

# HC88L051F4

數據手冊

20 引腳 8 位

低壓低功耗 FLASH 單片機

外設功能引腳全映射

# 目錄

<b>1</b>	<b>產品簡介.....</b>	<b>8</b>
1.1	功能特性 .....	8
1.2	應用注意 .....	10
1.3	系統框圖 .....	11
1.4	引腳配置 .....	12
1.5	引腳描述 .....	13
1.6	外設功能引腳全映射模組 PTM .....	20
<b>2</b>	<b>CPU .....</b>	<b>23</b>
2.1	CPU 特性 .....	23
2.2	CPU 相關寄存器.....	23
<b>3</b>	<b>記憶體.....</b>	<b>27</b>
3.1	程式記憶體 ( FLASH ) .....	27
3.2	資料記憶體 ( RAM ) .....	40
3.3	特殊功能寄存器 ( SFR ) .....	41
<b>4</b>	<b>系統時鐘.....</b>	<b>46</b>
4.1	系統時鐘特性 .....	46
4.2	內部時鐘 .....	48
4.3	外部時鐘 .....	48
4.4	系統時鐘相關寄存器 .....	49

<b>5</b>	<b>電源管理.....</b>	<b>60</b>
5.1	電源管理特性 .....	60
5.2	空閒模式 .....	60
5.3	掉電模式 .....	60
5.4	電源管理相關寄存器 .....	61
<b>6</b>	<b>復位 .....</b>	<b>63</b>
6.1	復位特性 .....	63
6.2	POR ( POWER-ON RESET ) 復位 .....	63
6.3	BOR ( BROWN-OUT RESET ) 復位 .....	63
6.4	外部 RST 復位 .....	64
6.5	外部埠低壓檢測復位 .....	64
6.6	軟體重定 .....	65
6.7	看門狗 ( WDT ) 復位 .....	65
6.8	堆疊溢位復位 .....	65
6.9	復位相關寄存器 .....	66
<b>7</b>	<b>通用及複用 I/O.....</b>	<b>71</b>
7.1	通用及複用 I/O 特性 .....	71
7.2	I/O 模式.....	71
7.3	I/O 功能框圖.....	72
7.4	I/O 埠相關寄存器 .....	72
7.5	外設功能引腳全映射控制 .....	81
<b>8</b>	<b>中斷 .....</b>	<b>84</b>

8.1	中斷特性 .....	84
8.2	中斷匯總 .....	84
8.3	中斷向量 .....	85
8.4	中斷優先順序 .....	85
8.5	中斷處理 .....	86
8.6	中斷回應時間 .....	87
8.7	外部中斷 .....	87
8.8	中斷相關寄存器 .....	89
<b>9</b>	<b>計時器/計數器.....</b>	<b>102</b>
9.1	計時器/計數器特性 .....	102
9.2	計時器/計數器 Tx(x = 0,1).....	102
9.3	計時器 3 .....	110
9.4	計時器/計數器 4 .....	115
9.5	計時器 5 .....	123
9.6	滴答計時器 TICK .....	135
<b>10</b>	<b>脈寬調製 PWM .....</b>	<b>138</b>
10.1	PWM 特性.....	138
10.2	PWM 輸出類型.....	139
10.3	PWM 輸出模式.....	141
10.4	PWM 相關寄存器.....	145
<b>11</b>	<b>單路 8 位 PWM .....</b>	<b>182</b>
11.1	PWM 特性.....	182

11.2	PWM 模組相關寄存器 .....	182
<b>12</b>	<b>看門狗計時器 WDT .....</b>	<b>186</b>
12.1	WDT 特性 .....	186
12.2	WDT 相關寄存器 .....	186
<b>13</b>	<b>通用非同步收發器 UART .....</b>	<b>190</b>
13.1	UART 特性 .....	190
13.2	工作方式 .....	190
13.3	串列傳輸速率發生器 .....	198
13.4	多機通信 .....	199
13.5	幀出錯檢測 .....	201
13.6	UART1 相關寄存器 .....	202
13.7	UART2 .....	208
<b>14</b>	<b>串列外部設備介面 SPI .....</b>	<b>215</b>
14.1	SPI 特性 .....	215
14.2	SPI 信號描述 .....	215
14.3	SPI 時脈速率 .....	216
14.4	SPI 功能框圖 .....	217
14.5	SPI 工作模式 .....	217
14.6	SPI 傳送形式 .....	219
14.7	SPI 出錯檢測 .....	221
14.8	SPI 中斷 .....	222
14.9	SPI 配置對照 .....	223
14.10	SPI 相關寄存器 .....	225

<b>15</b>	<b>IIC 匯流排.....</b>	<b>229</b>
15.1	IIC 特性.....	229
15.2	IIC 匯流排工作原理 .....	229
15.3	匯流排上資料的有效性 .....	230
15.4	匯流排上的信號 .....	231
15.5	匯流排上資料初始格式 .....	232
15.6	IIC 匯流排定址約定 .....	233
15.7	主機向從機讀寫 1 個位元組資料的過程.....	234
15.8	IIC 工作模式 .....	236
15.9	IIC 匯流排相關寄存器 .....	241
<b>16</b>	<b>模數轉換 ADC.....</b>	<b>246</b>
16.1	ADC 特性.....	246
16.2	ADC 省電喚醒.....	247
16.3	ADC 相關寄存器 .....	249
<b>17</b>	<b>低電壓檢測/比較器 .....</b>	<b>264</b>
17.1	低電壓檢測/比較器特性 .....	264
17.2	低電壓檢測/比較器相關寄存器 .....	265
<b>18</b>	<b>軟體 LCD .....</b>	<b>270</b>
18.1	LCD 特性 .....	270
18.2	LCD 幀.....	271
18.3	LCD 相關寄存器 .....	274
<b>19</b>	<b>迴圈冗餘校驗 CRC.....</b>	<b>276</b>

19.1	CRC 特性.....	276
19.2	CRC 相關寄存器 .....	277
20	代碼選項 .....	279
21	指令表 .....	280
22	電氣特性 .....	286
20.1	極限參數 .....	286
20.2	DC 特性.....	286
20.3	AC 特性.....	291
20.4	ADC 特性.....	292
20.5	FLASH 特性.....	293
20.6	BOR 檢測電壓特性 .....	293
20.7	LVD/PLVD 檢測電壓特性 .....	294
20.8	其他電氣特性 .....	294
23	開發工具 .....	295
23.1	HC-LINK 模擬工具.....	295
23.2	HC-PM51 燒錄工具.....	295
23.3	ISP 串口燒錄 .....	295
23.4	軟體下載 .....	296
24	封裝尺寸 .....	297
24.1	TSSOP20 .....	297
24.2	QFN20 .....	298
25	版本記錄 .....	299

# 1 產品簡介

HC88L051F4 是一顆採用高速低功耗 CMOS 工藝設計開發的增強型 8 位單片機，內部有 16K Bytes FLASH 程式記憶體，256 Bytes IRAM 和 1024 Bytes XRAM，最多 18 個雙向 I/O 口，1 個外設功能引腳全映射模組 PTM，6 個 16 位計時器/計數器，3 組 16 位帶死區控制互補 PWM，1 個 8 位 PWM，2 個 UART，1 個 SPI，1 個 IIC，16 個外部中斷，16+2 路 12 位元 ADC，四種系統工作模式（正常、低頻、掉電和空閒）和 19 個中斷源。

## 1.1 功能特性

- ◆ CPU
  - 增強型 1T 8051 內核
  - 多級低電壓重定 (BOR)
    - 4.2/3.9/3.6/3.0/2.6/2.4/2.0/1.8V
- ◆ ROM
  - 16K Bytes FLASH
  - 看門狗 (WDT) 復位
  - IAP 和 ICP 操作
  - 軟體重定
  - 靈活的代碼保護模式
  - 堆疊溢位復位
- ◆ RAM
  - 256 Bytes IRAM
  - 1024 Bytes XRAM
  - 外部管腳低電平復位
- ◆ 時鐘
  - 內部高精度 32MHz/24MHz RC
  - 內部 44KHz RC
  - 外部高頻晶振 4MHz - 24MHz
  - 外部低頻晶振 32.768KHz
  - CPU 最高運行頻率 24MHz
- ◆ I/O
  - 最多 18 個雙向 I/O 口
  - 多種模式可配：輸入、帶上拉輸入、帶下拉輸入、施密特輸入、模擬輸入、強推挽輸出、開漏輸出、開漏帶上拉輸出
  - 外設功能引腳全映射模組 PTM
- ◆ 多種重定方式
  - 上電復位 (POR)
  - 所有埠可配置 1/2 bias 軟體 LCD 驅動
- ◆ 中斷



- 19個中斷源
- 4級中斷優先順序
- 16個外部中斷
- ◆ 計時器/計數器
  - T0/T1相容標準8051，16位自動重載
  - T3可以工作在掉電模式
  - T4可以使用外部信號觸發定時
  - T5帶捕獲功能
  - TICK計時器，16位自動重載
- ◆ PWM
  - 最多3組16位帶死區控制互補PWM
    - 可配置為 6 路獨立輸出
    - 具有故障檢測功能
  - 1路8位單輸出PWM
- ◆ 通訊模組
  - 2 個 UART
  - 1 個 SPI
  - 1 個 IIC
- ◆ ADC 檢測電路
  - 支持16+2 ch 12位ADC檢測
- ADC參考電壓可選內部VREF、外部VREF、VDD
- 具有省電喚醒功能 ( 單通道 )
- 模擬看門狗功能
- ◆ 低電壓檢測模組
  - 比較器功能
  - VDD 多級電壓檢測，可中斷
    - 4.2/3.9/3.6/3.0/2.6/2.4/2.0/1.9V
- ◆ 迴圈冗餘校驗 ( CRC )
- ◆ 省電模式
  - 空閒模式
  - 掉電模式功耗小於 1uA
- ◆ 雙線模擬和下載
- ◆ 工作條件
  - 寬電壓 1.8V - 5.5V
  - 溫度範圍-40°C - +105°C
- ◆ 封裝類型
  - TSSOP20/QFN20

✓ 選型表

產品型號	ROM Bytes	RAM Bytes	MAX Freq	I/O	ADC	Timer	PWM	INT	WDT
HC88L051F4P7	16K	256+1024	24MHz	18	16+2	6	16Bit*3 組 8Bit*1	16	1
HC88L051F4U7	16K	256+1024	24MHz	18	16+2	6	16Bit*3 組 8Bit*1	16	1

產品型號	Voltag	TEMP	Package	Simulator	Programmer	Datasheet	DemoCode	DemoBoard
HC88L051F4P7	1.8~5.5V	-40~+105°C	TSSOP20	HC-LINK	HC-PM51	√	√	√
HC88L051F4U7	1.8~5.5V	-40~+105°C	QFN20	HC-LINK	HC-PM51	√	√	√

## 1.2 應用注意

1. 為保證系統的穩定性，必須在VDD和GND之間接一電容（容值須大於或等於0.1 $\mu$ F）。
2. P2.7引腳出廠時默認為重定管腳，埠模式為施密特輸入帶上拉，可以通過配置代碼選項將此口配置為普通IO引腳。
3. 如果需要進行FLASH IAP操作，請仔細閱讀3.1.4.1的注意事項。
4. 在進行IAP操作時，不回應任何中斷。
5. ADCEN 置 1 或切換轉換通道後，建議延時 20 $\mu$ s 後再啟動 ADC 轉換，如果外部輸入阻抗很大時，需要延長這個時間。
6. 當 ADC 的參考電壓為 VDD 時，ADC 轉換時鐘可以為 8MHz，一次轉換只需要 15 個 ADC\_CLK，這樣可以得到最快的 ADC 轉換速度。
7. 通過配置 LCD 的寄存器可以使多個 PORT 同時輸出 PWM。

## 1.3 系統框圖

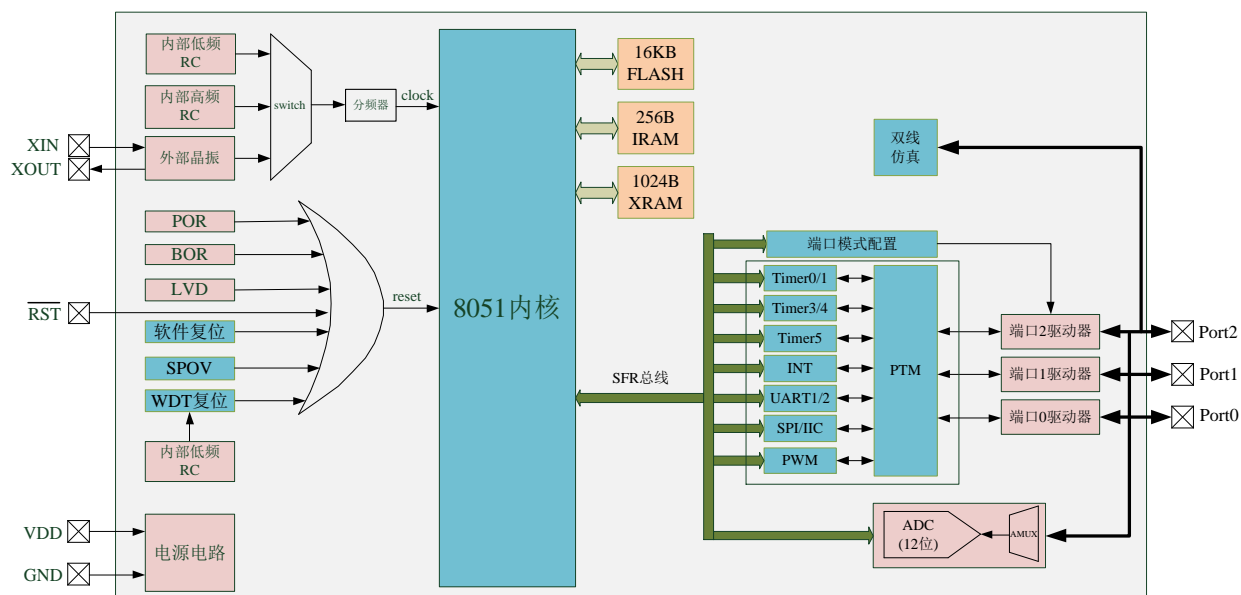


Figure 1-1 系統框圖

## 1.4 引腳配置

### 1.4.1 TSSOP20 引腳配置

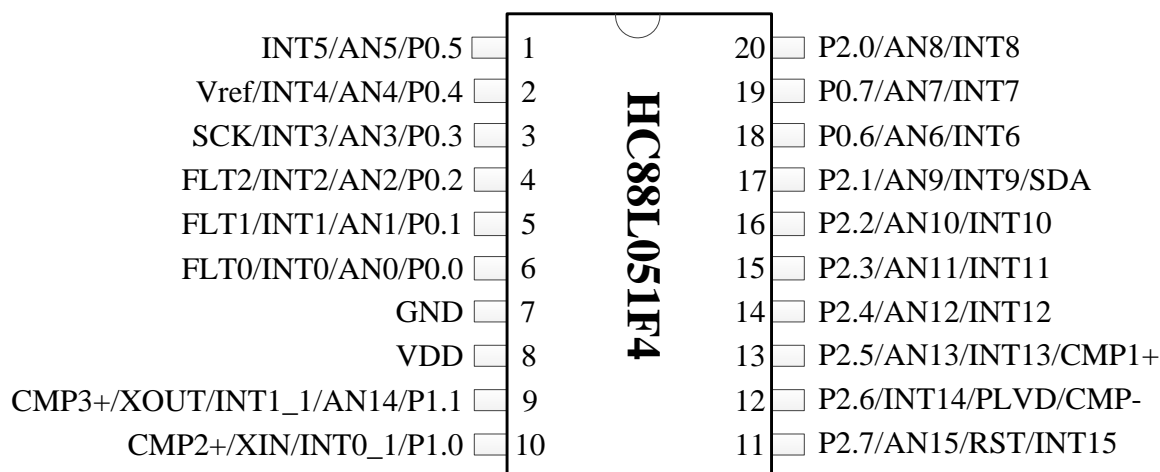


Figure 1-2 TSSOP20 引腳配置圖

### 1.4.2 QFN20 引腳配置

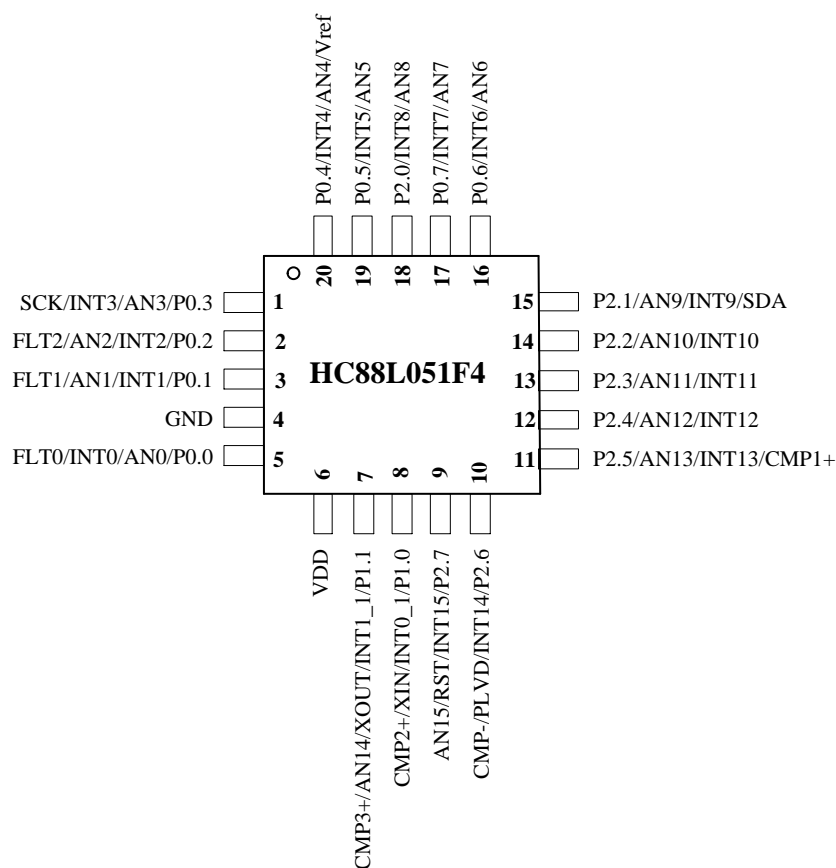


Figure 1-3 QFN20 引腳配置圖

## 1.5 引腳描述

### 1.5.1 TSSOP20 引腳描述

腳位	名稱	類型	說明
1	P0.5 AN5 INT5	I/O AN I	輸入/輸出口 ADC5 輸入口 外部中斷 5 輸入口
2	P0.4 AN4 Vref INT4	I/O AN AN I	輸入/輸出口 ADC4 輸入口 ADC 外部參考電壓輸入/輸出口 外部中斷 4 輸入口
3	P0.3 AN3 INT3 SCK	I/O AN I I	輸入/輸出口 ADC3 輸入口 外部中斷 3 輸入口 雙線模擬下載的時鐘輸入口
4	P0.2 AN2 INT2 FLT2	I/O AN I I	輸入/輸出口 ADC2 輸入口 外部中斷 2 輸入口 PWM2 故障檢測輸入引腳
5	P0.1 AN1 INT1 FLT1	I/O AN I I	輸入/輸出口 ADC1 輸入口

			外部中斷 1 輸入口 PWM1 故障檢測輸入引腳
6	P0.0 AN0 INT0 FLT0	I/O AN I I	輸入/輸出口 ADC0 輸入口 外部中斷 0 輸入口 PWM0 故障檢測輸入引腳
7	GND	P	電源地
8	VDD	P	電源輸入口
9	P1.1 AN14 XOUT INT1_1 CMP3+	I/O AN AN I AN	輸入/輸出口 ADC14 輸入口 外部晶振輸出口 可通過軟體配置為外部中斷 1 輸入口 比較器正端 3 輸入口
10	P1.0 XIN INT0_1 CMP2+	I/O AN I AN	輸入/輸出口 外部晶振輸入口 可通過軟體配置為外部中斷 0 輸入口 比較器正端 2 輸入口
11	P2.7 AN15 $\overline{\text{RST}}$ INT15	I/O AN I I	輸入/輸出口 ADC15 輸入口 外部復位輸入口 外部中斷 15 輸入口
12	P2.6 INT14	I/O I	輸入/輸出口

	PLVD CMP-	AN AN	外部中斷 14 輸入口  埠低電壓檢測埠  比較器負端輸入口
13	P2.5 AN13 INT13 CMP1+	I/O AN I AN	輸入/輸出口  ADC13 輸入口  外部中斷 13 輸入口  比較器正端 1 輸入口
14	P2.4 AN12 INT12	I/O AN I	輸入/輸出口  ADC12 輸入口  外部中斷 12 輸入口
15	P2.3 AN11 INT11	I/O AN I	輸入/輸出口  ADC11 輸入口  外部中斷 11 輸入口
16	P2.2 AN10 INT10	I/O AN I	輸入/輸出口  ADC10 輸入口  外部中斷 10 輸入口
17	P2.1 AN9 INT9 SDA	I/O AN I I	輸入/輸出口  ADC9 輸入口  外部中斷 9 輸入口  雙線模擬下載的資料登錄/輸出口
18	P0.6 AN6 INT6	I/O AN I	輸入/輸出口  ADC6 輸入口

			外部中斷 6 輸入口
19	P0.7 AN7 INT7	I/O AN I	輸入/輸出口 ADC7 輸入口 外部中斷 7 輸入口
20	P2.0 AN8 INT8	I/O AN I	輸入/輸出口 ADC8 輸入口 外部中斷 8 輸入口

注：I=輸入，O=輸出，I/O=輸入/輸出，P=電源，AN=模擬輸入輸出



## 1.5.2 QFN20 引腳描述

腳位	名稱	類型	說明
1	P0.3 AN3 INT3 SCK	I/O AN I I	輸入/輸出口  ADC3 輸入口  外部中斷 3 輸入口  雙線模擬下載的時鐘輸入
2	P0.2 AN2 INT2 FLT2	I/O AN I I	輸入/輸出口  ADC2 輸入口  外部中斷 2 輸入口  PWM2 故障檢測輸入引腳
3	P0.1 AN1 INT1 FLT1	I/O AN I I	輸入/輸出口  ADC1 輸入口  外部中斷 1 輸入口  PWM1 故障檢測輸入引腳
4	GND	P	電源地
5	P0.0 AN0 INT0 FLT0	I/O AN I I	輸入/輸出口  ADC0 輸入口  外部中斷 0 輸入口  PWM0 故障檢測輸入引腳
6	VDD	P	電源輸入口
7	P1.1 AN14 XOUT INT1_1	I/O AN AN I	輸入/輸出口  ADC14 輸入口

	CMP3+	AN	<p>外部晶振輸出口</p> <p>可通過軟體配置為外部中斷 1 輸入口</p> <p>比較器正端 3 輸入口</p>
8	P1.0 XIN INT0_1 CMP2+	I/O AN I AN	<p>輸入/輸出口</p> <p>外部晶振輸入口</p> <p>可通過軟體配置為外部中斷 0 輸入口</p> <p>比較器正端 2 輸入口</p>
9	P2.7 AN15 $\overline{\text{RST}}$ INT15	I/O AN I I	<p>輸入/輸出口</p> <p>ADC15 輸入口</p> <p>外部復位輸入口</p> <p>外部中斷 15 輸入口</p>
10	P2.6 INT14 PLVD CMP-	I/O I AN AN	<p>輸入/輸出口</p> <p>外部中斷 14 輸入口</p> <p>埠低電壓檢測埠</p> <p>比較器負端輸入口</p>
11	P2.5 AN13 INT13 CMP1+	I/O AN I AN	<p>輸入/輸出口</p> <p>ADC13 輸入口</p> <p>外部中斷 13 輸入口</p> <p>比較器正端 1 輸入口</p>
12	P2.4 AN12 INT12	I/O AN I	<p>輸入/輸出口</p> <p>ADC12 輸入口</p> <p>外部中斷 12 輸入口</p>

13	P2.3 AN11 INT11	I/O AN I	輸入/輸出口  ADC11 輸入口  外部中斷 11 輸入口
14	P2.2 AN10 INT10	I/O AN I	輸入/輸出口  ADC10 輸入口  外部中斷 10 輸入口
15	P2.1 AN9 INT9 SDA	I/O AN I I	輸入/輸出口  ADC9 輸入口  外部中斷 9 輸入口  雙線模擬下載的資料登錄/輸出口
16	P0.6 AN6 INT6	I/O AN I	輸入/輸出口  ADC6 輸入口  外部中斷 6 輸入口
17	P0.7 AN7 INT7	I/O AN I	輸入/輸出口  ADC7 輸入口  外部中斷 7 輸入口
18	P2.0 AN8 INT8	I/O AN I	輸入/輸出口  ADC8 輸入口  外部中斷 8 輸入口
19	P0.5 AN5 INT5	I/O AN I	輸入/輸出口  ADC5 輸入口  外部中斷 5 輸入口

20	P0.4 AN4 Vref INT4	I/O AN AN I	輸入/輸出口
			ADC4 輸入口
			ADC 外部參考電壓輸入/輸出口
			外部中斷 4 輸入口

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 外設功能引腳全映射模組 PTM

HC88L051F4 內置外設功能引腳全映射模組 ( PTM )，可通過使用者軟體操作將絕大多數的外設功能引腳配置在任意一個非電源口 ( VDD、GND ) 上。

### 1.6.1 PTM 模組特性

- 外設引腳為輸入功能 ( T0/1/3/5 外部輸入、RXD 等等 ) 特性時，系統將允許其多對一映射，即將多種輸入特性外設功能引腳分配到同一 IO 口上，此功能可使使用者系統得到更好優化。
- 外設引腳為輸出功能 ( T0/1/4 時鐘輸出、TXD 等等 ) 特性時，如果將多個輸出特性外設功能引腳分配到同一 IO 口，遵循固定的優先順序，只能有一個輸出有效。
- 軟體操作，使用靈活。使用者在應用系統設計時，不用考慮外設功能引腳佈局問題，從而降低開發成本。
- 用戶在開發過程中遇到如 PCB 上外設功能引腳佈局錯誤情況時，可使用該模組對外設功能引腳進行重新分配，從而縮短開發週期。
- 使用者應用系統升級如更換週邊器件或 MCU 時，改動項可減至最少，從而降低系統維護成本。

## 1.6.2 PTM 可全映射外設功能引腳

外設	名稱	類型	說明
計時器	T0	I/O	T0 的外部輸入或 T0 時鐘分頻輸出
	T1	I/O	T1 的外部輸入或 T1 時鐘分頻輸出
	T3	I	T3 的外部輸入
	T4	O	T4 的輸出
	T5	I	T5 的外部輸入
	CAP0	I	捕獲通道 0 的輸入
	CAP1	I	捕獲通道 1 的輸入
PWM	PWM0	O	PWM0 輸出口
	PWM01	O	PWM01 輸出口
	PWM1	O	PWM1 輸出口
	PWM11	O	PWM11 輸出口
	PWM2	O	PWM2 輸出口
	PWM21	O	PWM21 輸出口
	PWM3	O	PWM3 輸出口
CLK	CLKO	O	時鐘輸出口
UART	TXD	O	UART1 資料傳輸口
	RXD	I/O	UART1 接收腳
	TXD2	O	UART2 資料傳輸口
	RXD2	I	UART2 接收腳

ADC	ADCST	I	ADC 外部觸發啟動的輸入口
IIC	SCL	I/O	IIC 時鐘口
	SDA	I/O	IIC 數據口
SPI	MOSI	I/O	SPI 的資料口，主機的輸出和從機的輸入
	MISO	I/O	SPI 的資料口，主機的輸入和從機的輸出
	SCK	I/O	SPI 的時鐘口
	$\overline{\text{SS}}$	I	SPI 的片選口

## 2 CPU

### 2.1 CPU 特性

HC88L051F4 的 CPU 是一個增強型 1T 相容 8051 的內核，在同樣的系統時鐘下，較之傳統的 8051 晶片具有運行更快速，性能更優越的特性。

### 2.2 CPU 相關寄存器

#### 2.2.1 程式計數器 PC

程式計數器 PC 在物理上是獨立的，不屬於 SFR 之列。PC 字長 16 位元，是專門用來控制指令執行順序的寄存器。單片機上電或復位後，PC 的值為 0000H，這樣單片機從程式的零位址開始執行程式，假如第二重定向量使能，那上電或復位後，單片機將從第二復位向量指定的位址處開始執行程式。

#### 2.2.2 累加器 ACC

累加器 (ACC) 在指令系統中又記做 A，用於向 ALU 提供運算元和存放運算結果，它是 CPU 中最工作最頻繁的寄存器，大多數指令的執行都要通過累加器 ACC 進行。

#### 2.2.3 寄存器 B

寄存器 B 是專門為乘法和除法運算設置的寄存器，用於存放乘法和除法運算的運算元和運算結果，在不進行乘除運算時，可以作為通用寄存器使用。

#### 2.2.4 程式狀態字寄存器 PSW

此寄存器用來保存 ALU 運算結果的特徵和處理狀態，這些特徵和狀態可以作為控制程式轉移的條件，供程式判別和查詢，它的各位定義如下所示：

位編號	7	6	5	4	3	2	1	0
-----	---	---	---	---	---	---	---	---

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
復位值	0	0	0	0	0	0	0	0
位元符號	CY	AC	F0	RS[1:0]		OV	F1	P

位編號	位元符號	說明
7	CY	進位/借位標誌位元  0：算數運算中，無進位或借位  1：算數運算中，有進位或借位
6	AC	輔助進位/借位標誌位元  0：算數運算中，無輔助進位或借位  1：算數運算中，有輔助進位或借位
5	F0	使用者自訂標誌位元
4-3	RS[1:0]	工作寄存器組選擇位  00：第 0 組 ( 00H~07H )  01：第 1 組 ( 08H~0FH )  10：第 2 組 ( 10H~17H )  11：第 3 組 ( 18H~1FH )
2	OV	溢出標誌位元  0：無溢出  1：有溢出
1	F1	使用者自訂標誌位元



0	P	<p>奇偶標誌位元</p> <p>0 : ACC 寄存器中 1 的個數為 0 或偶數</p> <p>1 : ACC 寄存器中 1 的個數為奇數</p>
---	---	---

## 2.2.5 堆疊指標 SP

堆疊指標SP是一個8位元的專用寄存器，它指示出堆疊頂部在內部RAM中的位置。單片機復位後，SP值為07H，使得堆疊事實上由08H單元開始，考慮到08H~1FH單元分別屬於工作寄存器1~3，若在程式設計中要使用到這些區域，最好把SP的值改為較大的值。51單片機的堆疊是向上生成的，例如：SP=30H，CPU執行一條調用指令或回應中斷後，PC進棧，PCL保護到31H，PCH保護到32H，SP=32H。

## 2.2.6 數據指標 DPTR

數據指標DPTR是一個16位的專用寄存器，由兩個8位的寄存器DPH( 高8位 )和DPL( 低8位 )組成。此系列單片機有兩個16位元的數據指標DPTR0和DPTR1，其共用同一位址空間，可通過設置DPS ( INSCON.0 ) 位元來選擇具體使用的資料指標。

## 2.2.7 資料指標選擇寄存器 INSCON

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R/W	R	R	R	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-			IAPS	-			DPS
號								

位編號	位元符號	說明
7-5	-	保留位 ( 讀為 0，寫無效 )
4	IAPS	MOVC 操作區選擇位

		0：對程式區讀擦寫操作 1：對 OPTION 區讀操作
3-1	-	保留位 ( 讀為 0，寫無效 )
0	DPS	資料指標選擇位元 0：數據指標 DPTR0 1：數據指標 DPTR1

## 3 記憶體

### 3.1 程式記憶體 ( FLASH )

#### 3.1.1 FLASH 特性

- 在工作電壓範圍內都能進行擦除和程式設計操作
- 線上程式設計 ( ICP ) 操作支援寫入、讀取和擦除操作
- ICP 操作可設置 32 位元密碼進行保護
- 在應用程式設計 ( IAP ) 支援用戶自訂啟動代碼和類 EEPROM 區
- 靈活的代碼保護模式
- 擦寫次數至少 10 萬次
- 資料保存年限至少 10 年
- 128 位元組為 1 個磁區，8 個磁區為 1 頁，4 頁為一個保護單位

#### 3.1.2 FLASH 資料安全

FLASH 的操作可以分成兩種：第一種是通過工具 ( 模擬器、燒錄器 ) 對 FLASH 進行讀、擦、寫操作，這種方式被稱為在電路程式設計模式 ( ICP )；第二種是使用者程式碼對 FLASH 其他磁區進行讀、擦、寫操作，這種方式被稱為在應用程式設計模式 ( IAP )。

##### 3.1.2.1 ICP操作密碼保護

使用者可以通過上位機軟體對 ICP 操作進行密碼保護，密碼長度為 4 個位元組 ( 32 位元 )，一旦使用者設置了密碼，那麼只有輸入正確的密碼才能進行 ICP 操作，否則就不能對 FLASH 進行任何操作，這樣可以有效保護用戶的程式碼。

### 3.1.2.2 ICP讀擦寫FLASH保護

ICP 的讀保護以 4K 位元組為保護單位，當一個 4K 位元組空間的讀保護使能時，ICP 讀這個 4K 位元組空間，讀出來的資料為全 0，但是仍然可以通過 ICP 操作進行模擬。

ICP 的擦防寫也是以 4K 位元組為保護單位，當對應 4K 位元組的擦防寫使能時，ICP 將不能擦除和程式設計這個 4K 位元組空間，強寫也不允許。

若對應 4K 位元組空間讀保護被使能，但被允許擦除與寫入，則可先擦除後獲得該 4K 位元組空間的讀允許，直至復位或掉電。

ICP 的讀擦防寫通過上位機軟體來配置，詳細情況請參見 HC-LINK 用戶手冊。

### 3.1.2.3 IAP讀擦寫FLASH保護

IAP 通過 MOVC 指令來讀 FLASH，IAP 讀保護以 4K 位元組為單位，如果一個 4K 位元組空間設置了讀保護，其他 4K 位元組空間的 MOVC 指令讀這個 4K 位元組空間，讀出來的資料為全零，但這個 4K 位元組空間的 MOVC 指令可以讀取自身的資料。

IAP 擦寫 FLASH 的步驟詳見 FLASH IAP，IAP 的擦防寫以 4K 位元組為單位，IAP 擦寫之前需要先看相應磁區的擦防寫是否使能，沒有使能擦防寫才能進行 IAP 的擦寫。

若對應 4K 位元組空間讀保護被使能，但被允許擦除與寫入，則可先擦除後獲得該 4K 位元組空間的讀允許，直至復位或掉電。

IAP 的讀擦防寫通過上位機軟體來配置，詳細情況請參見 HC-LINK 用戶手冊。

## 3.1.3 OPTION

在 16K 的 ROM 之外有一個唯讀的 OPTION 區域，存放的內容包括：使用者定義的一些資料、使用者設置的密碼、晶片的一些配置、第二重定向量相關的內容。具體位址分配如下表。

地址	名稱	位址偏移量	名稱	地址	名稱	位址偏移量	名稱
0x0000	SN_DATA0	0x0020	FLASH_SC0	0x0031	ERST_ENB	0x0100	CHIP_ID0
0x0001	SN_DATA1	0x0021	FLASH_SC1	0x0038	WAIT_TS	0x0101	CHIP_ID1

0x0002	SN_DATA2	0x0022	FLASH_SC2	0x0039	BORVS	0x0102	CHIP_ID2
0x0003	SN_DATA3	0x0023	FLASH_SC3	0x003B	BORENB	0x0103	CHIP_ID3
0x0004	SN_DATA4	-	-	0x003E	RVCFG	0x0104	CHIP_ID4
0x0005	SN_DATA5	-	-	0x003F	nRVCFG	0x0105	CHIP_ID5
0x0006	SN_DATA6	-	-	-	-	0x0106	CHIP_ID6
0x0007	SN_DATA7	-	-	-	-	0x0107	CHIP_ID7
0x0008	ID_DATA0	-	-	-	-	-	-
0x0009	ID_DATA1	-	-	-	-	-	-
0x000A	ID_DATA2	-	-	-	-	-	-
0x000B	ID_DATA3	-	-	-	-	0x0128	rc32m_trim
0x000C	ID_DATA4	-	-	-	-	0x012C	rc24m_trim
0x000D	ID_DATA5	-	-	-	-	-	-
0x000E	ID_DATA6	-	-	-	-	-	-
0x000F	ID_DATA7	-	-	-	-	-	-

HC88L051F4 在出廠時都會固化一個 CHIP\_ID，一共 8 個位元組，一顆晶片一個 ID，不會重複，使用者可以在程式中通過 MOVC 來讀出，也可以通過工具讀出。

SN\_DATA 和 ID\_DATA 是使用者自訂資料，FLASH\_SC 為客戶密碼，通過工具軟體進行設置，如同設置代碼選項一樣，它們是可以被擦除和修改的，使用者也可以在程式中通過 MOVC 來讀出。

注意：1、使用者在進行讀 OPTION 操作前，需要將寄存器 INSCON[IAPS]位置 1。

2、首字母為“n”位元對應資料的反碼。

用C語言讀CHIP\_ID的程式如下：

//從FLASH裡面讀取任意長度的資料

```
void Flash_ReadArr(unsigned int fui_Address,unsigned char fuc_Length,unsigned char *fucp_SaveArr)
{
    while(fuc_Length--)
        *(fucp_SaveArr++)=((unsigned char code *)(fui_Address++));
}
```

//讀取CHIP\_ID的值，並保存到read\_chip\_id陣列裡

```
unsigned char read_chip_id[8];
```

```
INSCON |= 0x10;
```

```
Flash_ReadArr(0x0100,8,read_chip_id);    //CHIP_ID開始位址為0x0100
```

```
INSCON &=~ 0x10;
```

如果需要讀SN\_DATA或ID\_DATA，只需要修改讀位址即可。

### 3.1.3.1 外部復位使能ERST\_ENB

位編號	7	6	5	4	3	2	1	0
位元符號	-							ERST_ENB

位編號	位元符號	說明
7-1	-	保留位
0	ERST_ENB	復位引腳使能位 0：外部 $\overline{\text{RST}}$ 輸入 1：P2.7 為 GPIO

### 3.1.3.2 復位重讀OPTION後等待時間WAIT\_TS

位編號	7	6	5	4	3	2	1	0
位元符號	-	-	-	-	-	-	WAIT_TS	

位編號	位元符號	說明
7-2	-	保留位
1-0	WAIT_TS	復位重讀 option 後等待時間選擇位 00：8ms 01：4ms

		10 : 1ms  11 : 16ms
--	--	---------------------------

### 3.1.3.3 BOR檢測電壓選擇BORVS

位編號	7	6	5	4	3	2	1	0
位元符號	-	-	-	-	-	BORVS		
號								

位編號	位元符號	說明
7-3	-	保留位
2-0	BORVS	BOR 檢測電壓點選擇位元  000 : 1.8V  001 : 2.0V  010 : 2.4V  011 : 2.6V  100 : 3.0V  101 : 3.6V  110 : 3.9V  111 : 4.2V

### 3.1.3.4 BOR使能BORENB

位編號	7	6	5	4	3	2	1	0
-----	---	---	---	---	---	---	---	---

位元符號	-	BORENB
------	---	--------

位編號	位元符號	說明
7-1	-	保留位
0	BORENB	BOR 使能位 0 : BOR 使能 1 : BOR 禁止

### 3.1.3.5 第二重定向量配置RVCFG

位編號	7	6	5	4	3	2	1	0
位元符號	RVSEN	-	-	-	-	-	-	-

位編號	位元符號	說明
7	RVSEN	第二復位向量上電使能 0 : 禁止第二復位向量 1 : 使能第二復位向量
6-4	-	保留位
3-0	RVADR[3:0]	第二重定向量配置值 第二復位向量地址 = {RVADR[3:0], 0000000000B} 注：



		<ol style="list-style-type: none"> <li>RVADR[3:0]=0 時，表示第二復位向量位址和 0x0000H 重合。</li> <li>RVADR[3:0]只能配置 1000、1100、1110、1111 四個值，即第二復位向量的空間大小只能為 1K、2K、4K、8K。</li> </ol>
--	--	---

### 3.1.4 FLASH IAP 操作

HC88L051F4 的 FLASH 一共有 128 個磁區，128 個位元組為一個磁區， $8 \times 128 \text{ Bytes} = 1\text{K Bytes}$  為一頁， $1\text{K Bytes} \times 4 = 4\text{KBytes}$  為一塊。

IAP 寫之前必須先進行一次擦除操作，IAP 一次擦除一個磁區（128 位元組），IAP 擦除時的位址寄存器可以是磁區裡的任意位址。IAP 寫是單字節寫，每次寫一個位元組。

IAP 擦除一個磁區的時間是 5ms，在 CPU\_CLK 為 16MHz 時，IAP 寫一個位元組的時間是 23μs，在 CPU\_CLK 為 2MHz 時，IAP 寫一個位元組的時間是 37μs。

#### 3.1.4.1 IAP操作注意事項

HC88L051F4 的用戶程式碼可對 FLASH 進行讀、擦、寫操作，作為使用者更新代碼或存儲資料使用，為保證使用者對 FLASH 操作的安全性，使用過程中請注意：

- 在進行 FLASH 的 IAP 擦寫之前，需要配置擴展 SFR 裡 FREQ\_CLK 寄存器，指明目前 CPU 時鐘的頻率，FREQ\_CLK 寄存器配置的值等於 CPU 時鐘的頻率值，最小為 1MHz，假如目前 CPU 的運行頻率為 16MHz，那就配置寄存器 FREQ\_CLK=0x10。建議在 IAP 擦寫之前，將 CPU 時鐘頻率分頻為整數。當 CPU 時鐘頻率低於 1MHz 時，不能進行 FLASH 的 IAP 擦寫操作。
- 系統在進行 IAP 操作時，不回應任何中斷。
- 在 Option 中設置相關的 IAP 擦防寫，使能使用者程式所在磁區保護位元，可以有效保證程式區不會被改寫或誤擦除。
- IAP 擦寫操作前，建議關閉中斷（EA=0），確保在 IAP 操作期間不會被中斷影響，待 IAP 擦

寫操作完成後，再將中斷恢復。

- 5、在執行 IAP 操作時，不可避免的會遇到資料擦除結束後，尚未寫資料就掉電的情況，所以建議採用雙區域保存資料的方式，即使一個區域的資料被擦除，也可以保證另一個區域的資料被正常讀取。

### 3.1.4.2 IAP資料寄存器 IAP\_DATA

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	IAP_DATA[7:0]							

位編號	位元符號	說明
7-0	IAP_DATA[7:0]	IAP 資料寄存器

### 3.1.4.3 IAP位址寄存器 IAP\_ADDRL、IAP\_ADDRH

#### IAP\_ADDRL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	1	1	1	1	1	1	1	1
位元符號	IAP_ADDR[7:0]							

位編號	位元符號	說明
7-0	IAP_ADDR[7:0]	IAP 操作時的位址寄存器低八位

### IAP\_ADDRH

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	1	1	1	1	1	1
位元符號	-		IAP_ADDR[13:8]					

位編號	位元符號	說明
7-6	-	保留
5-0	IAP_ADDR[13:8]	IAP 操作時的位址寄存器高六位

注意：必須在解鎖後才能修改 IAP 位址寄存器，而且一次操作完成後，IAP 位址自動指向 0x3FFF。

### 3.1.4.4 IAP命令寄存器 IAP\_CMDH、IAP\_CMDL

#### IAP\_CMDH

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	IAP_CMDH[7:0]							

位編號	位元符號	說明
7-0	IAP_CMDH[7:0]	<p>操作命令模式選擇位元</p> <p>0xF0：解鎖(22 個 CPU 時鐘後自動鎖定，IAP_CMD[7:0] = 0x00)</p> <p>0xE1：觸發一次操作</p> <p>0xD2：磁區擦除</p> <p>0xB4：位元組程式設計</p> <p>0x87：軟體重定，重定位址為 0000H，不重讀代碼選項</p> <p>0x78：軟體重定，重定位址為 0000H，重讀代碼選項</p> <p>其它值：鎖定</p>

#### IAP\_CMDL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	1	1	1	1	1	1	1	1
位元符號	IAP_CMDL[7:0]							

位編號	位元符號	說明
7-0	IAP_CMDL[7:0]	<p>IAP_CMDH[7:0]反碼</p> <p>注：寫入 IAP_CMDL[7:0]資料必須為之前寫入 IAP_CMDH[7:0]的反碼，否則將鎖定相關操作，即相關操作會失敗。</p>

操作示例：

### 1、程式空間磁區擦除

IAP\_CMDH = 0xF0;

IAP\_CMDL = 0x0F;

IAP\_ADDRH = 0x80;

IAP\_ADDRH = 0x00; //選擇第 1 磁區被擦除，一個磁區為 128 位元組

IAP\_CMDH = 0xD2; //選擇操作方式，磁區擦除

IAP\_CMDL = 0x2D;

IAP\_CMDH = 0xE1; //觸發

IAP\_CMDL = 0x1E; //觸發後 IAP\_ADDRH 指向 0xFF，IAP\_ADDRH 指向 0x3F，同時自動鎖

定

### 2、程式空間位元組程式設計

IAP\_DATA = 0x02; //待程式設計資料，寫入資料寄存器必須放在解鎖之前

IAP\_CMDH = 0xF0;

IAP\_CMDL = 0x0F;

IAP\_ADDRH = 0x00;

IAP\_ADDRH = 0x00;

IAP\_CMDH = 0xB4; //選擇操作方式，位元組程式設計

IAP\_CMDL = 0x4B;

IAP\_CMDH = 0xE1; //觸發

IAP\_CMDL = 0x1E; //觸發後 IAP\_ADDRH 指向 0xFF，IAP\_ADDRH 指向 0x3F，IAP\_DATA

指向 0x00，同時自動鎖定

注意：解鎖之後，寫位址、選擇操作方式、觸發這三個步驟之間不能插入任何指令，必須連續操作。

### 3、軟體重定(不重讀代碼選項)

```
IAP_CMDH = 0xF0;  
IAP_CMDL = 0x0F;  
IAP_CMDH = 0x87;  
IAP_CMDL = 0x78;
```

### 4、軟體重定(重讀代碼選項)

```
IAP_CMDH = 0xF0;  
IAP_CMDL = 0x0F;  
IAP_CMDH = 0x78;  
IAP_CMDL = 0x87;
```

## 3.1.5 FLASH ICP 操作

### 3.1.5.1 雙線方式

用戶可以通過 HC-LINK 模擬器的雙線方式對 MCU 進行模擬和程式設計，當 MCU 已經焊在用戶板上後，如果用戶採用上電重定的方式，只需要連接四根線 ( VDD、GND、SDA、SCK )，使用者系統必須斷電，由模擬器提供電源。當使用者系統不希望掉電時，可以採用五根線進入程式設計模式，多了一個重定引腳，模擬器更詳細的使用說明請參見 HC-LINK 用戶手冊。

另外，因為程式設計信號非常敏感，使用者需要用 3 個跳線將程式設計引腳 ( VDD、SDA、SCK ) 從應用電路中分離出來，如下圖所示。如果使用外部重定引腳進入，也需要將外部重定引腳進行跳線分離。

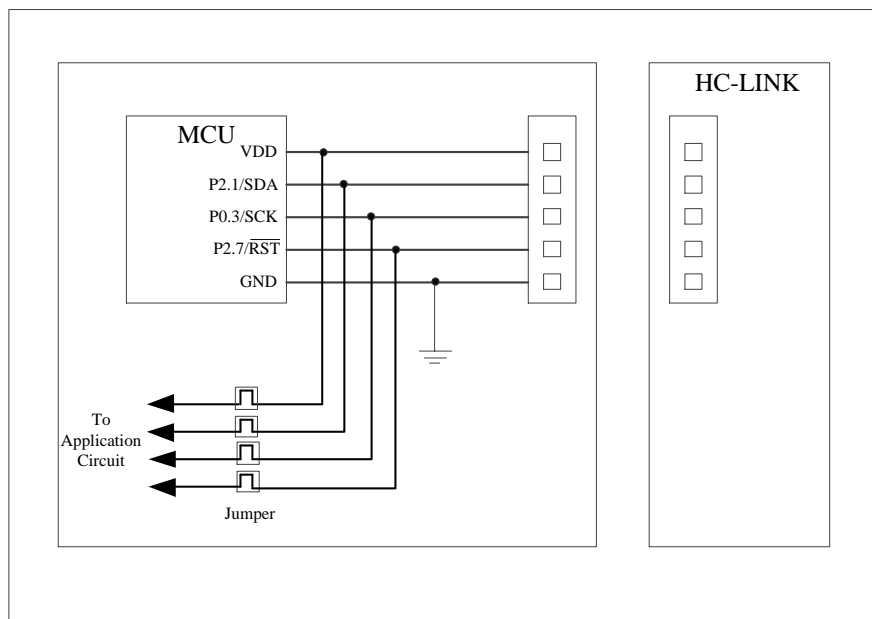


Figure 3-1 HC-LINK 程式設計硬體連接

當採用 ICP 模式進行操作時，建議按照如下步驟進行操作：

- 1、在開始程式設計前斷開跳線（Jumper），從應用電路中分離程式設計引腳。
- 2、將晶片程式設計引腳連接至 Flash 程式設計器介面，開始程式設計。
- 3、程式設計結束後斷開 Flash 程式設計器介面，連接跳線恢復應用電路。

### 3.1.6 第二復位向量操作

如果使用者在代碼選項中配置了第二重定向量使能和第二重定向量位址，那麼晶片上電重定後，PC 會首先指向第二向量位址，開始執行使用者的引導程式，使用者引導程式的最後需要放置一條不重讀代碼選項的軟體重定程式，那使用者就會復位到 0x0000H 處，開始執行使用者應用程式。

