



HY16F18 系列 用 戶 手 冊

高精密混合信號處理控制器
65nV 高解析度 24-位 ΣADC
32-位低功耗微控制器
64K 字節 Flash ROM
8K 字節 SRAM

目錄

01.晶片概述	7
01.1 整體總說明	7
02.功能概述	8
02.1 說明方塊圖	8
02.2 功能IO位址	9
02.3 基礎架構圖	10
03.記憶體結構	11
03.1 記憶體說明	11
03.2 記憶體位址	12
03.3 靜態記憶體 (SRAM)	12
03.4 快閃記憶體 (Flash ROM)	12
03.5 匯流排介面單元	12
03.6 開機唯讀記憶體 (Boot ROM)	13
03.7 嵌入式除錯模組 (EDM)	13
03.8 資訊記憶體區塊	13
04.系統暫存器	14
04.1 整體總說明	14
04.2 暫存器位址	14
04.3 暫存器功能	14
05.電源管理	15
05.1 整體總說明	15
05.2 暫存器位址	17
05.3 暫存器功能	17
06.時脈系統	19
06.1 整體總說明	19
06.2 暫存器位址	22
06.3 暫存器功能	22
07.中斷控制系統	27
07.1 整體總說明	27
07.2 暫存器位址	27
07.3 暫存器功能	28
07.4 範常式式流程	34
07.5 範常式式功能	34
07.6 範常式式說明	34
08.看門狗WDT	35
08.1 整體總說明	35

08.2 暫存器位址	36
08.3 暫存器功能	36
08.4 範常式式流程	37
08.5 範常式式功能	37
08.6 範常式式說明	38
09.定時器Timer A	39
09.1 整體總說明	39
09.2 暫存器位址	41
09.3 暫存器功能	41
09.4 範常式式流程	42
09.5 範常式式功能	42
09.6 範常式式說明	43
10.定時器Timer B	44
10.1 整體總說明	44
10.2 暫存器位址	56
10.3 暫存器功能	56
10.4 範常式式流程	59
10.5 範常式式功能	59
10.6 範常式式說明	60
11.定時器Timer C	61
11.1 整體總說明	61
11.2 暫存器位址	63
11.3 暫存器功能	63
11.4 範常式式流程	64
11.5 範常式式功能	65
11.6 範常式式說明	65
12.GPIO PT1 管理	67
12.1 整體總說明	67
12.2 暫存器位址	68
12.3 暫存器功能	68
12.4 範常式式流程	70
12.5 範常式式功能	70
12.6 範常式式說明	71
13.GPIO PT2 管理	72
13.1 整體總說明	72
13.2 暫存器位址	73
13.3 暫存器功能	73
14.GPIO PT3 管理	76

14.1 整體總說明	76
14.2 暫存器位址	77
14.3 暫存器功能	77
15. GPIO PT4 管理	79
15.1 整體總說明	79
16. GPIO複用功能管理	80
16.1 整體總說明	80
16.2 暫存器位址	81
16.3 暫存器功能	81
17. 模數轉換器ΣΔ 24 位元ADC	84
17.1 整體總說明	84
17.2 暫存器位址	89
17.3 暫存器功能	89
17.4 範常式式流程	93
17.5 範常式式功能	94
17.6 範常式式說明	95
18. 軌對軌運算放大器 (Rail-to-Rail OPAMP)	97
18.1 整體總說明	97
18.2 暫存器位址	99
18.3 暫存器功能	99
18.4 範常式式流程	101
18.5 範常式式功能	101
18.6 範常式式說明	103
18.4 應用電路系統暫功能 01	104
18.5 應用電路系統暫功能 02	104
18.6 應用電路系統暫功能 03	105
19. 8 bit Resistance Ladder	106
19.1 整體總說明	106
19.2 暫存器位址	107
19.3 暫存器功能	108
19.4 範常式式流程	109
19.5 範常式式功能	111
19.6 範常式式說明	111
19.7 應用電路功能 01	112
19.8 應用電路功能 02	112
20. 多功能比較器CMP	113
20.1 整體總說明	113
20.2 暫存器位址	117

20.3 暫存器功能	117
20.4 範常式式流程.....	119
20.5 範常式式功能.....	120
20.6 範常式式說明.....	121
20.7 應用電路功能 01	122
20.8 應用電路功能 02	122
21.串列通信SPI.....	123
21.1 整體總說明	123
21.2 暫存器位址	127
21.3 暫存器功能	128
21.4 範常式式流程.....	132
21.5 範常式式功能.....	132
21.6 範常式式說明.....	133
22.非同步串列通訊EUART.....	135
22.1 整體總說明	135
22.2 暫存器位址	139
22.3 暫存器功能	139
22.4 範常式式流程.....	143
22.5 範常式式功能.....	143
22.6 範常式式說明.....	143
23.I2C通信介面.....	145
23.1 整體總說明	145
23.2 暫存器位址	153
23.3 暫存器功能	153
23.4 範常式式流程.....	159
23.5 範常式式功能.....	159
23.6 範常式式說明.....	160
24.硬體萬年曆HW RTC	162
24.1 整體總說明	162
24.2 暫存器位址	164
24.3 暫存器功能	165
24.4 範常式式流程.....	174
24.5 範常式式功能.....	175
24.6 範常式式說明.....	175
25.省電模式介紹	177
25.1 整體總說明	177
25.2 暫存器位址	177
25.3 暫存器功能	178

26. HYCON Note 32	179
27.修訂歷史	180

關於這份操作手冊：

本檔所述的應用訊息及其他類似內容敘述僅為提供使用者便利，紜康對於內容的使用與因而引起的後果並不負擔相關責任。規格內容隨時可被更新訊息所替代，使用者有責任必需承擔並確保應用符合規範。

注意：

未經紜康授權，不得將紜康產品使用於生命維持系統中作為關鍵器件。

紜康有不需事先通知即可修改產品的權力。

本說明書中的內容，隨著產品的改進，有可能不經過預告而更改。

請客戶及時到本公司網站下載更新 <http://www.hycontek.com>

(2)本規格書中的圖形、應用電路等，因第三方工業所有權引發的問題，本公司不承擔其責任。

(3)本產品在單獨應用的情況下，本公司保證它的性能、典型應用和功能符合說明書中的條件。

當使用在客戶的產品或設備中，以上條件我們不作保證，建議客戶做充分的評估和測試。

(4)請注意輸入電壓、輸出電壓、負載電流的使用條件，使 IC 內部的功耗，不超過封裝的容許值。對於客戶在超出說明書中規定額定值使用產品，即使是瞬間的使用，由此所造成的損失，

本公司不承擔任何責任。

(5)本產品，雖內建防靜電保護電路，但請不要施加超過保護電路性能的過大靜電。

(6)本規格書中的產品，未經書面許可，不可使用在要求高可靠性的電路中。例如健康醫療器械、防災器械、車輛器械、車載器械及航空器械等對人體產生影響的器械或裝置，不得作為其部件使用。

(7)本公司一直致力於提高產品的品質和可靠度，但所有的半導體產品都有一定的失效概率，這些失效概率可能會導致一些人身事故、火災事故等。當設計產品時，請充分留意冗餘設計並採用安全指標，這樣可以避免事故的發生。

(8)本規格書中內容，未經本公司許可，嚴禁用於其他目的之轉載或複製。

01. 晶片概述

01.1 整體總說明

HY16F18X 是一款低功耗、高效能混合信號處理微控制器，適用於做非常精準的測量和控制。它能工作在 2.2V 到 3.6V 的寬電壓範圍，內部 HSRC 高頻最高可運行到 10MHZ，內建 64k/32k/16k-字節的 Flash，以及 8k/4k/2k-字節的 SRAM。HY16F18X 集成了高精度 24 位模數轉換器 ADC，低噪聲軌對軌運算放大器 OPA，8 位元數位電阻器(8bit Resistance Ladder)，硬體實現的實時時鐘 RTC，多功能比較器 CMP，提供高性能外圍介面，如 EAURT、SPI、I2C、GPIOS，及穩健的電源管理系統等功能。支援欠壓檢測，多個外圍介面喚醒功能。HY16F18X 屬於低電壓、低功耗、低待機電流，高集成度外設，高效率操作，快速喚醒及低成本的基於 C 開發平臺的 32-位微控制器。它可以讓設計者完成一個低電源低成本的混合信號處理系統。

這款控制器中的類比前端電路包含了一個 8 位元的數位類比轉換器、一個軌對軌運算放大器、和一個軌對軌輸入比較器。其中，這 8 位元的數位類比轉換器是保證單調性。它是一個梯形電阻，每一個最低有效位元(LSB)為 600 歐姆，該電阻具有低溫係數。軌對軌運算放大器有一個輸入網路，用於差分類比電路配置，例如積分器、電流電壓轉換器、可編程增益放大器、和連續漸進式類比數位轉換器。軌對軌輸入比較器則以非常低的耗電量對類比訊號做連續性的監控，它可作為電源電壓監控器、外部喚醒觸發源、或電容式觸控鍵驅動器。

這個超低噪音 24 位元類比數位轉換器是嵌入式的。它的最大輸出率為 10.24KSPS 且具 21 位元的有效位元數(ENOB)，最低噪音水準為 65nV RMS。這個具有可編程增益的低噪音放大器是和類比數位轉換器一起使用，最大增益為 128。有一個 4 位元數位類比轉換器到類比數位轉換器的輸入，用來擴張量測範圍。還有一個內建的基本類比輸入緩衝器，允許高輸出阻抗基準源。

電源管理提供可選擇的類比電路調節電壓。數位電源也是由內部的線性穩壓電源所提供之。電荷泵，則是用來阻擋來自系統的電力幹擾。它也可將低工作電壓降到 2.0V。

它有一個 64K 位元組的嵌入式快閃記憶體，可用來執行程式及儲存資料。資料可在執行期間被儲存在這個快閃記憶體。它也有一個 8K 位元組的靜態隨機存取記憶體供系統使用。

這個高效能 32 位元的 MCU 核心可以每一時脈週期(Clock Cycle)執行一個指令，最高可達 10MIPS。它提供了一些容易使用的程式撰寫工具，使用者可針對此 MCU 撰寫 C 語言或組合語言的程式。具有電路仿真功能的 IC 提供了容易除錯的環境。它可以用 2.2V 到 3.6V 的電壓，在-40 to 85°C 的溫度環境下工作。

02. 功能概述

02.1 說明方塊圖

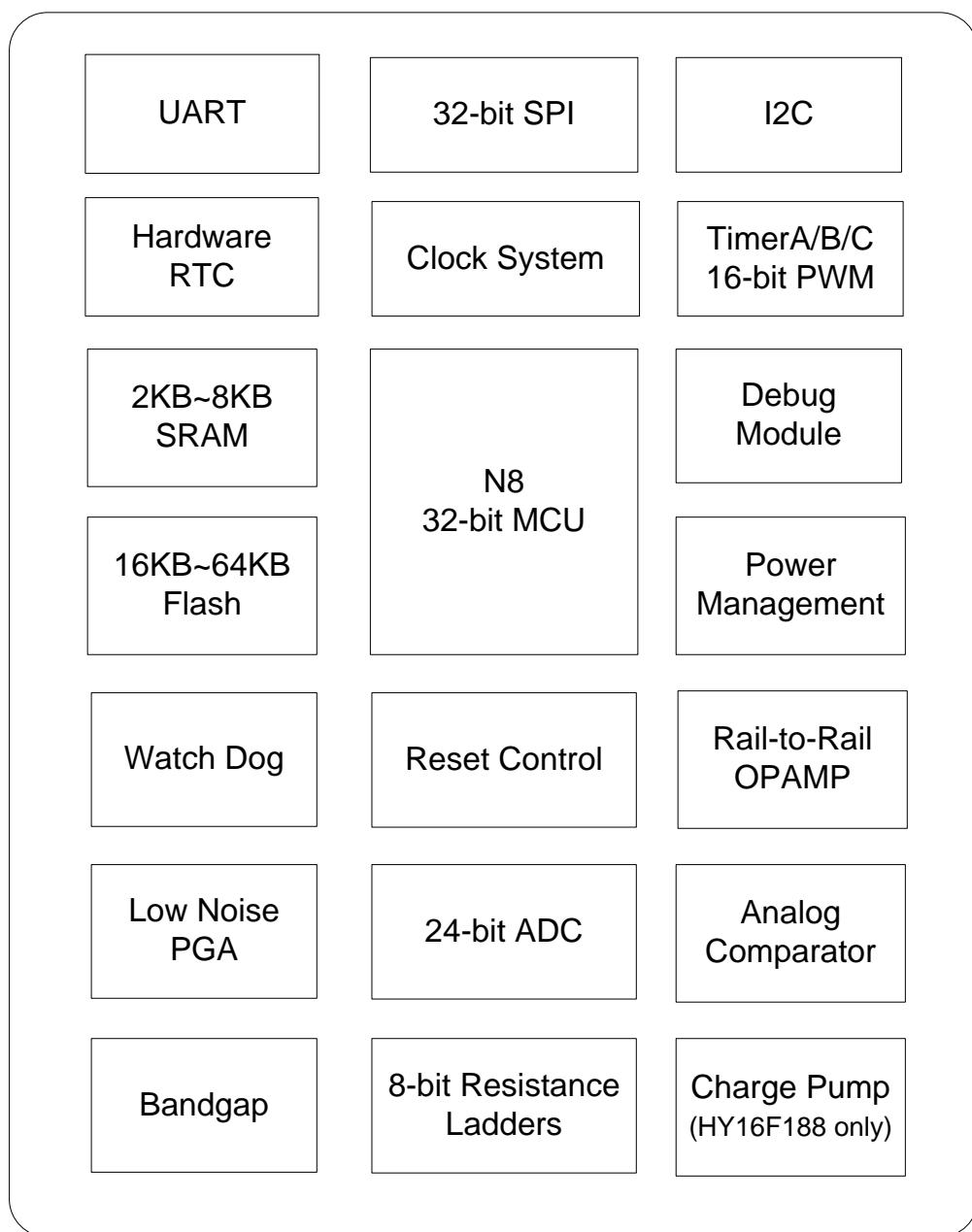


圖 2-1 晶片功能架構圖

02.2 功能 IO 位址

GPIO Port	OSC	Interrupt	Timer C Capture	SPI	IIC	UART	CMP	Analog	Timer B PWM
Priority	0	0	0	1	2	3	4	5	6
PT1.0		INT1.0	TCI1_1	CS_1	SCL_1	TX_1	CH1		PWM0_1
PT1.1		INT1.1	TCI2_1	CK_1	SDA_1	RX_1	CH2		PWM1_1
PT1.2		INT1.2	TCI1_2	MISO_1	SCL_2	TX_2	CH3		PWM0_2
PT1.3		INT1.3	TCI2_2	MOSI_1	SDA_2	RX_2	CL1		PWM1_2
PT1.4		INT1.4	TCI1_3	CS_2	SCL_3	TX_3	CL2		PWM0_3
PT1.5		INT1.5	TCI2_3	CK_2	SDA_3	RX_3	CL3		PWM1_3
PT1.6		INT1.6	TCI1_4	MISO_2	SCL_4	TX_4	CL4		PWM0_4
PT1.7		INT1.7	TCI2_4	MOSI_2	SDA_4	RX_4	CMPO1		PWM1_4
PT2.0		INT2.0	TCI1_5	CS_3	SCL_5	TX_5			PWM0_5
PT2.1		INT2.1	TCI2_5	CK_3	SDA_5	RX_5			PWM1_5
PT2.2		INT2.2	TCI1_6	MISO_3	SCL_6	TX_6			PWM0_6
PT2.3		INT2.3	TCI2_6	MOSI_3	SDA_6	RX_6			PWM1_6
PT2.4	LSXT1	INT2.4	TCI1_7	CS_4	SCL_7	TX_7			PWM0_7
PT2.5	LSXT2	INT2.5	TCI2_7	CK_4	SDA_7	RX_7			PWM1_7
PT2.6	HSXT1	INT2.6	TCI1_8	MISO_4	SCL_8	TX_8			PWM0_8
PT2.7	HSXT2	INT2.7	TCI2_8	MOSI_4	SDA_8	RX_8			PWM1_8
PT3.0							OPO1		
PT3.1							OPO2	DAO	
PT3.2								AIO4	
PT3.3								AIO5	
PT3.4								AIO6	
PT3.5								AIO7	
PT3.6								REFO	
PT3.7								OPO	
AIO0								AIO0	
AIO1								AIO1	
AIO2								AIO2	
AIO3								AIO3	

表 2-1 晶片 IO 引腳功能分佈表

02.3 基礎架構圖

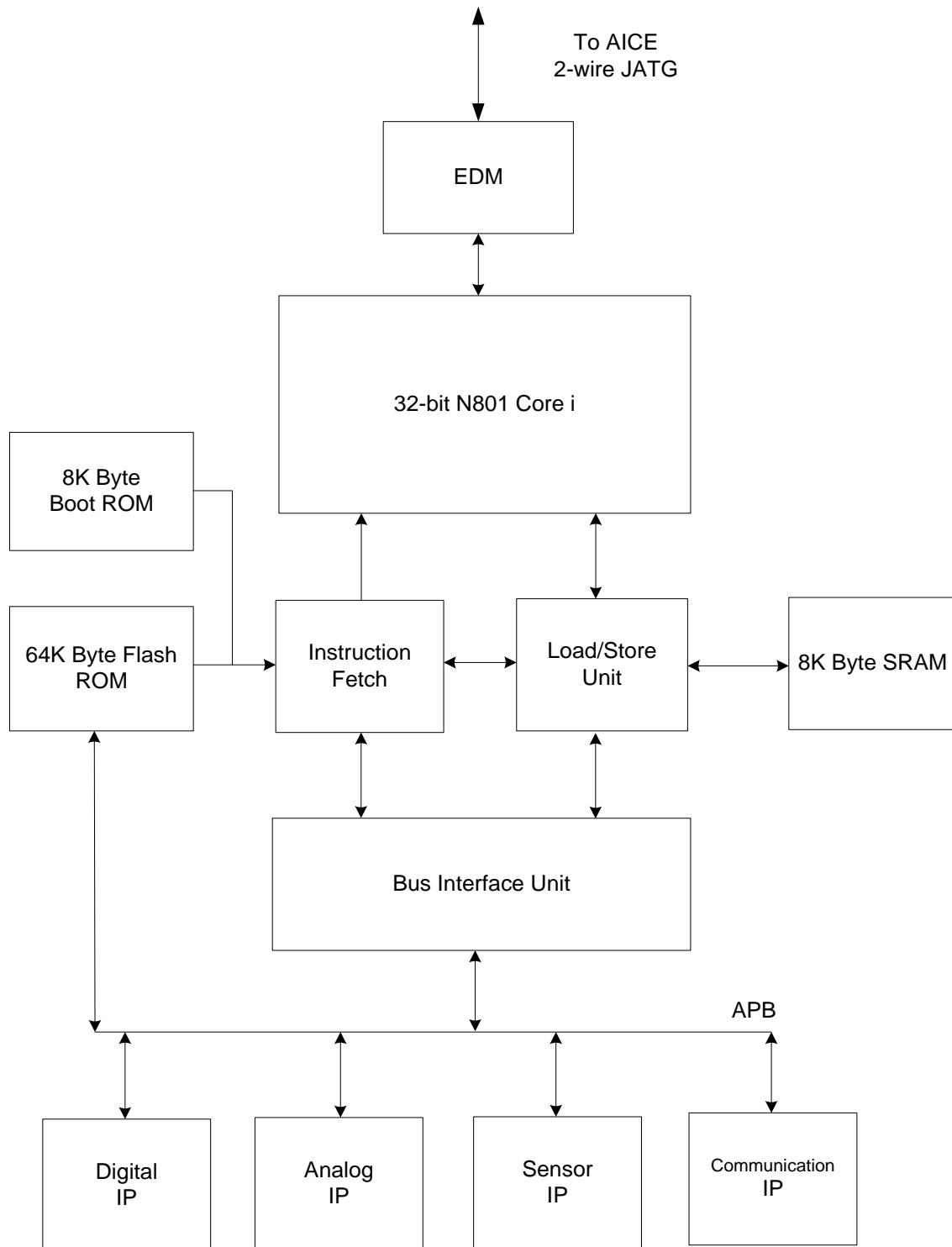


圖 2-2 晶片調試架構圖

03. 記憶體結構

03.1 記憶體說明

HY16F18X 所使用的 CPU 內核是 Andes N801，一個 32 位元的 CPU 核心。提供 64k 尋址空間，晶片上控制器的存儲器地址分配如下：

- (1) 0X0_0000–0X1_FFFF SRAM (8K Byte)
- (2) 0X4_0000–0X4_FFFF SOC Register (64K Byte)
- (3) 0X8_0000–0X8_1FFF Boot ROM (8K Byte)
- (4) 0X9_0000–0X9_FFFF Main Program Flash (64K Byte)
- (5) 0XD_0000–0XD_03FF Information Area Flash (1K Byte)

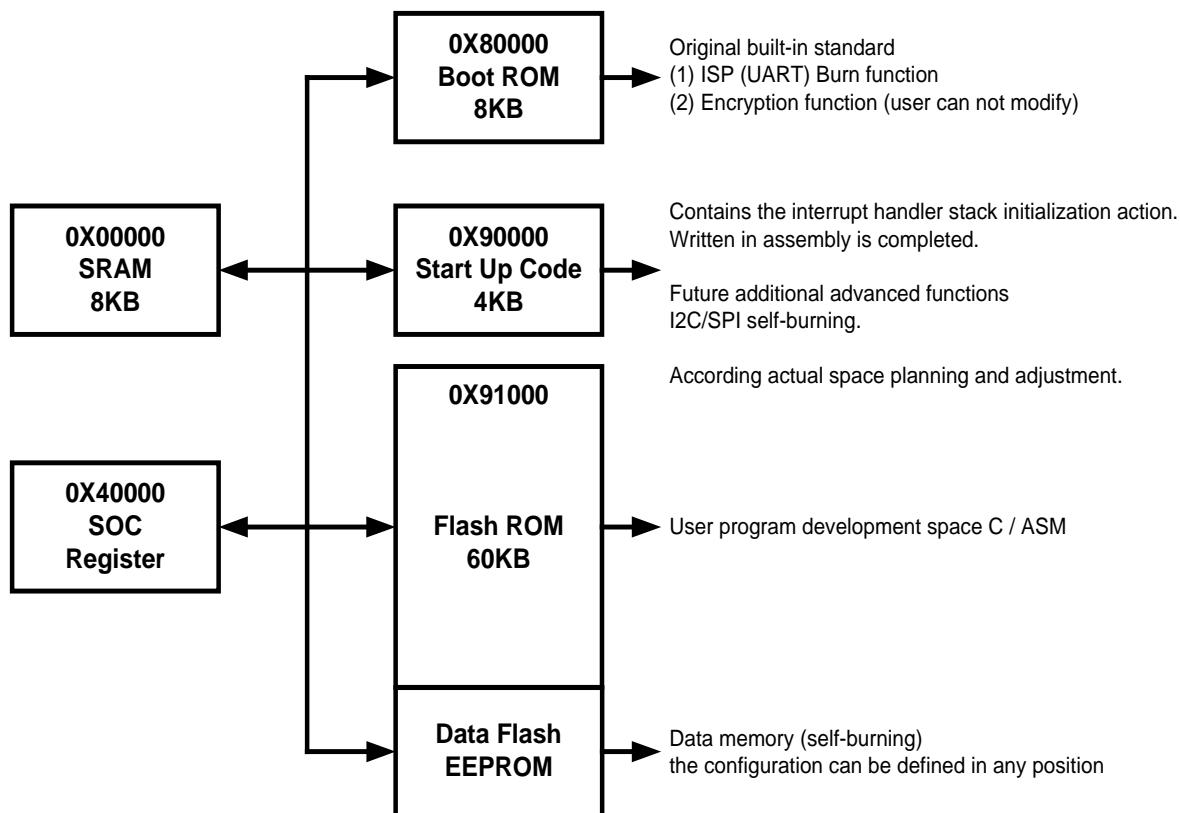


圖 3-1 Flash 分佈圖

03.2 記憶體位址

以下是晶片控制器的存儲器分配見下表。

表 3-1

功能模塊	描述	Base Address
INT	Interrupt Flag	0X4_0000
SYS	System Register	0X4_0100
CLK	Clock System Register	0X4_0300
PMU	Power Management Unit	0X4_0400
MC	Memory Controller	0X4_0600
PIO	Port I/O Control	0X4_0800
TMR	Timer Register	0X4_0C00
UART	UART Mode, Communication Interface	0X4_0E00
SPI	SPI Mode, Communication Interface	0X4_0F00
I2C	I2C Mode, Communication Interface	0X4_1000
ADC	Analog-to-Digital Module	0X4_1100
DAC	Digital-to-Analog Module	0X4_1700
CMP	Comparator Network Module	0X4_1800
OPN	Operational Amplifier	0X4_1900
RTC	Real Time Clock	0X4_1A00

每個暫存器都含有 mask 位，mask 位是用於使能對應控制位的寫入，只有在與控制位對應的 mask 位為 1 時，對應的控制位才能寫入值，否則出現寫入失敗，無法修改暫存器的值。如圖 3-2 所示：

暫存器總共有 32 位，其中有 16 位 mask 位。Mask 位又分為兩組 8 位，每 8 位控制著相應的 8 位控制位。根據寄存器的內容分佈：BIT[31:24]控制著 BIT[23:16]，而 BIT[15:8]控制著 BIT[7:0]；只有在 mask 為被置<1>的情況下，對應的位才能被寫入值。

如要對 bit[5:0]寫入 101010b，即是只要對低 16 位操作，則寄存器操作方式為：0x3F2A；其中 0X3F 是 BIT[15:8]的寫入值，使能 BIT[5:0]對應的 mask 位，0X2A 就是對 BIT[5:0]寫入的值。

INT Base Address + 0X10 (0X40010)									
Symbol	INTPT1 (Interrupt Control Register 4)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名稱	MASK	PT17IE	PT16IE	PT15IE	PT14IE	PT13IE	PT12IE	PT11IE	PT10IE
RW	R0W-0								RW-0
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名稱	MASK	PT17IF	PT16IF	PT15IF	PT14IF	PT13IF	PT12IF	PT11IF	PT10IF
RW	R0W-0								RW0-0

圖 3-2 暫存器的基本結構

03.3 靜態記憶體 (SRAM)

HY16F188 有 8K 位元組的靜態隨機存取記憶體。它的起始位址是從 0X0000 到 0X2000。MCU 可選擇一個位元組、兩個位元組(半字)、或四個位元組(全字)的存取。這個 SRAM 可在一個 Clock Cycle 存取一個資料。

03.4 快閃記憶體 (Flash ROM)

HY16F188 有 64K 位元組的嵌入式快閃記憶體。它的起始位置是從 0X9_0000 到 0XA_0000。使用者程式碼儲存在這快閃記憶體。要編寫這快閃記憶體的程式碼，使用者需用 EDM 指令來讀寫快閃控制單元。這快閃記憶體其中一部份被保留給使用者儲存資料。

03.5 匯流排介面單元

在 HY16F188 的結構中，暫存器的讀寫是由一個 32 位元的高階週邊匯流排(APB)所控制。

它可在一個 Clock Cycle 寫入一個 32 位元的資料。但為了防止資料寫入期間去覆蓋掉已寫入的資料值，HY16F188 使用了遮罩的功能。

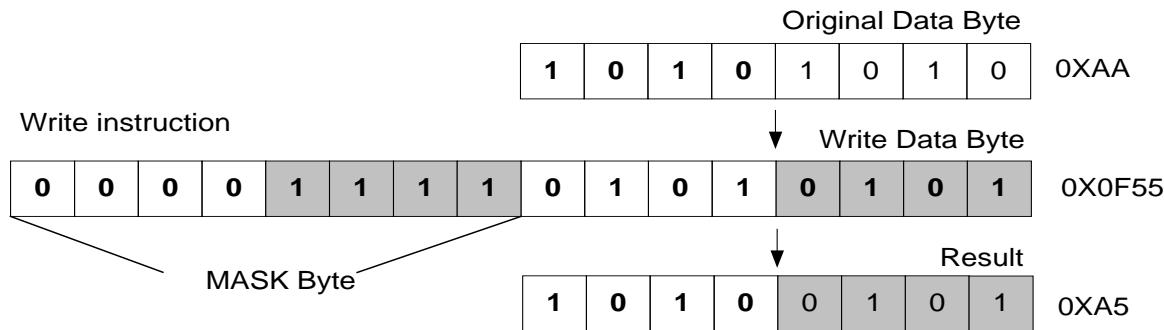


圖 3-3 數據流結構

03.6 開機唯讀記憶體 (Boot ROM)

HY16F188 有一個 8K 位元組的遮罩唯讀記憶體。它的起始位址是從 0X8_0000 到 0X8_2000。這記憶體是給開機碼、快閃控制碼、和安全保護碼所使用。若重設發生，程式計數器會從 0X8_0000 開始。開機唯讀記憶體中的軟體涵蓋硬體資訊、系統程式協定、和安全協定。

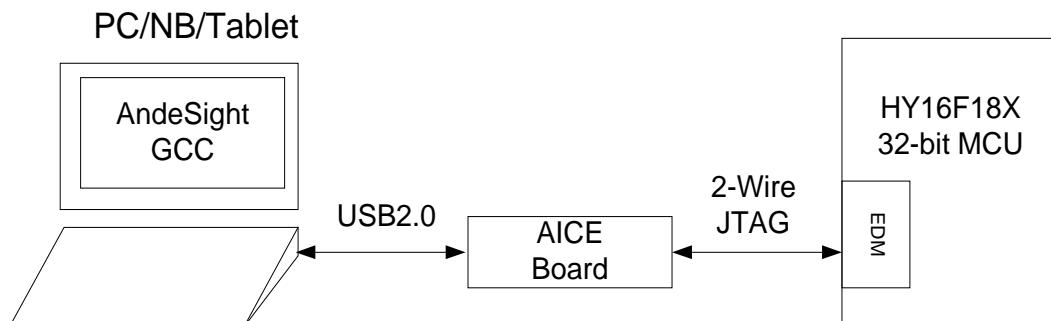


圖 3-4 晶片開發連接圖

03.7 嵌入式除錯模組 (EDM)

這個 EDM 是由 Andes 所提供。它可以輸入一個指令給 MCU 並在除錯模式中控制 HY16F188。它是 MCU 和 AICE 之間的橋樑，使用兩線協定及 JTAG 介面。它可以存取 APB 上的 IP 暫存器、一般用途的 GPR 暫存器、SRAM DLM、以及 ROM Flash ILM。它有控制 APB 的最高權限。用 EDM 送出指令就可以進入除錯或測試模式。針對除錯模式，這 EDM 有 2 個規則性的斷點暫存器和觀察點暫存器，以及 6 個簡單斷點暫存器。

03.8 資訊記憶體區塊

在快閃記憶體中有 1K 位元組額外的記憶體空間，用來儲存資訊資料。在正常模式期間，晶片 MCU 無法從這記憶體區塊中存取資料。只能在發展模式中存取資訊。在發展模式中，有 3 種方式來存取或寫入這區塊的資料：Boot ROM MCU、HYCON I2C。這個 EDM 匯流排在 EDM 控制 APB/MCU 時，受到保護；它通常是用來除錯用的。這個 Boot ROM MCU 在 MCU 執行 Boot ROM 中的指令時，受到保護；它通常是在 MCU 重設後或在程式發展模式中被使用。HYCON I2C 在 HYCON I2C 控制 APB/MCU 時，受到保護；它通常是用在測試模式及大量程式撰寫模式。

04. 系統暫存器

04.1 整體總說明

管理系統的工作模式及晶片的復位狀態，如看門狗、外部復位、欠電壓復位等。

04.2 暫存器位址

SYS Register Address	31	24	23	16	15	8	7	0
SYS Base Address + 0X04(0X40104)	-	-		MASK0	REG0			

-保留

04.3 暫存器功能

操作該寄存器 SYS0[4]可以對系統工作模式設置為休眠模式(SLEEP)/待機模式(IDLE)/等待模式(WAIT)。查詢該寄存器 SYS0[3:0]可知晶片處於什麼樣的工作狀態下。晶片工作模式的設置在後面的第 25 章節詳細解說。

04.3.1 系統暫存器 SYS0

SYS Base Address + 0X04 (0X40104)								
Symbol	SYS0 (SYS Control Register 0)							
Bit	[31:16]							
名稱	RSV							
RW	R-0							
Bit	[15:8]	[7:6]	[5]	[4]	[3]	[2]	[1]	[0]
名稱	MASK	-	F _{CRST}	IDLE	F _{SLP/IDLE}	F _{WDT}	-	F _{BOR}
RW	R0W-0	-			RW0-0			RW0-1

位元	名稱	描述
Bit[05]	F _{CRST}	CPU Core 復位旗標
		0 正常 1 ICP Core 復位已觸發
Bit[04]	IDLE	待機模式控制位元
		0 休眠模式 (Sleep Mode) 1 待機模式 (IDLE Mode)
Bit[03]	F _{SLP/IDLE}	休眠(Sleep)/待機(Idle Flag)(低電壓復位或復位電路復位可清零該位)
		0 正常 1 晶片進入休眠模式(Sleep Mode)或待機模式(Idle Mode)
Bit[02]	F _{WDT}	看門狗(WDT)旗標((低電壓復位或外部復位可清零該位)
		0 正常 1 看門狗產生復位或者中斷
Bit[00]	F _{BOR}	低電壓復位(BOR)旗標 (晶片電壓高於 1.8V 自動清理)
		0 正常 1 低電壓復位已發生

