Bit 4  | Bit 3  | Bit 2  | Bit 1  | Bit 0  |
|--------|--------|--------|-------|--------|--------|--------|--------|--------|
| ANSELL | ANSEL7 | ANSEL6 | -     | ANSEL4 | ANSEL3 | ANSEL2 | ANSEL1 | ANSEL0 |
| R/W    | R/W    | R/W    | -     | R/W    | R/W    | R/W    | R/W    | R/W    |
| POR值   | 1      | 1      | -     | 1      | 1      | 1      | 1      | 1      |

| 08Dh   | Bit 7   | Bit 6   | Bit 5   | Bit 4   | Bit 3   | Bit 2   | Bit 1  | Bit 0  |
|--------|---------|---------|---------|---------|---------|---------|--------|--------|
| ANSELH | ANSEL15 | ANSEL14 | ANSEL13 | ANSEL12 | ANSEL11 | ANSEL10 | ANSEL9 | ANSEL8 |
| R/W    | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W    | R/W    |
| POR值   | 1       | 1       | 1       | 1       | 1       | 1       | 1      | 1      |

ANSEL[15:0]：A/D引腳數模控制位

1：類比模式，作為類比信號口，僅可作為AD通道的模擬輸入。

0：數位模式，作為數位輸入或輸出口。

注：1、該寄存器上電初始值為 B'1111 1111'，即作為模擬輸入。無論是否應用到 AD，均需要在上電後對 IO 操作之前按需配置，否則 IO 口可能無法受控於對應的埠寄存器，狀態將不確定。

2、ANSEL[4:0]對應 AN4~AN0 (PA4~PA0)，ANSEL[7:6]對應 AN7、AN6 (PA7、PA6)；

ANSEL[8]對應 AN8 (即 PA5)；

ANSEL[13:9]對應 AN13~AN11 (PB4~PB2)，ANSEL[15:14]對應 AN15、AN14 (PB7、PB6)。

## 11.3 A/D 控制寄存器

| 094h   | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|-------|-------|-------|-------|-------|-------|-------|-------|
| ADCON0 | VHS1  | VHS0  | CHS3  | CHS2  | CHS1  | CHS0  | ADON  | ADEN  |
| R/W    | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR的值  | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

| 095h   | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|-------|-------|-------|-------|-------|-------|-------|-------|
| ADCON1 | ADFM  | ADCS2 | ADCS1 | ADCS0 | -     | -     | -     | ADREF |
| R/W    | R/W   | W     | W     | W     | -     | -     | -     | R/W   |
| POR的值  | 0     | 0     | 0     | 0     | -     | -     | -     | 0     |

| 096h  | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2  | Bit 1  | Bit 0  |
|-------|-------|-------|-------|-------|-------|--------|--------|--------|
| ADCLK | -     | -     | -     | -     | -     | ADCLK2 | ADCLK1 | ADCLK0 |
| R/W   | -     | -     | -     | -     | -     | R/W    | R/W    | R/W    |
| POR的值 | -     | -     | -     | -     | -     | 0      | 0      | 0      |

ADC類比頻道選擇

| CHS [3:0] | 類比頻道           |
|-----------|----------------|
| 0000      | AN0            |
| 0001      | AN1            |
| 0010      | AN2            |
| 0011      | AN3            |
| 0100      | AN4            |
| 0101      | 內部 1/4VDD(AN5) |
| 0110      | AN6            |
| 0111      | AN7            |
| 1000      | AN8            |
| 1001      | AN9            |
| 1010      | AN10           |
| 1011      | AN11           |
| 1100      | AN12           |
| 1101      | AN13           |
| 1110      | AN14           |
| 1111      | AN15           |

ADC參考電壓選擇

| ADREF | VHS[1:0] | 參考電壓   |
|-------|----------|--------|
| 0     | 00       | 內部2.0V |
| 0     | 01       | 內部3.0V |
| 0     | 10       | 內部4.0V |
| 0     | 11       | 內部VDD  |
| 1     | xx       | 外部參考電壓 |

ADC資料存放格式選擇

| ADFM | 資料格式                    |
|------|-------------------------|
| 0    | ADRESH[7:0]:ADRESL[7:4] |
| 1    | ADRESH[1:0]:ADRESL[7:0] |

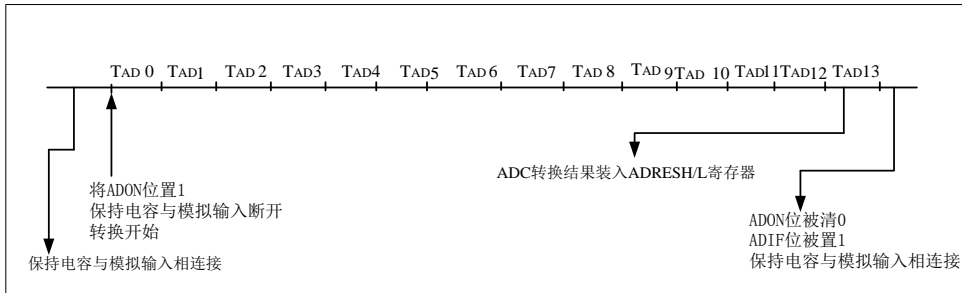
**注意：**

- 1、AN5為內部1/4VDD輸入通道，外部沒有輸入引腳。可作為電池系統的電池檢測器；
- 2、ADC所採集資料，當選擇存格式為左對齊時，ADC精度只能為12位，高8位存放在ADRESH寄存器中，低4位存放在ADRESL寄存器高4位上。當選擇存放格式為右對齊時，ADC精度只能為10位，高2位存放在ADRESH的低2位上，低8位存放在ADRESL上。

## 11.4 AD 轉換時間

ADC轉換一位元資料所需的時間定義為TAD，轉換一次完整的12位元資料需要14個TAD。為確保ADC正確轉換，必須滿足適當的TAD時間。

模數轉換TAD 週期



ADC轉換時間(TAD)與工作頻率關係表

| ADC 轉換時間 (TAD) |           | 系統頻率 (Fsys) :4MHz |
|----------------|-----------|-------------------|
| ADC 時鐘源        | ADCS[2:0] | 典型值               |
| Fsys           | 000       | 4us               |
| Fsys /2        | 001       | 8us               |
| Fsys /4        | 010       | 16us              |
| Fsys /8        | 011       | 32us              |
| Fsys /16       | 100       | 64us              |
| Fsys /32       | 101       | 128us             |
| Fsys /64       | 110       | 256us             |
| FRC            | 111       | 視 RC 的值而定         |

## ADCLK軟體配置表

ADCLK[2:0]：AD時鐘選擇位元（在確定系統頻率後，選擇不同的AD時鐘分頻對應的頻率。）

000:AD轉換頻率在4Mhz及以上

001：AD轉換頻率為1M

010：AD轉換頻率為2M

011/1XX：AD轉換頻率為1M以下

系統頻率/AD時鐘源選擇對應的ADC時鐘選擇為對應關係如下表所示：

| Fsys (系統頻率) | ADCS[2:0]<br>AD 分頻 | ADCLK<br>AD 時鐘 |
|-------------|--------------------|----------------|
| 32M         | 011                | 000            |
|             | 100                | 010            |
|             | 101                | 001            |
|             | 11x                | 011/1 xx       |
| 16M         | 010                | 000            |
|             | 011                | 010            |
|             | 100                | 001            |
|             | 101/11x            | 011/1 xx       |
| 8M          | 010                | 010            |
|             | 011                | 001            |
|             | 1xx                | 011/1 xx       |
| 4M          | 001                | 010            |
|             | 010                | 001            |
|             | 011/1xx            | 011/1xx        |
| 2M          | 000                | 010            |
|             | 001                | 001            |
|             | 01x/1xx            | 011/1 xx       |
| 1M          | 000                | 001            |
|             | 001/0xx/1xx        | 011/1 xx       |
| 500K        | xxx                | 011/1 xx       |

## 軟體配置ADC轉換時鐘使用注意說明：

- 1、 Fsys為系統時鐘，Fosc為RC時鐘，Fcpu為指令時鐘。如用戶在OPTION中選擇4M/2T，對應Fosc=8M，Fsys=4M，Fcpu=2M。
- 2、 為保證ADC轉換精度，在不同ADC轉換時鐘下需配置不同的ADC轉換時鐘個數：

| ADC轉換時鐘 | ADCKCS(ADCKCS[2:0]) | 說明                                   |
|---------|---------------------|--------------------------------------|
| 2 Mhz   | 010                 | 當ADC轉換時鐘頻率為2Mhz時，需將ADCKCS[2:0]配置成010 |
| 1 Mhz   | 001                 | 當ADC轉換時鐘頻率為1Mhz時，需將ADCKCS[2:0]配置成001 |
| < 1 Mhz | 011/1xx             | 當ADC轉換時鐘頻率小於1 Mhz時，需將                |



|  |  |                                             |
|--|--|---------------------------------------------|
|  |  | ADCKCS[2:0]配置成011/1xx (ADCKCS[2:0]上電默認為111) |
|--|--|---------------------------------------------|

- 3、為了加快ADC轉換速度，建議選用較快的時鐘源（ADC轉換時鐘不能超過4MHz）。

## 11.5 ADC 使用

1. 配置埠：
    - 設置 TRISA 寄存器禁止引腳輸出
    - 設置 ANSEL 寄存器配置引腳為類比輸入
  2. 配置ADC模組：
    - 選擇 ADC 轉換時鐘，設置 ADCS[2:0]、ADCKCS[2:0]
    - 選擇 ADC 參考電壓，設置 ADREF、ADCON0
    - 選擇 ADC 輸入通道，設置 CHS[3:0]
    - 使能 ADC 模組，設置 ADEN
  3. 配置ADC中斷（可選）：
    - 清零 ADC 中斷標誌
    - 使能 ADC 中斷
    - 使能外設中斷
    - 使能全域中斷
  4. 等待所需採集時間
  5. 設置ADON為1 啟動一次ADC轉換
  6. 通過以下方式之一等待ADC轉換完成：
    - 查詢 ADON 位
    - 等待 ADC 中斷（已使能中斷）
  7. 讀取ADC結果
  8. 清零ADC中斷標誌（如果已使能中斷則需要）
- 例：配置AD，結果保留在Bank0的NTCADHIGH、NTCADLOW中。

```

... ;其他程式
AD_TEST:
 BCF STATUS,RP0 ;Bank0
 BSF TRISA,0 ;設置AD口為輸入
 BCF STATUS,RP0 ;Bank0
 MOVLW B'01010000' ;INNER REF Fsys/32 ADRESH[7:0]
 ADRESL[7,6]
 MOVWF ADCON1
 MOVLW B'00000011'
 MOVWF ADCLK
 ;配置AD通道
 BCF STATUS,RP0 ;Bank0
 MOVLW B'00000001'
 MOVWF ANSELL ;PA0作為模擬輸入

```

```

 CLRFB ANSELH
 BCF STATUS,RP0 ;Bank0
 BCF ADCON0,CHS2
 BCF ADCON0,CHS1
 BCF ADCON0,CHS0
 BCF ADCON0,VHS1
 BCF ADCON0,VHS0 ;參考電壓為內部VDD
 NOP
 NOP
 BSF ADCON0,ADEN ;使能ADC
 CALL DELAY_1 ;延時，用戶可自行完成
 BSF ADCON0,ADON ;開始一次轉換
 AD_TEST_WAIT :
 BTFSC ADCON0,ADON ;等待轉換完成
 GOTO AD_TEST_WAIT
 ;轉換完成，保存結果
 MOVF ADRESH,W ;LOAD THE AD HIGH 8 BITS TO W
 MOVWF NTCADHIGH ;客戶應用時注意BANK
 MOVF ADRESL,W ;LOAD THE AD LOW 8 BITS TO W
 MOVWF NTCADLOW

```

**注意：**

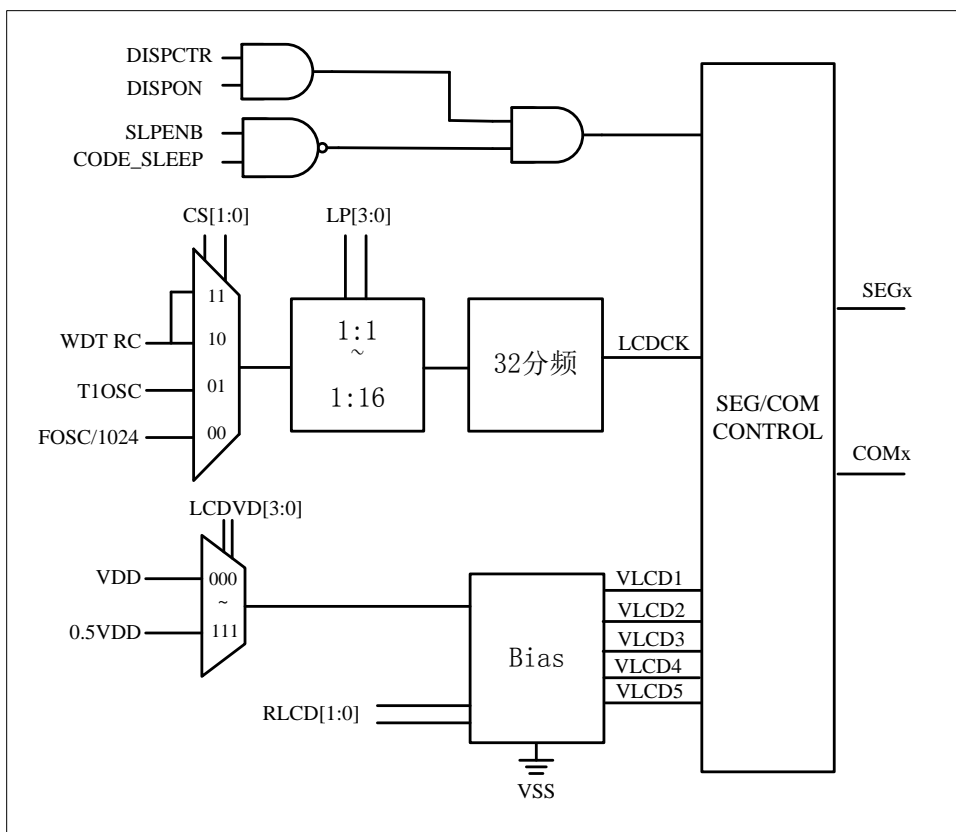
- 1、使能ADEN後（不是使能ADON），系統必須延遲一定的時間（視外部輸入信號而定）等待ADC電路穩定；
- 2、睡眠或綠色模式下，禁止ADC並將AD參考電壓設為非內部VDD以降低功耗；
- 3、為保證ADC轉換精度，晶片VDD電壓應高於所選ADC內部參考電壓（4V/3V/2V）0.7V以上。

# 12 液晶顯示驅動(LCD)

## 12.1 概述

HC18P23xL具有一個8×32的液晶驅動模組，而且所有的LCD驅動口都可以用作輸入輸出口，與8×16LED驅動模組複用。

在SLEEP模式下預設LCD顯示，如需在SLEEP模式下關閉LCD，則進入休眠前關閉LCD模組（DISPON置0），退出休眠後再打開LCD模組（DISPON置1）。



注：

1. 幀頻計算： $F = F_{LCD} \times N / (L \times 32)$ ；(L為LP分頻，N為1/4duty或1/8duty)；
2. 1/4Duty幀頻範圍（16Hz~250Hz）；1/8Duty幀頻範圍（8~125Hz）；
3. 在選擇1/4duty時，建議使用1:4分頻，在1/8duty時，使用1:2分頻。

## 12.2 LCD 相關寄存器

## 12.2.1 LCD 選擇寄存器

| 2B2h    | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2  | Bit 1  | Bit 0  |
|---------|-------|-------|-------|-------|-------|--------|--------|--------|
| DISPCTR | -     | -     | -     | -     | -     | SLPENB | DISPON | DISPCT |
| R/W     | -     | -     | -     | -     | -     | R/W    | R/W    | R/W    |
| POR的值   | -     | -     | -     | -     | -     | 0      | 0      | 0      |

bit [2] **SLPENB**: SLEEP下LCD使能位

0 = 使能，在SLEEP模式下，LCD正常使用

1 = 禁止，在SLEEP模式下，LCD關閉

bit [1] **DISPON** : LCD顯示使能位元

0 = 禁止LCD/LED驅動

1 = 使能LCD/LED驅動

Bit[2] **DISPCT** : 顯示選擇位元

0 = 選擇LCD驅動模組，LED驅動模組禁止

1 = 選擇LED驅動模組，LCD驅動模組禁止

| 2B0h   | Bit 7   | Bit 6   | Bit 5  | Bit 4  | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|---------|---------|--------|--------|-------|-------|-------|-------|
| LCDCON | BIASCT1 | BIASCT0 | DUTCT1 | DUTCT0 | CS1   | CS0   | RLCD1 | RLCD0 |
| R/W    | R/W     | R/W     | R/W    | R/W    | R/W   | R/W   | R/W   | R/W   |
| POR的值  | 0       | 0       | 0      | 0      | 0     | 0     | 0     | 0     |

bit [7 : 6] **BIASCT[1:0]**: LCD偏置控制位

11 = 1/4 bias

10 = 1/3 bias

01 = 1/2bias (1/4Duty)