此功能也可以用來實現 ISP 功能，用戶自己編寫 ISP 引導程式，然後將 ISP 引導程式下載到第二重定向量位址開始的 FLASH 中，並使能第二復位向量操作。這樣用戶就可以通過自己編寫的 ISP 引導程式完成使用者應用程式的更新。

## 3.2 資料記憶體 ( RAM )

HC88L051F4 為用戶提供了 256 Bytes 內部 RAM 和 1024 Bytes 內部擴展 XRAM 來作為資料記憶體。下圖為資料記憶體空間分配。



Figure 3-2 資料記憶體示意圖

內部 RAM 的高 128 Bytes ( 0x80 ~ 0xFF ) 必須採用寄存器間接定址方式。

內部擴展 RAM( XRAM )的位址範圍是 0x0000 ~ 0x03FF, 訪問內部擴展 RAM 的方法和傳統 8051 單片機訪問外部擴展 RAM 的方法相同, 但是不影響 I/O 口。在組合語言中, 內部擴展 RAM 通過 MOVX 指令訪問, 即 MOVX @DPTP 或者 MOVX @Ri。



### 3.3 特殊功能寄存器 ( SFR )

#### 3.3.1 特殊功能寄存器列表

##### 3.3.1.1 直接定址讀寫SFR

	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F
F8	RSTFR	IAP_ADDRL	IAP_ADDRH	IAP_DATA	IAP_CMDL	IAP_CMDH	-	-
F0	B	PWM2EN	PWM2PL	PWM2PH	PWM2DL	PWM2DH	PWM2DTL	PWM2DTH
E8	-	PWM1EN	PWM1PL	PWM1PH	PWM1DL	PWM1DH	PWM1DTL	PWM1DTH
E0	ACC	PWM0EN	PWM0PL	PWM0PH	PWM0DL	PWM0DH	PWM0DTL	PWM0DTH
D8	-	-	PWM0C	PWM1C	PWM2C	PWM3C	PWM3P	PWM3D
D0	PSW	LCDCON	-	-	-	-	-	-
C8	-	T3CON	TL3	TH3	T4CON	TL4	TH4	-
C0	-	T5CON	TL5	TH5	RCAP5L	RCAP5H	-	-
B8	IE1	IP2	IP3	LVDC	LVDCMP	WDTC	CRCL	CRCH
B0	-	IP4	-	-	ADCC0	ADCC1	ADCRL	ADCRH
A8	IE	IP0	IP1	SPDAT	SPCTL	SPSTAT	IICDAT	IICADR
A0	P2	-	INSCON	-	-	-	IICCON	IICSTA
98	SCON	SBUF	SADDR	SADEN	-	-	SCON2	-
90	P1	TICKCON	TICKL	TICKH	-	-	PINTF0	PINTF1
88	TCON	TMOD	TL0	TL1	TH0	TH1	CLKSWR	CLKCON
80	P0	SP	DPL	DPH	-	-	-	PCON

##### 3.3.1.2 外部擴展XSFR

擴展 XSFR 採用和 XRAM 同樣的訪問方式，使用 MOVX A, @DPTR 和 MOVX @DPTR, A 來進行讀寫。

比如寫一個位址為 0xFE88 的 XSFR，操作如下：

```
MOV A, #wdata
MOV DPTR, #0xFE88
MOVBX @DPTR, A
```

讀位址為0xFE89的XSFR，操作如下：

```
MOV DPTR, #0xFE89
MOVBX A, @DPTR
```

使用C語言來程式設計時，只需要`#define ALLOCATE_EXTERN`，並且`#include "HC88L051F4.h"`，就可以像操作直接定址寄存器一樣，直接賦值XSFR，比如：

```
ADCC2 = 0x4D;
```

### 擴展XSFR ( 基底位址0xFE80 )

偏移位址	XSFR 名稱	偏移位址	XSFR 名稱	偏移位址	XSFR 名稱	偏移位址	XSFR 名稱
0x0000	TCON1	0x0010	-	0x0020	WDTCCR	0x0030	PITS0
0x0001	-	0x0011	CLKDIV	0x0021	-	0x0031	PITS1
0x0002	-	0x0012	FREQ_CLK	0x0022	CRCC	0x0032	PITS2
0x0003	-	0x0013	CLKOUT	0x0023	-	0x0033	PITS3
0x0004	T4CON1	0x0014	XTALCFG	0x0024	BORC	0x0034	-
0x0005	T5CON1	0x0015	SPOV_RSTEN	0x0025	BORDBC	0x0035	-
0x0006	T5CON2	0x0016	LXTALSPD	0x0026	-	0x0036	-
0x0007	-	0x0017	-	0x0027	LVDDBC	0x0037	-
0x0008	S2CON	0x0018	ADCWC	0x0028	-	0x0038	PINTE0
0x0009	S2CON2	0x0019	-	0x0029	-	0x0039	PINTE1
0x000A	S2BUF	0x001A	ADCC3	0x002A	RSTDBC	0x003A	-
0x000B	BRTSEL	0x001B	ADCC2	0x002B	AWDCON	0x003B	-
0x000C	-	0x001C	ADCPLYH	0x002C	ADCHTRH	0x003C	INT01_PINS
0x000D	-	0x001D	ADCPLYL	0x002D	ADCHTRL	0x003D	TRMEN
0x000E	-	0x001E	ADCCONTV	0x002E	ADCLTRH	0x003E	TRMV
0x000F	-	0x001F	ADCGAPV	0x002F	ADCLTRL	0x003F	-

### 擴展XSFR ( 基底位址0xFEC0 )

偏移位址	XSFR 名稱	偏移位址	XSFR 名稱	偏移位址	XSFR 名稱	偏移位址	XSFR 名稱
0x0000	SCRH0	0x0010	CAPH0	0x0020	PWM0PHASEH	0x0030	PWM0CMPH
0x0001	SCRL0	0x0011	CAPL0	0x0021	PWM0PHASEL	0x0031	PWM0CMPH
0x0002	SCRH1	0x0012	CAPH1	0x0022	PWM1PHASEH	0x0032	-
0x0003	SCRL1	0x0013	CAPL1	0x0023	PWM1PHASEL	0x0033	-
0x0004	SCRH2	0x0014	-	0x0024	PWM2PHASEH	0x0034	-
0x0005	SCRL2	0x0015	-	0x0025	PWM2PHASEL	0x0035	-
0x0006	SCRH3	0x0016	-	0x0026	-	0x0036	-
0x0007	SCRL3	0x0017	-	0x0027	-	0x0037	-
0x0008	SCRH4	0x0018	CAPCON0	0x0028	PWM0INTDIV	0x0038	PWMENA
0x0009	SCRL4	0x0019	CAPCON1	0x0029	PWM1INTDIV	0x0039	PWMCON0
0x000A	SCRH5	0x001A	-	0x002A	PWM2INTDIV	0x003A	PWMCON1
0x000B	SCRL5	0x001B	-	0x002B	-	0x003B	PWMCON2
0x000C	SCRH6	0x001C	-	0x002C	-	0x003C	-
0x000D	SCRL6	0x001D	-	0x002D	-	0x003D	-
0x000E	SCRH7	0x001E	-	0x002E	PWM3CLKS	0x003E	-
0x000F	SCRL7	0x001F	-	0x002F	-	0x003F	-

### 擴展XSFR ( 基底位址0xFF00 )

偏移位址	XSFR 名稱	偏移位址	XSFR 名稱	偏移位址	XSFR 名稱	偏移位址	XSFR 名稱
0x0000	P0M0	0x0010	P2M0	0x0020	-	0x0030	-
0x0001	P0M1	0x0011	P2M1	0x0021	-	0x0031	-
0x0002	P0M2	0x0012	P2M2	0x0022	-	0x0032	-
0x0003	P0M3	0x0013	P2M3	0x0023	-	0x0033	-
0x0004	-	0x0014	-	0x0024	-	0x0034	-
0x0005	P0LPU	0x0015	-	0x0025	-	0x0035	-
0x0006	-	0x0016	-	0x0026	-	0x0036	-
0x0007	-	0x0017	-	0x0027	-	0x0037	-
0x0008	P1M0	0x0018		0x0028	-	0x0038	-
0x0009	-	0x0019		0x0029	-	0x0039	-
0x000A	-	0x001A		0x002A	-	0x003A	-
0x000B	-	0x001B	-	0x002B	-	0x003B	-
0x000C	-	0x001C	-	0x002C	-	0x003C	-
0x000D	-	0x001D	-	0x002D	-	0x003D	-
0x000E	-	0x001E	-	0x002E	-	0x003E	-
0x000F	-	0x001F	-	0x002F	-	0x003F	-

### 擴展XSFR ( 基底位址0xFF40 )

偏移位址	XSFR 名稱	偏移位址	XSFR 名稱	偏移位址	XSFR 名稱	偏移位址	XSFR 名稱
0x0000	P00DBC	0x0010	P0OUT	0x0020	COMP0EN	0x0030	-
0x0001	P01DBC	0x0011	P1OUT	0x0021	COMP1EN	0x0031	-
0x0002	P02DBC	0x0012	P2OUT	0x0022	COMP2EN	0x0032	-
0x0003	-	0x0013	-	0x0023	-	0x0033	-
0x0004	-	0x0014	-	0x0024	-	0x0034	-
0x0005	-	0x0015	-	0x0025	-	0x0035	-
0x0006	-	0x0016	-	0x0026	-	0x0036	-
0x0007	-	0x0017	-	0x0027	-	0x0037	-
0x0008	-	0x0018	-	0x0028	-	0x0038	--
0x0009	-	0x0019	-	0x0029	-	0x0039	-
0x000A	-	0x001A	-	0x002A	-	0x003A	-
0x000B	-	0x001B	-	0x002B	-	0x003B	-
0x000C	-	0x001C	-	0x002C	-	0x003C	-
0x000D	-	0x001D	-	0x002D	-	0x003D	-
0x000E	-	0x001E	-	0x002E	-	0x003E	-
0x000F	-	0x001F	-	0x002F	-	0x003F	-

### 擴展XSFR ( 基底位址0xFF80 )

偏移位址	XSFR 名稱	偏移位址	XSFR 名稱	偏移位址	XSFR 名稱	偏移位址	XSFR 名稱
0x0000	T0_MAP	0x0010	PWM0_MAP	0x0020	TXD_MAP	0x0030	-
0x0001	T1_MAP	0x0011	PWM01_MAP	0x0021	RXD_MAP	0x0031	-
0x0002	-	0x0012	-	0x0022	SCL_MAP	0x0032	-
0x0003	T3_MAP	0x0013	-	0x0023	SDA_MAP	0x0033	-
0x0004	T4_MAP	0x0014	PWM1_MAP	0x0024	$\overline{SS}$ _MAP	0x0034	-
0x0005	T5_MAP	0x0015	PWM11_MAP	0x0025	SCK_MAP	0x0035	-
0x0006	-	0x0016	-	0x0026	MOSI_MAP	0x0036	-
0x0007	-	0x0017	-	0x0027	MISO_MAP	0x0037	-
0x0008	CAP0_MAP	0x0018	PWM2_MAP	0x0028	TXD2_MAP	0x0038	-
0x0009	CAP1_MAP	0x0019	PWM21_MAP	0x0029	RXD2_MAP	0x0039	-
0x000A	-	0x001A	-	0x002A	-	0x003A	-
0x000B	-	0x001B	-	0x002B	-	0x003B	-
0x000C	-	0x001C	PWM3_MAP	0x002C	-	0x003C	-
0x000D	ADCST_MAP	0x001D	-	0x002D	-	0x003D	-
0x000E	-	0x001E	-	0x002E	-	0x003E	-
0x000F	CLKO_MAP	0x001F	-	0x002F	-	0x003F	-

## 4 系統時鐘

### 4.1 系統時鐘特性

HC88L051F4 單片機系統時鐘有 4 種時鐘源可選：

- 外部高頻晶振時鐘 ( 4MHz~24MHz )
- 外部低頻晶振時鐘 ( 32.768KHz )
- 內部高頻 RC 時鐘 ( 32MHz )
- 內部低頻 RC 時鐘 ( 44KHz )

使用者選擇後的系統時鐘 ( 如果選擇的是內部高頻 RC，則經 RC32M\_DIV[1:0]分頻後的時鐘 ) 記做 osc\_clk，其頻率為  $F_{osc}$ ，週期為  $T_{osc}$ ，主要用於外設模組，osc\_clk 可以進行 1-255 之間任意值的分頻，分頻後的時鐘記做 CPU 時鐘，其頻率為  $F_{cpu}$ ，週期為  $T_{cpu}$ 。

晶片上電重定後，默認選擇內部高頻 RC 作為系統時鐘，其  $F_{osc}$  為 4MHz， $F_{cpu}$  為 2MHz，可以通過配置相關寄存器改變 osc\_clk 和 cpu\_clk 的頻率。

CPU 最高可以運行在 24MHz 頻率下，如果所選時鐘源頻率高於 24MHz，需要對其進行分頻，使 CPU 時鐘頻率等於或低於 24MHz。

RC32M 可以通過配置 TRMV 寄存器切換成 RC24M，具體實現過程詳見 TRMV 寄存器。

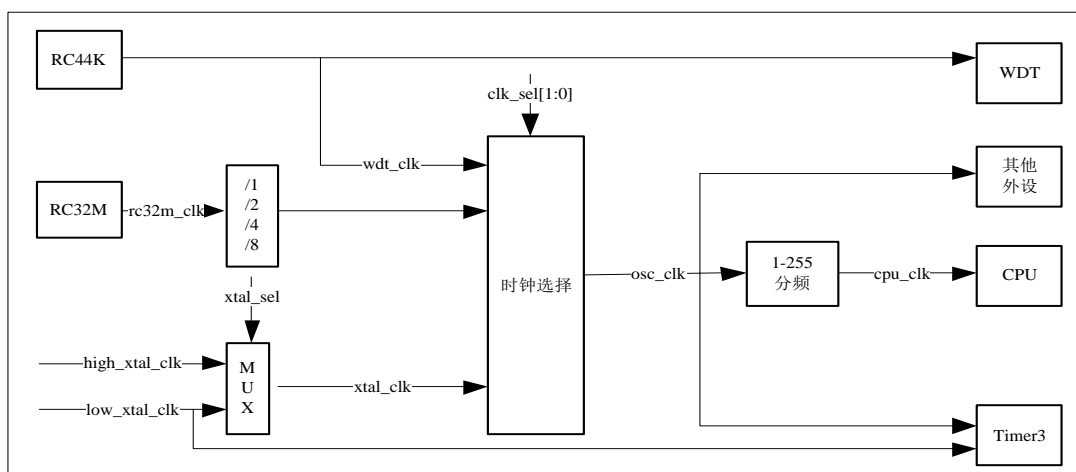


Figure 4-1 系統時鐘框圖

## 4.2 內部時鐘

內部時鐘有內部高頻 RC( RC32M )和內部低頻 RC( RC44K )兩種，使用者可通過軟體進行選擇。

內部低頻 RC ( RC44K ) 輸出的時鐘記做 wdt\_clk，用於看門狗計時器的計數，也可以用於系統時鐘；內部高頻 RC ( RC32M ) 輸出的時鐘記做 rc32m\_clk，可以進行 1/2/4/8 分頻。

## 4.3 外部時鐘

外部時鐘有外部高頻晶振時鐘 ( 4MHz~24MHz ) 和外部低頻晶振 ( 32.768KHz ) 兩種，使用者可通過軟體進行選擇。經過 XTALCFG 寄存器選擇的外部晶振時鐘記做 xtal\_clk。

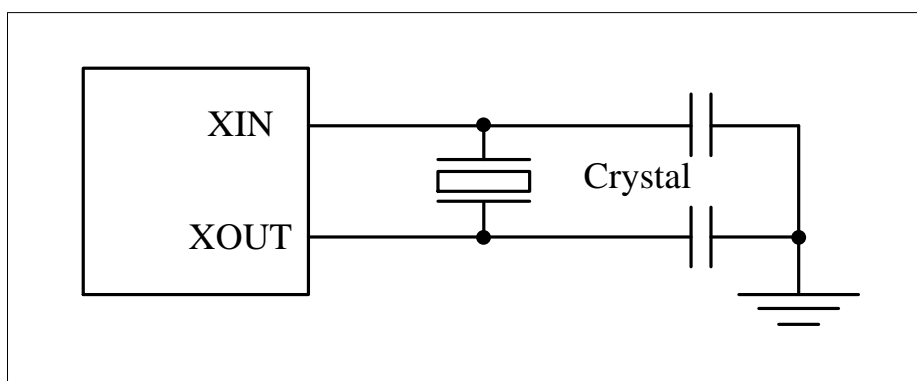


Figure 4-2 外部晶振典型應用

使用注意：

- 1、晶振起振電容推薦值為 20pF，該值可通過晶振基本的起振和運行測試，並非最優值。
- 2、外部晶振和 XIN、XOUT 埠之間的物理距離應在 10mm 以內。
- 3、使用外部晶振前，應充分瞭解所選晶振相關應用參數和要求，以獲得最佳性能。



## 4.4 系統時鐘相關寄存器

### 4.4.1 時鐘控制寄存器 CLKCON

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R	R/W	R/W	R
復位值	0	0	1	1	0	0	1	0
位元符號	HXTAL RDY	LXTAL RDY	HSRC RDY	LSRC RDY	-	XTALEN	HSRCEN	-

位編號	位元符號	說明
7	HXTALRDY	外部高頻晶振狀態位元  0：外部高頻晶振未準備 1：外部高頻晶振準備就緒  注：該位元硬體自動清 0 或置 1。
6	LXTALRDY	外部低頻晶振狀態位元  0：外部低頻晶振未準備 1：外部低頻晶振準備就緒  注：該位元硬體自動清 0 或置 1。
5	HSRCRDY	內部高頻 RC 振盪器狀態位元  0：內部高頻 RC 未準備 1：內部高頻 RC 準備就緒  注：該位元硬體自動清 0 或置 1。
4	LSRCRDY	內部低頻 RC 振盪器狀態位元

		0：內部低頻 RC 未準備 1：內部低頻 RC 準備就緒 注：該位元硬體自動清 0 或置 1。
3	-	保留位
2	XTALEN	外部晶振使能位 0：外部晶振關閉 1：外部晶振打開 注意：使能時，需要軟體將對應管腳的 IO 模式設置為類比頻道。
1	HSRCEN	內部高頻 RC 振盪器使能位 0：內部高頻 RC 關閉 1：內部高頻 RC 打開
0	-	保留位

#### 4.4.2 時鐘選擇寄存器 CLKSWR

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R/W	R/W	R	R	R/W	R/W
復位值	0	1	0	1	0	0	1	1
位元符號	CLKSTA[1:0]		CLKSEL[1:0]		-		RC32M_DIV[1:0]	
號								

位編	位元符號	說明
----	------	----

號		
7-6	CLKSTA[1:0]	<p>系統時鐘狀態位元</p> <p>00：當前系統時鐘為內部低頻 RC</p> <p>01：當前系統時鐘為內部高頻 RC</p> <p>10：當前系統時鐘為外部低頻晶振</p> <p>11：當前系統時鐘為外部高頻晶振</p> <p>注：系統根據當前系統時鐘自動切換各個狀態</p>
5-4	CLKSEL[1:0]	<p>系統時鐘選擇位元</p> <p>00：選擇系統時鐘為內部低頻 RC</p> <p>01：選擇系統時鐘為內部高頻 RC</p> <p>11、10：選擇系統時鐘為外部晶振</p> <p>注：系統時鐘選擇時，必須對應的時鐘源狀態位元為 1，否則將延續之前時鐘，切換後，原時鐘不會自動關閉；選擇後的系統時鐘記做 osc_clk，其頻率為 <math>F_{osc}</math>，週期為 <math>T_{osc}</math>。</p>
3-2	-	保留位
1-0	RC32M_DIV[1:0]	<p>內部高頻 RC 分頻係數</p> <p>00：rc32m_clk</p> <p>01：rc32m_clk /2</p> <p>10：rc32m_clk /4</p> <p>11：rc32m_clk /8 ( 默認 )</p>

### 4.4.3 時鐘分頻寄存器 CLKDIV

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	1	0
位元符號	CLKDIV[7:0]							

位編號	位元符號	說明
7-0	CLKDIV[7:0]	<p>CPU 時鐘分頻係數，預設為 2 分頻</p> <p>配置值為 0 或 1 時，時鐘不分頻；其他情況下，配置值等於分頻係數；</p> <p>注：分頻後的時鐘為 CPU 時鐘，其頻率為 <math>F_{cpu}</math>，週期為 <math>T_{cpu}</math>。</p>

### 4.4.4 時鐘輸出寄存器 CLKOUT

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R/W	R	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-			CLK_OUT_EN	-	CLK_OUT_SEL[2:0]		

位編號	位元符號	說明
7-5	-	保留位
4	CLK_OUT_EN	<p>時鐘輸出使能位元</p> <p>0：禁止時鐘輸出</p>

		1：允許時鐘輸出
3	-	保留位
2-0	CLK_OUT_SEL [2:0]	時鐘輸出選擇位元 000：選擇 cpu_clk 001：選擇 osc_clk 010：選擇 wdt_clk 011：選擇 xtal_clk 100：選擇 rc32m_clk 101：選擇 rc32m_clk/2 110：選擇 rc32m_clk/4 111：選擇 rc32m_clk/8

#### 4.4.5 外部晶振配置寄存器 XTALCFG

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	HXTAL_CFG		LXTAL_CFG		HXTAL_MODE_SEL		RC_PD_EN	XTAL_SEL

位編號	位元符號	說明
7-6	HXTAL_CFG	外部高頻晶振 warmup 計數值選擇，計數時鐘源為所選外部高頻晶振 00：2048 01：256

		10 : 16384  11 : 65536
5-4	LXTAL_CFG	外部低頻晶振 warmup 計數值選擇，計數時鐘源為所選外部低頻晶振  00 : 16384  01 : 4096  10 : 1024  11 : 65536
3-2	HXTAL_MODE_SEL	外部高頻晶振的選擇信號  00 : 選擇 4M/8M 晶振  01 : 選擇 4M/8M 晶振大驅動能力模式，在低壓工作時，起振時間短，且功耗適中  11 : 選擇 16M/24M 晶振
1	RC_EN_PD	系統產生 BOR 復位時，高頻內部 RC 關閉信號  0 : 系統產生 BOR 復位時，不關閉高頻內部 RC  1 : 系統產生 BOR 復位時，關閉高頻內部 RC  注：此位可用於 BOR 使能的情況下，降低 VDD 下降過程中的系統功耗。
0	XTAL_SEL	外部晶振選擇位  0 : 外部低頻晶振 32.768KHz  1 : 外部高頻晶振

#### 4.4.6 低頻晶振加速寄存器 LXTALSPD

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R	R	R	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-							SPD

位編號	位元符號	說明
7-1	-	保留位
0	SPD	32.768KHz 低頻晶振加速配置 當 $VDD \leq 3.6V$ 時，設置 SPD=1； 當 $VDD > 3.6V$ 時，設置 SPD=0。

#### 4.4.7 時鐘頻率寄存器 FREQ\_CLK

在進行 FLASH 的 IAP 擦寫或者系統進入掉電模式之前，需要配置擴展 SFR 裡 FREQ\_CLK 寄存器，指明目前 CPU 時鐘的頻率，FREQ\_CLK 寄存器配置的值等於 CPU 時鐘的頻率值，最小為 1MHz，假如目前 CPU 的運行頻率為 16MHz，那就配置寄存器 FREQ\_CLK=0x10。

##### FREQ\_CLK

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	1	0
位元符號	FREQ_CLK[7:0]							

位編號	位元符號	說明
7-0	FREQ_CLK[7:0]	<p>當前 CPU 時鐘頻率寄存器</p> <p>舉例如下：</p> <p>CPU 頻率為 24MHz 時，配置值為 0x18</p> <p>CPU 頻率為 16MHz 時，配置值為 0x10</p> <p>CPU 頻率為 8MHz 時，配置值為 0x08</p> <p>CPU 頻率為 4MHz 時，配置值為 0x04</p> <p>CPU 頻率為 2MHz 時，配置值為 0x02</p> <p>CPU 頻率小於等於 1MHz 時，配置值為 0x01</p>



#### 4.4.8 內部高頻 RC 調整使能寄存器 TRMEN

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R	R	R	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-							RCTRMEN

位編號	位元符號	說明
7-1	-	保留位
0	RCTRMEN	<p>內部高頻 RC 調整使能位</p> <p>1：使能內部高頻 RC 調整</p> <p>0：禁止內部高頻 RC 調整</p> <p>注：使能該寄存器後，必須立即配置 TRMV 寄存器，否則這個使能寄存器再執行完下一條指令後會被清零，內部高頻 RC 調整就會失效。</p>

#### 4.4.9 內部高頻 RC 調整配置寄存器 TRMV

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	x	x	x	x	x	x	x
位元符號	RC24M_SEL	RCTRMV						

位編號	位元符號	說明
7	RC24M_SEL	0 : RC32M 1 : RC24M 注：CPU 可以跑 24M，此時需要將 RC32M 切換成 RC24M，這樣外設時鐘和 CPU 時鐘都工作在 24MHz
6-0	RCTRMV	內部高頻 RC 調整配置值 注： 1. x 表示不確定的值，此寄存器的上電復位值為出廠的校準值。 2. 在配置這個寄存器值時，需要先將內部高頻 RC 調整使能位元配置為 1。 3. 根據校準曲線軟體先使能 RCTRMEN，緊接著就要配置 RCTRMV，在調整完成後 RCTRMEN 自動清零，防止重複操作

( 1 ) 將 RC32M 切換成 RC24M 代碼如下：

```

unsigned char code *hs24m_trim = 0x012c;
INSCON = 0X10;
TRMEN = 0x01;
TRMV = *hs24m_trim ^ 0x7f;
INSCON = 0X00;

```

( 1 ) 將 RC24M 切換成 RC32M 代碼如下：

```
unsigned char code *hs32m_trim = 0x0128;
```

```
INSCON = 0X10;
```

```
TRMEN = 0x01;
```

```
TRMV = *hs32m_trim ^ 0x7f;
```

```
INSCON = 0X00;
```

## 5 電源管理

### 5.1 電源管理特性

- 提供空閒模式 ( IDLE ) 和掉電模式 ( PD ), 作為省電模式
- 提供多種方式從空閒/掉電模式喚醒
- 提供低頻模式 ( 即時鐘分頻, 詳見[錯誤!未找到引用源。](#)章節相關介紹 )

### 5.2 空閒模式

空閒模式能夠降低系統功耗, 在此模式下, 程式中止運行, CPU時鐘停止, 但外部設備時鐘可繼續運行。空閒模式下, CPU在確定的狀態下停止, 並在進入空閒模式前所有CPU的狀態都被保存, 如PC、PSW、SFR、RAM等。

將PCON寄存器中的IDL位置1, 使HC88L051F4進入空閒模式。IDL位置1是CPU進入空閒模式之前執行的最後一條指令。

兩種方式可以退出空閒模式:

(1) 所有的有效中斷。HC88L051F4在檢測到一個有效中斷後, CPU時鐘立即恢復, 硬體清除PCON寄存器的IDL位元, 然後執行中斷服務程式, 隨後跳轉到進入空閒模式指令之後的指令。

(2) 重定信號 ( 外部重定引腳上出現有效電平、WDT 復位、BOR 復位或外部埠低壓檢測復位 )。

HC88L051F4 在檢測到有效復位後, PCON 寄存器中的 IDL 位被重定成零, 系統程式也會從重定位址 0000H 處開始執行, RAM 保持不變, SFR 的值根據不同功能模組改變。

### 5.3 掉電模式

掉電模式可以使HC88L051F4進入功耗非常低的狀態。掉電模式將停止CPU和週邊設備的所有時鐘

信號，但如果WDT和TIMER3使能且允許在掉電模式下工作，則WDT和TIMER3模組將繼續工作。在進入掉電模式前所有CPU的狀態都被保存，如PC、PSW、SFR、RAM等。

在晶片進入掉電模式之前，需要配置擴展SFR裡FREQ\_CLK寄存器，指明目前CPU時鐘的頻率，FREQ\_CLK寄存器配置的值等於CPU時鐘的頻率值，最小為1MHz，假如目前CPU的運行頻率為16MHz，那就配置寄存器FREQ\_CLK=0x10。

將PCON寄存器中的PD位置1，使HC88L051F4進入掉電模式。PD位置1是CPU進入掉電模式之前執行的最後一條指令。

注：如果同時設置IDL位和PD位，HC88L051F4進入掉電模式。退出掉電模式後，CPU也不會進入空閒模式，從掉電模式退出後硬體會清除IDL及PD位。

多種方式可以退出掉電模式：

(1) 有效外部中斷、LVD中斷、WDT中斷及TIMER3(計數時鐘源選擇外部低頻晶振或外部時鐘)中斷。在有效的外部中斷和TIMER3(計數時鐘源選擇外部低頻晶振或外部時鐘)中斷發生後，內部高頻RC振盪器啟動，CPU時鐘和外設時鐘立即恢復，PCON寄存器中的PD位元會被硬體清除，然後程式運行外部中斷服務程式。在完成外部中斷服務程式之後，跳轉到進入掉電模式之後的指令繼續運行。

(2) 重定信號 ( 外部重定引腳上出現有效電平、WDT 復位、BOR 復位或外部埠低壓檢測復位 )。

有效的重定信號將PCON寄存器中的PD位復位成零，振盪器重新開機，CPU時鐘和外設時鐘立即恢復，系統也會從重定位址0000H處開始運行，RAM保持不變，SFR的值根據不同功能模組改變。

## 5.4 電源管理相關寄存器

### 5.4.1 電源控制寄存器 PCON

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

復位值	0	0	0	0	0	0	0	0
位元符號	-				GF1	GF0	PD	IDL

位編號	位元符號	說明
7-4	-	保留位 ( 讀為 0 , 寫無效 )
3	GF1	使用者通用標誌位元 1
2	GF0	使用者通用標誌位元 0
1	PD	掉電模式控制位元 0 : 正常工作模式 1 : 進入掉電模式 ( 退出該模式後自動清 0 )
0	IDL	空閒模式控制位元 0 : 正常工作模式 1 : 進入空閒模式 ( 退出該模式後自動清 0 ) 注 : 若同時置 PD&IDL , 系統將進入掉電模式 , 喚醒後標誌被同時清除。

## 6 復位

### 6.1 復位特性

- 提供多種方式重定
- 所有的重定方式都有特定標誌

### 6.2 POR ( Power-On Reset ) 復位

HC88L051F4單片機在上電過程中，會產生一個POR信號，此信號會重定單片機，同時置位RSTFR寄存器裡的PORF位元，使用者可以判斷此標誌以來確定是否發生POR復位。

注：POR復位後的RAM值不穩定，建議使用者根據需要重新初始化相應RAM；其餘重定方式不會對RAM進行復位。

### 6.3 BOR ( Brown-Out Reset ) 復位

當VDD電壓下降到 $V_{BOR}$ 以下，且持續時間超過 $T_{BOR}$ 時，系統產生欠壓重定。BOR復位時，RSTFR寄存器的BORF位將被置1，使用者可以判斷此標誌以來確定是否發生BOR復位。

HC88L051F4可以通過代碼選項或者寄存器來選擇BOR檢測的電壓檔位元。當在代碼選項中配置完成BOR檔位元後，客戶也可以在使用時根據情況通過配置寄存器重新配置合適的BOR檢測電壓。BOR檔位：4.2V/3.9V/3.6V/3.0V/2.6V/2.4V/2.0V/1.8V。

BOR電壓檢測電路有一定的遲滯特性，遲滯電壓為0.1V左右。即當VDD電壓下降到所選BOR電壓檔位元時BOR重定有效，而VDD電壓需要上升到BOR檔位元電壓+0.1V時BOR復位才會解除。

欠壓復位示意圖如下所示，其中 $T_{BOR}$ 也可以通過寄存器配置，用來進行電壓消抖。

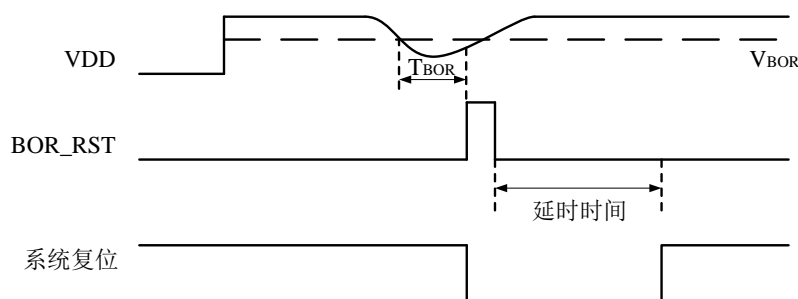


Figure 6-1 BOR 示意圖

## 6.4 外部 RST 復位

外部RST引腳復位就是從外部向RST引腳施加一定寬度的復位脈衝，從而實現單片機的復位，不使用時可以將其配置為I/O口，需要在代碼選項中設置。

做RST埠時，將RST復位管腳拉低並維持至少設定時間(軟體配置)後，單片機才會進入重定模式，將RST復位管腳拉回高電平後，單片機結束重定模式並從使用者程式區的0000H處開始正常工作。RST復位時，RSTFR寄存器的EXRSTF將被置1，使用者可以判斷此標誌以來確定是否發生外部RST復位。

注意：1、P2.7埠作為外部RST復位埠時，無法作為普通I/O使用。

2、如果晶片使能了外部RST功能且外部RST埠處於有效重定模式，此時晶片無法進入模擬或燒錄模式。

## 6.5 外部埠低壓檢測復位

當外部電壓過低時，無法保證單片機正常工作。此時，可以利用單片機的外部埠低壓檢測(PLVD)功能對單片機進行復位，外部埠檢測的電壓為1.2V，此復位功能可以被禁止。PLVD復位時，RSTFR寄存器的PLVRSTF將被置1，使用者可以判斷此標誌以來確定是否發生外部埠低壓檢測復位。另外，用戶也可以通過相關寄存器來對外部埠電壓檢測進行消抖。



## 6.6 軟體重定

對 IAP\_CMDH 和 IAP\_CMDL 寄存器按流程寫入相關值，系統將產生軟體重定，重定後 RSTFR 寄存器的 SWRF 將被置 1，使用者可以判斷此標誌以來確定是否發生軟體重定。具體操作詳見 FLASH IAP 章節相關介紹。

軟體重定前建議將系統時鐘切換到內部高頻 RC。軟體重定不會切換系統時鐘，但會將 CLKSWR 寄存器裡的 RC32M\_DIV[1:0]復位成 01B，CLKDIV 寄存器復位成 08H。

## 6.7 看門狗 ( WDT ) 復位

為了防止系統在異常情況下受到干擾，MCU程式跑飛，導致系統長時間異常工作，通常是引進看門狗，如果MCU程式中不在規定的時間內按要求操作看門狗，就認為MCU處於異常狀態，看門狗就會強制MCU重定，晶片重新從0000H開始運行。

注：要使WDT復位，必須置WDTRST為1，即允許WDT復位功能，否則即使允許WDT運行，WDT也只會置溢出標誌，並不會重定。

## 6.8 堆疊溢位復位

堆疊溢位時，系統將重定，並置 SPOVF 溢出標誌，必須軟體清除。

堆疊溢位包含入堆溢出及出棧溢出，入棧溢出是指當前棧頂地址為 0xFF，同時又有入棧動作；出棧溢出是指當前棧頂位址等於用戶設定的棧底位址，同時又有出棧動作。

堆疊溢位重定配置有使能寄存器，當使能時，堆疊溢位才能重定系統。

## 6.9 復位相關寄存器

### 6.9.1 重定標誌寄存器 RSTFR

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W
POR復位	1	x	x	x	x	0	x	x
EXRST復位	u	1	u	u	u	0	u	u
BOR復位	u	u	1	u	u	0	u	u
WDT復位	u	u	u	1	u	0	u	u
軟復位	u	u	u	u	1	0	u	u
堆疊溢位復位	u	u	u	u	u	0	1	u
PLVD復位	u	u	u	u	u	-	u	1
位元符號	PORF	EXRSTF	BORF	WDTRF	SWRF	-	SPOVF	PLVRSTF

注：x表示不確定的值，u表示該值由當前重定方式前的值決定，建議在POR復位後清零一下該寄存器。

位編號	位元符號	說明
7	PORF	上電重定標誌位元 0：無上電復位 1：發生上電重定，軟體清 0
6	EXRSTF	外部 RST 重定標誌位元 0：無外部 RST 復位 1：發生外部 RST 重定，軟體清 0
5	BORF	欠壓重定標誌位元 0：無欠壓復位

		1：發生欠壓重定，軟體清 0
4	WDTRF	WDT 重定標誌位元  0：無 WDT 復位  1：發生 WD 重定，軟體清 0
3	SWRF	軟體重定標誌位元  0：無軟體重定  1：發生軟體重定，軟體清0
2	-	保留
1	SPOVF	堆疊溢位標誌位元  0：無堆疊溢位復位  1：堆疊溢位重定，軟體清 0
0	PLVRSTF	外部埠電壓檢測重定標誌位元  0：外部埠電壓檢測重定  1：發生外部埠電壓檢測重定，軟體清 0

## 6.9.2 BOR 電壓檢測控制寄存器 BORC

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R	R	R	R/W	R/W	R/W
復位值	1	0	0	0	0	0	0	0
位元符號	BOREN	BOR_DBC_EN	-			BORVS[2:0]		

位編號	位元符號	說明
7	BOREN	BOR 使能位 0：禁止 BOR 1：允許 BOR
6	BOR_DBC_EN	BOR 消抖使能位 0：不使能 1：使能
5-3	-	保留位 ( 讀為 0，寫無效 )
2-0	BORVS[2:0]	BOR 檢測電壓點選擇位元  000：1.8V 001：2.0V 010：2.4V 011：2.6V 100：3.0V 101：3.6V 110：3.9V 111：4.2V

### 6.9.3 BOR 電壓檢測去抖控制寄存器 BORDBC

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符	BORDBC[7:0]							

號	
---	--

位編號	位元符號	說明
7-0	BORDBC[7:0]	<p>BOR 消抖控制位</p> <p>消抖時間 = BORDBC[7:0] * 8T<sub>CPU</sub> + 2 T<sub>CPU</sub></p> <p>注意：需要使能 BOR_DBC_EN，否則 BOR 不消抖。</p>

注：掉電模式下自動關閉 BOR 消抖功能，退出掉電模式自動打開。

#### 6.9.4 外部 RST 去抖控制寄存器 RSTDBC

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	1	1	1	1	1	1	1	1
位元符號	RSTDBC[7:0]							

位編號	位元符號	說明
7-0	RSTDBC[7:0]	<p>外部 RST 消抖控制位</p> <p>消抖時間 = RSTDBC[7:0] * 8T<sub>CPU</sub> + 2 T<sub>CPU</sub></p>

注意：掉電模式下自動關閉外部 RST 消抖功能，退出掉電模式又自動打開。

#### 6.9.5 堆疊溢位復位使能寄存器 SPOV\_RSTEN

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R	R	R	R/W
復位值	0	0	0	0	0	0	0	0

位元符 號	-	SPOV_RSTEN
----------	---	------------

位編號	位元符號	說明
7-1	-	保留位 ( 讀為 0 , 寫無效 )
0	SPOV_RSTEN	堆疊溢位復位使能位  0 : 不使能堆疊溢位復位  1 : 使能堆疊溢位復位

## 7 通用及複用I/O

### 7.1 通用及複用 I/O 特性

- 提供最多 18 個雙向 I/O 埠
- 多種模式可配

### 7.2 I/O 模式

HC88L051F4 所有 I/O 口均可由軟體配置成多種工作類型之一，具體為：輸入、帶上拉輸入、帶下拉輸入、模擬輸入、強推挽輸出、開漏輸出和開漏帶上拉輸出，並且輸入可以配置為施密特輸入。

如果 P2.7 被配置為重定腳，其埠是施密特輸入上拉狀態。

HC88L051F4在輸入模式時（不包含類比輸入），任何讀操作，資料來源都來自引腳電平。而在輸出模式時，通過指令來區分讀數據來源，採用“讀-修改-寫”指令時，為讀寄存器值，其它指令為讀引腳電平。

HC88L051F4首先將需要修改的寄存器的內容讀回ALU，對相應位進行修改，然後再整個寫回原來的寄存器位址，完成該功能的指令就叫做“讀-修改-寫”指令。

“讀-修改-寫”指令是單片機內部自己執行的，它發生在寫IO口的時候，當寫IO口的時候它先把IO的當前狀態讀回來，根據要寫的資料修改讀回來的資料，再寫到IO口；讀引腳是直接讀引腳的當前狀態，當前引腳是高電平，讀回來的就是高電平，低電平時讀回來的就是低電平。

“讀-修改-寫”指令包括以下指令：INC direct、DEC direct、ANL direct,A、ANL direct, #data、ORL direct,A、ORL direct, #data、XRL direct,A、XRL direct, #data、DJNZ direct,rel、MOV bit, C、CLR bit、SETB bit、CPL bit、JBC bit,rel。詳見第19章節的指令表。

### 7.3 I/O 功能框圖

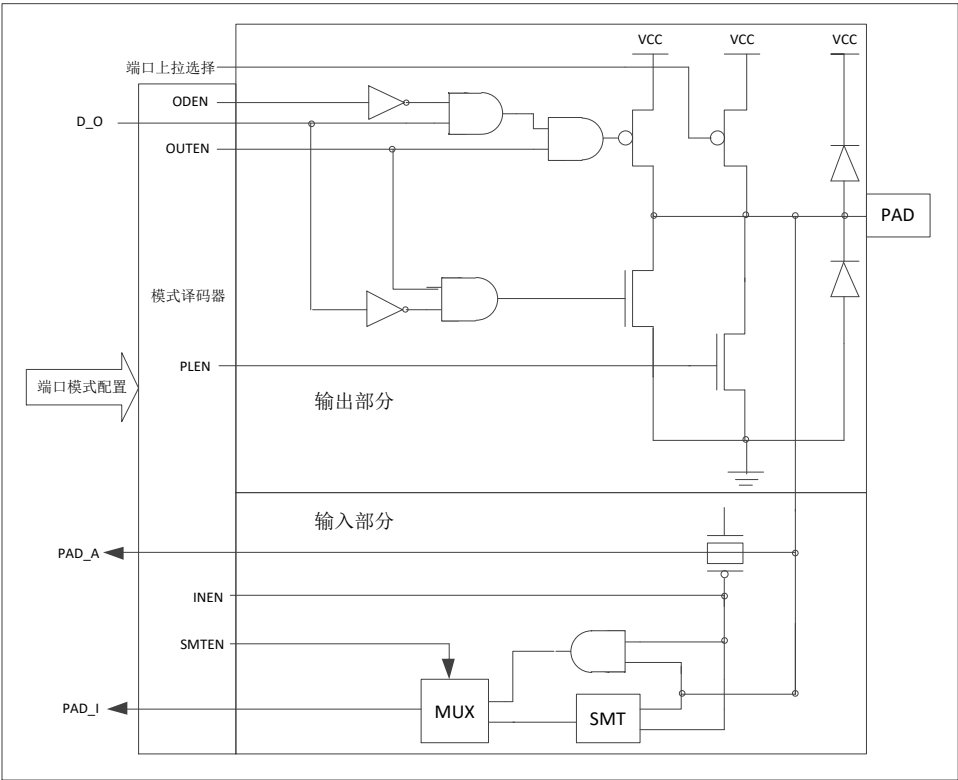


Figure 7-1 I/O 功能框圖

### 7.4 I/O 埠相關寄存器

#### 7.4.1 P0 埠資料寄存器 P0

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	P0[7:0]							
號								

位編號	位元符號	說明
7-0	P0[7:0]	P0 埠資料寄存器



### 7.4.2 P1 埠資料寄存器 P1

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R	R	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-	-	-	-	-	-	P1[1:0]	

位編號	位元符號	說明
1-0	P1[1:0]	P1 埠資料寄存器

### 7.4.3 P2 埠資料寄存器 P2

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	P2[7:0]							

位編號	位元符號	說明
7-0	P2[7:0]	P2 埠資料寄存器

### 7.4.4 P0 埠功能選擇寄存器 P0M0、P0M1、P0M2、P0M3

#### P0M0

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	1	1	0	0	1	1

位元符 號	P01M[3:0]	P00M[3:0]
----------	-----------	-----------

#### P0M1

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	1	1	0	0	1	1
位元符 號	P03M[3:0]				P02M[3:0]			

#### P0M2

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	1	1	0	0	1	1
位元符 號	P05M[3:0]				P04M[3:0]			

#### P0M3

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	1	1	0	0	1	1
位元符 號	P07M[3:0]				P06M[3:0]			

位編號	位元符號	說明
7-4 3-0	P0xM[3:0] (x = 0...7)	P0.x 埠模式配置位元  0000 : 輸入 ( 非施密特 )  0001 : 帶下拉輸入 ( 非施密特 )

		0010：帶上拉輸入（非施密特） 0011：模擬輸入 0100：輸入（施密特） 0101：帶下拉輸入（施密特） 0110：帶上拉輸入（施密特） 0111：保留（模擬輸入） 1x00：推挽輸出 1x01：開漏輸出 1x10：開漏帶上拉輸出 1x11：保留（推挽輸出）  注：x 為 0 或 1。
--	--	---

### 7.4.5 P1 埠功能選擇寄存器 P1M0

#### P1M0

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	1	1	0	0	1	1
位元符號	P11M[3:0]				P10M[3:0]			

位編號	位元符號	說明
7-4 3-0	P1xM[3:0] (x = 0...7)	P1.x 埠模式配置位元 0000：輸入（非施密特） 0001：帶下拉輸入（非施密特）

		0010：帶上拉輸入（非施密特）  0011：模擬輸入  0100：輸入（施密特）  0101：帶下拉輸入（施密特）  0110：帶上拉輸入（施密特）  0111：保留（模擬輸入）  1x00：推挽輸出  1x01：開漏輸出  1x10：開漏帶上拉輸出  1x11：保留（推挽輸出）  注：x 為 0 或 1。
--	--	---

#### 7.4.6 P2 埠功能選擇寄存器 P2M0、P2M1、P2M2、P2M3

##### P2M0

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	1	1	0	0	1	1
位元符號	P21M[3:0]				P20M[3:0]			

##### P2M1

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	1	1	0	0	1	1
位元符號	P23M[3:0]				P22M[3:0]			

號		
---	--	--

### P2M2

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	1	1	0	0	1	1
位元符號	P25M[3:0]				P24M[3:0]			
號								

### P2M3

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	1	1	0	0	1	1
位元符號	P27M[3:0]				P26M[3:0]			
號								

P2 組埠中 P2.7、P2.5、P2.4、P2.3 支援埠上下拉同時使能功能，此時埠輸出電壓約為 1/2VDD。

P2 組其餘埠不支援上述功能。具體配置說明見下：

位編號	位元符號	說明
7-4 3-0	P2xM[3:0] (x = 0、1、2、6)	P2.x 埠模式配置位元  0000：輸入（非施密特）  0001：帶下拉輸入（非施密特）  0010：帶上拉輸入（非施密特）  0011：模擬輸入  0100：輸入（施密特）  0101：帶下拉輸入（施密特）

		<p>0110：帶上拉輸入（施密特）</p> <p>0111：保留（模擬輸入）</p> <p>1x00：推挽輸出</p> <p>1x01：開漏輸出</p> <p>1x10：開漏帶上拉輸出</p> <p>1x11：保留（推挽輸出）</p> <p>注：x 為 0 或 1。</p>
--	--	--

位編號	位元符號	說明
<p>7-4</p> <p>3-0</p>	<p>P2xM[3:0]</p> <p>(x = 3、4、5、7)</p>	<p>P2.x 埠模式配置位元</p> <p>0000、0001、0010：輸入（非施密特）</p> <p>0011：模擬輸入</p> <p>0100：輸入（施密特）</p> <p>0101：帶下拉輸入（施密特）</p> <p>0110：帶上拉輸入（施密特）</p> <p>0111：類比頻道，上下拉同時使能</p> <p>1x00：推挽輸出</p> <p>1101：開漏輸出</p> <p>1110：開漏帶上拉輸出</p> <p>其他值：系統保留，請勿操作</p> <p>注：x 為 0 或 1。</p>

## 7.4.7 埠上拉電阻選擇寄存器

### P0LPU

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R/W	R/W	R	R	R	R
復位值	0	0	0	0	0	0	0	0
位元符號	-		P02PU[1:0]		-			

位編號	位元符號	說明
7-6	-	保留位
5-4	P02PU[1:0]	埠上拉電阻選擇位  00 : 50 K $\Omega$  01 : 100 K $\Omega$  10 : 150K $\Omega$  11 : 300 K $\Omega$  注：阻值為 VDD @5V 時參考值。
3-0	-	保留位

## 7.4.8 埠消抖控制寄存器 P00DBC、P01DBC、P02DBC

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	P0xDBCLK[1:0]		P0xDBCT[5:0]					

位編號	位元符號	說明
7-6	P0xDBCLK [1:0]	<p>埠消抖時鐘選擇</p> <p>00 : <math>F_{osc} / 1</math></p> <p>01 : <math>F_{osc} / 4</math></p> <p>10 : <math>F_{osc} / 16</math></p> <p>11 : <math>F_{osc} / 64</math></p> <p>注：x 為 0、1 或 2。</p>
5-0	P0xDBCT [5:0]	<p>埠消抖計數時鐘個數，當配置為 00 時，表示不消抖。</p> <p>消抖時間是指埠輸入時，其對應埠電平所需要維持的時間，需要注意的是，分配到這三個管腳上的功能腳、外部中斷輸入、故障檢測引腳也受消抖控制，其中 P02DBC[7:0]表示 P0.2 消抖控制寄存器。</p> <p>注意：P0xDBCT [5:0]配置的消抖時間是一個範圍，分頻係數 * <math>T_{osc}</math> * P0xDBCT [5:0] - <math>T_{osc}</math> &lt; 消抖時間 &lt; 分頻係數 * <math>T_{osc}</math> * (P0xDBCT [5:0] + 1) - <math>T_{osc}</math>。</p>