05. 電源管理

05.1 整體總說明

HY16F18X 的電源管理模塊包括一個電荷泵調節器、一個粗帶隙基準、一細帶隙基準、一個 VDDA LDO、一個 VDD18 LDO、以及基準輸出緩衝器。HY16F18X 只需要一個電壓源來運轉，這運轉電壓源介於 2.2 到 3.6V 之間。但是，如果運轉電壓源是介於 2 到 2.4V 之間，HY16F18X 就需要啟動電荷泵調節器來提供電源給一些類比 IP 和 Flash。HY16F18X 的電源系統可分成三個部份：輸入/輸出電路、類比電路、和數位電路。輸入/輸出電路的電源是由 VDD3V 所驅動。類比電路電源是由內部的 VDDA LDO 所驅動。最後，數位電路的電源是由 VDD18 LDO 所驅動。

當 MCU 在待機模式，晶片已最低的功耗來維持暫存器和 SRAM 的資料存儲操作。在待機模式中，粗帶隙基準、BOR、和 VDD18 LDO 是開啟的。這些功能模塊在室溫下總共只消耗 1.5uA 的電流。若在自動喚醒模式，低速振盪器需要被開啟。這時內部低速振盪器在室溫下消耗額外的 0.5uA 電流。

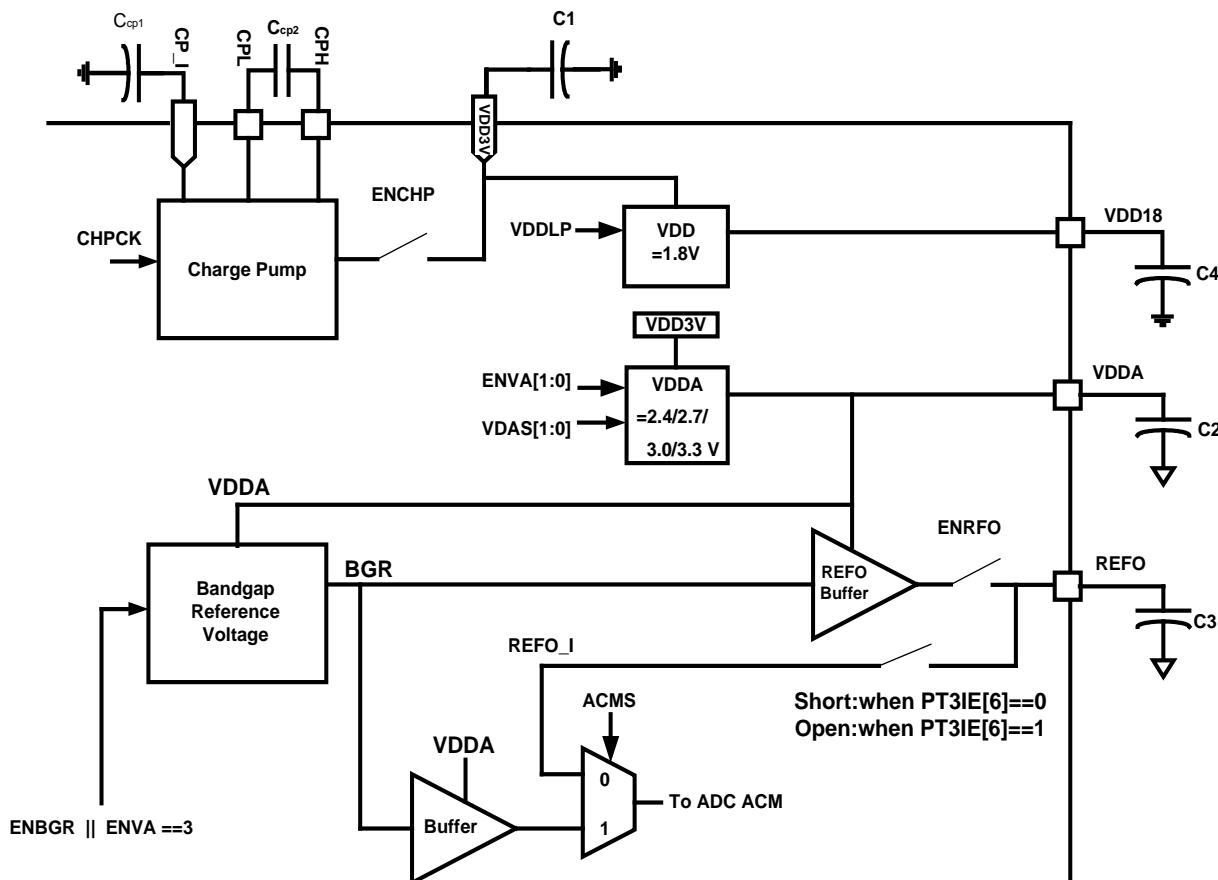


圖 5-1 功能方框圖

晶片工作電壓及 VDD18

晶片工作電壓是通過管腳 VDD3V(PIN43)輸入，電壓範圍是 2.2V~3.6V，且在管腳端需要接入一個 1~10uF 的對地電容，可使這 VDD3V 電壓更穩定。若應用中使用晶片工作電壓來驅動大電流負載，需要注意晶片工作電壓會被拉低導致晶片復位元，此時需要啟動電荷泵升壓電路，輸出穩定 3V 電壓供應給 VDD3V，保證晶片的穩定工作。

VDD18 LDO 通過管腳 VDD18(PIN41)輸出穩壓電壓 1.8V，且管腳端需要接入一個 0.1uF~10uF 的旁路電容器。VDD18 LDO 有一個低功耗電壓模式，要達到最低耗電，寄存器 PMU[0] (VDDLP) 要被設定為 1。這個控制位元在進入待命模式前通常是被設定為 1，且在 MCU 被喚醒後被清除為 0。

VDDA 電壓

HY16F18X 帶有一個轉為類比電路使用的穩壓電路 LDO : VDDA，使用 ADC 時必須開啟 VDDA 電壓。它可設置不同的操作模式和不同的輸出電壓。它有四個不同的操作模式，第一個模式是 VDD3V 短路，此時 VDDA 等於 VDD3V；第二個模式是接地放電(Weak pull down)，此時 VDDA 輸出為 0；第三個模式是高阻態(High Z)，可從外部灌入電壓給 VDDA，但是外部灌入的電壓應該不可超過 VDD3V；第四個模式是可調節穩壓模式 LDO，此模式 VDDA 輸出四個不同的電壓：2.4V、2.7V、3.0V、和 3.3V。要有更好的效能，壓差應該要大於 0.2V，且可向上驅動達到 20mA。它也需要在管腳 VDDA(PIN18)接入一個 0.1uF~10uF 的旁路電容器。

低電壓檢測電路(BOR)

BOR 電路是用來監控電源系統和 MCU 的穩定度。當 VDD3V 或 VDD18 的電壓低於 BOR 偵測電壓，非遮罩重設就會被觸發，系統因而被重設，晶片產生復位，直到晶片工作電壓達到 1.8V 以上才會恢復正常工作。

電荷泵 (Charge Pump)

電荷泵調節器提供穩定的 3.0V 電壓給晶片工作，可用來隔離系統電力和 IC 電力。有些應用需要用到大電流外圍電路，如驅動一個直流電馬達，這時啟動電荷泵保證晶片工作電壓穩定為 3.0V，可降低電機電感的浪湧電流所帶來的幹擾。

要啟動這個電荷泵，程式上需要將暫存器 PMU[2]設定為 1，並且開啟 ADC 時脈源；外圍電路需要在管腳 PIN46 與 PIN45 之間串聯一個外部電容(Ccp2)，及在管腳 PIN47 外接一個對地電容器(Ccp1)，電荷泵穩壓輸出端 CP_O 必須與晶片工作電壓管腳 P43(VDD3V)短接。當電荷泵運轉時，電源來自於 PIN47(CP_I)的輸入。如果沒有開啟電荷泵功能，電荷泵就可以關閉且電源是從內部連接到 VDD3V，開啟電荷泵功能，就會產生穩定 3V 電壓供應給 VDD3V 輸入。若不用到電荷泵升壓電路，則外部電容器 Ccp2 和 Ccp1 就不需要安裝。晶片的工作電壓可由 VDD3V 供應。

為了避免電源故障問題，Ccp2 和 Ccp1 的比例應為 0~1000nF。Ccp2 的最小邊應該大於 10nF。如果 Ccp2 是 10nF，Ccp1 的電容就應該要大於 100nF。較大電容的 Ccp1 可讓系統更穩定。CP_O 輸出與 VDD3V 連接，需靠外部 PCB 短路。

帶隙參考電壓 (BandGap) 和共模電壓(REFO)

當 VDDA 高於 2.4V，類比電路就可運轉。但是，類比電路要求提供電流偏差和參考電壓。因此，在啟動類比電路之前啟動帶隙參考電壓，將寄存器 PMU[4](ENBGR)設定為 1 便可啟動帶隙參考電壓。然後，V12 會變成大約 1.2V。只有開啟帶隙參考電壓後，共模電壓(REFO)才有效輸出 1.2V。

要啟動 ADC，需要為 ADC 一個共模電壓(REFO)。這個共模電壓可選擇外部或內部的電源。若要選擇內部電源，就要將 ACMS 設定為 1。但是，有些使用者會需要一個基準電壓來驅動外部電路。然後，就可以將 ENRFO 設定為 1，將共模電壓輸出至管腳，且 REFO 是具有緩衝器的帶隙基準電壓。它通常會有一個 100 PPM /°C 的溫度係數。這個輸出電壓大約是 1.2V 且具有 1mA 的推拉驅動能力。它可以驅動一個 22~1000nF 的大型電容器承載。如果使用外部的 REFO，此時 ADC 使用的共模電壓可用 ACMS 電壓來節省電力，使用者可將 ACMS 設定為 0。

下表展示各部份模塊使用的電壓源。

表 5-1 晶片電源分佈

Block 名稱	電壓源	備註	Block 名稱	電壓源	備註
32-bit CPU Core N801	VDD18	-	Timer A/B/C PWM	VDD18	-
08KB SRAM	VDD18	-	GPIO Port	VDD3V	-
64KB Flash ROM	VDD3V/ VDD18	-	24-bit SD ADC	VDDA	-
Clock System	VDD18	-	08-bit DAC	VDDA	-
Watch Dog Timer	VDD18	-	Rail-to-Rail OPAMP	VDDA	-
Hardware RTC	VDD18	-	Analog Comparator	VDD3V	-
Charge Pump	VIN	-			
BOR	VDD3V/ VDD18	-			
Band Gap/Reference	VDDA	1.2V			
Hardware EUART	VDD18/VDD3V	-			
32-bit Hardware SPI	VDD18/VDD3V	-			
Hardware I2C	VDD18/VDD3V	-			

05.2 暫存器位址

Power Register Address	31	24	23	16	15	8	7	0
PMU Base Address + 0X00 (0X40400)	MASK1	REG1		MASK0	REG0			

05.3 暫存器功能

05.3.1 電源管理暫存器 PMU

Power Base Address + 0X00 (0X40400)								
Symbol	PMU (PMU Control Register)							
Bit	[31:24]	[23:20]	[19:18]			[17:16]		
名稱	MASK	-	VDAS			ENVA		
RW	R0W-0	-				RW-0		
Bit	[15:08]	[7:6]	[5]	[4]	[3]	[2]	[1] [0]	
名稱	MASK	-	-	ENBGR	ACMS	ENCHP	ENRFO VDDLP	
RW	R0W-0	-	RW-0					

位元	名稱	描述
Bit[19~18]	VDAS	數字電壓 VDDA 輸出值選擇
		00 VDDA =2.4V
		01 VDDA =2.7V
		10 VDDA =3.0V
		11 VDDA =3.3V
Bit[17~16]	ENVA	VDDA LDO 電壓源設置，控制 VDDA 輸出電壓值範圍
		00 高阻抗(High Z)，VDDA 只輸出 0V
		01 短接至 VDD3V，VDDA 輸出值只等於 VDD3V
		10 弱下拉(Weak pull down)，VDDA 只輸出 0V
		11 VDDA LDO，VDDA 可調輸出： 2.4/2.7/3.0/3.3V

位元	名稱	描述	
Bit[04]	ENBGR	帶隙電壓開啟控制	
		0	關閉
		1	開啟(通過 LDO 開啟)
Bit[03]	ACMS	ADC 模擬地輸入源選擇	
		0	外部
		1	開啟緩衝器並使用內部地(需要搭配 ADC 使用)
Bit[02]	ENCHP	電荷泵(Charge Pump)開啟控制	
		0	關閉
		1	開啟
Bit[01]	ENRFO	共模電壓(REFO)開啟控制	
		0	開啟
		1	關閉
Bit[00]	VDDLP	VDD18 LDO 低功耗控制	
		0	正常(從休眠喚醒後該位需被置 0)
		1	低功耗

06.時脈系統

06.1 整體總說明

時脈控制系統為整個晶片提供時脈，包括系統時脈（CPU 時脈、APB 時脈）及所有外圍外設工作時脈（定時器、通訊介面、RTC、類比電路等外圍設備）。各個功能模塊的時脈都帶有時鐘開關控制器，時脈源選擇及除頻器。在休眠模式下，控制器始終關閉外部晶震、內部晶震及系統時脈使系統功耗降到最低。

工作時脈源具有外部晶震、內部 HAO 與 LPO 震盪器，透過除頻器可彈性分配與管理 CPU 及外圍設備的頻率源調整晶片功耗，達到節約功耗目的。

06.1.1 外部震盪器

外部震盪器有兩組，分別為高速晶震(HSXT)和低速晶震(LSXT)。晶片帶有高速和低速外部晶震兩組獨立輸入管腳，所以用戶可以同時接上兩組外部晶震。外部晶震需要並聯一個電阻，否則即使焊接了晶震，也無法正常起震，同時也可以焊接兩個 15~20pF 的對地電容，在無特別情況下，可以將電容缺省。

其相關的電阻 R1、電容 C2/C1 參數隨外接不同頻率、不同品牌的晶震及電路板佈局方式而有所差異。下表列出 R1/C1/C2 參數與頻率源配置，供使用參考。

型式	符號	外部晶震參數				指令執行狀態	
		頻率	R1/Ω	C1	C2	Sleep mode	Idle mode
低速	XTA	32768HZ	10M	20pF	20pF	停止	可設置
高速	XTB	4~16MHZ	1M	20pF	20pF	停止	可設置

表 6-1 晶震配置

使用外部晶震需要注意事項：

- ◆ 外接晶震 4MHZ/8MHZ 的穩定時間約為 25ms，外接 32768HZ 晶震的穩定時間約為 1.3s。
- ◆ 執行休眠指令後，外部晶震皆會停止震盪
- ◆ 外接晶震時，需要注意引腳的輸入/輸出配置，使用時需將引腳設置如輸入模式且不得配置內部上拉電阻，以免操作時發生異常。且外部電阻絕對不能缺省。

06.1.2 內部晶震 HAO 與 LPO

HAO 為晶片內部高速 RC 震盪器輸出典型頻率值為 2MHZ/4MHZ/10MHZ，且具有快速起震、抗幹擾能力強及功耗低的特性。HAO 輸出頻率值具有可校正性，用戶可通過軟件的方式對 HAO 的輸出頻率進行校正。

- ◆ 設置寄存器 0X40304[7:0]的值，可調整 HAO 輸出頻率值。
- ◆ 晶片在默認啟動的震盪器為內部 HAO 2MHZ，用戶進行設置與配置其他的 HAO 輸出頻率值。
- ◆ HAO 配置為 4MHZ 起震的穩定時間約為 0.5ms；
- ◆ 執行休眠 (SLEEP) 指令後，HAO 震盪器皆會停止震盪進入休眠狀態；
- ◆ 執行待機(IDLE)指令，並不會停止 HAO 震盪器，但 CPU 會進入待機狀態。

LPO 為晶片內部低速 RC 震盪器，輸出頻率為 33kHz，具有低功耗電流特性，且在晶片上電或喚醒後即起震，同時也是無法關閉，在整個晶片工作過程 LPO 都是運行的。

- ◆ LPO 起震穩定時間約為 510us 且是看門狗(WDT)的唯一工作時脈源。
- ◆ 執行休眠指令(SLEEP)後，LPO 震盪器皆會停止震盪；
- ◆ 執行待機指令(IDLE)後，LPO 不會停止震盪，但 CPU 會進入待機狀態。

典型的 HAO 與 LPO 輸出頻率值如下表 6-2 所示。

符號	頻率	頻率源配置	指令執行狀態
----	----	-------	--------

		ENHAO[1]	HAOM[1:0]	CKHS[1]	Sleep	Idle
HAO	2MHZ	1	00B	0	停止	震盪
	4MHZ	1	01B	0	停止	震盪
	10MHZ	1	10B	0	停止	震盪
LPO	33KHZ	晶片上電後起震		CKLS=0	停止	震盪

表 6-2 內部晶震配置

06.1.3 CPU 與周邊外設工作頻率源配置

內部外部晶震都可以為 CPU 提供頻率源，頻率源經過除頻器後供給 CPU。晶片可透過頻率選擇器 MCUCKS[1] 選擇 CPU 的頻率源為 HS_CK 或 LS_CK，通過除頻器 ENMCD[1] 對頻率源進行除頻。因而 CPU 的工作頻率模式有多種可選擇，進而決定晶片的指令週期。

周邊外設工作頻率源同樣由外部或內部晶震透過 HS_CK 或 LS_CK 經過除頻器供給。或者由晶震直接供給頻率源，如 WDT。由於周邊外設工作性質不同而有不同的規劃，使用者可參考圖 周邊外設工作頻率配置圖

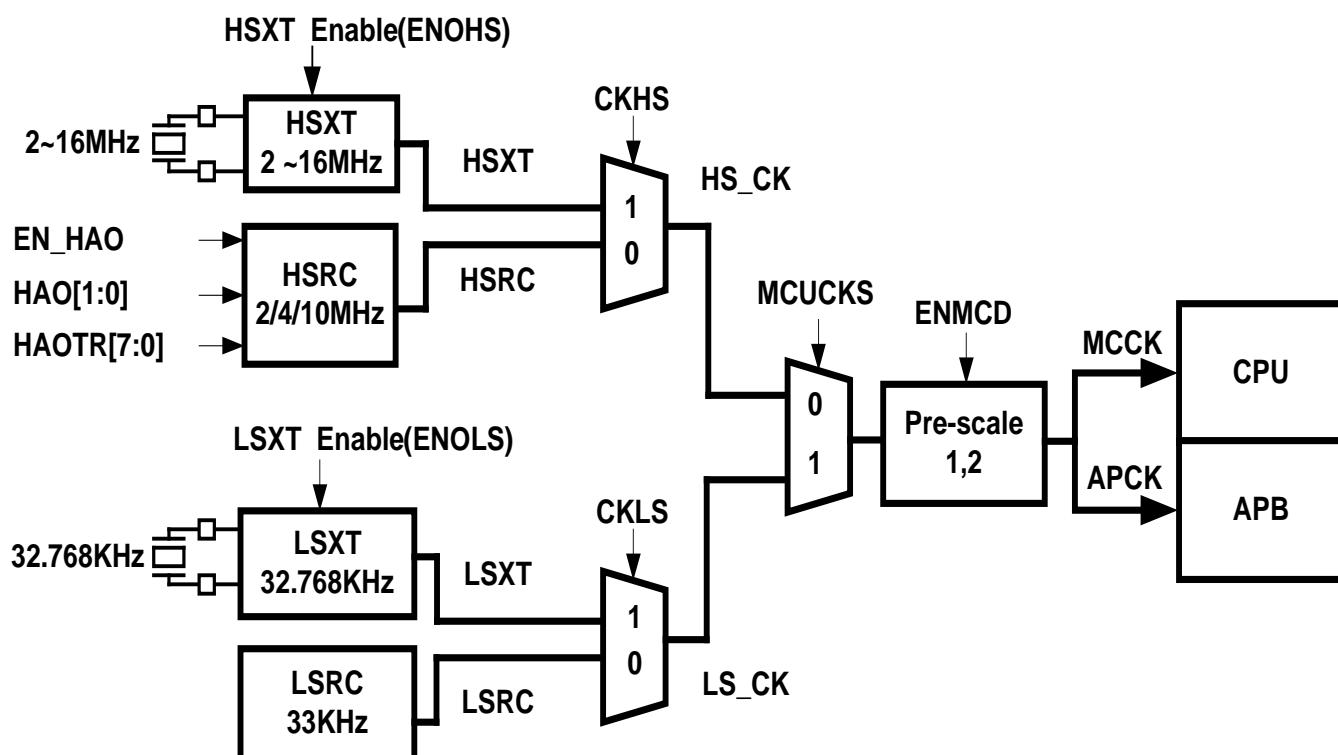


圖 6-1 CPU 工作頻率源配置圖

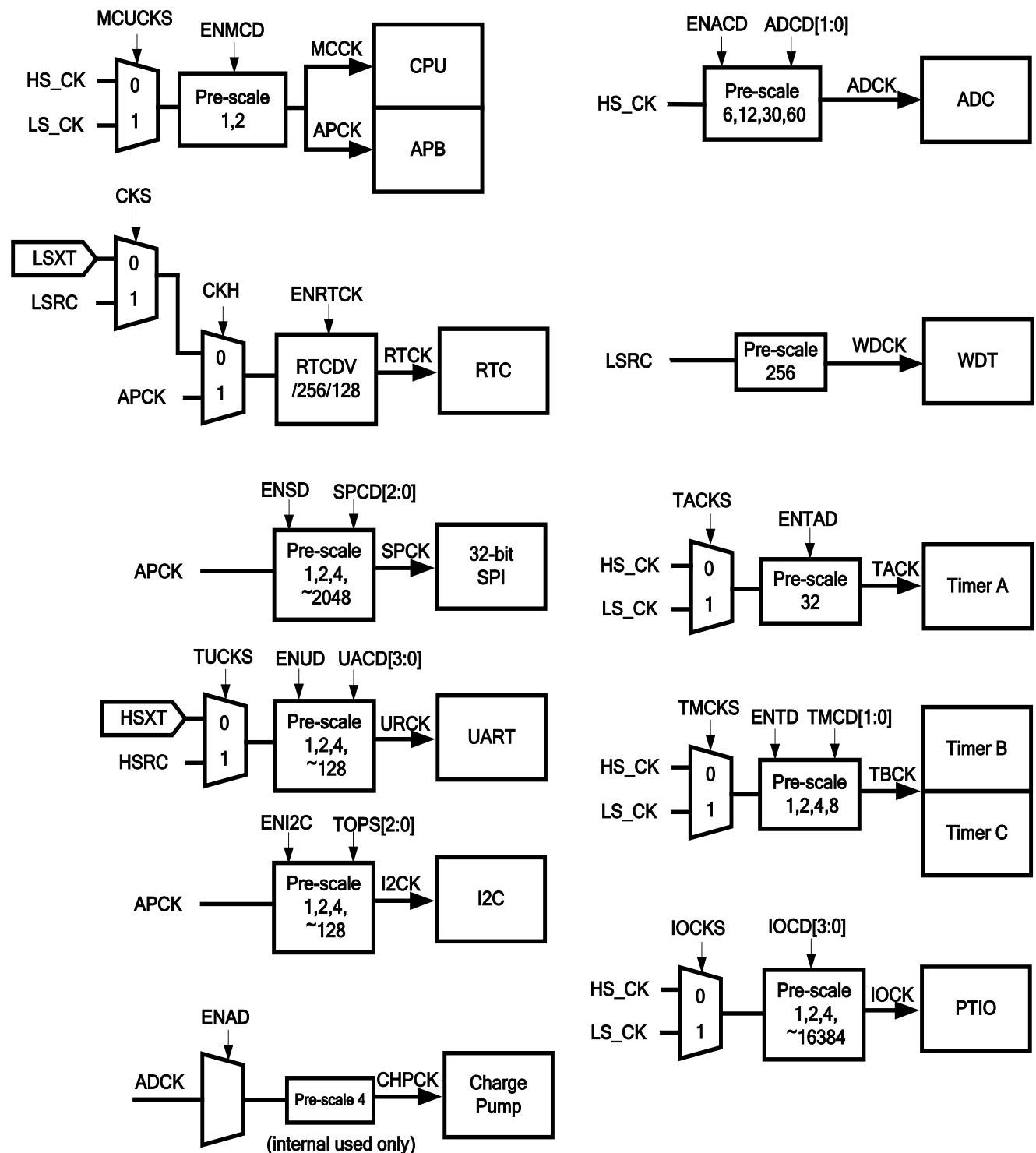


圖 6-2 周邊外設工作頻率配置圖

06.2 暫存器位址

Clock Register Address	31	24	23	16	15	8	7	0
CLK Base Address + 0X00 (0X40300)	-	-	-	-	MASK0	REG0		
CLK Base Address + 0X04 (0X40304)	-	-	-	-	-	HAOTR		
CLK Base Address + 0X08 (0X40308)	MASK1	REG1		MASK2	REG2			
CLK Base Address + 0X0C (0X4030C)	MASK3	REG3		MASK4	REG4			

-保留

06.3 暫存器功能

06.3.1 時脈系統暫存器 CLKCR0

Clock Base Address + 0X00 (0X40300)								
Symbol	CLK0 (Clock Control Register 0)							
Bit	[31:16]							
名稱	RSV(Reserved)							
RW	R-0							
Bit	[15:8]	[7]	[6]	[5]	[4:3]	[2]	[1]	[0]
名稱	MASK	OHS_HS	CKLS	CKHS	HAOM	ENOLS	ENOHS	ENHAO
RW	R0W-0							
RW-0								

位元	名稱	描述
Bit[7]	OHS_HS	外部震盪器模式選擇
		0 HSXT<4MHz
		1 HSXT>4MHz
Bit[6]	CKLS	晶片低速頻率源選擇
		0 內部低速震盪器(OSC_LSRC)
		1 外部低速振盪器(OSC_LSXT)
Bit[5]	CKHS	晶片高速頻率源選擇
		0 內部高速振盪器(OSC_HSRC)
		1 外部高速振盪器(OSC_HSXT)
Bit[4~3]	HAO	內部高速振盪器頻率模式設置
		[00] 2MHz
		[01] 4MHz
		[10] 10MHz
		[11] Reserved
Bit[02]	ENOLS	外部低速震盪器開啟控制
		0 關閉
		1 開啟
Bit[01]	ENOHS	外部高速振盪器開啟控制
		0 關閉
		1 開啟
Bit[00]	ENHAO	內部高速振盪器開啟控制
		0 關閉
		1 開啟

06.3.2 時脈系統暫存器 CLKCR1

Clock Base Address + 0X04 (0X40304)		
Symbol	CLK1 (Clock Control Register 1)	
Bit	[31:16]	
名稱	Reserved	
RW	R-0	
Bit	[15:8]	[7:0]
名稱	Reserved	HAOTR
RW	R-0	RW-80H

位元	名稱	描述	
Bit[7:0]	HAOTR	內部高速震盪器校正控制寄存器(Register Only)	
		0	置 0
		1	置 1

1LSB.Step = 0.125%
0000_0000 是最慢速度
1000_0000 是預設速度
1111_1111 為超頻速度

06.3.3 時脈系統暫存器 CLKCR2

Clock Base Address + 0X08 (0X40308)						
Symbol	CLK2 (Clock Control Register 2)					
Bit	[31:24]	[23]	[22]	[21]	[20]	[19:16]
名稱	MASK	ENRTCK	-	TUCKS	ENUD	UACD
RW	R0W-0	RW-0	-		RW-0	
Bit	[15:08]	[7]	[6]	[5:4]	[3]	[2]
名稱	MASK	TMCKS	ENTD	TMCD	TACKS	ENTAO
RW	R0W-0				RW-0	

位元	名稱	描述	
Bit[23]	ENRTCK	RTC 時脈源控制	
		0	關閉(RTC 寄存器不可寫入且無法解鎖)
		1	開啟(RTC 寄存器可解鎖)
Bit[21]	TUCKS	EUART 時脈源選擇	
		0	HSXT: 外部高速震盪器
		1	HSRC: 內部高速震盪器
Bit[20]	ENUD	EUART 時脈源開啟控制	
		0	關閉
		1	開啟

位元	名稱	描述
Bit[19~16]	UACD	EUART 時脈源除頻器設置
		0000 EUART 時脈源/ 1
		0001 EUART 時脈源/ 2
		0010 EUART 時脈源/ 4
		0011 EUART 時脈源/ 8
		0100 EUART 時脈源/ 16
		0101 EUART 時脈源/ 32
		0110 EUART 時脈源/ 64
		0111 EUART 時脈源/ 128
		1000 保留
		1001 保留
		1010 保留
		1011 保留
		1100 保留
		1101 保留
		1110 保留
		1111 保留
Bit[07]	TMCKS	Timer B,C 時脈源選擇
		0 HS_CK
		1 LS_CK
Bit[06]	ENTD	Timer B,C 時脈源開啟控制
		0 關閉
		1 開啟
Bit[5~4]	TMCD	Timer B,C 時脈除頻器設置
		00 clock/1
		01 clock/2
		10 clock/4
		11 clock/8
Bit[03]	TACKS	Timer A 時脈源選擇
		0 HS_CK
		1 LS_CK
Bit[02]	ENTAO	Timer A 時脈除頻器設置
		0 關閉除頻器
		1 Timer A clock/32
Bit[01]	ENMCD	MCU 輸入時脈除頻器設置
		0 MCU clock/1
Bit[00]	MCUCKS	MCU 輸入時脈源選擇
		0 HS_CK
		1 LS_CK

06.3.4 時脈系統暫存器 CLKCR3

Clock Base Address + 0X08 (0X4030C)					
Symbol	CLK3 (Clock Control Register 3)				
Bit	[31:24]	[23:21]	[20]	[19:16]	
名稱	MASK	-	IOCKS	IOCD	
RW	R0W-0	-		RW-0	
Bit	[15:08]	[7]	[6]	[5:4]	[3]
名稱	MASK	ADCKP	ENACD	ADCD	ENSD
RW	R0W-0			RW-0	SPCD

位元	名稱	描述
Bit[20]	IOCSK	IO 採樣頻率時脈源選擇
		0 HS_CK 1 LS_CK
Bit[19~16]	IOCD	I/O 採樣頻率時脈源除頻器設置
		0000 關閉除頻器
		0001 I/O 採樣頻率時脈源/1
		0010 I/O 採樣頻率時脈源/2
		0011 I/O 採樣頻率時脈源/4
		0100 I/O 採樣頻率時脈源/8
		0101 I/O 採樣頻率時脈源/16
		0110 I/O 採樣頻率時脈源/32
		0111 I/O 採樣頻率時脈源/64
		1000 I/O 採樣頻率時脈源/128
		1001 I/O 採樣頻率時脈源/256
		1010 I/O 採樣頻率時脈源/512
		1011 I/O 採樣頻率時脈源/1024
		1100 I/O 採樣頻率時脈源/2048
		1101 I/O 採樣頻率時脈源/4096
		1110 I/O 採樣頻率時脈源/8192
		1111 I/O 採樣頻率時脈源/16384

位元	名稱	描述
Bit[07]	ADCKP	ADC 輸入時脈源相位位移設置
		0 ADC 時脈上升沿為 CPU 時脈低電位
		1 ADC 時脈上升沿為 CPU 時脈高電位
Bit[06]	ENACD	ADC 時脈源開啟控制
		0 關閉
		1 開啟
Bit[5~4]	ADCD	ADC 時脈除頻器設置
		00 ADC clock/6
		01 ADC clock/12
		10 ADC clock/30
		11 ADC clock/60
Bit[03]	ENSD	SPI 時脈源開啟控制
		0 關閉
		1 開啟
Bit[2~0]	SPCD	SPI 時脈源除頻器設置
		000 SPI clock source/1
		001 SPI clock source/2
		010 SPI clock source/4
		011 SPI clock source/8
		100 SPI clock source/32
		101 SPI clock source/128
		110 SPI clock source/512
		111 SPI clock source/2048

07. 中斷控制系統

07.1 整體總說明

這個中斷模塊包含中斷啟動控制器、中斷使能控制器和中斷事件旗標寄存器，用於管理整體的中斷服務，如通訊功能中斷、定時器中斷、ADC 中斷、比較器中斷及 IO 外部中斷。晶片提供六級中斷源，同時也提供四級中斷響應優先權級別，也就是中斷 HW0~HW5 中斷向量函數響應的優先順序，如果當中斷響應優先權級別都設置相同的時候，則中斷響應優先權從高位到低位為 HW0、HW1 到 HW5。中斷服務由中斷事件旗標(INTF)，中斷事件服務智慧啟動(INTE)與中斷總控制 GIE 和向量位址 HW0~HW5 組成。當中斷事件成立且啟用中斷事件服務則，程式計數器 PC 在下一個指令週期會指向程式記憶體的中斷服務向量位址 HW0~HW5 以執行中斷服務程式。

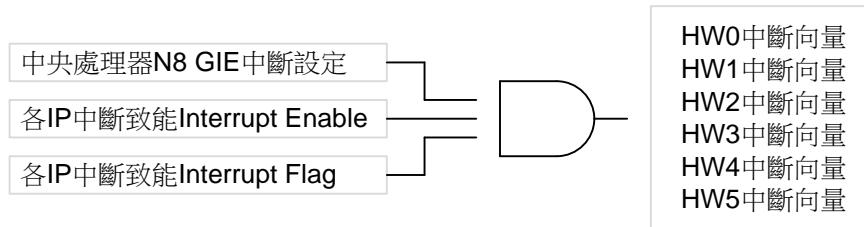


圖 7-1 中斷服務架構圖

操作細節說明:

- ◆ 用戶置 1 或清零相應中斷使能位，可實現開啟或關閉相應的中斷功能，寫 1 可以開啟中斷功能。
- ◆ 中斷事件發生後，會生產中斷請求旗標，使用者可自行清零旗標以便取消中斷請求。
- ◆ 必須開啟全局中斷使能位 **GIE=1**，否則無法響應任何中斷。
- ◆ 中斷向量優先權決定在多個中斷請求同時發生時，需先響應中斷優先權高的中斷向量。
- ◆ 當進入中斷向量服務程式，高級的中斷向量可終止當前中斷服務轉去執行高級中斷服務。
- ◆ 執行完中斷服務程式後，會自動返回中斷發生的程式位址，繼續往下執行程式

晶片中斷對應的中斷向量程式入口位址如下表

Interrupt Vector Address	N801	Interrupt Function
INT Base Address + 0x00 (COM)	HW0	void HW0_ISR(void)
INT Base Address + 0x04 (Timer ABC WDT RTC)	HW1	void HW1_ISR(void)
INT Base Address + 0x08 (ADC)	HW2	void HW2_ISR(void)
INT Base Address + 0x0C (CMP/OPA)	HW3	void HW3_ISR(void)
INT Base Address + 0x10 (PT1)	HW4	void HW4_ISR(void)
INT Base Address + 0x14 (PT2)	HW5	void HW5_ISR(void)

07.2 暫存器位址

Interrupt Register Address	31	24	23	16	15	8	7	0
INT Base Address + 0x00 (COM) (0X40000)	MASK0	REG0	MASK1	REG1				
INT Base Address + 0x04 (TMR) (0X40004)	MASK2	REG2	MASK3	REG3				
INT Base Address + 0x08 (ADC) (0X40008)	MASK4	REG4	MASK5	REG5				
INT Base Address + 0x0C (CMP) (0X4000C)	MASK6	REG6	MASK7	REG7				
INT Base Address + 0x10 (PT1) (0X40010)	MASK8	REG8	MASK9	REG9				
INT Base Address + 0x14 (PT2) (0X40014)	MASK10	REG10	MASK11	REG11				

-保留

07.3 暫存器功能

07.3.1 中斷控制暫存器 INTCOM

INT Base Address + 0X00 (0X40000)								
Symbol	INTCOM (Interrupt Control Register 0)							
Bit	[31:24]	[23:22]	[21]	[20]	[19]	[18]	[17]	[16]
名稱	MASK	-	I2CEIE	I2CIE	UTXIE	URXIE	STXIE	SRXIE
RW	R0W-0	-						RW-0
Bit	[15:08]	[7:6]	[5]	[4]	[3]	[2]	[1]	[0]
名稱	MASK	-	I2CEIF	I2CIF	UTXIF	URXIF	STXIF	SRXIF
RW	R0W-0	-						RW0-0

位元	名稱	描述
Bit[21]	I2CEIE	I2C 錯誤中斷向量控制
		0 關閉
		1 開啟
Bit[20]	I2CIE	I2C 中斷向量控制
		0 關閉
		1 開啟
Bit[19]	UTXIE	UART 發送(TX)中斷向量控制
		0 關閉
		1 開啟
Bit[18]	URXIE	UART 接收(RX)中斷向量控制
		0 關閉
		1 開啟
Bit[17]	STXIE	SPI 發送(TX)中斷向量控制
		0 關閉
		1 開啟
Bit[16]	SRXIE	SPI 接收(RX)中斷向量控制
		0 關閉
		1 開啟
Bit[05]	I2CEIF	I2C 錯誤中斷請求旗標(電平觸發)
		0 正常
		1 I2C 錯誤中斷請求
Bit[04]	I2CIF	I2C 中斷請求旗標(電平觸發)
		0 正常
		1 I2C 中斷請求
Bit[03]	UTXIF	UART 發送(TX)中斷請求旗標(電平觸發)
		0 正常
		1 UART 發送(TX)中斷請求
Bit[02]	URXIF	UART 接收(RX)中斷請求旗標(電平觸發)
		0 正常
		1 UART 接收(RX)中斷請求
Bit[01]	STXIF	SPI 發送(TX)中斷請求旗標(電平觸發)
		0 正常
		1 SPI 發送(TX)中斷請求

Bit[00]	SRXIF	SPI 接收(RX)中斷請求旗標(電平觸發)							
		0	正常						
		1	SPI 接收(RX)中斷請求						

07.3.2 中斷控制暫存器 INTTMR

INT Base Address + 0X04 (0X40004)								
Symbol	INTTMR (Interrupt Control Register 1)							
Bit	[31:24]	[23:22]	[21]	[20]	[19]	[18]	[17]	[16]
名稱	MASK	-	RTCIE	WDTIE	TMC1IE	TMC0IE	TMBIE	TMAIE
RW	R0W-0	-						RW-0
Bit	[15:08]	[7:6]	[5]	[4]	[3]	[2]	[1]	[0]
名稱	MASK	-	RTCIF	WDTIF	TMC1IF	TMC0IF	TMBIF	TMAIF
RW	R0W-0	-						RW0-0

位元	名稱	描述
Bit[21]	RTCIE	實時時鐘 RTC 中斷向量控制
		0 關閉
		1 開啟
Bit[20]	WDTIE	看門狗(WDT)中斷向量控制
		0 關閉
		1 開啟
Bit[19]	TMC1IE	TMC1 中斷向量控制
		0 關閉
		1 開啟
Bit[18]	TMC0IE	TMC0 中斷向量控制
		0 關閉
		1 開啟
Bit[17]	TMBIE	定時計數器 TMB 中斷向量控制
		0 關閉
		1 開啟
Bit[16]	TMAIE	定時計數器 TMA 中斷向量控制
		0 關閉
		1 開啟
Bit[05]	RTCIF	實時時鐘 RTC 中斷請求旗標
		0 正常
		1 實時時鐘 RTC 中斷請求
Bit[04]	WDTIF	看門狗(WDT)中斷請求旗標
		0 正常
		1 看門狗(WDT)中斷請求
Bit[03]	TMC1IF	TMC1 中斷請求旗標
		0 正常
		1 TMC1 中斷請求
Bit[02]	TMC0IF	TMC0 中斷請求旗標
		0 正常
		1 TMC0 中斷請求
Bit[01]	TMBIF	定時計數器 TMB 中斷請求旗標
		0 正常

Bit[00]	TMAIF	1	定時計數器 TMB 中斷請求
		0	定時計數器 TMA 中斷請求旗標
		1	正常

07.3.3 中斷控制暫存器 INTADC

INT Base Address + 0X08 (0X40008)			
Symbol	INTADC (Interrupt Control Register 2)		
Bit	[31:24]	[23:17]	[16]
名稱	MASK	-	ADCIE
RW	R0W-0	-	RW-0
Bit	[15:08]	[07:01]	[00]
名稱	MASK	-	ADCIF
RW	R0W-0	-	RW0-0

位元	名稱	描述	
Bit[16]	ADCIE	模數轉換器 ADC 中斷向量控制	
		0 關閉	
Bit[00]	ADCIF	模數轉換器 ADC 中斷請求旗標	
		0 正常	
		1 模數轉換器 ADC 中斷請求	

07.3.4 中斷控制暫存器 INTCMP

INT Base Address + 0X0C (0X4000C)				
Symbol	INTCMP (Interrupt Control Register 3)			
Bit	[31:24]	[23:18]	[17]	[16]
名稱	MASK	-	CPOIE	OPOIE
RW	R0W-0	-		RW-0
Bit	[15:08]	[07:02]	[01]	[00]
名稱	MASK	-	CPOIF	OPOIF
RW	R0W-0	-		RW0-0

位元	名稱	描述	
Bit[17]	CPOIE	多功能比較器輸出(CPO)中斷向量控制	
		0 關閉	
Bit[16]	OPOIE	低雜訊運算放大器輸出(OPO)中斷向量控制	
		0 關閉	
Bit[01]	CPOIF	多功能比較器輸出(CPO)中斷請求旗標	
		0 正常	
Bit[00]	OPOIF	1 多功能比較器輸出(CPO)中斷請求	
		低雜訊運算放大器輸出(OPO)中斷請求旗標	
		0 正常	
		1 低雜訊運算放大器輸出(OPO)中斷請求	

07.3.5 中斷控制暫存器 INTPT1

INT Base Address + 0X10 (0X40010)									
Symbol	INTPT1 (Interrupt Control Register 4)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名稱	MASK	PT17IE	PT16IE	PT15IE	PT14IE	PT13IE	PT12IE	PT11IE	PT10IE
RW	R0W-0								RW-0
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名稱	MASK	PT17IF	PT16IF	PT15IF	PT14IF	PT13IF	PT12IF	PT11IF	PT10IF
RW	R0W-0								RW0-0

位元	名稱	描述
Bit[23]	PT17IE	PT17IE: PT17 外部中斷向量控制
		0 關閉
		1 開啟
Bit[22]	PT16IE	PT16IE : PT16 外部中斷向量控制
		0 關閉
		1 開啟
Bit[21]	PT15IE	PT15IE : PT15 外部中斷向量控制
		0 關閉
		1 開啟
Bit[20]	PT14IE	PT14IE : PT14 外部中斷向量控制
		0 關閉
		1 開啟
Bit[19]	PT13IE	PT13IE : PT13 外部中斷向量控制
		0 關閉
		1 開啟
Bit[18]	PT12IE	PT12IE : PT12 外部中斷向量控制
		0 關閉
		1 開啟
Bit[17]	PT11IE	PT11IE : PT11 外部中斷向量控制
		0 關閉
		1 開啟
Bit[16]	PT10IE	PT10IE : PT10 外部中斷向量控制
		0 關閉
		1 開啟

位元	名稱	描述	
Bit[07]	PT17IF	PT17IF : PT17 外部中斷請求旗標	
		0	正常
		1	PT17 外部中斷請求
Bit[06]	PT16IF	PT16IF : PT16 外部中斷請求旗標	
		0	正常
		1	PT16 外部中斷請求
Bit[05]	PT15IF	PT15IF : PT15 外部中斷請求旗標	
		0	正常
		1	PT15 外部中斷請求
Bit[04]	PT14IF	PT14IF : PT14 外部中斷請求旗標	
		0	正常
		1	PT14 外部中斷請求
Bit[03]	PT13IF	PT13IF : PT13 外部中斷請求旗標	
		0	正常
		1	PT13 外部中斷請求
Bit[02]	PT12IF	PT12IF : PT12 外部中斷請求旗標	
		0	正常
		1	PT12 外部中斷請求
Bit[01]	PT11IF	PT11IF : PT11 外部中斷請求旗標	
		0	正常
		1	PT11 外部中斷請求
Bit[00]	PT10IF	PT10IF : PT10 外部中斷請求旗標	
		0	正常
		1	PT10 外部中斷請求

07.3.6 中斷控制暫存器 INTPT2

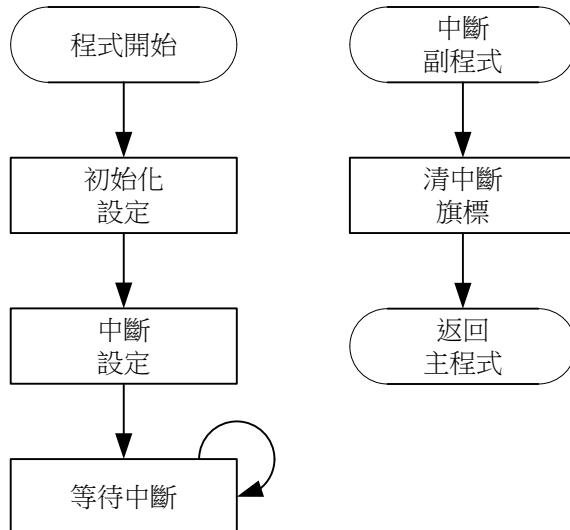
INT Base Address + 0X14 (0X40014)									
Symbol	INTPT2 (Interrupt Control Register 5)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名稱	MASK	PT27IE	PT26IE	PT25IE	PT24IE	PT23IE	PT22IE	PT21IE	PT20IE
RW	R0W-0								RW-0
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名稱	MASK	PT27IF	PT26IF	PT25IF	PT24IF	PT23IF	PT22IF	PT21IF	PT20IF
RW	R0W-0								RW0-0

位元	名稱	描述	
Bit[23]	PT27IE	PT27IE : PT27 外部中斷向量控制	
		0	關閉
		1	開啟
Bit[22]	PT26IE	PT26IE : PT26 外部中斷向量控制	
		0	關閉
		1	開啟

Bit[21]	PT25IE	PT25IE : PT25 外部中斷向量控制
		0 關閉 1 開啟
Bit[20]	PT24IE	PT24IE : PT24 外部中斷向量控制
		0 關閉 1 開啟
Bit[19]	PT23IE	PT23IE : PT23 外部中斷向量控制
		0 關閉 1 開啟
Bit[18]	PT22IE	PT22IE : PT22 外部中斷向量控制
		0 關閉 1 開啟
Bit[17]	PT21IE	PT21IE : PT21 外部中斷向量控制
		0 關閉 1 開啟
Bit[16]	PT20IE	PT20IE : PT20 外部中斷向量控制
		0 關閉 1 開啟
Bit[07]	PT27IF	PT27IF : PT27 外部中斷請求旗標
		0 正常 1 PT27 外部中斷請求
Bit[06]	PT26IF	PT26IF : PT26 外部中斷請求旗標
		0 正常 1 PT26 外部中斷請求
Bit[05]	PT25IF	PT25IF : PT25 外部中斷請求旗標
		0 正常 1 PT25 外部中斷請求
Bit[04]	PT24IF	PT24IF : PT24 外部中斷請求旗標
		0 正常 1 PT24 外部中斷請求
Bit[03]	PT23IF	PT23IF : PT23 外部中斷請求旗標
		0 正常 1 PT23 外部中斷請求
Bit[02]	PT22IF	PT22IF : PT22 外部中斷請求旗標
		0 正常 1 PT22 外部中斷請求
Bit[01]	PT21IF	PT21IF : PT21 外部中斷請求旗標
		0 正常 1 PT21 外部中斷請求
Bit[00]	PT20IF	PT20IF : PT20 外部中斷請求旗標
		0 正常 1 PT20 外部中斷請求

07.4 範常式流程

- (3)DrvTIMER_EnableInt(X)，表示將 Timer 系統中斷致能。
- (4)DrvTIMER_ClearIntFlag(x)，表示將 Timer 系統中斷旗標清除。
- (5)Timer A/B/C 與 WDT 隸屬於 HW1 中斷，使用格式為 void HW1_ISR(void)。



07.5 範常式功能

範例名稱:中斷使用方式與說明
範例說明:使用 WDT 中斷服務

07.6 範常式說明

00		
01	#include "HY16F18X.h"	
02		
03	void main(void)	
04	{	
05	DrvWDT_Open(E_IRQ,E_PRE_SCALER_D32);	//WDT IRQ Open Pre scale 32
06	DrvWDT_ClearWDT();	//Clear WDT
07	DrvTIMER_ClearIntFlag(E_WDT);	//Clear WDT interrupt flag
08	DrvTIMER_EnableInt(E_WDT);	//WDT interrupt enable
09	SYS_EnableGIE(7);	//Enable GIE All
10	while(1);	//Wait for Interrupt
11	}	
12	void HW1_ISR(void)	
13	{	
14	DrvTIMER_ClearIntFlag(E_WDT);	//Clear WDT interrupt flag
15	}	
16		

08.看門狗 WDT

08.1 整體總說明

看門狗 WDT 為顧名思義的為晶片的看守者，主要用於產生喚醒事件，或晶片發生不可預期當機後，能進行基礎自動重置功能。

- ◆ 運行模式
 - 看門狗計數器溢位產生復位信號，重新啟動晶片。
 - 可用軟體清零定時計數器。
- ◆ 休眠模式
 - 看門狗 WDT 關閉，無法使用。
- ◆ 待機模式
 - 看門狗計數器溢位產生中斷事件，喚醒晶片。

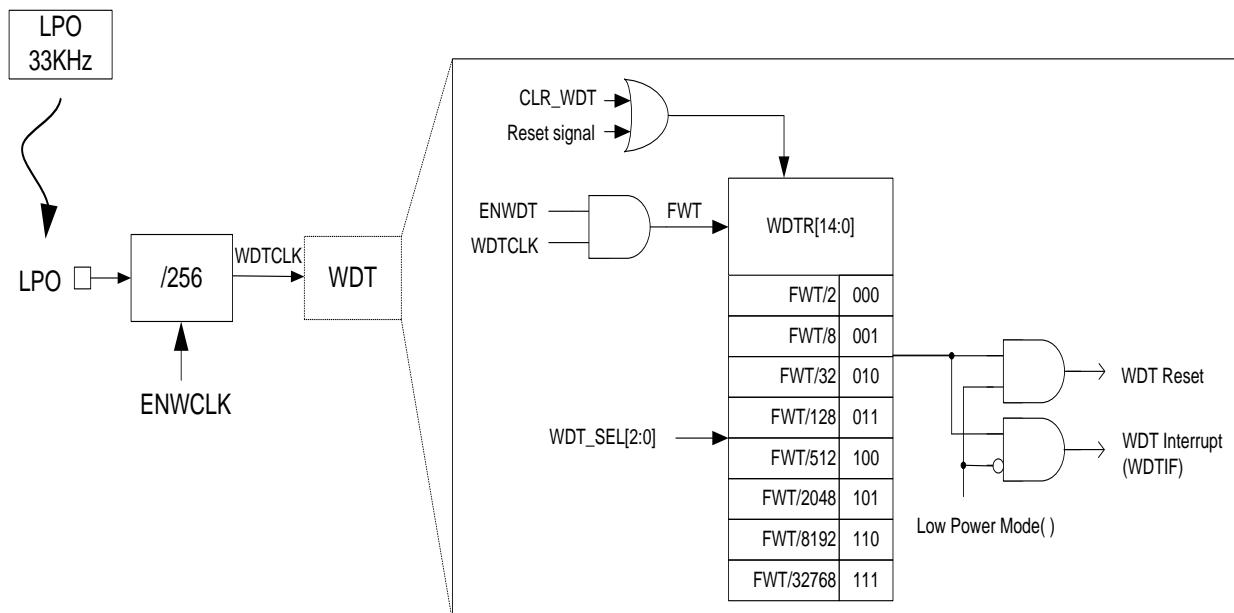


圖 8-1 WDT 方框圖

08.1.1 WDT 使用說明

設置除頻器 WDTP[2:0]可決定 WDT 計數器的工作頻率及計數溢出值，計數溢出後可產生 WDT 復位信號或中斷事件，控制位 WDTNMI[1]決定 WDT 計數溢出後產生復位信號還是中斷請求信號，寫入 0 時，WDT 會產生中斷請求信號。作為中斷模式請參考中斷控制章節的介紹。WDT 必須在晶片運行模式下啟動，將控制位 ENWDT[1]置<1>可啟動 WDT。開啟中斷功能需要先使能全局中斷控制位 GIE。

WDT 的工作頻率源為 LPOCK，因而 WDT 的工作頻率及計數溢出值的理論值計算方式如下：

$$WDT = LPOCK / 256 / WDTP[2:0] \quad (式 8-1)$$

LPOCK 為內部低速 RC 震盪器頻率，WDTP 為除頻器；

假設 LPOCK=33.9KHZ，WDTP=32768，則得到 WDT 工作頻率為：

$$33900\text{Hz}/256/WDT_PS(32768)=0.00404\text{Hz}$$

08.2 暫存器位址

SYS Register Address	31	24	23	16	15	8	7	0
SYS Base Address + 0X08(0X40108)	WDTO1	WDTO0	MASK0	REG0				

08.3 暫存器功能

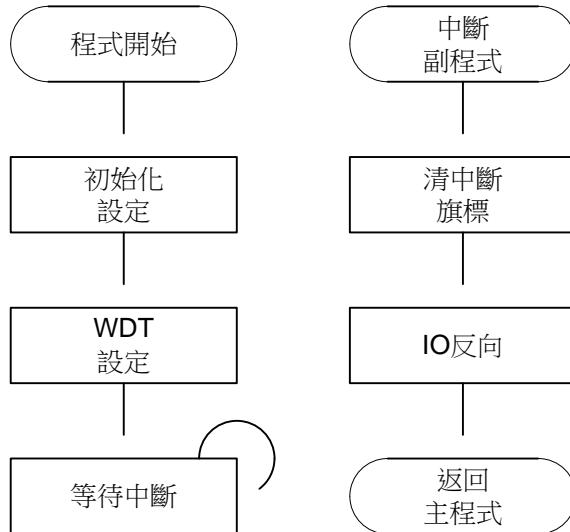
08.3.1 WDT 暫存器 WDTCR

SYS Base Address + 0X08 (0X40108)							
Symbol	WDTCR (WDT Control Register)						
Bit	[31]	[30:16]					
名稱	-	WDTO					
RW	-	R-0					
Bit	[15:8]	[7]	[6]	[5]	[4]	[3]	[2:0]
名稱	MASK	-	WDT_RST	CLWDT	ENWDT	-	WDTP
RW	R0W-0	-	RW1-0	RW-0	RW1-0	-	RW-7

位元	名稱	描述
Bit[30~16]	WDTO	看門狗的計數寄存器
		0 置 0
		1 置 1
Bit[06]	WDT_RST	看門狗中斷工作模式選擇
		0 定時計數模式(Timer mode)
		1 復位元模式(Reset mode)
Bit[05]	CLWDT	清零看門狗的計數寄存器
		0 關閉
		1 啟動
Bit[04]	ENWDT	看門狗開啟控制
		0 關閉
		1 開啟
Bit[2~0]	WDTP	看門狗計數溢出值設置
		[000] 0 : WCLK / 2
		[001] 1 : WCLK / 8
		[010] 2 : WCLK / 32
		[011] 3 : WCLK / 128
		[100] 4 : WCLK / 512
		[101] 5 : WCLK / 2048
		[110] 6 : WCLK / 8192
		[111] 7 : WCLK / 32768

08.4 範常式流程

- (1)時脈操頻設定，只能設為 35KHz(低速內部震盪器)。
- (2)DrvWDT_Open()會將 WDT 致能。
- (3)DrvTIMER_EnableInt(E_WDT)，表示將 WDT 中斷致能。
- (4)DrvTIMER_ClearIntFlag(E_WDT)，表示將 WDT 中斷旗標清除。
- (5)Timer A/B/C 與 WDT 隸屬於 HW1 中斷，使用格式為 void HW1_ISR(void)。



08.5 範常式功能

範例名稱:WDT 使用方式與說明

範例說明:

- (1)使用 WDT 中斷。
- (2)每進一次 WDT 中斷，會將 IO 反向輸出。
例如進中斷前 PT2=0X05，進中斷後變 PT2=0X0A。

08.6 範常式式說明

00	
01	#include "HY16F18X.h"
02	
03	unsigned int i;
04	
05	int main(void)
06	{
07	i=0X05;
08	DrvGPIO_Open(E_PT2,0X0F,E_IO_OUTPUT); //PT2_0~3 Set Output
09	DrvGPIO_SetPortBits(E_PT2,i); //PT2 Output i=0X05
10	
11	DrvWDT_Open(E_IRQ,E_PRE_SCALER_D32); //WDT IRQ open Prescaler 32
12	
13	DrvWDT_ClearWDT(); //Clear WDT
14	DrvTIMER_EnableInt(E_WDT); //WDT interrupt enable
15	
16	SYS_EnableGIE(7); //Enable GIE
17	
18	while(1); //Wait for Interrupt
20	}
21	
22	void HW1_ISR(void)
23	{
24	DrvTIMER_ClearIntFlag(E_WDT); //Clear WDT interrupt flag
25	i=i^0XF; //i XOR 0XF
26	DrvGPIO_SetPortBits(E_PT2,i); //PT2 Output i=0X0A~0X05
27	}
28	

09.定時器 Timer A

09.1 整體總說明

定時器 A 是 1 個 16 位元遞增型計數器，可工作於運行模式和待機模式。可用於產生不同輸出頻率。

主要特性:

- ◆ 遞增型計數器
- ◆ 16 階溢位值選擇
- ◆ 計數溢出產生中斷事件
- ◆ 可讀取計數器的值

定時器 A(TMA)的初始組態:

TMA 是一個 16 位的遞增型計數器。它的輸入時脈源為 TACK，且在 TACK 的每一個上升沿進行計數，輸入時脈源的頻率由時脈系統管理模塊控制。控制位 ENTA[1]被置 1 或寫 0，可開啟或關閉 TMA 功能。

TMA 的計數溢出值通過除頻器 TAS[3:0]進行設置，用戶可通過改變除頻器 TAS 的值來改變計數溢出值，達到不同頻率的計數值。控制位 TACLR[1]被置<1>，可是 TMA 發生復位，計數寄存器變為 0，計數器寄存器被清零後，控制位 TACLR 自動為 0。

TMA 計數溢出後，便會產生中斷請求，TMA 中斷旗標 TMAIF 被置<1>，若開啟 TMA 的中斷功能，且全局中斷控制位又被置<1>，晶片就會響應 TMA 的中斷請求進入 TMA 中斷服務事件。清零 TMA 中斷請求旗標，可取消 TMA 中斷請求，此時晶片便不響應 TMA 中斷。在待機模式下，TMA 中斷可用於喚醒。在休眠模式下，TMA 中斷不可用。

TMA 擁有 16 階除頻設置，使得 TMA 具有比較寬的計數範圍，TMA 計數溢出值計算方式：

$$TAR[15:0]=1/\left(TACK/32/TAS[3:0]\right) \quad (\text{式 9-1})$$

TACK 是 TMA 輸入時脈源，TAS[3:0]為除頻值；

假設 TMA 選擇 LS_CK，LS_CK 來自 LPO，在此假設 LPO 為 35KHZ，則 TACK=35KHZ，
TAS[3:0]=1001B=1024，則 Timer A 計數溢出理論值為:

$$35000\text{Hz}/32/TAS(1024)=35000\text{Hz}/32/1024=1.068\text{Hz}$$

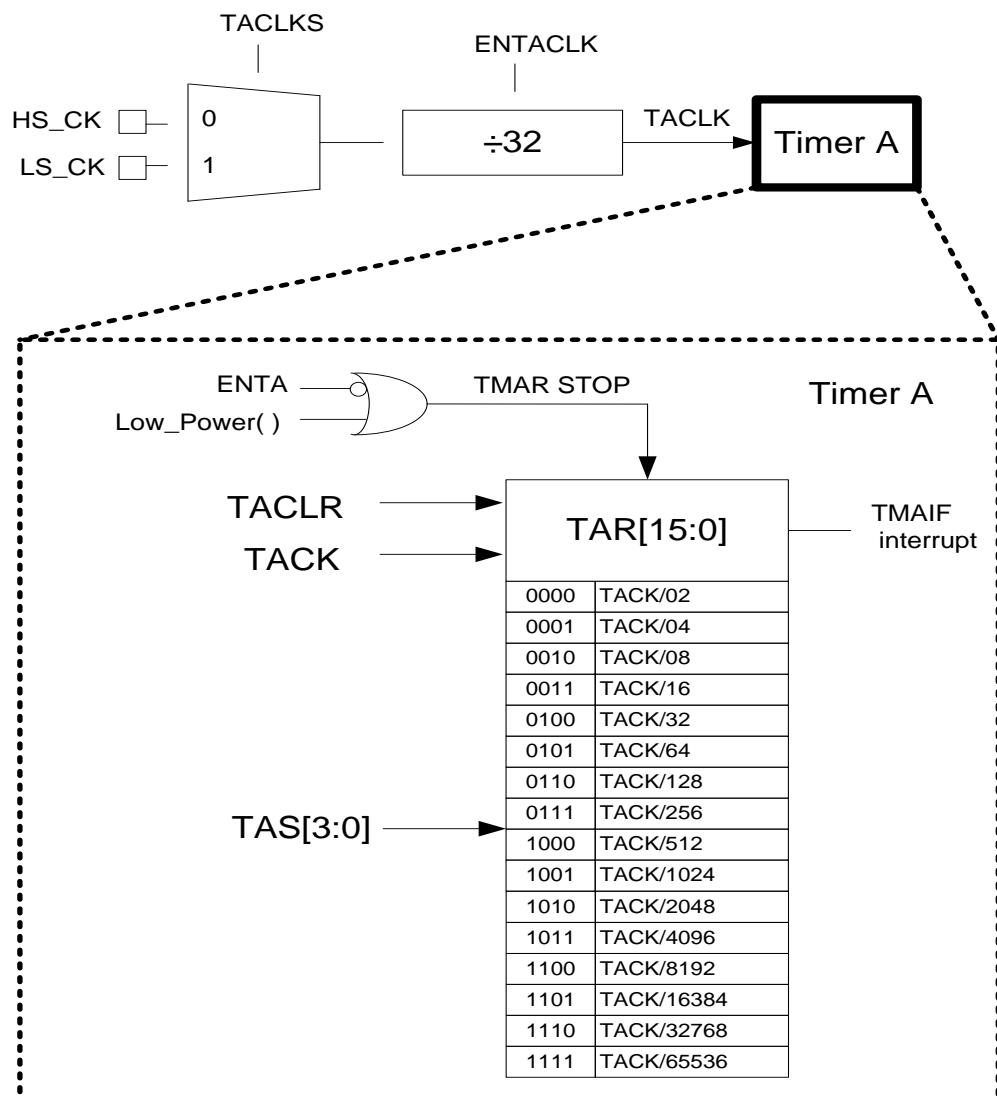


圖 9-1 定時計數器 A 方塊圖

09.2 暫存器位址

TMR Register Address	31	24	23	16	15	8	7	0
TMA Base Address + 0X00(0X40C00)	TAR1		TAR0		MASK0		REG0	

09.3 暫存器功能

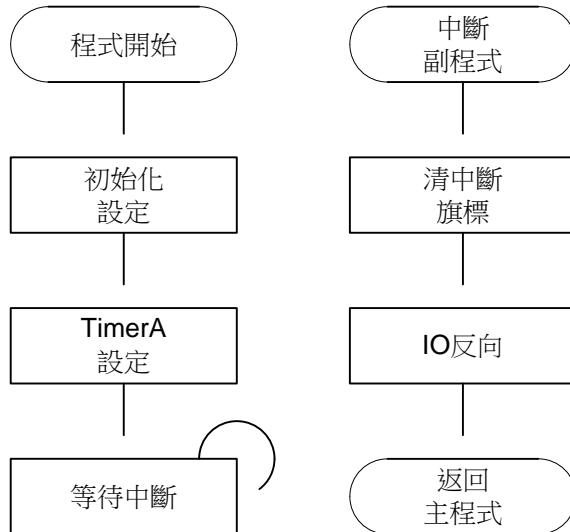
09.3.1 Timer A 暫存器 TMACR

TMA Base Address + 0X00 (0X40C00)							
Symbol	TMACR(TMA Control Register)						
Bit	[31:16]						
名稱	TAR						
RW		R-0					
Bit	[15:8]	[7:6]	[5]	[4]	[3:0]		
名稱	MASK	-	ENTA	TACLR	TAS		
RW	R0W-0	-	RW-0		RW-0XF		

位元	名稱	描述
Bit[31-16]	TAR	Timer A Counter 計數值 TAR[31:16] 是 16-bit Timer A 計數寄存器的計數值，輸出值從 MSB 至 LSB
Bit[5]	ENTA	開啟定時計數器 Timer A 0 關閉 1 開啟
Bit[4]	TACLR	清除 Timer A 計數值 0 正常 1 清零（清零完成後該位自動變為 0）
Bit[3~0]	TAS	Timer A 除頻器設定 0000 Timer A clock/2 0001 Timer A clock/4 0010 Timer A clock/8 0011 Timer A clock/16 0100 Timer A clock/32 0101 Timer A clock/64 0110 Timer A clock/128 0111 Timer A clock/256 1000 Timer A clock/512 1001 Timer A clock/1024 1010 Timer A clock/2048 1011 Timer A clock/4096 1100 Timer A clock/8192 1101 Timer A clock/16384 1110 Timer A clock/32768 1111 Timer A clock/65536

09.4 範常式流程

- (1) 時脈操頻設定，預設為 2MHz(高速內部震盪器)，最高可達 10MHz。
(2) 採用紜康 C 函式庫，DrvTMA_Open(X,Y) 可將 TimerA IP 與 Timer A Clock 啟動。
 其中 X 代表 TMAR 除頻，選擇 15 即除以 65536，Y 為選擇高速或低速 CPU Clock 進入 Timer A
(3) 紜康 C 函式庫 DrvTIMER_EnableInt(E_TMA)，表示將 Timer A 中斷致能。
(4) 紜康 C 函式庫 DrvTIMER_ClearIntFlag(E_TMA)，表示將 Timer A 中斷旗標清除。
(5) Timer A/B/C 與 WDT 隸屬於 HW1 中斷，使用格式為 void HW1_ISR(void)。



09.5 範常式功能

範例名稱: Timer A 使用方式與說明

範例說明:

- (1) 使用 Timer A 中斷。
(2) 每進一次 Timer A 中斷，會將 IO 反向輸出。
 例如進中斷前 PT2=0X05，進中斷後變 PT2=0X0A。