00 = 保留

注：在選擇01 = 1/2bias時只能驅動TN型LCD Panel。

bit 5 **DUTCT[1 : 0]**: LCD占空比控制位

00 = 1/8duty (COM0~COM7)

01 = 1/5duty (COM0~COM4) (COM5~COM7作普通IO)

10 = 1/4duty (COM0~COM3) (COM4~COM7作普通IO)

11 = 1/3duty (COM0~COM2) (COM4~COM7作普通IO, COM3為NC不能用作普通IO)

當DUTCT = 0時，COM4~COM7作為LCD COM口；當DUTCT = 1時，COM4~COM7為普通IO。

bit 3-2 **CS[1:0]** : 時鐘源選擇位元

00 = FOSC/1024

01 = T1OSC

1x = WDT RC (在SLEEP模式下LCD顯示，這時即使配置字關閉WDT，WDT也會強制打開)

bit 1-0 **RLCD[1:0]**: 分壓電阻選擇位

00: R1 = R2 = R3 = 270kΩ

01: R1 = R2 = R3 = 90kΩ

10: R1 = R2 = R3 = 30kΩ

11: R1 = R2 = R3 = 2.5kΩ

注：請根據LCD Panel尺寸選擇合適的偏置電阻，避免由於驅動不夠造成顯示效果不佳的現象。Panel較小時可選用270K，較大時可選用2.5K。

## 12.2.2 LCD 對比度控制位

| 2B3h   | Bit 7 | Bit 6 | Bit 5   | Bit 4   | Bit 3  | Bit 2  | Bit 1  | Bit 0  |
|--------|-------|-------|---------|---------|--------|--------|--------|--------|
| LCDLVD | -     | RSEL  | SPDSEL1 | SPDSEL0 | LCDVD3 | LCDVD2 | LCDVD1 | LCDVD0 |
| R/W    | -     | R/W   | R/W     | R/W     | R/W    | R/W    | R/W    | R/W    |
| POR的值  | -     | 0     | 0       | 0       | 0      | 0      | 0      | 0      |

bit [3:0] **LCDVD**:對比度控制位

1xxx = 50% VDD (8/16VDD)

0111 = 56.25% VDD (9/16VDD)

0110 = 62.5% VDD (10/16VDD)

0101 = 68.75% VDD (11/16VDD)

0100 = 75% VDD (12/16VDD)

0011 = 81.25% VDD (13/16VDD)

0010 = 87.5% VDD (14/16VDD)

0001 = 93.75% VDD (15/16VDD)

0000 = VDD (16/16VDD)

bit [5:4] **SPDSEL[1:0]**:加速時間選擇位元

00 =  $1/64 * F_{LCD}$

01 =  $2/64 * F_{LCD}$

10 =  $4/64 * F_{LCD}$

11 =  $8/64 * F_{LCD}$

bit 6 **RSEL**:加速電阻選擇位元

0 = 42.5K $\Omega$

1 = 2.5K $\Omega$

## 12.2.3 LCD 預分頻寄存器

LCD預分頻時鐘選擇寄存器

| 2B1h  | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| LCDPS | -     | -     | -     | -     | LP3   | LP2   | LP1   | LP0   |
| R/W   | -     | -     | -     | -     | R/W   | R/W   | R/W   | R/W   |
| POR的值 | -     | -     | -     | -     | 0     | 0     | 0     | 0     |

bit 3-0 **LP[3:0]** : LCD預分頻比選擇位

1111 = 1:16  
 1110 = 1:15  
 1101 = 1:14  
 1100 = 1:13  
 1011 = 1:12  
 1010 = 1:11  
 1001 = 1:10  
 1000 = 1:9  
 0111 = 1:8  
 0110 = 1:7  
 0101 = 1:6  
 0100 = 1:5  
 0011 = 1:4  
 0010 = 1:3  
 0001 = 1:2  
 0000 = 1:1

## 12.2.4 LCD SEG 輸出控制寄存器

LCD SEG驅動口輸出控制寄存器

| 2B6h ~2B9h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------------|-------|-------|-------|-------|-------|-------|-------|-------|
| SEGSE0     | SE7   | SE6   | SE5   | SE4   | SE3   | SE2   | SE1   | SE0   |
| SEGSE1     | SE15  | SE14  | SE13  | SE12  | SE11  | SE10  | SE9   | SE8   |
| SEGSE2     | SE23  | SE22  | SE21  | SE20  | SE19  | SE18  | SE17  | SE16  |
| SEGSE3     | SE31  | SE30  | SE29  | SE28  | SE27  | SE26  | SE25  | SE24  |
| R/W        | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR的值      | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

bit 7-0 **SE<sub>x</sub>** : SEG<sub>x</sub>的LCD使能位

1 = 使能引腳的SEG<sub>x</sub>功能

0 = 使能引腳的I/O功能

## 12.2.5 LCD RAM

LCD顯示圖元RAM映射位址

| 2C0h ~2DFh | Bit 7    | Bit 6    | Bit 5    | Bit 4    | Bit 3    | Bit 2    | Bit 1    | Bit 0    |
|------------|----------|----------|----------|----------|----------|----------|----------|----------|
| SEGDATA0   | SEG0/C7  | SEG0/C6  | SEG0/C5  | SEG0/C4  | SEG0/C3  | SEG0/C2  | SEG0/C1  | SEG0/C0  |
| SEGDATA1   | SEG1/C7  | SEG1/C6  | SEG1/C5  | SEG1/C4  | SEG1/C3  | SEG1/C2  | SEG1/C1  | SEG1/C0  |
| SEGDATA2   | SEG2/C7  | SEG2/C6  | SEG2/C5  | SEG2/C4  | SEG2/C3  | SEG2/C2  | SEG2/C1  | SEG2/C0  |
| SEGDATA3   | SEG3/C7  | SEG3/C6  | SEG3/C5  | SEG3/C4  | SEG3/C3  | SEG3/C2  | SEG3/C1  | SEG3/C0  |
| SEGDATA4   | SEG4/C7  | SEG4/C6  | SEG4/C5  | SEG4/C4  | SEG4/C3  | SEG4/C2  | SEG4/C1  | SEG4/C0  |
| SEGDATA5   | SEG5/C7  | SEG5/C6  | SEG5/C5  | SEG5/C4  | SEG5/C3  | SEG5/C2  | SEG5/C1  | SEG5/C0  |
| SEGDATA6   | SEG6/C7  | SEG6/C6  | SEG6/C5  | SEG6/C4  | SEG6/C3  | SEG6/C2  | SEG6/C1  | SEG6/C0  |
| SEGDATA7   | SEG7/C7  | SEG7/C6  | SEG7/C5  | SEG7/C4  | SEG7/C3  | SEG7/C2  | SEG7/C1  | SEG7/C0  |
| SEGDATA8   | SEG8/C7  | SEG8/C6  | SEG8/C5  | SEG8/C4  | SEG8/C3  | SEG8/C2  | SEG8/C1  | SEG8/C0  |
| SEGDATA9   | SEG9/C7  | SEG9/C6  | SEG9/C5  | SEG9/C4  | SEG9/C3  | SEG9/C2  | SEG9/C1  | SEG9/C0  |
| SEGDATA10  | SEG10/C7 | SEG10/C6 | SEG10/C5 | SEG10/C4 | SEG10/C3 | SEG10/C2 | SEG10/C1 | SEG10/C0 |
| SEGDATA11  | SEG11/C7 | SEG11/C6 | SEG11/C5 | SEG11/C4 | SEG11/C3 | SEG11/C2 | SEG11/C1 | SEG11/C0 |
| SEGDATA12  | SEG12/C7 | SEG12/C6 | SEG12/C5 | SEG12/C4 | SEG12/C3 | SEG12/C2 | SEG12/C1 | SEG12/C0 |
| SEGDATA13  | SEG13/C7 | SEG13/C6 | SEG13/C5 | SEG13/C4 | SEG13/C3 | SEG13/C2 | SEG13/C1 | SEG13/C0 |
| SEGDATA14  | SEG14/C7 | SEG14/C6 | SEG14/C5 | SEG14/C4 | SEG14/C3 | SEG14/C2 | SEG14/C1 | SEG14/C0 |
| SEGDATA15  | SEG15/C7 | SEG15/C6 | SEG15/C5 | SEG15/C4 | SEG15/C3 | SEG15/C2 | SEG15/C1 | SEG15/C0 |
| SEGDATA16  | SEG16/C7 | SEG16/C6 | SEG16/C5 | SEG16/C4 | SEG16/C3 | SEG16/C2 | SEG16/C1 | SEG16/C0 |
| SEGDATA17  | SEG17/C7 | SEG17/C6 | SEG17/C5 | SEG17/C4 | SEG17/C3 | SEG17/C2 | SEG17/C1 | SEG17/C0 |
| SEGDATA18  | SEG18/C7 | SEG18/C6 | SEG18/C5 | SEG18/C4 | SEG18/C3 | SEG18/C2 | SEG18/C1 | SEG18/C0 |
| SEGDATA19  | SEG19/C7 | SEG19/C6 | SEG19/C5 | SEG19/C4 | SEG19/C3 | SEG19/C2 | SEG19/C1 | SEG19/C0 |
| SEGDATA20  | SEG20/C7 | SEG20/C6 | SEG20/C5 | SEG20/C4 | SEG20/C3 | SEG20/C2 | SEG20/C1 | SEG20/C0 |
| SEGDATA21  | SEG21/C7 | SEG21/C6 | SEG21/C5 | SEG21/C4 | SEG21/C3 | SEG21/C2 | SEG21/C1 | SEG21/C0 |
| SEGDATA22  | SEG22/C7 | SEG22/C6 | SEG22/C5 | SEG22/C4 | SEG22/C3 | SEG22/C2 | SEG22/C1 | SEG22/C0 |
| SEGDATA23  | SEG23/C7 | SEG23/C6 | SEG23/C5 | SEG23/C4 | SEG23/C3 | SEG23/C2 | SEG23/C1 | SEG23/C0 |
| SEGDATA24  | SEG24/C7 | SEG24/C6 | SEG24/C5 | SEG24/C4 | SEG24/C3 | SEG24/C2 | SEG24/C1 | SEG24/C0 |
| SEGDATA25  | SEG25/C7 | SEG25/C6 | SEG25/C5 | SEG25/C4 | SEG25/C3 | SEG25/C2 | SEG25/C1 | SEG25/C0 |
| SEGDATA26  | SEG26/C7 | SEG26/C6 | SEG26/C5 | SEG26/C4 | SEG26/C3 | SEG26/C2 | SEG26/C1 | SEG26/C0 |
| SEGDATA27  | SEG27/C7 | SEG27/C6 | SEG27/C5 | SEG27/C4 | SEG27/C3 | SEG27/C2 | SEG27/C1 | SEG27/C0 |
| SEGDATA28  | SEG28/C7 | SEG28/C6 | SEG28/C5 | SEG28/C4 | SEG28/C3 | SEG28/C2 | SEG28/C1 | SEG28/C0 |
| SEGDATA29  | SEG29/C7 | SEG29/C6 | SEG29/C5 | SEG29/C4 | SEG29/C3 | SEG29/C2 | SEG29/C1 | SEG29/C0 |
| SEGDATA30  | SEG30/C7 | SEG30/C6 | SEG30/C5 | SEG30/C4 | SEG30/C3 | SEG30/C2 | SEG30/C1 | SEG30/C0 |
| SEGDATA31  | SEG31/C7 | SEG31/C6 | SEG31/C5 | SEG31/C4 | SEG31/C3 | SEG31/C2 | SEG31/C1 | SEG31/C0 |
| R/W        | R/W      | R/W      | R/W      | R/W      | R/W      | R/W      | R/W      | R/W      |
| POR的值      | X        | X        | X        | X        | X        | X        | X        | X        |

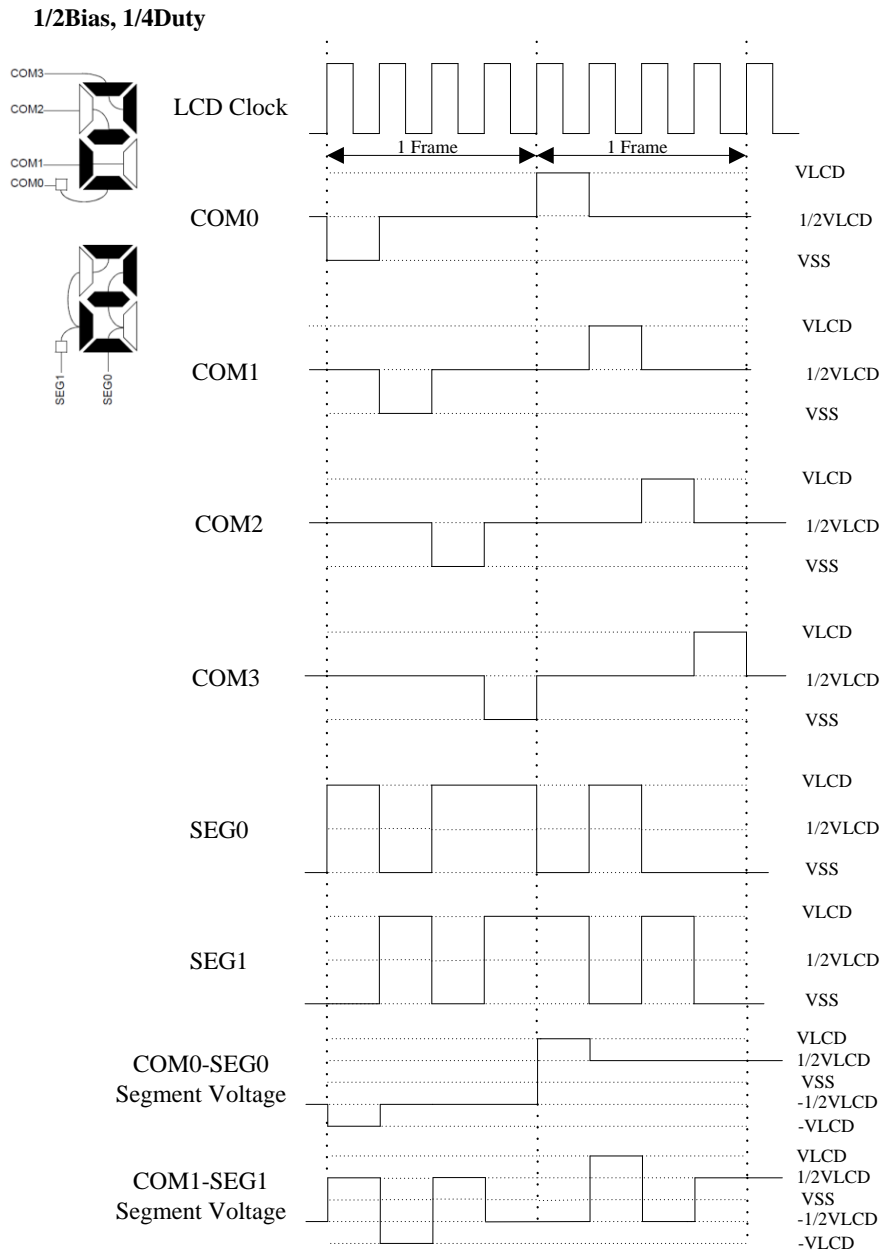
bit 7-0 SEGx/Cy : SEGx、COMy圖元點亮控制位元

0 = 點亮圖元 (不透明)

1 = 不點亮象素 (透明)

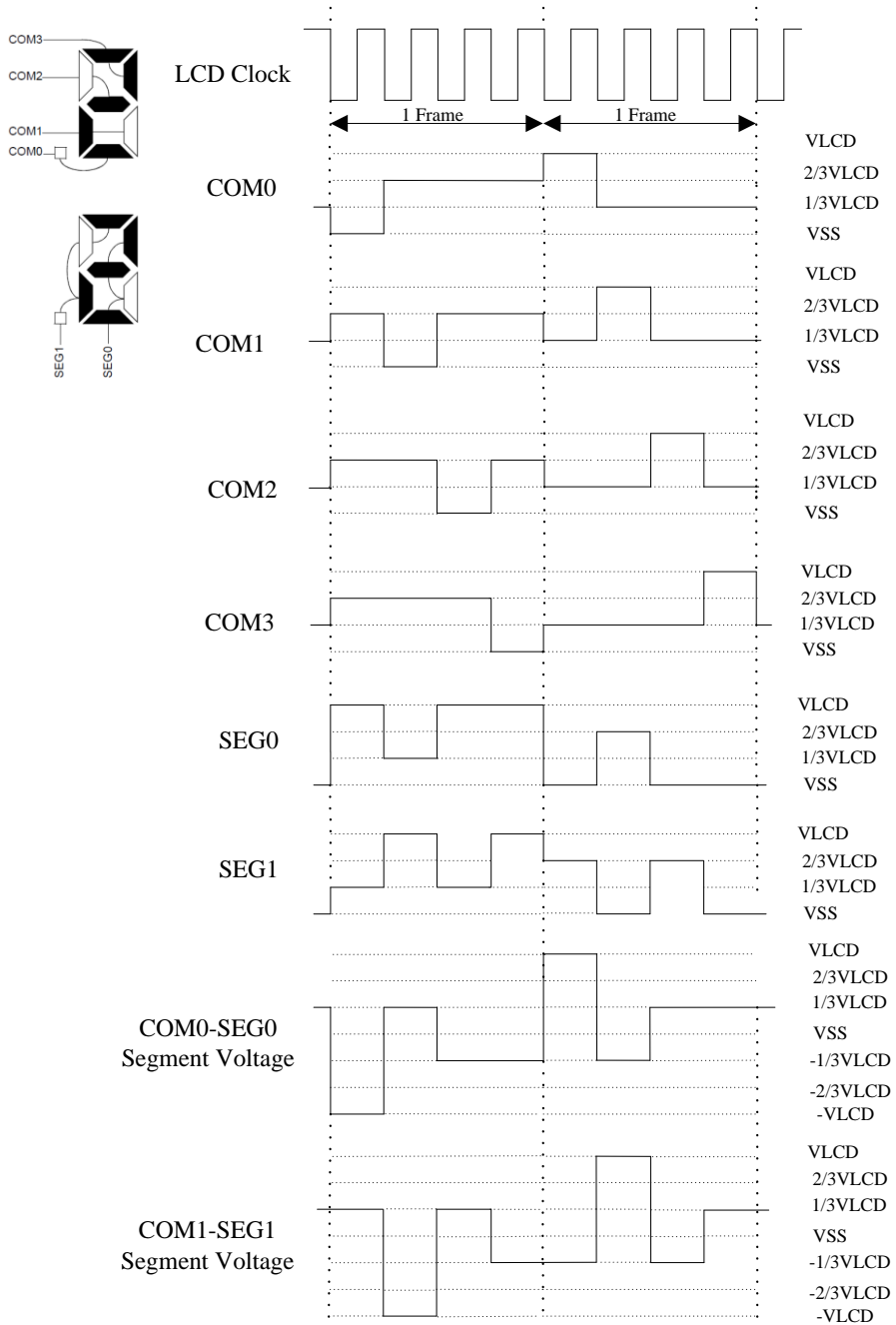
使用說明：HC18P23xL亮度控制為0有效，即0點亮圖元，1不點亮，在應用中請注意。