## 7.5 外設功能引腳全映射控制

### 7.5.1 外設功能引腳映射控制寄存器

擴展 SFR 地址	擴展 SFR 名稱	擴展 SFR 地址	擴展 SFR 名稱	擴展 SFR 地址	擴展 SFR 名稱	擴展 SFR 地址	擴展 SFR 名稱
0xFF80	T0_MAP	0xFF90	PWM0_MAP	0xFFA0	TXD_MAP	0xFFB0	-
0xFF81	T1_MAP	0xFF91	PWM01_MAP	0xFFA1	RXD_MAP	0xFFB1	-
0xFF82	-	0xFF92	-	0xFFA2	SCL_MAP	0xFFB2	-
0xFF83	T3_MAP	0xFF93	-	0xFFA3	SDA_MAP	0xFFB3	-
0xFF84	T4_MAP	0xFF94	PWM1_MAP	0xFFA4	$\overline{SS}$ _MAP	0xFFB4	-
0xFF85	T5_MAP	0xFF95	PWM11_MAP	0xFFA5	SCK_MAP	0xFFB5	-
0xFF86	-	0xFF96	-	0xFFA6	MOSI_MAP	0xFFB6	-
0xFF87	-	0xFF97	-	0xFFA7	MISO_MAP	0xFFB7	-
0xFF88	CAP0_MAP	0xFF98	PWM2_MAP	0xFFA8	TXD2_MAP	0xFFB8	-
0xFF89	CAP1_MAP	0xFF99	PWM21_MAP	0xFFA9	RXD2_MAP	0xFFB9	-
0xFF8A	-	0xFF9A	-	0xFFAA	-	0xFFBA	-
0xFF8B	-	0xFF9B	-	0xFFAB	-	0xFFBB	-
0xFF8C	-	0xFF9C	PWM3_MAP	0xFFAC	-	0xFFBC	-
0xFF8D	ADCST_MAP	0xFF9D	-	0xFFAD	-	0xFFBD	-
0xFF8E	-	0xFF9E	-	0xFFAE	-	0xFFBE	-
0xFF8F	CLKO_MAP	0xFF9F	-	0xFFAF	-	0xFFBF	-

注：以上 SFR 為外部擴展 XSFR，採用 MOVX 來進行讀寫。

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R/W	R/W	R	R/W	R/W	R/W
復位值	0	0	1	1	0	1	1	1
位元符號	-	-	FPORT[1:0]		-	FPIN[2:0]		
號								

位編號	位元符號	說明
7-6	-	保留位
5-4	FPORT[1:0]	映射埠選擇 00：P0

		01 : P1  10 : P2
3	-	保留位
2-0	FPIN[2:0]	映射埠輸出腳選擇  FPIN[2:0] = x(x = 0...7), 表示選擇對應埠名的 x(x = 0...7)腳

注意：輸出功能，系統將禁止其多對一映射，但輸入功能，系統將允許其多對一映射。

上面寄存器的復位值為 0x37，這樣復位後 IO 都為 GPIO，使用者在使用外設功能腳之前必須先配置上面的寄存器，否則外設功能將無法使用。

使用舉例：

將 UART1 的 TXD 和 RXD 分別映射到 P2.1 和 P2.2 上，用戶在啟動 UART1 之前應該配置下面命令：

```
TXD_MAP = 0x21;    //TXD-->P2.1
RXD_MAP = 0x22;    //RXD-->P2.2
```

如果使用者在下一次的設計時，需要重新將 UART1 的 TXD 和 RXD 分別映射到 P0.4 和 P0.5 上，

那使用者需要進行如下的配置：

```
TXD_MAP = 0x04;    //TXD-->P0.4
RXD_MAP = 0x05;    //RXD-->P0.5
```

多個輸出映射到一個埠上時，只能有一個輸出有效，下面是默認的優先順序

優先順序順序	複用埠功能
1	PWM0
2	PWM01
3	PWM1
4	PWM11
5	PWM2
6	PWM21
7	PWM3
8	CLKO
9	T0_OUT

10	T1_OUT
11	T4_OUT
12	TXD
13	RXD
14	SCK
15	MOSI
16	MISO
17	TXD2
18	SCL
19	SDA
20	T5_OUT

比如：CLKO\_MAP 配置為 0x01 選擇 P0.1 口作為 CLKO 的輸出口，T4\_MAP 也配置為 0x01，這個時候硬體會按上面的優先順序，P0.1 將配置為 CLKO 的輸出口，而 T4\_MAP 的配置無效。

當所有的埠映射控制寄存器都不等於 0x01 時，即所有的功能口都不選擇 P0.1 作為輸入輸出口，此時這個埠的輸出就是 P0 埠資料寄存器的第 1 位。

輸入可以配置為多個功能從一個 PAD 引腳進入，比如：

T3\_MAP 配置為 0x23，則選擇 P2.3 作為 T0 的輸入口，T5\_MAP 也配置為 0x23，這樣從 P2.3 埠進入的信號同時作用於 T3 和 T5。

將 TXD 和 RXD 都配置到一個埠上時，並且此埠設置為輸出，則 TXD 和 RXD 將內部連接起來。

在輸入時，無論埠是什麼功能，讀埠資料寄存器都讀晶片引腳上值。

## 8 中斷

### 8.1 中斷特性

- 19 個中斷源
- 4 級中斷優先順序
- 16 個外部中斷

### 8.2 中斷匯總

中斷源	向量地址	允許位	標誌位元	查詢優先順序	中斷號 ( C 語言 )
INT0	0003H	EX0	INT0F	1(最高)	0
T0	000BH	ET0	TF0	2	1
INT1	0013H	EX1	INT1F	3	2
T1	001BH	ET1	TF1	4	3
UART1	0023H	ES1	TI/RI	5	4
WDT	002BH	EWDI	WDTRF	6	5
LVD	0033H	LVDIE	LVDF	7	6
UART2	003BH	ES2	TI/RI	8	7
SPI	0043H	ESPI	SPIF/MODF	9	8
T3	004BH	ET3	TF3	10	9
T4	0053H	ET4	TF4	11	10
PWM	005BH	PWMxIE (x = 0...3)	PWMxIF (x = 0...3)	12	11
T5	0063H	ET5	TF5	13	12
ADC	006BH	EADC	ADCIF/AMWIF	14	13
INT2-INT7	0073H	EINx (x = 2...7)	INTxF (x = 2...7)	15	14
INT8-INT15	007BH	EINx ( x=8...15 )	INTxF (x = 8...15)	16	15
IIC	0083H	EIIC	SI	17	16
FLT	008BH	FLT_INT_EN	FLTx_INT_FLAG	18	17

			(x = 0...2)		
TICK	0093H	TICKIE	TICKIF	19	18

注意：除以上允許位元及標誌位元被置位元外，要回應中斷必須中斷總開關 EA 也被使能，否則不回應任何中斷。

## 8.3 中斷向量

當一個中斷產生時，程式計數器內容被壓棧，相應的中斷向量位址被載入程式計數器。中斷向量的入口位址詳見錯誤!未找到引用源。章節。

## 8.4 中斷優先順序

每個中斷源都可被單獨設置為4個中斷優先順序之一，分別通過IP0、IP1、IP2、IP3中相應位來實現。中斷優先順序服務程式描述如下：

回應一個中斷服務程式時，可回應更高優先順序的中斷，但不能回應同優先順序或低優先順序的另一個中斷。

回應最高級中斷服務程式時，不回應其它任何中斷。如果不同中斷優先順序的中斷源同時申請中斷時，回應較高優先順序的中斷申請。

如果同優先順序的中斷源在運算速度開始時同時申請中斷，那麼內部查詢優先順序確定插斷要求回應順序。查詢優先順序詳細參照中斷匯總。

中斷優先順序	
優先順序控制位(x 為功能模組)	優先順序
Px[1:0]	
00	優先順序 0 ( 最低 )

01	優先順序 1
10	優先順序 2
11	優先順序 3 ( 最高 )

## 8.5 中斷處理

中斷標誌在CPU時鐘的上升沿被採樣，如果一個標誌被置起，那麼CPU捕獲到後中斷系統調用一個長轉移指令（LCALL）調用其中斷服務程式，但由硬體產生的LCALL會被下列任何條件阻止：

1. 同級或更高級的優先順序中斷在運行中。
2. 當前的週期不是執行中指令的最後一個週期。換言之，正在執行的指令完成前，任何插斷要求都得不到回應。
3. 正在執行的是一條 RETI 或者訪問專用寄存器 IE/IE1 或是 IP0/IP1/IP2/IP3 的指令。換言之，在 RETI 或者讀寫 IE/IE1 或是 IP0/IP1/IP2/IP3 之後，不會馬上回應插斷要求，而至少在執行一條其它指令之後才會回應。

中斷服務程式ISR完成和該中斷相應的一些操作。ISR以RETI（中斷返回）指令結束，將PC值從棧中取回，並恢復原來的中斷設置，之後從主程序的中斷點處繼續執行。

當某中斷被回應時，被裝載到程式計數器PC中的數值稱為中斷向量，是該中斷源相對應的中斷服務程式的起始位址。各中斷源服務程式的入口位址（即中斷向量）明細可參照中斷匯總。

由於中斷向量入口位址位於程式記憶體的开始部分，所以主程序的第1條指令通常為跳轉指令，越過中斷向量區（LJMP MAIN）。

需要注意的是，不能用RET指令代替RETI指令，RET指令雖然也能控制PC返回到原來中斷的地方，但RET指令沒有清零中斷優先順序狀態觸發器的功能，中斷控制系統會認為中斷仍在進行，其後果是與此同級或低級的插斷要求將不被回應。

若使用者在中斷服務程式中進行了入棧操作，則在RETI指令執行前應進行相應的出棧操作，即在中斷服務程式中PUSH指令與POP指令必須成對使用，否則不能正確返回。

## 8.6 中斷回應時間

每一個中斷的回應時間都不同，這取決於中斷自身特點和發生中斷時正在執行的指令。如果檢測出一個中斷，這個中斷的請求標誌位元就會被置起，內部電路會保持這個標誌位元，CPU會在第二個時鐘週期產生中斷。如果回應有效且條件允許，在下一個指令執行的時候硬體LCALL指令將調用請求中斷的服務程式，否則中斷被掛起。執行LCALL指令需要3個時鐘週期。因此，從中斷標誌置位元到開始執行中斷服務程式至少需要5個時鐘週期。

當插斷要求因前述的三個情況受阻時，中斷回應時間就會加長。如果同級或更高優先順序的中斷正在執行，額外的等待時間取決於正執行的中斷服務程式的長度。

如果正在執行的指令還沒有進行到最後一個週期，假如正在執行RETI指令，則完成正在執行的RETI指令，需要4個時鐘週期，加上為完成下一條指令所需的最長時間4個時鐘週期，若系統中只有一個中斷源，再加上LCALL調用指令3個時鐘週期，則最長的回應時間是13個時鐘週期。

因此一個簡單的中斷系統回應時間，總是大於5個時鐘週期並且不超過13個時鐘週期。

## 8.7 外部中斷

HC88L051F4有4個外部中斷向量入口，外部中斷0~1分別有一個獨立的中斷向量入口，外部中斷2~7共用一個中斷向量入口，外部中斷8~15共用一個中斷向量入口，因而共有16個外部中斷埠輸入，所有的中斷都可以設定4種觸發方式，分別為上升沿、下降沿、雙沿及低電平。

當調用中斷服務程式後，外部中斷0~15標誌必須軟體清除。如果中斷服務完成後而外部中斷仍舊維持，則會產生下一次中斷。

外部中斷0~2功能埠分別在埠P0.0~P0.2上，在使用外部中斷0~2時用戶可設置外部中斷0~2產生有效  
插斷要求所需要的消抖時間，設置埠P0.0~P0.2的消抖時間即可，具體設置詳見埠消抖控制寄存器  
P00DBC、P01DBC、P02DBC。



## 8.8 中斷相關寄存器

### 8.8.1 中斷允許寄存器 IE、IE1

IE

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	EA	ES2	EWDT	ES1	ET1	EX1	ET0	EX0

位編號	位元符號	說明
7	EA	CPU 總中斷允許控制位 0：禁止 CPU 中斷 1：允許 CPU 中斷
6	ES2	UART2 中斷允許位 0：禁止 UART2 中斷 1：允許 UART2 中斷
5	EWDT	WDT 中斷允許位 0：禁止 WDT 中斷 1：允許 WDT 中斷
4	ES1	UART1 中斷允許位 0：禁止 UART1 中斷 1：允許 UART1 中斷

3	ET1	T1 中斷允許位 0：禁止 T1 中斷 1：允許 T1 中斷
2	EX1	外部中斷 1 中斷允許位 0：禁止 INT1 中斷 1：允許 INT1 中斷
1	ET0	T0 中斷允許位 0：禁止 T0 中斷 1：允許 T0 中斷
0	EX0	外部中斷 0 中斷允許位 0：禁止 INT0 中斷 1：允許 INT0 中斷

# IE1

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	EX8_15	EX2_7	EADC	ET5	EIIC	ET4	ET3	ESPI

位編號	位元符號	說明
7	EX8_15	外部中斷 8~15 中斷允許位 0：禁止 INT8~INT15 中斷

		1：允許 INT8~INT15 中斷  注：INT8~INT15 共用同一中斷向量。
6	EX2_7	外部中斷 2~7 中斷允許位  0：禁止 INT2~INT7 中斷 1：允許 INT2~INT7 中斷  注：INT2~INT7 共用同一中斷向量。
5	EADC	ADC 轉換完成中斷允許位  0：禁止 ADC 中斷 1：允許 ADC 中斷
4	ET5	T5 中斷允許位  0：禁止 T5 中斷 1：允許 T5 中斷
3	EIIC	IIC 中斷允許位  0：禁止 IIC 中斷 1：允許 IIC 中斷
2	ET4	T4 中斷允許位  0：禁止 T4 中斷 1：允許 T4 中斷
1	ET3	T3 中斷允許位  0：禁止 T3 中斷 1：允許 T3 中斷
0	ESPI	SPI 中斷允許位

		0 : 禁止 SPI 中斷  1 : 允許 SPI 中斷
--	--	------------------------------------

## 8.8.2 中斷優先順序選擇寄存器 IP0、IP1、IP2、IP3、IP4

### IP0

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PT1[1:0]		PX1[1:0]		PT0[1:0]		PX0[1:0]	

位編號	位元符號	說明
7-6	PT1[1:0]	T1 中斷優先順序控制位
5-4	PX1[1:0]	INT1 中斷優先順序控制位
3-2	PT0[1:0]	T0 中斷優先順序控制位
1-0	PX0[1:0]	INT0 中斷優先順序控制位

### IP1

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PS2[1:0]		PLVD[1:0]		PWDT[1:0]		PS1[1:0]	

位編號	位元符號	說明
7-6	PS2[1:0]	UART2 中斷優先順序控制位

5-4	PLVD[1:0]	LVD 中斷優先順序控制位
3-2	PWDT[1:0]	WDT 中斷優先順序控制位
1-0	PS1[1:0]	UART1 中斷優先順序控制位

## IP2

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PPWM[1:0]		PT4[1:0]		PT3[1:0]		PSPI[1:0]	

位編號	位元符號	說明
7-6	PPWM [1:0]	PWM 中斷優先順序控制位
5-4	PT4[1:0]	T4 中斷優先順序控制位
3-2	PT3[1:0]	T3 中斷優先順序控制位
1-0	PSPI [1:0]	SPI 中斷優先順序控制位

## IP3

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PX8_15 [1:0]		PX2_7[1:0]		PADC[1:0]		PT5[1:0]	

位編號	位元符號	說明
7-6	PX8_15 [1:0]	INT8_15 中斷優先順序控制位

5-4	PX2_7[1:0]	INT2_7 中斷優先順序控制位
3-2	PADC[1:0]	ADC 中斷優先順序控制位
1-0	PT5[1:0]	T5 中斷優先順序控制位

# IP4

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-		PTICK[1:0]		PFLT[1:0]		PIIC [1:0]	
號								

位編號	位元符號	說明
7-6	-	保留位
5-4	PTICK[1:0]	TICK 計時器中斷優先順序控制位
3-2	PFLT[1:0]	PWM 故障檢測中斷優先順序控制位元
1-0	PIIC [1:0]	IIC 中斷優先順序控制位

中斷優先順序	
優先順序控制位(x 為功能模組)	優先順序
Px[1:0]	
00	優先順序 0 ( 最低 )
01	優先順序 1
10	優先順序 2
11	優先順序 3 ( 最高 )

## 8.8.3 外部中斷電平選擇寄存器 PITS0、PITS1、PITS2、PITS3

### PITS0

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

復位值	0	0	0	0	0	0	0	0
位元符 號	IT3[1:0]		IT2[1:0]		IT1[1:0]		IT0[1:0]	

### PITS1

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符 號	IT7[1:0]		IT6[1:0]		IT5[1:0]		IT4[1:0]	

### PITS2

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符 號	IT11[1:0]		IT10[1:0]		IT9[1:0]		IT8[1:0]	



### PITS3

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	IT15[1:0]		IT14[1:0]		IT13[1:0]		IT12[1:0]	
號								

位編號	位元符號	說明
7-6 5-4 3-2 1-0	ITx[1:0] (x = 0...15)	外部中斷觸發沿選擇位  00：低電平中斷  01：下降沿中斷  10：上升沿中斷  11：雙沿中斷

## 8.8.4 外部中斷 2-15 使能控制寄存器 PINTE0、PINTE1

### PINTE0

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R
復位值	0	0	0	0	0	0	0	0
位元符號	EINT7	EINT6	EINT5	EINT4	EINT3	EINT2	-	
號								

位編號	位元符號	說明
7-2	EINTx (x = 2...7)	外部中斷控制位(INT2~INT7)  0：禁止該埠中斷

		<p>1：允許該埠中斷</p> <p>注意：只要相應的 EINT<sub>x</sub>(x =2...7)被允許，對應的中斷標誌就可能被置 1，反之，對應標誌不會被置 1。</p>
1-0	-	保留位（讀為 0，寫無效）

### PINTE1

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	EINT15	EINT14	EINT13	EINT12	EINT11	EINT10	EINT9	EINT8

位編號	位元符號	說明
7-0	EINT <sub>x</sub> (x =8...15)	<p>外部中斷控制位(INT8~INT15)</p> <p>0：禁止該埠中斷</p> <p>1：允許該埠中斷</p> <p>注意：1、只要相應的 EINT<sub>x</sub>(x =8...15)被允許，對應的中斷標誌就可能被置 1，反之，對應標誌不會被置 1。</p> <p>2、外部中斷在使用查詢方式時，無法正常清除中斷標誌位元。可以通過先禁止中斷使能位元再清除中斷標誌，在完成清除中斷標誌位元後再使能外部中斷來進行正常的外部中斷產生查詢。使用中斷方式時，無該問題。</p>

## 8.8.5 外部中斷標誌寄存器 PINTF0、PINTF1

### PINTF0

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	INT7F	INT6F	INT5F	INT4F	INT3F	INT2F	INT1F	INT0F

位編號	位元符號	說明
7-2	INTxF (x=2...7)	INT2-INT7 插斷要求標誌位元  0：軟體清 0  1：符合外部中斷時，硬體置 1
1-0	INTxF (x = 0,1)	INT0 和 INT1 插斷要求標誌位元  0：軟體清 0  1：符合外部中斷時，硬體置 1

### PINTF1

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	INT15F	INT14F	INT13F	INT12F	INT11F	INT10F	INT9F	INT8F

位編號	位元符號	說明
7-0	INTxF (x=8...15)	INT8-INT15 插斷要求標誌位元  0：軟體清 0  1：符合外部中斷時，硬體置 1

## 8.8.6 外部中斷 01 管腳選擇寄存器 INT01\_PINS

### INT01\_PINS

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R	R	R/W	R/W
復位值	0	0	0	0	0	0	0	0

位元符 號	-	INT1_PINS	INT0_PINS
----------	---	-----------	-----------

位編號	位元符號	說明
7-2	-	保留
1	INT1_PINS	INT1 管腳選擇位 0 : P0.1 1 : P1.1
0	INT0_PINS	INT0 管腳選擇位 0 : P0.0 1 : P1.0

## 9 計時器/計數器

### 9.1 計時器/計數器特性

- 計時器/計數器 T0&T1 是不完全相容標準 8051，差異主要是在方式 0 的功能定義不同
- 計時器/計數器 T0&T1 支持 16 位自動重載

### 9.2 計時器/計數器 Tx(x = 0,1)

#### 9.2.1 計時器/計數器 Tx(x = 0,1)的工作方式

每個計時器的兩個資料寄存器 ( THx & TLx (x = 0,1) ) 可作為一個 16 位寄存器來訪問，它們由寄存器 TCON 和 TMOD 控制。IE0 寄存器的 ET0 和 ET1 位置 1 能允許計時器 0 和計時器 1 中斷。( 詳見 [錯誤!未找到引用源。](#) 章節 )。

通過計數器/計時器方式寄存器 ( TMOD ) 的方式選擇位元 Mx[1:0]，選擇計時器工作方式。

Mx[1:0]	工作方式	描述
00	方式0	16位自動重載計時器/計數器
01	方式1	16 位計時器/計數器
10	方式2	8位自動重載計時器/計數器
11	方式3	T0分成兩個 ( TL0/TH0 ) 獨立的8位計時器/計數器 ( T1無此模式 )

### 9.2.1.1 方式0: 16位自動重載計時器/計數器

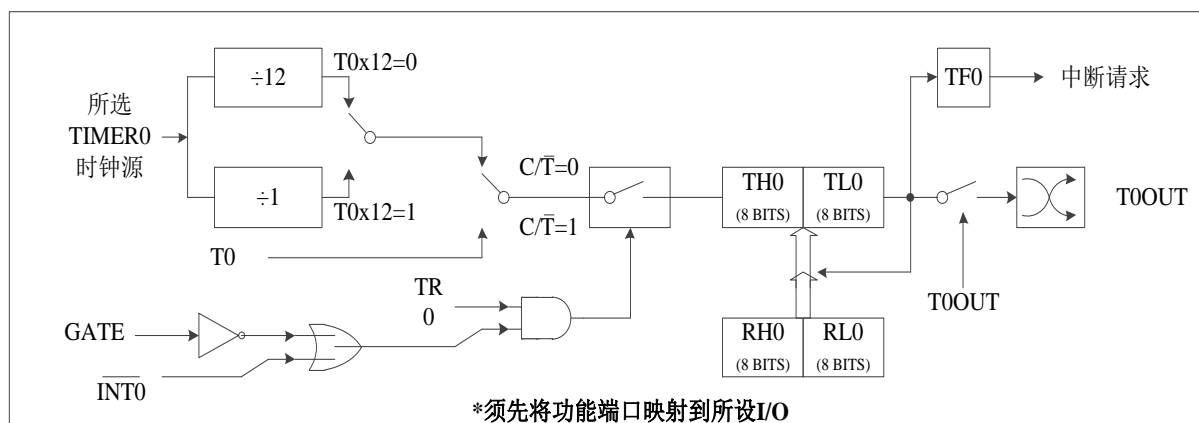


Figure 9-1 TIMER0 方式 0 功能框圖

方式 0 與標準 8051 功能不相容，在此方式下為 16 位元自動重載計時器/計數器，當  $TH_x$  &  $TL_x$  ( $x = 0, 1$ ) 被寫時，用作計時器重載寄存器；當被讀時，被用作計數寄存器。 $TR_x$  ( $x = 0, 1$ ) 為 0 時，按順序寫  $TH_x$  &  $TL_x$  ( $x = 0, 1$ ) 兩個寄存器，寫的值同時被寫進重載寄存器和計數寄存器， $TR_x$  ( $x = 0, 1$ ) 置 1，計數寄存器開始從寫的值遞增計數，在計數到 0xFFFF 後，再來一個計數時鐘，計數器就會發生溢出，此時  $TF_x$  ( $x = 0, 1$ ) 被置為 1，同時重載寄存器的 16 位元資料被自動重載入計數寄存器中，計數器又開始從這個重載的數值遞增計數。

在  $TR_x$  ( $x = 0, 1$ ) 為 1 時，對  $TH_x$  &  $TL_x$  ( $x = 0, 1$ ) 的寫操作，不會影響計數器的值，只能改變重載寄存器的值，這個改變後的值在下次溢出時重載進計數寄存器中。只有  $TR_x$  ( $x = 0, 1$ ) 為 0 時，對  $TH_x$  &  $TL_x$  ( $x = 0, 1$ ) 的寫操作，同時會改變計數寄存器和重載寄存器的值。

由於對  $TL_x$  ( $x = 0, 1$ ) 和  $TH_x$  ( $x = 0, 1$ ) 的寫操作需要 2 條指令才能完成，為保證精確計數，對  $TH_x$  ( $x = 0, 1$ ) 和  $TL_x$  ( $x = 0, 1$ ) 寄存器的寫操作都以對  $TL_x$  ( $x = 0, 1$ ) 寄存器的寫操作作為基準。當寫入重載寄存器時，寫  $TH_x$  ( $x = 0, 1$ ) 寄存器不會立即生效，而是暫存在一個緩衝寄存器中，只有對  $TL_x$  ( $x = 0, 1$ ) 寄存器的寫操作才會使  $TH_x$  ( $x = 0, 1$ ) 和  $TL_x$  ( $x = 0, 1$ ) 寄存器同時生效。

因此， $TH_x$  ( $x = 0, 1$ ) 和  $TL_x$  ( $x = 0, 1$ ) 讀寫操作遵循以下順序：

寫操作：先高位後低位元

讀操作：先高位後低位元

需要注意的是寫操作時，當  $TR_x(x=0,1)$  為 0，先寫高位再寫低位元，重載資料將直接重載到計數寄存器中，當  $TR_x(x=0,1)$  為 1，先寫高位再寫低位元，重載資料只會在下一次溢出時才會被重載到計數寄存器中。若先寫低位元再寫高位，高位資料將無效(無效：表示發生重載時對應資料不能被更新)，直到下一次操作寫入低位元資料，前一次寫高位的資料才會有效(有效表示發生重載時對應資料可以被更新)。若只寫低位元時，低位元資料也將有效，例如對 T0 依次做如下操作時：

- (1)  $TH0 = 0x05$ ;
- (2)  $TL0 = 0x08$ ; //此時若發生重載，重載到計數器中的資料為 0x0508
- (3)  $TH0 = 0x06$ ; //此時若發生重載，重載到計數器中的資料仍為 0x0508
- (4)  $TL0 = 0x08$ ; //此時若發生重載，重載到計數器中的資料為 0x0608
- (5)  $TL0 = 0x09$ ; //此時若發生重載，重載到計數器中的資料為 0x0609

顯然只要修改重載資料，低位元都必須再寫入一次，建議每次修改同時修改。

注：方式 1、2、3 時無此要求。

### 9.2.1.2 方式1: 16位計時器/計數器

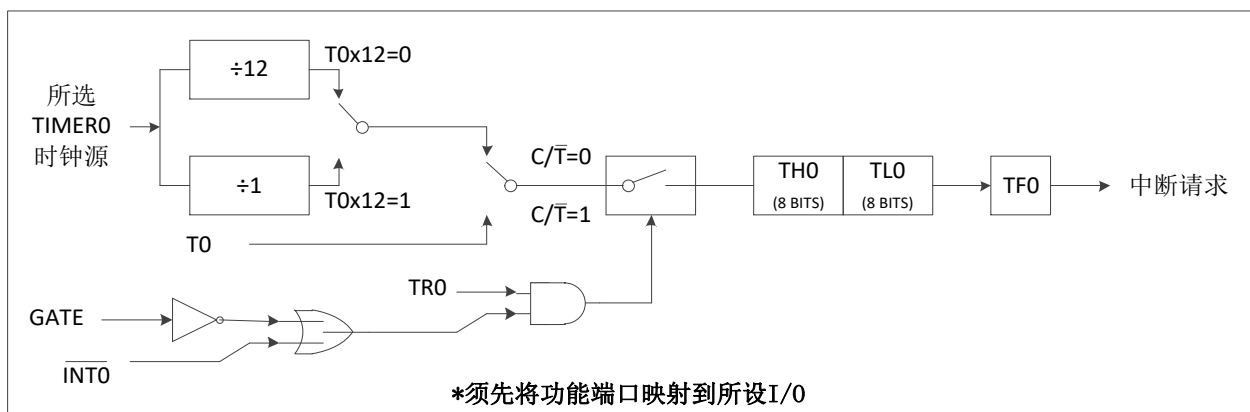


Figure 9-2 TIMER0 方式 1 功能框圖

在方式1中，計時器 $T_x(x=0,1)$ 為16位元數目器/計時器。 $TH_x(x=0,1)$ 寄存器存放16位元數目器/計時器的高8位， $TL_x(x=0,1)$ 存放低8位。當16位元計時器寄存器遞增溢出時，系統置起計時器溢出標誌 $TF_x(x=0,1)$ 。如果計時器 $x$ 中斷被允許，將會產生一個中斷。



$C/\overline{T}_x$  ( $x = 0,1$ )位選擇計數器/計時器的功能，如果 $C/\overline{T}_x$  ( $x = 0,1$ ) = 1，將工作在外部計數模式，當出現計時器 $T_x$  ( $x = 0,1$ )外部計數時鐘的下降沿時，將使計時器 $T_x$ 資料寄存器加1。如果 $C/\overline{T}_x$  ( $x = 0,1$ ) = 0，選擇系統時鐘為計時器 $T_x$  ( $x = 0,1$ )的時鐘源。

當 $GATE_x$  ( $x = 0,1$ ) = 0時， $TR_x$ 置1則打開計時器。

當 $GATE_x$  ( $x = 0,1$ ) = 1時，只有在外輸入信號 $INT_x$  ( $x = 0,1$ )為高電平時 $TR_x$  ( $x = 0,1$ )才會被置1，計時器 $T_x$ 才會計數，從而可測量 $INT_x$  ( $x = 0,1$ )的正脈衝寬度。 $TR_x$  ( $x = 0,1$ )位置1不強行復位計時器，這意味著如果 $TR_x$ 置1，計時器寄存器將從上次 $TR_x$  ( $x = 0,1$ )清0時的值開始計數。所以在允許計時器之前，應該設定計時器寄存器的初始值。

### 9.2.1.3 方式2: 8位自動重載計時器/計數器

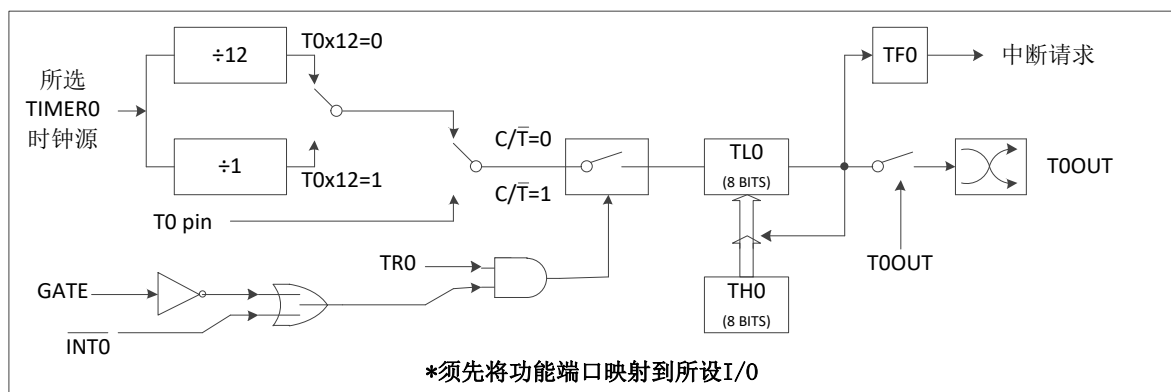


Figure 9-3 TIMER0 方式 2 功能框圖

方式2中，計時器 $T_x$  ( $x = 0,1$ )是8位自動重載計數器/計時器。 $TL_x$  ( $x = 0,1$ )存放計數值， $TH_x$  ( $x = 0,1$ )存放重載值。當在 $TL_x$  ( $x = 0,1$ )中的計數器遞增至0x00時，置起計時器溢出標誌 $TF_x$  ( $x = 0,1$ )，寄存器 $TH_x$  ( $x = 0,1$ )的值被重載入寄存器 $TL_x$  ( $x = 0,1$ )中。如果計時器中斷使能，當 $TF_x$  ( $x = 0,1$ )置1時將產生一個中斷。而在 $TH_x$  ( $x = 0,1$ )中的重載值不會改變。在允許計時器正確計數開始之前， $TL_x$  ( $x = 0,1$ )必須初始化為所需的值。

除了自動重載功能外，方式2中的計數器/計時器的使能和配置與方式1和0是一致的。可配置寄存器TCON2中的 $T_xX12$  ( $x = 0,1$ )位元選擇系統時鐘或系統時鐘的1/12作為計時器 $T_x$  ( $x = 0,1$ )的時鐘源。

當作為計時器應用時，可配置寄存器 TCON1 中的  $TxOUT[1:0](x = 0,1)$  位使計時器  $Tx(x = 0,1)$  溢出時  $Tx(x = 0,1)$  腳自動翻轉。

### 9.2.1.4 方式3： 兩8位計時器/計數器(T1無此方式)

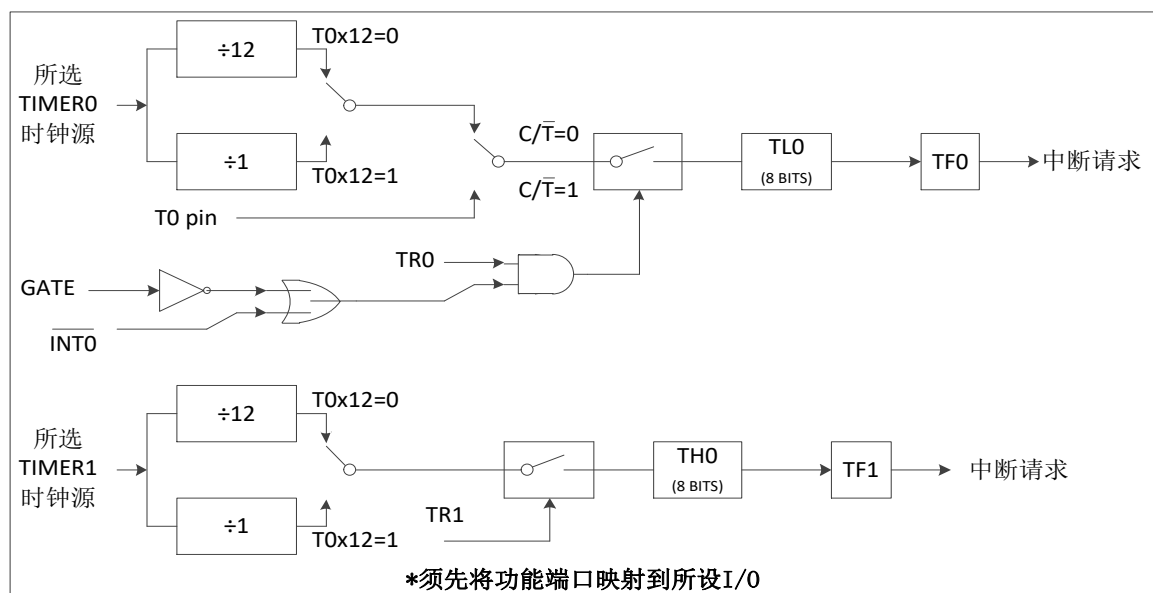


Figure 9-4 TIMER0 方式 3 功能框圖

在方式3中，計時器T0用作兩個獨立的8位元數目器/計時器，分別由TL0和TH0控制。TL0使用計時器0的控制（在TCON中）和狀態（在TMOD中）位：TR0， $C/\bar{T}0$ ，GATE0和TF0。TL0能用系統時鐘或外部輸入信號作為時鐘源。

TH0只能用作計時器功能，時鐘源來自系統時鐘。TH0由計時器T1的控制位TR1控制使能，溢出時計時器T1溢出標誌TF1置1，控制計時器T1中斷。

計時器0工作在方式3時，計時器1可以工作在方式0、1或2，但是不能置TF1標誌和產生中斷。TH1和TL1只能用作計時器功能，時鐘源來自系統時鐘，GATE1位元無效。T1輸入腳的上拉電阻也無效。計時器1由方式控制使能與否，因為TR1被計時器0佔用。計時器1在方式0、1或2時使能，在方式3時被關閉。

可配置寄存器TCON1中的 $TxX12(x = 0,1)$ 位元選擇系統時鐘或系統時鐘的1/12作為計時器 $Tx(x =$

0,1)的時鐘源。

## 9.2.2 計時器/計數器 $T_x(x = 0,1)$ 相關寄存器

### 9.2.2.1 計時器 $T_x(x = 0,1)$ 控制寄存器 TCON、TCON1

#### TCON

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R
復位值	0	0	0	0	0	0	0	0
位元符號	TF1	TR1	TF0	TR0	-			

位編號	位元符號	說明
7,5	TF <sub>x</sub> (x = 0,1)	T <sub>x</sub> (x = 0,1)的溢出標誌位元  0：中斷回應時硬體自動清 0，或軟體清 0  1：計數溢出時，硬體置 1
6,4	TR <sub>x</sub> (x = 0,1)	T <sub>x</sub> (x = 0,1)運行控制位  0：停止 T <sub>x</sub> 工作  1：啟動 T <sub>x</sub> 工作
3-0	-	保留位

#### TCON1

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R/W	R/W	R	R	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符	-		T1OUT	T1X12	-		T0OUT	T0X12

號						
---	--	--	--	--	--	--

位編號	位元符號	說明
7,6,3,2	-	保留位 ( 讀為 0 , 寫無效 )
5,1	TxOUT (x = 0,1)	Tx(x = 0,1)比較輸出功能允許位 0 : 禁止計時器 Tx 比較輸出功能 1 : 允許計時器 Tx 比較輸出功能
4,0	TxX12 (x = 0,1)	Tx(x = 0,1)計時器系統時鐘源分頻選擇位元 0 : Tx 計時器時鐘為 $F_{osc}/12$ 1 : Tx 計時器時鐘為 $F_{osc}$

### 9.2.2.2 計時器Tx(x = 0,1)工作方式寄存器 TMOD

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	GATE1	C/T1	M1[1:0]		GATE0	C/T0	M0[1:0]	

位編號	位元符號	說明
7,3	GATE <sub>x</sub> (x = 0,1)	<p>T<sub>x</sub>(x = 0,1)門控位</p> <p>0 : 只需軟體置 TR<sub>x</sub> 即可啟動 T<sub>x</sub></p> <p>1 : 只有在 INT<sub>x</sub> 埠電平為高電平時 TR<sub>x</sub> 置 1 , T<sub>x</sub> 才工作</p>
6,2	C/T <sub>x</sub> (x = 0,1)	<p>T<sub>x</sub>(x = 0,1)定時/計數功能選擇位</p> <p>0 : T<sub>x</sub> 用於內部定時</p> <p>1 : T<sub>x</sub> 用於外部計數</p>
5-4 1-0	M <sub>x</sub> [1:0] (x = 0,1)	<p>T<sub>x</sub>(x = 0,1)工作方式選擇位元</p> <p>00 : 方式 0 16 位重裝載計時器/計數器</p> <p>01 : 方式 1 16 位計時器/計數器</p> <p>10 : 方式 2 8 位自動重裝初值計時器/計數器</p> <p>11 : 方式 3 T0 分成兩個(TL0/TH0)獨立的 8 位計時器/計數器 ; T1 停止計數</p> <p>注意 : 方式 3 時 T0 佔用 T1 的 TR1、TF1 及中斷源,由於 TR1 被 T0 佔用 , 此時需要關閉 T1 , 可將 T1 設為工作方式 3。</p>

### 9.2.2.3 計時器 $T_x(x = 0,1)$ 資料寄存器 $TL_x(x = 0,1)$ 、 $TH_x(x = 0,1)$

#### $TL_x(x = 0,1)$

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	$TL_x[7:0] (x = 0,1)$							

位編號	位元符號	說明
7-0	$TL_x[7:0] (x = 0,1)$	$T_x(x = 0,1)$ 資料寄存器低位元組

#### $TH_x(x = 0,1)$

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	$TH_x[7:0] (x = 0,1)$							

位編號	位元符號	說明
7-0	$TH_x[7:0] (x = 0,1)$	$T_x(x = 0,1)$ 資料寄存器高位元組

## 9.3 計時器 3

計時器 3 是 16 位自動重載計時器，通過兩個資料寄存器  $TH3$  和  $TL3$  訪問，由  $T3CON$  寄存器控制。 $IE1$  寄存器的  $ET3$  位置 1 允許計時器 3 中斷（詳見错误!未找到引用源。章節）。

計時器 3 工作方式為 16 位元自動重載計數器/計時器。當  $TH3$  和  $TL3$  被寫入數值時，用作計時器

重載寄存器；當被讀時，被用做計數寄存器。TR3 位置 1 啟動計時器 3，且 T3 內部計數器開始遞增計數，在 0xFFFF 到 0x0000 時發生溢出，溢出會置 TF3 位為 1，同時將重載寄存器（即 TH3，TL3）的 16 位元資料重新載入 T3 內部計數器寄存器中。

在 TR3 為 1 時，對 TH3/TL3 的寫操作，不會影響 T3 內部計數器的值，只會改變重載寄存器的值，這個改變後的值在下一次溢出時會被重載進 T3 內部計數寄存器。只有 TR3 為 0 時，對 TH3/TL3 的寫操作會同時會改變 T3 內部計數寄存器和重載寄存器的值。

TH3 和 TL3 讀寫操作遵循以下順序：先高位後低位元。

當 T3CLKS[1:0] 為 01，T3 埠輸入外部時鐘，計時器 3 可以工作在普通模式或掉電模式。當 T3CLKS[1:0] 為 10，即計時器 3 的計數時鐘源為外部 32.768KHz 低頻晶振（外部低頻晶振在掉電模式下不會關閉）時，此時計時器 3 也可以工作在普通模式或掉電模式。

當 T3PD\_EN 為 1 且 T3CLKS[1:0] 為 01 或 10 時，T3 可工作在掉電模式下。當 T3 內部計數器從 0xFFFF 到 0x0000 時發生溢出時，會將晶片從掉電模式喚醒。如果中斷允許的話，喚醒後的晶片會進入計時器 3 中斷。

注意：在讀 TH3 和 TL3 時，要確保 TR3 = 0（當 TR=1 時，由於正在計數，讀出的 TH3 和 TL3 不准）。

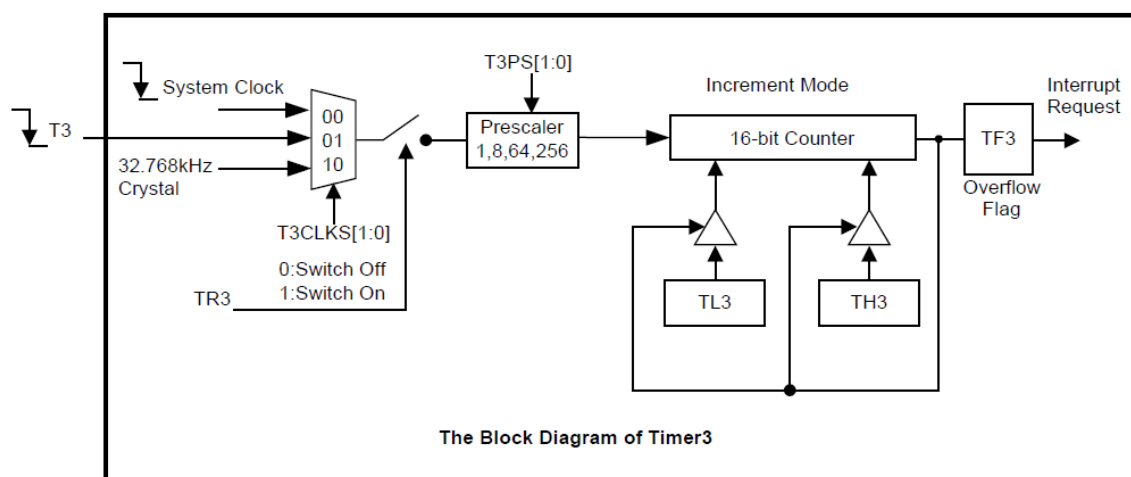


Figure 9-5 TIMER3 功能框圖





### 9.3.1 計時器/計數器 T3 相關寄存器

#### 9.3.1.1 計時器T3控制寄存器 T3CON

##### T3CON

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	TF3	T3PD_EN	T3PS[1:0]		-	TR3	T3CLKS[1:0]	

位編號	位元符號	說明
7	TF3	計時器 3 溢出標誌位元  0：無溢出（硬體清 0）軟體也可以清 0  1：溢出（硬體置 1）
6	T3PD_EN	計時器 3 在掉電模式下運行控制位元  0：掉電模式下禁止計時器 3 工作  1：掉電模式下允許計時器 3 工作，此時 T3CLKS[1:0]為 01 或 10  注意：外部晶振使能位 XTALEN 為 1，晶振選擇低頻晶振，而且配置掉電模式下允許計時器 3 工作，T3CLKS[1:0]為 10 時，如果 EA=1 且 ET3=1 時，計時器 3 溢出後會喚醒系統，並執行計時器 3 的中斷程式。
5-4	T3PS[1:0]	計時器 3 預分頻比選擇位  00：1/1  01：1/8  10：1/64

		11 : 1/256
3	-	保留位
2	TR3	計時器 3 允許控制位 0 : 禁止計時器 3 1 : 允許計時器 3
1-0	T3CLKS[1:0]	計時器 3 計數時鐘來源選擇位元 00 : 系統時鐘 $F_{osc}$ 01 : T3 埠輸入外部時鐘 10 : 外部低頻晶振 32.768KHz 11 : 內部低頻 RC ( RC44K )

### 9.3.1.2 計時器T3資料寄存器 TL3、TH3

#### TL3

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	TL3							

位編號	位元符號	說明
7-0	TL3	T3 資料寄存器低位元組

#### TH3

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

復位值	0	0	0	0	0	0	0	0
位元符號	TH3							
號								

位編號	位元符號	說明
7-0	TH3	T3 資料寄存器高位元組

## 9.4 計時器/計數器 4

計時器 4 是 16 位自動重載計時器。兩個資料寄存器 TH4 和 TL4 可作為一個 16 位寄存器來訪問，由 T4CON 寄存器控制。IE1 寄存器的 ET4 位置 1 允許計時器 4 中斷（詳見錯誤!未找到引用源。章節）。

當 TH4 和 TL4 資料寄存器被寫時，用作計時器重載寄存器；當被讀時，用做計數寄存器。TR4 位置 1 使計時器 4 內部 16 位元數目器開始遞增計數。內部 16 位元數目器在 0xFFFF 到 0x0000 溢出時置 TF4 位為 1。溢出同時，計時器重載寄存器的 16 位元資料重新載入內部 16 位元數目器中。

TH4 和 TL4 讀寫操作遵循以下順序：先高位後低位元。

### 9.4.1 計時器/計數器 T4 的工作方式

計時器 4 有三種工作方式：16 位元自動重載計時器，串口串列傳輸速率發生器和有 T4 邊沿觸發的 16 位自動重載計時器。這些方式通過 T4CON 寄存器的 T4M[1:0]設置。

#### 9.4.1.1 方式0: 16位自動重載計時器/計數器

TH4 寄存器存放 16 位元數目器/計時器高 8 位，TL4 存放低 8 位。TR4 為 0 時，按順序寫 TH4 和 TL4 兩個寄存器，寫的值同時被寫進重載寄存器和內部 16 位元數目器；TR4 置 1，內部 16 位元數目器開始從寫的值遞增計數，在 0xFFFF 到 0x0000 時發生溢出，此時 TF4 被置為 1。同時重載寄存器的 16 位元資料被自動重載入內部 16 位元數目器中，內部 16 位元數目器又開始從這個重載的數值遞增計

數，如果允許計時器 4 中斷則產生中斷。

在 TR4 為 1 時，對 TH4 和 TL4 的寫操作，不會影響內部 16 位元數目器的值，只能改變重載寄存器的值，這個改變後的值在下一次溢出時重載進內部 16 位元數目器中。只有 TR4 為 0 時，對 TH4 和 TL4 的寫操作會同時會改變內部 16 位元數目器和重載寄存器的值。

T4CON.0 寄存器的 T4CLKS 位元選擇時鐘源。當 T4CLKS = 1 時，計時器 4 的時鐘源為外部時鐘；當 T4CLKS = 0，計時器 4 的時鐘源為系統時鐘。

在比較方式中，需要軟體將 T4 埠設為輸出。計時器 4 從 TH4 和 TL4 預設值開始向 0xFFFF 計數，當計數器溢出時，T4 埠輸出電平翻轉，同時計時器 4 中斷標誌位元被置 1。在比較方式中，計時器 4 必須工作在定時方式 ( T4CLKS = 0 )。