09.6 範常式說明

00	
01	#include "HY16F18X.h"
02	
03	unsigned int i;
04	
05	int main(void)
06	{
07	i=0X05;
08	DrvGPIO_Open(E_PT2,0X0F,E_IO_OUTPUT); //PT2_0~3 Set Output
09	DrvGPIO_SetPortBits(E_PT2,i); //PT2 Output i=0X05
10	
11	DrvTMA_Open(15,0); //Timer A Overflow
12	//15 :TACLK/65536/32;TMRDV=/32
13	// 0 : HS_CK
14	
15	DrvTIMER_ClearIntFlag(E_TMA); //Clear Timer A interrupt flag
16	DrvTIMER_EnableInt(E_TMA); //Timer A Interrupt Enable
17	
18	SYS_EnableGIE(7); //Enable GIE
20	
21	while(1); //Wait for Interrupt
22	}
23	
24	void HW1_ISR(void)
25	{
26	DrvTIMER_ClearIntFlag(E_TMA); //Clear TMA interrupt Flag
27	i=i^0XF; //i XOR 0XF
28	DrvGPIO_SetPortBits(E_PT2,i); //PT2 Output i=0X0A~0X05
29	}
30	

10.定時器 Timer B

10.1 整體總說明

定時計數器 B 是一個 16 位的計數器，可用於時間計數或時間控制，時鐘產生、延遲時間等功能。它在計數溢出時發生中斷信號，程式可讀取 TM B 當前計數值；TMB 還用於 PWM 波形的產生。它可工作於運行模式和待機模式。

定時計數器 B 的 16 位計數寄存器可拆分為兩個獨立的 8 位計數寄存器，因而 TMB 具有四種計數方式：

- ◆ 16 位元遞增計數方式，可產生中斷信號；
- ◆ 16 位元計數方式，先遞增至溢出值，然後從溢出值遞減至 0；可產生中斷信號
- ◆ 兩組獨立的 8 位元遞增計數方式，最高計數值為 0xFF，可產生中斷信號
- ◆ 8+8 位元遞增計數方式，低 8 位元計數器溢出，高 8 位計數器才自動加 1，可產生中斷信號

同時 TMB 具有三個計數溢出控制器：TBC0、TBC1、TBC2，TMB 可複用為 PWM 波形發生器，具有兩路 PWM 波形 PWM0/PWM1，且每路 PWM 都具有多種工作模式，可適應不同的 PWM 波形輸出需求，工作模式如下：

- ◆ PWMA /PWMB /PWMC /PWMD /PWMF /PWMG

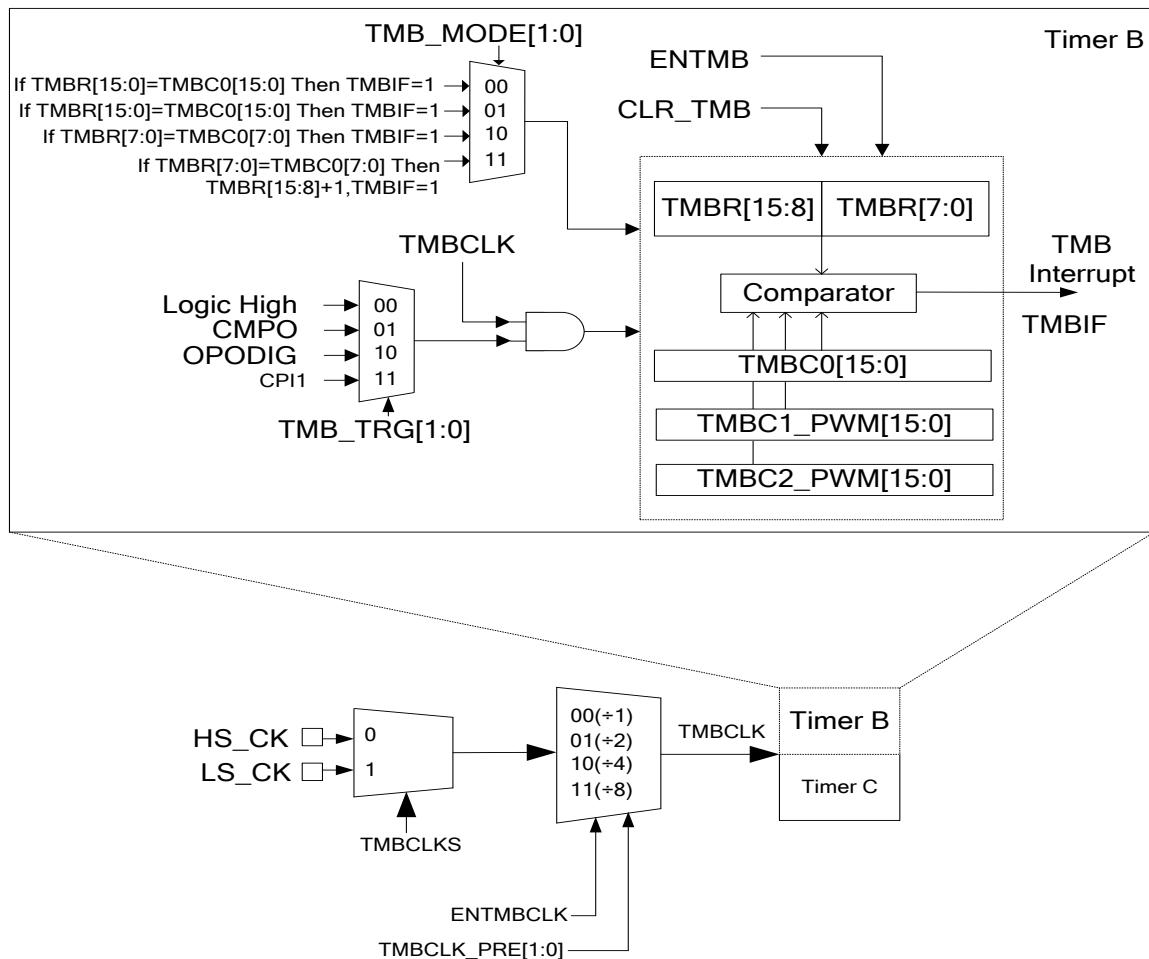


圖 10-1 定時計數器 B 方框圖

10.1.1 定時計數器模式

定時器 B(TMB)是一個 16 位元遞增型計數器，可用於脈衝寬度調製 (PMW) 波形的產生。可執行時間計數或時間控制、時間計算、時鐘產生等功能，且可在計數溢出時產生中斷信號。TMB 可工作於運行模式和待機模式。具有 4 種不同計數方式，可產生不同頻率的定時計數值：

- ◆ 16 位元遞增計數方式，可產生中斷信號；
- ◆ 16 位元計數方式，先遞增至溢出值，然後從溢出值遞減至 0；可產生中斷信號
- ◆ 兩組獨立的 8 位元遞增計數方式，最高計數值為 0xFF，可產生中斷信號
- ◆ 8+8 位元遞增計數方式，低 8 位元計數器溢出，高 8 位計數器才自動加 1，可產生中斷信號

具有 4 種不同的觸發計數信號源，可適用於不同事件計數：

- ◆ 總是啟用，連續計數方式；
- ◆ 比較器輸出(CMPO)高電位觸發
- ◆ 運算放大器數字輸出 (OPOD) 高電位觸發
- ◆ Timer C 輸出(CPI1)高電位觸發

TMB 的工作時脈源為 HS_CK 或 LS_CK，經過除頻器產生頻率源 TBCLK，提供給 TMB 工作頻率。提供設置除頻器 TMCD[1:0]，可設置 TMB 不同計數週期。在時脈系統控制模塊可設置 TMB 的時脈源。

TBR：16 位元定時/計數暫存器。

TBR 是個 16 位元的定時/計數暫存器，可拆分為兩個獨立 8 位的定時/計數暫存器，以適應 TMB 四種不同的計數方式。TBR 在 TBCLK 的上升沿進行遞增或遞減，在不同的計數方式下，TBR 以不同的條件發生遞增或遞減。TBR 可通過設置控制位 TBRST[1]為<1>，自動清零，並且清零後控制位 TBRST 自動為 0。程式上亦可讀取 TBR 當前計數值，以作為其他用途。

TBEN 是 TMB 的使能控制信號，該位元被寫 1，開啟 TMB 計數功能，被寫 0 則關閉 TMB 計數功能。

TBEBS[1:0]是 TMB 的觸發計數信號源控制器，透過該控制器可設置 4 種不同的觸發計數信號源。

TBM[1:0]是 TMB 的計數方式控制器，透過設置該控制器可設置 4 種不同的計數方式。

TBRST 是 TMB 計數暫存器自動清零的控制位，該位被置<1>時，自動清零計數暫存器，完成後該位自動為 0。

TMB 作為定時計數器操作設置：

- ◆ 設置 TMB 工作時脈源，設置控制位 0X40308[6](ENTD)及 0X40308[5:4](TMCD)；
- ◆ 選擇計數模式，設置控制位元 TBM[1:0]；
- ◆ 選擇觸發計數信號源，設置 TBEBS[1:0]，作為定時器，可設置為 00b，即總是啟用，連續計數；
- ◆ 設置定時計數溢出值 TBC0[15:0]；
- ◆ 控制位 TBRST 置<1>，清零計數暫存器；
- ◆ 使能 TMB，控制位 TBEN 被置<1>。

Timer B 溢出理論值_計算方式：

$$T = TBC0 * 1 / TBCLK; TBCLK=HS_CK(或 LS_CK) / TMCD; \quad (\text{式 10-1})$$

則

$$T=TBC0 * TMCD / HS_CK(或 LS_CK); \quad (\text{式 10-2})$$

TMB 具有四種不同的計數方式，不同的計數方式具有不同的溢出條件，以下做詳細介紹。

TMB 計數方式 0

當 TBM [1:0] =00，TBR 作為 16 位遞增型計數器。TMB 工作於該模式，TBR 會在 TBCLK 的每一個上升沿自動加 1，若 TBR 計數值大於 TBC0，TBR 就會在 TBCK 的下一個上升沿變成 0，且定時器中斷請求旗標 TMBIF 被置 $<1>$ ，若此時開啟 TMB 的中斷功能及使能全局中斷功能，晶片就會響應 TMB 中斷。然後，TBR 重新開始遞增計數。該模式的計數波形示意圖如圖 10-2 所示。

此模式下 TMB 的計數週期計算方式： $T=TBC0*TMCD / HS_CK(或 LS_CK)$

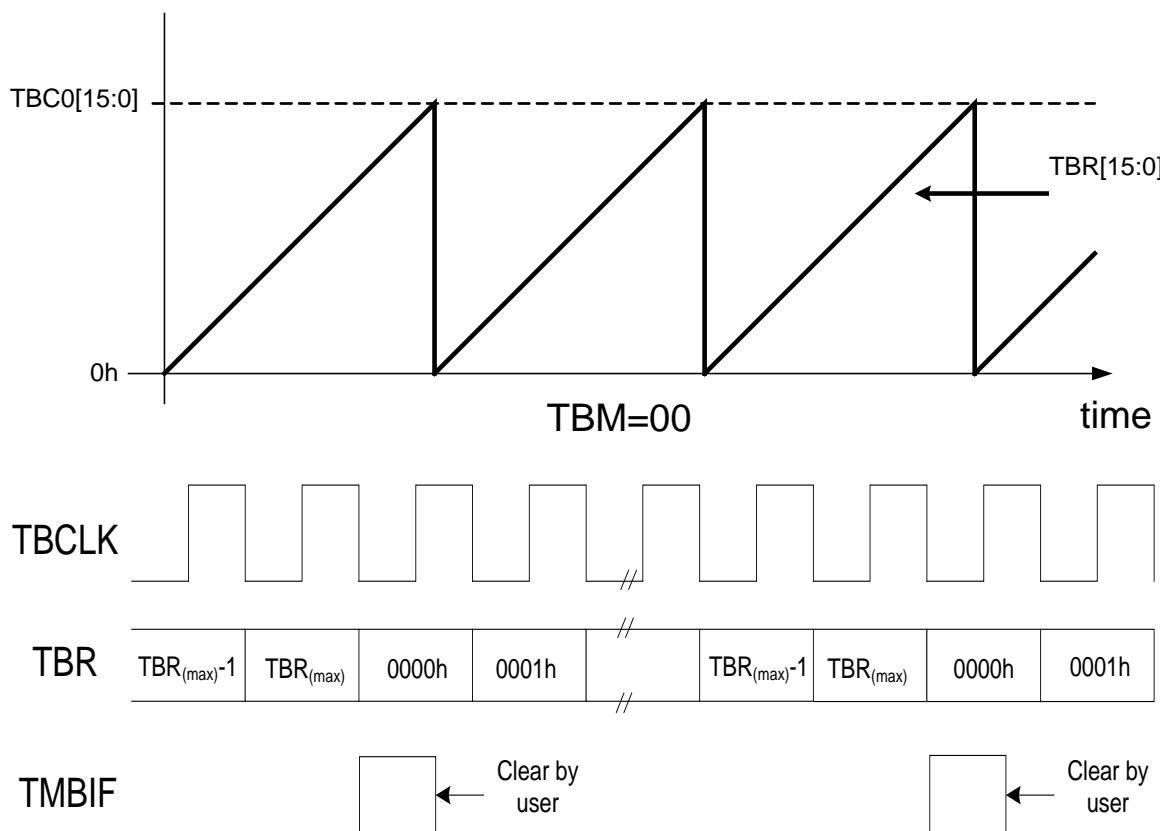


圖 10-2 計數方式 0 的計數波形示意圖

TMB 計數方式 1

當 TBM [1:0] =01，TMB 工作於先遞增後遞減的計數方式，TBR 是 16 位元計數器。啟動 TMB 後，TBR 先遞增計數，TBR 會在 TBCLK 的每一個上升沿自動加 1。當 TBR 等於 TBC0，TBR 就會改變成向下模式，但是中斷請求旗標 TMBIF 還是為 0，在 TBCLK 的下一個上升沿開始，TBR 變為遞減計數，直到 TBR 遞減至 0 時，發生中斷請求，中斷旗標 TMBIF 被置 $<1>$ ，且 TBR 在 TBCLK 的下一個上升沿開始遞增計數模式，如此循環運行。該模式的計數波形示意圖如圖 10-3 所示。

此模式下 TMB 的計數週期計算方式為： $T=2*TBC0*TMCD / HS_CK(或 LS_CK)$

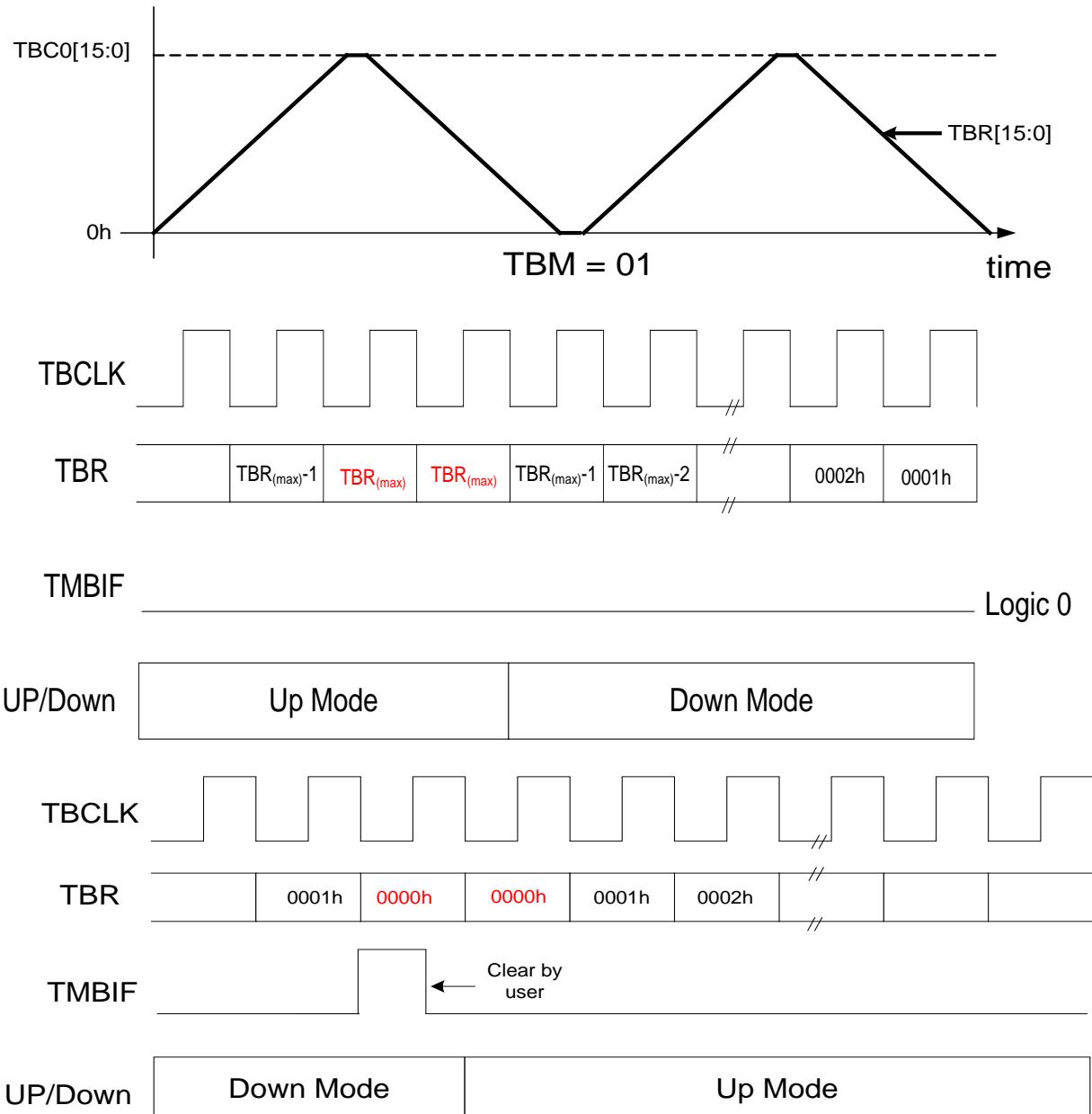


圖 10-3 計數方式 1 的計數波形示意圖

TMB 計數方式 2

當 $TBM[1:0]=10$ ，TMB 工作與遞增計數方式，但 TBR 被拆分為兩個 8 位元的獨立計數器：TBR [15:8]和 TBR [7:0]。且兩個兩個獨立的 8 位元是同時遞增計數的。TBR [15:8]的溢出值由 $TBC0[15:8]$ 所控制，而 TBR [7:0]的溢出值由 $TBC0[7:0]$ 所控制。這兩個計數器在 TBCLK 的每一個上升沿自動加 1，若 TBR [15:8]等於 $TBC0[15:8]$ ，則 TBCLK 的下一個上升沿會將 TBR [15:8]變成 0，但中斷請求旗標 TMBIF 仍維持為 0；若 TBR [7:0]等於 $TBC0[7:0]$ ，則會在 TBCLK 的下一個上升沿將 TBR [7:0]變成 0，且同時將中斷請求旗標 TMBIF 置 $<1>$ ，此時若開啟 TMB 中斷功能及全局中斷使能，晶片就會響應 TMB 中斷。該模式是由計數器 TBR[7:0]I 控制中斷請求，因而在使用該模式時，需要注意設置 $TBC0[7:0]$ 的值，以便控制 TMB 中斷向量。該模式的計數波形示意圖如圖 所示。

此模式的中斷方式計數週期計算方式為： $T=TBC0[7:0]*TMCD / HS_CK(或 LS_CK)$ ；

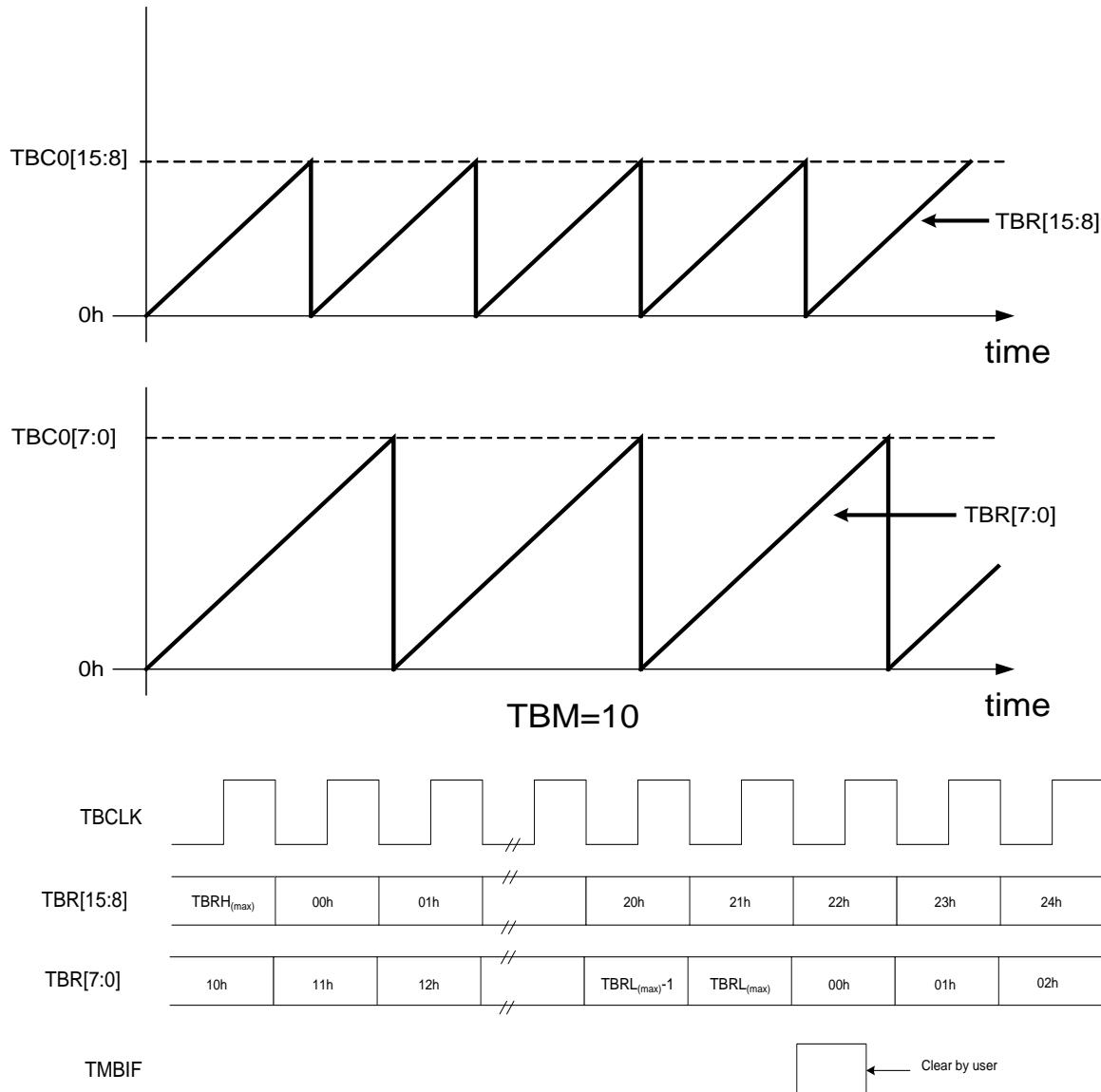


圖 10-4 計數方式 2 的計數波形示意圖

TMB 計數方式 3

當 $TBM[1:0]=11$ ，TMB 工作於遞增計數模式，TBR 會分解為兩個計數器：TBR [15:8]和 TBR [7:0]，兩者都是遞增計數模式。TBR [7:0]的溢出值由 TBC0 [7:0]所控制。TBR [7:0]會在 TBCLK 的每一個上升沿自動加 1，若 TBR [7:0]等於 TBC0 [7:0]，則 TBR 會在 TBCLK 的下一個上升沿變成 0。此外，TMBIF 會變成 1 且 TBR [15:8] 會增加 1。此時若開啟 TMB 中斷功能及使能全局中斷，晶片便響應中斷請求。該模式的計數波形示意圖如圖 所示。

此模式中斷方式的計數週期為： $T=TBC0[7:0]*TMCD / HS_CK(或 LS_CK)$ ；

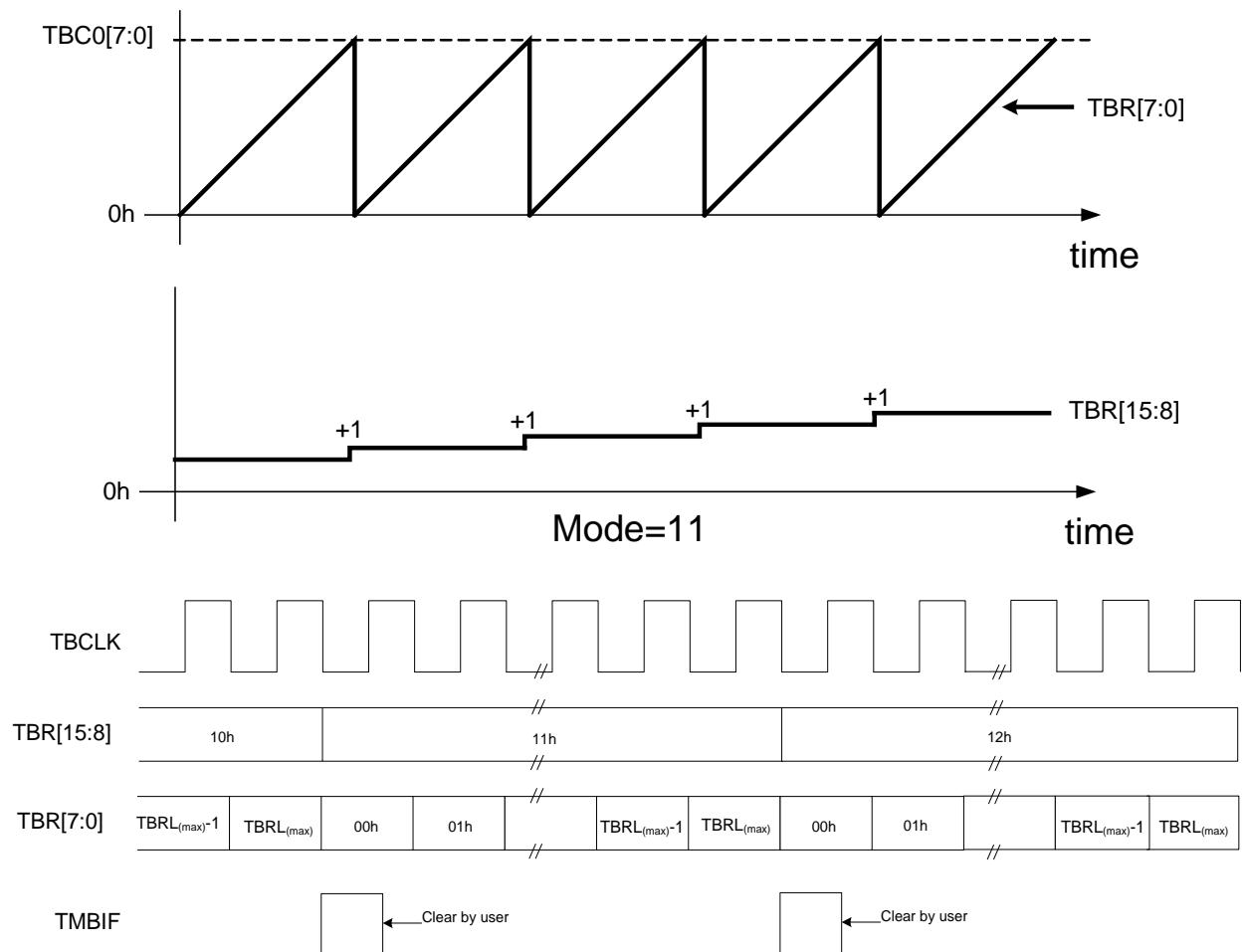


圖 10-5 計數模式 3 的計數波形示意圖

14.1.2 PWM 功能模式

定時器 B 工作於 PWM 模式，當 TMB 不同的計數方式與不同脈衝寬度調製（PWM）模式選擇器組合使用，可產生多種形式的 PWM 波形。晶片只有兩路 PWM：PWM0/PWM1；可簡單的認為這是兩個 PWM 波形發生器，根據 TMB 不同的計數方式，及 PWM 不同的工作模式，兩者組合便可產生多種形式 PWM 波形。晶片為 PWM 輸出提供多個輸出管腳，沒一個 PWM 波形發生器對應 8 個輸出 IO 口，所以 PWM 的使用及輸出比較靈活。但是該功能需要搭配 TMB 使用，也即是需要開啟 TMB 及設置 TMB 的計數週期值。

兩路 PWM 波形（PWM0/PWM1）都具有多種工作模式：PWMA、PWMB、PWMC、PWMD、PWMF、PWMG；通過設置控制器 O0MD[2:0]、O1MD[2:1]，分別設置 PWM0、PWM1 的工作模式。控制位 O1PMR、O0PMR 的設置，可改變 PWM 輸出波形的相位。可透過 PWM 工作模式旗標暫存器查看當前的 PWM 工作模式，旗標為 1 標明該工作模式被使能。TBC1/TBC2 分別為 PWM0/PWM1 的占空比控制器，設置 TBC1/TBC2 的值，可調節輸出 PWM 的占空比。

晶片為每一路 PWM 提供 8 個輸出 IO，對應管腳分佈在 PT1/PT2，由控制器 PTPW[2:0]、PTPW1E、PTTPW0E 控制 PWM1、PWM0 的輸出管腳的選擇與開啟。控制 PWM 輸出管腳的開啟與關閉，可以控制到 PWM 的輸出與關閉，若想完全關閉 PWM 功能，只能關閉 TMB。PWM 輸出管腳如表 10-1 所示。

序號 PTPW[2:0]	PWM0 輸出管腳	PWM1 輸出管腳	序號 PTPW[2:0]	PWM0 輸出管腳	PWM1 輸出管腳
000	PT1.0	PT1.1	100	PT2.0	PT2.1
001	PT1.2	PT1.3	101	PT2.2	PT2.3
010	PT1.4	PT1.5	110	PT2.4	PT2.5
011	PT1.6	PT1.7	111	PT2.6	PT2.7

表 10-1 PWM 輸出引腳分佈

PWM 操作說明：

- ◆ 設置 TMB 的工作時脈頻率源，設置 ENTD=1，TMCD[1:0]；
- ◆ 選擇 TMB 的計數方式及觸發計數信號源，設置控制位 TBM、TBEBS；
- ◆ 設置 TMB 的計數週期值，並開啟 TMB，即 TBEN=1，寫入值至 TBC0；
- ◆ 選擇 PWM 工作模式及占空比、輸出波形相位，即設置 O0MD/O0PMR、O1MD/O1PMR，寫入值至 TBC1/TBC2；
- ◆ 選擇 PWM 輸出 IO 口，並且相應 IO 口需要被設置為輸出模式；控制輸出 IO 的使能與關閉，可控制 PWM 的輸出與關閉，若想要完全關閉 PWM，就必須關閉 TMB。

PWM 波形是由 TBR、TBC0、TBC1、TBC0 組合使用產生的，且工作模式有 6 種不同的模式，因而每種模式的工作條件有所差異，以下針對 6 中模式做出一一介紹。兩個獨立的 PWM：PWMO0 和 PWMO1，使用條件與控制是一樣的，所以不做區分介紹。

PWMA 模式

PWMA 模式是 16 位元元 PWM，TBR 計數值與 TBC1 做比較，PWM 波形週期由 TBC0 控制。

PWM = 1，當 TBR [15:0] >= TBC1 [15:0]；

PWM = 0，當 TBR [15:0] < TBC1 [15:0]；

PWM 的週期：

PWM Period = TBR[15:0]*TMCD / HS_CK(或 LS_CK)；

PWM 的占空比：

PWM Duty= TBC1/(TBR[15:0]+1)

PWM Duty Cycle= (PWM Duty) *TMCD / HS_CK(或 LS_CK)；

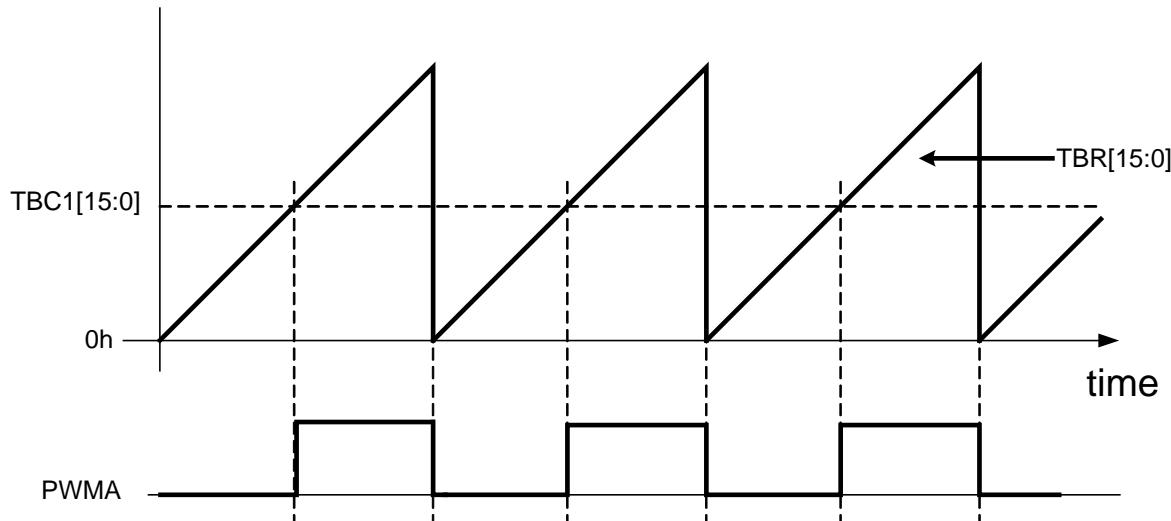


圖 10-6 PWM 模式 A 波形示意圖及計數波形示意圖

PWMB 模式

PWMB 模式是 16 位元元 PWM，TBR 計數值與 TBC2 做比較，PWM 波形週期值由 TBC0 控制。

PWM = 1，當 TBR [15:0] >= TBC2 [15:0]；

PWM = 0，當 TBR [15:0] < TBC2 [15:0]；

PWM 的週期：

$$\text{PWM Period} = \text{TBR}[15:0] * \text{TMCD} / \text{HS_CK(或 LS_CK)} ;$$

PWM 的占空比：

$$\text{PWM Duty} = \text{TBC2} / (\text{TBR}[15:0] + 1)$$

$$\text{PWM Duty Cycle} = (\text{PWM Duty}) * \text{TMCD} / \text{HS_CK(或 LS_CK)} ;$$