### 12.3 LCD 波形時序圖

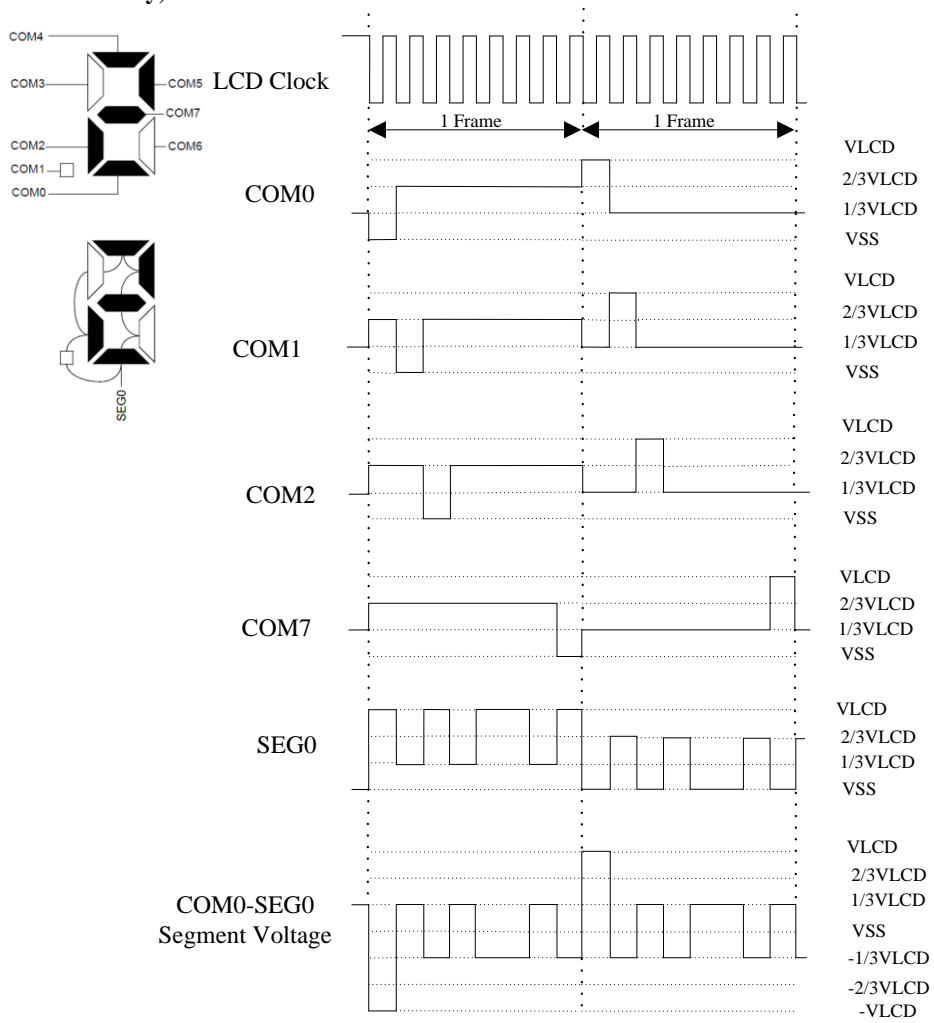




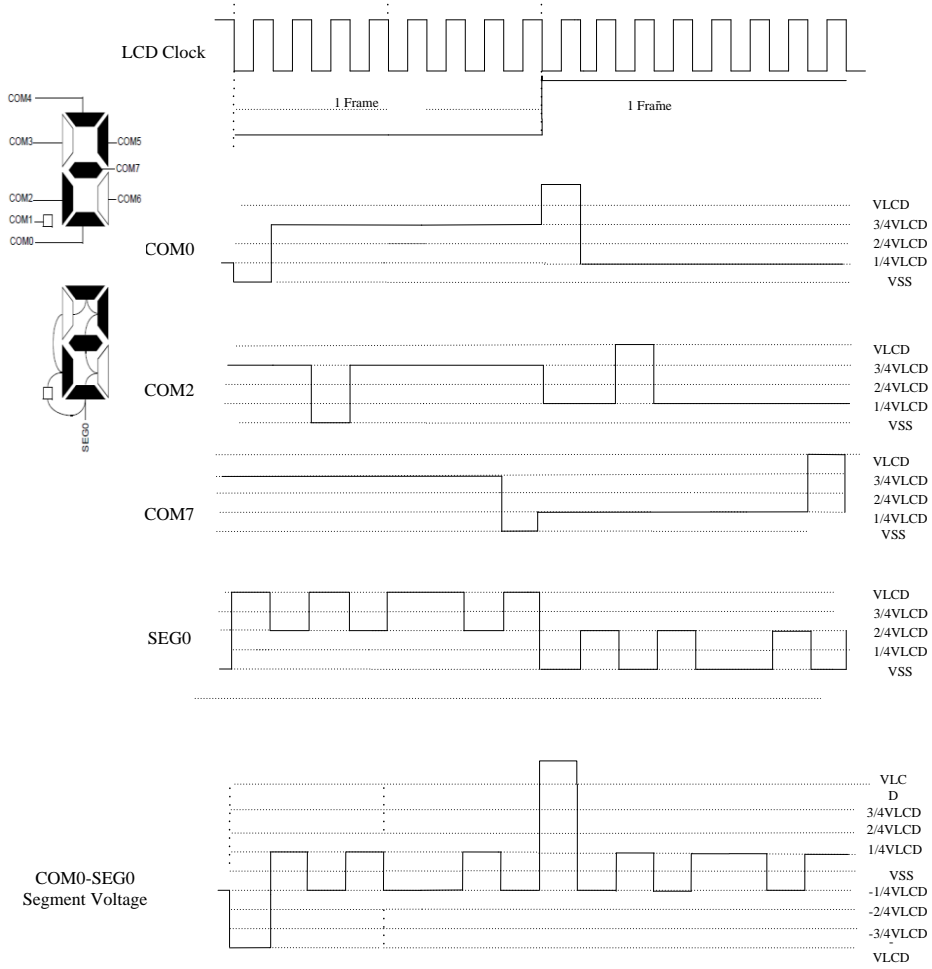
1/3Bias, 1/4Duty



**1/8Duty,1/3Bias**



1/4Bais, 1/8Duty



# 13 LED驅動器

## 13.1 概述

HC18P23xL具有LED驅動功能通過輸出驅動外部LED點陣設備，可驅動8×16 LED點陣。

## 13.2 LED 控制寄存器

| 2B2h    | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2  | Bit 1  | Bit 0  |
|---------|-------|-------|-------|-------|-------|--------|--------|--------|
| DISPCTR | -     | -     | -     | -     | -     | SLPENB | DISPON | DISPCT |
| R/W     | -     | -     | -     | -     | -     | R/W    | R/W    | R/W    |
| POR的值   | -     | -     | -     | -     | -     | 0      | 0      | 0      |

bit [0] **DISPCT**：顯示選擇位元

0 = 選擇LCD驅動模組，LED驅動模組禁止

1 = 選擇LED驅動模組，LCD驅動模組禁止

bit [1] **DISPON**：LED顯示使能位元

0 = 禁止LED驅動

1 = 使能LED驅動

在 LED 模式下請將 DISPCT 置 1。

| 2B0h   | Bit 7   | Bit 6   | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|---------|---------|-------|-------|-------|-------|-------|-------|
| LCDCON | BIASCT1 | BIASCT0 | DUTCT | -     | CS1   | CS0   | RLCD1 | RLCD0 |
| R/W    | R/W     | R/W     | R/W   | -     | R/W   | R/W   | R/W   | R/W   |
| POR的值  | 0       | 0       | 0     | -     | 0     | 0     | 0     | 0     |

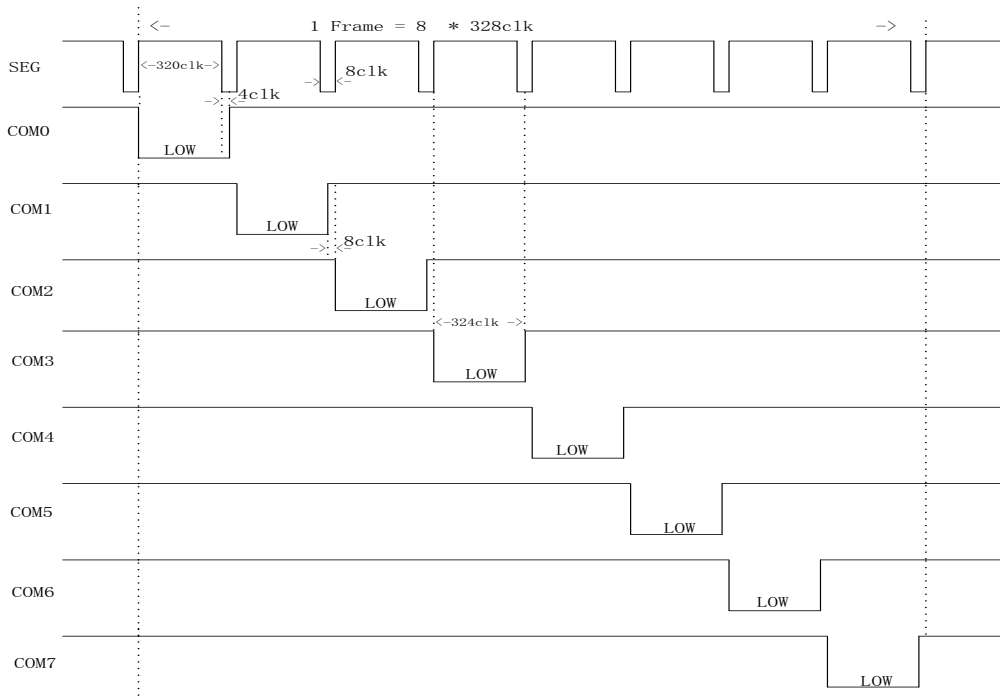
bit 3-2 **CS[1:0]**：時鐘源選擇位元

00 = FOSC/128 (256KHz)

01 = T1OSC (256KHz)

1x = 保留

### 13.3 LED 驅動波形



注：1 CLK = 1/256KHz

### 13.4 LED 亮度控制寄存器

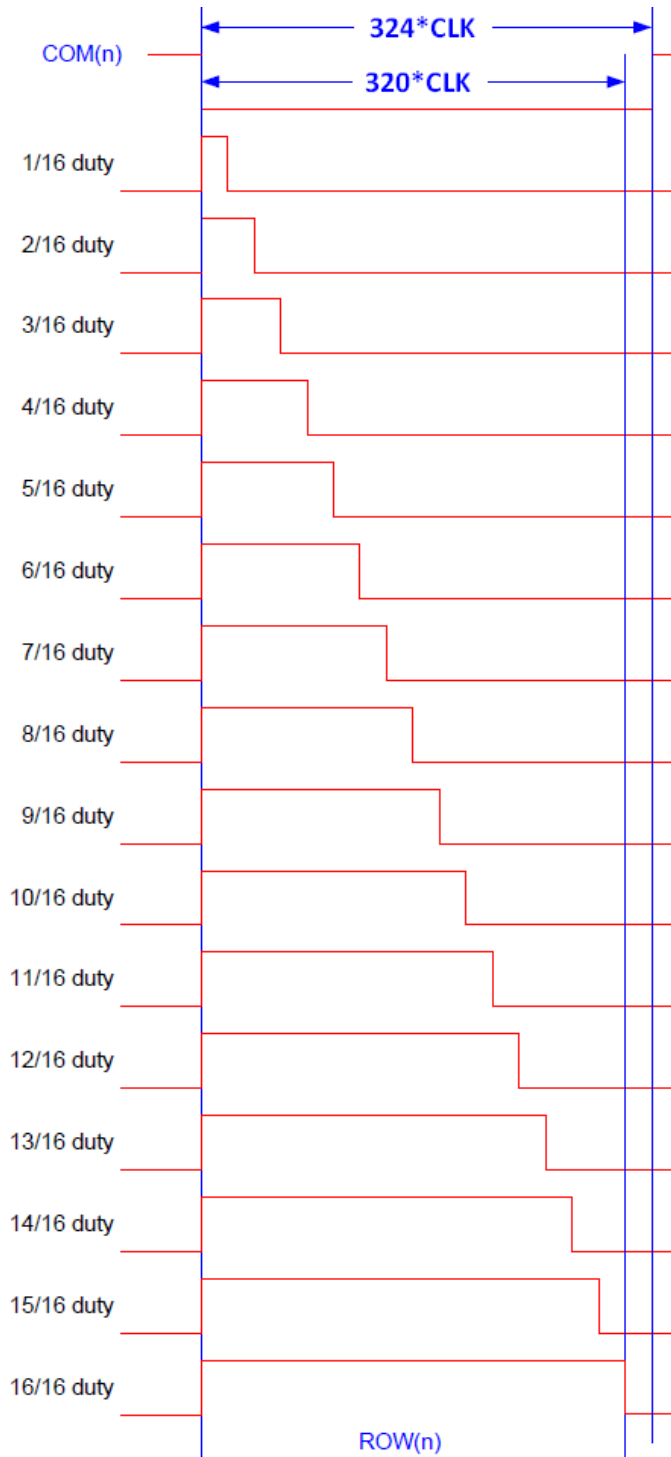
在 LED 驅動模式下，配置 DISPS 寄存器，調節 LED 亮度

| 2B1h  | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| DISPS | -     | -     | -     | -     | LP3   | LP2   | LP1   | LP0   |
| R/W   | -     | -     | -     | -     | R/W   | R/W   | R/W   | R/W   |
| POR的值 | -     | -     | -     | -     | 0     | 0     | 0     | 0     |

bit 3-0 LP[3:0]：LED亮度控制位

- 1111 = 1/16Duty
- 1110 = 2/16Duty
- 1101 = 3/16Duty
- 1100 = 4/16Duty
- 1011 = 5/16Duty
- 1010 = 6/16Duty
- 1001 = 7/16Duty
- 1000 = 8/16Duty
- 0111 = 9/16Duty
- 0110 = 10/16Duty
- 0101 = 11/16Duty
- 0100 = 12/16Duty
- 0011 = 13/16Duty
- 0010 = 14/16Duty
- 0001 = 15/16Duty
- 0000 = 16/16Duty

配置亮度控制位元時，SEG 波形如下：



### 13.5 說明

在配置為 LED 驅動時，SEG 輸出需選擇 SEG0~SEG15，LED 對應為 SEGDATA0~SEGDATA15。

# 14 串列口通信

## 14.1 概述

HC18P23xL 具有 1 個採用 UART(Universal Asynchronous Receiver/Transmitter)工作方式的全雙工串列通信介面。串列口由 2 個資料緩衝器、一個移位暫存器、一個串列控制寄存器和一個串列傳輸速率發生器等組成。串列口的資料緩衝器由 2 個互相獨立的接收、發送緩衝器構成，可以同時發送和接收資料。發送緩衝器只能寫入而不能讀出，接收緩衝器只能讀出而不能寫入，因而兩個緩衝器可以共用一個位址碼。串列口的兩個緩衝器共用的位址碼是 23Dh。兩個緩衝器統稱串列通信特殊功能寄存器 SBUF。

HC18P23xL 的串列口都有 4 種工作方式，其中兩種方式的串列傳輸速率是可變的，另兩種是固定的，以供不同應用場合選用。使用者可用軟體設置不同的串列傳輸速率和選擇不同的工作方式。主機可通過查詢或中斷方式對接收/發送進程式處理，使用十分靈活。