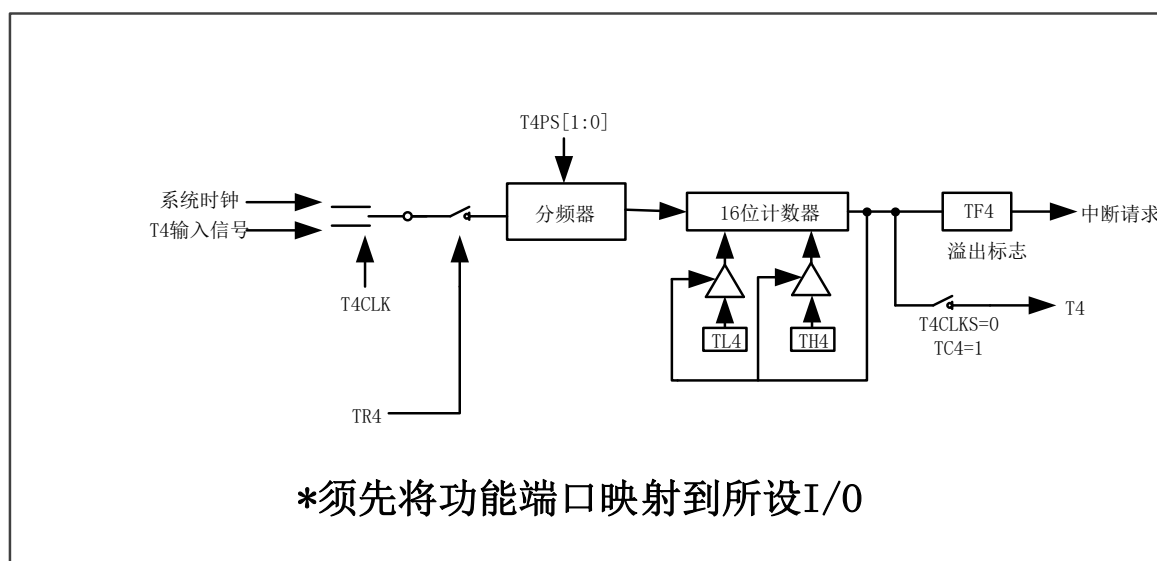


Figure 9-6 TIMER4 方式 0 功能框圖

#### 9.4.1.2 方式1: 串列傳輸速率發生器

通過設置 T4CON 寄存器中的 T4M[1:0]為 01 選擇計時器 4 作為串列傳輸速率發生器。該方式與自動重載方式相似。計時器 4 的溢出會引起軟體將計時器 4 重載寄存器中的 16 位值載入計時器 4 的內部 16 位元數目器中，此時也會產生溢出中斷。如果不希望中斷發生，可以關閉 ET4。串列傳輸速率由下列公式計算：

$$\text{BaudRate} = \frac{1}{16} \times \frac{f_{T4}/\text{PRESCALER}}{65536 - [\text{TH4}, \text{TL4}]}, \text{ 用計時器 4 作為串列傳輸速率發生器。}$$

上式中， $f_{T4}$  為計時器 4 計數所選時鐘源，PRESCALER 為計時器 4 預分頻比，TH4 和 TL4 為計時器 4 資料寄存器。

如果 TC4=1 時，可以將串列傳輸速率從 T4 口輸出，輸出波形的頻率為串列傳輸速率的 1/2，此時計時器 4 必須工作在定時方式 ( T4CLKS = 0 )。

#### 9.4.1.3 方式2/3: 帶邊沿觸發的16位自動重載計時器

計時器 4 在方式 2/3 為 16 位自動重載計時器。T4CON.0 寄存器的 T4CLKS 位一直為 0，計時器 4 只能選擇系統時鐘為時鐘源，其餘設置與方式 0 一致。

方式 2 中，當 TR4 位置 1 後，計時器 4 等待 T4 埠的觸發信號 ( 由 T4M[1:0]控制上升/下降沿 )，一個有效的觸發信號使計時器 4 開始運行。當計時器 4 內部 16 位元數目器從 0xFFFF 到 0x0000 溢出時，TF4 ( T4CON.7 ) 會被置起，如計時器 4 的中斷使能，將產生計時器 4 中斷。溢出同時，計時器重載寄存器的 16 位元資料被重新載入內部 16 位元數目器 TH4 和 TL4 中，計時器 4 將保持狀態並等待下一個觸發沿。

如果 TC4 = 0，在計時器 4 計數時，一個觸發信號不會中止內部 16 位元數目器的計數，內部 16 位元數目器溢出後進行重載並保持狀態，等待下一個有效觸發信號；

如果 TC4 = 1，在計時器 4 計數時，一個觸發信號會使重載寄存器的 16 位元資料被重新載入內部 16 位元數目器 TH4 和 TL4 中，並開始計數，但不會產生中斷，內部 16 位元數目器溢出後才會產生中斷。

TR4 置 1 不清計時器 4 的內部 16 位元數目器，在允許計時器之前應該把希望的初始化值寫入重載寄存器。

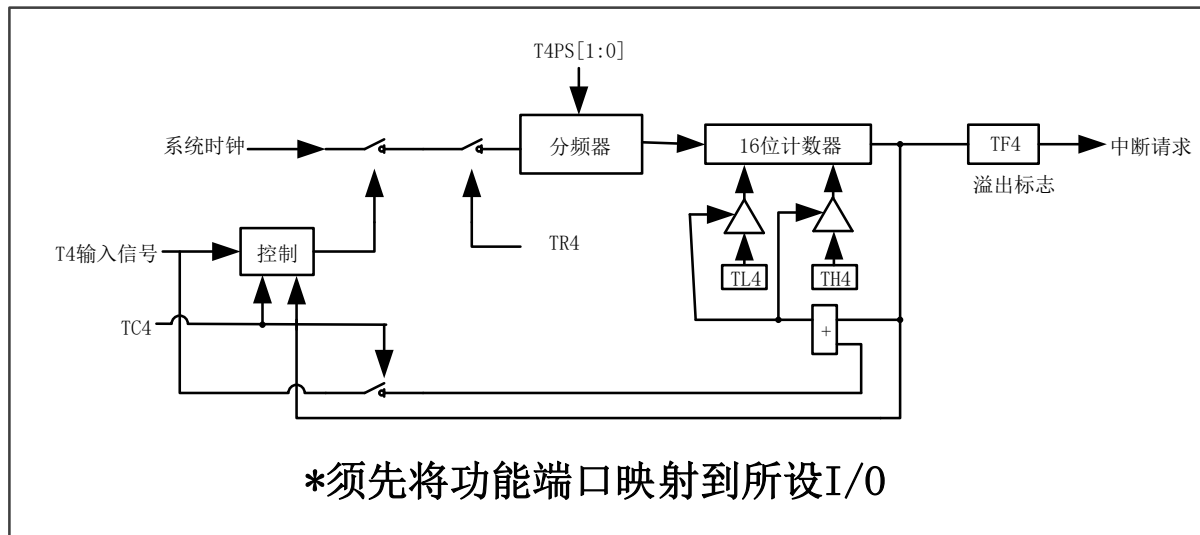


Figure 9-7 TIMER4 方式 2/3 功能框圖

注意：

(1) 當計時器 4 在串列傳輸速率發生器方式下作為計時器工作時 ( $TR4 = 1$ )，TH4 或 TL4 不能讀取或寫入。

因為計時器在每個狀態時間遞增，可能導致讀取或寫入的結果不精確。因此，在訪問 TH4/TL4 寄存器之前，計時器 4 必須被關閉 ( $TR4 = 0$ )。

(2) 當計時器 4 用作計數器時，T4 引腳的輸入信號頻率要小於系統時鐘的一半。

## 9.4.2 計時器/計數器 T4 相關寄存器

### 9.4.2.1 計時器T4控制寄存器 T4CON、T4CON1

#### T4CON

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	TF4	TC4	T4PS[1:0]		T4M[1:0]		TR4	T4CLKS

位編號	位元符號	說明
7	TF4	計時器 4 溢出標誌位元  0：無溢出（硬體清 0）軟體也可以清 0  1：溢出（硬體置 1）
6	TC4	比較功能允許位  當 T4M[1:0] = 00 或 01  0：禁止計時器 4 比較功能  1：允許計時器 4 比較功能  當 T4M[1:0] = 10 或 11  0：計時器 4 不能被再觸發  1：計時器 4 可以被再觸發
5-4	T4PS[1:0]	計時器 4 預分頻比（PRESCALER）選擇位  00：1/1  01：1/8

		10 : 1/64 11 : 1/256
3-2	T4M[1:0]	計時器 4 方式選擇位元 00 : Mode0 , 16 位自動重載計時器 01 : Mode1 , UART1 串列傳輸速率發生器 10 : Mode2 , T4 埠上升沿觸發 ( 只用系統時鐘 , T4CLKS 無效 ) 11 : Mode3 , T4 埠下降沿觸發 ( 只用系統時鐘 , T4CLKS 無效 )
1	TR4	計時器 4 允許控制位 0 : 禁止計時器 4 1 : 允許計時器 4
0	T4CLKS	計時器 4 計數時鐘來源選擇位元 0 : 系統時鐘 $F_{osc}$ 1 : T4 埠輸入外部時鐘 注 : 計時器 4 選擇後的計數時鐘源記作 : $f_{T4o}$



#### T4CON1

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R	R	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-						T4_PWMS	T4_PWM
號								

位編號	位元符號	說明
7-2	-	保留位
1	T4_PWMS	PWM 輸出極性選擇位 0：有效期間為高電平 1：有效期間為低電平 注：修改此控制位，立即生效，有效期間是指占空比期間
0	T4_PWM	計時器 4 當做 8 位的 PWM 0：正常的計時器功能 1：8 位 PWM 輸出，當 T4_PWM=1 時，需要配置 T4M=00，T4CLKS=0，TC4=0，T4_OUT 輸出 PWM 波形，其中 TH4 為週期寄存器，TL4 為占空比寄存器，TR4 用於啟動 PWM 的輸出

#### 9.4.2.2 計時器T4資料寄存器 TL4、TH4

##### TL4

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0

位元符號	TL4
------	-----

位編號	位元符號	說明
7-0	TL4	T4 資料寄存器低位元組

#### TH4

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	TH4							

位編號	位元符號	說明
7-0	TH4	T4 資料寄存器高位元組

## 9.5 計時器 5

計時器 5 是 16 位自動重載計時器。通過兩個資料寄存器 TH5 和 TL5 訪問，由 T5CON 寄存器控制。IEN1 寄存器的 ET5 位置 1 允許計時器 5 中斷（詳見错误!未找到引用源。章節）。

### 9.5.1 計時器 T5 的工作方式

#### 9.5.1.1 方式0: 16位自動重載計時器/計數器

計時器 5 在方式 0 時，TH5 寄存器存放 16 位計時器高 8 位，TL5 存放低 8 位。

當 EXEN5 = 0，16 位定時寄存器從 0xFFFF 到 0x0000 遞增，當溢出時，置起 TF5 位，同時計時器自動將使用者軟體寫好的寄存器 RCAP5H 和 RCAP5L 的 16 位值裝入 TH5 和 TL5 寄存器，如果允許計時器 5 中斷則產生中斷。

如果 EXEN5 = 1，溢出或在外部輸入 T5 上的下降沿/上升沿都能觸發一次 16 位重載，置起 EXF5 位。如果 ET5 被使能，TF5 和 EXF5 位都能產生中斷。

T5CON.1 寄存器的 TR5 位置 1 使能計時器 5，且不清計時器 5 的計數器。在允許計時器 5 之前，將希望的初始值寫入計時器重載寄存器中。

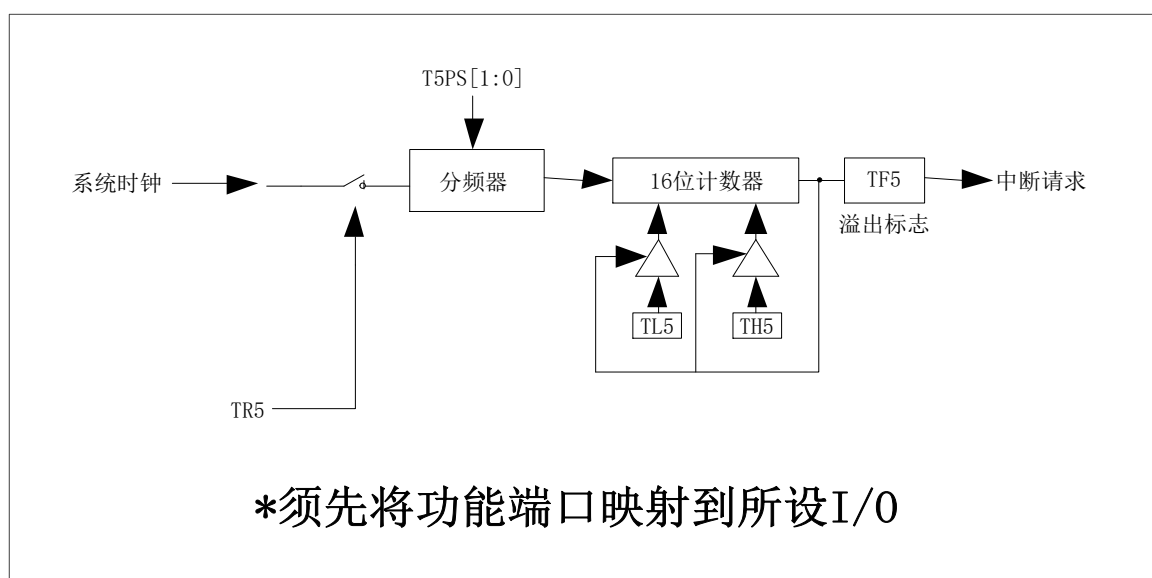


Figure 9-8 TIMER5 方式 0 功能框圖

### 9.5.1.2 方式1: 串列傳輸速率發生器

通過設置 T5CON 寄存器中的 T5M 為 01，選擇計時器 5 作為 UART1/UART2 的串列傳輸速率發生器或 IIC 的通信時鐘。該方式與自動重載方式相似。計時器 5 的溢出會將計時器 5 重載寄存器中的 16 位值載入計時器 5 的計數器中，但溢出不會產生中斷。此時如果 EXEN5=1，則 T5 引腳上的一個下降沿/上升沿會置位 EXF5，但不會引起重載。因此當計時器 5 作為串列傳輸速率發生器時，T5 引腳可作為一個額外的外部中斷。

UART1/UART2 的串列傳輸速率由下列公式計算：

$$\text{BaudRate} = \frac{1}{16} \times \frac{f_{T5}/\text{PRESCALER}}{65536 - [\text{RCAP5H}, \text{RCAP5L}]}, \text{ 用計時器 5 作為串列傳輸速率發生器。}$$

上式中， $f_{T5}$  為  $F_{\text{osc}}$ ，PRESCALER 為計時器 5 預分頻比，RCAP5H 和 RCAP5L 為計時器 5 的重載捕獲寄存器。

### 9.5.1.3 方式2: 16位上升沿捕獲

在捕獲方式中，T5CON 的 EXEN5 位元有兩個選項。

如果 EXEN5 = 0，計時器 T5 作為 16 位計時器，如果 ET5 被允許的話，計時器 T5 能設置 TF5 溢出生成一個中斷。溢出後計數不會被重載，從 0 開始計數。

如果 EXEN5 = 1，計時器 T5 執行相同操作，但是在外部輸入 T5 (可通過 T5CON1 來選擇捕獲輸入的類型) 的上升沿能引起 TH5 和 TL5 的當前值分別被捕獲到 RCAP5H 和 RCAP5L 中，此外，在 T5 上的上升沿也能引起在 T5CON 中的 EXF5 被設置。如果 ET5 被允許，EXF5 位也像 TF5 一樣也產生一個中斷。

### 9.5.1.4 方式3: 16位下降沿捕獲

在捕獲方式中，T5CON 的 EXEN5 位元有兩個選項。

如果 EXEN5 = 0，計時器 T5 作為 16 位計時器，如果 ET5 被允許的話，計時器 T5 溢出會產生一個中斷。溢出後計數不會被重載，從 0 開始計數。

如果 EXEN5 = 1，計時器 T5 執行相同操作，但是在外部輸入 T5 (可通過 T5CON1 來選擇捕獲輸入的類型) 的下降沿能引起 TH5 和 TL5 的當前值分別被捕獲到 RCAP5H 和 RCAP5L 中，此外，在 T5 上的下降沿也能引起在 T5CON 中的 EXF5 被設置。如果 ET5 被允許，EXF5 位也像 TF5 一樣也產生中斷。

## 9.5.2 計時器/計數器 T5 相關寄存器

### 9.5.2.1 計時器T5控制寄存器 T5CON、T5CON1

#### T5CON

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	TF5	EXF5	T5PS[1:0]		T5M[1:0]		TR5	EXEN5

位編號	位元符號	說明
7	TF5	計時器 5 溢出標誌位元 0：無溢出 1：溢出時硬體置 1，必須軟體清 0
6	EXF5	T5 引腳外部輸入事件發生標誌位元 0：無外部輸入事件發生，必須軟體清 0 1：檢測到外部輸入事件且 EXEN5= 1 時，硬體置 1，也做插斷要求標誌位元
5-4	T5PS[1:0]	計時器 5 預分頻比 (PRESCALER) 選擇位 00：1/1

		01 : 1/8  10 : 1/64  11 : 1/256
3-2	T5M[1:0]	計時器 5 方式選擇位元  00 : Mode0 , 16 位自動重載計時器  01 : Mode1 , UART2 串列傳輸速率發生器  10 : Mode2 , 16 位上升沿捕獲  11 : Mode3 , 16 位下降沿捕獲
1	TR5	計時器 5 允許控制位  0 : 禁止計時器 5  1 : 允許計時器 5
0	EXEN5	T5 引腳上的外部事件輸入用作重載/擷取觸發程式允許/禁止控制位  0 : 忽略 T5 引腳上的事件  1 : T5 引腳上一個下降沿或上升沿 , 產生一個捕獲或重載  注意 : 當捕獲內部低頻 RC 或 RXD 引腳時 , 也需要使能 EXEN5 , 同時還要配置上升沿捕獲或下降沿捕獲。

#### T5CON1

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R	R	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-						CAPM[1:0]	

位編號	位元符號	說明
7-2	-	保留
1-0	CAPM[1:0]	計時器 5 捕獲類型選擇位  00 : T5 腳的變化沿  01 : 內部低頻 RC，即看門狗的計數時鐘  10 : UART1 的 RXD1 腳  11 : UART2 的 RXD2 腳

### T5CON2

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	T5_PWMS	RL_SEL			T5_EXMOD		CMPCR	T5CAPCR

位編號	位元符號	說明
7	T5_PWMS	T5_PWM 輸出極性選擇位  0 : 有效期間為高電平  1 : 有效期間為低電平  注：修改此控制位，立即生效，有效期間是指占空比期間
6-4	RL_SEL	T5 重載功能選擇位  000 : 計時器 5 作為 16 位自動重載計時器時，T5 的計數溢出會產生一次重載。  001 : 計時器 5 作為 16 位自動重載計時器時，EXEN5=1 時，T5 引腳上

		<p>一個上升沿會產生一次重載。</p> <p>010：計時器 5 作為 16 位自動重載計時器時，EXEN5=1 時，T5 引腳上一個下降沿會產生一次重載。</p> <p>011：計時器 5 作為 16 位自動重載計時器時，EXEN5=1 時，T5 引腳上一個下降沿或上升沿會產生一次重載。</p> <p>100：計時器 5 作為 16 位自動重載計時器時，CAP0 上的一個事件會產生一次重載。</p> <p>101：計時器 5 作為 16 位自動重載計時器時，CAP1 上的一個事件會產生一次重載。</p>
3-2	T5_EXMOD	<p>T5 的擴展模式選擇</p> <p>00：由 T5M[1:0]決定</p> <p>01：上升沿或下降沿捕獲</p> <p>10：比較器模式</p> <p>11：PWM 模式</p> <p>在比較器模式下，RCAP5H 和 RCAP5L 預存待比較資料。由於計時器 5 向上計數，一旦 TH5 和 TL5 匹配 RCAP5H 和 RCAP5L 的設定值，TF5 將會由硬體置 1，用以標示發生了比較匹配事件。如果 CMPCR 置 1，當發生比較匹配事件後，計時器 5 計數器將硬體自動清 0。</p> <p>在 PWM 模式下，需要配置 T5M[1:0]=00 TH5=0 TL5=0</p> <p>{CAPH0，CAPL0}用於存放 PWM 的週期；</p> <p>{CAPH1，CAPL1}用於存放 PWM 的占空比；</p>
1	CMPCR	T5 比較匹配自動清除



		<p>該位僅當計時器 5 在比較模式下有效。當比較符合後，硬體將自動清除 TH5 及 TL5 計數器值。</p> <p>0：比較匹配之後，計時器 5 計數值按之前值繼續計數。</p> <p>1：比較匹配之後，計時器 5 計數值清 0。</p>
0	T5CAPCR	<p>T5 埠捕獲自動清除</p> <p>該位使能，當捕獲事件發生，在 TH5 和 TL5 內的資料移入捕獲寄存器後，硬體將自動清除 TH5 及 TL5 計數寄存器的值。注意：當該位置 1，如果有一個捕獲事件發生，僅清除 TH5、TL5 的值。</p> <p>0：T5 埠捕獲事件發生後計時器 5 計數值按之前計數值繼續累加</p> <p>1：T5 埠捕獲事件發生後計時器 5 計數值自動清 0</p>

### 9.5.2.2 計時器T5資料寄存器 TL5、TH5

#### TL5

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	TL5							

位編號	位元符號	說明
7-0	TL5	T5 資料寄存器低位元組

#### TH5

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

復位值	0	0	0	0	0	0	0	0
位元符號	TH5							

位編號	位元符號	說明
7-0	TH5	T5 資料寄存器高位元組

### 9.5.2.3 計時器T5重載捕獲寄存器 RCAP5L、RCAP5H

#### RCAP5L

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	RCAP5L							
位編號	位元符號	說明						
7-0	RCAP5L	T5 重載捕獲寄存器低位元組						

#### RCAP5H

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	RCAP5H							

位編號	位元符號	說明
-----	------	----

7-0	RCAP5H	T5 重載捕獲寄存器高位元組
-----	--------	----------------

#### 9.5.2.4 捕獲控制寄存器 CAPCON0、CAPCON1

##### CAPCON0

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	CAPEN1	CAPF1	CAP1LS[1:0]		CAPEN0	CAPF0	CAP0LS[1:0]	

位編號	位元符號	說明
7	CAPEN1	T5 輸入捕獲通道 1 使能控制位  1：使能 T5 輸入捕獲通道 1  0：禁止 T5 輸入捕獲通道 1
6	CAPF1	T5輸入捕獲通道1標誌位元  1：如果輸入捕獲通道 1 探測到邊沿觸發事件發生，該位元由硬體置位元  0：軟體清零。
5-4	CAP1LS[1:0]	T5輸入捕獲通道1捕獲條件選擇位  00：下降沿  01：上升沿  10：上升沿或下降沿  11：保留

3	CAPEN0	T5 輸入捕獲通道 0 使能控制位  1：使能 T5 輸入捕獲通道 0  0：禁止 T5 輸入捕獲通道 0
2	CAPF0	T5輸入捕獲通道0標誌位元  1;如果輸入捕獲通道 0 探測到邊沿觸發事件發生，該位元由硬體置位元  0:軟體清零。
1-0	CAP0LS[1:0]	T5輸入捕獲通道0捕獲條件選擇位  00：下降沿  01：上升沿  10：上升沿或下降沿  11：保留

#### CAPCON1

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-				CAP1_INTE	CAP0_INTE	CAP1CR	CAP0CR
號								

位編號	位元符號	說明
7-4	-	保留位
3	CAP1_INTE	CAP1 捕獲中斷使能位  0：中斷禁止

		1：中斷使能
2	CAP0_INTE	CAP0 捕獲中斷使能位 0：中斷禁止 1：中斷使能
1	CAP1CR	CAP1 埠捕獲自動清除 該位使能，當捕獲事件發生，在 TH5 和 TL5 內的資料移入捕獲寄存器後，硬體將自動清除 TH5 及 TL5 計數寄存器的值。 0：CAP1 埠捕獲事件發生後計時器 5 計數值按之前計數值繼續累加 1：CAP1 埠捕獲事件發生後計時器 5 計數值自動清 0
0	CAP0CR	CAP0 埠捕獲自動清除 該位使能，當捕獲事件發生，在 TH5 和 TL5 內的資料移入捕獲寄存器後，硬體將自動清除 TH5 及 TL5 計數寄存器的值。 0：CAP0 埠捕獲事件發生後計時器 5 計數值按之前計數值繼續累加 1：CAP0 埠捕獲事件發生後計時器 5 計數值自動清 0

將 CAP0 和 CAP1 的映射到一個腳上，然後設置 CAP0 進行上升沿捕獲，CAP1 下降沿捕獲，CAP0 配置捕獲後清零，CAP1 配置為捕獲不清零，這樣捕獲寄存器{CAPH0，CAPL0}可以捕獲得到 PWM 的週期，{CAPH1，CAPL1}寄存器可以得到占空比。

說明：捕獲使用的是計時器 5 的計數器，需要配置 T5\_MODE=00，T5M[1:0]=00，同時 RCAP5L 和 RCAP5H 都要配置為 0，（或者配置 RL\_SEL=111）

如果計時器 5 的中斷使能 ET5 打開，在計數器從 0 計數到 FFFF 也會產生溢出中斷。

CAP0 和 CAP1 的捕獲中斷不需要用到 IE1 寄存器中的 ET5

### 9.5.2.5 捕獲寄存器 CAPL0、CAPH0、CAPL1、CAPH1

#### CAPLn (n = 0~1)

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	CAPLn[7:0] (n = 0~1)							

位編號	位元符號	說明
7-0	CAPLn [7:0] (n = 0~1)	CAPn(n = 0~1)輸入捕獲寄存器低位元組

#### CAPHn (n = 0,1)

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	CAPHn[7:0] (n = 0~1)							

位編號	位元符號	說明
7-0	CAPHn [7:0] (n = 0~1)	CAPn(n = 0~1)輸入捕獲寄存器高位元組

## 9.6 滴答計時器 TICK

### 9.6.1 TICK 計時器特性

滴答計時器是 16 位自動重載遞減計時器。通過兩個資料寄存器 TICKH 和 TICKL 訪問，由 TICKCON 寄存器控制。

TICK 計時器的定時時間計算公式： $\{ \text{TICKHIGH}, \text{TICKLOW} \} / (\text{Fosc} / \text{分頻係數})$

### 9.6.2 TICK 計時器相關寄存器

#### 9.6.2.1 TICK控制寄存器TICKCON

TICKCON

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R	R/W	R/W	R	R	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	TICKIF	-	TICKIE	TICKEN	-	-	TICKPS	

位編號	位元符號	說明
7	TICKIF	TICK 計時器中斷標誌位元 0：軟體清 0 1：硬體置 1，在計數器減到 0 時才置 1
6	-	保留位
5	TICKIE	TICK 計時器中斷允許位 0：禁止 TICK 計時器中斷 1：允許 TICK 計時器中斷

4	TICKEN	TICK 計時器允許控制位  0：禁止 TICK 計時器  1：允許 TICK 計時器
3-2	-	保留位
1-0	TICKPS[1:0]	TICK 計時器的計數時鐘分頻  00：F <sub>osc</sub> 的 1 分頻  01：F <sub>osc</sub> 的 8 分頻  10：F <sub>osc</sub> 的 64 分頻  11：F <sub>osc</sub> 的 256 分頻

### 9.6.2.2 滴答計時器TICKL、TICKH

#### TICKL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	TICKLOW							

位編號	位元符號	說明
7-0	TICKLOW	滴答計時器的資料寄存器低位元組



**TICKH**

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	TICKHIGH							

位編號	位元符號	說明
7-0	TICKHIGH	滴答計時器的資料寄存器低位元組

## 10 脈寬調製PWM

### 10.1 PWM 特性

- 3 組帶死區互補 PWM 或 6 路占空比獨立的 PWM 輸出
- 12 位或 16 位 PWM
- 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

HC88L051F4 集成了三個 12/16 位元 PWM 模組 PWM0、PWM1 和 PWM2，三個模組各有一個計數器，PWM0 的計數器由 PWM0\_EN 來控制，只要使能 PWM0\_EN，計數器就會啟動，計數器的時鐘源通過 PWM0C 控制寄存器裡的 CK0 來選擇。

當需要從晶片管腳輸出 PWM 波形時，還需要使能 PWM0\_OEN 或 PWM01\_OEN，同時需要將埠設置為強推挽模式。如果不希望從晶片管腳上輸出 PWM 波形，可以不用使能 PWM0\_OEN 或 PWM01\_OEN，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 EFLT0 置 1，PWM0 輸出和其互補輸出可由 FLT0 引腳輸入信號變化自動關閉。一旦檢測到 FLT0 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 FLT0 引腳錯誤去除後繼續 PWM 輸出。在 FLT0 輸入信號有效期間，FLT0S 位無法清除。只有當 FLT0 輸入信號消失後，才能軟體清除 FLT0S 狀態位元，此時 PWM 恢復正常輸出。

PWM0 故障檢測埠 FLT0 ( P0.0 埠 ) 具有消抖功能，使用者可根據需要設置合適的消抖時間。配

置 FLT0 的消抖時間即配置埠 P0.0 的消抖時間，設置方法詳見埠消抖控制寄存器 P00DBC、P01DBC、P02DBC 章節。

三個 PWM 模組的功能及操作完全一樣，用戶可通過控制寄存器產生 3 路帶死區互補的 PWM 或 6 路單獨的 PWM 輸出。

3 個 PWM 模組共用一個中斷向量入口，但有各自的控制位元及標誌位元，方便使用者定時修改 PWM 模組下一次迴圈的週期或占空比。

## 10.2 PWM 輸出類型

HC88L051F4 的 PWM 輸出包含兩種類型：邊沿對齊和中心對齊。通過設置 TYP<sub>x</sub>(PWMCON0[7:5]) (x=0,1,2)來決定。

### 10.2.1 邊沿對齊模式

邊沿對齊模式下，模組產生邊沿對齊的 PWM 信號。PWM 輸出信號的週期由[PWM0PH:PWM0PL]決定，其占空比由相應的占空比寄存器（PWM 獨立輸出時，PWM01、PWM11、PWM21 的占空比寄存器是其死區時間寄存器）。

12/16 位元數目器採用單週期模式，從 0000H 開始向上計數，所有使能的 PWM 輸出在 PWM 週期開始時被驅動為有效狀態。當計數器的值與 PWM 占空比寄存器的值匹配時，PWM 輸出被驅動為無效狀態。計數器直到與[PWM0PH:PWM0PL]匹配，然後重新從 0000H 開始向上計數。

$$PWM_x \text{ 週期} = [PWM0PH : PWM0PL] * PWM0 \text{ 工作時鐘源週期}$$

$$PWM_x \text{ 占空比} = [PWMxDH : PWMxDL] * PWM0 \text{ 工作時鐘週期}$$

$$PWM_{x1} \text{ 占空比} = [PWMxDTH : PWMxDTL] * PWM0 \text{ 工作時鐘週期}$$

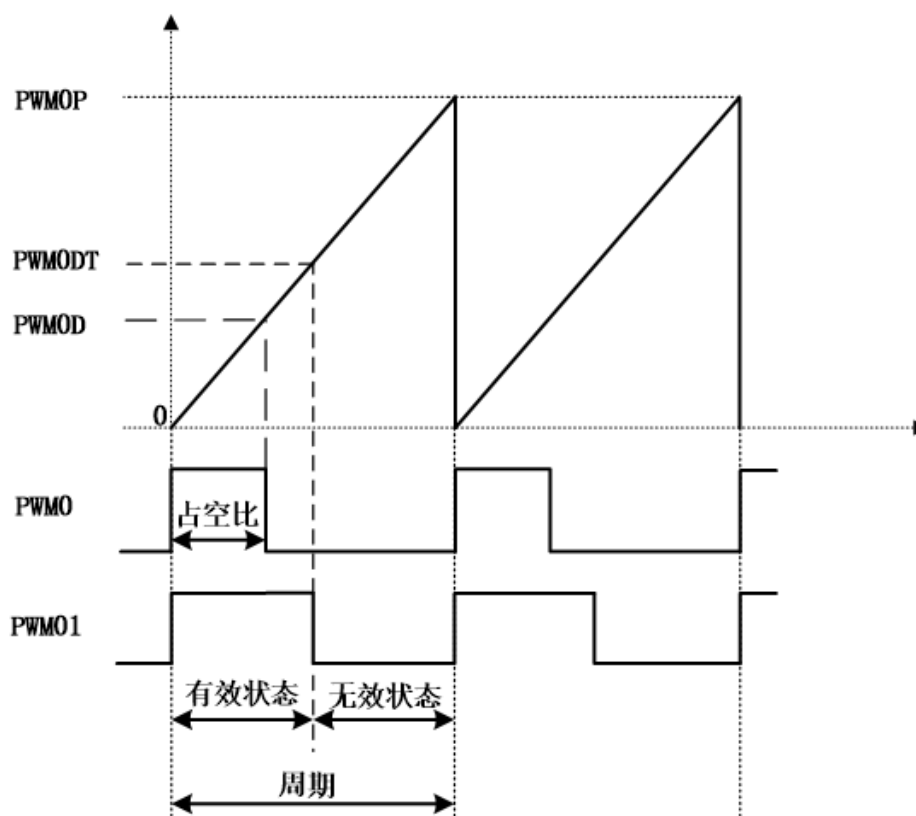


Figure 10-1 PWM 邊沿對齊波形

## 10.2.2 中心對齊模式

中心對齊模式下，模組產生中心對齊的 PWM 信號。PWM 輸出信號的週期由[PWM0PH:PWM0PL]決定，其占空比由相應的占空比寄存器（PWM 獨立輸出時，PWM01、PWM11、PWM21 的占空比寄存器是其死區時間寄存器）。

12/16 位元數目器採用雙週期模式，先從 0000H 開始向上計數至[PWM0PH:PWM0PL]，然後由[PWM0PH:PWM0PL]向下計數至 0000H，此為一個 PWM 的完整週期。所有使能的 PWM 輸出在 PWM 週期開始時被驅動為有效狀態。在向上計數的過程中，當計數器的值與 PWM 占空比寄存器的值匹配時，PWM 輸出被驅動為無效狀態，直到計數器轉換為向下計數且計數器值與 PWM 占空比寄存器的值匹配時，PWM 輸出才被驅動為有效狀態。

$$\text{PWMx週期} = [\text{PWM0PH:PWM0PL}] * \text{PWM0工作時鐘源週期} * 2$$

$PWM_x \text{ 占空比} = [PWM_{xDH}:PWM_{xDL}] * PWM_0 \text{ 工作時鐘週期} * 2$

$PWM_{x1} \text{ 占空比} = [PWM_{xDTH}:PWM_{xDTL}] * PWM_0 \text{ 工作時鐘週期} * 2$

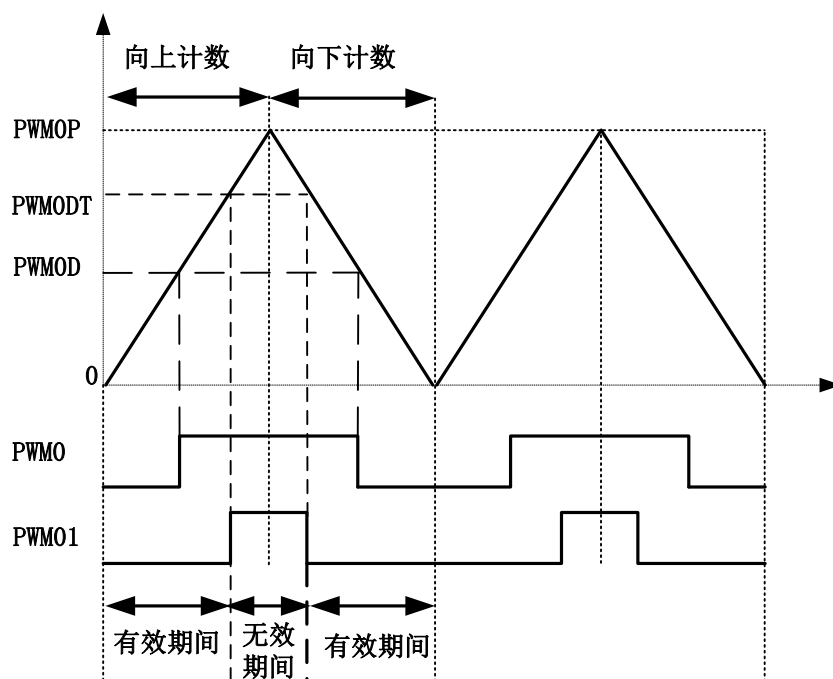


Figure 10-2 PWM 中心沿對齊波形

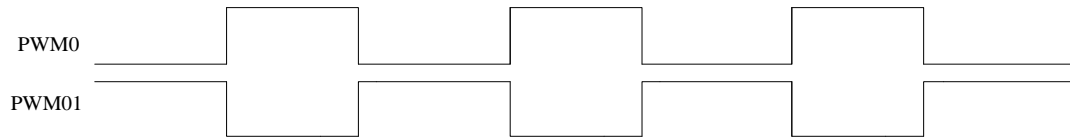
## 10.3 PWM 輸出模式

PWM 模組包含 3 個獨立的波形發生模組，對應的 3 對 PWM 輸出為 PWM0/PWM01、PWM1/PWM11、PWM2/PWM21，通過控制相關寄存器可使每對 PWM 輸出配置成互補輸出模式或獨立輸出模式。

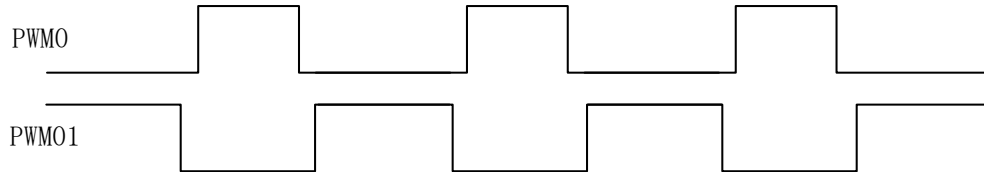
### 10.3.1 互補輸出模式

當  $PWM_{xM}(x=0,1,2)$  置 0：PWM 將工作在互補輸出模式，互補輸出模式時，通常使能  $PWM_x \& PWM_{x1}(x=0,1,2)$  輸出，此時可以控制對應的週期寄存器、占空比寄存器及死區時間寄存器，從而控制互補波形的輸出。互補輸出時可選擇  $PWM_x \& PWM_{x1}(x=0,1,2)$  輸出極性，方便用戶各種電平驅動需求。

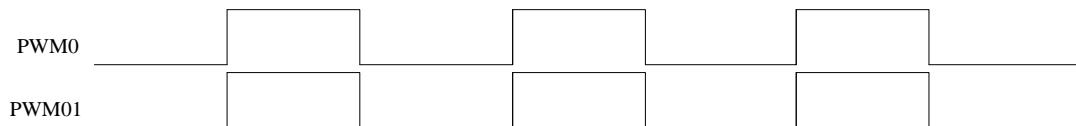
PWM0S=00& PWM0M=0 : PWM0和PWM01工作於互補模式且均為高有效



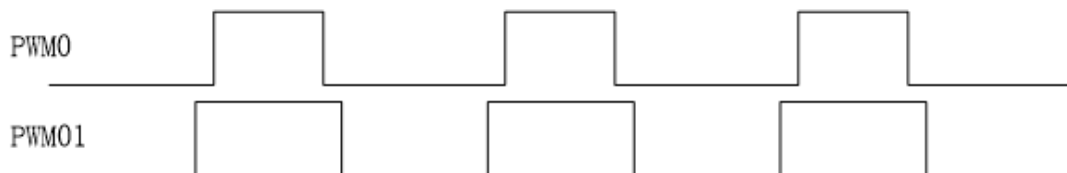
PWM0S=00& PWM0M=0 : PWM0 和 PWM01 工作於互補模式 (帶死區) 且均為高有效



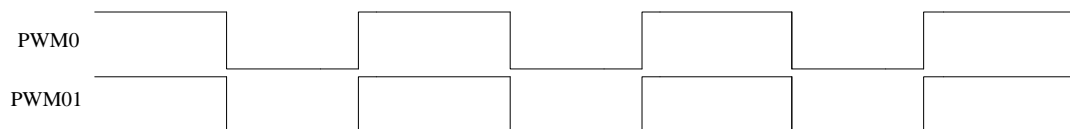
PWM0S=01& PWM0M=0 : PWM0和PWM01工作於互補模式且PWM0為高有效、PWM01為低有效



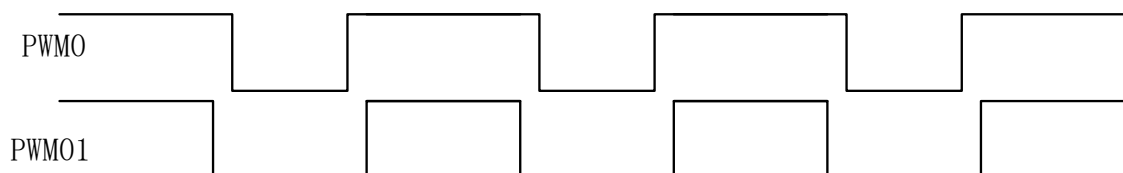
PWM0S=01& PWM0M=0 : PWM0 和 PWM01 工作於互補模式 (帶死區) 且 PWM0 為高有效、PWM01 為低有效



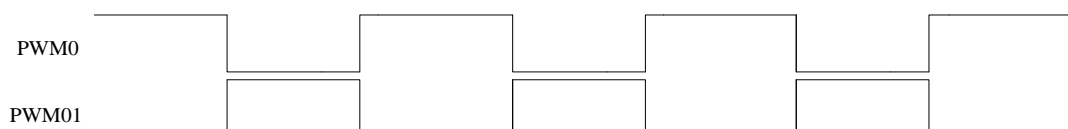
PWM0S=10& PWM0M=0 : PWM0和PWM01工作於互補模式且PWM0為低有效、PWM01為高有效



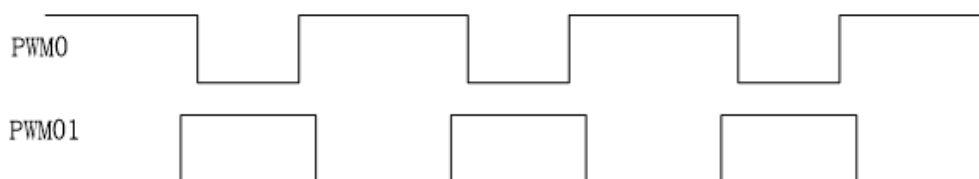
PWM0S=10& PWM0M=0 : PWM0 和 PWM01 工作於互補模式 (帶死區) 且 PWM0 為低有效、PWM01 為高有效



PWM0S=11 & PWM0M=0 : PWM0和PWM01工作於互補模式且PWM0和PWM01均為低有效



PWM0S=11 & PWM0M=0 : PWM0 和 PWM01 工作於互補模式 (帶死區) 且 PWM0 和 PWM01 均為低有效

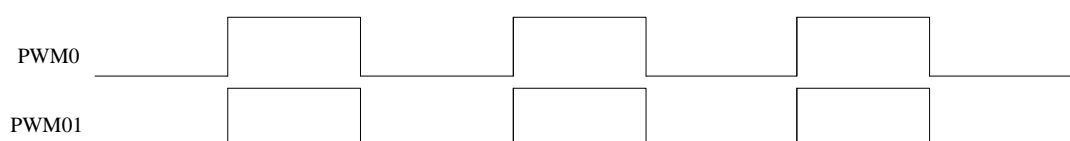


### 10.3.2 獨立輸出模式

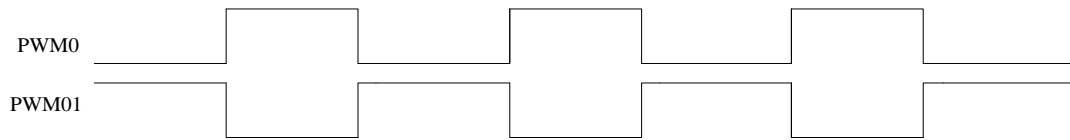
當 PWMxM 置 1 : PWM 將工作在獨立輸出模式，獨立輸出模式時，可以控制相關寄存器使能對應 PWM 埠單一輸出或同時輸出，同時讓 PWMx & PWMx1 輸出時，其週期相同但占空比可單獨設置。

當為獨立輸出模式時占空比寄存器將控制 PWMx 的占空比，死區時間控制寄存器將控制 PWMx1 的占空比，獨立輸出時也可控制 PWMx & PWMx1 輸出極性，方便用戶各種電平驅動需求。(x = 0,1,2)

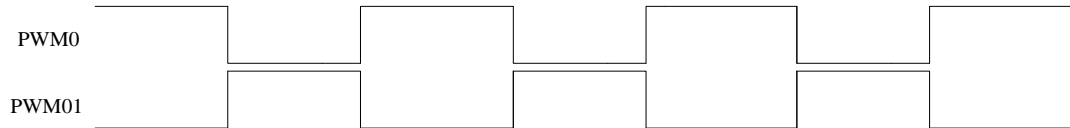
PWM0S=00 & PWM0M=1 : PWM0和PWM01工作於獨立模式且均為高有效



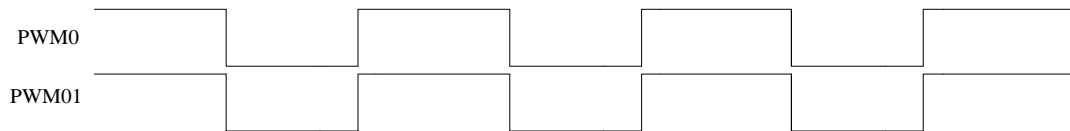
PWM0S=01 & PWM0M=1 : PWM0和PWM01工作於獨立模式且PWM0為高有效、PWM01為低有效



PWM0S=10& PWM0M=1：PWM0和PWM01工作於獨立模式且PWM0為低有效、PWM01為高有效



PWM0S=11& PWM0M=1：PWM0和PWM01工作於獨立模式且PWM0和PWM01均為低有效





## 10.4 PWM 相關寄存器

### 10.4.1 PWM 控制寄存器 PWMCON0、PWMCON1、PWMCON2

#### PWMCON0

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	1	1	1	0	0
位元符號	TYP2	TYP1	TYP0	RLOAD2	RLOAD1	RLOAD0	PWMLLEN	PENCTRL

位編號	位元符號	說明
7	TYP2	PWM2 輸出類型選擇位 0 : PWM2 邊沿對齊 1 : PWM2 中心對齊
6	TYP1	PWM1 輸出類型選擇位 0 : PWM1 邊沿對齊 1 : PWM1 中心對齊
5	TYP0	PWM0 輸出類型選擇位 0 : PWM0 邊沿對齊 1 : PWM0 中心對齊
4	RLOAD2	PWM2 自動重載使能位 0 : 禁止自動重載 1 : 使能自動重載

		<p>注：預設值為 1，預設狀態下修改參數後，參數自動載入，並在下一週期修改 PWM2 週期、占空比、死區。</p> <p>在修改參數前禁止自動重載，修改參數之後使能，可實現多組 PWM 間的同步，在禁止期間輸出之前狀態。</p> <p><b>適合在三組 PWM 週期相同，想修改 PWM 週期或占空比的情況，如果三組 PWM 的週期不同，那麼 pwm_ov 信號就不同</b></p>
3	RLOAD1	<p>PWM1 自動重載使能位</p> <p>0：禁止自動重載</p> <p>1：使能自動重載</p>
2	RLOAD0	<p>PWM0 自動重載使能位</p> <p>0：禁止自動重載</p> <p>1：使能自動重載</p>
1	PWMLEN	<p>PWM 精度選擇位</p> <p>0：12 位 PWM 計數</p> <p>1：16 位 PWM 計數</p> <p>注：當用戶選擇 12 位 PWM 計數時，PWMxPH、PWMxDH、PWMxDTH(x = 0,1,2)的高 4 位元為無效位元，客戶對高 4 位的操作不起作用。</p>
0	PENCTRL	<p>PWM0/1/2 模組使能控制位元</p> <p>0：由 PWM0_EN、PWM1_EN、PWM2_EN 控制</p> <p>1：由 PWMENA 寄存器控制 PWM0/1/2 的模組使能和輸出，且 PWM0_EN、PWM1_EN、PWM2_EN 位不會受此影響改變狀態。</p>

		<p>注：關閉時，PWM0 計數停止，輸出立即關閉。打開時，PWM0 計數器都重新從 1 開始計數，輸出受 PWM0_OEN 和 PWM01_OEN 控制。</p>
--	--	--

### PWMCON1

位編號	7	6	5	4	3	2	1	0
R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-	PWM0_CMP_INTF	PWM0_CMP_INTEN	PWM0_CMPEN	-	PWM2_PHASE	PWM1_PHASE	PWM0_PHASE

位編號	位元符號	說明
7	-	保留位
6	PWM0_CMP_INTF	<p>PWM0 計數器匹配標誌</p> <p>0：軟體清零</p> <p>1：當 PWM0 的計數器與 PWM0_CMP 一致時，置位</p>
5	PWM0_CMP_INTEN	<p>PWM0 計數器匹配中斷使能</p> <p>0：禁止</p> <p>1：使能</p>
4	PWM0_CMPEN	<p>PWM0 計數器匹配模式使能</p> <p>0：禁止</p> <p>1：使能</p>
3	-	保留位

2	PWM2_PHASE	<p>PWM2 移相控制位</p> <p>0：不移相</p> <p>1:PWM21 相對於 PWM2 移相 PWM2_PHASE_CNT 個 PWM CLK</p>
1	PWM1_PHASE	<p>PWM1 移相控制位</p> <p>0：不移相</p> <p>1:PWM11 相對於 PWM1 移相 PWM1_PHASE_CNT 個 PWM CLK</p>
0	PWM0_PHASE	<p>PWM0 移相控制位</p> <p>0：不移相</p> <p>1:PWM01 相對於 PWM0 移相 PWM0_PHASE_CNT 個 PWM CLK</p> <p>注：只有在獨立模式邊沿對齊時，移相才能起作用，移相的個數不能大於週期寄存器所配值。</p>

# PWMCON2

位編號	7	6	5	4	3	2	1	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號		FLT_IO_SEL			DBG_STBEN	FLT_CTRL_MODE		FLT_INT_EN