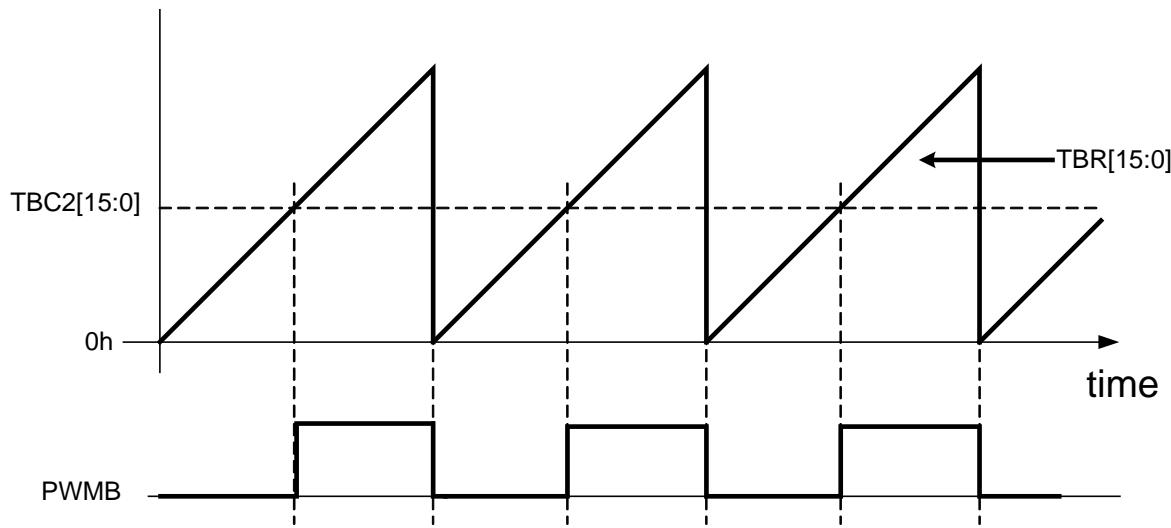


圖 10-7 PWM 模式 B 波形圖及計數波形示意圖

PWMC 模式

PWMC 是一個 8 位元 PWM，TBR 計數值與 TBC1 [7:0] 做比較，在週期值 TBC0 範圍內，出現多個 PWM 波形。

PWM 輸出狀態控制條件：

PWM = 1，當 TBR [7:0] \geq TBC1 [7:0]。

PWM = 0，當 TBR [7:0] $<$ TBC1 [7:0]。

PWM 的週期：

PWM Period = TBR[7:0]*TMCD / HS_CK(或 LS_CK)；

PWM 的占空比：

PWM Duty = TBC1[7:0]/(TBR[7:0]+1)

PWM Duty Cycle = (PWM Duty) * TMCD / HS_CK(或 LS_CK)；

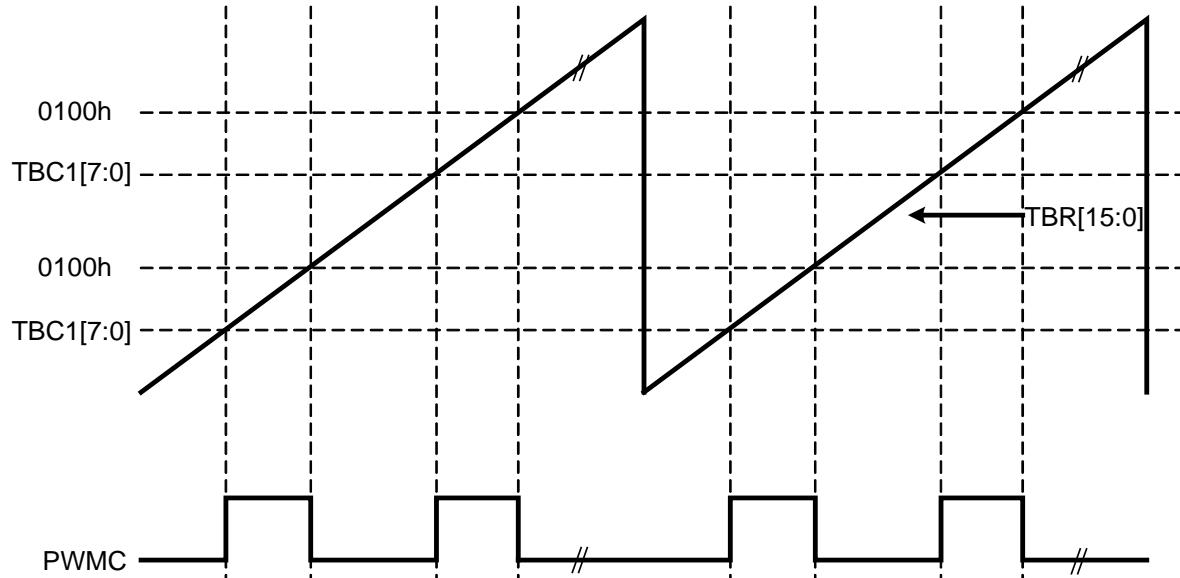


圖 10-8 PWM 模式 C 波形圖及計數波形示意圖

PWMD 模式

PWMD 是一個 8 位元 PWM，TBR 計數值與 TBC2 [7:0] 做比較，在週期值 TBC0 範圍內，出現多個 PWM 波形。

PWM 輸出狀態控制條件：

PWM = 1，當 TBR [15:8] >= TBC2 [7:0]；

PWM = 0，當 TBR[15:8] < TBC2[7:0]；

PWM 的週期：

PWM Period = TBR[15:8]*TMCD / HS_CK(或 LS_CK)；

PWM 的占空比：

PWM Duty = TBC2[7:0]/(TBR[15:8]+1)

PWM Duty Cycle = (PWM Duty) *TMCD / HS_CK(或 LS_CK)；

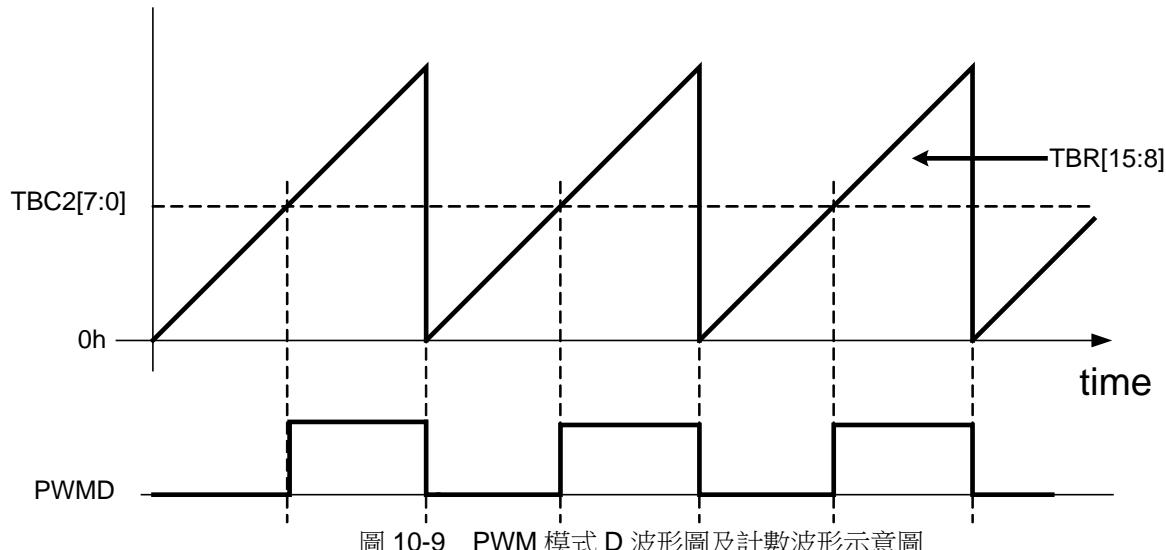


圖 10-9 PWM 模式 D 波形圖及計數波形示意圖

PWMF 模式

PWMF 是一個 16 位元 PWM，TBR 計數值與 TBC1 及 TBC2 做比較，且 TBC2 的值必須大於 TBC1 的值，TBR 不斷遞增至溢出。

PWM 輸出狀態控制條件：

$PWM = 1$ ，當 $TBC1[15:0] \leq TBR[15:0] \leq TBC2[15:0]$ ；
 $PWM = 0$ ，當 $TBR[15:0] > TBC2[15:0]$ 或 $TBR[15:0] \leq TBC1[15:0]$ ；
 $PWM=1$ 的時間為 $t = t_{clock} \times (TBC2 - TBC1)$ ；

PWM 的週期值：

$PWM\ Period = TBR[15:0] * TMCD / HS_CK(或 LS_CK)$ ；

PWM 的占空比：

$PWM\ Duty = (TBC2 - TBC1) / (TBR[15:0] + 1)$
 $PWM\ Duty\ Cycle = (PWM\ Duty) * TMCD / HS_CK(或 LS_CK)$ ；

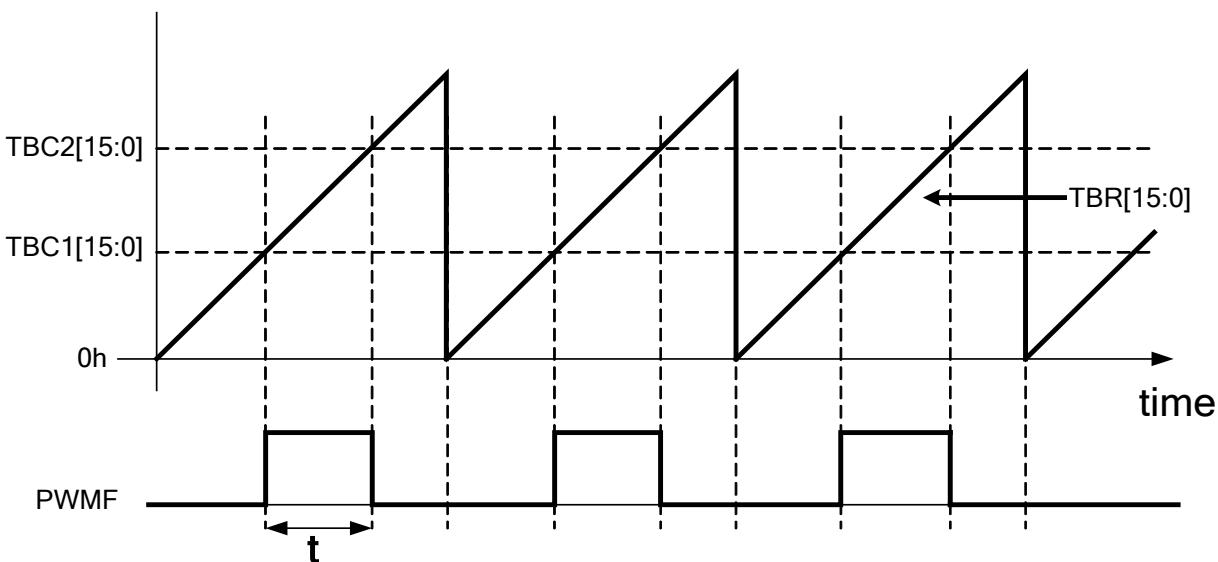


圖 10-10 PWM 模式 F 波形圖及計數波形示意圖

PWMG 模式

PWMG 是 16 位元元的 PWM 模式，且輸出波形占空比為 50%，即是輸出 PFD 波形。TBR 計數值不與 TBC1/TBC2 做比較，且輸出波形的週期值至於 TBC0 相關。

PWM 的週期值：

PWM Period = $TBC0[15:0] * TMCD / HS_CK(或 LS_CK)$ ；

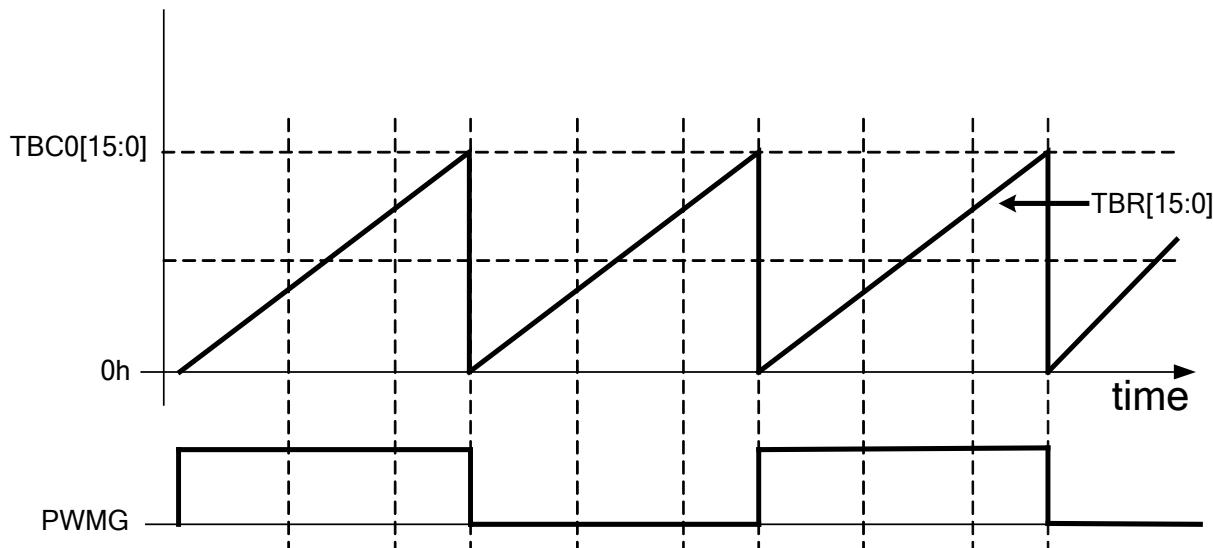


圖 10-11 PWM 模式 G 波形圖及計數波形示意圖

10.2 暫存器位址

TMR Register Address	31	24	23	16	15	8	7	0
TMR Base Address + 0X00(0X40C04)	MASK1		REG1		MASK0		REG0	
TMR Base Address + 0X08(0X40C08)	-		REG2		TBCR		TBCR	
TMR Base Address + 0X0C(0X40C0C)	-		-		TBC0		TBC0	
TMR Base Address + 0X10(0X40C10)	TBC2		TBC2		TBC1		TBC1	

- 保留

10.3 暫存器功能

10.3.1 Timer B 暫存器 TMBCR0

TMA Base Address + 0X04 (0X40C04)					
Symbol	TMBCR0(TMB Control Register 0)				
Bit	[31:24]	[23]	[22:20]	[19]	[18:16]
名稱	MASK	O1PMR	O1MD	O0PMR	O0MD
RW	R0W-0			RW-0	
Bit	[15:8]	[7:6]	[5]	[4]	[3:2]
名稱	MASK	-	TBEN	TBRST	TBM
RW	R0W-0	-			RW-0

位元	名稱	描述
Bit[23]	O1PMR	PWM1 波形輸出相位控制
		0 反相輸出
		1 正常輸出
Bit[21-20]	O1MD	PWM1 工作模式選擇
		0 PWMA
		1 PWMB
		2 PWMC
		3 PWMD
		4 RSV
		5 PWMF
		6 PWMG
Bit[19]	O0PMR	PWM0 波形輸出相位控制
		0 反相輸出
		1 正常輸出
Bit[18-16]	O0MD	PWM0 工作模式選擇
		0 PWMA
		1 PWMB
		2 PWMC
		3 PWMD
		4 RSV
		5 PWMF
		6 PWMG
		7 PWMG

位元	名稱	描述		
Bit[5]	TBEN	Timer B 開啟控制		
		<table border="1"> <tr> <td>0</td><td>關閉</td></tr> <tr> <td>1</td><td>開啟</td></tr> </table>	0	關閉
0	關閉			
1	開啟			
Bit[4]	TBRST	Timer B 復位		
		<table border="1"> <tr> <td>0</td><td>正常</td></tr> <tr> <td>1</td><td>清零 Timer B 計數寄存器 TBR , 完成後自動置為 0</td></tr> </table>	0	正常
0	正常			
1	清零 Timer B 計數寄存器 TBR , 完成後自動置為 0			
Bit[3~2]	TBM	Timer B 計數模式選擇		
		00 16-bit 遞增計數器，鋸齒波類型的計數方式，以步長為 1 遞增至最大值 TBC0		
		01 16-bit 遞增遞減計數器，三角波類型的計數方式，以步長為 1 先遞增至最大值 TBC0 後再由最大值遞減至 0		
		10 2 個獨立 8Bit 遞增計數器 TBR[15:8]及 TBR[7:0]，鋸齒波類型的計數方式，兩個計數器以步長為 1，同時遞增計數至最大值 TBC0[15:8]及 TBC0[7:0]		
		11 2 個 8Bit 遞增計數器 TBR[15:8]及 TBR[7:0]，步長為 1 的鋸齒波類型計數方式，當計數器 TBR[7:0]遞增計數溢出後，計數器 TBR[15:8]才自動加 1，且 TBR[7:0]又從 0 開始遞增計數		
Bit[1~0]	TBEBS	Timer B 計數觸發模式選擇		
		00 1 總是啟用，連續計數方式		
		01 CMPO 多功能比較器輸出高電位觸發		
		10 OPOD 運算放大器輸出高電位觸發		
		11 CPI1 Timer C 的輸出 CPI1 高電位觸發		

10.3.2 Timer B 暫存器 TMBCR1

TMR Base Address + 0X08 (0X40C08)							
Symbol	TMBCR1(TMB Control Register 1)						
Bit	[31:22]	[21]	[20]	[19]	[18]	[17]	[16]
名稱	-	PWMFF	-	PWMDF	PWMCF	PWMBF	PWMAF
RW	-	R-X	-			R-X	
Bit	[15:0]						
名稱	Timer B 計數值						
RW	R-X						

位元	名稱	描述		
Bit[23-16]	PWM Flag	PWM A/B/C/D/F 工作模式狀態旗標		
		<table border="1"> <tr> <td>0</td><td>正常</td></tr> <tr> <td>1</td><td>啟用</td></tr> </table>	0	正常
0	正常			
1	啟用			
Bit[15-0]	TMBC	Timer B 16-bit 計數值		

10.3.3 Timer B 暫存器 TMBCOD

TMR Base Address + 0X08 (0X40C0C)	
Symbol	TMBCOD(TMB Counter overflow condition Register)
Bit	[31:16]
名稱	-
RW	-
Bit	[15:0]
名稱	TBC0:Timer B Overflow Condition
RW	RW-0xFFFF

位元	名稱	描述
Bit[15-0]	TBC0	Timer B 計數溢出門限值

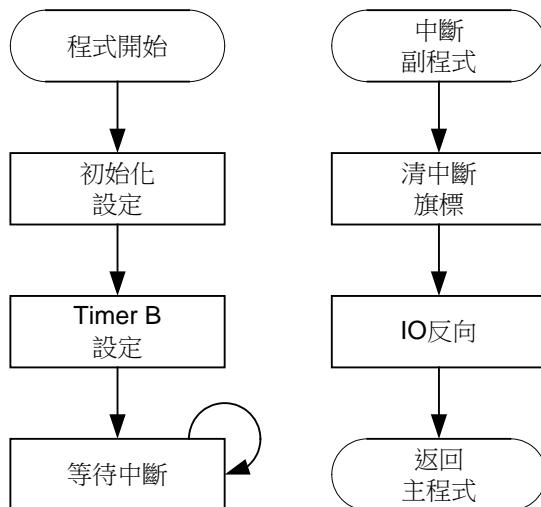
10.3.4 Timer B 暫存器 PWMDOD

TMR Base Address + 0X08 (0X40C10)	
Symbol	PWMDOD(PWM counter overflow condition Control Register)
Bit	[31:16]
名稱	TBC2: PWM1 占空比計數溢出值
RW	RW-0xFFFF
Bit	[15:0]
名稱	TBC1: PWM0 占空比計數溢出值
RW	RW-0xFFFF

位元	名稱	描述
Bit[31-16]	TBC2	PWM1 占空比計數溢出值
Bit[15-00]	TBC1	PWM0 占空比計數溢出值

10.4 範常式式流程

- (1)採用紜康 C 函式庫，DrvTMBC_Clk_Source(X,Y) 可選擇 Timer B 時脈來源，X 為高低速選擇。
 $X=0$ 為 HS_CK，Y 為除頻選擇，若 $Y=3$ 代表 Timer B IP 時脈 除頻 8。
- (2)函式 DrvTMB_Open(E_TMB_MODE0,E_TMB_NORMAL,0xFFFF)，代表 TimerB IP 設定，包含 Mode 選擇，
致能 Timer B 以及 Timer B counter 計數設定。
- (3)紜康 C 函式庫 DrvTIMER_EnableInt(E_TMB)，表示將 Timer B 中斷致能。
- (4)紜康 C 函式庫 DrvTIMER_ClearIntFlag(E_TMB)，表示將 TimerB 中斷旗標清除。
- (5)Timer A/B/C 與 WDT 隸屬於 HW1 中斷，使用格式為 void HW1_ISR(void)。



10.5 範常式式功能

- (1)使用 TimerB 中斷。
- (2)每進一次 TimerB 中斷，會將 IO 反向輸出。
例如進中斷前 0X5，進中斷後變 0XA。

10.6 範常式說明

00		
01	#include "HY16F18X.h"	
02		
03	unsigned int i;	
04		
05	int main(void)	
06	{	
07	i=0X05;	
08	DrvGPIO_Open(E_PT2,0X0F,E_IO_OUTPUT);	//PT2_0~3 Set Output
09	DrvGPIO_SetPortBits(E_PT2,i);	//PT2 Output i=0X05
10		
11	DrvTMBC_Clk_Source(0,3);	//Timer B Prescaler 1
12		//0: HS_CK Clock Source.
13		//3: clock divider.:8
14		
15	DrvTMB_Open(E_TMB_MODE0,E_TMB_NORMAL,0xFFFF);	//Timer B overflow 0xFFFF
16	DrvTIMER_ClearIntFlag(E_TMB);	//Clear TMB interrupt flag
17	DrvTIMER_EnableInt(E_TMB);	//Timer B interrupt enable
18		
19	SYS_EnableGIE(7);	//Enable GIE
20		
21	while(1);	//Wait for Interrupt
22	}	
23		
24	void HW1_ISR(void)	
25	{	
26	DrvTIMER_ClearIntFlag(E_TMB);	//Clear TMB interrupt flag
27	i=i^0XF;	//i XOR 0XF
28	DrvGPIO_SetPortBits(E_PT2,i);	//PT2 Output i=0XA~0X05
29	}	
30		

11.定時器 Timer C

11.1 整體總說明

定時器 C 是設計用來做捕捉的功能，可用於執行頻率測量、事件計數、間隔時間測量等功能，可以在計數溢出時產生中斷信號。在使用時需要配合 TMB 計數暫存器一起使用。

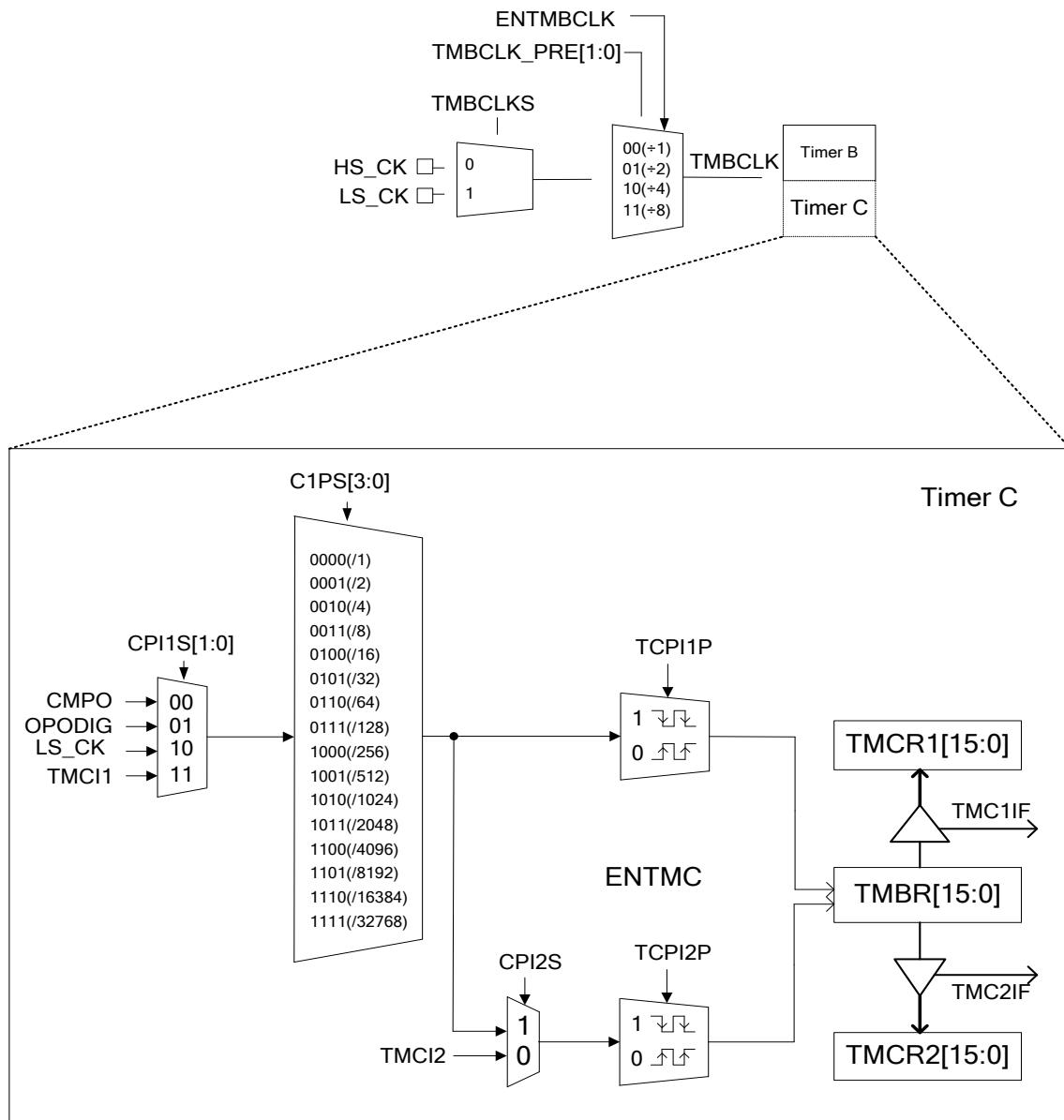


圖 11-1 TC 功能方框圖

TMC 時脈源

TMC 的時脈源與 TMB 一致，都是由 HS_CK 或 LS_CK 經過除頻器產生時脈源 TMBCLK。使能控制位 TCEN[0] 的設置，可開啟或關閉 TMC 捕捉功能。

TMC 捕捉計數值

TMC 的捕捉計數值是由 TMB 來計數器 TBR 完成，當 TMC 的捕捉到輸入信號的第一個觸發沿，TBR 就開始計數，當 TMC 捕捉到輸入信號的第二個觸發沿，TBR 停止計數，捕捉事件完成後硬體上會自動將 TBR 的值寫入 TMCR1 與 TMCR2，並產生中斷請求旗標 TMC1IF 及 TMC2IF。用戶可程式讀取 TCR1/TCR2 的值。

捕捉比較器 1

捕捉比較器 1 有 4 個捕捉信號輸入源，透過選擇器 CPI1S[1:0] 設置輸入信號源；且輸入信號還需經過除頻器 C1PS[3:0]，除頻器的設置對輸入信號源進行除頻，可以將輸入信號減慢，這樣可以測量頻率較快的輸入信號。同時可以透過控制器 TCPI1P 的設置，設置捕捉信號的觸發沿為上升沿或下降沿捕捉。當捕捉事件完成後，可產生中斷信號，中斷旗標 TMC1IF 被置`<1>`。

捕捉比較器 1 的捕捉信號輸入源：

輸入信號源符號	功能描述
CMP0	比較器的輸出狀態
OPOD	運算放大器的輸出狀態
LS_CK	晶片低速頻率源
TCI1	從 IO 口輸入

捕捉比較器 1 輸入 IO

序號	TCI1	TCI2	序號	TCI1	TCI2
000	PT1.0	PT1.1	100	PT2.0	PT2.1
001	PT1.2	PT1.3	101	PT2.2	PT2.3
010	PT1.4	PT1.5	110	PT2.4	PT2.5
011	PT1.6	PT1.7	111	PT2.6	PT2.7

捕捉比較器 1 的操作：

- ◆ 選擇 TMC 工作時脈源 TMBCLK；
- ◆ 設置捕捉信號輸入源及輸入信號源除頻值，即設置 CPI1S[1:0]、C1PS[3:0]的值；
- ◆ 設置捕捉信號觸發沿，即是設置 TCPI1P 的值；
- ◆ 若是選擇 TCI1 作為捕捉信號輸入源，需要設置輸入 IO，配置對應 IO 作為輸入模式；
- ◆ 若使用中斷功能，需使能 TMC0IE=1;並使能全局中斷 GIE=1；
- ◆ 啟動 TMC 功能，使能 TCEN。

捕捉比較器 2

捕捉比較器 2 具有 2 個捕捉信號輸入源，透過選擇器 CPI2S[0] 設置設置不同輸入信號源，但其輸入信號無須經過除頻器。透過控制器 TCPI2P 可設置信號捕捉觸發沿為上升沿或下降沿捕捉。當捕捉事件完成後，可產生中斷信號，中斷旗標 TMC2IF 被置`<1>`。

捕捉比較器 2 的捕捉信號輸入為：

- ◆ 從 IO 口輸入；
- ◆ 與捕捉比較器 1 的輸入源一致；

捕捉比較器 2 的操作：

- ◆ 選擇 TMC 工作時脈源 TMBCLK；

- ◆ 設置捕捉信號輸入源，即設置 CPI2S[0]的值；
- ◆ 設置捕捉信號觸發沿，即是設置 TCPI2P 的值；
- ◆ 若是選擇 TCI2 作為捕捉信號輸入源，需要設置輸入 IO，配置對應 IO 作為輸入模式；
- ◆ 若使用中斷功能，需使能 TMC1IE=1；且使能全局中斷 GIE=1；
- ◆ 啟動 TMC 功能，使能 TCEN。

11.2 暫存器位址

TMR Register Address	31	24	23	16	15	8	7	0
TMR Base Address + 0X14(0X40C14)	MASK1		REG1		MASK0		REG0	
TMR Base Address + 0X18 (0X40C18)	TCR2		TCR2		TCR1		TCR1	

11.3 暫存器功能

11.3.1 Timer C 暫存器 TMCCR0

TMR Base Address + 0X14 (0X40C14)						
Symbol	TMCCR0(TMC Control Register 0)					
Bit	[31:24]	[23]	[22]	[21:20]	[19:16]	
名稱	MASK	-	CPSS	CPI1S	CP1PS	
RW	R0W-0	-	RW-0	RW-0	RW-0	
Bit	[15:08]	[7:3]			[2]	[1]
名稱	MASK	-			TCPI2P	TCPI1P
RW	R0W-0	-			TCEN	RW-0

位元	名稱	描述
Bit[22]	CPI2S	Capture 1 輸入信號源選擇
		0 TC2 來自 IO 口的輸入
		1 與 capture 1 一樣的輸入源
Bit[21~20]	CPI1S	Capture 0 輸入信號源選擇
		00 CMPO 比較器輸出
		01 OPOD 運算放大器輸出
		10 低頻時脈 LS_CK
		11 TC1 來自 IO 口的輸入
Bit[19~16]	C1PS	Capture1 輸入信號的除頻器設置
		0000 CPI1 frequency/1
		0001 CPI1 frequency/2
		0010 CPI1 frequency/4
		0011 CPI1 frequency/8
		0100 CPI1 frequency/16
		0101 CPI1 frequency/32
		0110 CPI1 frequency/64
		0111 CPI1 frequency/128
		1000 CPI1 frequency/256
		1001 CPI1 frequency/512
		1010 CPI1 frequency/1024

		1011	CPI1 frequency/2048
		1100	CPI1 frequency/4096
		1101	CPI1 frequency/8192
		1110	CPI1 frequency/16384
		1111	CPI1 frequency/32768

位元	名稱	描述	
Bit[02]	TCPI2P	Capture2 觸發沿設置	
		0 上升沿觸發	
Bit[01]	TCP1P	Capture1 觸發沿設置	
		0 上升沿觸發	
Bit[00]	TCEN	Timer C 開啟控制	
		0 關閉 (但不清零 TCR0 及 TCR1)	
		1 開啟	