串列口對應的硬體部分是 TxD 和 RxD 引腳。

## 14.2 UART 相關寄存器

### 14.2.1 串列口的控制寄存器 SCON

|        |        |       |       |       |       |       |       |       |
|--------|--------|-------|-------|-------|-------|-------|-------|-------|
| 23Ch   | Bit 7  | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| SCON   | SM0/FE | SM1   | SM2   | REN   | TB8   | RB8   | RXWK  | -     |
| R/W    | R/W    | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | -     |
| POR 的值 | 0      | 0     | 0     | 0     | 0     | x     | 0     | -     |

bit [7:6] SM0/SM1：串口工作方式選擇位元

| SM0 | SM1 | 工作方式 | 功能說明            | 串列傳輸速率                                                                  |
|-----|-----|------|-----------------|-------------------------------------------------------------------------|
| 0   | 0   | 方式0  | 同步移位元串列方式：移位暫存器 | 當 UX6 = 0 時，串列傳輸速率是 $F_{CPU} / 12$<br>當 UX6 = 1 時，串列傳輸速率是 $F_{CPU} / 2$ |
| 0   | 1   | 方式1  | 8位UART，串列傳輸速率可變 | BRT獨立串列傳輸速率發生器的溢出率/16                                                   |
| 1   | 0   | 方式2  | 9位UART          | $(2^{SMOD} / 64) \times F_{CPU}$                                        |
| 1   | 1   | 方式3  | 9位UART，串列傳輸速率可變 | BRT獨立串列傳輸速率發生器的溢出率/16                                                   |

**FE**：幀錯誤檢測位元

1 = 有偵錯誤，硬體置 1

0 = 無偵錯誤或軟體清零

bit 5 **SM2**：多機通信使能控制位(第九位“1”校驗器)

1 = 在方式 1 下，允許停止位確認檢驗，只有有效的停止位“1”才能置位 RXIF

在方式 2 和 3 下，只有位元組 (第 9 位元 = “1”) 才能置位 RXIF

0 = 在方式 1 下，禁止停止位確認檢驗，任何停止位元都會置位元 RXIF

在方式 2 和 3 下，任何位元組都會置位元 RXIF

bit 4 **REN**：串列接收使能控制位元

1 = 允許串列接收

0 = 禁止串列接收

bit 3 **TB8**：方式2/方式3時，為要發送的第9位元資料，有軟體置1或清零

bit 2 **RB8**：方式2/方式3時，為要發送的第9位元資料，作為同位檢查位元或者位址幀/資料幀的標誌位元

bit 1 **RXWK**：接收中斷喚醒使能位

1 = 允許RXD下降沿置RXIF，允許RXD喚醒SLEEP；

0 = 允許RXD下降沿置RXIF，禁止RXD喚醒SLEEP；

### 14.2.2 串列口資料緩衝寄存器 SBUF

| 23Dh  | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| SBUF  |       |       |       |       |       |       |       |       |
| R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR的值 | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

bit [7:0] SBUF[7:0]

串口緩衝寄存器，寫為需要發送的資料，讀為接收到的資料

串列口1緩衝寄存器(SBUF)的位址是23Dh，實際是2個緩衝器，寫SBUF的操作完成待發送資料的載入，讀SBUF的操作可獲得已接收到的資料。兩個操作分別對應兩個不同的寄存器，1個是只寫寄存器，1個是唯讀寄存器。

串列通道內設有資料寄存器。在所有的串列通信方式中，在寫入SBUF信號的控制下，把資料裝入相同的9位移位暫存器，前面8位元為資料位元組，其最低位元為移位暫存器的輸出位。根據不同的工作方式會自動將“1”或TB8的值裝入移位暫存器的第9位，並進行發送。

串列通道的接收寄存器是一個輸入移位暫存器。在方式0時它的字長為8位元，其他方式時為9位元。當一幀接收完畢，移位暫存器中的資料位元組裝入串列資料緩衝器SBUF中，其第9位則裝入SCON寄存器中的RB8位。如果由於SM2使得已接收到的資料無效時，RB8和SBUF中內容不變。

由於接收通道內設有輸入移位暫存器和SBUF緩衝器，從而能使一幀接收完將資料由移位暫存器裝入SBUF後，可立即開始接收下一幀資訊，主機應在該幀接收結束前從SBUF緩衝器中將資料取走，否則前一幀資料將丟失。SBUF以並行方式送往內部資料匯流排。

### 14.2.3 輔助寄存器 AUXR

| 23Bh  | Bit 7 | Bit 6  | Bit 5  | Bit 4 | Bit 3  | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|--------|--------|-------|--------|-------|-------|-------|
| AUXR  | -     | UARTEN | UARTM0 | BRTR  | BRTX12 | S1BRS | SMOD  | SMOD0 |
| R/W   | -     | R/W    | R/W    | R/W   | R/W    | R/W   | R/W   | R/W   |
| POR的值 | -     | 0      | 0      | 0     | 0      | x     | 0     | 0     |

bit 6 **UARTEN**：UART管腳控制位

1 = 使能UART管腳功能

0 = 禁止UART管腳功能，做普通IO口

bit 5 **UARTM0**：串口模式0的通信速度設置位元

1 = UART串口模式0 2分頻



0 = UART 串口模式 0 12 分頻

bit 4 **BRTR**：獨立串列傳輸速率發生器運行控制位元

1 = 允許獨立串列傳輸速率發生器運行

0 = 不允許獨立串列傳輸速率發生器運行

bit 3 **BRTX12**：獨立串列傳輸速率發生器計數控制位元

1 = 獨立串列傳輸速率發生器每 1 個時鐘計數一次

0 = 獨立串列傳輸速率發生器每 12 個時鐘計數一次

bit 2 **S1BRS**：串口（UART）的串列傳輸速率發生器控制位元

1 = 選擇獨立串列傳輸速率發生器作為串口（UART）串列傳輸速率發生器

0 = 未用

bit 1 **SMOD**：串列傳輸速率選擇位

1 = 串列通信方式 1、2、3 的串列傳輸速率加倍

0 = 串列通信方式 1、2、3 的串列傳輸速率不加倍

bit 0 **SMOD0**：幀錯誤檢測有效控制位元

1 = SCON 寄存器中的 SM0/FE 應用於 FE（幀錯誤檢測）功能

0 = SCON 寄存器中的 SM0/FE 應用於 SM0 功能，和 SM1 一起指定串列口的工作方式

## 14.2.4 獨立串列傳輸速率發生器寄存器 BRT

| 23Ah   | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|-------|-------|-------|-------|-------|-------|-------|-------|
| BRT    |       |       |       |       |       |       |       |       |
| R/W    | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR 的值 | 0     | 0     | 0     | 0     | 0     | x     | 0     | 0     |

獨立串列傳輸速率發生器寄存器 BRT 用於保存重裝時間常數。

## 14.2.5 自動位址識別寄存器

| 23Eh   | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|-------|-------|-------|-------|-------|-------|-------|-------|
| SADEN  |       |       |       |       |       |       |       |       |
| R/W    | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR 的值 | 0     | 0     | 0     | 0     | 0     | x     | 0     | 0     |

Bit[7:0] SADEN[7:0]：從機位址寄存器

| 23Fh   | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|-------|-------|-------|-------|-------|-------|-------|-------|
| SADDR  |       |       |       |       |       |       |       |       |
| R/W    | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR 的值 | 0     | 0     | 0     | 0     | 0     | x     | 0     | 0     |

Bit[7:0] SADDR[7:0]：從機位址遮罩寄存器

### 14.3 工作方式

UART有4種工作方式，在四種方式中，任何將SBUF作為目標寄存器的寫操作都會啟動發送。在方式0中由條件RXIF= 0和REN = 1初始化接收。這會在TXD引腳上產生一個時鐘信號，然後在RXD引腳上移出8位元資料。在其它方式中由輸入的起始位初始化接收（如果RXIF = 0和REN = 1）。外部發送器通信以發送起始位開始。在發送之前TXD引腳必須被設置為輸出高電平。

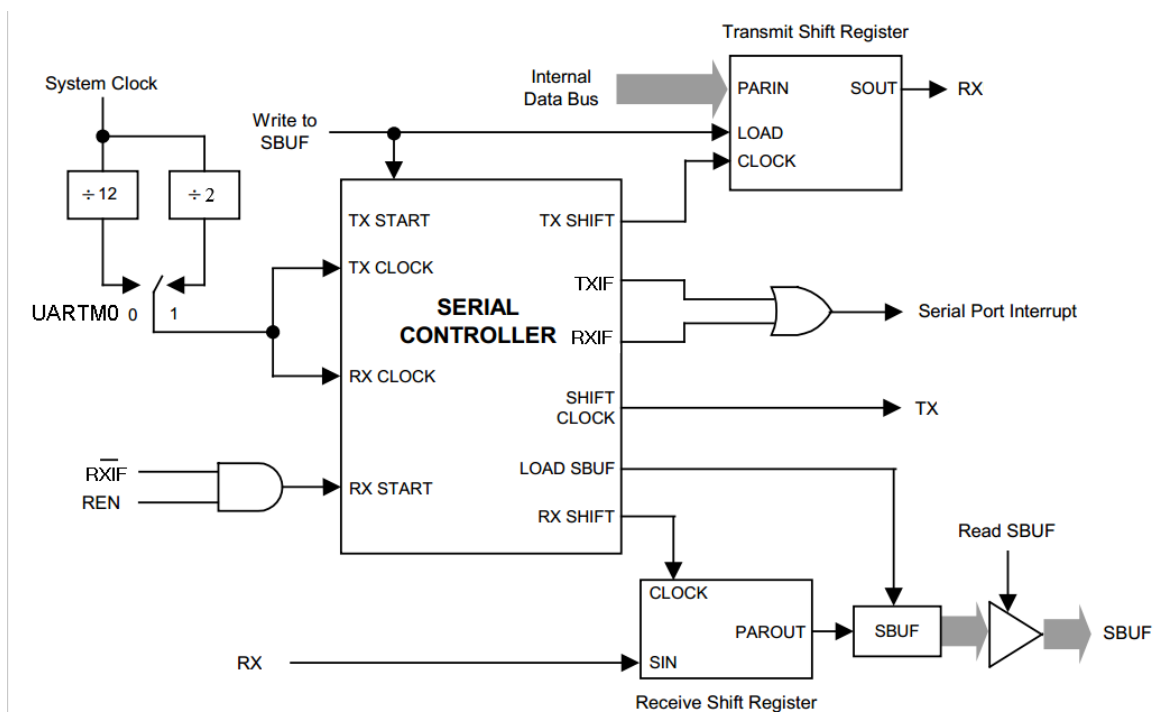
| SM0 | SM1 | 工作方式 | 類型  | 串列傳輸速率                                |
|-----|-----|------|-----|---------------------------------------|
| 0   | 0   | 方式0  | 同步  | 串列傳輸速率是 $F_{CPU} / 12 \times 6^{UX6}$ |
| 0   | 1   | 方式1  | 非同步 | BRT獨立串列傳輸速率發生器的溢出率/16                 |
| 1   | 0   | 方式2  | 非同步 | $(2^{SMOD} / 64) \times F_{CPU}$      |
| 1   | 1   | 方式3  | 非同步 | BRT獨立串列傳輸速率發生器的溢出率/16                 |

#### 14.3.1 方式 0：同步半雙工通訊

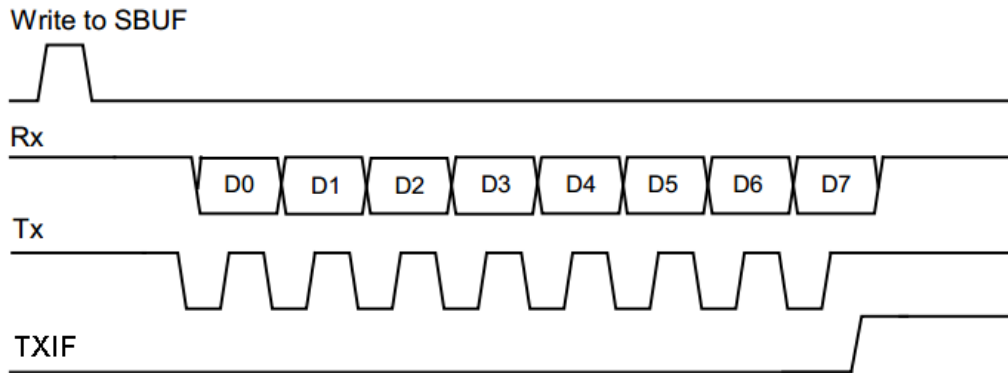
方式0支援與外部設備的同步通信，在RX引腳上收發串列資料，TX引腳發送移位元時鐘。HC18P23xL提供TX引腳上的移位元時鐘，因此這種方式是串列通信的半雙工方式。在這個方式中，每幀收發8位，低位先接收或發送。

通過置UARTM0位為0或1，串列傳輸速率固定為 $F_{CPU}$ 的1/12或1/2。當UARTM0位等於0時，序列埠以 $F_{CPU}$ 的1/12運行，當UARTM0位等於1時，序列埠以 $F_{CPU}$ 的1/2運行。

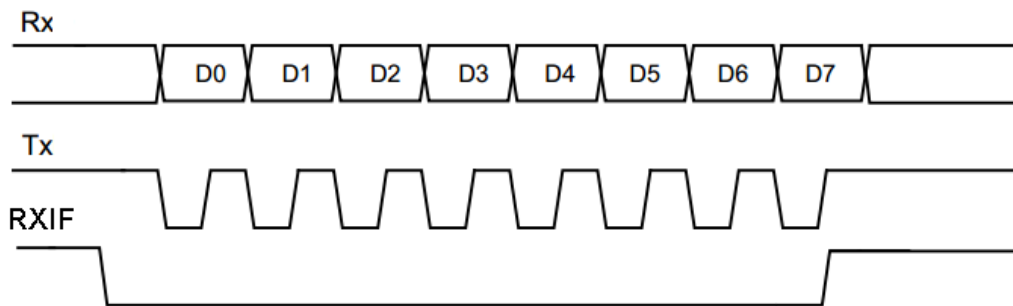
功能塊框圖如下圖所示，資料通過 RX 引腳移入和移出序列埠，移位元時鐘由 TX 引腳輸出。



任何將 SBUF 作為目標寄存器的寫操作都會啟動發送。下一個系統時鐘 TX 控制塊開始發送。資料轉換發生在移位元時鐘的下降沿，移位暫存器的內容逐次從左往右移位，空位置 0。當移位暫存器中的所有 8 位都發送後，TX 控制模組停止發送操作，然後在下一個系統時鐘的上升沿將 TXIF 位置 1。

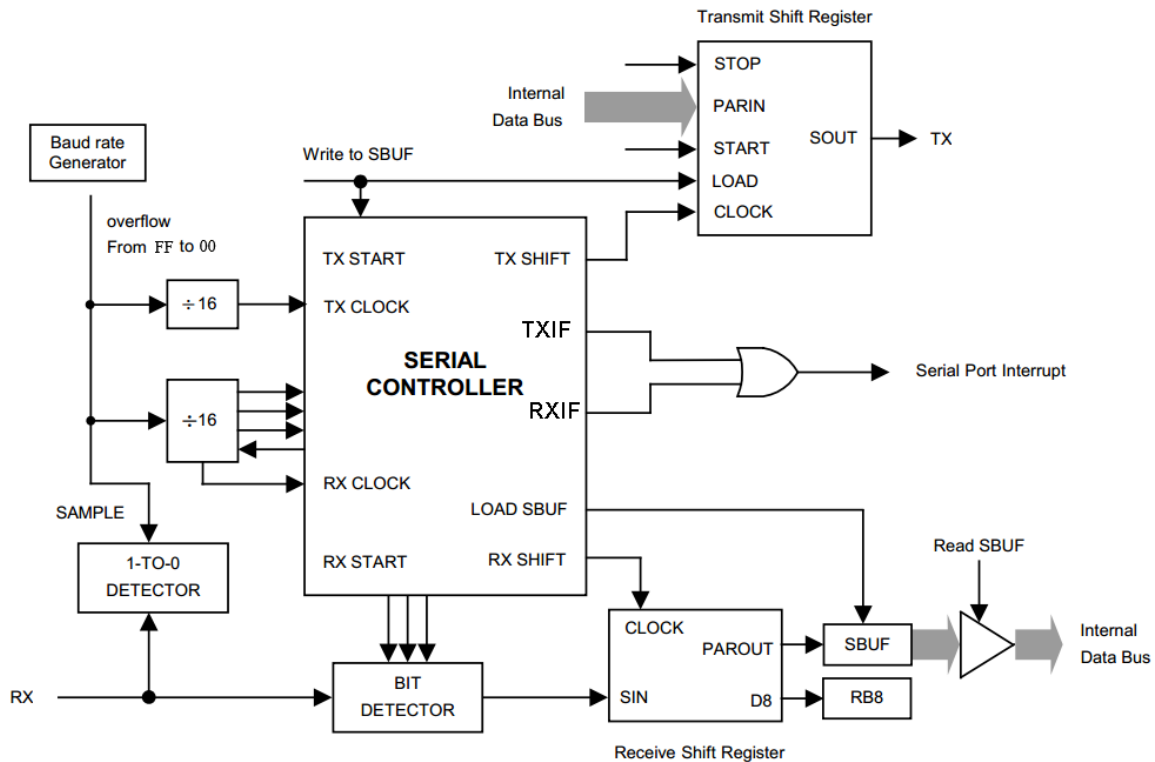


REN 位置 1 和 RXIF 位清零初始化接收。下一個系統時鐘啟動接收，在移位元時鐘的上升沿鎖存資料，接收轉換寄存器的內容逐次向左移位元。當所有 8 位元資料都移到移位暫存器中後，RX 控制塊停止接收，在下一個系統時鐘的上升沿 RXIF 置位元，直到被軟體清零才允許下一次接收。