位編號	位元符號	說明
7	-	保留位
6-4	FLT_IO_SEL	<p>FLT 的埠選擇</p> <p>000 : FLT0 固定在 P0.0 , FLT1 固定在 P0.1 , FLT2 固定在 P0.2</p> <p>001 : FLT0/1/2 固定在 P0.0</p> <p>010 : FLT0/1/2 固定在 P0.1</p> <p>011 : FLT0/1/2 固定在 P0.2</p> <p>100 : FLT0/1/2 固定來自 ADC 模擬看門狗</p> <p>101 : FLT0/1/2 固定來自 LVD 的比較器</p> <p>110 : FLT0 使用計時器 0 的引腳映射寄存器 , FLT1 使用計時器 1 的映射寄存器 , FLT2 使用計時器 3 的映射寄存器</p> <p>其他值 : 保留</p>
3	DBG_STBEN	<p>在模擬模式 PWM 停止使能</p> <p>0 : 模擬模式不停止 PWM 輸出</p> <p>1 : 模擬模式下停止 PWM 輸出 , 全速時會恢復輸出</p>
2-1	FLT_CTRL_MODE	FTL 有效信號撤銷之後 , PWM 的輸出選擇

		<p>00：自動恢復以前的輸出</p> <p>01：在當前 PWM 週期結束後恢復以前的輸出，</p> <p>10：不能恢復以前的輸出，需要重新配置 PWM 使能位</p> <p>11：保留</p>
0	FLT_INT_EN	<p>FLT 中斷使能位</p> <p>0：不使能</p> <p>1：使能</p> <p>FLT 單獨一個中斷向量。</p>

## 10.4.2 PWM 使能寄存器 PWMENA

### PWMENA

位編號	7	6	5	4	3	2	1	0
R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-	PWM21EN	PWM11EN	PWM01EN	-	PWM2EN	PWM1EN	PWM0EN

位編號	位元符號	說明
7	-	保留位
6	PWM21EN	PWM21 輸出控制位 0：禁止 PWM21 輸出 1：允許 PWM21 輸出
5	PWM11EN	PWM21 輸出控制位 0：禁止 PWM11 輸出 1：允許 PWM11 輸出
4	PWM01EN	PWM21 輸出控制位 0：禁止 PWM01 輸出 1：允許 PWM01 輸出
3	-	保留位
2	PWM2EN	PWM2 輸出控制位 0：禁止 PWM2 輸出 1：允許 PWM2 輸出

1	PWM1EN	<p>PWM0 輸出控制位</p> <p>0：禁止 PWM1 輸出</p> <p>1：允許 PWM1 輸出</p>
0	PWM0EN	<p>PWM21 輸出控制位</p> <p>0：禁止 PWM0 輸出</p> <p>1：允許 PWM0輸出</p> <p>注：PWM允許輸出(輸出時對應埠必須設為輸出模式)，必須在PWM管腳映射管腳，否則為PWM輸出關閉狀態(輸出電平狀態與輸出模式選擇位元有關)；即使都禁止輸出，只要相關位被使能，PWM都可以溢出中斷，即此時PWM可以作為計時器使用，此控制位修改立即生效。</p>



### 10.4.3 PWM0 模組

#### 10.4.3.1 PWM0使能寄存器PWM0EN

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	FLT0_INTF	FLT0_MODE		EFLT0	PWM0M	PWM01_OEN	PWM0_OEN	PWM0_EN

位編號	位元符號	說明
7	FLT0_INTF	<p>PWM0 故障中斷標誌位元</p> <p>0：軟體清 0。</p> <p>1：當檢測到有效故障時，此位置 1。</p>
6-5	FLT0_MODE	<p>PWM0 故障輸出預定狀態選擇位元</p> <p>00：PWM0&amp;PWM01故障期間均為低電平</p> <p>01：PWM0故障期間低電平，PWM01故障期間高電平</p> <p>10：PWM0故障期間高電平，PWM01故障期間低電平</p> <p>11：PWM0&amp;PWM01故障期間均為高電平</p>
4	EFLT0	<p>PWM0 FLT0 控制引腳使能位</p> <p>0：禁止故障檢測，GPIO 功能或其它功能</p> <p>1：允許故障檢測，PWM0 故障檢測輸入引腳</p> <p>注意：互補輸出模式及獨立輸出模式都可受故障檢測腳控制。</p>
3	PWM0M	PWM0 工作模式選擇位元

		<p>0 : PWM0&amp;PWM01 工作於互補輸出模式</p> <p>1 : PWM0&amp;PWM01 工作於獨立輸出模式</p> <p>注意：修改 PWM0 工作模式時建議先關閉 PWM0 模組。</p>
2	PWM01_OEN	<p>PWM01 輸出控制位</p> <p>0 : 禁止 PWM01 輸出</p> <p>1 : 允許 PWM01 輸出</p>
1	PWM0_OEN	<p>PWM0 輸出控制位</p> <p>0 : 禁止 PWM0 輸出</p> <p>1 : 允許 PWM0 輸出</p> <p>注意：PWM0 允許輸出，必須在 PWM0_EN 置 1 下才有效，否則為 PWM0 輸出關閉狀態（輸出時對應埠必須設為輸出模式）；即使都禁止輸出，只要 PWM0_EN 位被使能，PWM0 都可以溢出中斷，即此時 PWM0 可以作為計時器使用，此控制位修改立即生效。</p>
0	PWM0_EN	<p>PWM0 模組使能控制位元</p> <p>0 : 關閉 PWM0 模組</p> <p>1 : 打開 PWM0 模組（重新計數）</p> <p>注意：關閉時，PWM0 計數停止，輸出立即關閉。打開時，PWM0 計數器都重新從 1 開始計數，輸出受 PWM0_OEN 和 PWM01_OEN 控制。</p>

#### 10.4.3.2 PWM0控制寄存器PWM0C

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

復位值	0	0	0	0	0	0	0	0
位元符號	PWM0IE	PWM0IF	FLT0S	FLT0C	PWM0S	CK0		

位編號	位元符號	說明
7	PWM0IE	PWM0 中斷允許位 0：禁止 PWM0 中斷 1：允許 PWM0 中斷
6	PWM0IF	PWM0 中斷標誌位元 0：軟體清 0 1：PWM0 週期計數器溢出，由硬體置 1
5	FLT0S	PWM0 FLT 狀態位元 0：PWM 正常狀態，軟體清 0 1：PWM 輸出關閉，硬體置 1
4	FLT0C	PWM0 FLT 引腳配置位元 0：FLT0 為低電平時，PWM 輸出關閉 1：FLT0 為高電平時，PWM 輸出關閉
3-2	PWM0S	PWM0 和 PWM01 輸出模式選擇位元 00：PWM0和PWM01均為高有效 01：PWM0為高有效，PWM01為低有效 10：PWM0為低有效，PWM01為高有效 11：PWM0 和 PWM01 均為低有效

		<p>注意：對於獨立模式，輸出模式選擇位元同樣有效，但與互補模式不同的是：有效期間為占空比期間；而互補模式中對於 PWM0 的有效期間為占空比期間，PWM01 的有效期間為占空比的互補期間。</p>
1-0	CK0	<p>PWM0 時鐘源選擇位元</p> <p>00 : <math>F_{osc}/1</math></p> <p>01 : <math>F_{osc}/8</math></p> <p>10 : <math>F_{osc}/32</math></p> <p>11 : <math>F_{osc}/128</math></p>

### 10.4.3.3 PWM0週期寄存器PWM0PL、PWM0PH

#### PWM0PL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM0PL[7:0]							

位編號	位元符號	說明
7-0	PWM0PL[7:0]	PWM0 週期寄存器低 8 位

#### PWM0PH

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-	-	-	-	PWM0PH[3:0]			

位編號	位元符號	說明
7-4	-	保留位 ( 讀為 0，寫無效 )
3-0	PWM0PH[3:0]	PWM0 週期寄存器高 4 位

注意：修改PWM0週期時先修改高位，後修改低位元，讀時不受限制，例如

- (1) PWM0PH = 0x05;
- (2) PWM0PL = 0x08; //此時 PWM 計數器溢出，則下一個週期開始週期計算資料為 0x0508
- (3) PWM0PH = 0x06; //此時 PWM 計數器溢出，則下一個週期開始週期計算資料為 0x0508
- (4) PWM0PL = 0x08; //此時 PWM 計數器溢出，則下一個週期開始週期計算資料為 0x0608

(5)  $PWM0PL = 0x09$ ; //此時 PWM 計數器溢出，則下一個週期開始週期計算資料為  $0x0609$

顯然只要修改 PWM 週期，無論低位元寄存器是否需要修改，低位元都必須寫入一次，且週期修改都只會在下一個 PWM 週期才會生效。

$PWM0週期 = [PWM0PH : PWM0PL] * PWM0工作時鐘源週期$

### 10.4.3.4 PWM0占空比寄存器PWM0DL、PWM0DH

#### PWM0DL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM0DL[7:0]							

位編號	位元符號	說明
7-0	PWM0DL[7:0]	PWM0 占空比寄存器低 8 位

#### PWM0DH

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-	-	-	-	PWM0DH[3:0]			

位編號	位元符號	說明
7-4	-	保留位 ( 讀為 0 , 寫無效 )
3-0	PWM0DH[3:0]	PWM0 占空比寄存器高 4 位

注意：修改 PWM0 占空比寄存器，操作類似修改 PWM0 週期寄存器，都是必須先修改高位後修改低位元，且修改都在下一個週期才有效。

PWM0 占空比 = [ PWM0DH : PWM0DL ] \* PWM0 工作時鐘週期

### 10.4.3.5 PWM0死區時間寄存器PWM0DTL、PWM0DTH

#### PWM0DTL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM0DTL[7:0]							

位編號	位元符號	說明
7-0	PWM0DTL[7:0]	PWM0 死區時間寄存器低 8 位

### PWM0DTH

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-	-	-	-	PWM0DTH[3:0]			

位編號	位元符號	說明
7-4	-	保留位 ( 讀為 0 , 寫無效 )
3-0	PWM0DTH[3:0]	PWM0 死區時間寄存器高 4 位

當 PWM0M=1 時, PWM0 工作在 2 路獨立模式, 此時的死區時間寄存器被用來當做 PWM01 的占空比寄存器, 即獨立模式的 PWM0 可以產生 2 路週期相同, 但占空比可以不同的 PWM 波形。

互補模式下: PWM0 死區時間 = [ PWM0DTH : PWM0DTL ] \* PWM0 工作時鐘週期。

互補模式下: 死區時間必須小於占空比時間, 死區時間與占空比時間的和必須小於 PWM0 週期。

獨立模式下: PWM01 占空比時間 = [ PWM0DTH : PWM0DTL ] \* PWM0 工作時鐘週期。



### 10.4.3.6 PWM0移相個數寄存器PWM0PHASEH、PWM0PHASEL

#### PWM0PHASEH

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM0PHASE[15:8]							

位編號	位元符號	說明
7-0	PWM0PHASE[15:8]	PWM0 移相個數寄存器高 8 位

#### PWM0PHASEL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM0PHASE [7:0]							

位編號	位元符號	說明
7-0	PWM0PHASE [7:0]	PWM0 移相個數寄存器低 8 位

### 10.4.3.7 PWM0計數匹配寄存器PWM0CMPH、PWM0CMPL

#### PWM0CMPH

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0

位元符號	PWM0CMP[15:8]
------	---------------

位編號	位元符號	說明
7-0	PWM0CMP[15:8]	PWM0 計數匹配寄存器高 8 位

#### PWM0CMPL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM0CMP[7:0]							

位編號	位元符號	說明
7-0	PWM0CMP[7:0]	PWM0 計數匹配寄存器低 8 位

注：當 PWM0 的計數值與 PWM0\_CMP 相匹配時，拉高 PWM0 計數器匹配標誌

#### 10.4.3.8 PWM0中斷分頻數PWM0INTDIV

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM0_CMP_INT_DIV				PWM0_OV_INT_DIV			

位編號	位元符號	說明
7-4	PWM0_CMP_INT_DIV	PWM0 匹配中斷分頻

3-0	PWM0_OV_INT_DIV	PWM0 溢出中斷分頻
-----	-----------------	-------------

## 10.4.4 PWM1 模組

### 10.4.4.1 PWM1使能寄存器PWM1EN

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	FLT1_INTF	FLT1_MODE		EFLT1	PWM1M	PWM11_OEN	PWM1_OEN	PWM1_EN

位編號	位元符號	說明
7	FLT1_INTF	PWM1 故障中斷標誌位元  0：軟體清 0。  1：當檢測到有效故障時，此位置 1。
6-5	FLT1_MODE	PWM1 故障輸出預定狀態選擇位元  00：PWM1&PWM11故障期間均為低電平  01：PWM1故障期間低電平，PWM11故障期間高電平  10：PWM1故障期間高電平，PWM11故障期間低電平  11：PWM1&PWM11故障期間均為高電平
4	EFLT1	PWM1 FLT1 控制引腳使能位  0：禁止故障檢測，GPIO 功能或其它功能  1：允許故障檢測，PWM1 故障檢測輸入引腳  注意：互補輸出模式及獨立輸出模式都可受故障檢測腳控制。
3	PWM1M	PWM1 工作模式選擇位元

		<p>0 : PWM1&amp;PWM11 工作於互補輸出模式</p> <p>1 : PWM1&amp;PWM11 工作於獨立輸出模式</p> <p>注意：修改 PWM1 工作模式時建議先關閉 PWM1 模組。</p>
2	PWM11_OEN	<p>PWM11 輸出控制位</p> <p>0 : 禁止 PWM11 輸出</p> <p>1 : 允許 PWM11 輸出</p>
1	PWM1_OEN	<p>PWM1 輸出控制位</p> <p>0 : 禁止 PWM1 輸出</p> <p>1 : 允許 PWM1 輸出</p> <p>注意：PWM1 允許輸出，必須在 PWM1_EN 置 1 下才有效，否則為 PWM1 輸出關閉狀態（輸出時對應埠必須設為輸出模式）；即使都禁止輸出，只要 PWM1_EN 位被使能，PWM1 都可以溢出中斷，即此時 PWM1 可以作為計時器使用，此控制位修改立即生效。</p>
0	PWM1_EN	<p>PWM1 模組使能控制位元</p> <p>0 : 關閉 PWM1 模組</p> <p>1 : 打開 PWM1 模組（重新計數）</p> <p>注意：關閉時，PWM1 計數停止，輸出立即關閉。打開時，PWM1 計數器都重新從 1 開始計數，輸出受 PWM1_OEN 和 PWM11_OEN 控制。</p>

#### 10.4.4.2 PWM1控制寄存器PWM1C

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

復位值	0	0	0	0	0	0	0	0
位元符號	PWM1IE	PWM1IF	FLT1S	FLT1C	PWM1S	CK1		

位編號	位元符號	說明
7	PWM1IE	PWM1 中斷允許位 0：禁止 PWM1 中斷 1：允許 PWM1 中斷
6	PWM1IF	PWM1 中斷標誌位元 0：軟體清 0 1：PWM1 週期計數器溢出，由硬體置 1
5	FLT1S	PWM1 FLT 狀態位元 0：PWM 正常狀態，軟體清 0 1：PWM 輸出關閉，硬體置 1
4	FLT1C	PWM1 FLT 引腳配置位元 0：FLT1 為低電平時，PWM 輸出關閉 1：FLT1 為高電平時，PWM 輸出關閉
3-2	PWM1S	PWM1 和 PWM11 輸出模式選擇位元 00：PWM1和PWM11均為高有效 01：PWM1為高有效，PWM11為低有效 10：PWM1為低有效，PWM11為高有效 11：PWM1 和 PWM11 均為低有效

		注意：對於獨立模式，輸出模式選擇位元同樣有效，但與互補模式不同的是：有效期間為占空比期間；而互補模式中對於 PWM1 的有效期間為占空比期間，PWM11 的有效期間為占空比的互補期間。
1-0	CK1	PWM1 時鐘源選擇位元  00 : $F_{osc}/1$  01 : $F_{osc}/8$  10 : $F_{osc}/32$  11 : $F_{osc}/128$

#### 10.4.4.3 PWM1週期寄存器PWM1PL、PWM1PH

##### PWM1PL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM1PL[7:0]							

位編號	位元符號	說明
7-0	PWM1PL[7:0]	PWM1 週期寄存器低 8 位

##### PWM1PH

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符	-	-	-	-	PWM1PH[3:0]			

號					
---	--	--	--	--	--

位編號	位元符號	說明
7-4	-	保留位 ( 讀為 0 , 寫無效 )
3-0	PWM1PH[3:0]	PWM1 週期寄存器高 4 位

注意：修改PWM1週期時先修改高位，後修改低位元，讀時不受限制，例如

- (1) PWM1PH = 0x05;
- (2) PWM1PL = 0x08; //此時 PWM 計數器溢出，則下一個週期開始週期計算資料為 0x0508
- (3) PWM1PH = 0x06; //此時 PWM 計數器溢出，則下一個週期開始週期計算資料為 0x0508
- (4) PWM1PL = 0x08; //此時 PWM 計數器溢出，則下一個週期開始週期計算資料為 0x0608
- (5) PWM1PL = 0x09; //此時 PWM 計數器溢出，則下一個週期開始週期計算資料為 0x0609

顯然只要修改 PWM 週期，無論低位元寄存器是否需要修改，低位元都必須寫入一次，且週期修改都只會在下一個 PWM 週期才會生效。

$$\text{PWM1週期} = [\text{PWM1PH} : \text{PWM1PL}] * \text{PWM1工作時鐘源週期}$$

#### 10.4.4.4 PWM1占空比寄存器PWM1DL、PWM1DH

##### PWM1DL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM1DL[7:0]							
號								

位編號	位元符號	說明
7-0	PWM1DL[7:0]	PWM1 占空比寄存器低 8 位



### PWM1DH

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-	-	-	-	PWM1DH[3:0]			

位編號	位元符號	說明
7-4	-	保留位 ( 讀為 0 , 寫無效 )
3-0	PWM1DH[3:0]	PWM1 占空比寄存器高 4 位

注意：修改 PWM1 占空比寄存器，操作類似修改 PWM1 週期寄存器，都是必須先修改高位後修改低位元，且修改都在下一個週期才有效。

$$\text{PWM1 占空比} = [\text{PWM1DH} : \text{PWM1DL}] * \text{PWM1 工作時鐘週期}$$

### 10.4.4.5 PWM1死區時間寄存器PWM1DTL、PWM1DTH

#### PWM1DTL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM1DTL[7:0]							

位編號	位元符號	說明
7-0	PWM1DTL[7:0]	PWM1 死區時間寄存器低 8 位

### PWM1DTH

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-				PWM1DTH[3:0]			
號								

位編號	位元符號	說明
7-4	-	保留位 ( 讀為 0 , 寫無效 )
3-0	PWM1DTH[3:0]	PWM1 死區時間寄存器高 4 位

當 PWM1M=1 時, PWM1 工作在 2 路獨立模式, 此時的死區時間寄存器被用來當做 PWM11 的占空比寄存器, 即獨立模式的 PWM1 可以產生 2 路週期相同, 但占空比可以不同的 PWM 波形。

互補模式下: PWM1 死區時間 = [ PWM1DTH : PWM1DTL ] \* PWM1 工作時鐘週期;

互補模式下: 死區時間必須小於占空比時間, 死區時間與占空比時間的和必須小於 PWM1 週期;

獨立模式下: PWM11 占空比時間 = [ PWM1DTH : PWM1DTL ] \* PWM1 工作時鐘週期;

#### 10.4.4.6 PWM1移相個數寄存器PWM1PHASEH、PWM1PHASEL

##### PWM1PHASEH

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM1PHASE[15:8]							

位編號	位元符號	說明
7-0	PWM1PHASE[15:8]	PWM1 移相個數寄存器高 8 位

##### PWM0PHASEL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM0PHASE [7:0]							

位編號	位元符號	說明
7-0	PWM0PHASE [7:0]	PWM0 移相個數寄存器低 8 位

#### 10.4.4.7 PWM0中斷分頻數PWM0INTDIV

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號					PWM0_OV_INT_DIV			

號		
---	--	--

位編號	位元符號	說明
7-4	-	保留位
3-0	PWM0_OV_INT_DIV	PWM0 溢出中斷分頻

## 10.4.5 PWM2 模組

### 10.4.5.1 PWM2使能寄存器PWM2EN

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	FLT2_INTF	FLT2_MODE		EFLT2	PWM2M	PWM21_OEN	PWM2_OEN	PWM2_EN

位編號	位元符號	說明
7	FLT2_INTF	<p>PWM2 故障中斷標誌位元</p> <p>0：軟體清 0。</p> <p>1：當檢測到有效故障時，此位置 1。</p>
6-5	FLT2_MODE	<p>PWM2 故障輸出預定狀態選擇位元</p> <p>00：PWM2&amp;PWM21故障期間均為低電平</p> <p>01：PWM2故障期間低電平，PWM21故障期間高電平</p> <p>10：PWM2故障期間高電平，PWM21故障期間低電平</p> <p>11：PWM2&amp;PWM21故障期間均為高電平</p>
4	EFLT2	<p>PWM2 FLT2 控制引腳使能位</p> <p>0：禁止故障檢測，GPIO 功能或其它功能</p> <p>1：允許故障檢測，PWM2 故障檢測輸入引腳</p> <p>注意：互補輸出模式及獨立輸出模式都可受故障檢測腳控制。</p>
3	PWM2M	PWM2 工作模式選擇位元

		<p>0 : PWM2&amp;PWM21 工作於互補輸出模式</p> <p>1 : PWM2&amp;PWM21 工作於獨立輸出模式</p> <p>注意：修改 PWM2 工作模式時建議先關閉 PWM2 模組。</p>
2	PWM21_OEN	<p>PWM21 輸出控制位</p> <p>0 : 禁止 PWM21 輸出</p> <p>1 : 允許 PWM21 輸出</p>
1	PWM2_OEN	<p>PWM2 輸出控制位</p> <p>0 : 禁止 PWM2 輸出</p> <p>1 : 允許 PWM2 輸出</p> <p>注意：PWM2 允許輸出，必須在 PWM2_EN 置 1 下才有效，否則為 PWM2 輸出關閉狀態（輸出時對應埠必須設為輸出模式）；即使都禁止輸出，只要相 PWM2_EN 位被使能，PWM2 都可以溢出中斷，即此時 PWM2 可以作為計時器使用，此控制位修改立即生效。</p>
0	PWM2_EN	<p>PWM2 模組使能控制位元</p> <p>0 : 關閉 PWM2 模組</p> <p>1 : 打開 PWM2 模組（重新計數）</p> <p>注意：關閉時，PWM2 計數停止，輸出立即關閉。打開時，PWM2 計數器都重新從 1 開始計數，輸出受 PWM2_OEN 和 PWM21_OEN 控制。</p>

#### 10.4.5.2 PWM2控制寄存器PWM2C

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

復位值	0	0	0	0	0	0	0	0
位元符號	PWM2IE	PWM2IF	FLT2S	FLT2C	PWM2S	CK2		

位編號	位元符號	說明
7	PWM2IE	PWM2 中斷允許位 0：禁止 PWM2 中斷 1：允許 PWM2 中斷
6	PWM2IF	PWM2 中斷標誌位元 0：軟體清 0 1：PWM2 週期計數器溢出，由硬體置 1
5	FLT2S	PWM2 FLT 狀態位元 0：PWM 正常狀態，軟體清 0 1：PWM 輸出關閉，硬體置 1
4	FLT2C	PWM2 FLT 引腳配置位元 0：FLT2 為低電平時，PWM 輸出關閉 1：FLT2 為高電平時，PWM 輸出關閉
3-2	PWM2S	PWM2 和 PWM21 輸出模式選擇位元 00：PWM2和PWM21均為高有效 01：PWM2為高有效，PWM21為低有效 10：PWM2為低有效，PWM21為高有效 11：PWM2 和 PWM21 均為低有效

		注意：對於獨立模式，輸出模式選擇位元同樣有效，但與互補模式不同的是：有效期間為占空比期間；而互補模式中對於 PWM2 的有效期間為占空比期間，PWM21 的有效期間為占空比的互補期間。
1-0	CK2	PWM2 時鐘源選擇位元  00 : $F_{osc}/1$  01 : $F_{osc}/8$  10 : $F_{osc}/32$  11 : $F_{osc}/128$

### 10.4.5.3 PWM2週期寄存器PWM2PL、PWM2PH

#### PWM2PL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM2PL[7:0]							

位編號	位元符號	說明
7-0	PWM2PL[7:0]	PWM2 週期寄存器低 8 位

#### PWM2PH

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符	-	-	-	-	PWM2PH[3:0]			



號					
---	--	--	--	--	--

位編號	位元符號	說明
7-4	-	保留位 ( 讀為 0 , 寫無效 )
3-0	PWM2PH[3:0]	PWM2 週期寄存器高 4 位

注意：修改PWM2週期時先修改高位，後修改低位元，讀時不受限制，例如

- (1) PWM2PH = 0x05;
- (2) PWM2PL = 0x08; //此時 PWM 計數器溢出，則下一個週期開始週期計算資料為 0x0508
- (3) PWM2PH = 0x06; //此時 PWM 計數器溢出，則下一個週期開始週期計算資料為 0x0508
- (4) PWM2PL = 0x08; //此時 PWM 計數器溢出，則下一個週期開始週期計算資料為 0x0608
- (5) PWM2PL = 0x09; //此時 PWM 計數器溢出，則下一個週期開始週期計算資料為 0x0609

顯然只要修改 PWM 週期，無論低位元寄存器是否需要修改，低位元都必須寫入一次，且週期修改都只會在下一個 PWM 週期才會生效。

$$\text{PWM2週期} = [\text{PWM2PH} : \text{PWM2PL}] * \text{PWM2工作時鐘源週期}$$

#### 10.4.5.4 PWM2占空比寄存器PWM2DL、PWM2DH

##### PWM2DL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM2DL[7:0]							
號								

位編號	位元符號	說明
-----	------	----

7-0	PWM2DL[7:0]	PWM2 占空比寄存器低 8 位
-----	-------------	------------------

#### PWM2DH

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-				PWM2DH[3:0]			

位編號	位元符號	說明
7-4	-	保留位 ( 讀為 0 , 寫無效 )
3-0	PWM2DH[3:0]	PWM2 占空比寄存器高 4 位

注意：修改 PWM2 占空比寄存器，操作類似修改 PWM2 週期寄存器，都是必須先修改高位後修改低位元，且修改都在下一個週期才有效。

$$\text{PWM2 占空比} = [\text{PWM2DH} : \text{PWM2DL}] * \text{PWM2 工作時鐘週期}$$

#### 10.4.5.5 PWM2死區時間寄存器PWM2DTL、PWM2DH

##### PWM2DTL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM2DTL[7:0]							

位編號	位元符號	說明
7-0	PWM2DTL[7:0]	PWM2 死區時間寄存器低 8 位

### PWM2DTH

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-				PWM2DTH[3:0]			

位編號	位元符號	說明
7-4	-	保留位 ( 讀為 0 , 寫無效 )
3-0	PWM2DTH[3:0]	PWM2 死區時間寄存器高 4 位

當 PWM2M=1 時，PWM2 工作在 2 路獨立模式，此時的死區時間寄存器被用來當做 PWM21 的占空比寄存器，即獨立模式的 PWM2 可以產生 2 路週期相同，但占空比可以不同的 PWM 波形。

互補模式下：PWM2 死區時間 = [ PWM2DTH : PWM2DTL ] \* PWM2 工作時鐘週期；

互補模式下：死區時間必須小於占空比時間，死區時間與占空比時間的和必須小於 PWM2 週期；

獨立模式下：PWM21 占空比時間 = [ PWM2DTH : PWM2DTL ] \* PWM2 工作時鐘週期；

### 10.4.5.6 PWM2移相個數寄存器PWM2PHASEH、PWM2PHASEL

#### PWM2PHASEH

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM2PHASE[15:8]							

位編號	位元符號	說明
7-0	PWM2PHASE[15:8]	PWM2 移相個數寄存器高 8 位

#### PWM2PHASEL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM2PHASE [7:0]							

位編號	位元符號	說明
7-0	PWM2PHASE [7:0]	PWM2 移相個數寄存器低 8 位

### 10.4.5.7 PWM2中斷分頻數PWM2INTDIV

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號					PWM2_OV_INT_DIV			

號		
---	--	--

位編號	位元符號	說明
7-4	-	保留位
3-0	PWM2_OV_INT_DIV	PWM2 溢出中斷分頻

# 11 單路8位PWM

## 11.1 PWM 特性

- 提供 PWM 週期溢出中斷，但中斷與 PWM0、PWM1、PWM2 共用同一向量
- 輸出極性可選擇
- PWM 可做計時器/計數器使用，即週期寄存器寫入時做計時器使用，讀時做計數器使用

## 11.2 PWM 模組相關寄存器

### 11.2.1 PWM3控制寄存器PWM3C

#### PWM3C

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM3EN	PWM3IE	PWM3IF	PWM3OEN	PWM3S	PTCK3[2:0]		

位編號	位元符號	說明
7	PWM3EN	PWM3 模組使能控制位元  0：關閉 PWM3 模組  1：打開 PWM3 模組（重新計數）  注意：關閉時，PWM 計數停止，輸出立即關閉。  打開時，PWM 計數器都重新從 1 開始計數，輸出受 PWM3OEN 控制。
6	PWM3IE	PWM3 中斷允許位

		<p>0：禁止 PWM3 中斷</p> <p>1：允許 PWM3 中斷</p>
5	PWM3IF	<p>PWM3 中斷標誌位元</p> <p>0：軟體清 0</p> <p>1：硬體置 1，僅在 PWM3 計數器溢出（大於 PWM3P 時）才置 1</p>
4	PWM3OEN	<p>PWM3 輸出使能位</p> <p>0：PWM3 禁止輸出</p> <p>1：PWM3 允許輸出</p> <p>注意：PWM3 允許輸出，必須在 PWM3EN 置 1 下才有效，否則為 PWM3 輸出關閉狀態（輸出時對應埠必須設為輸出模式）；即使都禁止輸出，只要相關位被使能，PWM3 都可以溢出中斷，即此時 PWM3 可以作為計時器/計數器使用，此控制位修改立即生效。</p>
3	PWM3S	<p>PWM3 輸出極性選擇位</p> <p>0：PWM3 有效期間為高電平</p> <p>1：PWM3 有效期間為低電平</p> <p>注意：修改此控制位，立即生效，有效期間是指占空比期間。</p>
2-0	PTCK3[2:0]	<p>PWM3 工作時鐘源選擇位元</p> <p>000：<math>F_{osc}/1</math></p> <p>001：<math>F_{osc}/2</math></p> <p>010：<math>F_{osc}/4</math></p> <p>011：<math>F_{osc}/8</math></p> <p>100：<math>F_{osc}/16</math></p>

		<p>101 : <math>F_{osc}/32</math></p> <p>110 : <math>F_{osc}/64</math></p> <p>111 : <math>F_{osc}/128</math></p> <p>注意：修改此控制位，立即生效，不建議在輸出過程中修改。</p>
--	--	--

## 11.2.2 PWM3週期寄存器PWM3P

### PWM3週期寄存器PWM3P

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM3P[7:0]							

位編號	位元符號	說明
7-0	PWM3P[7:0]	PWM3P 週期寄存器

## 11.2.3 PWM3D占空比寄存器PWM3D

### PWM3占空比寄存器PWM3D

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	PWM3D[7:0]							



位編號	位元符號	說明
7-0	PWM3D[7:0]	<p>PWM3D 占空比寄存器</p> <p><math>PWM3P \leq PWM3D</math> 時，占空比 100%</p> <p><math>PWM3D = 0x00</math> 時，占空比 0%</p>

# 12 看門狗計時器WDT

## 12.1 WDT 特性

- 可配置是否溢出重定
- 可配置在空閒/掉電模式下是否允許
- 可靈活配置溢出時間

HC88L051F4看門狗計時器是一個遞增計數器，其時鐘源內部低頻RC，可以通過寄存器選擇在空閒/掉電模式下是否運行。WDT溢出時，晶片是否重定可通過RSTFR寄存器裡的WDTRF位來判斷。如果WDTRST為1，則WDT溢出時會重定系統，如果WDTRST為0，而且WDT的中斷使能，則會產生WDT中斷。

由於內部低頻RC振盪器頻率隨工藝角有偏差，可通過計時器5進行捕獲測量內部低頻RC振盪器的真實頻率，然後根據實際的頻率值進行溢出時間的計算。

HC88L051F4看門狗計時器溢出後有溢出標誌，重定有專用的重定標誌，可設分頻，可設計數溢出值，清WDT操作只需置相應控制位即可，操作靈活。

## 12.2 WDT 相關寄存器

### 12.2.1 WDT 控制寄存器 WDTC

位編號	7	6	5	4	3	2	1	0
R/W	R	R/W	R/W	W	R/W	R/W	R/W	R/W
復位值	0	1	0	0	1	1	1	1
位元符號	-	WDTRST	WDTF	WDTCLR	WDTPD	WDTPS[2:0]		

位編號	位元符號	說明
7	-	保留位
6	WDTRST	<p>WDT 復位允許位</p> <p>0：禁止 WDT 復位</p> <p>1：允許 WDT 復位</p> <p>注意：禁止 WDT 復位時，WDT 計數溢出依舊可以置插斷要求標誌。</p>
5	WDTF	<p>WDT 插斷要求標誌位元</p> <p>0：無 WDT 計數溢出，中斷回應時軟體清 0</p> <p>1：WDT 計數溢出，WDTF 硬體置 1，可用於插斷要求</p>
4	WDTCLR	<p>看門狗清零位</p> <p>置 1 會清零 WDT 計數器，硬體自動清 0 該位元</p>
3	WDTPD	<p>WDT 空閒/掉電模式下運行控制位元</p> <p>0：空閒/掉電模式下允許 WDT 運行，如果 WDTRST=1 會重定喚醒系統，如果 WDTRST=0，而且 EA=1，EWDT=1，會中斷喚醒系統。</p> <p>1：空閒/掉電模式下禁止 WDT 運行</p>
2-0	WDTPS[2:0]	<p>看門狗計時器時鐘源分頻選擇位元</p> <p>000：1/8</p> <p>001：1/16</p> <p>010：1/32</p> <p>011：1/64</p> <p>100：1/128</p>

		101 : 1/256  110 : 1/512  111 : 1/1024
--	--	--

### 12.2.2 WDT 計數比較寄存器 WDTCCR

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	1	1	1	1	1	1	1	1
位元符號	WDTCCR[7:0]							

位編號	位元符號	說明
7-0	WDTCCR[7:0]	WDT 計數比較寄存器  注意：WDT 計數器與 WDTCCR[7:0]匹配時，溢出並且計數器清 0 重新計數，寫入 00 時，將關閉 WDT 功能（但不關閉內部低頻 RC），即相當於禁止 WDT。寫入非 0 數據時，將啟動 WDT。

下麵以 44KHz 為例進行計算，真實的內部低頻 RC 頻率可通過計時器 5 進行捕獲測量得到。

溢出時間 = (WDT分頻係數 \* (WDTCCR[7:0]+1))/實際內部低頻RC頻率。

WDTCCR[7:0] = 0xFF 看門狗溢出時間如下表。

PS2	PS1	PS0	WDT分頻係數	調整步進值	WDT最大溢出時間@44K
0	0	0	8	0.182ms	46.55ms
0	0	1	16	0.364ms	93.09ms
0	1	0	32	0.728ms	186.18ms
0	1	1	64	1.456ms	372.36ms
1	0	0	128	2.912ms	744.73ms

1	0	1	256	5.824ms	1489.45ms
1	1	0	512	11.648 ms	2978.91ms
1	1	1	1024	23.296ms	5957.82ms

## 13 通用非同步收發器UART

### 13.1 UART 特性

- 2 個 UART
- 多種工作方式
- 多種錯誤檢測

### 13.2 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RI = 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RI = 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 $F_{osc}/12 \times 6^{UX6}$
0	1	方式1	非同步	計時器4的溢出率/16
1	0	方式2	非同步	$(2^{SMOD}/64) \times F_{osc}$
1	1	方式3	非同步	計時器4的溢出率/16

#### 13.2.1 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RXD引腳上收發串列資料，TXD引腳發送移位元時鐘。

HC88L051F4提供TXD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。

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

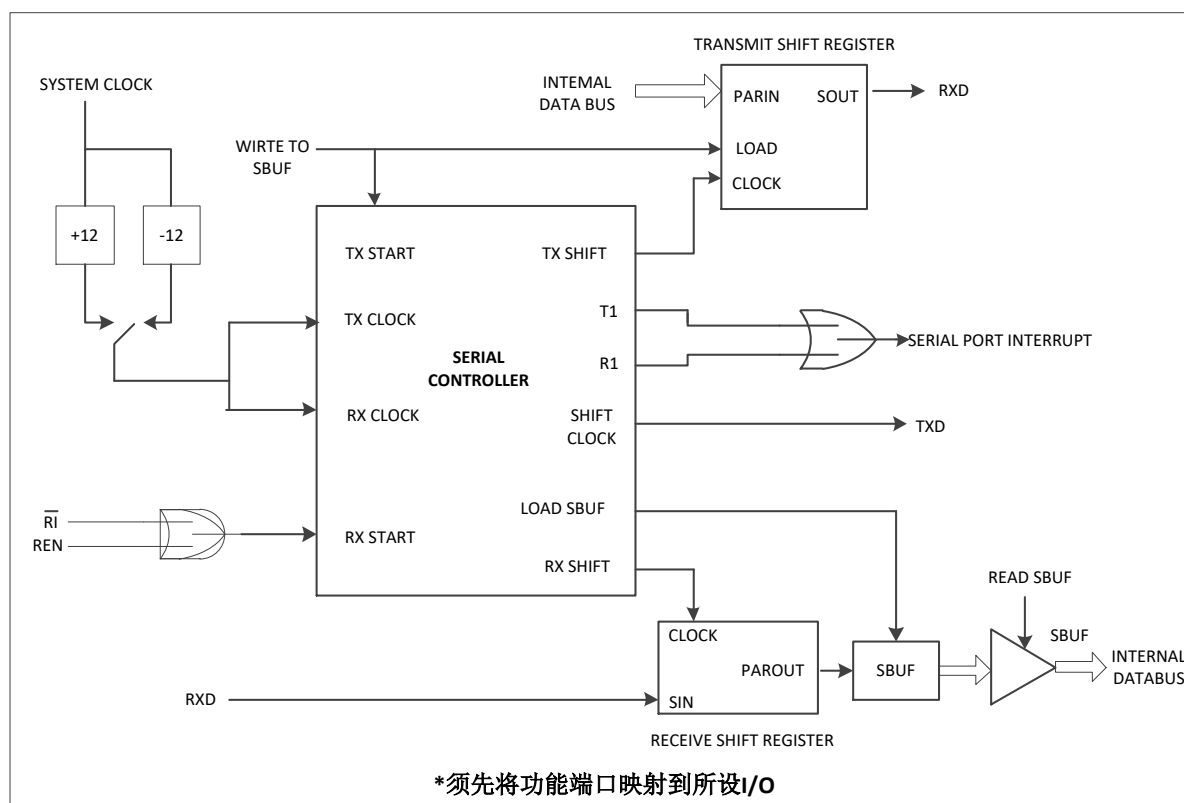


Figure 13-1 UART 模式 0 功能框圖

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 TI 位置 1。

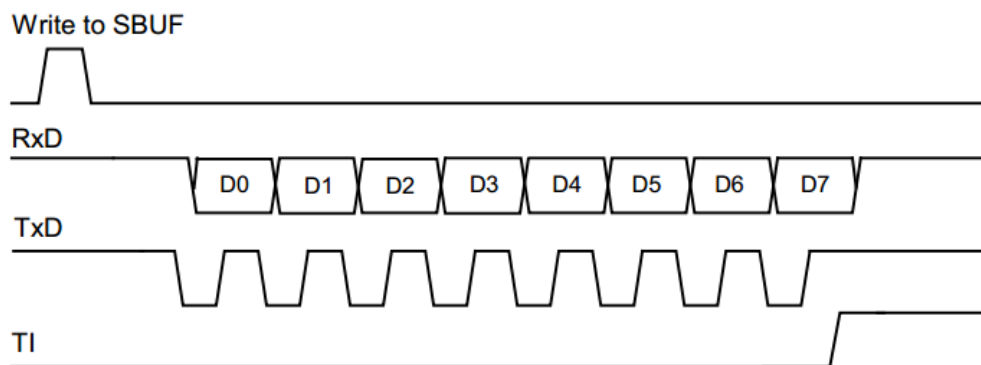


Figure 13-2 模式 0 資料發送時序框圖

REN 位置 1 和 RI 位清 0 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 RI 置位元，直到被軟體清零才允許下一次接收。

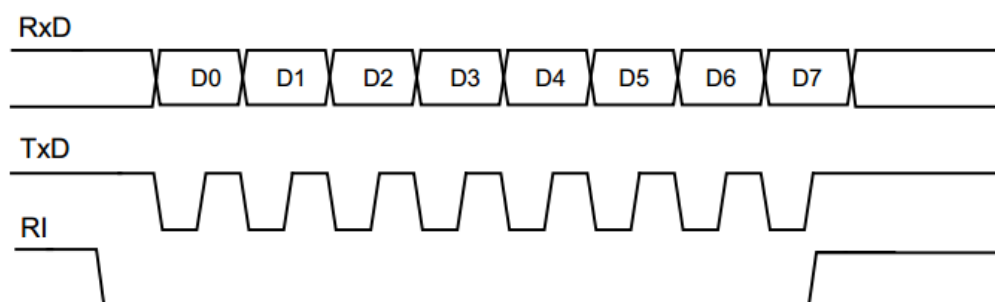


Figure 13-3 模式 0 資料接收時序框圖

### 13.2.2 方式 1：8 位 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 中的串列傳輸速率為計時器 4 的溢出率/16。

功能塊框圖如下圖所示：





193

只有REN置1時才允許接收。當RXD引腳檢測到下降沿時串行口開始接收串行資料。為此，CPU對RXD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RXD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RXD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RXD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，RI置1，但必須滿足下列條件：

- (1) RI = 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，RI被置位。否則接收的幀會丟失。這時，接收器將重新去探測RXD端是否有另一個下降沿。使用者必須用軟體清零RI，然後才能再次接收。

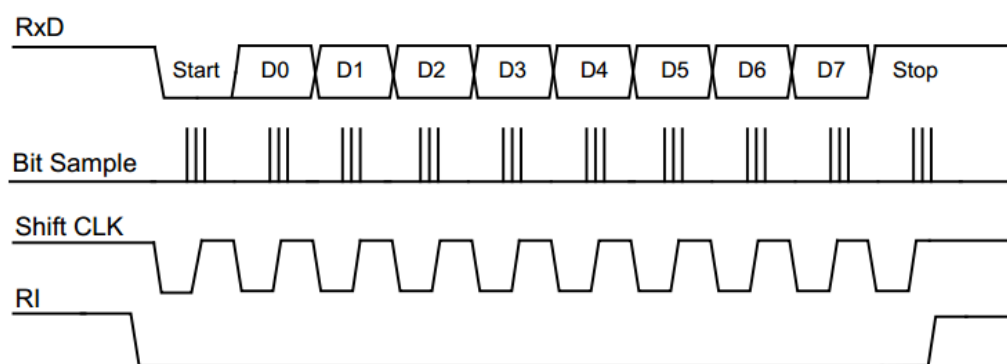


Figure 13-6 模式 1 資料接收時序框圖

### 13.2.3 方式 2：9 位 UART，固定串行傳輸速率，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

[illegible]

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 TXD 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 TXD 引腳上移出，在停止位元開始發送時 TI 標誌置位元。

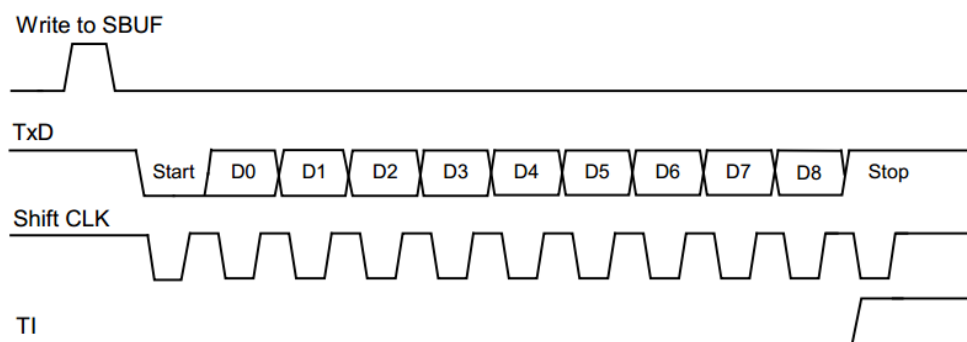


Figure 13-8 模式 2 資料發送時序框圖

只有REN置位時才允許接收。當RXD引腳檢測到下降沿時串列口開始接收串列資料。為此，CPU對RXD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RXD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RXD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RXD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，RI置1，但必須滿足下列條件：

- (1)  $RI = 0$
- (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，RI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RXD引腳上的另一個下降沿。使用者必須用軟體清除RI，然後才能再次接收。

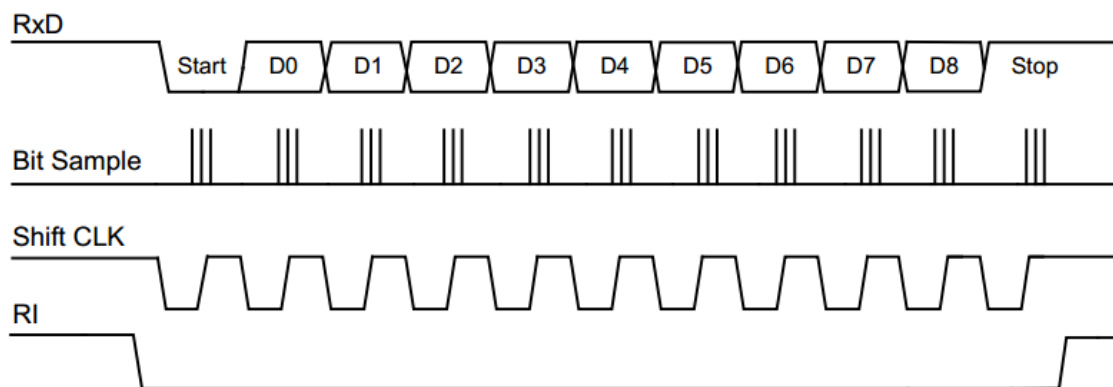


Figure 13-9 模式 2 資料接收時序框圖

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。

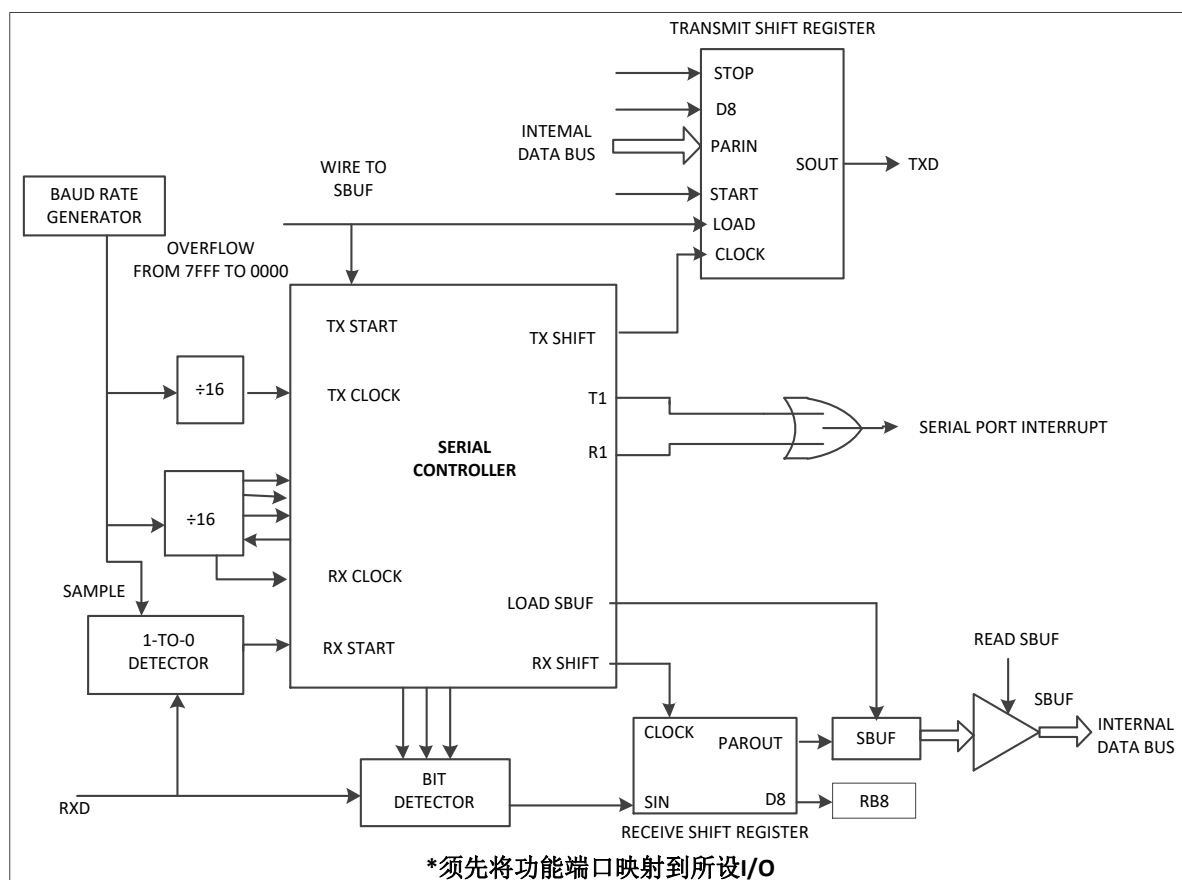


Figure 13-10 UART 模式 3 功能框圖

計時器 4 可以作為 UART1/UART2 的串列傳輸速率發生器。選擇計時器 4 的工作方式 1 作為串列傳輸速率發生器。該方式與自動重載方式相似。計時器 4 的溢出會引起軟體將計時器 4 重載寄存器中的 16 位值載入計時器 4 的計數器中，此時也會產生溢出中斷。如果不希望中斷發生，可以關閉 ET4。