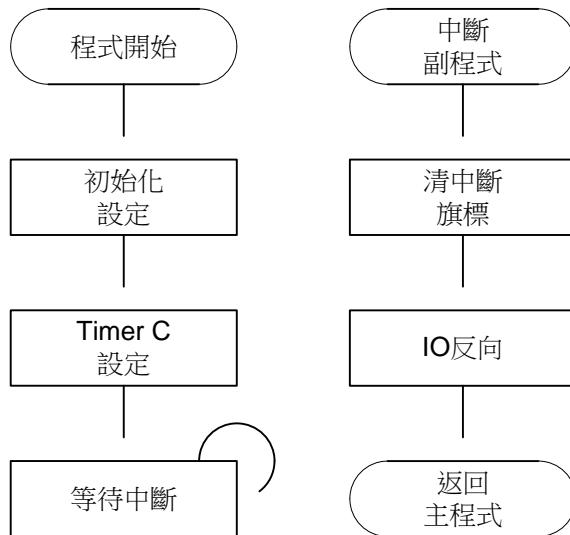
11.3.2 Timer C 暫存器 TMCCR1

Symbol	TMR Base Address + 0X18 (0X40C18)	
Bit	TMCCR1(TMC Control Register 1)	
名稱	TCR2	
RW	R-X	
Bit	[15:00]	
名稱	TCR1	
RW	R-X	

位元	名稱	描述
Bit[31-16]	TCR2	Capture2 頻率捕捉計數器
Bit[15-00]	TCR1	Capture1 頻率捕捉計數器

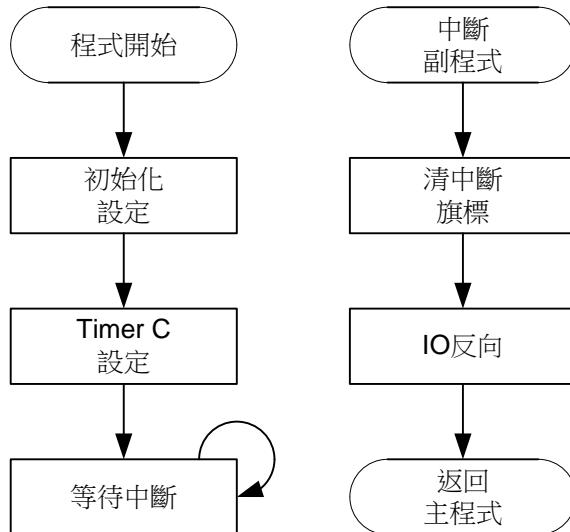
11.4 範常式式流程

- (1)DrvTMB_Open()會將 Timer B 啟能。
- (2)DrvCapture1_Open(),設定 Timer C0。
- (3)DrvCapture2_Open(),設定 Timer C1。



11.5 範常式式功能

- (1) 使用 Timer C 中斷。
- (2) 每進一次 Timer C 中斷，會將 IO 反向輸出。
例如進中斷前 0X5，進中斷後變 0XA。



11.6 範常式式說明

00		
01	#include "HY16F18X.h"	
02		
03	unsigned int i;	
04		
05	int main(void)	
06	{	
07	i=0X05;	

HY16F18 系列用戶手冊
21-位 ENOB ΣΔADC, 32-位 MCU & 64 KB Flash



08	DrvGPIO_Open(E_PT2,0X0F,E_IO_OUTPUT);	//PT2_0~3 Set Output
09	DrvGPIO_SetPortBits(E_PT2,i);	//PT2 Output i=0X05
10		
11	DrvTMBC_Clk_Source(0,0);	//Timer B Prescaler 1
12		//0: HS_CK,clock source.
13		//0: clock divider.÷1
14		
15	DrvTMB_Open(E_TMB_MODE0,E_TMB_NORMAL,0xFFFF);	//Timer B overflow 0xFFFF
16		
17	DrvCapture1_Open(2,14,1);	//TimerC0 use as Capture 1
18		//input source selection
19		//2:LS_CK
20		//14:÷16384 1:Positive-edge trigger
21		
22	DrvCapture2_Open(1,1);	//TimerC1 use as Capture 2
23		// input source selection
24		
25	DrvTIMER_ClearIntFlag(E_TMC0);	//Clear TMC0 interrupt flag
26	DrvTIMER_ClearIntFlag(E_TMC1);	//Clear TMC1 interrupt flag
27		
28	DrvTIMER_EnableInt(E_TMC0);	//TimerC0 interrupt enable
29	DrvTIMER_EnableInt(E_TMC1);	//TimerC1 interrupt enable
30		
31	SYS_EnableGIE(7);	//Enable GIE
32		
33	while(1);	//Wait for Interrupt
34	}	
37		
38	void HW1_ISR(void)	
39	{	
40	DrvTIMER_ClearIntFlag(E_TMC0);	//Clear TMC0 interrupt flag
41	DrvTIMER_ClearIntFlag(E_TMC1);	//Clear TMC1 interrupt flag
42	i=i^0XF;	//i XOR 0XF
43	DrvGPIO_SetPortBits(E_PT2,i);	//PT2 Output i=0XA~0X05
44	}	
45		

12.GPIO PT1 管理

12.1 整體總說明

PT1 具有 8 個 IO 引腳，可以作為通用的普通 IO 口，亦可複用為捕捉比較器、SPI、IIC、比較器、PWM 及外部中斷等功能模塊的輸入或輸出 IO 口。針對不同的複用，需要做不同的設置。

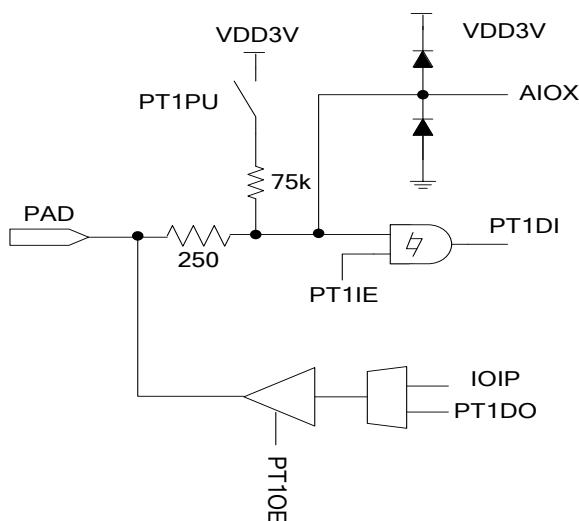


圖 12-1 PT1 功能方框圖

PT1 具有輸入、輸出、內部上拉電阻及作為外部中斷輸入口的功能，且分別有不同的控制器來設置。

內部上拉電阻

控制器 PT1PU[7:0]可設置每個 IO 口的內部上拉電阻的開啟與關閉，每一位元對應一個 IO 口引腳。當 IO 口對應位被置<1>，則開啟內部 75k 上拉電阻，若被置<0>，則關閉內部上拉電阻。IO 口作為輸入模式時，若外部沒有上拉電阻，必須開啟內部上拉電阻，尤其在低功耗模式時，可以防止漏電，而增加功耗。作為類比信號輸入口時，不用開啟內部上拉電阻。

輸出模式

控制器 PT1OE[7:0]可設置每個 IO 口輸出模式的開啟與關閉，每一位元對應一個 IO 口引腳。當 IO 口對應位被置<1>，則開啟對應 IO 口輸出模式；若被置<0>，則關閉輸出模式。透過控制器 PT1DO[7:0]來控制對應 IO 口引腳的輸出狀態為 1 或 0。在低功耗模式下，若 IO 必須開啟輸出模式，可根據外圍電路來設置輸出狀態，來減低晶片功耗。此模式下不能開啟 IO 內部上拉電阻，且不能同時開啟輸入、輸出模式，因而在開啟輸出模式時，需要關閉 IO 口的輸入模式。

輸入模式

控制器 PT1IE[7:0]可設置每個 IO 口引腳輸入模式的開啟與關閉，每一位元對應一個 IO 口引腳。當控制器對應位元被置<1>，則開啟對應 IO 口引腳的輸入模式；若被置<0>，則關閉輸入模式。透過控制器 PT1DI[7:0]可讀取當前對應 IO 引腳的輸入狀態為 1 或 0。當 IO 被設置為輸入模式，若晶片沒有接入外部上拉電阻，則必須開啟晶片內部上拉電阻，不能允許 IO 引腳出現浮接狀態，以免造成晶片產生漏電現象；特別是在低功耗模式下，建議將 IO 引腳設置為輸入模式。作為類比信號輸入口時，不用設置對應 IO 引腳為輸入模式。在開啟輸入模式前，需要關閉對應 IO 引腳的輸出模式。

外部中斷輸入

PT1 具有的 8 個 IO 引腳都可複用為外部中斷輸入引腳。此模式需要將 IO 口設置為輸入模式且使能內部上拉電阻。需要透過控制器 PT1XITT[2:0]設置外部中斷觸發沿，並使能控制位 PT1ITD[0]，以使能中斷觸發沿有效。透過控制器 PT1XIE[7:0]使能對應 IO 引腳的中斷響應功能，當外部中斷信號產生時，對應 IO 引腳的中斷請求旗標被置 1。在使能全局中斷 GIE 及開啟 IO 外部中斷功能的條件下，晶片就馬上暫停當前程式轉去執行 IO 外部中斷程式。

12.2 暫存器位址

GPIO Register Address	31	24	23	16	15	8	7	0
GPIO base address + 0x00(0X40800)	MASK1	PT1PU	MASK0	PT1OE				
GPIO base address + 0x04 (0X40804)	MASK3	PT1IE	MASK2	PT1DO				
GPIO base address + 0x08(0X40808)	-	-	-	PT1DI				
GPIO base address + 0x0C (0X4080C)	PT1ITD	PT1ITT	PT1ITT	PT1ITT				

-保留

12.3 暫存器功能

12.3.1 PT1 暫存器 PT1CR0

GPIO Base Address + 0X00 (0X40800)									
Symbol	PT1CR0 (PT1 Control Register 0)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名稱	MASK	PT1PU7	PT1PU6	PT1PU5	PT1PU4	PT1PU3	PT1PU2	PT1PU1	PT1PU0
RW	R0W-0								RW-0
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名稱	MASK	PT1OE7	PT1OE6	PT1OE5	PT1OE4	PT1OE3	PT1OE2	PT1OE1	PT1OE0
RW	R0W-0								RW-0

位元	名稱	描述
Bit[23~16]	PT1PU	Port 1 內部上拉控制
		0 關閉內部上拉
		1 開啟內部上拉
Bit[07~00]	PT1OE	Port 1 PAD 輸出模式開啟控制
		0 關閉輸出模式
		1 開啟輸出模式

12.3.2 PT1 暫存器 PT1CR1

GPIO Base Address + 0X04 (0X40804)									
Symbol	PT1CR1 (PT1 Control Register 1)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名稱	MASK	PT1IE7	PT1IE6	PT1IE5	PT1IE4	PT1IE3	PT1IE2	PT1IE1	PT1IE0
RW	R0W-0						RW-0		
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名稱	MASK	PT1DO7	PT1DO6	PT1DO5	PT1DO4	PT1DO3	PT1DO2	PT1DO1	PT1DO0
RW	R0W-0						RW-0		

位元	名稱	描述							
Bit[23~16]	PT1IE	Port 1 PAD 輸入模式控制							
		0	關閉輸入模式						
		1	開啟輸入模式						
Bit[07~00]	PT1DO	Port 1 PAD 輸出狀態值							
		0	置 0						
		1	置 1						

12.3.3 PT1 暫存器 PT1CR2

GPIO Base Address + 0X08 (0X40808)									
Symbol	PT1CR2(PT1 Control Register 2)								
Bit	[31:16]								
名稱	-								
RW	-								
Bit	[15:8]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名稱	-	PT1DI7	PT1DI6	PT1DI5	PT1DI4	PT1DI3	PT1DI2	PT1DI1	PT1DI0
RW	-					R-0			

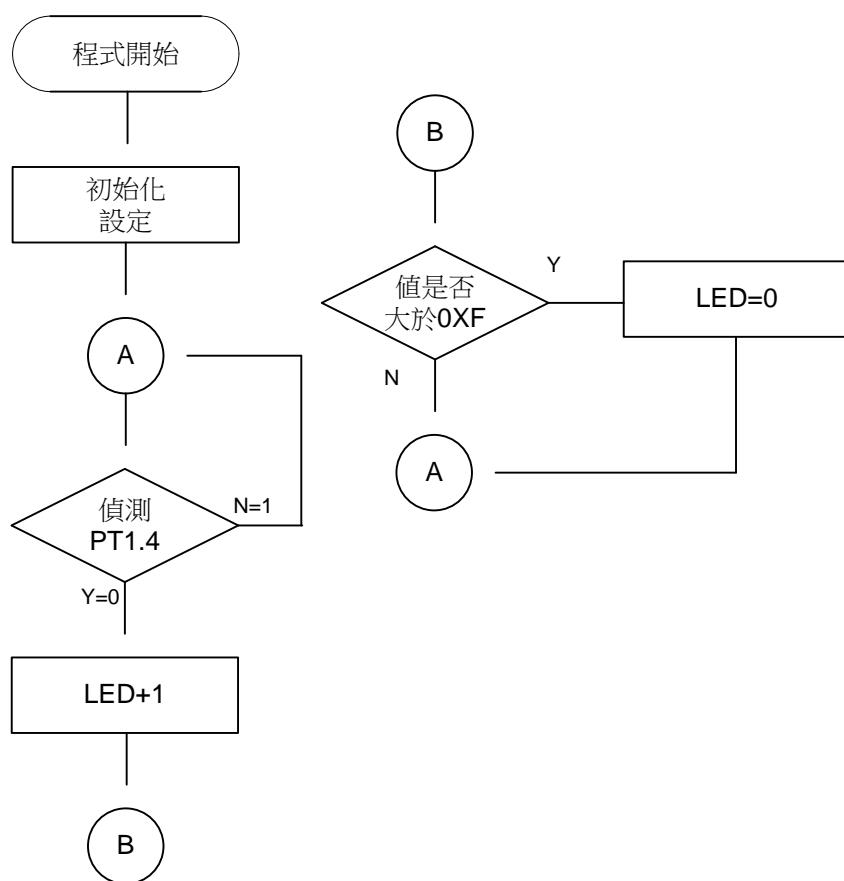
位元	名稱	描述							
Bit[7~0]	PT1DI	Port1 PAD 輸入狀態值							
		0	輸入低電位						
		1	輸入高電位						

12.3.4 PT1 暫存器 PT1CR3

GPIO Base Address + 0X0C (0X4080C)									
Symbol	PT1CR3 (PT1 Control Register 3)								
Bit	[31:24]	[23:21]	[21:18]	[17:15]					
名稱	PT1ITD	PT17ITT	PT16ITT	PT15ITT					
RW			RW-0						
Bit	[15]	[14:12]	[11:9]	[8:6]	[5:3]	[2:0]			
名稱	<<	PT14ITT	PT13ITT	PT12ITT	PT11ITT	PT10ITT			
RW	<<			RW-0					

位元	名稱	描述
Bit[31~24]	PT1ITD	PT1 中斷觸發沿開啟控制
		0 關閉
		1 開啟
Bit[23~00]	PT1XITT	Port 1.X 中斷觸發沿模式選擇. X 的值為 0~7
		000 關閉 GPIO 中斷觸發，不能響應中斷
		001 上升沿觸發
		010 下降沿觸發
		011 電位變化觸發
		100 低電位觸發
		101 高電位觸發
		110 低電位觸發
		111 高電位觸發

12.4 範常式式流程



12.5 範常式式功能

- (1)每按 1 次 PT1.4，LED 加 1 顯示。
(2)若 LED 加到 0XF，則歸 0。回到每按 1 次 PT1.4，LED 加 1 顯示。

12.6 範例程式說明

00	
01	#include "HY16F188.h"
02	
03	void Delay (unsigned int num);
04	
05	int main(void)
06	{
07	unsigned int i=0,j=0;
08	DrvGPIO_Open(E_PT1,0X10,E_IO_INPUT); //Set PT1_4 INPUT
09	DrvGPIO_Open(E_PT1,0X10,E_IO_PullHigh); //Enable PT1_4 pull hi R
11	DrvGPIO_Open(E_PT2,0X0F,E_IO_OUTPUT); //Set PT2_0~3 OUTPUT
12	
13	while(1)
14	{
15	i=DrvGPIO_GetBit(E_PT1,4); //Read PT1.4 high or low
16	if(i==0) //IF PT1.4 is low
17	{
18	DrvGPIO_SetPortBits(E_PT2, j++); //J++
19	if(j>0XF)j=0X00; //IF J>0XF J=0
20	}
21	}
22	Delay(0X8000);
23	}
24	
25	void Delay(unsigned int num)
26	{
27	int a;for(a=0;a<=num;a++); //Delay loop
28	}
29	

13. GPIO PT2 管理

13.1 整體總說明

PT2 具有 8 個 IO 引腳，可以作為通用的普通 IO 口，亦可複用為捕捉比較器、SPI、IIC、PWM、外部晶震輸入及外部中斷輸入等功能模塊的輸入或輸出 IO 口。針對不同的複用，需要做不同的設置。

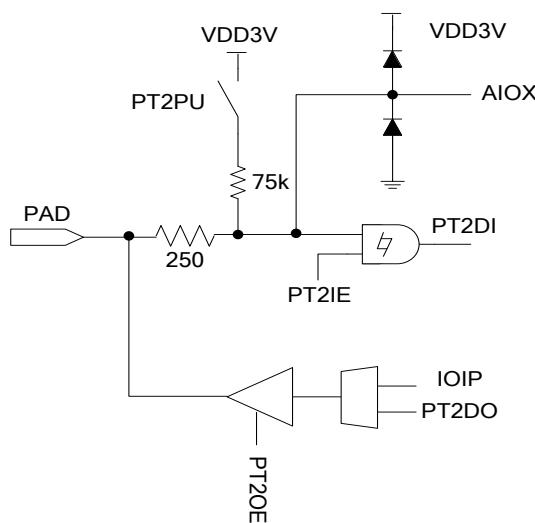


圖 13-1 PT2 功能方框圖

PT2 具有輸入、輸出、內部上拉電阻及作為外部中斷輸入口的功能，且分別有不同的控制器來設置。

內部上拉電阻

控制器 PT2PU[7:0]可設置每個 IO 口的內部上拉電阻的開啟與關閉，每一位元對應一個 IO 口引腳。當 IO 口對應位被置`<1>`，則開啟內部 75k 上拉電阻，若被置`<0>`，則關閉內部上拉電阻。IO 口作為輸入模式時，若外部沒有上拉電阻，必須開啟內部上拉電阻，尤其在低功耗模式時，可以防止漏電，而增加功耗。作為類比信號輸入口及外部晶震輸入引腳時，不用開啟內部上拉電阻。

注意：PT2.4~PT2.7 作為外部晶震輸入引腳，不可開啟內部上拉電阻，否則晶震不能正常起震。

輸出模式

控制器 PT2OE[7:0]可設置每個 IO 口輸出模式的開啟與關閉，每一位元對應一個 IO 口引腳。當 IO 口對應位被置`<1>`，則開啟對應 IO 口輸出模式；若被置`<0>`，則關閉輸出模式。透過控制位 PT2DO[7:0]來控制對應 IO 口引腳的輸出狀態為 1 或 0。在低功耗模式下，若 IO 必須開啟輸出模式，可根據外圍電路來設置輸出狀態，來減低晶片功耗。此模式下不能開啟 IO 內部上拉電阻，且不能同時開啟輸入、輸出模式，因而在開啟輸出模式時，需要關閉 IO 口的輸入模式。

注意：PT2.4~PT2.7 作為外部晶震輸入引腳時，必須關閉輸出模式。

輸入模式

控制器 PT2IE[7:0]可設置每個 IO 口引腳輸入模式的開啟與關閉，每一位元對應一個 IO 口引腳。當控制器對應位元被置`<1>`，則開啟對應 IO 口引腳的輸入模式；若被置`<0>`，則關閉輸入模式。透過控制器 PT2DI[7:0]可讀取當前對應 IO 引腳的輸入狀態為 1 或 0。當 IO 被設置為輸入模式，若晶片沒有接入外部上拉電阻，則必須開啟晶片內部上拉電阻，不能允許 IO 引腳出現浮接狀態，以免造成晶片產生漏電現象；特別是在低功耗模式下，建議

將 IO 引腳設置為輸入模式。作為類比信號輸入口時，不用設置對應 IO 引腳為輸入模式。在開啟輸入模式前，需要關閉對應 IO 引腳的輸出模式。

外部中斷輸入

PT2 具有的 8 個 IO 引腳都可複用為外部中斷輸入引腳。此模式需要將 IO 口設置為輸入模式且使能內部上拉電阻。需要透過控制器 PT2XITT[2:0]設置外部中斷觸發沿，並使能控制位 PT2ITD[0]，以使能中斷觸發沿有效。透過控制器 PT2XIE[7:0]使能對應 IO 引腳的中斷響應功能，當外部中斷信號產生時，對應 IO 引腳的中斷請求旗標被置 1。在使能全局中斷 GIE 及開啟 IO 外部中斷功能的條件下，晶片就馬上暫停當前程式轉去執行 IO 外部中斷程式。

13.2 暫存器位址

GPIO Register Address	31	24	23	16	15	8	7	0
GPIO Base Address + 0x00(0X40810)	MASK1	PT2PU	MASK0	PT2OE				
GPIO Base Address + 0x04 (0X40814)	MASK3	PT2IE	MASK2	PT2DO				
GPIO Base Address + 0x08(0X40818)	-	-	-	PT2DI				
GPIO Base Address + 0x0C (0X4081C)	PT2ITD	PT2ITT	PT2ITT	PT2ITT				

-保留

13.3 暫存器功能

13.3.1 PT2 暫存器 PT2CR0

GPIO Base Address + 0X10 (0X40810)									
Symbol	PT2CR0 (PT2 Control Register 0)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名稱	MASK	PT2PU7	PT2PU6	PT2PU5	PT2PU4	PT2PU3	PT2PU2	PT2PU1	PT2PU0
RW	R0W-0	RW-0							
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名稱	MASK	PT2OE7	PT2OE6	PT2OE5	PT2OE4	PT2OE3	PT2OE2	PT2OE1	PT2OE0
RW	R0W-0	RW-0							

位元	名稱	描述
Bit[23~16]	PT2PU	Port 2 內部上拉控制
		0 關閉內部上拉
		1 開啟內部上拉
Bit[07~00]	PT2OE	Port 2 PAD 輸出模式開啟控制
		0 關閉輸出模式
		1 開啟輸出模式

13.3.2 PT2 暫存器 PT2CR1

GPIO Base Address + 0X14 (0X40814)									
Symbol	PT2CR1 (PT2 Control Register 1)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名稱	MASK	PT2IE7	PT2IE6	PT2IE5	PT2IE4	PT2IE3	PT2IE2	PT2IE1	PT2IE0
RW	R0W-0						RW-0		
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名稱	MASK	PT2DO7	PT2DO6	PT2DO5	PT2DO4	PT2DO3	PT2DO2	PT2DO1	PT2DO0
RW	R0W-0						RW-0		

位元	名稱	描述
Bit[23~16]	PT2IE	Port 2 PAD 輸入模式開啟控制
		0 關閉輸入模式
	1	開啟輸入模式
Bit[7~0]	PT2DO	Port 2 PAD 輸出狀態值
		0 置 0
	1	置 1

13.3.3 PT2 暫存器 PT2CR2

GPIO Base Address + 0X18 (0X40818)									
Symbol	PT2CR2 (PT2 Control Register 2)								
Bit	[31:16]								
名稱									
RW									
Bit	[15:8]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名稱	-	PT2DI[7]	PT2DI[6]	PT2DI[5]	PT2DI[4]	PT2DI[3]	PT2DI[2]	PT2DI[1]	PT2DI[0]
RW	-					R-0			

位元	名稱	描述
Bit[7~0]	PT2DI	Port2 PAD 輸入狀態值
		0 輸入低電位
		1 輸入高電位

13.3.4 PT2 暫存器 PT2CR3

GPIO Base Address + 0X1C (0X4081C)					
Symbol	PT2CR3 (PT2 Control Register 3)				
Bit	[31:24]	[23:21]	[20:18]	[17:15]	
名稱	PT2ITD	PT27ITT	PT26ITT	PT25ITT	
RW			RW-0		
Bit	[15]	[14:12]	[11:9]	[8:6]	[2:0]
名稱	<<	PT24ITT	PT23ITT	PT22ITT	PT21ITT
RW	<<			RW-0	

位元	名稱	描述
Bit[31~24]	PT2ITD	PT2 中斷觸發沿開啟控制
		0 關閉 1 開啟
Bit[23~0]	PT2XITT	Port 2.X 中斷觸犯沿模式選擇，X 的值為 0~7
		000 關閉 GPIO 中斷沿，不能響應中斷
		001 上升沿觸發
		010 下降沿觸發
		011 電位變化觸發
		100 低電位觸發
		101 高電位觸發
		110 低電位觸發
		111 高電位觸發

14.GPIO PT3 管理

14.1 整體總說明

PT3 具有 8 個 IO 引腳，可以作為通用的普通 IO 口，亦可複用為運算放大器、DAC 及模數轉換器 ADC 等功能模塊的輸入或輸出 IO 口。針對不同的複用，需要做不同的設置。

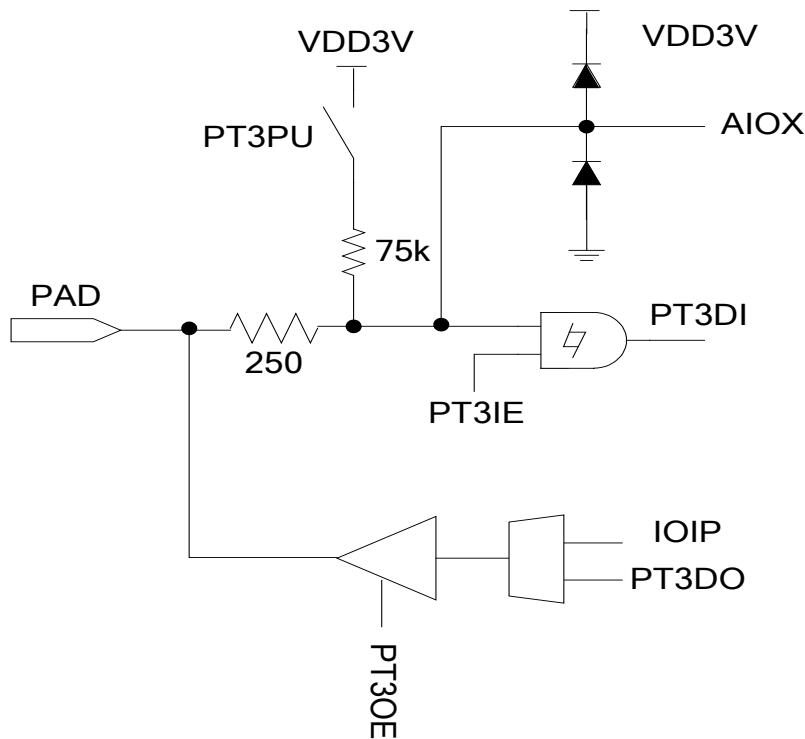


圖 14-1 PT3 功能方框圖

PT3 具有輸入、輸出、內部上拉電阻的功能，且分別有不同的控制器來設置。

內部上拉電阻

控制器 PT3PU[7:0]可設置每個 IO 口的內部上拉電阻的開啟與關閉，每一位元對應一個 IO 口引腳。當 IO 口對應位被置<1>，則開啟內部 75k 上拉電阻，若被置<0>，則關閉內部上拉電阻。IO 口作為輸入模式時，若外部沒有上拉電阻，必須開啟內部上拉電阻，尤其在低功耗模式時，可以防止漏電，而增加功耗。作為類比信號輸入口時，不用開啟內部上拉電阻。

輸出模式

控制器 PT3OE[7:0]可設置每個 IO 口輸出模式的開啟與關閉，每一位元對應一個 IO 口引腳。當 IO 口對應位被置<1>，則開啟對應 IO 口輸出模式；若被置<0>，則關閉輸出模式。透過控制位 PT3DO[7:0]來控制對應 IO 口引腳的輸出狀態為 1 或 0。在低功耗模式下，若 IO 必須開啟輸出模式，可根據外圍電路來設置輸出狀態，來減低晶片功耗。此模式下不能開啟 IO 內部上拉電阻，且不能同時開啟輸入、輸出模式，因而在開啟輸出模式時，需要關閉 IO 口的輸入模式。

輸入模式

控制器 PT3IE[7:0]可設置每個 IO 口引腳輸入模式的開啟與關閉，每一位元對應一個 IO 口引腳。當控制器對應位元被置<1>，則開啟對應 IO 口引腳的輸入模式；若被置<0>，則關閉輸入模式。透過控制器 PT3DI[7:0]可讀取當前對應 IO 引腳的輸入狀態為 1 或 0。當 IO 被設置為輸入模式，若晶片沒有接入外部上拉電阻，則必須開啟晶片內部上拉電阻，不能允許 IO 引腳出現浮接狀態，以免造成晶片產生漏電現象；特別是在低功耗模式下，建議將 IO 引腳設置為輸入模式。作為類比信號輸入口時，不用設置對應 IO 引腳為輸入模式。在開啟輸入模式前，需要關閉對應 IO 引腳的輸出模式。

14.2 暫存器位址

GPIO Register Address	31	24	23	16	15	8	7	0
GPIO Base Address + 0X00(0X40820)	MASK1	PT3PU	MASK0	PT3OE				
GPIO Base Address + 0X04 (0X40824)	MASK3	PT3IE	MASK2	PT3DO				
GPIO Base Address + 0X08(0X40828)	-	-	REG4	PT3DI				
GPIO Base Address + 0X0C (0X4082C)	PT3ITD	PT3ITT	PT3ITT	PT3ITT				

-Reserved

14.3 暫存器功能

14.3.1 PT3 暫存器 PT3CR0

GPIO Base Address + 0X20 (0X40820)									
Symbol	PT3CR0 (PT3 Control Register 0)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名稱	MASK	PT3PU7	PT3PU6	PT3PU5	PT3PU4	PT3PU3	PT3PU2	PT3PU1	PT3PU0
RW	R0W-0	RW-0							
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名稱	MASK	PT3OE7	PT3OE6	PT3OE5	PT3OE4	PT3OE3	PT3OE2	PT3OE1	PT3OE0
RW	R0W-0	RW-0							

位元	名稱	描述
Bit[23~16]	PT3PU	Port 3 內部上拉開啟控制
		0 關閉內部上拉
		1 開啟內部上拉
Bit[7~0]	PT3OE	Port 3 PAD 輸出模式開啟控制
		0 關閉輸出模式
		1 開啟輸出模式

14.3.2 PT3 暫存器 PT3CR1

GPIO Base Address + 0X24 (0X40824)									
Symbol	PT3CR1 (PT3 Control Register 1)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名稱	MASK	PT3IE7	PT3IE6	PT3IE5	PT3IE4	PT3IE3	PT3IE2	PT3IE1	PT3IE0
RW	R0W-0				RW-0				
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名稱	MASK	PT3DO7	PT3DO6	PT3DO5	PT3DO4	PT3DO3	PT3DO2	PT3DO1	PT3DO0
RW	R0W-0				RW-0				

位元	名稱	描述	
Bit[23~16]	PT3IE	Port 3 PAD	輸入模式開啟控制
		0	關閉輸入模式
		1	開啟輸入模式
Bit[7~0]	PT3DO	Port 3 PAD	輸出狀態值
		0	置 0
		1	置 1

14.3.3 PT3 暫存器 PT3CR2

GPIO Base Address + 0X28 (0X40828)									
Symbol	PT3CR2 (PT3 Control Register 2)								
Bit	[31:24]	[23:18]						[17]	[16]
名稱	MASK	-						PT3AO	-
RW	R0W-0	-						R-X	-
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名稱	-	PT3DI[7]	PT3DI[6]	PT3DI[5]	PT3DI[4]	PT3DI[3]	PT3DI[2]	PT3DI[1]	PT3DI[0]
RW	-				R-0				

位元	名稱	描述	
Bit[17]	PT3A0	DAO	輸出至 PT3.1 使能控制
		0	關閉
		1	開啟
Bit[7~0]	PT3DI	Port3 PAD	輸入狀態值
		0	輸入低電位
		1	輸入高電位

15.GPIO PT4 管理

15.1 整體總說明

PT4 目前只有一個 IO 引腳 PT4.0，可以作為通用的普通 IO 口，亦可複用外部復位(Reset)輸入 IO 口。但是目前 PT4.0 功能不開放使用。

16.GPIO 複用功能管理

16.1 整體總說明

晶片具有 3 組通用 IO 口，且都具有複用功能，需要暫存器控制複用功能的開啟與關閉。本章節介紹 IO 口複用功能的控制。

IO 口都具有多個複用功能，但是一般同一時間只能設置某個功能有效，所以在不需用到複用功能時，請記得關閉複用功能，以便作為其他功能使用。也有一些是多個複用功能組合使用，如 PT1/PT2，在被設置為外部中斷輸入口時，同樣也可配置為 IIC、SPI、UART 等信號的輸入口，切記要同樣是輸入口才行，這樣可以通過通訊信號配合外部中斷功能，實現由通訊產生的外部中斷功能。