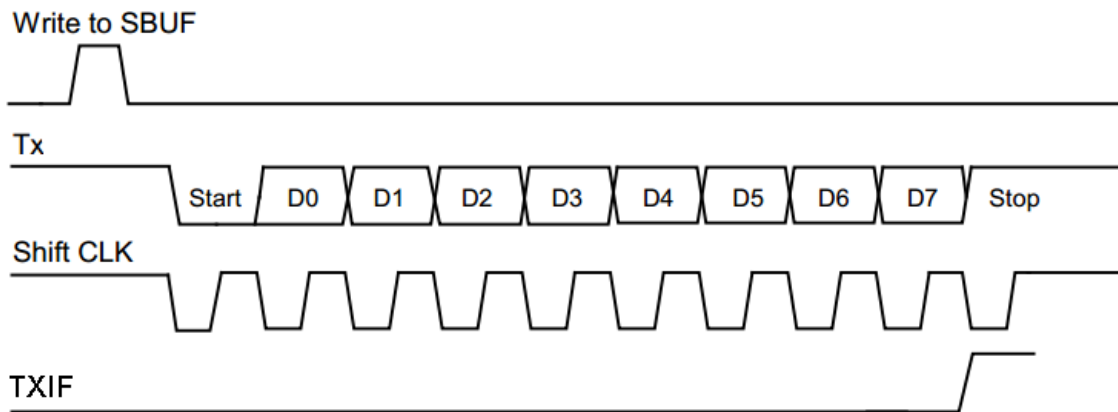


### 14.3.2 方式 1：8 位元 UART，可變串列傳輸速率，非同步全雙工

方式 1 提供 10 位元全雙工非同步通信，10 位元由一個起始位元（邏輯 0），8 個資料位元（低位元在前）和一個停止位元（邏輯 1）組成。在接收時，這 8 個資料位元存儲在 SBUF 中而停止位元儲存在 RB8 中。方式 1 中的串列傳輸速率固定為自帶串列傳輸速率發生器溢出率的 1/16。功能塊框圖如下圖所示。



任何將 SBUF 作為目標寄存器的寫操作都會啟動發送，實際上發送是從 16 分頻計數器中的下一次跳變之後的系統時鐘開始的，因此位元時間與 16 分頻計數器是同步的，與對 SBUF 的寫操作不同步。起始位首先在 TX 引腳上移出，然後是 8 位元資料位元。在發送移位暫存器中的所有 8 位元資料都發送完後，停止位在 TX 引腳上移出，在停止位元發出的同時 TXIF 標誌置位元。

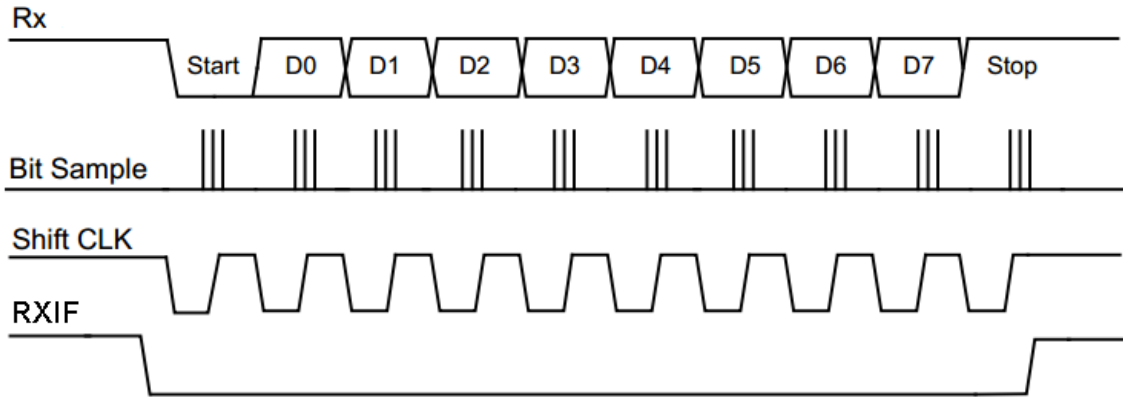


只有REN置1時才允許接收。當RX引腳檢測到下降沿時串行口開始接收串行資料。為此，CPU對RX不斷採樣，採樣速率為串行傳輸速率的16倍。當檢測下降沿時，16分頻計數器立即復位，這有助於16分頻計數器與RX引腳上的串行資料位元同步。16分頻計數器把每一位的時間分為16個狀態，在第7、8、9狀態時，位元檢測器對RX端的電平進行採樣。為抑制雜訊，在這3個狀態採樣中至少有2次採樣值一致資料才被接收。如果所接收的第一位不是0，說明這位元不是一幀資料的起始位元，該位元被忽略，接收電路被重定，等待RX引腳上另一個下降沿的到來。若起始位有效，則移入移位暫存器，並接著移入其它位到移位暫存器。8個資料位元和1個停止位元（包含錯誤的停止位元，詳細見寄存器SM2位說明）移入之後，移位暫存器的內容和停止位元(包含錯誤的停止位元)被分別裝入SBUF和RB8中，RXIF置1，但必須滿足下列條件：

- (1) RXIF = 0

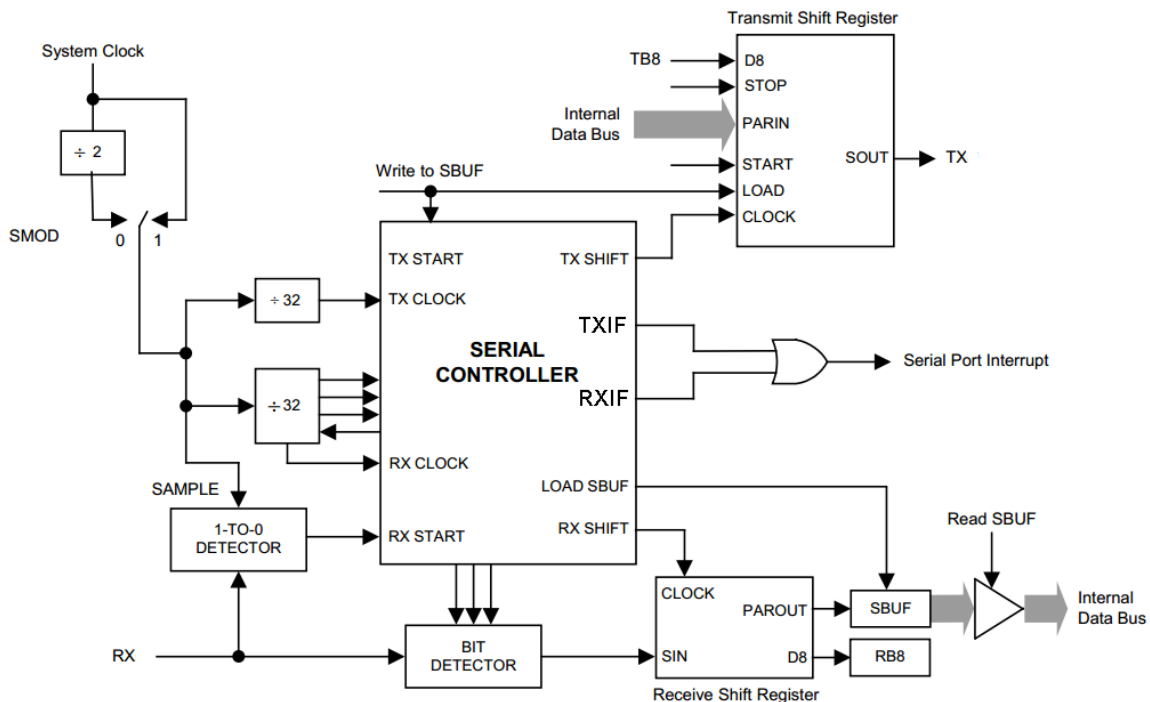
(2) SM2 = 0或者接收的停止位= 1

如果這些條件被滿足，那麼停止位元（包含錯誤的停止位元）裝入 RB8，8 個資料位元裝入 SBUF，RXIF 被置位。否則接收的幀會丟失。這時，接收器將重新去探測 RX 端是否另一個下降沿。使用者必須用軟體清零 RXIF，然後才能再次接收。

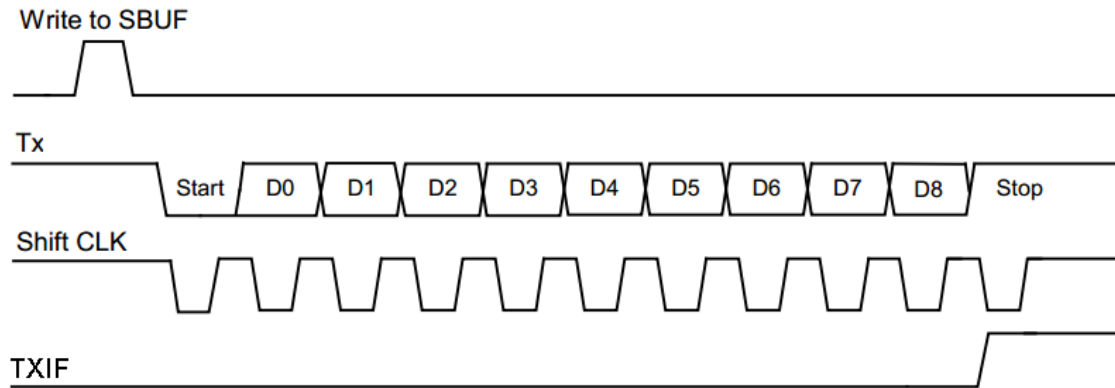


### 14.3.3 方式 2：9 位元 UART，固定串列傳輸速率，非同步全雙工

這個方式使用非同步全雙工通信中的 11 位。一幀由一個起始位元（邏輯 0），8 個資料位元（低位元在前），一個可程式設計的第 9 資料位元和一個停止位元（邏輯 1）組成。方式 2 支援多機通信和硬體位址識別（詳見多機通訊章節）。在資料傳送時，第 9 資料位元（TB8 位元）可以寫 0 或 1。當接收到資料時，第 9 資料位移入 RB8 而停止位不保存。SMOD 位元選擇串列傳輸速率為系統工作頻率的 1/32 或 1/64。功能塊框圖如下所示。



任何將 SBUF 作為目標寄存器的寫操作都會啟動發送，同時也將 TB8 載入到發送移位暫存器的第 9 位中。實際上發送是從 16 分頻計數器中的下一次跳變之後的系統時鐘開始的，因此位元時間與 16 分頻計數器是同步的，與對 SBUF 的寫操作不同步。起始位首先在 TX 引腳上移出，然後是 9 位元資料。在發送轉換寄存器中的所有 9 位元資料都發送完後，停止位在 TX 引腳上移出，在停止位元開始發送時 TXIF 標誌置位元。

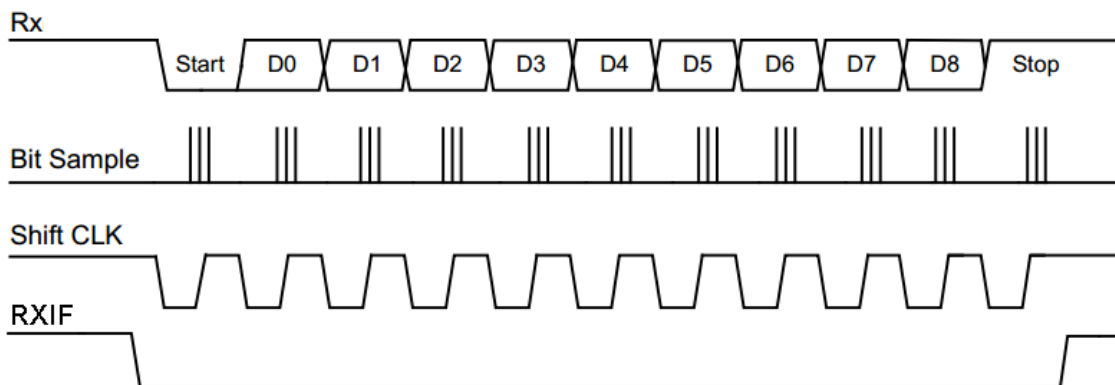


只有REN置位時才允許接收。當RX引腳檢測到下降沿時串列口開始接收串列資料。為此，CPU對RX不斷採樣，採樣速率為串列傳輸速率的16倍。當檢測下降沿時，16分頻計數器立即復位。這有助於16分頻計數器與RX引腳上的串列資料位元同步。16分頻計數器把每一位的時間分為16個狀態，在第7、8、9狀態時，位元檢測器對RX端的電平進行採樣。為抑制雜訊，在這3個狀態採樣中至少有2次採樣值一致資料才被接收。如果所接收的第一位不是0，說明這位元不是一幀資料的起始位元，該位元被忽略，接收電路被重定，等待RX引腳上另一個下降沿的到來。若起始位有效，則移入移位暫存器，並接著移入其它位到移位暫存器。9個資料位元和1個停止位移入之後，移位暫存器的內容被分別裝入SBUF和RB8中，RXIF置1，但必須滿足下列條件：

- (1) RXIF = 0
- (2) SM2 = 0 或者接收的第9位 = 1，且接收的位元組符合約定從機位址

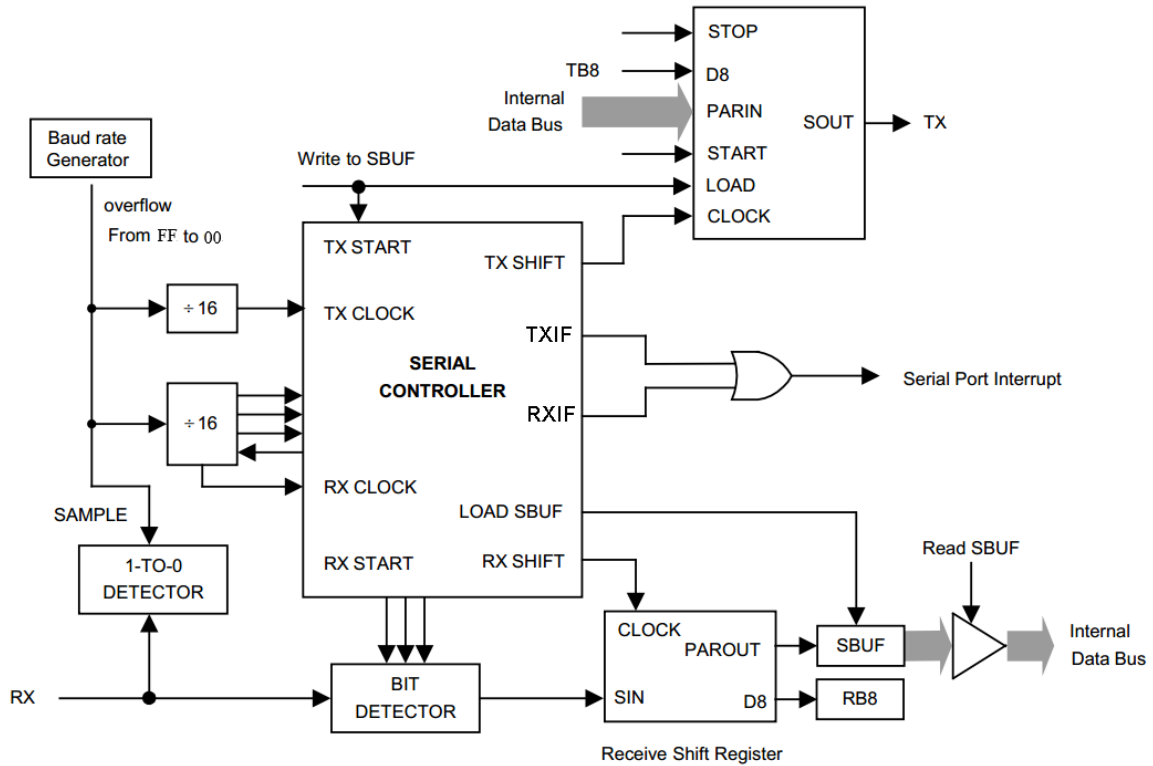
如果這些條件被滿足，那麼第9位移入RB8，8位元數據移入SBUF，RXIF被置位。否則接收的資料幀會丟失。

在停止位的當中，接收器回到尋找RX引腳上的另一個下降沿。使用者必須用軟體清除RXIF，然後才能再次接收。



### 14.3.4 方式 3：9 位元 UART，可變串列傳輸速率，非同步全雙工

方式 3 使用方式 2 的傳輸協定以及方式 1 的串列傳輸速率產生方式。



## 14.4 串列傳輸速率發生器

UART 自帶一個串列傳輸速率發生器，它實質上就是一個 8 位遞增計數器。

在方式0中，串列傳輸速率可程式設計為系統時鐘的1/12或1/2，由UARTM0位決定。當UARTM0為0時，序列埠在F<sub>CPU</sub>的1/12下運行。當UX6為1時，序列埠在F<sub>CPU</sub>的1/2下運行。