串列傳輸速率由下列公式計算：

$$\text{BaudRate} = \frac{1}{16} \times \frac{f_{T4}/\text{PRESCALER}}{65536 - [\text{TH4}, \text{TL4}]}, \text{ 用計時器 4 作為串列傳輸速率發生器。}$$

上式中，TH4 和 TL4 為計時器 4 資料寄存器。

下表為常用Fosc頻率與常用串列傳輸速率所對應的計時器4的計數值：

常用串列傳輸 速率	Fosc			
	4MHz	8MHz	16MHz	32MHz
1200	FF30	FE5F	FCBF	F97D
2400	FF98	FF30	FE5F	FCBF
4800	FFCC	FF98	FF30	FE5F
9600	FFE6	FFCC	FF98	FF30
19200	FFF3	FFE6	FFCC	FF98
38400	/	FFF3	FFE6	FFCC
57600	/	/	FFEF	FFDD
115200	/	/	/	FFEF

## 13.4 多機通信

### 13.4.1 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RI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。

注意：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。

### 13.4.2 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。

## 13.5 幀出錯檢測

3個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。

### 13.5.1 發送衝突

如果在一個資料發送正在進行時，使用者軟體寫資料到SBUF寄存器時，發送衝突位 ( TXCOL位 ) 置1。如果發生了衝突，新資料會被忽略，不能被寫入發送緩衝器 ( 即不影響傳送 )。

### 13.5.2 接收溢出

RI置1，接收緩衝器中的資料未被讀取，RI被清0，又開始新的資料接收，若在新的資料接收完成前

( RI置1 ) 還未讀取之前接收緩衝區中的資料，在那麼接收溢出位 ( RXROV位 ) 置位。如果發生了接收溢出，接收緩衝器中原來的資料不影響，後面的資料則丟失。

### 13.5.3 幀出錯

如果檢測到一個無效 ( 低 ) 停止位，那麼幀出錯位 ( FE位 ) 置1。

## 13.6 UART1 相關寄存器

### 13.6.1 UART1 控制寄存器 SCON、SCON2

#### SCON

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	FE	RXROV	TXCOL	REN	TB8	RB8	TI	RI

位編號	位元符號	說明
7	FE	幀錯誤檢測位元 0：無幀錯誤或軟體清 0 1：有幀錯誤，硬體置 1
6	RXROV	接收溢出標誌位元

		0：無接收溢出或軟體清 0 1：接收溢出，硬體置 1
5	TXCOL	發送衝突標誌位元 0：無發送衝突或軟體清 0 1：有發送衝突，硬體置 1
4	REN	串列接收使能控制位元 0：禁止串列接收 1：允許串列接收
3	TB8	方式 2/方式 3 時，為要發送的第 9 位元資料，由軟體置 1 或清 0
2	RB8	方式 2/方式 3 時，為接收到的第 9 位元資料，作為同位檢查位元或位址幀/資料幀的標誌位元
1	TI	發送插斷要求中斷標誌位元 0：軟體清 0 1：方式 0 時，當串列發送資料第 8 位元結束時，由硬體自動置 1，其它方式時，在停止位元開始發送時由硬體置 1
0	RI	接收插斷要求中斷標誌位元 0：軟體清 0 1：方式 0 時，當串列接收資料第 8 位元結束時，由硬體自動置 1，其它方式時，串列接收到停止位元開始時刻由硬體置 1

## SCON2

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R	R/W	R	R	R/W	R/W	R/W

復位值	0	0	0	0	0	0	0	0
位元符號	SMOD	-	UX6	-	SM0	SM1	SM2	

位編號	位元符號	說明
7	SMOD	串列傳輸速率加倍控制位 0：在方式 2 中，串列傳輸速率為系統時鐘 $F_{osc}$ 的 1/64 1：在方式 2 中，串列傳輸速率為系統時鐘 $F_{osc}$ 的 1/32
6	-	保留位（讀為 0，寫無效）
5	UX6	串口模式 0 的通信速度設置位元 0：串口模式 0 時鐘為 $F_{osc}/12$ 1：串口模式 0 時鐘為 $F_{osc}/2$
4-3	-	保留位（讀為 0，寫無效）
2-1	SM0:SM1	串口工作方式選擇位元，詳細見下表
0	SM2	多機通信使能控制位(第九位“1”校驗器) 0：在方式 1 時，不檢測停止位，停止位無論是 0 還是 1 都會置位元 RI 在方式 2 和 3 時，不檢測第 9 位元，任何位元組都會置位元 RI 1：在方式 1 時，允許停止位確認檢驗，只有有效的停止位“1”才能置位 RI 在方式 2 和 3 時，只有位址位元組（第 9 位元=“1”）才能置位 RI

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	當 UX6 = 0 時，串列傳輸速率是 $F_{osc}/12$

			器	當UX6 = 1時，串列傳輸速率是 $F_{osc}/2$
0	1	方式1	8位UART，串列傳輸速率可變	計時器4/5的溢出率/16
1	0	方式2	9位UART	$(2^{SMOD}/64) \times F_{osc}$
1	1	方式3	9位UART，串列傳輸速率可變	計時器4/5的溢出率/16

### 13.6.2 UART1 資料緩衝寄存器 SBUF

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	SBUF[7:0]							

位編號	位元符號	說明
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

### 13.6.3 UART1 自動位址識別 SADDR、SADEN

#### 從機位址寄存器SADDR

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	SADDR[7:0]							

位編號	位元符號	說明
-----	------	----

7-0	SADDR[7:0]	從機位址寄存器
-----	------------	---------

#### 從機位址遮罩寄存器 SADEN

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	SADEN [7:0]							

位編號	位元符號	說明
7-0	SADEN [7:0]	從機位址遮罩寄存器

### 13.6.4 串列傳輸速率選擇寄存器 BRTSEL

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R	R	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號							UART2_BRT_SEL	UART1_BRT_SEL

位編號	位元符號	說明
7-2	-	保留位
1	UART2_BRT_SEL	UART2 串列傳輸速率選擇位 0：計時器 5 的溢出率 1：計時器 4 的溢出率
0	UART1_BRT_SEL	UART1 串列傳輸速率選擇位

		0 : 計時器 4 的溢出率 1 : 計時器 5 的溢出率
--	--	----------------------------------

## 13.7 UART2

UART2的控制和工作方式與UART1相同，寄存器請參考UART1

不同點：

1. UART2的寄存器存放在擴展SFR中；
2. UART2只有兩種工作方式；
3. UART2沒有錯誤檢測；
4. UART2沒有自動硬體位址識別。

### 13.7.1 UART2 工作方式

#### 13.7.1.1 方式0：8位UART，可變串列傳輸速率，非同步全雙工

方式0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S2BUF中而停止位元儲存在RB8中。方式0中的串列傳輸速率為計時器5溢出率的1/16。

任何將S2BUF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S2BUF的寫操作不同步。起始位首先在TXD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TXD引腳上移出，在停止位元發出的同時TI標誌置位元。

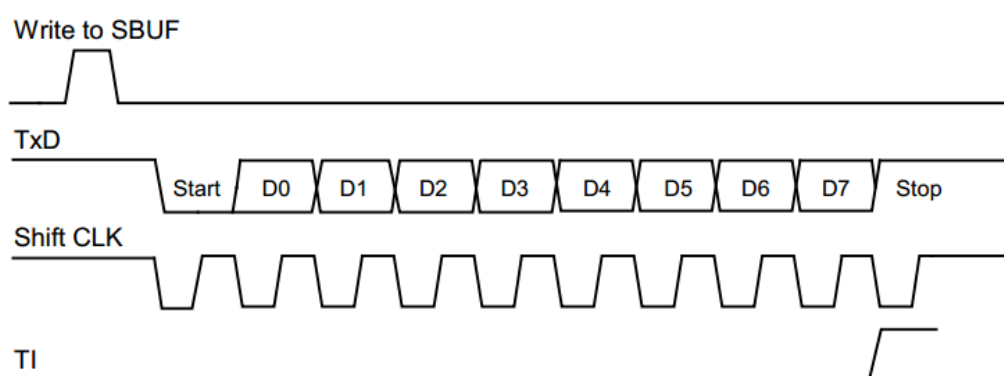


Figure 13-11 Send Timing of Mode 1



只有REN置1時才允許接收。當RXD引腳檢測到下降沿時串行口開始接收串行資料。為此，CPU對RXD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RXD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RXD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RXD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，RI置1，但必須滿足下列條件：

- (1) RI = 0
- (2) SM2 = 0不判斷停止位或者SM2=1判斷停止位，且停止位必須為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，RI被置位。否則接收的幀會丟失。這時，接收器將重新去探測RXD端是否另一個下降沿。使用者必須用軟體清零RI，然後才能再次接收。

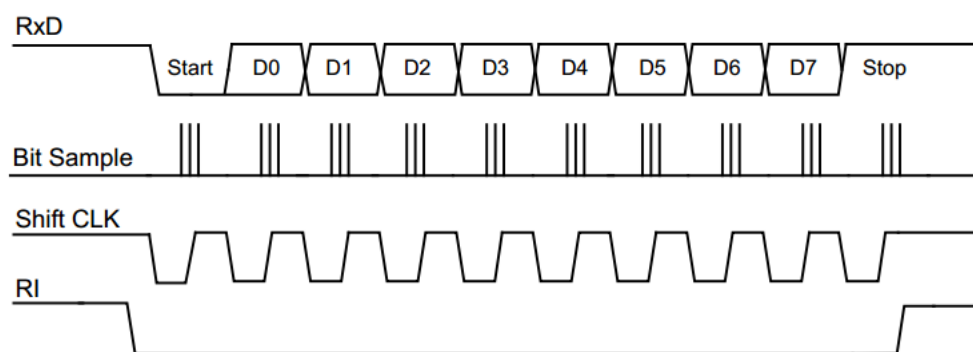


Figure 13-12 Receive Timing of Mode 1

### 13.7.1.2 方式1：9位UART，可變串行傳輸速率，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1支援多機通信，在資料傳送時，第9資料位元（TB8位元）可以寫0或1，例如，可寫入PSW中的奇偶位P，或用作

多機通信中的資料/位址標誌位元。當接收到資料時，第 9 資料位移入 RB8 而停止位不保存。

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 TXD 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 TXD 引腳上移出，在停止位元開始發送時 TI 標誌置位元。

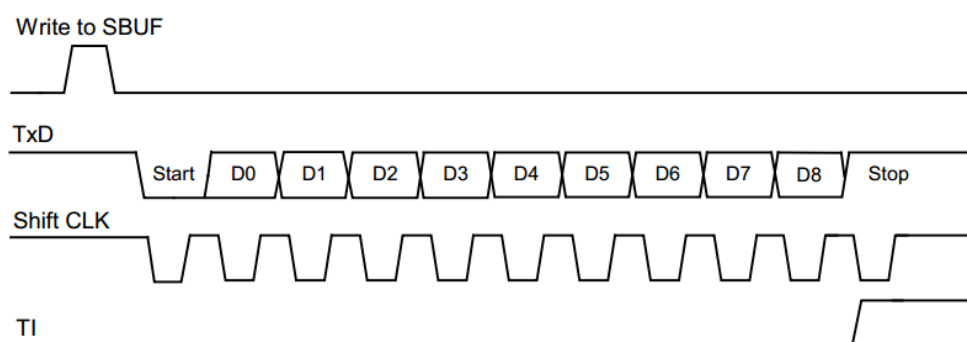


Figure 13-13 Send Timing of Mode 2

只有REN置位時才允許接收。當RXD引腳檢測到下降沿時串列口開始接收串列資料。為此，CPU對RXD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RXD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RXD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RXD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個資料位移入之後，移位暫存器的內容被分別裝入SBUF和RB8中，但必須滿足下列條件：

- (3) RI = 0
- (4) SM2 = 0

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。但還需要檢測停止位元，只有停止位為1，才能置位RI，如果停止位為0，則RI不會置位。

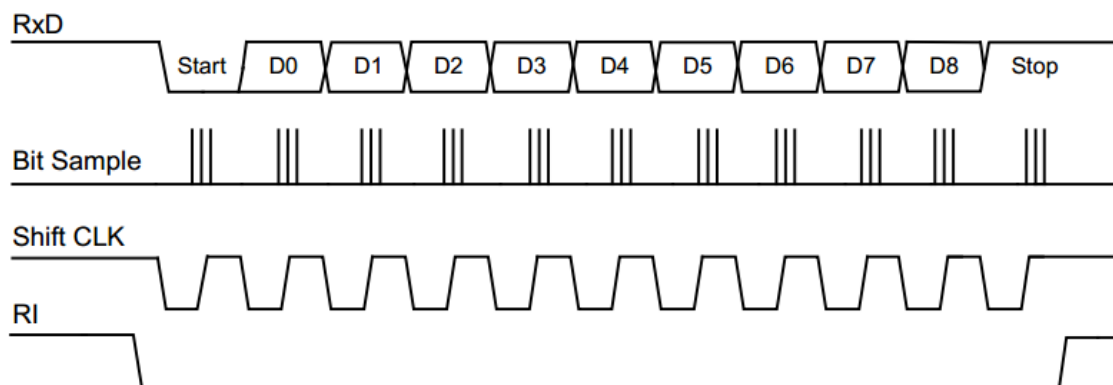


Figure 13-14 Receive Timing of Mode 2

## 13.7.2 UART2 控制寄存器 S2CON、S2CON2

### S2CON

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	FE			REN	TB8	RB8	TI	RI

位編號	位元符號	說明
7	FE	幀錯誤檢測位元 0：無幀錯誤或軟體清 0 1：有幀錯誤，硬體置 1
6-5	-	保留位（讀為 0，寫無效）
4	REN	串列接收使能控制位元 0：禁止串列接收 1：允許串列接收

3	TB8	方式 1 時，為要發送的第 9 位元資料，由軟體置 1 或清 0
2	RB8	方式 1 時，為接收到的第 9 位元資料，作為同位檢查位元或位址幀/資料幀的標誌位元
1	TI	發送插斷要求中斷標誌位元  0：軟體清 0  1：在停止位元開始發送時由硬體置 1
0	RI	接收插斷要求中斷標誌位元  0：軟體清 0  1：串列接收到停止位元開始時刻由硬體置 1

## S2CON2

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R	R	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-						SM1	SM2
號								

位編號	位元符號	說明
7-2	-	保留位 ( 讀為 0 , 寫無效 )
1	SM1	<p>0 : 8 位 UART , 計時器 5 的溢出率/16</p> <p>1 : 9 位 UART , 計時器 5 的溢出率/16</p> <p>注 : 1、UART2 的串列傳輸速率設置詳見計時器 5 方式 1: 章節。</p> <p>2、UART2 的串列傳輸速率的配置值可以參考 UART1。</p>
0	SM2	<p>第九位檢測使能位</p> <p>0 : 在方式 0 時 , 不檢測停止位 , 停止位無論是 0 還是 1 都會置位元 RI</p> <p>在方式1時 , 不檢測第9位 , 第9位無論是0還是1都會置位元RI</p> <p>1 : 在方式 0 時 , 只有停止位為 1 才能置位 RI</p> <p>在方式1時 , 只有第9位為1才能置位RI</p>

## 13.7.3 UART2 資料緩衝寄存器 S2BUF

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符	S2BUF[7:0]							

號	
---	--

位編號	位元符號	說明
7-0	S2BUF[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

## 14 串列外部設備介面SPI

### 14.1 SPI 特性

- 全雙工，三/四線同步傳輸
- 主從機操作
- 4級可程式設計主時鐘頻率
- 極性相位可程式設計的串列時鐘
- 可選擇資料傳輸方向
- 寫衝突及接收溢出標誌
- 帶MCU中斷的主模式模式衝突檢測
- 帶MCU中斷的傳輸結束標誌
- 主模式支援高達8Mbps的通信速率 (  $F_{osc}=32\text{MHz}$  )，從模式下通信速率須在 $F_{osc}/16$ 及 $F_{osc}/16$ 以下

### 14.2 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 位也將被清 0，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。

## 14.3 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。



The diagram illustrates the internal architecture of the SPI module. It shows the following components and their interconnections:

- System Clock (系统时钟)**: Input to the **分频器** (Divider).
- 分频器**: Outputs the **SPI时钟** (SPI Clock) to the **选择** (Select) block and the **时钟逻辑** (Clock Logic) block.
- 选择** (Select): Receives the SPI Clock and outputs **SPR0** and **SPR1** signals to the **SPI控制** (SPI Control) block and the **8位移位寄存器** (8-bit Shift Register).
- 8位移位寄存器**: Contains the **读数据缓冲区** (Read Data Buffer) and is connected to the **管脚控制逻辑** (Pin Control Logic) block.
- 管脚控制逻辑**: Manages the external pins: **MISO**, **MOSI**, **SCLK**, and **SS**. It receives signals from the **8位移位寄存器** and the **时钟逻辑** block.
- 时钟逻辑** (Clock Logic): Receives the SPI Clock and outputs the **Clock** signal to the **8位移位寄存器**. It also receives signals from the **管脚控制逻辑** block.
- SPI控制** (SPI Control): Receives signals from the **选择** block and the **管脚控制逻辑** block. It outputs **SPR0** and **SPR1** signals to the **8位移位寄存器** and the **SPI控制寄存器** (SPI Control Register).
- SPI控制寄存器** (SPI Control Register): Receives signals from the **SPI控制** block and the **管脚控制逻辑** block. It outputs **SSIG**, **DORD**, **SPEN**, **SPR0**, **SPR1**, **MSTR**, **CPHA**, and **CPOL** signals to the **8位移位寄存器** and the **管脚控制逻辑** block.
- SPI状态寄存器** (SPI Status Register): Receives signals from the **SPI控制** block and the **管脚控制逻辑** block. It outputs **SPIF**, **WCOL**, **ROV**, and **MODF** signals to the **SPI控制** block.
- 内部数据总线** (Internal Data Bus): Connected to the **SPI控制** block and the **SPI状态寄存器**.
- SPI中断请求** (SPI Interrupt Request): Output from the **SPI控制** block.

**\*须先将功能端口映射到所设I/O**

Figure 14-1 SPI 功能方框圖

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。

注意：寫入的資料不會影響到需要讀出的資料。

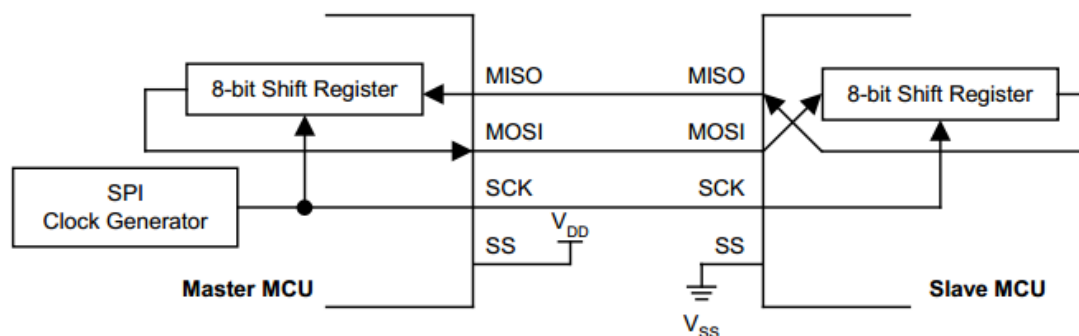


Figure 14-2 全雙工主從互聯圖

## 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 SPIF 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 SPIF 置 1 後讀出，但必須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，SPIF 可正常置 1。

## 從模式

### (1) 模式啟動

將 MSTR 置 0 ( 若  $\overline{SS}$  被使能則必須拉低 ) 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 (  $\overline{SS}$  引腳必須維持低電平 )，否則資料傳送將失敗 ( SPIF 不會被置 1 )。

### (2) 發送

SPI 從設備下不能啟動資料傳送，所以 SPI 從設備必須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，但必須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。

## 14.6 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。

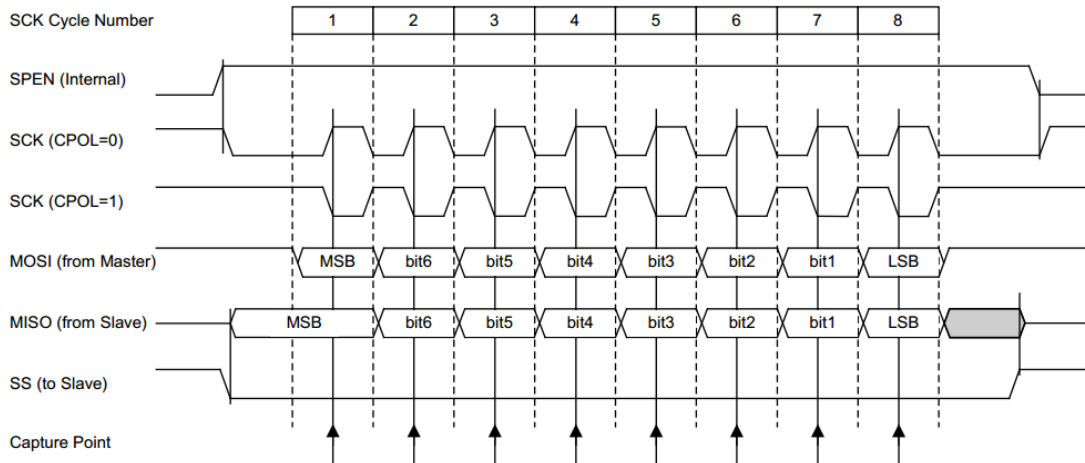


Figure 14-3 資料傳送形式 ( CPHA=0 )

如果  $CPHA = 0$ ；數據在 SCK 的第一沿就被捕獲，所以從設備必須在 SCK 的第一個沿之前就準備好資料，因此， $\overline{SS}$  引腳的下降沿從設備就開始資料。 $\overline{SS}$  引腳在每次傳送完一個位元組後必須拉高，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  $\overline{SS}$  腳被強制使能。

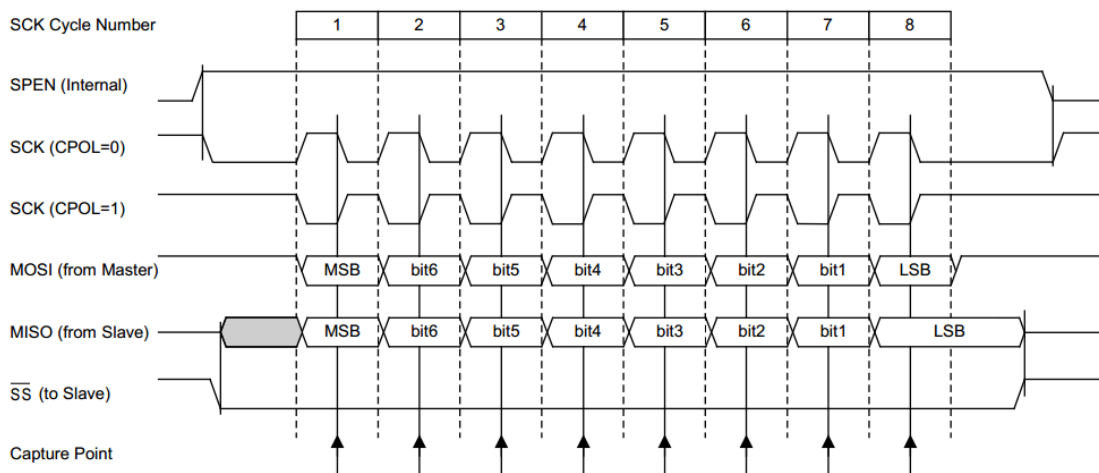
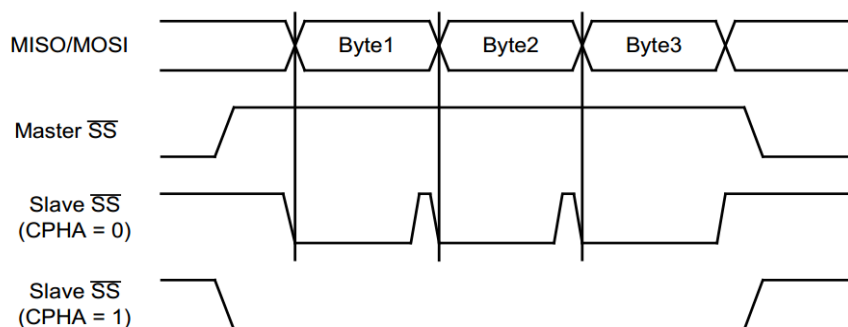


Figure 14-4 資料發送形式 ( CPHA=1 )

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。用戶必須在第一個 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。

Figure 14-5 CPHA/ $\overline{SS}$  時序

## 14.7 SPI 出錯檢測

SPSTA 寄存器中的一些標誌位元表示 SPI 通信中的通信錯誤情況：

### (1) 模式故障 (MODF)

SPI 主模式下的模式故障出錯表明  $\overline{SS}$  引腳上的電平狀態與實際設備模式不一致，MODF 標誌位元將被置 1 (可產生中斷)，以來表明 SPI 控制系統中存在多主設備的衝突情況，此時硬體將自動清除 SPEN 位，即先關閉 SPI 模組；同時硬體也將自動清除 MSTR 位。需要重啟 SPI 模組時，MODF 必須先軟體寫 1 清 0，再置 SPEN 位。

### (2) 寫衝突 (WCOL)

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。需軟體寫 1 清 0

### (3) 接收溢出 (RXOV)

在接收第二資料完成前仍未清除之前接收資料產生的 SPIF 標誌，將置接收溢出標誌 RXOV，SPIF 被置 1 時，後面的資料將不會被傳入接收寄存器，故接收的資料存入 SPDAT 前必須清除 SPIF，RXOV 位元需軟體寫 1 清 0。

## 14.8 SPI 中斷

兩種 SPI 狀態標誌 SPIF&MODF 都能產生一個 CPU 插斷要求。

串列資料傳輸完成標誌 SPIF：完成一個位元組資料發送/接收後由硬體置 1。

故障模式標誌 MODF：該位被置 1 是指設備模式(主機)與 $\overline{SS}$ 引腳電平不一致，SSIG 位為 1 ( $\overline{SS}$ 未被使能)時，無 MODF 插斷要求。

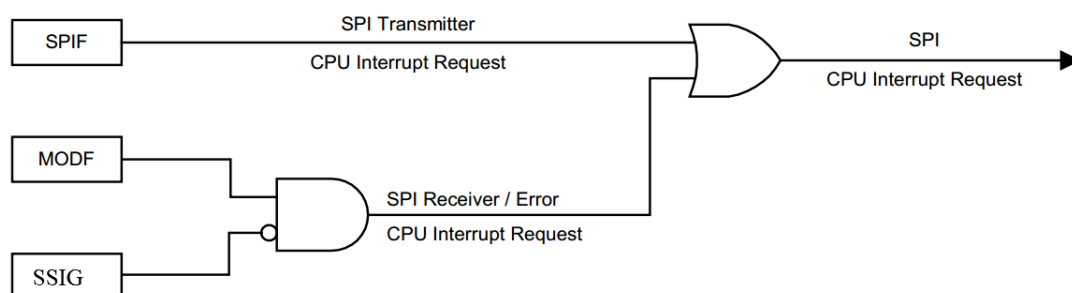


Figure 14-6 SPI 插斷要求的產生

## 14.9 SPI 配置對照

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	-
---	---	-----	---	---	----	----	----	---



## 14.10 SPI 相關寄存器

### 14.10.1 SPI 控制寄存器 SPCTL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	SSIG	SPEN	DORD	MSTR	CPOL	CPHA	SPR[1:0]	

位編號	位元符號	說明
7	SSIG	<p><math>\overline{SS}</math> 引腳使能位</p> <p>0 : <math>\overline{SS}</math> 腳被使能用於確定器件為主機還是從機</p> <p>1 : MSTR 確定器件為主機還是從機，<math>\overline{SS}</math> 腳作為普通 I/O 使用</p>
6	SPEN	<p>SPI 使能位</p> <p>0 : 禁止 SPI 模組，相關管腳為普通 I/O(建議 I/O 時設為高阻)</p> <p>1 : 使能 SPI 模組，相關管腳為 SPI 通信管腳</p>
5	DORD	<p>傳送方向選擇位</p> <p>0 : MSB 先發送</p> <p>1 : LSB 先發送</p>
4	MSTR	<p>主/從機模式選擇位元</p> <p>0 : 從機模式</p> <p>1 : 主機模式</p>
3	CPOL	SPI 時鐘極性選擇位元

		<p>0 : SCK 空閒時為低電平</p> <p>1 : SCK 空閒時為高電平</p>
2	CPHA	<p>SPI 時鐘相位選擇位元</p> <p>0 : 數據在 SPI 時鐘的第一個邊沿採樣</p> <p>1 : 數據在 SPI 時鐘的第二個邊沿採樣</p> <p>注：SSIG = 0 &amp; CPHA = 0 時，數據在 <math>\overline{SS}</math> 為低被驅動；CPHA = 1 時，數據在 SCK 的前時鐘沿驅動。</p>
1-0	SPR[1:0]	<p>SPI 時脈速率選擇控制位</p> <p>00 : <math>F_{osc} / 4</math></p> <p>01 : <math>F_{osc} / 16</math></p> <p>10 : <math>F_{osc} / 64</math></p> <p>11 : <math>F_{osc} / 128</math></p>

## 14.10.2 SPI 狀態寄存器 SPSTAT

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R
復位值	0	0	0	0	0	0	0	0
位元符號	SPIF	WCOL	RXOV	MODF	-			

位編號	位元符號	說明
7	SPIF	<p>SPI 傳輸完成標誌位元</p> <p>0：軟體寫 1 清 0</p> <p>1：一次傳送完成時，硬體置 1，也做插斷要求標誌位元</p>
6	WCOL	<p>SPI 寫衝突標誌位元</p> <p>0：軟體寫 1 清 0</p> <p>1：傳送過程中對 SPDAT 執行寫操作硬體置 1 ( 正在傳送的資料不受影響 )</p>
5	RXOV	<p>SPI 接收溢出標誌位元</p> <p>0：軟體寫 1 清 0</p> <p>1：發生接收溢出，硬體置 1</p> <p>注意：接收為雙 BUFF，接收溢出發生在第二個資料傳送完成前仍未清除之前接收資料產生的 SPIF 標誌，故每次準備接收下一個資料前必須先清除 SPIF，否則 RXOV 將置 1，RXOV 置 1 不會影響 SPI 接收。</p>
4	MODF	<p>模式故障標誌位元</p> <p>0：軟體寫 1 清 0</p>

		1 : SS 引腳電平與 SPI 模式不一致時，硬體置 1 ( 且立即切成從機模式 )，也 做插斷要求標誌位元
3-0	-	保留位 ( 讀為0，寫無效 )

### 14.10.3 SPI 資料寄存器 SPDAT

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符 號	SPDAT[7:0]							

位編號	位元符號	說明
7-0	SPDAT[7:0]	SPI 資料寄存器

# 15 IIC匯流排

## 15.1 IIC 特性

- 雙線通信
- 支援主機模式及從機模式
- 支援多主機通信時仲裁功能
- 支援位址可程式設計
- 支援標準速率 ( 最多 100kbps ) 和快速 ( 最多 400kbps )

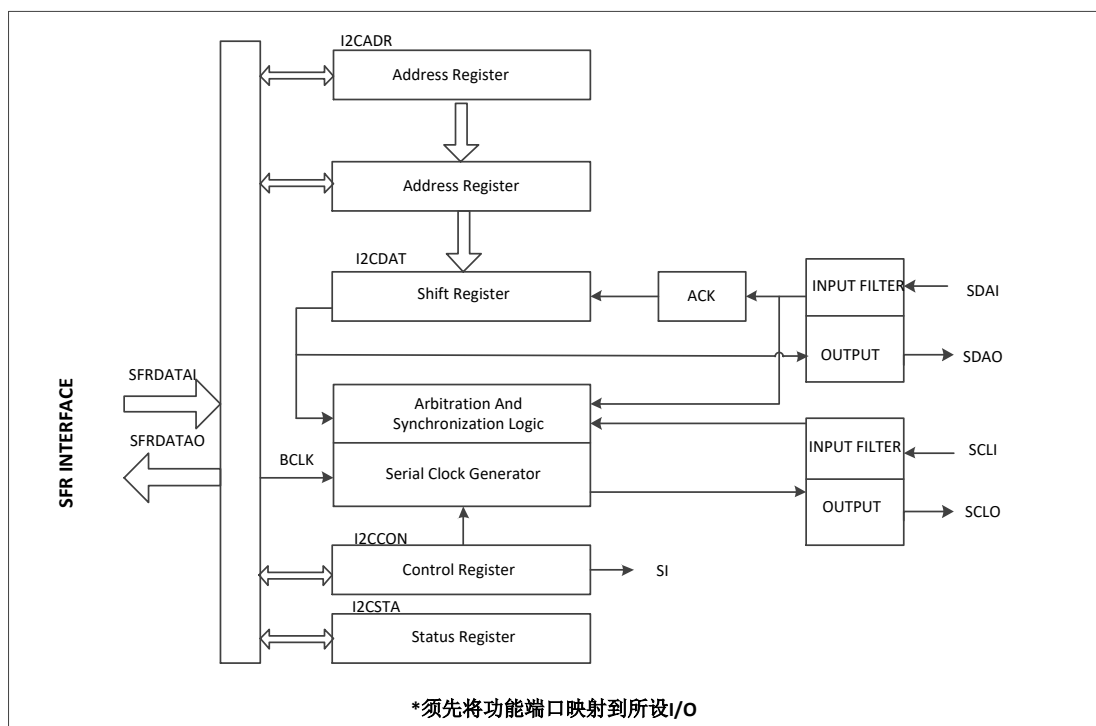


Figure 15-1 IIC 功能框圖

## 15.2 IIC 匯流排工作原理

物理結構上，IIC 系統由一條串列資料線 SDA 和一條串列時鐘線 SCL 組成。主機按一定的通信協定向從機定址和進行資訊傳輸，在資料傳輸時，由主機初始化一次資料傳輸，主機使資料在 SDA 線上

傳輸的同時還通過 SCL 線傳輸時鐘。資訊傳輸的物件和方向以及資訊傳輸的開始和終止均由主機決定。

每個器件都有一個唯一的位址，而且可以是單接收的器件或者可以接收也可以發送的器件。發送器或接收器可以在主模式或從模式下操作，這取決於晶片是否必須啟動資料的傳輸還是僅僅被定址。

下圖是最常用、最典型的 IIC 匯流排連接方式。

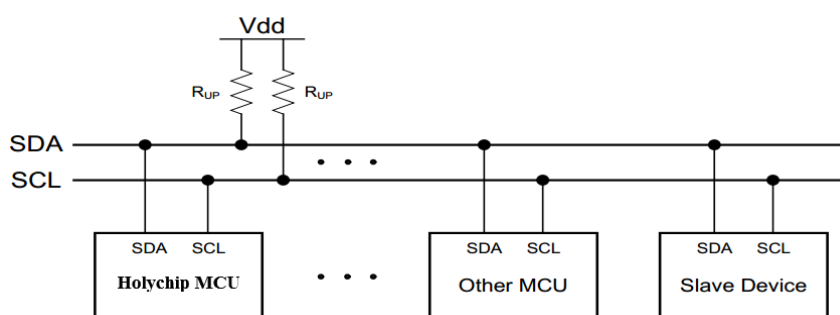


Figure 15-2 IIC 匯流排連接圖

### 15.3 匯流排上資料的有效性

IIC 匯流排是以串列方式傳輸資料，從資料位元組的最高位元開始傳送，每一個資料位元在 SCL 上都有一個時鐘脈衝相對應。在時鐘線高電平期間資料線上必須保持穩定的邏輯電平狀態，高電平為資料 1，低電平為資料 0。只有在時鐘線為低電平時，才允許資料線上的電平狀態變化，如 Figure15-3 所示。

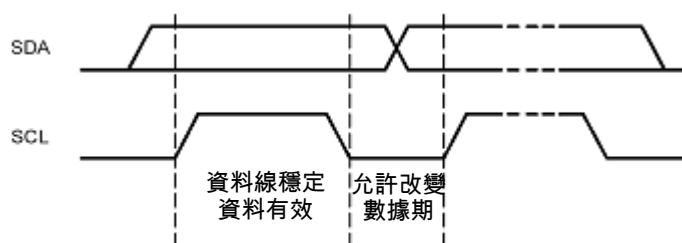


Figure 15-3 IIC 匯流排上資料的有效性

## 15.4 匯流排上的信號

IIC 匯流排在傳送資料過程中共有四種類型信號，它們分別是：開始信號、停止信號、重新開始信號和應答信號。

開始信號 ( START )：如 Figure 15-4 所示，當 SCL 為高電平時，SDA 由高電平向低電平跳變，產生開始信號。當匯流排空閒的時候，例如，沒有主動設備在使用匯流排( SDA 和 SCL 都處於高電平 )，主機通過發送開始 ( START ) 信號建立通信。

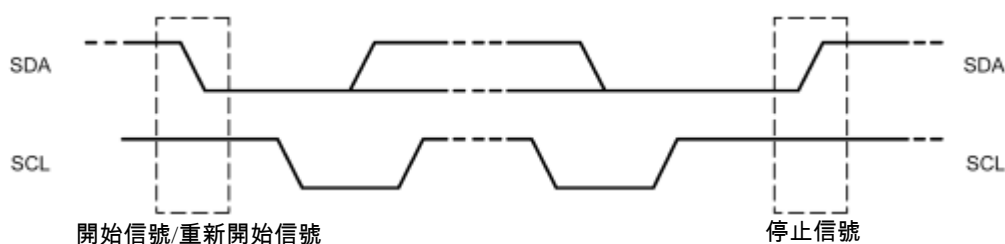


Figure 15-4 開始、重新開始、停止信號

停

止信號 ( STOP )：如 Figure15-4 所示，當 SCL 為高電平時，SDA 由低電平向高電平跳變，產生停止信號。主機通過發送停止信號，結束資料通信。

重新開始信號 ( Repeated START )：在 IIC 匯流排上，由主機發送一個開始信號啟動一次通信後，在首次發送停止信號之前，主機通過發送重新開始信號，可以轉換與當前從機的通信模式，或是切換到與另一個從機通信。如 Figure15-5 所示，當 SCL 為高電平時，SDA 由高電平向低電平跳變，產生重新開始信號，它的本質就是一個開始信號。

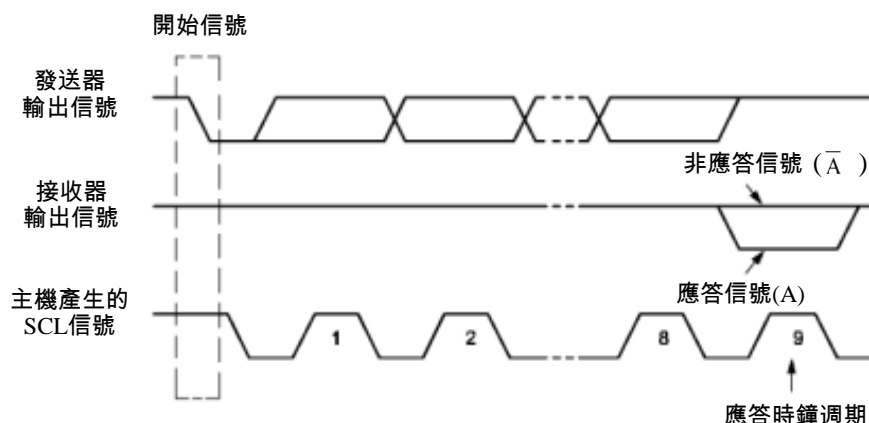


Figure 15-5 IIC 匯流排的應答信號

應答信號 (A): 接收資料的 IIC 在接收到 8 位元資料後，向發送資料的 IC 發出的特定的低電平脈衝。每一個資料位元組後面都要跟一位元應答信號，表示已收到資料。應答信號在第 9 個時鐘週期出現，這時發送器必須在這一時鐘位元上釋放資料線，由接收設備拉低 SDA 電平來產生應答信號，由接收設備保持 SDA 的高電平來產生非應答信號 ( $\bar{A}$ )，如 Figure15-5 所示。所以，一個完整的位元組資料傳輸需要 9 個時鐘脈衝。如果從機作為接收方向主機發送非應答信號，這樣，主機方就認為此次資料傳輸失敗；如果是主機作為接收方，在從機發送器發送完一個位元組資料後，發送了非應答信號，從機就認為資料傳輸結束，並釋放 SDA 線。不論是以上哪種情況都會終止資料傳輸，這時，主機或是產生停止信號釋放匯流排，或是產生重新開始信號，開始一次新的通信。開始信號、重新開始信號和停止信號都是由主控制器產生，應答信號由接收器產生，匯流排上帶有 IIC 匯流排界面的器件很容易檢測到這些信號。

## 15.5 匯流排上資料初始格式

一般情況下，一個標準的 IIC 通信由四部分組成：開始信號、從機位址傳輸、資料傳輸、停止信號。

由主機發送一個開始信號，啟動一次 IIC 通信；在主機對從機定址後，再在匯流排上傳輸資料。

IIC 匯流排上傳送的每一個位元組均為 8 位元，首先發送的資料位元為最高位元，每傳送一個位元組後都必須跟隨一個應答位元，每次通信的資料位元組數是沒有限制的；在全部資料傳送結束後，由主機發送停止信號，結束通信。



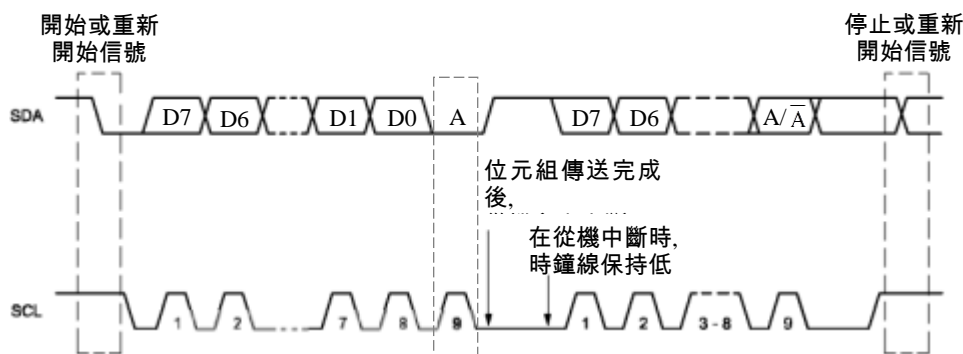


Figure 15-6 IIC 匯流排的資料傳輸格式

如

Figure 15-6 所示，時鐘線為低電平時資料傳送將停止進行。這種情況可以用於當接收器接收到一個位元組資料後要進行一些其它工作而無法立即接收下一個資料時，迫使匯流排進入等候狀態，直到接收器準備好接收新資料時，接收器再釋放時鐘線使資料傳送得以繼續正常進行。例如，當接收器接收完主控制器的一個位元組資料後，產生中斷信號並進行中斷處理，中斷處理完畢才能接收下一個位元組資料，這時接收器在中斷處理時將鉗住 SCL 為低電平，直到中斷處理完畢才釋放 SCL。

## 15.6 IIC 匯流排定址約定

IIC 匯流排系統中掛接的所有週邊器件，一般均擁有一個專用的 7 位從器件位址碼。由於 7 位元從器件位址碼，其編碼空間最多只有 128 個，後來在原有的 7 位位址碼格式基礎上，又發展了 10 位地址碼格式。10 位元位址格式仍然符合匯流排協定。

“廣播呼叫”是個例外，它可以通過將第一個位元組的資料全部賦值為 0 來定址所有器件。廣播呼叫用於當主機希望發送相同資訊到幾個從機時情況。當該位址在使用時，其他器件根據軟體配置可能回應應答或忽略。如果器件回應廣播呼叫，其操作就像從機接收器模式。

## 15.7 主機向從機讀寫 1 個位元組資料的過程

如 Figure 15-7 所示，主機要向從機寫 1 個位元組資料時，主機首先產生 START 信號，然後緊跟著發送一個從機位址，這個位址共有 7 位，緊接著的第 8 位元是資料方向位元 ( R/W )，0 表示主機發送資料 ( 寫 )，1 表示主機接收資料 ( 讀 )，這時候主機等待從機的應答信號 ( A )，當主機收到應答信號時，發送要訪問的位址，繼續等待從機的應答信號，當主機收到應答信號時，發送 1 個位元組的資料，繼續等待從機的應答信號，當主機收到應答信號時，產生停止信號，結束傳送過程。

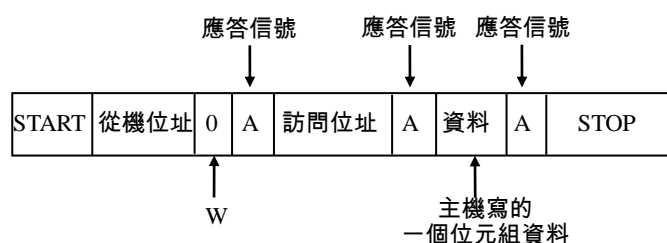
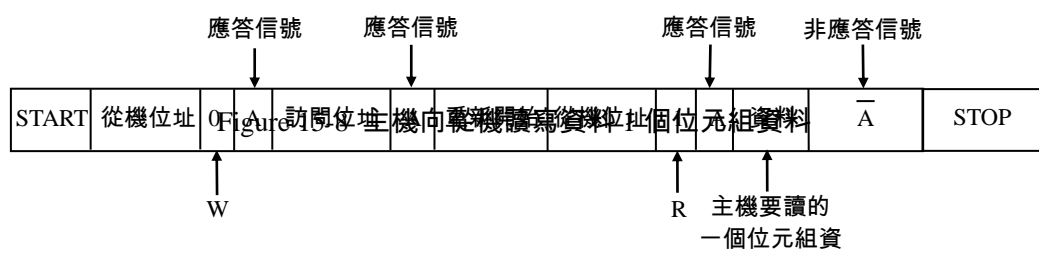


Figure 15-7 主機向從機寫資料

如 Figure 15-7 所示，主機要從從機讀 1 個位元組資料時，主機首先產生 START 信號，然後緊跟著發送一個從機位址，注意此時該位址的第 8 位為 0，表明是向從機寫命令，這時候主機等待從機的應答信號 ( A )，當主機收到應答信號時，發送要訪問的位址，繼續等待從機的應答信號，當主機收到應答信號後，主機要改變通信模式 ( 主機將由發送變為接收，從機將由接收變為發送 ) 所以主機發送重新開始信號，然後緊跟著發送一個從機位址，注意此時該位址的第 8 位為 1，表明將主機設置成接收模式開始讀取資料，這時候主機等待從機的應答信號，當主機收到應答信號時，就可以接收 1 個位元組的資料，當接收完成後，主機發送非應答信號，表示不在接收資料，主機進而產生停止信號，結束傳送過程。



## 15.8 IIC 工作模式

### 15.8.1 主機發送模式

在主機發送模式下，向從機接收器發送幾個資料位元組。主機通過 CR[2:0]設置期望時脈速率並向 IICEN 位寫 1 使能 IIC 匯流排，設置 STA 位元為 1 進入主機發送模式，只要匯流排空閒，硬體將測試匯流排並產生起始信號，成功產生起始信號後，SI 標誌位元將置位元且 IICSTA 的狀態碼為 08H，之後就是給 IICDAT 載入目標從機位址和資料方向位元“寫”(SLA+W)，SLA+W 開始傳輸時 SI 位必須清零。

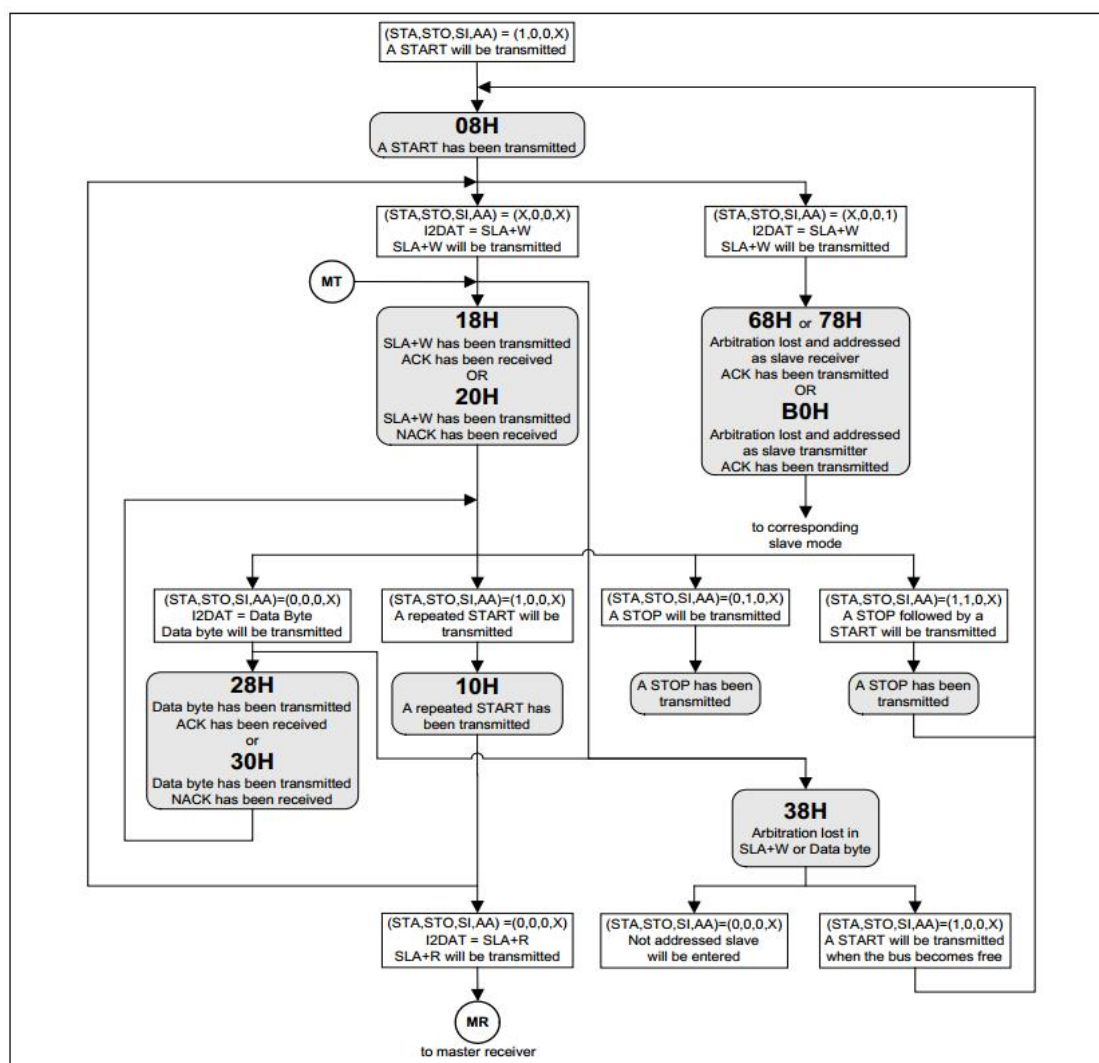


Figure 15-9 主機發送模式流程與狀態

## 15.8.2 主機接收模式

在主機接收模式下，從從機發送器接收幾個位元組的資料。傳輸開始與主機發送模式相似，在起始信號之後，IICDAT 應該載入目標從機位址和資料方向位元“讀”(SLA+R)，SLA+R 位元組發送後，且返回應答位，重新置位元 SI 標誌且 IICSTA 讀出為 40H，SI 標誌應該被清零以便接收從機發送過來的資料，如果 AA 標誌位置位元，主機接收器將應答從機發送器，如果清零 AA，主機接收器將不會應答從機，並釋放從機發送器為不被定址的從機，然後主機產生停止信號或重複起始信號中止傳輸或開始另一次傳輸。