表 16-1 列出所有 IO 引腳的複用功能，及複用功能的優先權級別，0 代表最高級別，6 代表最低級別。

GPIO Port	OSC	Interrupt	Timer C Capture	SPI	IIC	UART	CMP	Analog	Timer B PWM
Priority	0	0	0	1	2	3	4	5	6
PT1.0		INT1.0	TCI1_1	CS_1	SCL_1	TX_1	CH1		PWM0_1
PT1.1		INT1.1	TCI2_1	CK_1	SDA_1	RX_1	CH2		PWM1_1
PT1.2		INT1.2	TCI1_2	MISO_1	SCL_2	TX_2	CH3		PWM0_2
PT1.3		INT1.3	TCI2_2	MOSI_1	SDA_2	RX_2	CL1		PWM1_2
PT1.4		INT1.4	TCI1_3	CS_2	SCL_3	TX_3	CL2		PWM0_3
PT1.5		INT1.5	TCI2_3	CK_2	SDA_3	RX_3	CL3		PWM1_3
PT1.6		INT1.6	TCI1_4	MISO_2	SCL_4	TX_4	CL4		PWM0_4
PT1.7		INT1.7	TCI2_4	MOSI_2	SDA_4	RX_4	CMPO1		PWM1_4
PT2.0		INT2.0	TCI1_5	CS_3	SCL_5	TX_5			PWM0_5
PT2.1		INT2.1	TCI2_5	CK_3	SDA_5	RX_5			PWM1_5
PT2.2		INT2.2	TCI1_6	MISO_3	SCL_6	TX_6			PWM0_6
PT2.3		INT2.3	TCI2_6	MOSI_3	SDA_6	RX_6			PWM1_6
PT2.4	LSXT1	INT2.4	TCI1_7	CS_4	SCL_7	TX_7			PWM0_7
PT2.5	LSXT2	INT2.5	TCI2_7	CK_4	SDA_7	RX_7			PWM1_7
PT2.6	HSXT1	INT2.6	TCI1_8	MISO_4	SCL_8	TX_8			PWM0_8
PT2.7	HSXT2	INT2.7	TCI2_8	MOSI_4	SDA_8	RX_8			PWM1_8
PT3.0							OPO1		
PT3.1							OPO2	DAO	
PT3.2								AIO4	
PT3.3								AIO5	
PT3.4								AIO6	
PT3.5								AIO7	
PT3.6								REFO	
PT3.7								OPO	
AIO0								AIO0	
AIO1								AIO1	
AIO2								AIO2	
AIO3								AIO3	

表 16-1 IO 引腳複用功能分佈及優先級別

16.2 暫存器位址

GPIO Register Address	31	24	23	16	15	8	7	0
GPIO Base Address + 0X40(0X40840)	MASK1	PTCN1	MASK0	PTCN0				
GPIO Base Address + 0X44(0X40844)	MASK3	PTCN3	MASK2	PTCN2				

16.3 暫存器功能

16.3.1 GPIO 複用功能控制暫存器 GPIOMCR0

Symbol	GPIO Base Address + 0X40 (0X40840)							
	GPIOMCR0 (GPIO multiplex Control Register 0)							
Bit	[31:24]	[23:22]	[21]	[20]	[19]	[18]	[17]	[16]
\名稱	MASK	-	-	-	PTCOPS	PTCOPE	-	PTCCPE
RW	R0W-0	-	RW-0	RW-1	RW-0	RW-0	-	RW-0
Bit	[15:08]	[7:5]		[4:2]			[1]	[0]
名稱	MASK	PTCTC[2:0]		PTPW[2:0]			PTPW1E	PTPW0E
RW	R0W-0	RW-0						

位元	名稱	描述
Bit[19]	PTCOPS	軌對軌 OPAMP 數字信號輸出埠選擇
		0 Port 3.0
		1 Port 3.1
Bit[18]	PTCOPE	軌對軌 OPAMP 數字信號輸出埠開啟控制
		0 關閉，無輸出
Bit[16]	PTCCPE	1 開啟，輸出至設置的目標埠
		比較器輸出端 IO 口開啟控制
		0 關閉，只作為普通 IO，無信號輸出。
		1 開啟

位元	名稱	描述		
Bit[7~5]	PTCTC	捕捉比較器信號輸入端 IO 口選擇		
		000	Port 1.0 =TCI1	Port 1.1 =TCI2
		001	Port 1.2 =TCI1	Port 1.3 =TCI2
		010	Port 1.4 =TCI1	Port 1.5 =TCI2
		011	Port 1.6 =TCI1	Port 1.7 =TCI2
		100	Port 2.0 =TCI1	Port 2.1 =TCI2
		101	Port 2.2 =TCI1	Port 2.3 =TCI2
		110	Port 2.4 =TCI1	Port 2.5 =TCI2
		111	Port 2.6 =TCI1	Port 2.7 =TCI2
Bit[4~2]	PTPW	PWM 輸出端 IO 口選擇		
		000	Port 1.0 =PWMO0	Port 1.1 =PWMO1
		001	Port 1.2 =PWMO0	Port 1.3 =PWMO1
		010	Port 1.4 =PWMO0	Port 1.5 =PWMO1
		011	Port 1.6 =PWMO0	Port 1.7 =PWMO1
		100	Port 2.0 =PWMO0	Port 2.1 =PWMO1
		101	Port 2.2 =PWMO0	Port 2.3 =PWMO1
		110	Port 2.4 =PWMO0	Port 2.5 =PWMO1
		111	Port 2.6 =PWMO0	Port 2.7 =PWMO1
Bit[1]	PTPW1E	PWM 1 IO 口輸出開啟控制		
		0	關閉 (IO 口沒有輸出)	
Bit[0]	PTPW0E	PWM 0 IO 口輸出開啟控制		
		0	關閉 (IO 口沒有輸出)	
		1	開啟 (輸出口有 PTPW 設置)	

16.3.2 GPIO 暫存器 GPIO MCR1

GPIO Base Address + 0X44 (0X40844)					
Symbol	GPIO MCR1 (GPIO Multiplex Control Register 1)				
Bit	[31:24]	[23:20]	[19:17]	[16]	
名稱	MASK	-	I2CPTS[2:0]	I2CPTEn	
RW	R0W-0	-	-	RW-0	
Bit	[15:08]	[7]	[6:5]	[4]	[3:1] [0]
名稱	MASK	-	PTCSP[1:0]	PTSPE	PTUR[2:0] PTURE
RW	R0W-0	-	-	-	RW-0

位元	名稱	描述
Bit[19~17]	I2CPTS	I2C 通訊 IO 口選擇
		000 Port 1.0 =SCL Port 1.1 =SDA
		001 Port 1.2 =SCL Port 1.3 =SDA
		010 Port 1.4 =SCL Port 1.5 =SDA
		011 Port 1.6 =SCL Port 1.7 =SDA
		100 Port 2.0 =SCL Port 2.1 =SDA
		101 Port 2.2 =SCL Port 2.3 =SDA
		110 Port 2.4 =SCL Port 2.5 =SDA
		111 Port 2.6 =SCL Port 2.7 =SDA
Bit[16]	I2CPTEn	I2C 通訊 IO 口複用功能開啟控制
		0 關閉 (無信號輸出)
		1 開啟 (IO 口複用為 I2C 通訊口，IO 口由 I2CPTS 設置)

位元	名稱	描述
Bit[6~5]	PTCSP	SPI 通訊 IO 口選擇
		00 Port1.0 =CS, Port1.1 =CK, Port1.2 =MISO, Port1.3 =MOSI
		01 Port1.4 =CS, Port1.5 =CK, Port1.6 =MISO, Port1.7 =MOSI
		10 Port2.0 =CS, Port2.1 =CK, Port2.2 =MISO, Port2.3 =MOSI
		11 Port2.4 =CS, Port2.5 =CK, Port2.6 =MISO, Port2.7 =MOSI
Bit[4]	PTSPE	SPI 通訊 IO 複用功能開啟控制
		0 關閉(只作為普通 IO 口)
		1 開啟(IO 口複用為 SPI 通訊口，通訊 IO 口由 SPPTS 設置)
Bit[3~1]	PTUR	UART 通訊 IO 口選擇
		000 Port 1.0 =TX Port 1.1 =RX
		001 Port 1.2 =TX Port 1.3 =RX
		010 Port 1.4 =TX Port 1.5 =RX
		011 Port 1.6 =TX Port 1.7 =RX
		100 Port 2.0 =TX Port 2.1 =RX
		101 Port 2.2 =TX Port 2.3 =RX
		110 Port 2.4 =TX Port 2.5 =RX
Bit[0]	PTSPE	EURAT 通訊 IO 口複用功能開啟控制
		0 關閉(只作為普通 IO 口)
		1 開啟(IO 口複用為 EURAT 通訊口，通訊 IO 口由 PTUR 設置)

17. 模數轉換器 ΣΔ 24 位元 ADC

17.1 整體總說明

晶片帶有一個嵌入式的高效 24 位元類比模數轉換器(24-bitΣΔADC)。ADC 前置一個低噪聲，可編程增益放大器(Low Noise PGA)，被用來放大輸入訊號。這增益可編程設定範圍為 1~128。LN PGA 的最小輸入參考噪聲為 65nV，且它的增益溫度系數小於 5ppm/°C。ADC 的採樣率可通過暫存器編程設定。所設計的採樣率為每秒 350K 個樣本。它有一個三階調節器用於過濾調節器的量子化噪音。ADC 的過採樣率可編程範圍是 32~32768。它是設計用來測量輸出信號非常小的感應器，例如應變計、壓力錶、和工業處理控制。

特性：

- (1) 可設置採樣頻率 40KHZ~3MHZ；
- (2) 解析度高達 21 位元的有效位元數(ENOB)；
- (3) 最低輸入噪音為 65nV RMS；
- (4) 可設置超採樣頻率為 32~32768；
- (5) 最高輸出率為 10 KHz；
- (6) 內建低噪音可編程增益放大器，增益倍數為 1~128；
- (7) 內建溫度感應器；
- (8) 內建 4 位元 DAC 來調整偏移；
- (9) 三階梳狀濾波器；

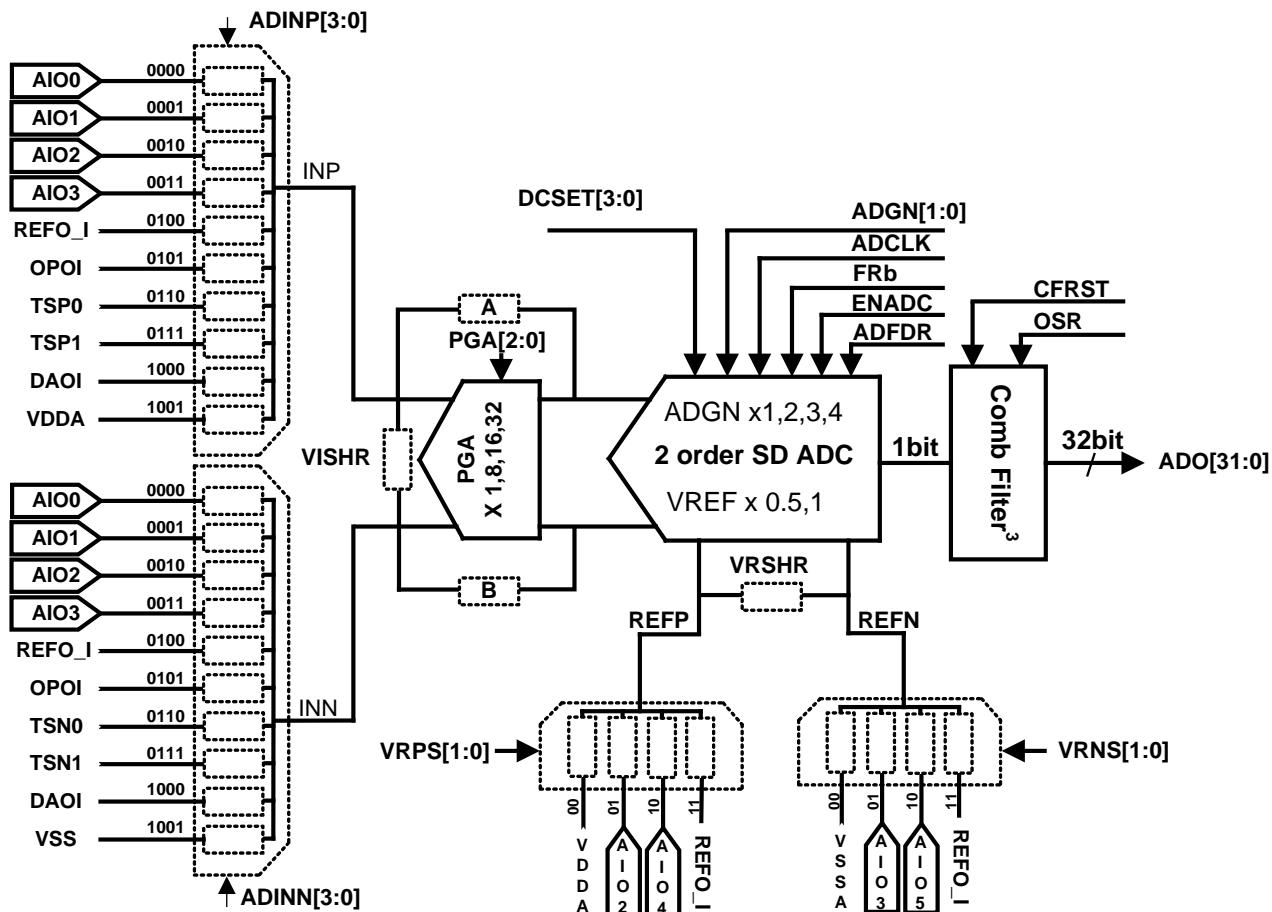


圖 17-1 ADC 功能方框圖

17.1.1 全差動信號輸入端

ADC 的輸入信號為全差動輸入模式，即輸入端由正向輸入端和負向輸入端構成。正向與負向的信號輸入通道都包含 4 個外部信號輸入通道和 6 個內部信號輸入通道，且 ADC 信號輸入端的輸入阻抗為 200K。透過控制器 ADINP[3:0]、ADINN[3:0]選擇正向、負向的信號輸入通道，但是正向輸入端在同一時間只能選擇一路信號輸入通道，負向輸入端在同一時間只能選擇一路信號輸入通道。正向與負向可選同樣的輸入通道，這樣差分信號值為 0。ADC 內部配置一個信號輸入通道短路開關，透過控制位 VISHR 可設置從內部將正向與負向輸入端短路。下圖列出正向端與負向端的信號輸入通道。

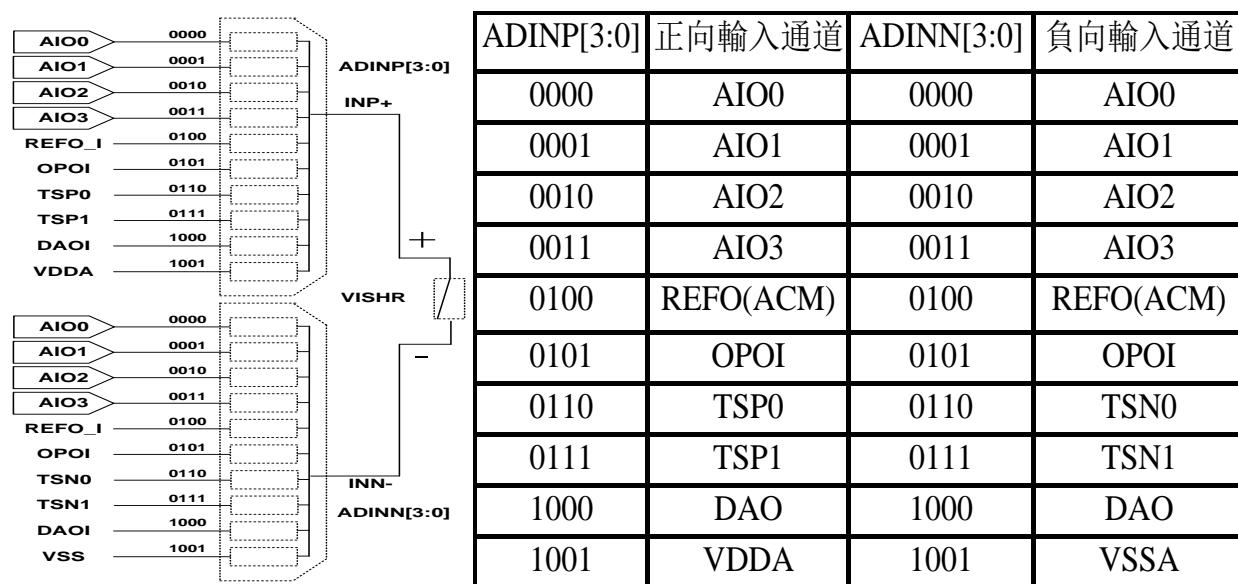


圖 17-2 ADC 信號輸入通道

輸入信號經由內部增大放大再進行轉換，所以對於輸入信號的電壓範圍亦有限制，為了 ADC 輸出能夠得到較高的解析度及線性度，建議猜的信號的差分電壓值 $\Delta SI = \pm 0.9 * \Delta VREF (\Delta SI = INP - INN)$ 。輸入信號電壓如表 17-1 所示。

外部輸入通道	電壓輸入範圍
INP+	$VSSA - 0.2V \leq VREFP \leq VDDA + 0.1V$
INN-	$VSSA - 0.2V \leq VREFN \leq VDDA + 0.1V$

表 17-1 輸入信號電壓範圍表

17.1.2 內置增益放大器

ADC 內置兩個增益放大器：一個低雜訊、低溫度係數的可編程增益放大器 PGA，放大倍數為 8/16/32；一個是可編程增益放大器 ΣAD，放大倍數為 1/2/4。因此兩個增益放大器組合使用內部最大放大倍數為 128。但是放大倍數與 ADC 輸出值有效位數(ENOB)成反比列的，放大倍數越大，ENOB 的值越小。所以在設置放大倍數需要根據實際需要來配置。透過控制器 PGA[2:0]可選擇 PGA 增益器的放大倍率，PGA 的放大倍率選擇如表 所示；透過控制器 ADGN[1:0]可選擇 ΣAD 的增益倍率，ΣAD 的放大倍率選擇如表 所示

PGA[2:0]	PGA				ADGN[1:0]	ΣAD			
	000	001	011	111		00	01	10	11
放大倍率	關閉	x 8	x16	x32	放大倍率	x1	x2	RSV	x4

表 17-2 內部增益放大倍率

17.1.3 參考電壓輸入通道

ADC 參考電壓輸入屬於全差動輸入模式，即參考電壓輸入端由正向輸入端與負向輸入端構成。正向與負向輸入端都包含 2 個外部輸入通道和 2 個內部輸入通道。透過控制器 VRPS[1:0]、VRNS[1:0]可分別設置參考電壓的正向輸入通道、負向輸入通道。正向輸入端在同一時間只能選擇一路輸入通道，負向輸入端在同一時間只能選擇一路輸入通道。參考電壓端還配置一個短路開關，透過控制位 VRSHR 可設置短路開關閉合，將參考電壓的正向輸入端與負向輸入端短路。

參考電壓由 VREFP 與 VREFN 輸入後產生的 $\Delta VREF$ 電壓差，在經過可編程參考電壓衰減器後作為 ADC 的參考電壓值。控制器 FRb[0]可設置參考電壓衰減倍率，參考電壓衰減倍率如表 17-3 所示。

參考電壓計算如下：

$$\Delta VREF = VREFP - VREFN \quad (\text{式 17-1})$$

$$VREF = \text{Gain} \times \Delta VREF \quad (\text{式 17-2})$$

$\Delta VREF$: 輸入引腳輸入電壓的電壓差值； VREF : ADC 內部參考電壓值

VREFP/VREFN : 輸入參考電壓值

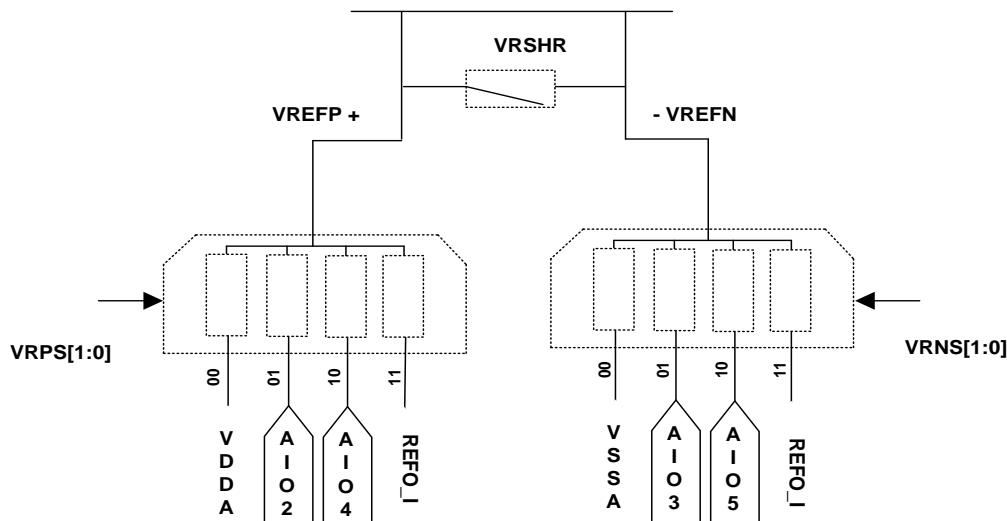


圖 17-3 參考電壓輸入通道

參考電壓衰減倍率		
FRb[0]	0	1
Gain	1	1/2

表 17-3 參考電壓衰減倍率

參考電壓正負輸入通道的輸入阻抗為 $500k\Omega$ ，且 VREFP 或 VREFN 的輸入電壓不可小於 VSS-100mV，也不可超過 VDD+100mV；透過控制器設置為外部輸入通道，可增加輸入阻抗，但也許注意外部輸入通道的電壓值範圍。

為了是 ADC 輸出得到較高的解析度及線性度，估建議參考電壓 $\Delta VREF=0.8V\sim1.2V$ 。

外部輸入通道	電壓輸入範圍
AIO2 / AIO4	$VSSA-0.2V \leq VREFP \leq VDDA+0.1V$
AIO3 / AIO5	$VSSA-0.2V \leq VREFN \leq VDDA+0.1V$

表 17-4 參考電壓外部輸入通道電壓輸入範圍

17.1.4 輸入信號輸入偏壓

ADC 具有零點偏壓平移控制器，零點偏壓平移控制器 DCSET[3:0]透過改變輸入信號零點的位置，以避免因為輸入信號電壓過大而導致超出最大測量範圍溢出。待測信號經過前置 PGA 與 ΣAD 增益放大及零點偏壓平移調整後，等效的待測信號 ΔSI_I 的計算公式如下：

$$\Delta SI_I = PGA \times ADGN \times \Delta SI \pm (DCSET \times \Delta VREF) \quad (\text{式 17-3})$$

	DCSET[3:0]							
設置值	0000	0001	0010	0011	0100	0101	0110	0111
平移量	0 VREF	+1/8VREF	+1/4VREF	+3/8VREF	+1/2VREF	+5/8VREF	+3/4VREF	+7/8VREF
設置值	1000	1001	1010	1011	1100	1101	1110	1111
平移量	0 VREF	-1/8VREF	-1/4VREF	-3/8VREF	-1/2VREF	-5/8VREF	-3/4VREF	-7/8VREF

表 17-5 待測輸入信號零點偏壓設置對照表

17.1.5 梳狀濾波器

$\Sigma\Delta$ ADC 採用三階的梳狀濾波器設置，透過控制器 OSR[3:0]的設置及與 ADC 的採樣頻率組合，可得到不同的過採樣頻率，實現不同的 ADC 轉換值輸出頻率。OSR[3:0]設置參數如表 所示。

	OSR[3:0]										
設置值	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010
除頻值	32768	16384	8192	4096	2048	1024	512	256	128	64	32

表 17-6 過採樣頻率除頻表

類比數位轉換結果存放於暫存器 ADCO[23:0]，最高位為符號位，所以轉換結果與輸入信號的關係如表 所示。

兩極性輸出 二補數格式	等效待測信號	ADCO[23:0]	
		十六進制	二進制
	ΔVR	7F FF FF	0111-1111 1111-1111 1111-1111
	$\Delta VR^*(1/2)$	00 00 01	0000-0000 0000-0000 0000-0001
		00 00 00	0000-0000 0000-0000 0000-0000
	$-\Delta VR$	FF FF FF	1111-1111 1111-1111 1111-1111
	ΔVR	80 00 00	1000-0000 0000-0000 0000-0000

表 17-7 ADCO[23:0]與輸入信號關係表

梳狀濾波器提供復位控制功能，當控制位 CFRST 被置<0>時， Σ 梳狀濾波器進行復位，接著設置 CFRST=1，啟動梳狀濾波器，這樣 $\Sigma\Delta$ ADC 就會自動丟棄前 3 筆資料，用戶讀取到的第一筆 ADC 轉換值就為有效的 ADC 值。

17.1.6 絕對溫度傳感器 TS

絕對溫度傳感器有二極體(BJT)組成，其電壓信號對溫度的變化為已通過 0°K 曲線，其具有以下特色：
溫度傳感器在環境溫度為 0°K 時其輸出的電壓值 $V_{-tpS} @ 0^{\circ}\text{K} = 0\text{V}$ ；

透過測量方式可使得類比數位轉換器 ADC 的偏移電壓 (VADC-OFFSET) 與 BJT 之不對稱性自動抵消；校正溫度僅需單點校正即可滿足±2°C 誤差；若對溫度精度要求較高，可採用兩點校正方式以滿足±1°C 誤差。

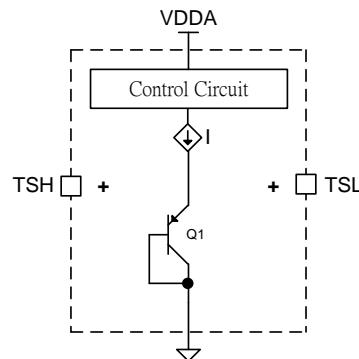


圖 絕對溫度感測器應用方塊圖

◆ TS 初始化設置與計算方式如下操作：

- 啟用 ADC 則 TS 的功能隨即被自動啟用。
- 固定 ADC 與系統工作頻率相關設置，TS 校正時的設置與 TS 測量時的設置需相同。
- 在同一溫度 $T_a(^{\circ}\text{C})$ 下，測量得 V_{TSH0} / V_{TSL0} 與 V_{TSH1} / V_{TSL1} 的數值後，將兩數相加並取平均值即可求得在溫度 T_a 下測得 TS 相對應的電壓值 $V_{TS@Ta}$ 。
 - 測量 V_{TSH0} / V_{TSL0} 時，INxP[2:0] 設置 <111> 且 INxN[2:0] 設置 <110>
 - 測量 V_{TSH1} / V_{TSL1} 時，INxP[2:0] 設置 <110> 且 INxN[2:0] 設置 <111>
 - 將 V_{TSH1} / V_{TSL1} 的數值取 2's 補數後與 V_{TSH0} / V_{TSL0} 數值相加再除 2 即可得到 $V_{TS@Ta}$
- TS 的輸出電壓 V_{TS} 對溫度變化為一線性曲線，故可推導得出其增益值 G_{TS} （或稱斜率）。

$$G_{TS} = \frac{V_{TS@Ta} - V_{TS@0^{\circ}\text{K}}}{273.15 + T_{offset} + Ta} = \frac{V_{TS@Ta}}{289 + Ta} \quad \dots \dots \dots \text{(式 17-4)}$$

G_{TS} : TS 傳感器的增益或斜率
 V_{TS} : TS 傳感器的電壓
 Ta : 環境溫度
 $^{\circ}\text{K}$: 絕對溫度

◆ TS 範例說明

- 假設校正溫度點為 $T_a = 25^{\circ}\text{C} = 298.15^{\circ}\text{K}$
 - 設置一：測得 $V_{TSP@25^{\circ}\text{C}} = 52.515\text{mV}$ (1)
 - 設置二：得測 $V_{TSN@25^{\circ}\text{C}} = -53.626\text{mV}$ (2)
 - $((1) - (2)) \div 2$ 得 $\Delta V_{TS@25^{\circ}\text{C}} = 53.0705\text{mV}$
 - $\therefore V_{TS@0^{\circ}\text{K}} = 0.0\text{mV}$
 - \therefore 曲線斜率 $G_{TS} = (V_{TS@298.15^{\circ}\text{K}} - V_{TS@0^{\circ}\text{K}}) \div (298.15 - 0) = 178\text{uV/}^{\circ}\text{K}$
 - 故任一點的溫度值 $T(^{\circ}\text{C}) = (V_{TS@T} \div G_{TS}) - 289$

17.1.7 ADC 操作說明

ADC 是 24 位元解析度的 $\Delta-\Sigma$ 架構。要啟用 ADC 功能，就需要正確設定一些週邊電路。ADC 的電源是 VDDA

電壓。因此，VDDA 需要高於 2.4V。要獲得較好的 ADC 效能就需有一個穩定的 VDDA 電源供應。因為 VDDA 需要一些時間來就緒，ADC 須等待 VDDA 就緒後才開始量測的動作。將 ENBGR 設為 1 來開啟偏移和帶隙。然後需要一個 1.2V ACM 以啟動 ADC。這個 ACM 電壓可經由外部或內部選擇。ADC 也需要有一個最大 350 KHz 的時脈輸入。這輸入的時脈應該被設定為高於 40 KHz。

詳細的配置操作如下：

- ◆ 配置並啟動 ADC 工作時脈源，建議將 ADC 採樣頻率設置在 330kHz 左右；
- ◆ 開啟 VDDA 電壓及帶隙參考電壓(BandGap Voltage),共模參考電壓(REFO)及模擬地來源；
- ◆ 選擇 ADC 待測信號輸入通道，包括正向、負向輸入通道，並斷開輸入短路開關；
- ◆ 配置 ADC 內部增益放大倍率，根據實際情況設置，讓 ΔSI 在 $0.9 \times VREF$ 範圍內；
- ◆ 設置零點偏壓 DCSET，若不需要，請設置 0 VREF；
- ◆ 選擇 ADC 參考電壓輸入通道，斷開輸入短路開關，並選參考電壓衰減率，建議參考電壓 $VREF=0.8v\sim1.2v$ ；
- ◆ 設置超採樣除頻值 OSR[3:0]，需要根據實際 ENOB 需要來設置；
- ◆ 根據需要開啟 ADC 中斷功能，並使能全局中斷 GIE；
- ◆ 開啟 ADC 功能；
- ◆ 復位梳狀濾波器，CFRST=0，再啟動梳狀濾波器，CFRST=1；自動丟棄前 3 筆資料。

17.2 暫存器位址

ADC Register Address	31	24	23	16	15	8	7	0
ADC Base Address + 0X00 (0X41100)	MASK0		REG0		MASK1		REG1	
ADC Base Address + 0X04 (0X41104)		REG2		REG3		MASK4		REG4
ADC Base Address + 0X08 (0X41108)	ADO3		ADO2		ADO1		0X00	

-保留

17.3 暫存器功能

17.3.1 類比 ADC 暫存器 ADCCR0

ADC Base Address + 0X00 (0X41100)						
Symbol	ADCCR0 (ADC Control Register 0)					
Bit	[31:24]	[23:22]	[21]	[20]	[19:18]	[17:16]
名稱	MASK	-	VISHR	VRSHR	VRPS[1:0]	VRNS[1:0]
RW	R0W-0	-			RW-0	
Bit	[15:08]	[7]	[6]	[5:2]	[1]	[0]
名稱	MASK	-	ADFDR	OSR[3:0]	CFRST	ENADC
RW	R0W-0	-			RW-0	

位元	名稱	描述
Bit[21]	VISHR	ADC 信號輸入端（正向與負向）短路開關控制
		0 短路開關斷開
		1 短路開關閉合
Bit[20]	VRSHR	ADC 參考電壓輸入端（正向與負向）短路開關控制
		0 短路開關斷開
		1 短路開關閉合
Bit[19~18]	VRPS	參考電壓正向輸入源選擇
		00 VDDA
		01 AIO2
		10 AIO4

		11	Reference buffer output(REFO)
Bit[17~16]	VRNS	參考電壓負向輸入源選擇	
		00	VSSA
		01	AIO3
		10	AIO5
		11	Reference buffer output(REFO)

位元	名稱	描述		
Bit[06]	ADFDR	快速斬波器 (Fast chooper) 穩定模式設置		
		0	正常模式，斬波器頻率 = ADCLK/128	
Bit[5~2]	OSR	ADC 過採樣輸出頻率設置 (以 ADC 時脈為 327680Hz 說明)		
		0000	32768	數據輸出頻率 10sps
		0001	16384	數據輸出頻率 20sps
		0010	8192	數據輸出頻率 40sps
		0011	4096	數據輸出頻率 80sps
		0100	2048	數據輸出頻率 160sps
		0101	1024	數據輸出頻率 320sps
		0110	512	數據輸出頻率 640sps
		0111	256	數據輸出頻率 1280sps
		1000	128	數據輸出頻率 2560sps
		1001	64	數據輸出頻率 5120sps
		1010	32	數據輸出頻率 10240sps
		1011	保留 (32768)	
		1100	保留(32768)	
		1101	保留(32768)	
		1110	保留(32768)	
		1111	保留(32768)	
Bit[01]	CFRST	梳狀濾波器開啟控制		
		0	復位 (電平復位)	
		1	開啟	
Bit[00]	ENADC	ADC 開啟控制		
		0	關閉	
		1	開啟	