在方式2中，串列傳輸速率固定為系統時鐘的1/32或1/64，由SMOD位中決定。當SMOD位為0時，UART以F<sub>CPU</sub>的1/64運行。當SMOD位為1時，UART以F<sub>CPU</sub>的1/32運行。

$$\text{Baud} = 2^{\text{SMOD}} \times \left(\frac{F_{\text{CPU}}}{64}\right)$$

在方式1和方式3中，串列傳輸速率公式如下：

$$\text{Baud} = \frac{F_{\text{CPU}}}{16 \times (256 - \text{SBRT})}$$

下表為常用CPU頻率與常用串列傳輸速率所對應的串列傳輸速率發生器的重載值：

| 常用<br>串列傳輸速<br>率 | F <sub>CPU</sub> |      |      |
|------------------|------------------|------|------|
|                  | 2MHz             | 4MHz | 8MHz |
| 1200             | 98               | 30   | -    |
| 2400             | CC               | 98   | 30   |
| 4800             | E6               | CC   | 98   |
| 9600             | F3               | E6   | CC   |



## 14.5 多機通信

### 14.5.1 軟體位址識別

方式2和方式3具有適用於多機通訊功能。在這兩個方式下，接收的是9位元資料，第9位移入RB8中，之後是停止位。可以這樣設定UART：當接收到停止位，且RB8 = 1時，串列口中斷有效（請求標誌RXIF置位元）。此時置位SM2位，UART工作在多機通訊模式。

在多機通訊系統中，按如下所述來使用這一功能。當主機要發送一資料塊給幾個從機中的一個時，先發送一位址位元組，以定址目標從機。位址位元組與資料位元組可用第9資料位元來區別，位址位元組的第9位元為1，資料位元組的第9位元為0。

如果從機SM2為1，則不會回應資料位元組中斷。位址位元組可以使所有從機產生中斷，每一個從機都檢查所接收到的位址位元組，以判別本機是不是目標從機。被尋到的從機對SM2位執行清零操作，並準備接收即將到來的資料位元組。當接收完畢時，從機再一次將SM2置位。沒有被定址的從機，則保持SM2位為1，不回應資料位元組。

注：在方式1中，SM2用來檢測停止位是否有效，如果SM2 = 1，接收中斷不會回應直到接收到一個有效的停止位。

### 14.5.2 自動（硬體）位址識別

在方式2和方式3中，SM2置位元，UART運行狀態如下：接收到停止位，RB8的第9位元為1（位址位元組），且接收到的資料位元組符合UART的從機位址，UART產生一個中斷。從機將SM2清零，接收後續資料位元組。

第9位元為1表明該位元組是位址而非資料。當主機要發送一組資料給幾個從機中的一個時，必須先發送目標從機位址。所有從機等待接收位址位元組，為了確保僅在接收位址位元組時產生中斷，SM2位必須置位。自動位址識別的特點是只有位址匹配的從機才能產生中斷，硬體完成位址比較。

中斷產生後，位址匹配的從機清零SM2，繼續接收資料位元組。位址不匹配的從機不受影響，將繼續等待接收和它匹配的位址位元組。全部資訊接收完畢後，位址匹配的從機應該再次把SM2置位，忽略所有傳送的非位址位元組，直到接收到下一個位址位元組。

使用自動位址識別功能時，主機可以通過調用給定的從機位址選擇與一個或多個從機通信。主機使用廣播位址可以定址所有從機。有兩個特殊功能寄存器，從機位址（SADDR）和位址遮罩（SADEN）。從機位址是一個8位元的位元組，存於SADDR寄存器中。SADEN用於定義SADDR各位的有效與否，如果SADEN中某一位為0，則SADDR中相應位被忽略，如果SADEN中某一位置位，則SADDR中相應位將用於產生約定位址。這可以使用戶在不改變SADDR寄存器中的從機位址的情況下靈活地定址多個從機。

|       | 從機1      | 從機2      |
|-------|----------|----------|
| SADDR | 10100100 | 10100111 |
| SADEN | 11111010 | 11111001 |
| 約定地址  | 10100x0x | 10100xx1 |
| 廣播地址  | 1111111x | 11111111 |

從機1和從機2的約定位址最低位是不同的。從機1忽略了最低位，而從機2的最低位是1。因此只與從機1通訊時，主機必須發送最低位元為0的地址（10100000）。類似地，從機1的第1位為0，從機2的第1位被忽略。因此，只與從機2通訊時，主機必須發送第1位元為1的地址（10100011）。如果主機需要同時與兩從機通訊，則第0位元為1，第1位為0，第2位被兩從機都忽略，兩個不同的位址用於選定兩個從機（1010 0001和1010 0101）。



主機可以通過廣播位址與所有從機同時通訊。這個位址等於SADDR和SADEN的位或，結果中的0表示該位被忽略。多數情況下，廣播位址為0xFF，該位址可被所有從機應答。

系統重定後，SADDR和SADEN兩個寄存器初始化為0，這兩個結果設定了約定地址和廣播地址為xxxxxxx（所有位都被忽略）。這有效地去除了多從機通訊的特性，禁止了自動定址方式。這樣的UART將對任何位址都產生應答，相容了不支持自動位址識別的8051控制器。使用者可以按照上面提到的方法實現軟體位址識別的多機通訊。

## 14.6 幀出錯檢測

錯誤標誌位元被置位元後，只能通過軟體清零，儘管後續接收的幀沒有任何錯誤也不會自動清零。如果檢測到一個無效（低）停止位，那麼幀出錯位（FE位）置1。

# 15 串列外部設備介面SPI

此系列單片機還提供另一種高速串列通信介面SPI介面。SPI是一種全雙工、高速、同步的通信匯流排，有兩種操作模式：主模式和從模式。在主模式中支援高達4 Mbps的速率(CPU採用更高主頻時，則可更高。從模式時速度無法太快， $F_{sys}/8$ 以內較好)，還具有傳輸完成標誌和寫衝突標誌保護。

## 15.1 SPI 相關寄存器

### 15.1.1 SPI 控制寄存器 SPCTL

| 231h  | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| SPCTL | SSIG  | SPEN  | DORD  | MSTR  | CPOL  | CPHA  | SPR1  | SPR0  |
| R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR的值 | 0     | 0     | 0     | 0     | 0     | x     | 0     | 0     |

bit 7 **SSIG**：SS引腳忽略控制位

1 = MSTR確定器件為主機還是從機

0 = SS腳用於確定器件為主機還是從機，SS腳可作為I/O使用

bit 6 **SPEN**：SPI使能位

1 = SPI使能

0 = SPI被禁止，所有SPI引腳都作為I/O使用

bit 5 **DORD**：SPI輸出控制位

1 = 資料字的LSB（最低位）最先發送

0 = 資料字的MSB（最高位）最先發送

bit 4 **MSTR**：主/從模式選擇位元

1 = 主機

0 = 從機

bit 3 **CPOL**：SPI時鐘極性

1 = SPICLK空閒時為高電平。SPICLK的前時鐘沿為下降沿而後沿為上升沿

0 = SPICLK空閒時為低電平。SPICLK的前時鐘沿為上升沿而後沿為下降沿

bit 2 **CPHA**：SPI時鐘相位選擇

1 = 資料在SPICLK的前時鐘沿驅動，並在後時鐘沿採樣

0 = 資料在SS為低（SSIG=0）時被驅動，在SPICLK的後時鐘沿被改變，並在前時鐘沿被採樣。（SSIG=1時的操作未定義）

bit [1:0] **SPR[1:0]**：SPI時脈速率選擇控制位

11 =  $F_{cpu}/128$

10 =  $F_{cpu}/64$

01 =  $F_{cpu}/16$

00 =  $F_{cpu}/4$

### 15.1.2 SPI 狀態寄存器

| 054h  | Bit 7  | Bit 6  | Bit 5  | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0  |
|-------|--------|--------|--------|-------|-------|-------|-------|--------|
| PIR2  | PWM2IF | PWM1IF | PWM0IF | -     | RXIF  | TXIF  | SPIF  | CCP2IF |
| R/W   | R/W    | R/W    | R/W    | -     | R/W   | R/W   | R/W   | R/W    |
| POR的值 | 0      | 0      | 0      | -     | 0     | 0     | 0     | 0      |

bit 1 **SPIF**：SPI傳輸完成標誌

1 = 序列傳輸完成時，硬體置1，軟體清零，當SPI處於主模式且SSIG=0時，如果SS為輸入並被驅動為高電平，SPIF也將置位，表示“模式改變”

0 = 序列傳輸未完成。

| 230h   | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|-------|-------|-------|-------|-------|-------|-------|-------|
| SPSTAT | -     | -     | -     | -     | -     | RXOV  | MODF  | WCOL  |
| R/W    | -     | -     | -     | -     | -     | R/W   | R/W   | R/W   |
| POR的值  | -     | -     | -     | -     | -     | 0     | 0     | 0     |

bit 2 **RXOV**：接收溢出標誌

1 = 發生接收溢出，軟體清零

0 = 未發生接收溢出。

bit 1 **MODF**：模式改變錯誤標誌

1 = 發生模式改變錯誤，軟體清零

0 = 未發生模式改變錯誤。

bit 0 **WCOL**：SPI寫衝突標誌

1 = 在資料傳輸的過程中如果對SPI資料寄存器SPDAT執行寫操作，WCOL將置1，軟體清零

0 = 未發生寫衝突。

### 15.1.3 SPI 資料寄存器 SPDAT

| 232h  | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| SPDAT |       |       |       |       |       |       |       |       |
| R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| POR的值 | 0     | 0     | 0     | 0     | 0     | x     | 0     | 0     |

bit [7:0] **SPDAT[7:0]**：SPI 資料寄存器

## 15.2 SPI 信號描述

主輸出從輸入 (**MOSI**)：該信號連接主設備和一個從設備，資料通過 **MOSI** 從主設備串列傳送到從設備，主設備輸出，從設備輸入。

主輸入從輸出 (**MISO**)：該信號連接主設備和一個從設備。資料通過 **MISO** 從從設備串列傳入到主設備，從設備輸出，主設備輸入。若該設備為從設備且未被選時，從設備的 **MISO** 引腳處於高阻狀態。

串列時鐘 (**SCK**)：該信號用作控制 **MOSI** 和 **MISO** 線上輸入輸出資料的同步移動，每 8 個時鐘週期 **MOSI** 和 **MISO** 線上傳送一個位元組，如果從設備未被選中，**SCK** 信號將被此設備忽略。注意：只有主設備才能產生 **SCK** 信號。

從設備選擇引腳 ( $\overline{SS}$ )：每個從屬週邊設備由一個從選擇引腳 $\overline{SS}$ 選擇，當引腳信號為低電平時，表明該從設備被選中。主設備可以通過軟體控制連接於從設備 $\overline{SS}$ 引腳的埠電平選擇每個從設備，很明顯，

只有一個主設備可以驅動通訊網路。為了防止 MISO 匯流排衝突，同一時間只允許一個從設備與主設備通訊。在主設備模式中， $\overline{SS}$  引腳狀態關聯 SPI 狀態寄存器 SPSTAT 中 MODF 標誌位元以防止多個主設備驅動 MOSI 和 SCK。

下列情況， $\overline{SS}$  引腳可以作為普通埠或其它功能使用：

(1) 設備作為主設備，SPI 控制寄存器 SPCTL 寄存器的 SSIG 位置 1。這種配置僅僅存在於通訊網路中只有一個主設備的情況，因此，SPI 狀態寄存器 SPSTA 中 MODF 標誌位元不會被置 1。

(2) 設備配置為從設備，SPI 控制寄存器 SPCTL 的 CPHA 位和 SSIG 位置 1。這種配置情況存在於只有一個主設備一個從設備的通訊網路中，因此，設備總是被選中的，主設備也不需要控制從設備的  $\overline{SS}$  引腳選擇其作為通訊目標。

從設備的  $\overline{SS}$  引腳被使能時，其它主設備可通過使該引腳維持低電平，從而選中該從設備。為防止 MISO 匯流排衝突，原則上不允許兩個及以上的從設備被選中。

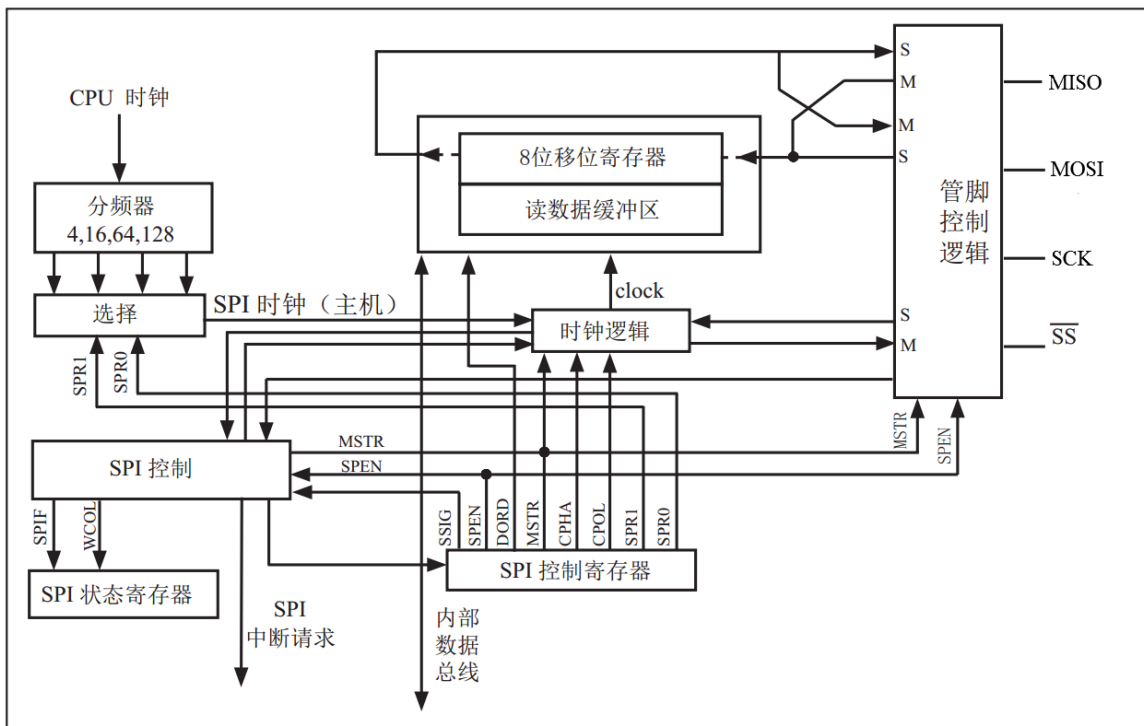
主設備的  $\overline{SS}$  引腳被使能時，若  $\overline{SS}$  被拉低將置模式錯誤標誌 MODF（可中斷），且 MSTR 位元也將被清零，從而使該設備強制切換成從設備。

當 MSTR = 0（從模式）及 CPHA = 0 時，SSIG 必須為 0，因為此時資料傳送需要  $\overline{SS}$  引腳配合，才能完成多資料傳送。

### 15.3 SPI 時脈速率

在主模式下，SPI 的速率有 4 級選擇，分別是內部時鐘的 4、16、64 或 128 分頻，可通過 SPCTL 寄存器的 SPR[1:0] 位進行選擇。

### 15.4 SPI 功能框圖

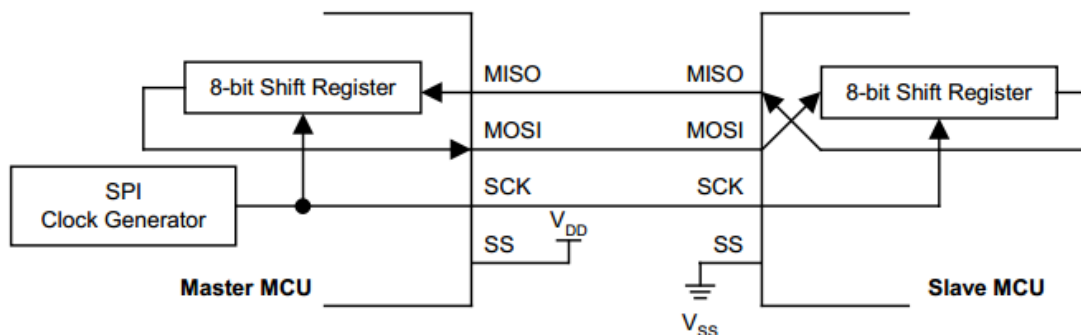


## 15.5 SPI 工作模式

SPI 可配置為主模式或從模式中的一種。SPI 模組的配置和初始化通過設置相關寄存器來完成。進一步設置相關寄存器即可完成資料傳送。

在 SPI 通訊期間，資料同步地被串列的移進移出，串列時鐘線（SCK）使兩條串列資料線（MOSI&MISO）上資料的移動和採樣保持同步。從設備選擇線（ $\overline{SS}$ ）可以獨立地選擇從屬設備；如果從設備沒有被選中，則不能參與 SPI 匯流排上的活動。