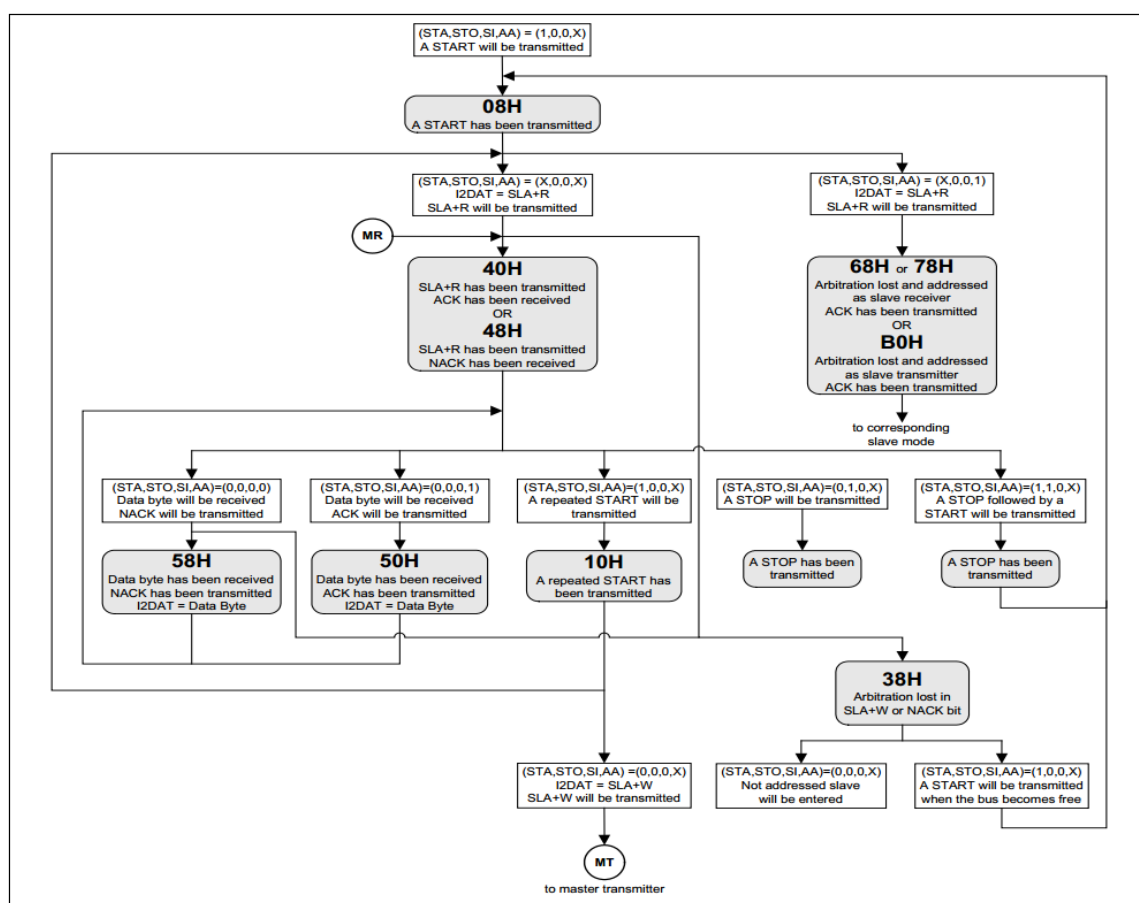


Figure 15-10 主機接收模式流程與狀態

### 15.8.3 從機發送模式

在從機發送模式下，發送幾個位元組資料到主機接收器。確定 IICADR 和 IICCON 的值之後，IIC 等待自己的地址被定址“讀”（SLA+R）。如果仲裁失敗後，也可以進入從機發送模式。

在從機被 SLA+W 定址後，應該清 SI 標誌以便傳輸資料到主機發送器，通常主機接收器將在從機發送每個位元組資料之後返回應答，如果沒有接收到應答，如果繼續傳輸將發送全“1”，就成為不被定址的從機，如果在傳輸中清了 AA 標誌，從機發送最後一個位元組資料，下一次傳輸資料全為“1”，從機成為不被定址。

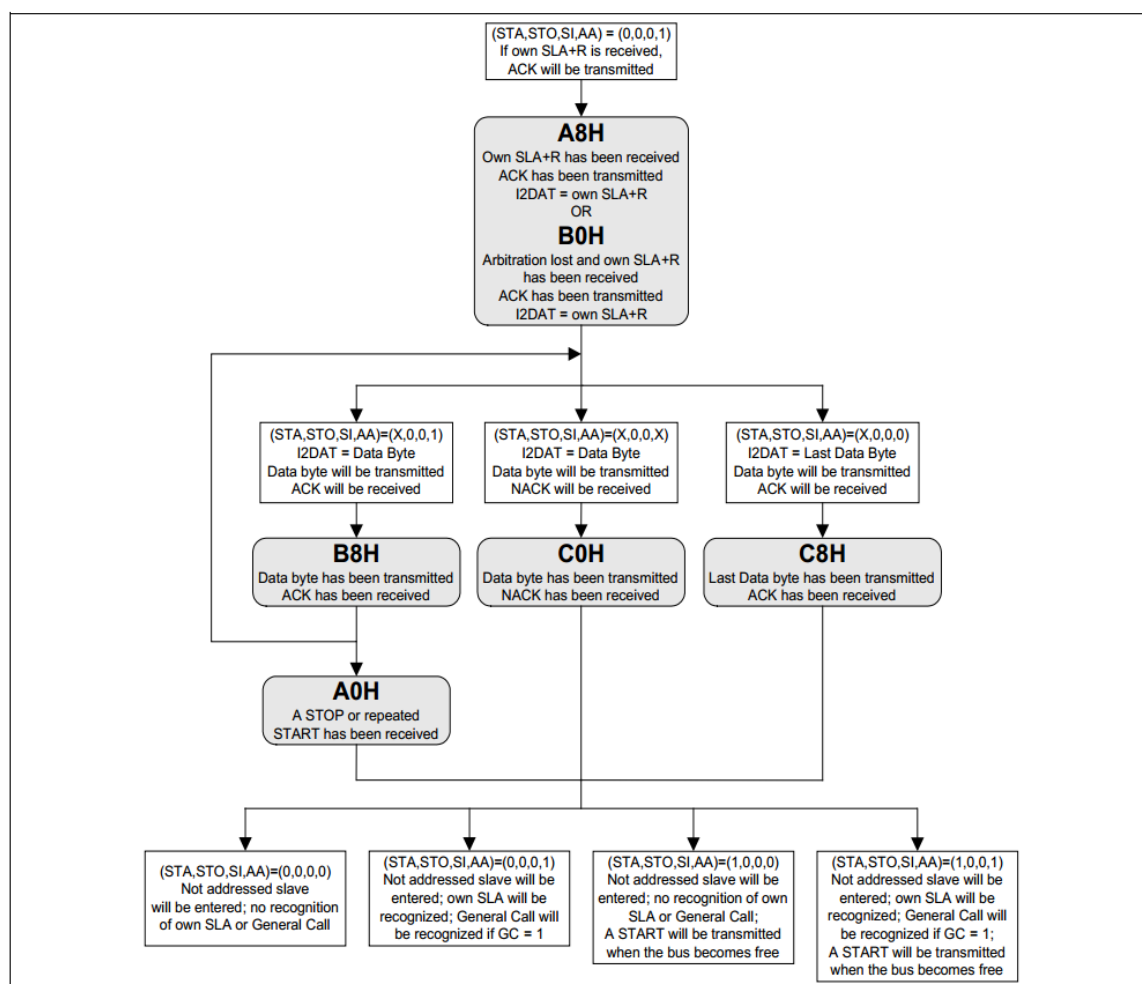


Figure 15-11 從機發送模式流程與狀態

## 15.8.4 從機接收模式

在從機接收模式下，從主機發送器接收幾個位元組資料。發送開始之前，IICADR 必須裝載回應器件的位址，以讓主機定址，AA 位必須設置使能應答自身從機位址或廣播呼叫，完成以上初始過程後，IIC 等待自身位址被定址與資料方向位元“寫”(SLA+W)或被廣播呼叫定址。如果在仲裁失敗時，也可以進入從機接收模式。

在從機被 SLA+W 定址後，應該清 SI 標誌以便接收主機發送過來的資料，傳輸期間，如果 AA 位為 0，從機將在下次接收到的資料位元組之後返回無應答 (non-acknowledge)，從機也不被定址並與主機分離，不能接收 IICDAT 的任何位元組，而保持當前接收到的資料位元組。

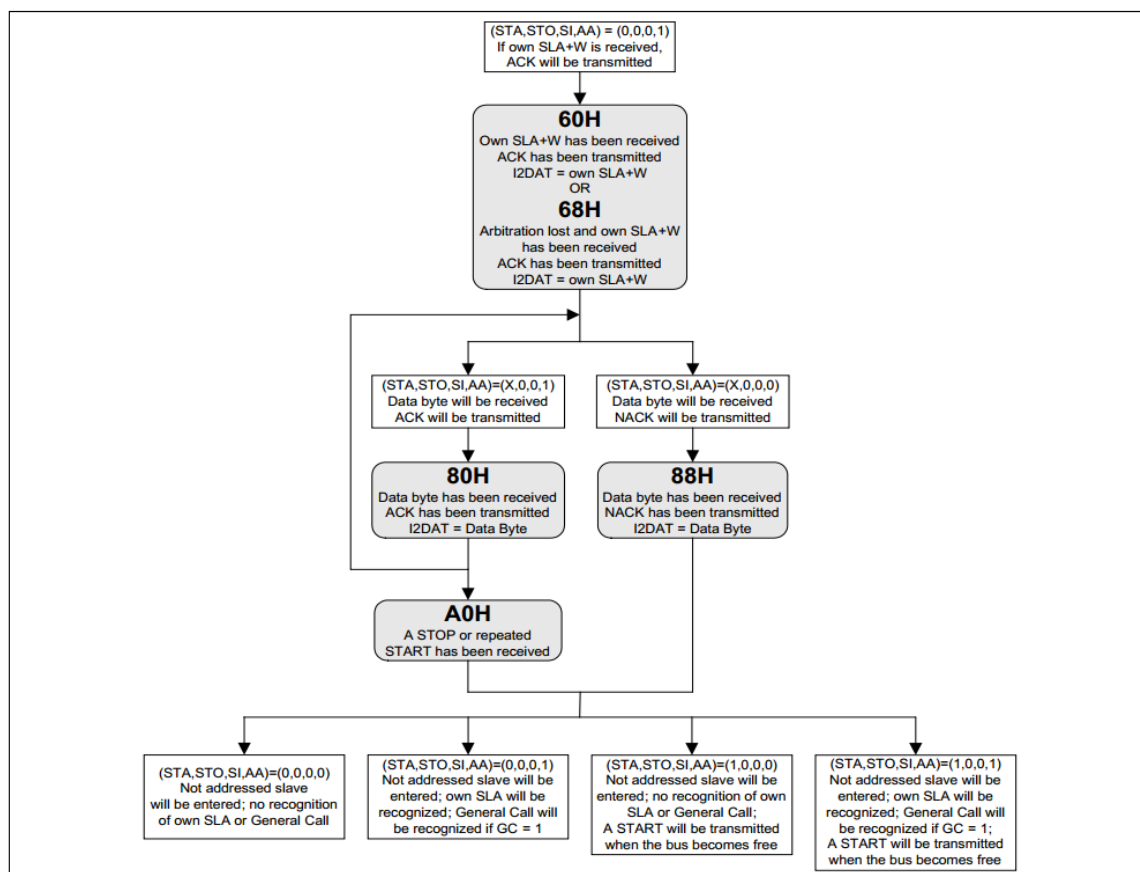


Figure 15-12 從機接收模式流程與狀態

## 15.8.5 廣播呼叫

廣播呼叫是從機接收模式的一種特殊情況，即從機位址和資料方向位元全為 0，被廣播呼叫定址的從機在正常從機接收模式的 IICSTA 裡有不同狀態碼，如果仲裁失敗，也可以產生廣播呼叫。

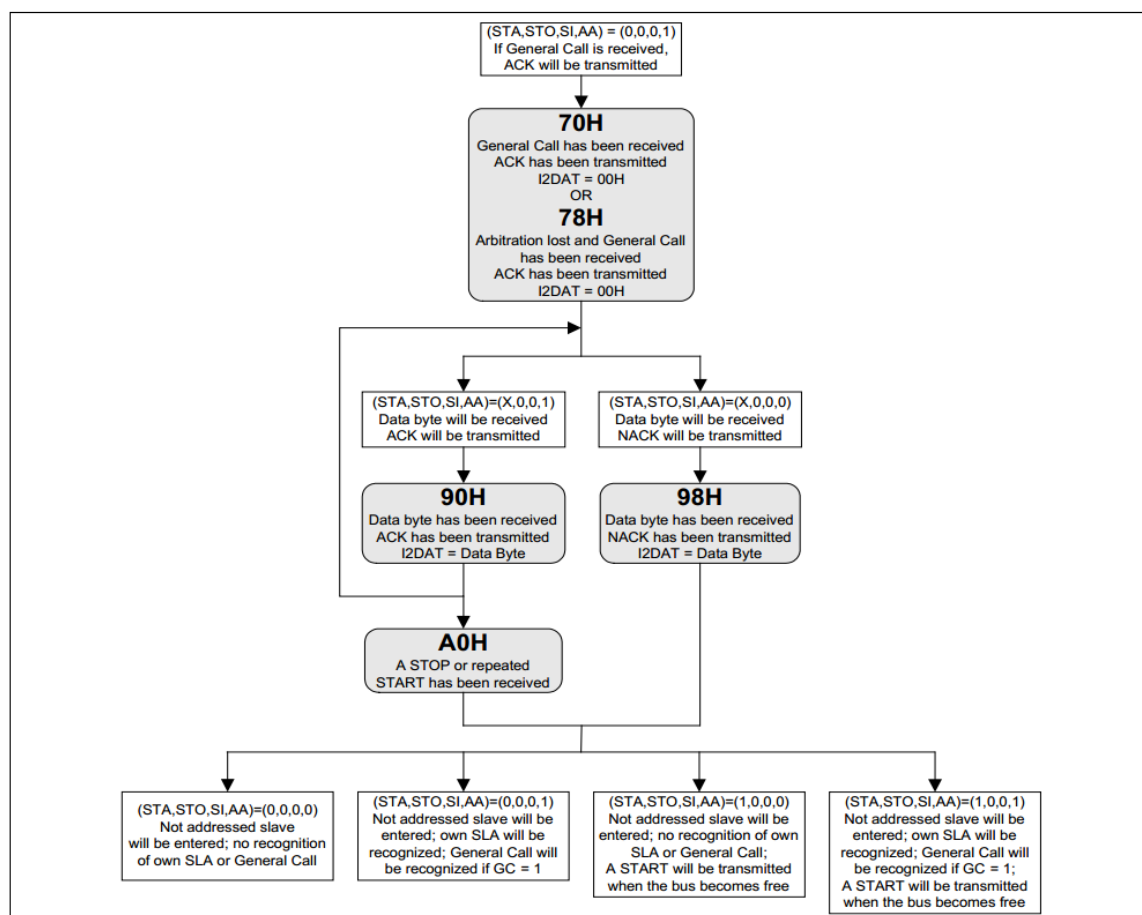


Figure 15-13 廣播呼叫模式流程與狀態

## 15.8.6 其他狀態

有兩個 IICSTA 狀態碼與 24 個定義狀態不一致，即前面提到的 0F8H 和 00H 狀態。

第一個狀態碼 0F8H 表示在每次傳輸期間沒有得到相關資訊，同時，SI 標誌為 0 且沒有 IIC 插斷要求。

另一個標誌碼 00H 意味在傳輸過程中發生錯誤，匯流排錯誤是由 START 或停止信號暫時出現在一個非法的位置，如地位址位元組裡第 2 位換到第 8 位元，或資料位元組包括應答位元，當出現匯流排錯誤時，SI 標誌立即置位元，當在 IIC 匯流排上檢測到匯流排錯誤，工作器件立即切換到不被定址



從機模式，釋放 SDA 和 SCL 總，置位 SI 標誌，將 00H 載入 IICSTA。要從匯流排錯誤恢復，STO 位元必須設置為邏輯 1 且 SI 必須清零，然後，STO 由硬體清零且在沒有停止信號就釋放 IIC 匯流排。

特例：如果沒有成功產生 START 或重複起始信號，IIC 匯流排被 SDA 的低電平阻擋，如一個從 CPU 時鐘件沒有位元同步，可以通過在 SCL 匯流排上發送額外時鐘脈衝解決這個問題。當 STA 位置位時，IIC 硬體發送額外時鐘脈衝，但是由於 SDA 被拉低，不能產生起始信號，當 SDA 匯流排最終被釋放，發送一個普通的 START 條件，進入狀態 08H，繼續進行序列傳輸。當 SDA 為低，如果發送重複起始信號，IIC 硬體也執行以上相同的動作。此情況下，在成功發送起始信號後，進入狀態 08H，而不是進入 10H。

注：軟體不能解決這類匯流排問題。

## 15.9 IIC 匯流排相關寄存器

### 15.9.1 IIC 控制寄存器 IICCON

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	CR2	IICEN	STA	STO	SI	AA	CR1	CR0

位編號	位元符號	說明
7	CR2	IIC 通信時鐘選擇位元 2
6	IICEN	IIC 模組使能位元 0：禁止 IIC 模組 1：啟動 IIC 模組

5	STA	<p>起始位</p> <p>0：不發送起始信號</p> <p>1：匯流排空閒時產生起始信號。忙時，等待停止信號後產生一個起始信號。主機模式下，IIC 準備好發送或接收一個或多個位元組時，置 1 產生一個重複的起始信號</p>
4	STO	<p>停止位</p> <p>0：不發送停止信號</p> <p>1：主機模式時產生停止信號，當檢測到匯流排上出現停止信號。IIC 硬體清除 STO 標誌。STO 標示的設置也用於將 IIC 設備從錯誤狀態( IICSTA 為 00H )恢復，此條件下，沒有停止信號發送 IIC 匯流排上。若 STA 和 STO 都置 1，且在主機模式下設備為原始的，IIC 匯流排將產生停止信號並立即伴隨著起始信號。如果設備為從機模式，置 STO 恢復到非定址從機，STO 將會硬體清 0。</p>
3	SI	<p>IIC 串列中斷標誌位元</p> <p>0：沒有 IIC 串列中斷發生</p> <p>1：產生 IIC 通信狀態碼中除 0F8H 之外的狀態碼時置 1。必須軟體清 0</p>
2	AA	<p>應答標誌位元</p> <p>0：回復 NACK ( SDA 上為高電平 )</p> <p>1：回復 ACK ( SDA 上為低電平 )</p>
1	CR1	IIC 通信時鐘選擇位元 1
0	CR0	IIC通信時鐘選擇位元0

CR[2:0] IIC 通信時鐘選擇位元：

CR2	CR1	CR0	F <sub>osc</sub>				分頻係數
			6MHz	12 MHz	16 MHz	24 MHz	
0	0	0	23KHz	47KHz	63KHz	92KHz	256
0	0	1	27KHz	54KHz	71KHz	108KHz	224
0	1	0	31KHz	63KHz	83KHz	124KHz	192
0	1	1	37KHz	75KHz	100KHz	148KHz	160
1	0	0	6.25KHz	12.5KHz	17KHz	25KHz	960
1	0	1	50KHz	100KHz	133KHz	200KHz	120
1	1	0	100KHz	200KHz	266KHz	400KHz	60
1	1	1	T5 溢出率/8 ( 需要配置計時器 5 為串列傳輸速率發生器 )				

## 15.9.2 IIC 狀態寄存器 IICSTA

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R	R	R	R
復位值	1	1	1	1	1	0	0	0
位元符號	IICSTA[7:3]					-		
號								

位編號	位元符號	說明
7-3	IICSTA[7:3]	IIC 狀態碼，共有 26 個可能的狀態碼，狀態碼中除 0F8H 外都可置 SI 標誌
2-0	-	保留位

## 15.9.3 IIC 資料寄存器 IICDAT

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

復位值	0	0	0	0	0	0	0	0
位元符號	IICDAT[7:0]							

位編號	位元符號	說明
7:0	IICDAT[7:0]	<p>IIC 數據</p> <p>IICDAT 包含一個位元組的將被發送或剛接收到的 IIC 資料。只要 SI 為邏輯 1，IICDAT 中的資料保持不變，在 IIC 發送接收過程中，讀或寫 IICDAT 的結果都是不確定的。</p> <p>當 IICDAT 的資料被移出，匯流排上的資料同步被移入以更新 IICDAT。IICDAT 常顯示當前 IIC 匯流排上的最後位元組。因此失去仲裁，在傳輸之後的 IICDAT 原始值被改變。</p>

### 15.9.4 IIC 位址寄存器 IICADR

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	IICADR[7:1]							GC

位編號	位元符號	說明
7-1	IICADR[7:1]	從機模式：IIC 設備自身從機位址

		主機模式：該資料無影響
0	GC	<p>廣播呼叫位</p> <p>0：廣播呼叫常被忽略</p> <p>1：如果 AA 標誌為 1，廣播呼叫被識別；如果 AA 為 0，忽略廣播呼叫</p> <p>注：該位元只在從機模式有效，主機模式無影響。做從機時，置 AA 標誌，在空閒模式下，若匯流排其它主機定址位址與本從機位址匹配，則將喚醒本從機。</p>

# 16 模數轉換ADC

## 16.1 ADC 特性

- 最多 16 個外部通道及 2 個內部通道 ( 包括 GND ) 的 12/10 位 ADC 檢測
- 參考電壓可選內部 1V、2V、3V、4V、VDD 及外部 Vref
- 可選擇轉換資料對齊方向
- 可選擇轉換資料位元數
- ADC 轉換完成可中斷
- 模擬看門狗
- 多種觸發方式
- ADC 連續轉換
- 單通道 ( P0.2 埠 ) ADC 省電喚醒

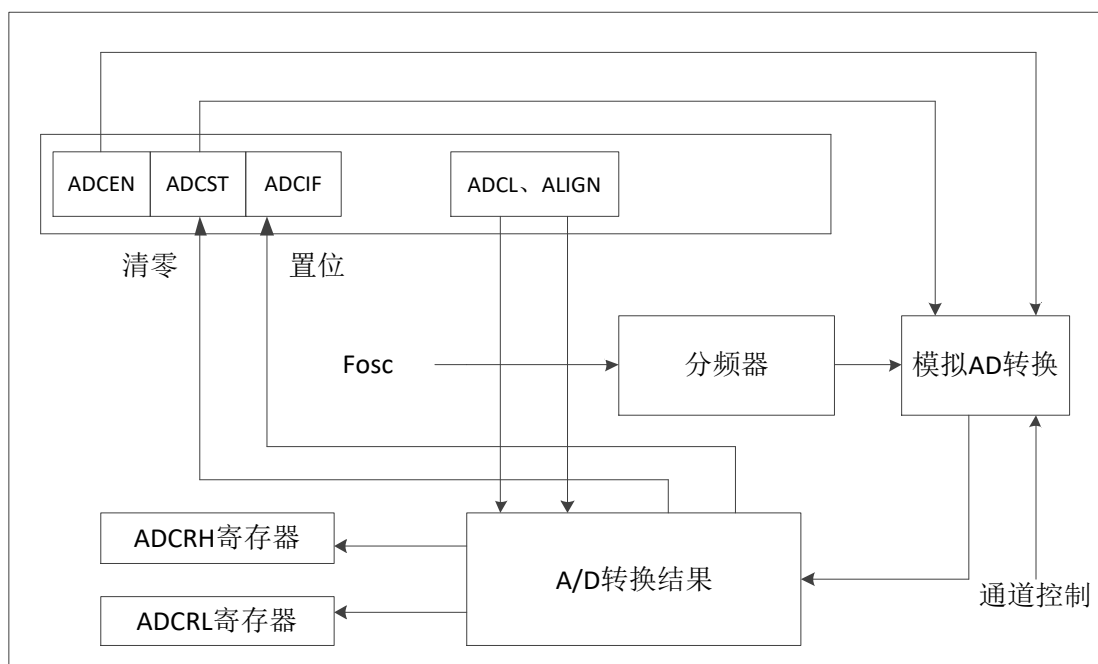


Figure 16-1 ADC 功能框圖

## 16.2 ADC 省電喚醒

在晶片進入省電模式( IDLE 或 PD )後，可通過使能 ADC 省電喚醒功能將晶片從省電模式喚醒，這個功能可以實現用一個 IO 口檢測多個按鍵。

具體操作如下：

1. 通過配置 ADC 喚醒控制寄存器 ADCWC 允許喚醒；
2. 將 P0.2 配置為類比頻道；
3. 通過 P0.2 埠上拉電阻選擇寄存器配置喚醒電阻（比如配置 P0.2 上拉電阻為 50 K $\Omega$ ）；
4. 使能 IE 和 IE1 寄存器裡的 EA 和 EADC；
5. 進入省電模式（IDLE 或 PD）；
6. 如果 Figure16-3 中的某個鍵被按下，埠上就會有一個電壓值，當這個電壓值小於 4.2V（@VDD=5V）時，AMWIF 標誌位元會被拉高。晶片也會從省電模式醒來，同時進入 ADC 中斷服務程式，軟體清零 AMWIF 標誌；
7. 開啟 ADC 功能，採集 P0.2 上的電壓值，根據不同的電壓值判斷不同的按鍵。

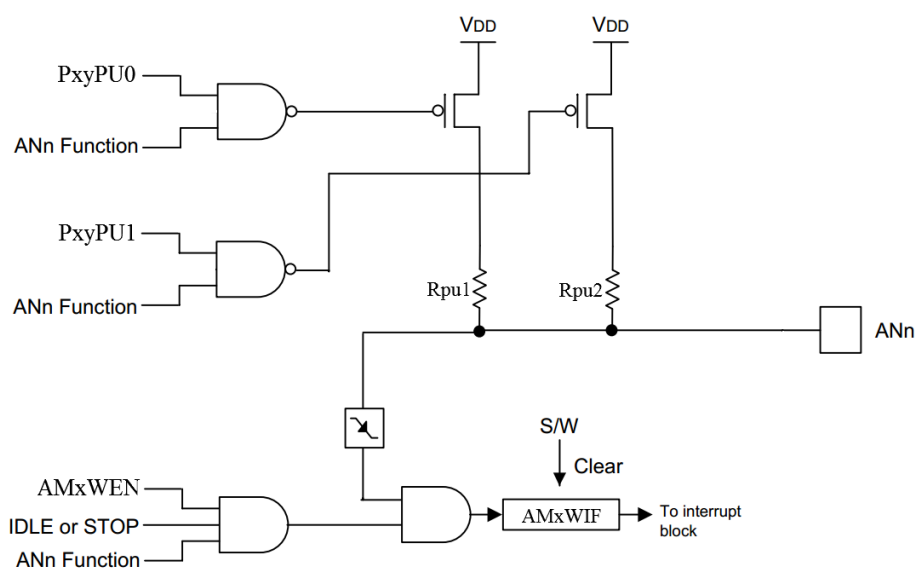


Figure 16-2 ADC 省電喚醒功能框圖

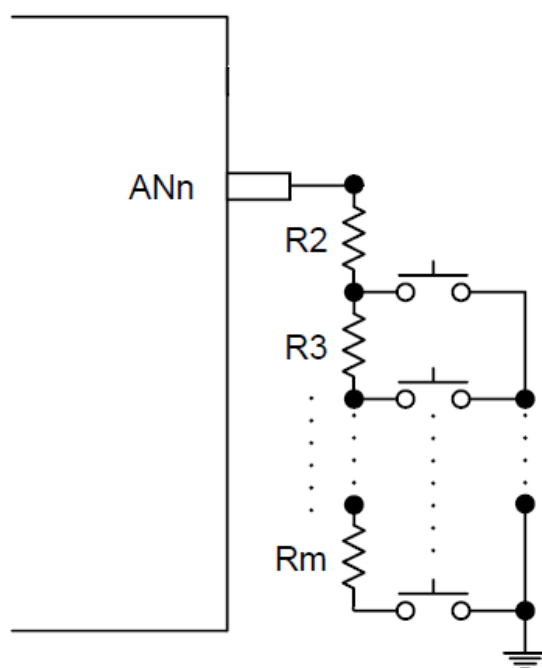


Figure 16-3 ADC 按鍵輸入串聯電阻應用參考圖



## 16.3 ADC 相關寄存器

### 16.3.1 ADC 控制寄存器 ADCC0、ADCC1

#### ADCC0

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	1	1
位元符號	ADCEN	ADCST	ADCIF	-	VREFO	VREFS	INREF_S[1:0]	

位編號	位元符號	說明
7	ADCEN	<p>ADC 模組電源控制位元</p> <p>0：關閉 ADC 轉換電源</p> <p>1：打開 ADC 轉換電源</p> <p>注意：1. 在掉電模式下，ADCEN 強制為 0。</p> <p>2. ADCEN 置 1 或切換轉換通道後，建議延時 20us 後再啟動 ADC 轉換。</p> <p>3. 啟動 ADC 轉換時，需要關閉 ADC 省電喚醒功能。</p>
6	ADCST	<p>ADC 啟動控制位</p> <p>0：轉換結束後，硬體自動清 0，在轉換過程中，軟體清 0 將終止轉換。</p> <p>1：啟動轉換</p> <p>注意：啟動轉換時，ADCIF 需要先清 0，ADCIF 位為 1 時，置 ADCST 不能啟動新的轉換。</p>
5	ADCIF	ADC 中斷標誌位元

		<p>0：無 ADC 轉換中斷</p> <p>1：轉換結束後，硬體置 1，可用於插斷要求（必須軟體清 0）</p>
4	-	保留位（讀為 0，寫無效）
3	VREFO	<p>VREF 輸出使能位</p> <p>0：VREF 不輸出</p> <p>1：從引腳 P0.4 輸出內部 VREF，此時需要設置 P0.4 為模擬輸入，而且 VREFS 必須為 0</p> <p>注：VREF 輸出驅動能力較弱，僅供測試使用。</p>
2	VREFS	<p>VREF 選擇</p> <p>0：選內部 VREF</p> <p>1：選外部 VREF（此時 P0.4 只做 ADC 參考電壓輸入，且必須設為模擬輸入）</p>
1-0	INREF_S	<p>ADC 內部參考電壓選擇位元</p> <p>00：VDD</p> <p>01：內部 4V</p> <p>10：內部 3V</p> <p>11：內部 2V</p> <p>注意：內部參考電壓選擇為 2V 時，VDD 電壓須高於 2.7V；內部參考電壓選擇 3/4V 時，VDD 須高於內部參考電壓 0.5V 以上。</p> <p>系統進入掉電模式前，建議將 ADC 參考電壓選擇非 VDD，可以進一步降低系統功耗。</p>

#### ADCC1

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W

復位值	0	0	0	0	0	0	0	0
位元符號	ICHS[1:0]		-	-	XCHS[3:0]			
號								

位編號	位元符號	說明
7-6	ICHS[1:0]	<p>ADC 內部輸入通道選擇</p> <p>00：禁止內部通道接入</p> <p>01：1/4VDD 作為 ADC 輸入通道</p> <p>10：保留位</p> <p>11：GND 接入</p> <p>注：在進行內部通道選擇時，外部通道選擇 XCHS[3:0]應配置為 1111，否則可能會造成內部通道和外部通道同時打開的情況。</p>
5-4	-	保留位（讀為 0，寫無效）
3-0	XCHS[3:0]	<p>ADC 外部輸入通道選擇</p> <p><math>XCHS[3:0] = x(x = 0 \dots 15)</math>，表示當前檢測通道為 <math>AN_x</math>，如 <math>XCHS[3:0] = 3</math>，表示當前檢測通道為外部通道 <math>AN_3</math>。外部通道除設置 <math>XCHS[3:0]</math>，還需設置對應管腳的功能為模擬輸入。</p>

## 16.3.2 ADC 控制寄存器 ADCC2、ADCC3

### ADCC2

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0

位元符號	ADCL	ALIGN	ADCTS[2:0]	ADCS[2:0]
------	------	-------	------------	-----------

位編號	位元符號	說明
7	ADCL	<p>ADC 轉換資料長度控制位元</p> <p>0 : ADC 轉換結果為 12 位元數據</p> <p>1 : ADC 轉換結果為 10 位元數據 ( 取 12 位元數據的高 10 位 )</p>
6	ALIGN	ADC 資料對齊方向控制位元，見下面的 ADC 轉換資料格式說明表
5-3	ADCTS [2:0]	<p>ADC 時鐘為 4MHZ 時，配置此 3bits 為 000；一次轉換需要 22 個 ADC_CLK</p> <p>ADC 時鐘為 2MHZ&amp;1MHZ 時，配置此 3bits 為 001 或者 010；一次轉換需要 19 個 ADC_CLK</p> <p>ADC 時鐘為&lt;1MHZ 時，配置此 3bits 為 011 或 100 或 101 或 110 或 111；一次轉換需要 15 個 ADC_CLK</p> <p>注：此配置位元只針對選擇內部參考電壓 1/2/3/4V，如果選擇 VDD 作參考電壓，無論 ADC 時鐘為多少，都可以選擇 15 個 ADC_CLK</p>
2-0	ADCS[2:0]	<p>ADC 時鐘選擇位元</p> <p>000 : <math>F_{osc}/2</math></p> <p>001 : <math>F_{osc}/4</math></p> <p>010 : <math>F_{osc}/6</math></p> <p>011 : <math>F_{osc}/8</math></p> <p>100 : <math>F_{osc}/12</math></p> <p>101 : <math>F_{osc}/16</math></p>

		110 : $F_{osc}/24$  111 : $F_{osc}/32$
--	--	--

ADC轉換資料格式說明表：

ADCL	ALIGN	ADCRH								ADCRL							
		7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	0	D11	D10	D9	D8	D7	D6	D5	D4	/	/	/	/	D3	D2	D1	D0
0	1	/	/	/	/	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
1	0	D11	D10	D9	D8	D7	D6	D5	D4	/	/	/	/	/	/	D3	D2
1	1	/	/	/	/	/	/	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2

### ADCC3

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	IVREFS	FCLKEN	ADCST_OEN	TRIGSEL[4:0]				

位編號	位元符號	說明
7	IVREFS	ADC 1V 內參選擇位  0：有 ADCC0 寄存器裡的 INREF_S 來決定  1：內參為 1V
6	FCLKEN	選擇內部參考電壓時，ADC 轉換的加速使能  0：使用內參的 ADC 可以最快工作在 2MHz 的頻率  1：使用內參的 ADC 可以最快工作在 4MHz 的頻率
5	ADCST_OEN	ADCST 輸出使能位  0：ADCST 信號從 P2.0 腳輸出禁止

		1 : ADCST 信號從 P2.0 腳輸出使能
4-0	TRIGSEL[4:0]	<p>ADC 觸發信號選擇位元</p> <p>00000 : ADC 轉換啟動僅由 ADCST(ADCC0.6)控制</p> <p>00001 : PWM0 上升沿</p> <p>00010 : PWM0 下降沿</p> <p>00011 : PWM0 中點</p> <p>00100 : PWM0 終點</p> <p>00101 : PWM1 上升沿</p> <p>00110 : PWM1 下降沿</p> <p>00111 : PWM1 中點</p> <p>01000 : PWM1 終點</p> <p>01001 : PWM2 上升沿</p> <p>01010 : PWM2 下降沿</p> <p>01011 : PWM2 中點</p> <p>01100 : PWM2 終點</p> <p>01101 : PWM0 匹配中斷</p> <p>01110 : ADC_ST 管腳上升沿</p> <p>01111 : ADC_ST 管腳下降沿</p> <p>10000 : ADC_ST 管腳雙沿</p> <p>10001 : 計時器 5 的 T5 通道發生一次輸入捕獲事件</p> <p>10010 : 計時器 5 的輸入捕獲通道 0 發生一次輸入捕獲事件</p> <p>10011 : 計時器 5 的輸入捕獲通道 1 發生一次輸入捕獲事件</p>

		<p>10100：計時器 5 發生一次比較匹配事件</p> <p>其他值：保留</p> <p>注意：1、PWM 中點觸發僅適用於中心對齊模式的 PWM 輸出</p> <p>2、當 ADCST 為 1 時（正在轉換），外部觸發信號不會影響 ADC 直到本次 ADC 轉換結束即 ADCST 被硬體清 0</p>
--	--	--

### 16.3.3 ADC 觸發延時計時器 ADCDLYH、ADCDLYL

#### ADCDLYH

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-				ADCDLY[11:8]			

位編號	位元符號	說明
7-4	-	保留位
3-0	ADCDLY[11:8]	ADC 外部觸發延時啟動計時器的高 4 位

#### ADCDLYL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	ADCDLY[7:0]							

位編號	位元符號	說明
7-0	ADCDLY[7:0]	ADC 外部觸發延時啟動計時器的低 8 位，用於在外部觸發啟動 ADC 之前插入一段延時，在延時計時結束時開始 ADC 轉換  延時時間=ADCDLY[11:0]* ADC 時鐘



### 16.3.4 ADC 模擬看門狗控制寄存器 AWDCON

#### AWDCON

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-			CONT	AWDIF	AWDIE	AWDMOD	AWDEN

位編號	位元符號	說明
7-5	-	保留位
4	CONT	<p>ADC 連續轉換使能</p> <p>通過 ADCC1 配置 ADC 的通道，配置 CONT=1，第一次 ADC 轉換需要軟體啟動，ADC 完成後會將 ADCIF 置位元，軟體通過查詢或中斷讀取結果，同時清零 ADCIF，硬體自動啟動下一次的轉換，直到轉換次數到了使用者配置的寄存器或使用者清零 CONT 或關閉 ADCEN</p>
3	AWDIF	<p>類比看門狗標誌位元</p> <p>該位元由硬體根據 AWDMOD 位元來設置，由軟體清除</p> <p>0：沒有發生模擬看門狗事件</p> <p>1：發生模擬看門狗事件</p>
2	AWDIE	<p>模擬看門狗中斷使能位</p> <p>該位元由軟體設置和清除，用於禁止或允許模擬看門狗。</p> <p>0：禁止模擬看門狗中斷</p> <p>1：允許模擬看門狗中斷。</p>

1	AWDMOD	<p>類比看門狗模式選擇位元</p> <p>0：如果被 ADC 轉換的類比電壓低於低閾值或高於高閾值，AWDIF 置位</p> <p>1：如果被 ADC 轉換的類比電壓高於低閾值且低於高閾值，AWDIF 置位</p>
0	AWDEN	<p>AWDEN：在 ADC 通道上開啟模擬看門狗</p> <p>該位元由軟體設置和清除。</p> <p>0：禁用模擬看門狗</p> <p>1：使用模擬看門狗</p>

### 16.3.5 ADC 高閾值比較寄存器 ADC\_HTRH、ADC\_HTRL

#### ADC\_HTRH

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	-				HTR[11:8]			

位編號	位元符號	說明
7-4	-	保留位
3-0	HTR[11:8]	ADC 高閾值比較寄存器高 4 位

#### ADC\_HTRL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符	HTR[7:0]							

號	
---	--

位編號	位元符號	說明
7-0	HTR[7:0]	ADC 高閾值比較寄存器低 8 位

### 16.3.6 ADC 低閾值比較寄存器 ADC\_LTRH、ADC\_LTRL

#### ADC\_LTRH

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號					LTR[11:8]			

位編號	位元符號	說明
7-4	-	保留位
3-0	LTR[11:8]	ADC 低閾值比較寄存器高 4 位

#### ADC\_LTRL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	LTR[7:0]							

位編號	位元符號	說明
7-0	LTR[7:0]	ADC 低閾值比較寄存器低 8 位

### 16.3.7 ADC 連續轉換次數寄存器 ADCCONTV

#### ADCCONTV

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	CONTV							

位編號	位元符號	說明
7-0	CONTV	ADC 連續轉換次數 當配置為 0 時，為一直連續轉換模式，只要不關閉 CONT 和 ADCEN

### 16.3.8 ADC 連續轉換間隔寄存器 ADCGAPV

#### ADCGAPV

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	GAPV							

位編號	位元符號	說明
7-0	GAPV	ADC 連續轉換間隔配置值 當間隔配置為 0 時，ADC 連續轉換次數到了，會置位元 ADCIF，然後硬體會自動進行下一次連續轉換。

		當間隔配置的值是非 0 時，ADC 連續轉換次數到了，會置位 ADCIF，然後 ADC 轉換不再進行，等待軟體開啟下一次轉換。
--	--	---

### 16.3.9 AD 轉換結果寄存器 SCRHX、SCRLX(x=0...7)

位編號	位元符號	說明
7-0	SCRHX[7:0]	檢測資料高8位
7-0	SCRLX[7:0]	檢測資料低8位

注：ADC 只使用 12/10 位元，也受資料對齊方向控制位元控制。

當連續轉換時，可以存放 8 個轉換後的值，第一個轉換的值同時存放在 SCRHO、SCRL0 和 ADCRL、ADCRH 中。

當不使能連續轉換時，SCRHX、SCRLX(x=1...7)可以當 RAM 使用，一共有 14 位元組。

### 16.3.10 ADC 轉換結果寄存器 ADCRL、ADCRH

#### ADCRL

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	ADCRL[7:0]							

#### ADCRH

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	ADCRH[7:0]							

號	
---	--

位編號	位元符號	說明
7-0	ADCRH[7:0]	ADC 轉換結果寄存器，見上面的 ADC 轉換資料格式說明表。
7-0	ADCRL[7:0]	

啟動 ADC 轉換步驟：

- (1) 使能 ADC 模組；
- (2) 選擇類比輸入通道、參考電壓、轉換時鐘、轉換結果對齊方式等；
- (3) ADCST 置 1 開始 ADC 轉換；
- (4) 等待 ADCST = 0 或者 ADCIF = 1，如果 ADC 中斷使能，則 ADC 中斷將會產生，使用者需要軟體  
清零 ADCIF；
- (5) 從 ADCRH/ADCRL 獲得轉換資料；
- (6) 重複步驟 3-5 開始另一次轉換。

### 16.3.11 ADC 喚醒控制寄存器 ADCWC

#### ADCWC

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R	R	R	R	R	R
復位值	0	0	0	0	0	0	0	0
位元符號	AMWEN	AMWIF	-					

位編號	位元符號	說明
7	AMWEN	<p>ADC 喚醒模組允許位元</p> <p>0：禁止 ADC 喚醒模組</p> <p>1：允許 ADC 喚醒模組</p> <p>注意：允許 ADC 喚醒模組時，需要關閉 ADC。</p>
6	AMWIF	<p>ADC 喚醒模組中斷標誌位元（與 ADC 轉換共用中斷向量）</p> <p>0：無 ADC 喚醒模組中斷，軟體清 0</p> <p>1：發生 ADC 喚醒模組中斷，硬體置 1</p> <p>注意：相應喚醒模組被禁止時，即使條件滿足，對應標誌也不會被置 1。</p> <p>需要使能 EADC。</p>
5-0	-	保留位

注：喚醒電阻通過埠上拉電阻來配置。

## 17 低電壓檢測/比較器

### 17.1 低電壓檢測/比較器特性

- 支持內部 VDD 多檔位元電壓檢測，並且可以產生中斷
- 支援埠電壓檢測，並且可以產生中斷或復位
- LVD 檔位：4.2V/3.9V/3.6V/3.0V/2.6V/2.4V/2.0V/1.9V
- 支援比較器功能

內部電壓檢測和 BOR 一樣，也是檢測 VDD 電壓，但獨立於 BOR，因此可以檢測比 BOR 設定電壓更高的各檔檢測電壓點，可通過寄存器設定其檢測電壓點、其是否工作、其是否允許中斷，方便客戶需求。

LVD 電壓檢測電路有一定的遲滯特性，遲滯電壓為 0.1V 左右。即當待檢測電壓下降到所選 LVD 電壓檔位元時 LVD 會產生插斷要求或重定，而待檢測電壓需要上升到 LVD 檔位元電壓+0.1V 時 LVD 插斷要求或復位才會解除。

LVD 檢測埠 P2.6 引腳上的電壓，當低於電壓檢測電壓點 0.8V 時，置相應標誌，若中斷允許，則可產生插斷要求；若中斷不允許，埠電壓檢測將產生重定。埠電壓檢測產生的有效中斷和重定可以將晶片從 PD 模式和 IDLE 模式喚醒。

LVD檢測VDD電壓時，不能產生重定，可通過有效中斷將晶片從PD模式和IDLE模式喚醒。



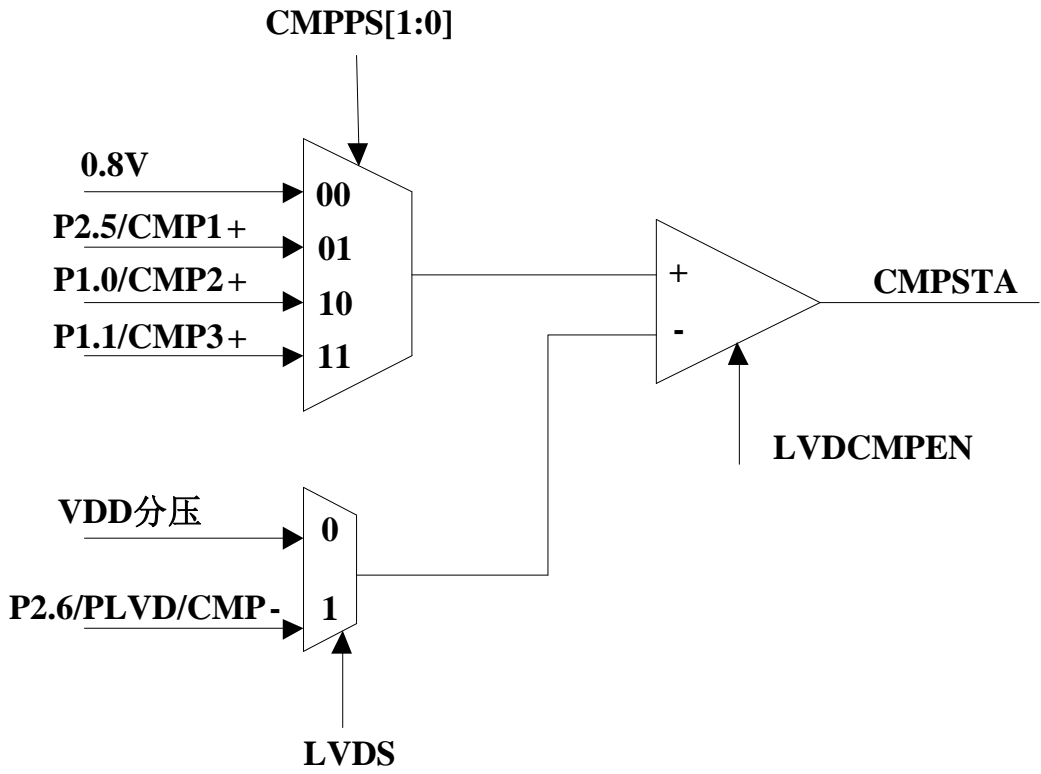


Figure 17-1 LVD/比較器框圖

## 17.2 低電壓檢測/比較器相關寄存器

### 17.2.1 控制寄存器 LVDC

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	LVDCMPEN	LVDS	LVDIE	-	LVDF	LVDV		

位編號	位元符號	說明
7	LVDCMPEN	LVD/比較器使能位 0：禁止 LVD/比較器

		1：允許 LVD/比較器
6	LVDS	LVD 檢測選擇位 0：檢測 VDD 電壓 1：檢測 P2.6 埠電壓 ( 0.8V , ±5% )
5	LVDIE	LVD 中斷允許位 0：禁止 LVD 中斷 1：允許 LVD 中斷 注意：禁止時，只要允許檢測，LVDF 也可被置 1，但即使此時 EA 被置 1，也不會產生插斷要求。 當 LVDS 為 1，檢測 P2.6 埠電壓時： LVDIE=0：埠電壓檢測重定 LVDIE=1：埠電壓檢測中斷
4	-	保留位
3	LVDF	低電壓檢測標誌位元 0：必須軟體清 0 1：VDD 電壓低於檢測選擇電壓時，硬體置 1，也做插斷要求 注意：VDD 電壓低於檢測電壓時間大於 LVDDBC 寄存器設置的消抖時間後才會置 LVDF，高於檢測電壓時，不會自動清除該位元，必須軟體清除，只有在 VDD 電壓持續高於檢測電壓時，軟體清除才能起作用，如果 VDD 電壓持續持續低於檢測電壓，軟體是無法清除 LVDF 的。
2-0	LVDV[2:0]	VDD 電壓檢測電壓點選擇位元 000：1.9V