當 SPI 主設備通過 MOSI 線傳送資料到從設備時，從設備通過 MISO 線發送資料到主設備作為相應，從而實現在同一時鐘下資料發送與接收的同步全雙工傳輸。發送移位暫存器和接收寄存器使用相同的 SFR 地址，對 SPI 資料寄存器 SPDAT 進行寫操作將寫入發送移位暫存器，對 SPDAT 寄存器進行讀操作將獲得接收移位暫存器的資料。注：寫入的資料不會影響到需要讀出的資料。



### 主模式

#### (1) 模式啟動

SPI 主設備控制 SPI 匯流排上的所有資料傳送的啟動。一個 SPI 匯流排中只允許一個主設備可以啟動傳送。

#### (2) 發送

在 SPI 主模式下，寫一個位元組資料到 SPI 資料寄存器 SPDAT，資料將會寫入發送移位緩衝器。如果發送移位暫存器中已經存在一個資料或正在傳送一個資料，那麼主 SPI 將產生一個 WCOL 信號以表明寫入太快。但是發送移位暫存器中的資料不會受到影響，發送也不會中斷。

#### (3) 接收

當主設備通過 MOSI 線傳送資料到從設備時，同時對應的從設備也可以通過 MISO 線將其發送移位暫存器的資料傳送給主設備的接收移位暫存器，實現全雙工操作。故 SPIIF 標誌置 1 即表示資料發送完成也表示資料接收完成。本 SPI 模組接收為雙緩衝器，即資料可以在 SPIIF 置 1 後讀出，但必需在下一位元組資料接收完成前讀出，否則將置接收溢出標誌 RXOV，如果發生接收溢出，則後面的資料將不會被移入接收寄存器，接收溢出時，SPIIF 可正常置 1。

### 從模式

#### (1) 模式啟動

將 MSTR 置 0（若  $\overline{SS}$  被使能則必需拉低）時，設備處於從模式下運行，資料傳送過程中設備模式不能改變（ $\overline{SS}$  引腳必需維持低電平），否則資料傳送將失敗（SPIIF 不會被置 1）。

#### (2) 發送

SPI 從設備下不能啟動資料傳送，所以 SPI 從設備必需在主設備開始一次新的資料傳送之前將要傳送給主設備的資料寫入發送移位暫存器。若發送前未寫入資料到發送移位暫存器，從設備將傳送資料“0x00”給主設備。若寫入資料時發送移位暫存器已經存在資料（或發生在傳送過程中），那麼 SPI 從設

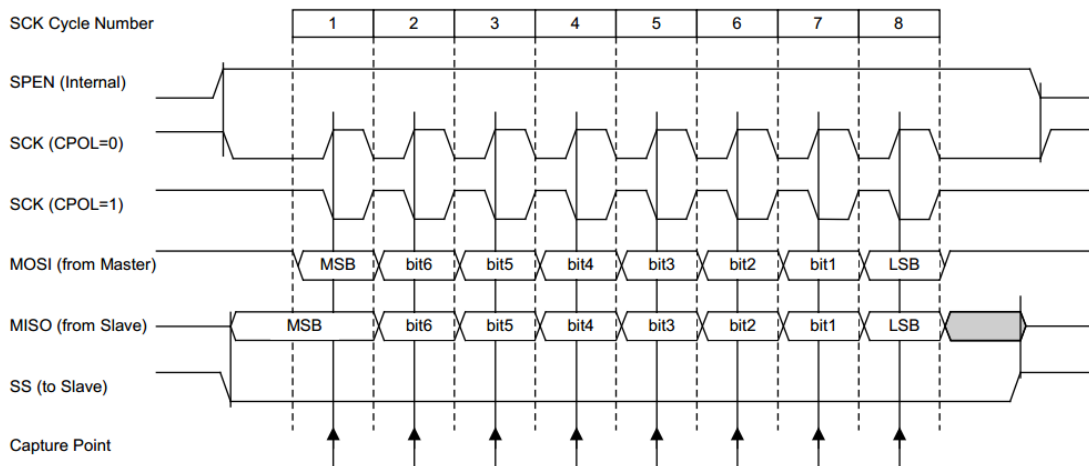
備的 WCOL 標誌位元將置 1，表示發生寫 SPDAT 衝突。但是移位暫存器的資料不受影響，傳送也不會被中斷，傳送完成 SPIF 將被置 1。

(3) 接收

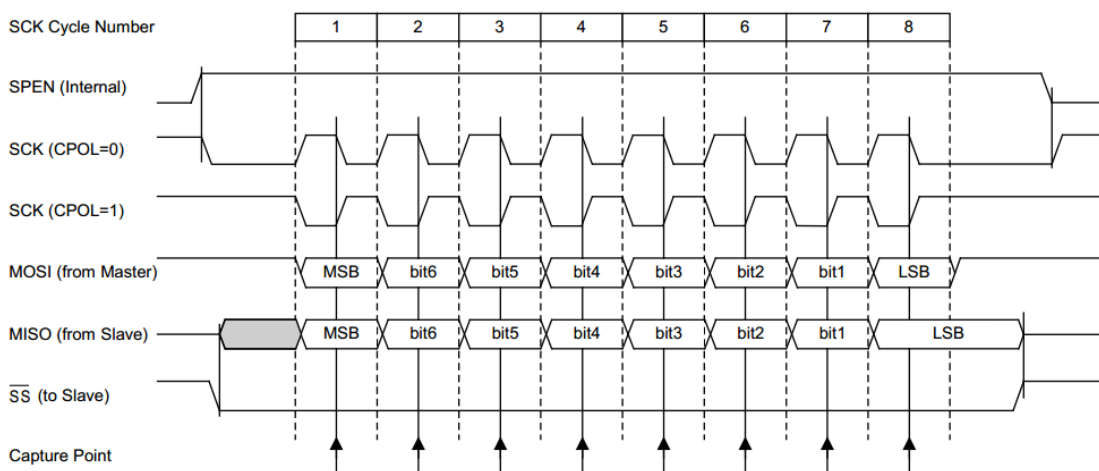
從模式下，按照主設備控制的 SCK 信號，資料通過 MOSI 引進移入，當計數器計數 SCK 邊緣數到 8 時，表示一個位元組資料接收完畢，SPIF 將置 1，資料可以通過此時讀取 SPDAT 寄存器獲得，但必需在下一資料接收完成前被讀出，否則將置接收溢出標誌 RXOV，如果發生接收溢出，則後面的資料將不會被移入接收寄存器，接收溢出時，SPIF 可正常置 1。

## 15.6 SPI 傳送形式

通過軟體設置寄存器的 CPOL 位和 CPHA 位，用戶可以選擇 SPI 時鐘極性和相位的四種組合方式。CPOL 位元定義時鐘的極性，即空閒時的電平狀態。CPHA 位元定義時鐘相位，即定義允許資料移位元採樣的時鐘邊沿。在通信的兩個主從設備中，時鐘極性相位設置應當保持一致。

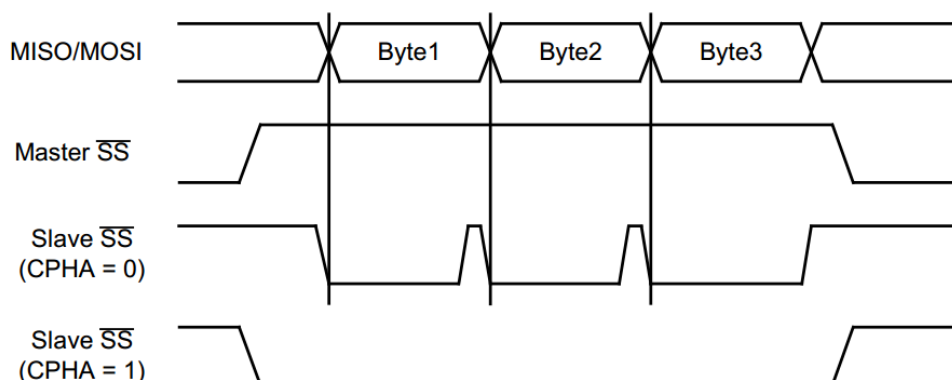


如果 CPHA = 0；數據在 SCK 的第一沿就被捕獲，所以從設備必需在 SCK 的第一個沿之前就準備好資料，因此，SS 引腳的下降沿從設備就開始資料。SS 引腳在每次傳送完一個位元組後必需拉高，在發送下一位元組之前重新又被拉低，故 CPHA = 0 時，SSIG 位無效，即 SS 腳被強制使能。



如果 CPHA = 1，主設備在 SCK 的第一個沿將資料輸出到 MOSI 線上，從設備把 SCK 的第一個沿作為開始發送信號。用戶必需在第一個 SCK 的前 2 個沿內完成對 SPDAT 完成寫操作。傳送過程中彼此模式不能改變，否則資料發送接收將失敗，模式被改變的寄存器資料（發送資料）及狀態（接收為空）不

變。這種資料傳送形式為單一主從設備間通信的首先形式。



## 15.7 SPI 出錯檢測

在資料未發送或發送期間繼續對 SPDAT 做寫入操作會引起寫衝突，WCOL 位會被置 1，但發送不會終止。需軟體清零

## 15.8 SPI 配置對照表

| SPEN | SSIG | $\overline{SS}$ | MSTR | 主或從模式    | MISO | MOSI | SCK | 備註                                                                  |
|------|------|-----------------|------|----------|------|------|-----|---------------------------------------------------------------------|
| 0    | x    | I/O             | x    | SPI功能禁止  | I/O  | I/O  | I/O | SPI禁止                                                               |
| 1    | 0    | 0               | 0    | 從機模式     | 輸出   | 輸入   | 輸入  | 選擇從機                                                                |
| 1    | 0    | 1               | 0    | 從機模式未被選中 | 高阻   | 輸入   | 輸入  | 未被選中。MISO為高阻，以避免匯流排衝突                                               |
| 1→0  | 0    | 0               | 1→0  | 關閉SPI    | 輸出   | 輸入   | 輸入  | SS配置為輸入，SSIG為0。如果SS被驅動為低電平。則被選擇作為從機。此時MSTR將清零，並置模式錯誤標誌MODF，可用於請求中斷。 |
| 1    | 0    | 1               | 1    | 主（空閒）    | 輸入   | 高阻   | 高阻  | 當主機空閒時MOSI和SCK為高阻態以避免匯流排衝突。用戶必須將SCK上拉或下拉（根據CPOL的取值）以避免SCK出現懸浮狀態。    |
|      |      |                 |      | 主（啟動）    |      | 輸出   | 輸出  | 作為主機啟動時，MOSI和SCK為推挽輸出。                                              |
| 1    | 1    | I/O             | 0    | 從        | 輸出   | 輸入   | 輸入  | CPHA不能為0                                                            |
| 1    | 1    | I/O             | 1    | 主        | 輸入   | 輸出   | 輸出  | -                                                                   |



# 16 指令表

| Field | 指令格式        | 描述                                                            | C | DC | Z | 週期   |
|-------|-------------|---------------------------------------------------------------|---|----|---|------|
| 移動    | MOVWF F     | $F \leftarrow W$                                              | - | -  | - | 1    |
|       | MOVF F, D   | $D \leftarrow F$ (D = 0 時為 W, D = 1 時為 F)                     | - | -  | √ | 1    |
|       | MOVLW k     | $W \leftarrow k$                                              | - | -  | - | 1    |
| 算術    | ADDWF F, D  | $D \leftarrow W + F$ (D = 0 時為 W, D = 1 時為 F)                 | √ | √  | √ | 1    |
|       | ADDLW k     | $W \leftarrow W + k$                                          | √ | √  | √ | 1    |
|       | SUBWF F, D  | $D \leftarrow F - W$ (D = 0 時為 W, D = 1 時為 F)                 | √ | √  | √ | 1    |
|       | SUBLW k     | $W \leftarrow k - W$ (D = 0 時為 W, D = 1 時為 F)                 | √ | √  | √ | 1    |
|       | DAW         | W 寄存器值進行 BCD 調整                                               | √ | √  | - | 1    |
|       | INCF F, D   | $D \leftarrow F + 1$ (D = 0 時為 W, D = 1 時為 F)                 | - | -  | √ | 1    |
|       | DECF F, D   | $D \leftarrow F - 1$ (D = 0 時為 W, D = 1 時為 F)                 | - | -  | √ | 1    |
| 邏輯    | ANDWF F, D  | $D \leftarrow W$ 與 $F$ (D = 0 時為 W, D = 1 時為 F)               | - | -  | √ | 1    |
|       | ANDLW k     | $W \leftarrow W$ 與 $k$                                        | - | -  | √ | 1    |
|       | IORWF F, D  | $D \leftarrow W$ 或 $F$ (D = 0 時為 W, D = 1 時為 F)               | - | -  | √ | 1    |
|       | IORLW k     | $W \leftarrow W$ 或 $k$                                        | - | -  | √ | 1    |
|       | XORWF F, D  | $D \leftarrow W$ 異或 $F$ (D = 0 時為 W, D = 1 時為 F)              | - | -  | √ | 1    |
|       | XORLW k     | $W \leftarrow W$ 異或 $k$                                       | - | -  | √ | 1    |
|       | COMF F, D   | $D \leftarrow F$ 取反 (D = 0 時為 W, D = 1 時為 F)                  | - | -  | √ | 1    |
| 處理    | SWAPF F, D  | $D[7:4, 3:0] \leftarrow F[3:0, 7:4]$ (D = 0 時為 W, D = 1 時為 F) | - | -  | - | 1    |
|       | RRF F, D    | $D \leftarrow F$ 帶進位右移 (D = 0 時為 W, D = 1 時為 F)               | √ | -  | - | 1    |
|       | RLF F, D    | $D \leftarrow F$ 帶進位左移 (D = 0 時為 W, D = 1 時為 F)               | √ | -  | - | 1    |
|       | CLRW        | $W \leftarrow 0$                                              | - | -  | √ | 1    |
|       | CLRF F      | $F \leftarrow 0$                                              | - | -  | √ | 1    |
|       | CLRWDT      | 清零看門狗計時器，影響 TO，PD 位                                           | - | -  | - | 1    |
|       | BCF F, d    | $F[d] \leftarrow 0$ ( $0 \leq d \leq 7$ )                     | - | -  | - | 1    |
|       | BSF F, d    | $F[d] \leftarrow 1$ ( $0 \leq d \leq 7$ )                     | - | -  | - | 1    |
| 分支    | INCFSZ F, D | $D \leftarrow F + 1$ (D = 0 時為 W, D = 1 時為 F)，如果 D = 0 則跳過下一句 | - | -  | - | 1(2) |
|       | DECFSZ F, D | $D \leftarrow F - 1$ (D = 0 時為 W, D = 1 時為 F)，如果 D = 0 則跳過下一句 | - | -  | - | 1(2) |
|       | BTFSC F, d  | 如果 $F[d] = 0$ ( $0 \leq d \leq 7$ ) 則跳過下一句                    | - | -  | - | 1(2) |
|       | BTFSS F, d  | 如果 $F[d] = 1$ ( $0 \leq d \leq 7$ ) 則跳過下一句                    | - | -  | - | 1(2) |
|       | GOTO k      | 無條件跳轉                                                         | - | -  | - | 2    |
|       | CALL k      | 調用副程式                                                         | - | -  | - | 2    |
| 其他    | RETURN      | 從副程式返回                                                        | - | -  | - | 2    |
|       | RETFIE      | 從中斷返回，並置位 GIE                                                 | - | -  | - | 2    |
|       | RETLW k     | $W \leftarrow k$ ，帶參數返回                                       | - | -  | - | 2    |
|       | NOP         | 空操作                                                           | - | -  | - | 1    |
|       | SLEEP       | 進入待機模式，影響 TO，PD 位                                             | - | -  | - | 1    |



# 17 開發工具

## 17.1 OTP 燒錄器 (PM18-4.0)

- HC-PM18-4.0：支援HC18系列MCU大批量的離線燒錄

注：

詳情請參考 HC-PM18 用戶手冊。如有更新請到網站下載最新資料。

<http://www.holychip.cn>

## 17.2 HC-IDE

Holychip 8位單片機的整合式開發環境HC-IDE包括編譯器。

- HC-IDE：HC-IDE\_2.10A\_235L(支持彙編)
- HC-IDE：HC-IDE V3.0.x.0(支持彙編/C語言)

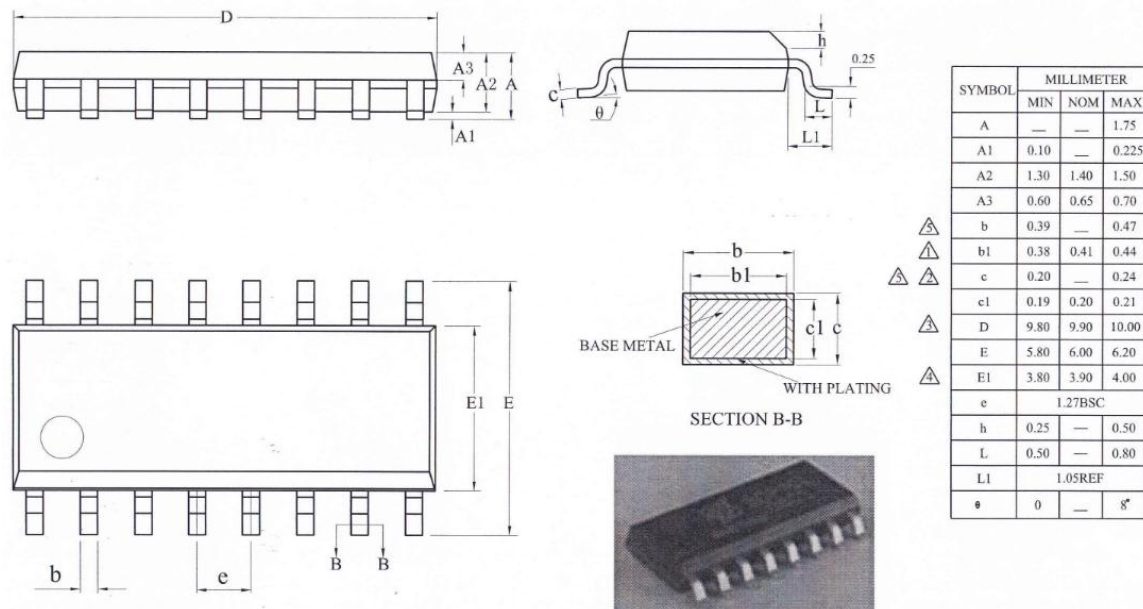
注：

1、詳情請參考 HC-IDE 用戶手冊。

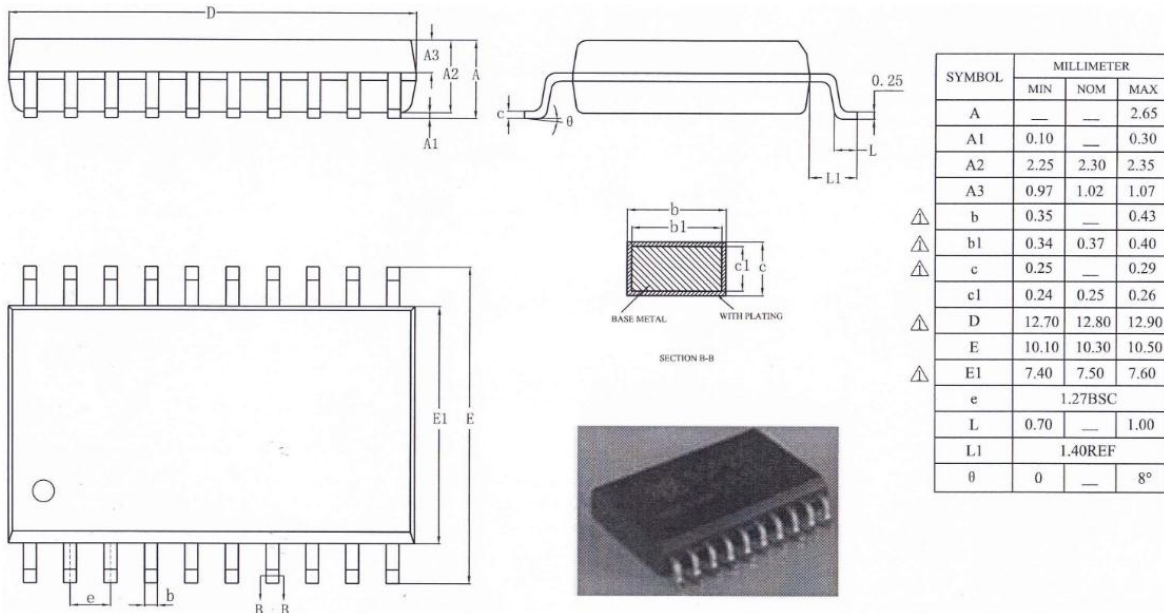
2、IDE 版本請關注芯聖官網：<http://www.holychip.cn/>

# 18 封裝尺寸

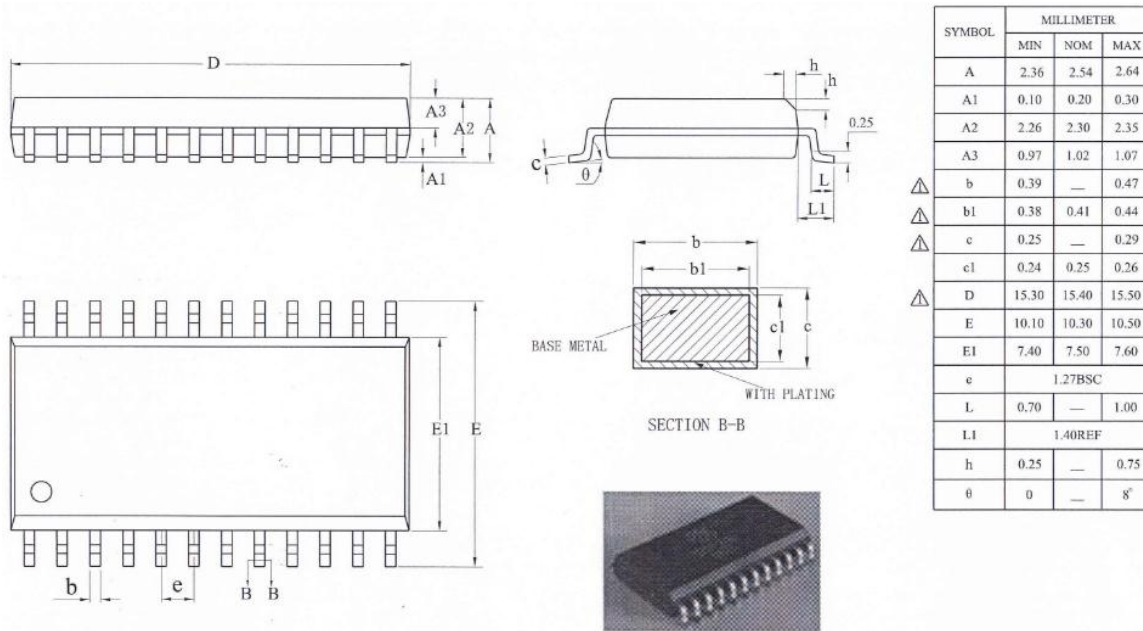
## 18.1 SOP16



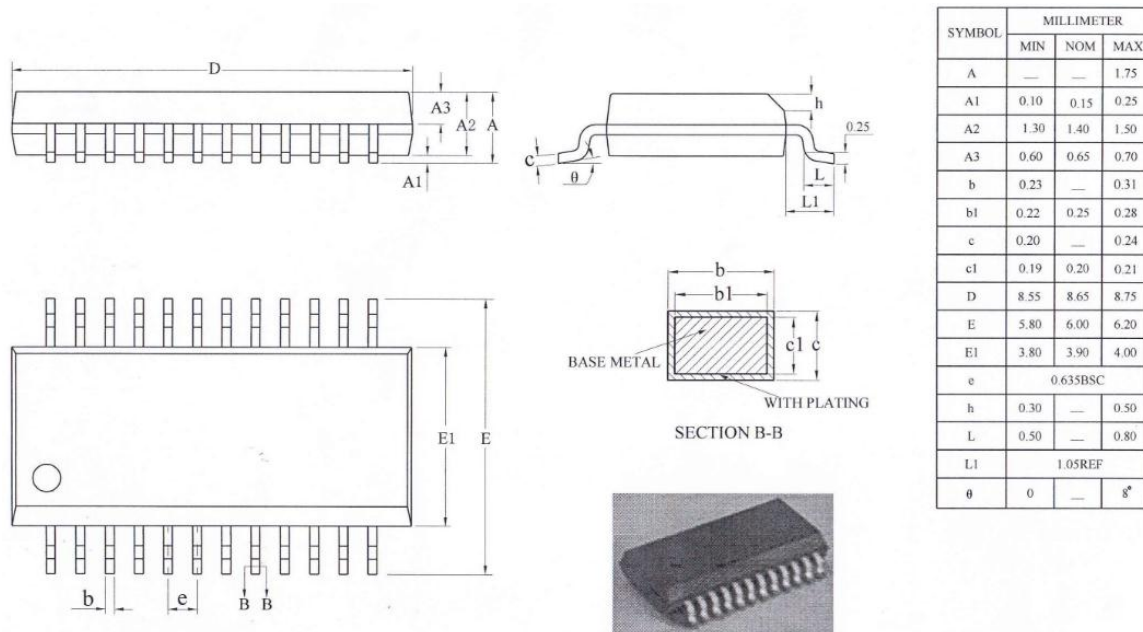
## 18.2 SOP20



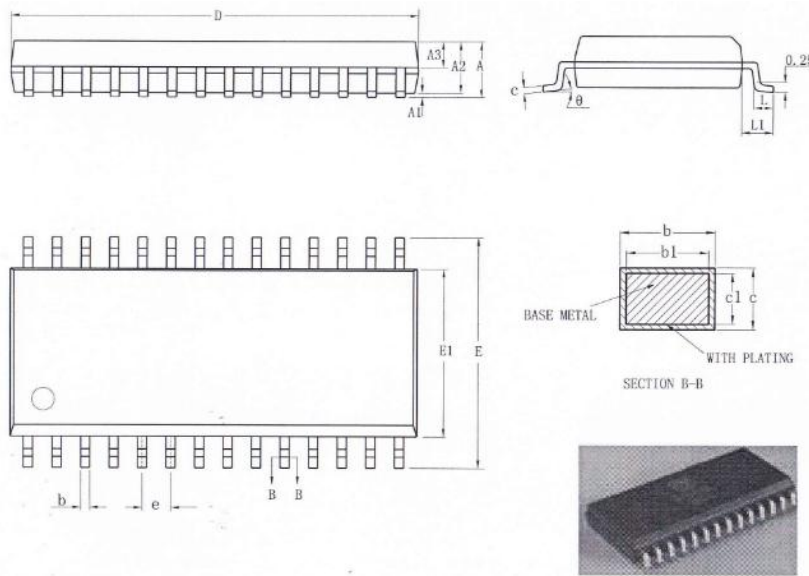
### 18.3 SOP24



### 18.4 SSOP24

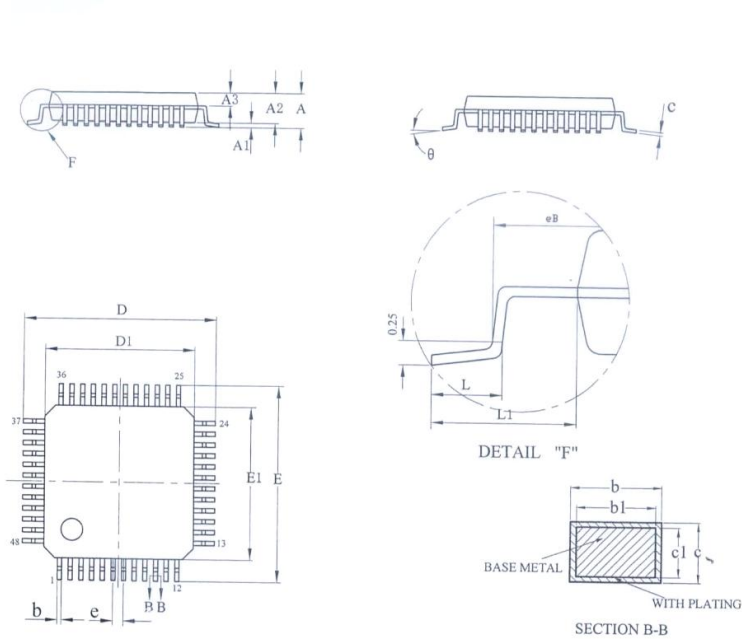


### 18.5 SOP28



| SYMBOL   | MILLIMETER |       |       |
|----------|------------|-------|-------|
|          | MIN        | NOM   | MAX   |
| A        | —          | —     | 2.65  |
| A1       | 0.10       | —     | 0.30  |
| A2       | 2.25       | 2.30  | 2.35  |
| A3       | 0.97       | 1.02  | 1.07  |
| b        | 0.39       | —     | 0.47  |
| b1       | 0.38       | 0.41  | 0.44  |
| c        | 0.25       | —     | 0.29  |
| c1       | 0.24       | 0.25  | 0.26  |
| D        | 17.90      | 18.00 | 18.10 |
| E        | 10.10      | 10.30 | 10.50 |
| E1       | 7.40       | 7.50  | 7.60  |
| e        | 1.27BSC    |       |       |
| L        | 0.70       | —     | 1.00  |
| L1       | 1.40REF    |       |       |
| $\theta$ | 0          | —     | 8°    |

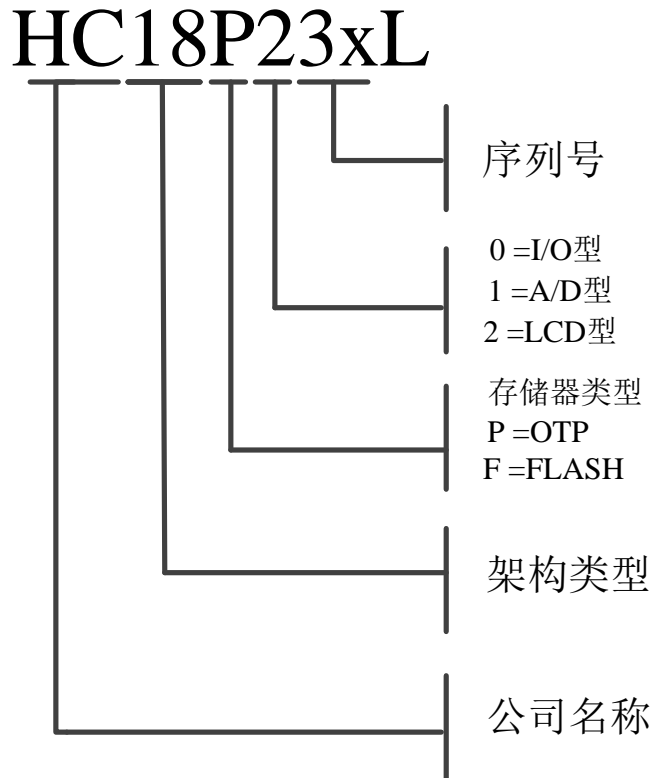
### 18.6 LQFP48



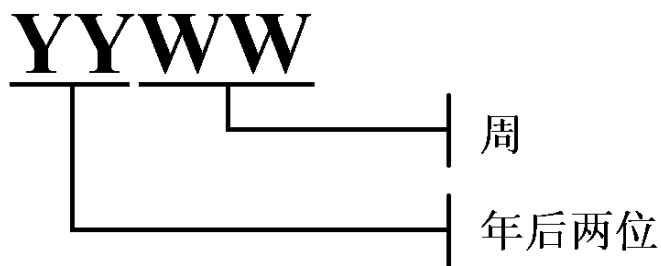
| SYMBOL   | MILLIMETER |      |      |
|----------|------------|------|------|
|          | MIN        | NOM  | MAX  |
| A        | —          | —    | 1.60 |
| A1       | 0.05       | —    | 0.15 |
| A2       | 1.35       | 1.40 | 1.45 |
| A3       | 0.59       | 0.64 | 0.69 |
| b        | 0.19       | —    | 0.27 |
| b1       | 0.18       | 0.20 | 0.23 |
| c        | 0.13       | —    | 0.18 |
| c1       | 0.12       | 0.13 | 0.14 |
| D        | 8.80       | 9.00 | 9.20 |
| D1       | 6.90       | 7.00 | 7.10 |
| E        | 8.80       | 9.00 | 9.20 |
| E1       | 6.90       | 7.00 | 7.10 |
| eB       | 8.10       | —    | 8.25 |
| e        | 0.50BSC    |      |      |
| L        | 0.45       | —    | 0.75 |
| L1       | 1.00BSC    |      |      |
| $\theta$ | 0          | —    | 7°   |

# 19 晶片正印命名規則

## 19.1 晶片型號說明（第一行）



## 19.2 日期碼規則（第二行）



## 19.3 生產批號（第三行）

例：**FA126026A**

## 20 修改記錄

| 版本      | 日期         | 描述                                        |
|---------|------------|-------------------------------------------|
| Ver1.00 | 2016-8     | 第一版                                       |
| Ver1.01 | 2016-10-12 | 第二版，增加軟體方式實現 1/2bias 說明 P7                |
| Ver1.02 | 2016-11-20 | 第三版，增加 HC18P232L_20Pin/HC18P233L_20Pin 腳位 |
| Ver1.03 | 2016-12-23 | 第四版，修改 LED 相關描述                           |
| V1.04   | 2017-5-25  | 新增 SOP24 及 SOP28 腳位描述、說明及封裝資訊             |
| V1.05   | 2017-12-14 | 修改格式,增加 HC-IDE V3.0.x.0                   |

HOLYCHIP公司保留對以下所有產品在可靠性、功能和設計方面的改進作進一步說明的權利。HOLYCHIP不承擔由本手冊所涉及的產品或電路的運用和使用所引起的任何責任，HOLYCHIP的產品不是專門設計來應用於外科植入、生命維持和任何HOLYCHIP產品產生的故障會對個體造成傷害甚至死亡的領域。如果將HOLYCHIP的產品用於上述領域，即使這些是由HOLYCHIP在產品設計和製造上的疏忽引起的，用戶應賠償所有費用、損失、合理的人身傷害或死亡所直接或間接所產生的律師費用，並且用戶保證HOLYCHIP及其雇員、子公司、分支機構和銷售商與上述事宜無關。

芯聖電子

2019年8月