		001 : 2.0V  010 : 2.4V  011 : 2.6V  100 : 3.0V  101 : 3.6V  110 : 3.9V  111 : 4.2V  注意：LVD 檢測電壓設置在 BOR 電壓之上才有意義。
--	--	--

### 17.2.2 比較功能控制寄存器 LVDCMP

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R/W	R	R/W	R/W	R/W	R/W
復位值	0	0	1	0	0	0	0	0
位元符號	-	-	DBEN	CMPSTA	CMPIM[1:0]		CMPPS[1:0]	

位編號	位元符號	說明
7-6	-	保留位
5	DBEN	消抖使能  0：不消抖  1：消抖  注意：  1. LVD 和 CMP 都適用；

		2. 掉電模式和空間模式下自動不消抖，退出掉電模式和空間模式有 DBEN 來控制是否消抖
4	CMPSTA	比較器輸出狀態 0：比較器正端電壓小於負端電壓 1：比較器正端電壓大於負端電壓
3-2	CMPIM[1:0]	當 CMPPS[1:0]不選擇 00 時，需要配置這個寄存器： 00：不置位 LVDF 01：CMP+從小於 CMP-到大於 CMP-後會置位 LVDF； 10：CMP+從大於 CMP-到小於 CMP-後會置位 LVDF； 11：CMP+從小於 CMP-到大於 CMP-或 CMP+從大於 CMP-到小於 CMP-都置位 LVDF 當作為 PWM FLT 來源時，需要配置此寄存器為 01 或 10； 01：CMP+的電壓大於 CMP-的電壓，會產生一個 PWM FLT 高的標誌； 10：CMP+的電壓小於 CMP-的電壓，會產生一個 PWM FLT 高的標誌；
1-0	CMPPS[1:0]	比較器正端選擇位 00：0.8V 01：CMP1 管腳為比較器的正端輸入 10：CMP2 管腳為比較器的正端輸入 11：CMP3 管腳為比較器的正端輸入

### 17.2.3 去抖控制寄存器 LVDDBC

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	LVDDBC[7:0]							

位編號	位元符號	說明
7-0	LVDDBC[7:0]	LVD 消抖控制位 $\text{消抖時間} = \text{LVDDBC}[7:0] * 8T_{\text{CPU}} + 2T_{\text{CPU}}$

注意：掉電模式和空閒模式下自動關閉，退出掉電模式和空閒模式又自動打開。

# 18 軟體LCD

## 18.1 LCD 特性

- 支持 1/2Bias 的 LCD 點陣
- 驅動能力可配置
- COM 口數量可任意配置
- LCD 控制信號 ( COM 和 SEG ) 由軟體程式實現

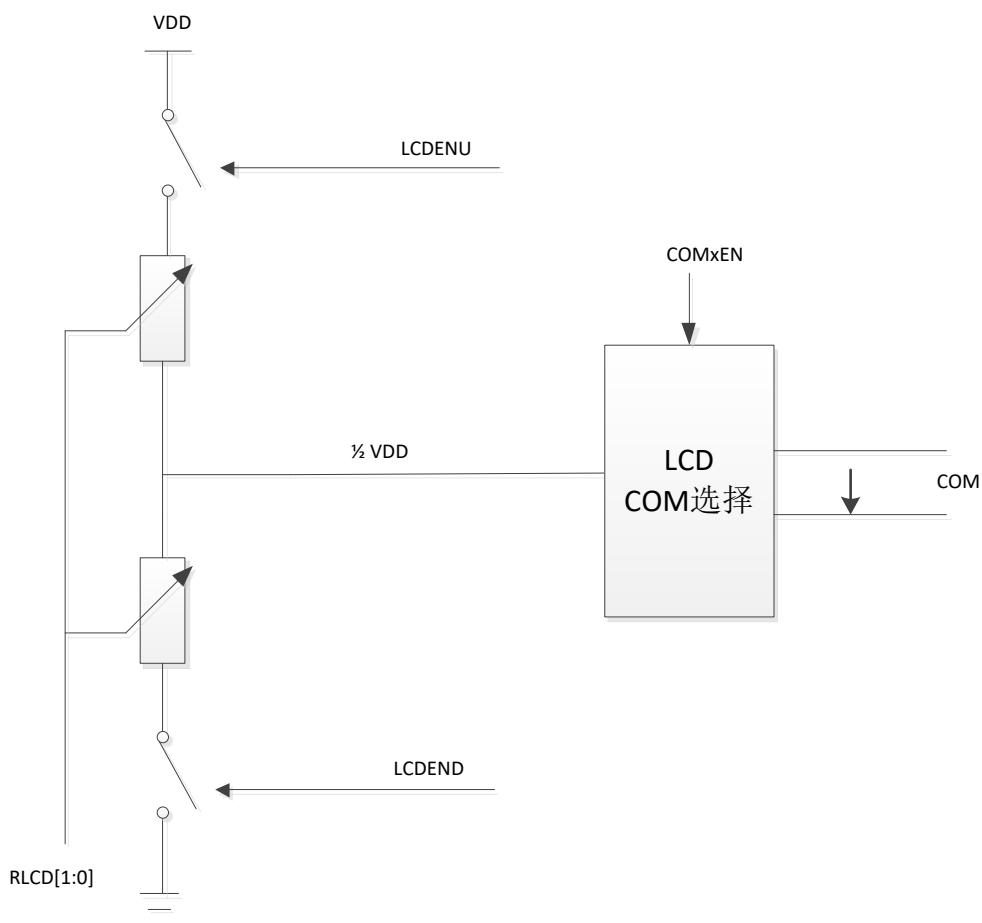


Figure 18-1 LCD 系統框圖

## 18.2 LCD 幀

一個完整的LCD波形週期包含兩個Frame，即Frame0和Frame1。

### Frame 0

在Frame0中，COM信號輸出可以是VDD，或是VBIAS=1/2VDD；

在Frame0中，SEG信號輸出可以是GND，或是VDD。

### Frame 1

在Frame1中，COM信號輸出可以是GND，或是VBIAS=1/2VDD；

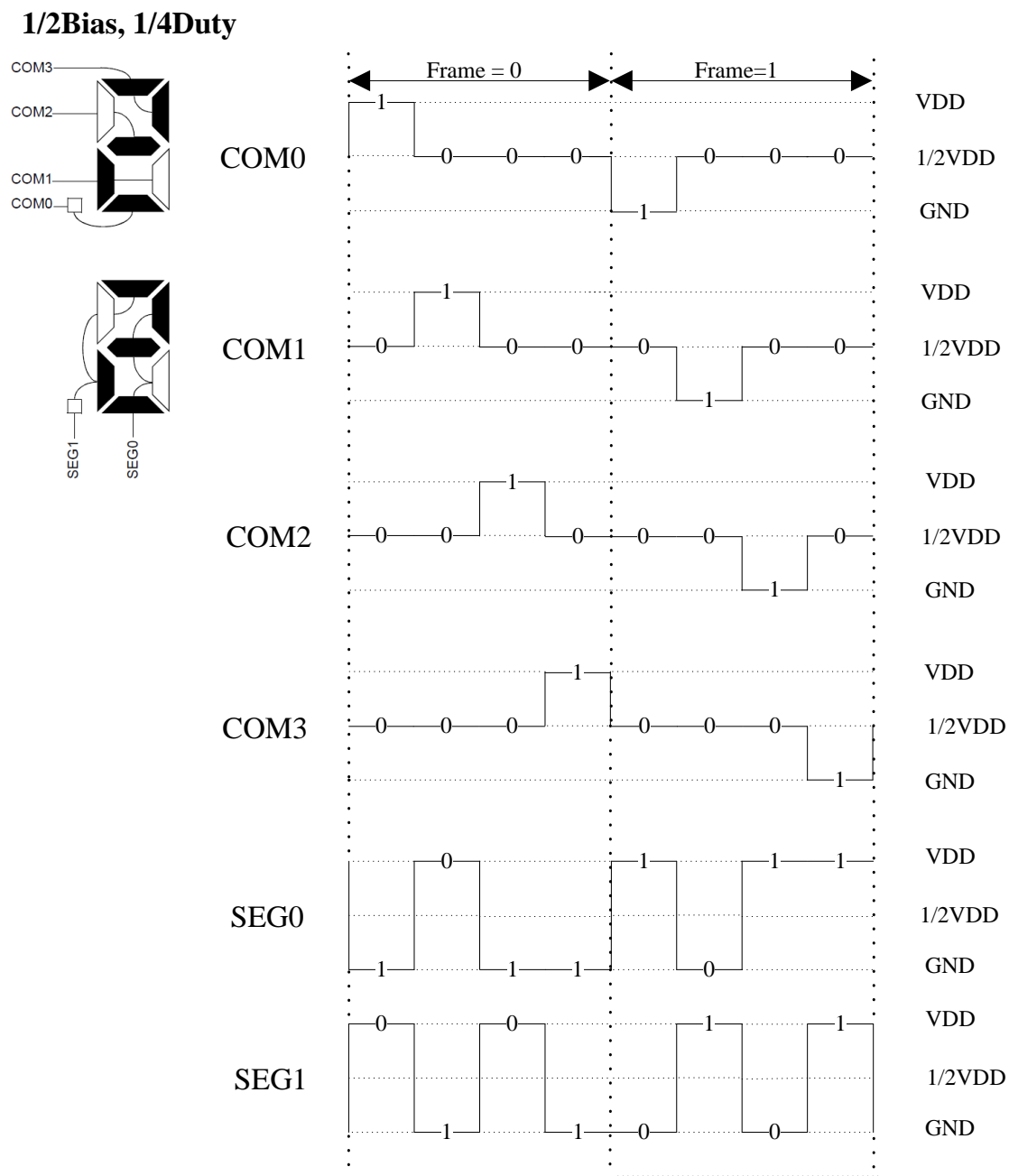
在Frame1中，SEG信號輸出可以是VDD，或是GND。

通過軟體設定相應的I/O資料寄存器來決定COM口輸出的是VDD，GND或VBIAS。1/2VDD是通過LCD模組的分壓輸出到IO上，此時IO的模式配置為類比輸入，相應的COMPxEN使能；而VDD、GND是通過IO口的強推挽輸出1和0來實現。

通過軟體設定相應的I/O資料寄存器來決定SEG口目前輸出VDD或GND。設置為SEG口的IO是通過IO口的強推挽輸出1和0來實現。

下面的波形圖顯示了一個利用應用程式產生的典型1/2Bias LCD波形。其中寫“1”代表點亮LCD。

COM<sub>n</sub>和SEG<sub>m</sub>引腳上所產生的COM和SEG信號極性（0或1）通過相應的埠資料寄存器位元來產生。



注：圖形中的邏輯值為 COM 或 SEG 對應埠資料寄存器的位元值。

Figure 18-2 1/2bias LCD 波形圖





## 18.3 LCD 相關寄存器

### 18.3.1 LCD 控制寄存器 LCDCON

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R
復位值	0	0	0	0	0	0	0	0
位元符號	LCDENU	LCDEND	RLCD1	RLCD0	-	-		

位編號	位元符號	說明
7	LCDENU	LCD 上使能控制位 0：禁止 1：使能
6	LCDEND	LCD 下使能控制位 0：禁止 1：使能
5-4	RLCD[1:0]	軟體 LCD 電阻選擇位 00：10kΩ 01：25kΩ 10：75kΩ 11：175kΩ
3-0	-	保留

### 18.3.2 COM 口使能控制寄存器 COMP0EN-COMP2EN

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	COMPxEN[7:0]							

位編號	位元符號	說明
7-0	COMPxENy	<p>軟體 LCD COM 功能使能位</p> <p>0：禁止，普通 IO</p> <p>1：使能</p> <p>注：x = 0~2 y = 0-7</p>

## 19 迴圈冗餘校驗CRC

### 19.1 CRC 特性

- 16 位 CRC
- CRC 生成校驗遵從 CRC-CCITT 多項式，即 0x1021
- 初值可設為 0x0000 或 0xFFFF
- 計算與結果共用同一寄存器

每一次讀出資料寄存器[CRCH: CRCL]，其值都為最後的CRC計算結果。

可通過設置寄存器 CRCC 的 CRCRSV 位來選擇計算初值，但不會影響之前的 CRC 計算資料，只有置寄存器 CRCC 的 CRCRST 位後，才會復位 CRC 計算器，後寫入的資料將以新的初值計算 CRC 結果。

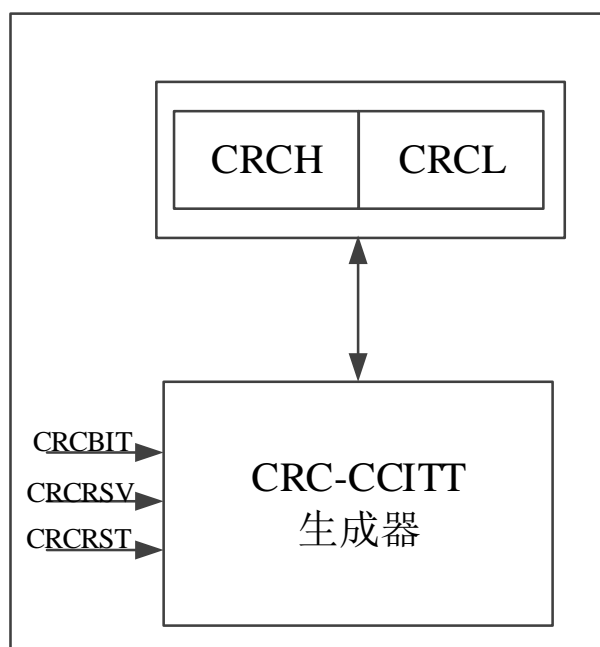


Figure 19-1 CRC 功能框圖

## 19.2 CRC 相關寄存器

### 19.2.1 CRC 控制寄存器 CRCC

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R	R/W	R/W	W
復位值	0	0	0	0	0	0	0	0
位元符號	-					CRCBIT	CRCRSV	CRCRST

位編號	位元符號	說明
7-3	-	保留位 ( 讀為 0 , 寫無效 )
2	CRCBIT	CRC BIT 翻轉控制位 0 : MSB first 1 : LSB first
1	CRCRSV	CRC 復位初值選擇位 0 : 復位初值為 0x0000 1 : 復位初值為 0xFFFF
0	CRCRST	CRC 計算器復位控制位 置 1 復位 CRC 計算器 , 硬體自動清 0

### 19.2.2 CRC 資料寄存器 CRCL、CRCH

#### CRCL

位編號	7	6	5	4	3	2	1	0
-----	---	---	---	---	---	---	---	---

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	CRCL[7:0]							

位編號	位元符號	說明
7-0	CRCL[7:0]	<p>寫入資料時為 CRC 計算器的輸入資料</p> <p>讀出數據時為 CRC 計算結果的低位元組</p> <p>注意：寫入資料時，自動啟動 CRC 計算，完後自動關閉。</p>

#### CRCH

位編號	7	6	5	4	3	2	1	0
R/W	R	R	R	R	R	R	R	R
復位值	0	0	0	0	0	0	0	0
位元符號	CRCH[7:0]							

位編號	位元符號	說明
7-0	CRCH[7:0]	<p>對該寄存器寫入資料無效</p> <p>讀出數據時為 CRC 計算結果的高位元組</p>

注意：每一次寫入待計算資料，其計算結果是由與前一次計算結果共同生成。

## 20 代碼選項

### 1. 外部復位使能

- P2.7 為外部復位引腳 ( 默認 )。該口作為外部復位引腳時，無法作為普通 I/O 使用
- P2.7 為普通 IO 引腳

### 2. BOR 使能

- 默認 BOR 使能
- 通過代碼選項可將 BOR 關閉，這樣晶片的啟動電壓可以低於 1.8V

### 3. BOR 檢測電壓

- 1.8V
- 2.0V
- 2.4V
- 2.6V
- 3.0V
- 3.6V
- 3.9V
- 4.2V

### 4. 復位後等待時間

- 1ms
- 4ms
- 8ms ( 默認 )
- 16ms

### 5. 第二重定向量配置

使用者可以通過此選擇來配置自訂啟動代碼的首位址，配置值必須以 1K 位元組為單位，即第二重定向量位址的最低 10 位必須為全零，默認不使能第二復位向量。

## 21 指令表

助記符		指令說明	位元組	週期	代碼
資料傳遞類指令					
MOV	A , Rn	寄存器送到累加器	1	1	0xE8-0xEF
MOV	A , direct	直接位址資料送到累加器	2	2	0xE5
MOV	A , @Ri	間接 RAM 資料送到累加器	1	2	0xE6-0xE7
MOV	A , #data	立即數送到累加器	2	2	0x74
MOV	Rn , A	累加器送到寄存器	1	1	0xF8-0xFF
MOV	Rn , direct	直接位址資料送到寄存器	2	2	0xA8-0xAF
MOV	Rn , #data	立即數送到寄存器	2	2	0x78-0x7F
MOV	direct , Rn	寄存器送到直接位址	2	2	0x88-0x8F
MOV	direct , direct	直接位址資料送到直接位址	3	3	0x85
MOV	direct , A	累加器送到直接地址	2	2	0xF5
MOV	direct , @Ri	間接 RAM 資料送到直接位址	2	2	0x86-0x87
MOV	direct , #data	立即數送到直接地址	3	3	0x75
MOV	@Ri , A	累加器送到間接 RAM	1	1	0xF6-0xF7
MOV	@Ri , direct	直接位址資料送到間接 RAM	2	2	0xA6-0xA7
MOV	@Ri , #data	立即數送到間接 RAM	2	2	0x76-0x77
MOV	DPTR , #data16	16 位元立即數載入到資料指標	3	3	0x90
MOVC	A , @A+DPTR	代碼位元組送到累加器	1	3	0x93
MOVC	A , @A+PC	代碼位元組送到累加器	1	3	0x83



MOVX	A , @Ri	外部 RAM(8 位位址)送到累加器	1	3	0xE2-0xE3
MOVX	A , @DPTR	外部 RAM(16 位位址)送到累加器	1	3	0xE0
MOVX	@Ri , A	累加器送到外部 RAM(8 位位址)	1	3	0xF2-0xF3
MOVX	@DPTR , A	累加器送到外部 RAM(16 位位址)	1	3	0xF0
PUSH	direct	直接位址資料壓入堆疊	2	2	0xC0
POP	direct	直接位址資料彈出堆疊	2	2	0xD0
XCH	A,Rn	寄存器和累加器交換	1	1	0xC8-0xCF
XCH	A, direct	直接位址資料和累加器交換	2	2	0xC5
XCH	A, @Ri	間接 RAM 和累加器交換	1	2	0xC6-0xC7
XCHD	A, @Ri	間接 RAM 和累加器交換低 4 位元位元組	1	2	0xD6-0xD7
算數運算類指令					
INC	A	累加器加 1	1	1	0x04
INC	Rn	寄存器加 1	1	1	0x08-0x0F
INC	direct	直接位址資料加 1	2	2	0x05
INC	@Ri	間接 RAM 加 1	1	2	0x06-0x07
INC	DPTR	數據指標加 1	1	1	0xA3
DEC	A	累加器減 1	1	1	0x14
DEC	Rn	寄存器減 1	1	1	0x18-0x1F
DEC	direct	直接位址資料減 1	2	2	0x15
DEC	@Ri	間接 RAM 減 1	1	2	0x16-0x17
MUL	AB	累加器和 B 寄存器相乘	1	4	0xA4

DIV	AB	累加器除以 B 寄存器	1	4	0x84
DA	A	累加器十進位調整	1	1	0xD4
ADD	A,Rn	寄存器與累加器求和	1	1	0x28-0x2F
ADD	A,direct	直接位址資料與累加器求和	2	2	0x25
ADD	A,@Ri	間接 RAM 與累加器求和	1	2	0x26-0x27
ADD	A,#data	立即數與累加器求和	2	2	0x24
ADDC	A,Rn	寄存器與累加器求和 ( 帶進位 )	1	1	0x38-0x3F
ADDC	A,direct	直接位址資料與累加器求和 ( 帶進位 )	2	2	0x35
ADDC	A,@Ri	間接 RAM 與累加器求和 ( 帶進位 )	1	2	0x36-0x37
ADDC	A,#data	立即數與累加器求和 ( 帶進位 )	2	2	0x34
SUBB	A,Rn	累加器減去寄存器 ( 帶借位 )	1	1	0x98-0x9F
SUBB	A,direct	累加器減去直接位址資料 ( 帶借位 )	2	2	0x95
SUBB	A,@Ri	累加器減去間接 RAM ( 帶借位 )	1	2	0x96-0x97
SUBB	A,#data	累加器減去立即數 ( 帶借位 )	2	2	0x94
<b>邏輯運算類指令</b>					
ANL	A,Rn	寄存器“與”到累加器	1	1	0x58-0x5F
ANL	A,direct	直接位址資料“與”到累加器	2	2	0x55
ANL	A,@Ri	間接 RAM“與”到累加器	1	2	0x56-0x57
ANL	A,#data	立即數“與”到累加器	2	2	0x54
ANL	direct,A	累加器“與”到直接地址	2	2	0x52
ANL	direct, #data	立即數“與”到直接地址	3	3	0x53
ORL	A,Rn	寄存器“或”到累加器	1	1	0x48-0x4F

ORL	A,direct	直接位址資料“或”到累加器	2	2	0x45
ORL	A,@Ri	間接 RAM“或”到累加器	1	2	0x46-0x47
ORL	A,#data	立即數“或”到累加器	2	2	0x44
ORL	direct,A	累加器“或”到直接地址	2	2	0x42
ORL	direct, #data	立即數“或”到直接地址	3	3	0x43
XRL	A,Rn	寄存器“異或”到累加器	1	1	0x68-0x6F
XRL	A,direct	直接位址資料“異或”到累加器	2	2	0x65
XRL	A,@Ri	間接 RAM“異或”到累加器	1	2	0x66-0x67
XRL	A,#data	立即數“異或”到累加器	2	2	0x64
XRL	direct,A	累加器“異或”到直接地址	2	2	0x62
XRL	direct, #data	立即數“異或”到直接地址	3	3	0x63
CLR	A	累加器清零	1	1	0xE4
CPL	A	累加器求反	1	1	0xF4
RL	A	累加器迴圈左移	1	1	0x23
RLC	A	帶進位累加器迴圈左移	1	1	0x33
RR	A	累加器迴圈右移	1	1	0x03
RRC	A	帶進位累加器迴圈右移	1	1	0x13
SWAP	A	累加器高、低 4 位交換	1	1	0xC4
<b>控制轉移類指令</b>					
JMP	@A+DPTR	相對 DPTR 的無條件間接轉移	1	2	0x73
JZ	rel	累加器為 0 則轉移	2	3	0x60
JNZ	rel	累加器為 1 則轉移	2	3	0x70

CJNE	A,direct,rel	比較直接地址和累加器，不相等轉移	3	4	0xB5
CJNE	A,#data,rel	比較立即數和累加器，不相等轉移	3	4	0xB4
CJNE	Rn,#data,rel	比較寄存器和立即數，不相等轉移	2	4	0xB8-0xBF
CJNE	@Ri,#data,rel	比較立即數和間接 RAM，不相等轉移	3	5	0xB6-0xB7
DJNZ	Rn,rel	寄存器減 1，不為 0 則轉移	2	3	0xD8-0xDF
DJNZ	direct,rel	直接位址資料減 1，不為 0 則轉移	3	4	0xD5
NOP		空操作，用於短暫延時	1	1	0x00
ACALL	add11	絕對調用副程式	2	2	xxx10001b
LCALL	add16	長調用副程式	3	3	0x12
RET		從副程式返回	1	4	0x22
RETI		從中斷服務副程式返回	1	4	0x32
AJMP	add11	無條件絕對轉移	2	2	xxx00001b
LJMP	add16	無條件長轉移	3	3	0x02
SJMP	rel	無條件相對轉移	2	3	0x80
布林指令					
CLR	C	清零進位位	1	1	0xC3
CLR	bit	清零直接定址位	2	2	0xC2
SETB	C	置位進位位	1	1	0xD3
SETB	bit	置位直接定址位	2	2	0xD2
CPL	C	取反進位位	1	1	0xB3
CPL	bit	取反直接定址位	2	2	0xB2
ANL	C,bit	直接定址位“與”到進位位	2	2	0x82

ANL	C , /bit	直接定址位的反碼“與”到進位位	2	2	0xB0
ORL	C,bit	直接定址位“或”到進位位	2	2	0x72
ORL	C , /bit	直接定址位的反碼“或”到進位位	2	2	0xA0
MOV	C,bit	直接定址位送到進位位	2	2	0xA2
MOV	bit, C	進位位送到直接定址	2	2	0x92
JC	rel	如果進位位為 1 則轉移	2	3	0x40
JNC	rel	如果進位位為 0 則轉移	2	3	0x50
JB	bit , rel	如果直接定址位為 1 則轉移	3	4	0x20
JNB	bit , rel	如果直接定址位為 0 則轉移	3	4	0x30
JBC	bit , rel	直接定址位為 1 則轉移並清除該位	3	4	0x10

## 22 電氣特性

除非另外說明，以下資料測試條件均為：VDD=5.0V，GND=0V，25°C。

### 20.1 極限參數

參數	符號	最小值	典型值	最大值	單位
直流供電電壓	VDD	-0.3	-	+6.0	V
輸入/輸出電壓	V <sub>I</sub> /V <sub>O</sub>	GND-0.3	-	VDD+0.3	V
工作環境溫度	T <sub>OTG</sub>	-40	-	+105	°C
存儲溫度	T <sub>STG</sub>	-55	-	+125	°C

注：(1) 流過 VDD 的最大電流值在 5.0V，25°C 下須小於 100mA。

(2) 流過 GND 的最大電流值在 5.0V，25°C 下須小於 150mA。

### 20.2 DC 特性

參數	符號	條件 ( VDD=5V )	最小值	典型值	最大值	單位
工作電壓	VDD	F <sub>CPU</sub> =16MHz 或 44KHz，ADC 模組關閉	1.8	5.0	5.5	V
工作電流	I <sub>OP1</sub>	F <sub>OSC</sub> =32MHz，F <sub>CPU</sub> =16MHz，無負載， 無浮動輸入管腳，執行 NOP 指令，其它 模組關閉	-	3.3	-	mA
		F <sub>OSC</sub> =32MHz，F <sub>CPU</sub> =8MHz，無負載，無 浮動輸入管腳，執行 NOP 指令，其它模	-	2.5	-	

		組關閉				
		F <sub>OSC</sub> =32MHz , F <sub>CPU</sub> =4MHz , 無負載 , 無浮動輸入管腳 , 執行 NOP 指令 , 其它模	-	2.0	-	
		組關閉				
		F <sub>OSC</sub> =32MHz , F <sub>CPU</sub> =2MHz , 無負載 , 無浮動輸入管腳 , 執行 NOP 指令 , 其它模	-	1.8	-	
		組關閉				
		F <sub>OSC</sub> =32MHz , F <sub>CPU</sub> =1MHz , 無負載 , 無浮動輸入管腳 , 執行 NOP 指令 , 其它模	-	1.7	-	
		組關閉				
		F <sub>OSC</sub> =32MHz , F <sub>CPU</sub> =512KHz , 無負載 , 無浮動輸入管腳 , 執行 NOP 指令 , 其它模	-	1.6	-	
		模組關閉				
		F <sub>OSC</sub> =16MHz , F <sub>CPU</sub> =16MHz , 無負載 , 無浮動輸入管腳 , 執行 NOP 指令 , 其它模		2.6		
		模組關閉				
		F <sub>OSC</sub> =16MHz , F <sub>CPU</sub> =8MHz , 無負載 , 無浮動輸入管腳 , 執行 NOP 指令 , 其它模		1.9		
		組關閉				
		F <sub>OSC</sub> =16MHz , F <sub>CPU</sub> =4MHz , 無負載 , 無浮動輸入管腳 , 執行 NOP 指令 , 其它模		1.4		
		組關閉				

		F <sub>OSC</sub> =16MHz , F <sub>CPU</sub> =2MHz , 無負載 , 無浮動輸入管腳 , 執行 NOP 指令 , 其它模組關閉		1.2	
		F <sub>OSC</sub> =16MHz , F <sub>CPU</sub> =1MHz , 無負載 , 無浮動輸入管腳 , 執行 NOP 指令 , 其它模組關閉		1.1	
		F <sub>OSC</sub> =16MHz , F <sub>CPU</sub> =512KHz , 無負載 , 無浮動輸入管腳 , 執行 NOP 指令 , 其它模組關閉		1.0	
		F <sub>OSC</sub> =8MHz , F <sub>CPU</sub> =8MHz , 無負載 , 無浮動輸入管腳 , 執行 NOP 指令 , 其它模組關閉		1.5	
		F <sub>OSC</sub> =8MHz , F <sub>CPU</sub> =4MHz , 無負載 , 無浮動輸入管腳 , 執行 NOP 指令 , 其它模組關閉		1.0	
		F <sub>OSC</sub> =8MHz , F <sub>CPU</sub> =2MHz , 無負載 , 無浮動輸入管腳 , 執行 NOP 指令 , 其它模組關閉		0.8	
		F <sub>OSC</sub> =8MHz , F <sub>CPU</sub> =1MHz , 無負載 , 無浮動輸入管腳 , 執行 NOP 指令 , 其它模組關閉		0.7	
		F <sub>OSC</sub> =8MHz , F <sub>CPU</sub> =512KHz , 無負載 ,		0.6	



		無浮動輸入管腳，執行 NOP 指令，其它 模組關閉				
		F <sub>OSC</sub> =4MHz，F <sub>CPU</sub> =4MHz，無負載，無 浮動輸入管腳，執行 NOP 指令，其它模 組關閉		0.9		
		F <sub>OSC</sub> =4MHz，F <sub>CPU</sub> =2MHz，無負載，無 浮動輸入管腳，執行 NOP 指令，其它模 組關閉		0.6		
		F <sub>OSC</sub> =4MHz，F <sub>CPU</sub> =1MHz，無負載，無 浮動輸入管腳，執行 NOP 指令，其它模 組關閉		0.5		
		F <sub>OSC</sub> =4MHz，F <sub>CPU</sub> =512KHz，無負載， 無浮動輸入管腳，執行 NOP 指令，其它 模組關閉		0.4	-	
	I <sub>OP2</sub>	F <sub>OSC</sub> =44KHz，無負載，無浮動輸入管腳， 執行 NOP 指令，其它模組關閉	-	70	-	μA
	I <sub>PD</sub>	進入掉電模式，無負載，無浮動輸入管 腳，所有模組關閉，ADC 參考電壓選擇 非 VDD	-	0.8	-	μA
	I <sub>IDLE1</sub>	F <sub>OSC</sub> =16MHz，進入空閒模式，無負載， 無浮動輸入管腳，所有模組關閉	-	1.0	-	mA
	I <sub>IDLE2</sub>	F <sub>OSC</sub> =8MHz，進入空閒模式，無負載，	-	0.6	-	mA

		無浮動輸入管腳，所有模組關閉				
	I <sub>IDLE3</sub>	F <sub>OSC</sub> = 4MHz，進入空間模式，無負載， 無浮動輸入管腳，所有模組關閉	-	0.4	-	mA
	I <sub>IDLE4</sub>	F <sub>OSC</sub> = 44KHz，進入空間模式，無負載， 無浮動輸入管腳，所有模組關閉， 內部高頻 RC 時鐘關閉	-	63	-	μA
掉電定時中斷喚醒電流	I <sub>PW</sub>	F <sub>CPU</sub> = 16MHz，關閉 BOR，TIMER3 計數 時鐘源選擇 RC44K，系統進入掉電， TIMER3 定時 1S 中斷喚醒的平均電流	-	2	-	μA
WDT 電流	I <sub>WDT</sub>	VDD = 5V	-	2.0	-	μA
LVD 電流	I <sub>LVD</sub>	VDD = 5V	-	8.0	-	
BOR 電流	I <sub>BOR</sub>	VDD = 5V	-	8.0	-	μA
輸入低電壓 1	V <sub>IL1</sub>	I/O 埠非施密特輸入	GND	-	0.3*V <sub>DD</sub>	V
輸入高電壓 1	V <sub>IH1</sub>	I/O 埠非施密特輸入	0.7*VDD	-	VDD	V
輸入低電壓 2	V <sub>IL2</sub>	I/O 埠施密特輸入	GND	-	0.2*V <sub>DD</sub>	V
輸入高電壓 2	V <sub>IH2</sub>	I/O 埠施密特輸入	0.8*VDD	-	VDD	V
輸入漏電流	I <sub>ILC</sub>	I/O埠輸入模式，V <sub>IN</sub> = VDD 或GND	-1	0	1	μA
輸出漏電流	I <sub>OLC</sub>	I/O埠輸出模式，V <sub>OUT</sub> = VDD 或GND	-1	0	1	μA
灌電流	I <sub>OL</sub>	V <sub>out</sub> = GND + 0.6	-	25	-	mA
拉電流	I <sub>OH</sub>	V <sub>out</sub> = VDD - 0.6	-	21	-	
上拉電阻	R <sub>PU1</sub>	P0.2 埠，V <sub>IN</sub> = GND	-	50	-	kΩ
	R <sub>PU2</sub>	P0.2 埠，V <sub>IN</sub> = GND	-	100	-	

	R <sub>PU3</sub>	P0.2 埠, VIN=GND	-	150	-	
	R <sub>PU4</sub>	P0.2 埠, VIN=GND	-	300	-	
	R <sub>PU5</sub>	普通埠, VIN=GND	-	50	-	
下拉電阻	R <sub>PD</sub>	普通埠, VIN=VDD	-	50	-	
上下拉電阻值	R <sub>PUPD</sub>	P2.3、P2.4、P2.5、P2.7 埠, VIN=GND	-	上拉:60 下拉:55	-	kΩ
ADC 喚醒電壓	V <sub>AW1</sub>	常溫, VDD = 5V	4.0	4.2	4.4	V
	V <sub>AW2</sub>	常溫, VDD = 3V	2.3	2.5	2.6	
RAM 保持電壓	V <sub>RAM</sub>	-	-	0.7	-	

注：除非另外說明，以上資料測試條件均為：VDD=5.0V，GND=0V，25°C。

## 20.3 AC 特性

參數	符號	條件	最小值	典型值	最大值	單位
內部 RC32M 啟動時間	Tset1	常溫, VDD=5V	-	-	5	μs
內部 RC44K 啟動時間	Tset2	常溫, VDD=5V	-	-	150	μs
外部高頻振盪器 啟動時間	Tset3	16MHz, 常溫, VDD=5V	-	200	-	μs
外部高頻振盪器 工作電壓	Vset3	16MHz	2.5	-	5.5	V
外部低頻振盪器 啟動時間	Tset4	常溫, VDD=5V	-	2	-	s
頻率精度	FIRC1	VDD=2V~5.5V, 25°C	32(1-1%)	32	32(1+1%)	MHz

	FIRC2	VDD=5.0V, -20°C ~ +105°C	32(1-2%)	32	32(1+2%)	MHz
	FIRC2	VDD=5.0V, -40°C ~ +105°C	32(1-5%)	32	32(1+5%)	MHz
	FWRC	-	31	44	58	KHz

## 20.4 ADC 特性

參數	符號	條件	最小值	典型值	最大值	單位
供電電壓	VAD	-	1.8	5.0	5.5	V
精度	NR	$GND \leq VAIN \leq V_{ref}$	-	10	12	bit
ADC 輸入電壓	VAIN	-	GND	-	Vref	V
ADC 輸入電阻	RAIN	VAIN=5V	2	-	-	MΩ
類比電壓源推薦阻抗	ZAIN	-	-	-	10	kΩ
ADC 轉換電流	IAD	ADC 模組打開, VDD=5.0V	-	0.6	1	mA
ADC 輸入電流	IADIN	VDD=5.0V	-	-	10	μA
微分非線性誤差	DLE	VDD=5.0V	-	-	±2	LSB
積分非線性誤差 (1MHz 轉換頻率)	ILE	VDD=5.0V, Vref=1V	-	-	-5~2	LSB
		VDD=5.0V, Vref=2V	-	-	-5~2	
		VDD=5.0V, Vref=3V	-	-	-4~2	
		VDD=5.0V, Vref=4V	-	-	-3~2	
		VDD=5.0V, Vref=VDD	-	-	±2	
		VDD=5.0V, Vref=外參	-	-	±2	
滿刻度誤差	EF	VDD=5.0V	-	-	±5	LSB
偏移量誤差	EZ	VDD=5.0V	-	-	±3	LSB
總絕對誤差	EAD	VDD=5.0V	-	-	±5	LSB

總轉換時間 1	TCON1	VDD=5.0V Vref =2/3/4V	10	-	-	μs
總轉換時間 2	TCON2	VDD=5.0V Vref =VDD	2	-	-	μs
內部參考電壓	VADREF	VDD=5.0V , Vref =2V	2(1-1%)	2	2(1+1%)	V

## 20.5 FLASH 特性

參數	符號	條件	最小值	典型值	最大值	單位
讀寫測試	NENDUR	-	100000	-	-	Cycle
資料保存時間	T <sub>RET</sub>	T=25°C	-	10	-	year
磁區擦除時間	T <sub>ERASE</sub>	1 個磁區 ( 128 位元組 )	-	5	-	ms
位元組寫入時間	T <sub>PROG</sub>	1 個位元組 , Fcpu=16MHz	-	23	-	us
讀取耗電流	I <sub>DD1</sub>	Fcpu=16MHz	-	4	-	mA
寫入耗電流	I <sub>DD2</sub>	-	-	4	-	mA
擦除耗電流	I <sub>DD3</sub>	-	-	2	-	mA

注：除非另外說明，以上資料測試條件均為：VDD=5.0V，GND=0V，25°C。

## 20.6 BOR 檢測電壓特性

參數	符號	條件	最小值	典型值	最大值	單位
BOR 設定電壓 1	VBOR1	BOR 使能，VDD=2V~5.5V	1.7	1.8	1.9	V
BOR 設定電壓 2	VBOR2		1.9	2.0	2.1	V
BOR 設定電壓 3	VBOR3		2.3	2.4	2.5	V
BOR 設定電壓 4	VBOR4		2.5	2.6	2.7	V

BOR 設定電壓 5	VBOR5		2.9	3.0	3.1	V
BOR 設定電壓 6	VBOR6		3.5	3.6	3.7	V
BOR 設定電壓 7	VBOR7		3.8	3.9	4.0	V
BOR 設定電壓 8	VBOR8		4.1	4.2	4.3	V

## 20.7 LVD/PLVD 檢測電壓特性

參數	符號	條件	最小值	典型值	最大值	單位
LVD 設定電壓 0	VPLVD	LVD 使能，VDD=2V~5.5V	-	0.8	-	V
LVD 設定電壓 1	VLVD1		1.8	1.9	2.0	V
LVD 設定電壓 2	VLVD2		1.9	2.0	2.1	V
LVD 設定電壓 3	VLVD3		2.3	2.4	2.5	V
LVD 設定電壓 4	VLVD4		2.5	2.6	2.7	V
LVD 設定電壓 5	VLVD5		2.9	3.0	3.1	V
LVD 設定電壓 6	VLVD6		3.5	3.6	3.7	V
LVD 設定電壓 7	VLVD7		3.8	3.9	4.0	V
LVD 設定電壓 8	VLVD8		4.1	4.2	4.3	V

## 20.8 其他電氣特性

1、ESD ( HBM ) : CLASS 3A (  $\geq 8000V$  )

2、Latch\_up : CLASS I ( 200mA )

## 23 開發工具

### 23.1 HC-LINK 模擬工具

HC88L051F4 使用 HC-LINK 模擬器進行程式的下載和模擬,HC-LINK 通過 SWD 介面可以對芯聖所有的增強型 8051 內核單片機(非固化 ISP)實現下載和模擬。關於 HC-LINK 的使用,請參見 HC-LINK 用戶手冊。

HC-LINK 特性：

- 支持 Keil C51 集成編譯環境 ( uVision4.0 及以上版本 )
- 支持所有的芯聖 8051 ( 非固化 ISP ) 晶片
- 可以對 FLASH 進行擦除、程式設計和校驗
- 可以對加密位元以及代碼選項進行程式設計
- 直接從 USB 供電，不需要外接電源

### 23.2 HC-PM51 燒錄工具

HC-PM51 燒錄器是芯聖新一代量產燒錄工具，適用於芯聖 8051 內核系列的 Flash MCU 的燒錄。

關於 HC-PM51 的使用，請參見 HC-PM51 工具用戶手冊。

HC-PM51 特性：

- 採用 USB 方式連接
- 支持單路離線燒錄

### 23.3 ISP 串口燒錄

HC-LINK V4.0 和 HC-PM51 支持離線固化 ISP 程式，當把 ISP 程式固化到 HC88L051F4 之後，用

戶就可以使用 TXD/RXD 兩個引腳進行程式的下載和更新。

用戶可以使用 HC-LINK V4.0 工具，配合上位機軟體 HC-ISP，通過串口對固化 ISP 程式的 FLASH 單片機實現一鍵下載的功能。另外，用戶也可以使用 HC-PM51 的 ISP 模式進行使用者程式的離線下載。

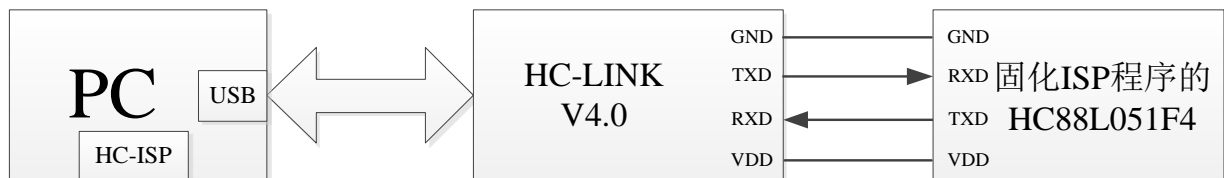


Figure 23-1 ISP 串口燒錄框圖

## 23.4 軟體下載

直接在瀏覽器位址欄中輸入位址：[www.holychip.cn](http://www.holychip.cn) 進行軟體下載。



## 24 封裝尺寸

### 24.1 TSSOP20

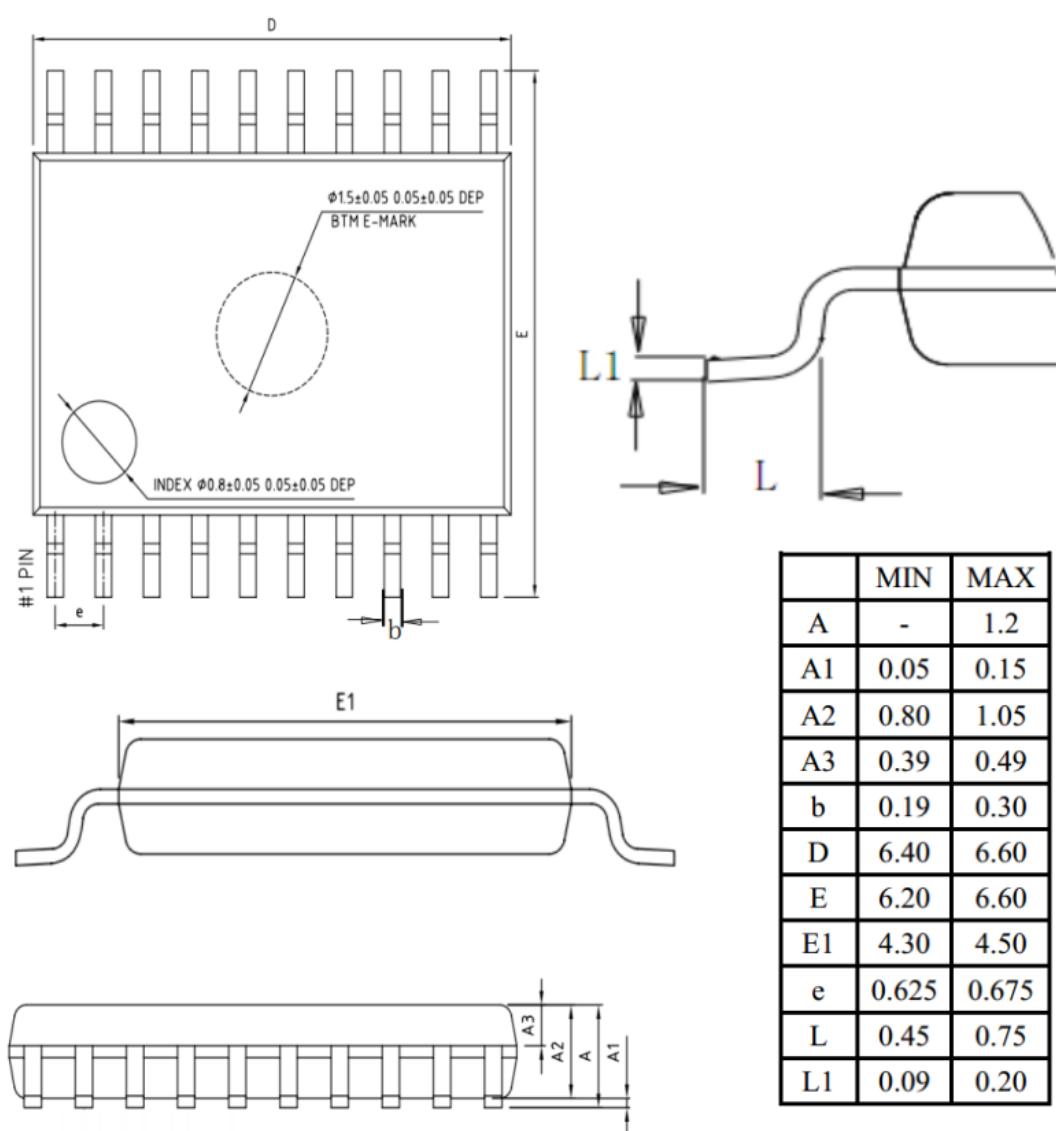
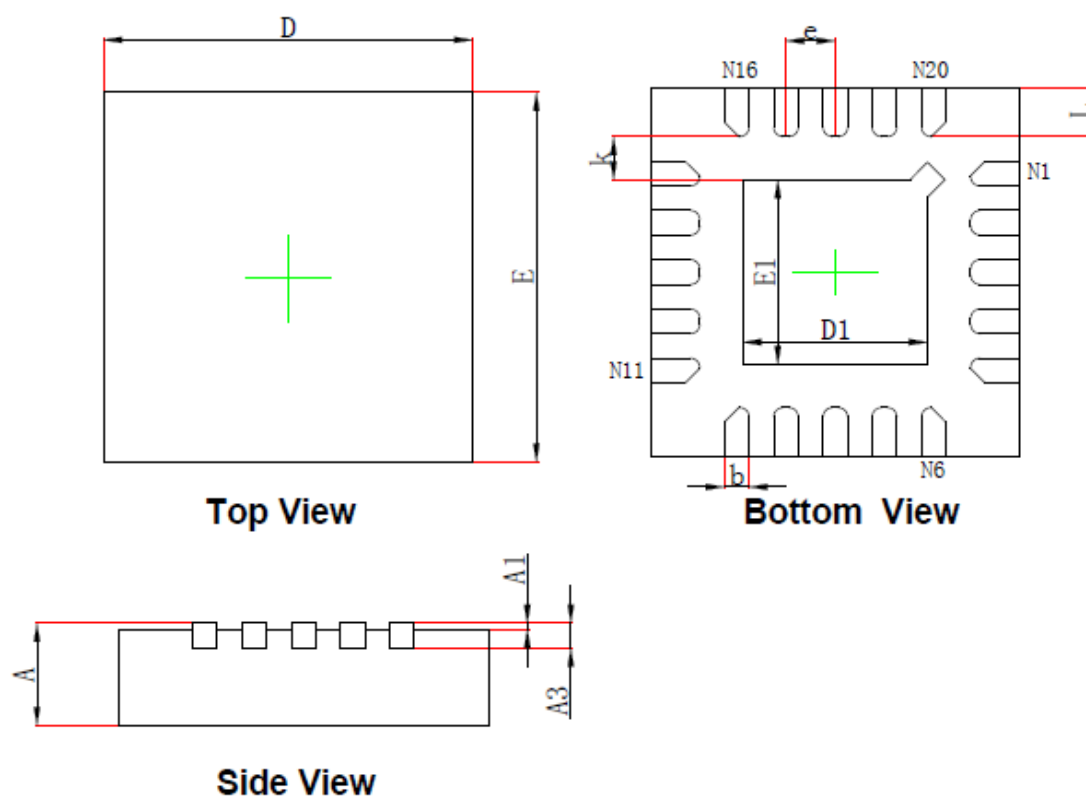


Figure 24-1 TSSOP20 封裝尺寸

## 24.2 QFN20



Symbol	Dimensions In Millimeters		Dimensions In Inches	
	Min.	Max.	Min.	Max.
A	0.700/0.800	0.800/0.900	0.028/0.031	0.031/0.035
A1	0.000	0.050	0.000	0.002
A3	0.203REF.		0.008REF.	
D	2.924	3.076	0.115	0.121
E	2.924	3.076	0.115	0.121
D1	1.400	1.600	0.055	0.063
E1	1.400	1.600	0.055	0.063
k	0.200MIN.		0.008MIN.	
b	0.150	0.250	0.006	0.010
e	0.400TYP.		0.016TYP.	
L	0.324	0.476	0.013	0.019

Figure 24-2 QFN20 封装尺寸

## 25 版本記錄

版本	日期	描述
Ver1.00	2020-06-06	第一版
Ver1.01	2020-12-07	添加 PWM 波形示意圖和比較器框圖

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产品或電路的運用和使用所引起的任何責任，HOLYCHIP 的产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 的产品用於上述領域，即使這些是由 HOLYCHIP 在产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。