17.3.2 類比 ADC 暫存器 ADCCR1

ADC Base Address + 0X04 (0X41104)						
Symbol	ADCCR1 (ADC Control Register 1)					
Bit	[31:29]	[28]	[27:24]	[23:22]	[21:20]	[19]
名稱	-	DA	DCSET[3:0]	-	ADGN[1:0]	FRb
RW	-	RW-0	RW-0	-	RW-0	RW-0
Bit	[15:08]			[7:4]	[3:0]	
名稱	MASK			ADINP[3:0]	ADINN[3:0]	
RW	R0W-0			RW-0	RW-0	

adc_04=0X1000FF00; //ADC1 寄存器 0X41104[28]=1 , 表示設置 AIO6 作為 DAC 正向輸入埠

位元	名稱	描述
Bit[28]	DA	AIO6 使能控制，用作 DAC 正向輸入埠 AIO6

		0	關閉	
		1	開啟	
Bit[27~24]	DCSET	DC 零點平移輸入電壓選擇 ($V_{REF} = REFP-REFN$)		
		0000	0 VREF	
		0001	+1/8 VREF	
		0010	+1/4 VREF	
		0011	+3/8 VREF	
		0100	+1/2 VREF	
		0101	+5/8 VREF	
		0110	+3/4 VREF	
		0111	+7/8 VREF	
		1000	0 VREF	
		1001	-1/8 VREF	
		1010	-1/4 VREF	
		1011	-3/8 VREF	
		1100	-1/2 VREF	
		1101	-5/8 VREF	
		1110	-3/4 VREF	
		1111	-7/8 VREF	

Bit	名稱	描述	
Bit[21~20]	ADGN	ADC 輸入信號放大倍數 Gain 調整器設置	
		00	Gain = 1
		01	Gain = 2
		10	保留
		11	Gain = 4
Bit[19]	FRb	參考電壓值範圍選擇	
		0	滿量程參考電壓輸入，即 $V_{REF}*1$
		1	1/2 倍參考電壓輸入，即 $V_{REF}*1/2$
Bit[18~16]	PGA	ADC 輸入信號放大倍數 PGA 調整器設置	
		000	Gain = 1
		001	Gain = 8
		010	保留 (Gain = 8)
		011	Gain = 16
		100	保留(Gain = 16)
		101	保留(Gain = 24)
		110	保留(Gain = 24)
		111	Gain = 32

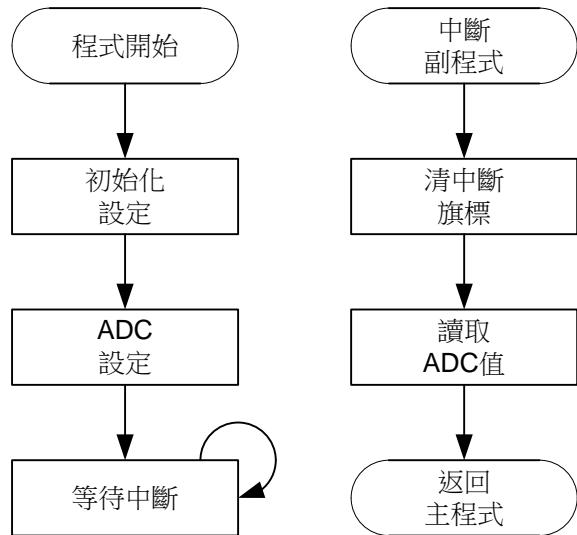
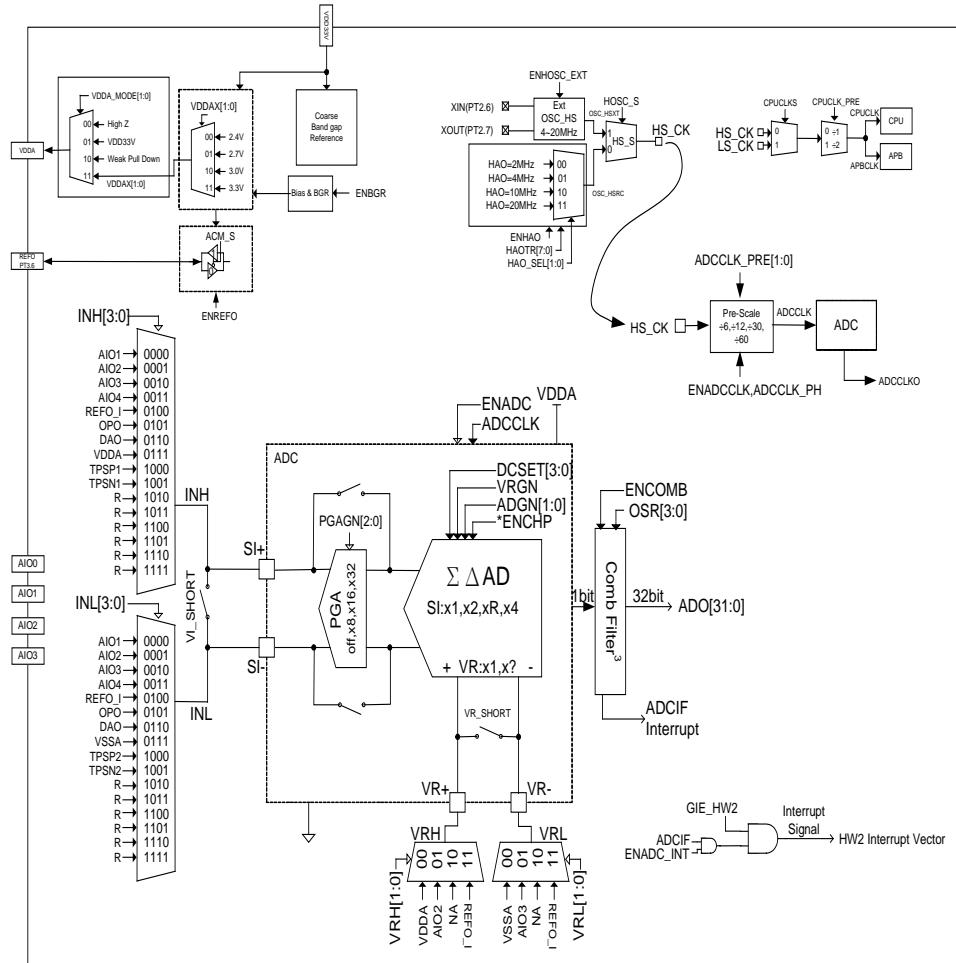
位元	名稱	描述
Bit[7~4]	ADINP	ADC 正向信號輸入端選擇 r
		0000 AIO0
		0001 AIO1
		0010 AIO2
		0011 AIO3
		0100 ACM(REFO_I)
		0101 OPO
		0110 TPSP0
		0111 TPSP1
		1000 DAO
		1001 VDDA
		1010 保留
		1011 保留
		1100 保留
		1101 保留
		1110 保留
		1111 保留
Bit[3~0]	ADINN	ADC 負向信號輸入端選擇
		0000 AIO0
		0001 AIO1
		0010 AIO2
		0011 AIO3
		0100 ACM(REFO_I)
		0101 OPO
		0110 TPSN0
		0111 TPSN1
		1000 DAO
		1001 VSS
		1010 保留
		1011 保留
		1100 保留
		1101 保留
		1110 保留
		1111 保留

17.3.3 類比 ADC 暫存器 ADCCR2

ADC Base Address + 0X08 (0X41108)		
Symbol	ADCCR2 (ADC Control Register 2)	
Bit	[31:16]	
名稱	ADCO	
RW	R-0	
Bit	[15:8]	[7:0]
名稱	ADCO	RSV
RW	R-0	R-0

ADCO[31:0] ADC 轉換值輸出寄存器，只有高 24-bit 數據有效

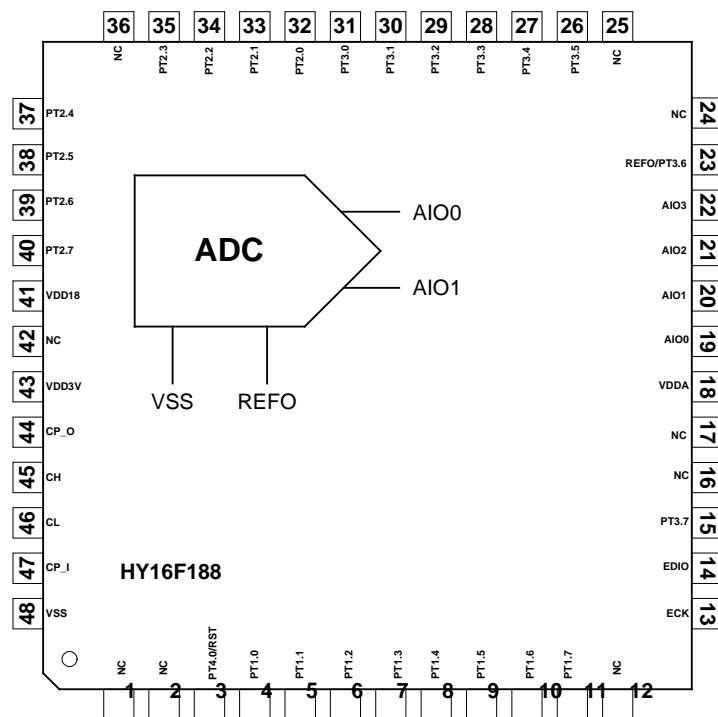
17.4 範常式流程



17.5 範常式式功能

範例名稱:ADC 中斷使用方式

- (1)透過 ADC 相關設定，可以實現 ADC 中斷功能。
- (2)設定 ADC 的電源 VDDA 與 REFO 。
- (3)設定 ADC 的時脈 ADCCLK 。
- (4)ADC 的 Vin 為 AIO0-AIO1
- (5)ADC 的 Vref 為 REFO 對 VSS



17.6 範例程式說明

00	
01	#include "HY16F18X.h"
02	
03	unsigned int ADCData;
04	
05	#define Disable 0
06	#define Enable 1
07	
08	int main(void)
09	{
10	//Set ADC input pin
11	
12	DrvADC_SetADCInputChannel(ADC_Input_AIO0,ADC_Input_AIO1); //Set ADC positive input AIO0
13	//Set ADC negative input AIO1
14	DrvADC_InputSwitch(OPEN); //ADC input short control
15	DrvADC_RefInputShort(OPEN); //ADC reference short control
16	
17	DrvADC_Gain(ADC_PGA_Disable,ADC_PGA_Disable); //Input signal gain
18	
19	DrvADC_DCoffset(0); //DC offset
20	DrvADC_RefVoltage(REF_BUFFER_OUT,VSSA); //ADC reference voltage.
21	
22	DrvADC_FullRefRange(0); //ADC full reference range select
23	//0: Full reference range input
24	//1: 1/2 reference range input

29	//ADC Speed	
30		
31	DrvADC_OSR(1);	//1 : OSR=16384
32	DrvADC_CombFilter(Enable);	//Enable OSR
33		
34	DrvADC_ClkEnable(0,1);	//Setting ADC CLOCK
35		//ADCK=HS_CK/6
36		// Rising edge is high
37		
38	//Set VDDA voltage	
39		
40	DrvPMU_VDDA_Voltage(E_VDDA2_4);	//VDDA=2.4
41	DrvPMU_VDDA_LDO_Ctrl(E_LDO);	//LDO ON
42	DrvPMU_BandgapEnable();	
43	DrvPMU_REF0_Enable();	
44		
45	DrvPMU_AnalogGround(Enable);	//ADC analog ground source set
46		//1 : Enable buffer
47		//and use internal source
48		//need to work with ADC
49		
50	//Set ADC interrupt	
51		
52	DrvADC_EnableInt();	
53	DrvADC_ClearIntFlag();	
54	DrvADC_Enable();	
55		
56	SYS_EnableGIE(7);	//Enable GIE
57		
58	while(1);	//Wait for Interrupt
59		
60	}	
61		
62	void HW2_ISR(void)	//ADC interrupt
63	{	
64	DrvADC_ClearIntFlag();	//Clear ADC interrupt flag
65	ADCData=DrvADC_GetConversionData();	//Get ADC data
66	}	
67		

18. 軌對軌運算放大器 (Rail-to-Rail OPAMP)

18.1 整體總說明

晶片嵌入一個軌對軌運算放大器網路 (OPN)，主要用於類比信號處理。輸入範圍和輸出範圍都是從 VSSA 至 VDDA。輸入信號範圍為 VSSA +0.1 V 和 VDDA -0.1V 之間時，開環增益為 80dB 以上。輸出負載為 50pF 時，單位增益帶寬為 1MHz。它具有 1mA 電流的輸入輸出推挽驅動能力。可驅動的最大電容負載為 1000pF。正輸入端有 4 個獨立的選擇開關，負輸入端有 6 獨立的選擇開關。運算放大器網路內建一個 10pF 的電容器。它可作為輸入採樣電容或用作積分器。不同的輸入通道配置和 8 位元 DAC 設置，它可以被用於不同的應用。運算放大器的輸出端可以被連接到一個 I/O 引腳，或者它可以在內部被其他 IP 使用。當它被用作一個比較器，其輸出是數字格式。用戶可設置運算放大器的輸出經過一個 2ms 的尖峰脈衝濾波器。此外，比較器的輸出可以是斷續或反相的輸出狀態。

OPN 特性包括：

- (1) 軌對軌輸入範圍，以及軌對軌輸出範圍；
- (2) 在 22pF 負載情況下，它有 1MHz 單位的增益頻寬和 60 相位裕量；
- (3) 直流增益 80dB 以上；
- (4) 1mA 推挽輸出驅動能力；
- (5) 正輸入端有 4 個獨立的選擇開關，負輸入端有 6 獨立的選擇開關；
- (6) 內置 10pF 電容；
- (7) 可作為比較器使用，作為比較器時具有斬波器功能；
- (8) 內置尖峰脈衝數字低通濾波器；

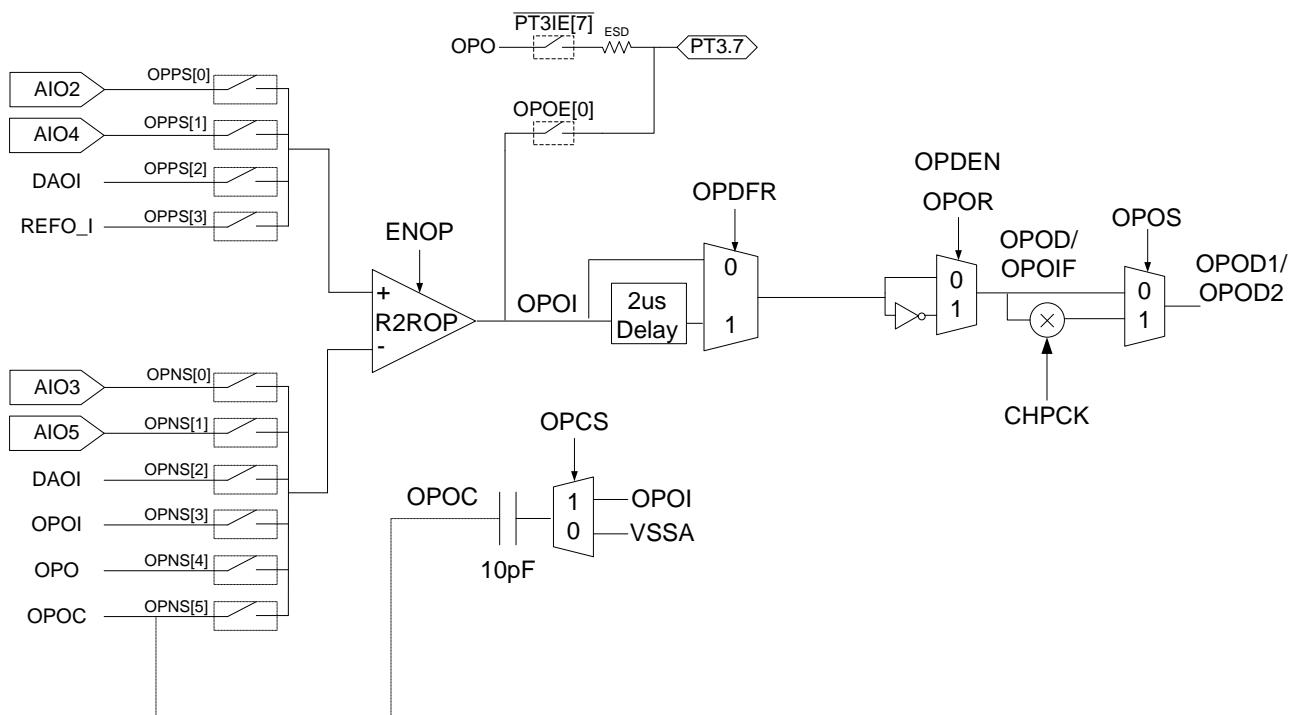


圖 18-1 OPAMP 功能方框圖

18.1.1 輸入通道獨立選擇開關

運放 OPAMP 的輸入通道選擇器不是一個多工器，它們是獨立的選擇開關。運放的正輸入通道，它是由 4 個開關控制： AIO 2, AIO 4, DAO 和 REFO_I，通過控制位 OPPS[0]、OPPS[1]、OPPS[2]和 OPPS[3]，分別進行獨立的控制。可同時選擇多個正輸入通道。運放的負輸入通道，它是由 6 個開關控制： AIO3, AIO5, DAO OPOI, OPO 和 OPOC，通過控制位 OPNS[0]， OPNS[1]，OPNS[2]，OPNS[3]，OPNS[4]和 OPNS[5]，分別進行獨立的控制。亦可同時選擇多個負輸入通道。

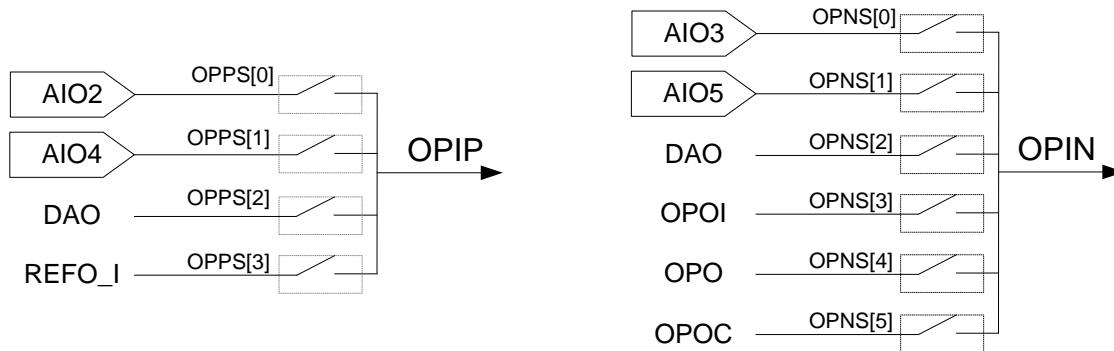


圖 18-2 輸入通道配置圖

18.1.2 內置 10pF 電容

OPN 內置一個 10pF 的電容，它在不同設置下有不同的功能。電容的上端連接 OPOC，它可以連接到運算放大器的負輸入端，該開關是由控制位元 OPNS [5]設置；電容的下端可連接到 OPOI 或 VSSA，可透過控制位 OPCS[0]的設置選擇。有兩種方法來採樣模擬輸入。一種是閉環採樣技術，該方法要求模擬信號是從 AIO3 或 AIO5 輸入。通道開關設置如下：先將 OPNS[5]=1 及將 OPCS[0]=1；然後，設置 OPNS[0]=1 (選擇 AIO3)或設置 OPNS[1]=1 (選擇 AIO5)；採樣完成後，設置 OPNS[5]=0；電壓資訊被存儲在電容器相對於 VSSA。另一種是閉環採樣技術：該方法必須先開啟 OPAMP，即設置 ENOP=1；然後開啟 OPOI 和 OPO，即設置 OPNS[4]=1 及 OPNS[3]=1；再將電容下端連接至 OPOI，即 OPCS=1；然後開啟 AIO2 和 AIO4，即設置 OPPS[0]=1，及 OPPS[1]=1；採樣完成後關閉 OPOC，即 OPNS[5]=0；電壓資訊也存儲在電容器相對於 VSSA。閉環方法可以將運放(OPAMP)的偏移量存儲到電容器。此外，對於傳感器具有非常高的輸出阻抗的應用中，應該使用閉環採樣技術。最後，電容的下端可以連接到運放的輸出端即設置 OPCS=0。同時可過 AIO3 或 AIO5 引腳作為累加充電。

18.1.3 比較器功能

若將運放配置為開環功能，則 OPAMP 可以作為一個比較器使用。通過 OPOD 輸出 1 位二進制代碼。若正向輸入大於負向輸入，則 OPOD 輸出 1；若正向輸入小於負向輸入，則 OPOD 輸出 0。為了防止尖峰脈衝幹擾，OPOD 輸出還可以經過 2us 的低通濾波器。如果任何尖峰脈衝小於 2us，比較器輸出結果不會改變。比較器輸出機型可以通過控制位 OPDR 的設置來改變。

這個比較器的輸出也可以連接到 I/O 引腳，PT3.0/PT3.1 分別是 OPO1/OPO2 的輸出引腳。比較器的輸出結果還可以與電荷泵的時脈頻率(CHPCK)相乘，輸出一個高頻信號，可以作為 LED 驅動器。

18.1.4 操作描述

只是一個比較通用的軌對軌運算放大器(OPAMP)。它可用於模擬信號的處理，被用作運放時，VDDA 電壓必須高於 2.4V，且必須先開啟帶隙參考電壓(BandGap Voltage)。然後設置 ENOP=1，開啟運算放大器 OPAMP。在有效的輸入範圍內，OPAMP 是軌對軌的。但是為獲取更好的性能，估建議輸入共模電壓範圍在 VSSA+0.1V~VDDA-0.1V 之間。運算放大器的輸入阻抗高於 1GΩ。

初始化配置：

- 開啟 VDDA、帶隙參考電壓，VDDA 電壓要大於 2.4V；
- 選擇 OPO1/OPO2 的輸出 IO 引腳，對應的 IO 引腳設置為輸出模式，若不用，可以不配置；
- 選擇正向輸入通道、負向輸入通道，根據實際應用配置；
- 配置 2us 的低通濾波器；根據實際需要設置是否開啟；
- 配置電荷泵時脈頻率，根據實際需要設置是否需要乘以該平率；
- 使能 OPAMP 的模擬輸出，即使能 OPOE；
- 根據需要，使能 OPAMP 的數字輸出，使能 OPDEN；
- 若使能 OPAMP 的數字輸出，根據實際需要，設置輸出結果是否反相，設置 OPDR。
- 使能 OPAMP 功能，開啟運算放大器，即使能 ENOP；

18.2 暫存器位址

OPAMP Register Address	31	24	23	16	15	8	7	0
OPAMP Base Address + 0X00 (0X41900)	-	-			MASK0	REG0		
OPAMP Base Address + 0X04 (0X41904)	OPPSM	OPPS	OPNSM	OPNS				

-Reserved

18.3 暫存器功能

18.3.1 類比 OPA 暫存器 0

OPA Base Address + 0X00 (0X41900)									
Symbol	OPAMP0 (OPAMP Control Register 0)								
Bit	[31:16]								
名稱	RSV								
RW	R-0								
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名稱	MASK	OPOD	OPOS	OPDR	OPCS	OPDFR	OPDEN	OPOE	ENOP
RW	R0W-0	R-0				RW-0			

位元	名稱	描述
Bit[7]	OPOD	OPAMP 數字輸出值，只能讀取。
		0 負向輸入端信號 > 正向輸入端信號；
		1 正向輸入端信號 > 負向輸入端信號；
Bit[6]	OPOS	設置 OPO1/OPO2 輸出經過 CPCLK 多功能器
		0 不經過 CPCLK 多功能器，OPO1/OPO2 輸出值等效於 OPOD
		1 經過 CPCLK 多功能器，OPO1/OPO2 等效於 OPOD 疊加上 CPCLK
Bit[5]	OPDR	OPAMP 數字輸出相位選擇
		0 正常輸出
		1 反相輸出
Bit[4]	OPCS	OPAMP 內置電容用途設置
		1 電容作為集成電容器，下端連接至 OPOI
		0 電容作為採樣電容，下端連接至 VSSA
Bit[3]	OPDFR	OPAMP 輸出數字濾波器開啟控制
		0 關閉
		1 開啟(經過 2us 抗尖峰脈衝)
Bit[2]	OPDEN	OPAMP 數字輸出功能控制

		0	關閉
		1	開啟
Bit[1]	OPOE	OPAMP 模擬輸出功能控制	
		0	關閉
Bit[0]	ENOP	OPAMP 功能開啟控制.	
		0	關閉
		1	開啟

18.3.2 類比 OPA 暫存器 1

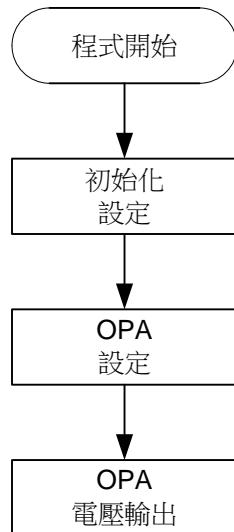
OPA Base Address + 0X04 (0X41904)			
Symbol	OPAMP1 (OPAMP Control Register 1)		
Bit	[31:24]	[23:20]	[19:16]
名稱	MASK	-	OPPS[3:0]
RW	R0W-0	-	RW-0
Bit	[15:08]	[07:06]	[05:00]
名稱	MASK	-	OPNS[5:0]
RW	R0W-0	-	RW-0

位元	名稱	描述
Bit[19]	OPPS[3]	OPAMP 正向輸入通道 3
		0 關閉，高阻態 1 開啟並連接至 REFO_I
Bit[18]	OPPS[2]	OPAMP 正向輸入通道 2
		0 關閉，高阻態 1 開啟並連接至 DAO
Bit[17]	OPPS[1]	OPAMP 正向輸入通道 1
		0 關閉，高阻態 1 開啟並連接至 AIO4
Bit[16]	OPPS[0]	OPAMP 正向輸入通道 0
		0 關閉，高阻態 1 開啟並連接至 AIO2

位元	名稱	描述
Bit[5]	OPNS[5]	OPAMP 負向輸入通道 5
		0 關閉，高阻態 1 開啟並連接至 OPC: 內部 10pF 電容
Bit[4]	OPNS[4]	OPAMP 負向輸入通道 4
		0 關閉，高阻態 1 開啟並連接至 OPO: 內部 OPAMP 輸出
Bit[3]	OPNS[3]	OPAMP 負向輸入通道 3
		0 關閉，高阻態 1 開啟並連接至 OPOI: 內部 OPAMP 輸出
Bit[2]	OPNS[2]	OPAMP 負向輸入通道 2

		0	關閉，高阻態
		1	開啟並連接至 DAO
Bit[1]	OPNS[1]	OPAMP	負向輸入通道 1
		0	關閉，高阻態
Bit[0]	OPNS[0]	OPAMP	負向輸入通道 0
		0	關閉，高阻態
		1	開啟並連接至 AIO3

18.4 範常式式流程

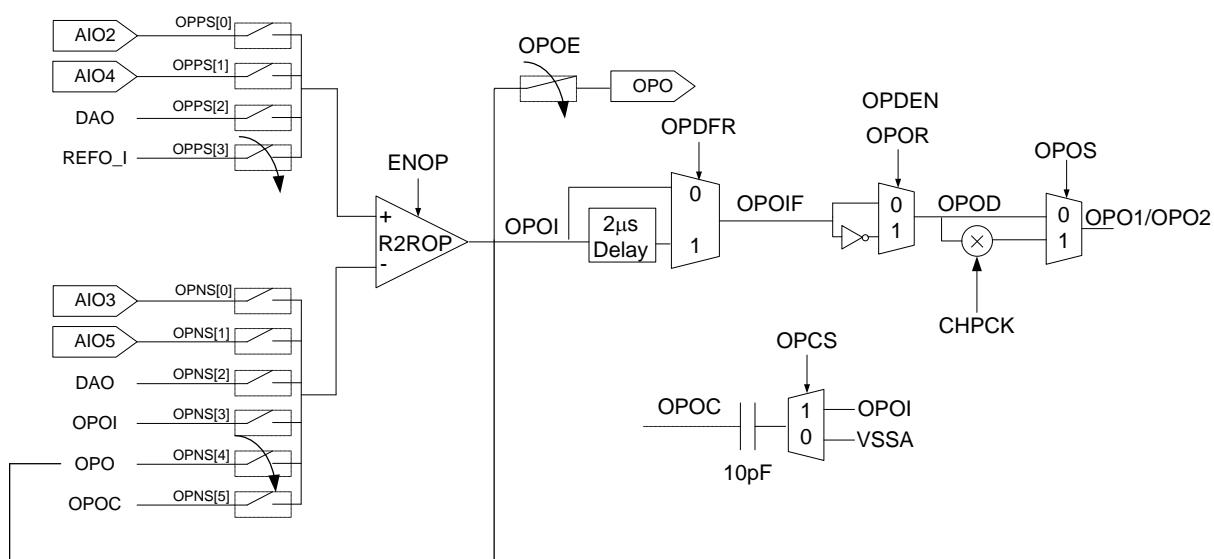
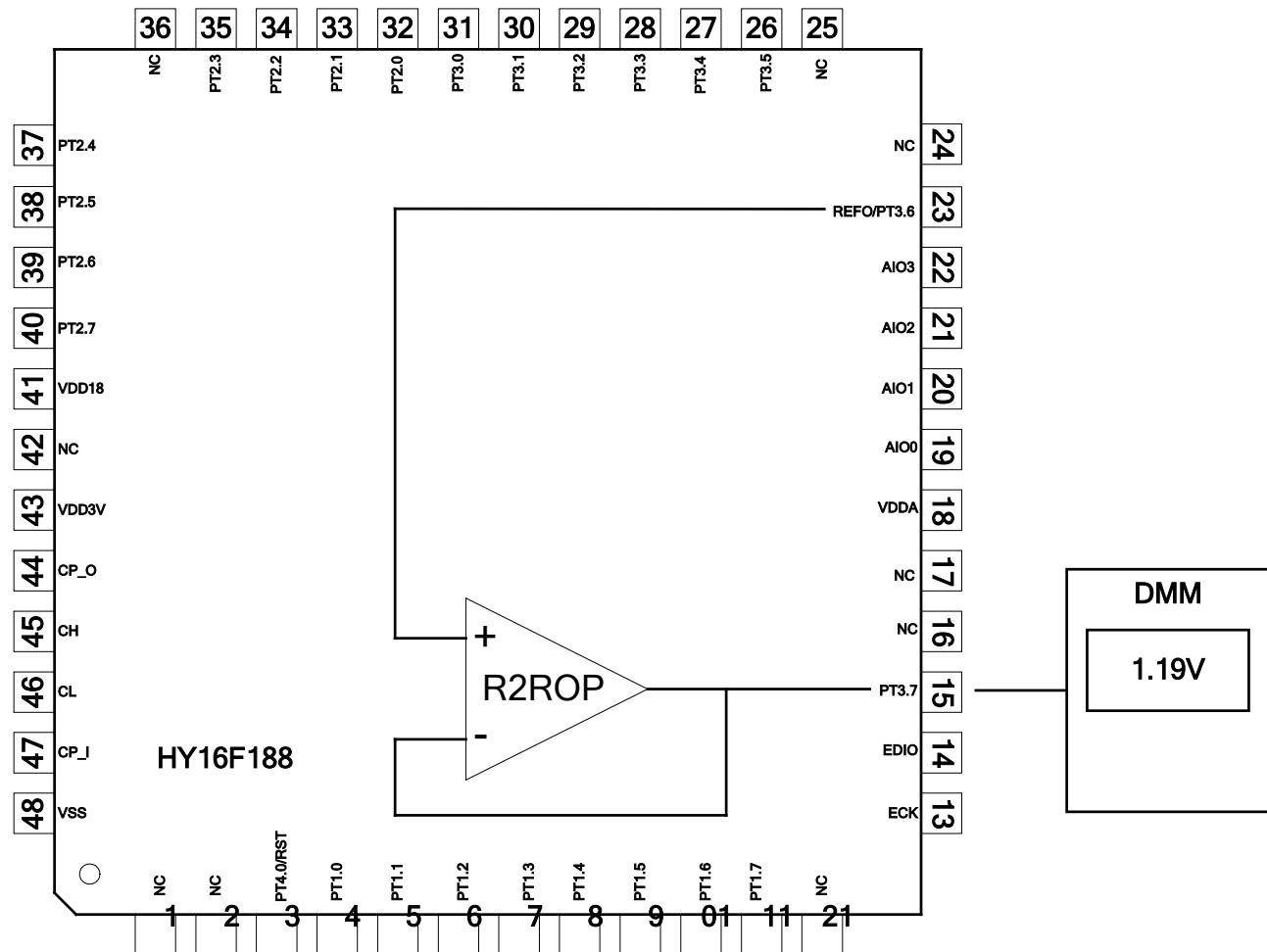


18.5 範常式式功能

範例名稱:OPAMP 使用與設定方式

範例說明:

- (1) 將類比電壓 REFO=1.2V 打開
- (2) 將 REFO 接到 OPAMP 的正端 V+
- (3) OPAMP 的負端選擇 OPOI，此時會接成 OPA Unit Gain Buffer
- (4) 透過適當的 OPAMP 網路設定，可在 OPO 輸出端量到 REFO=1.2V

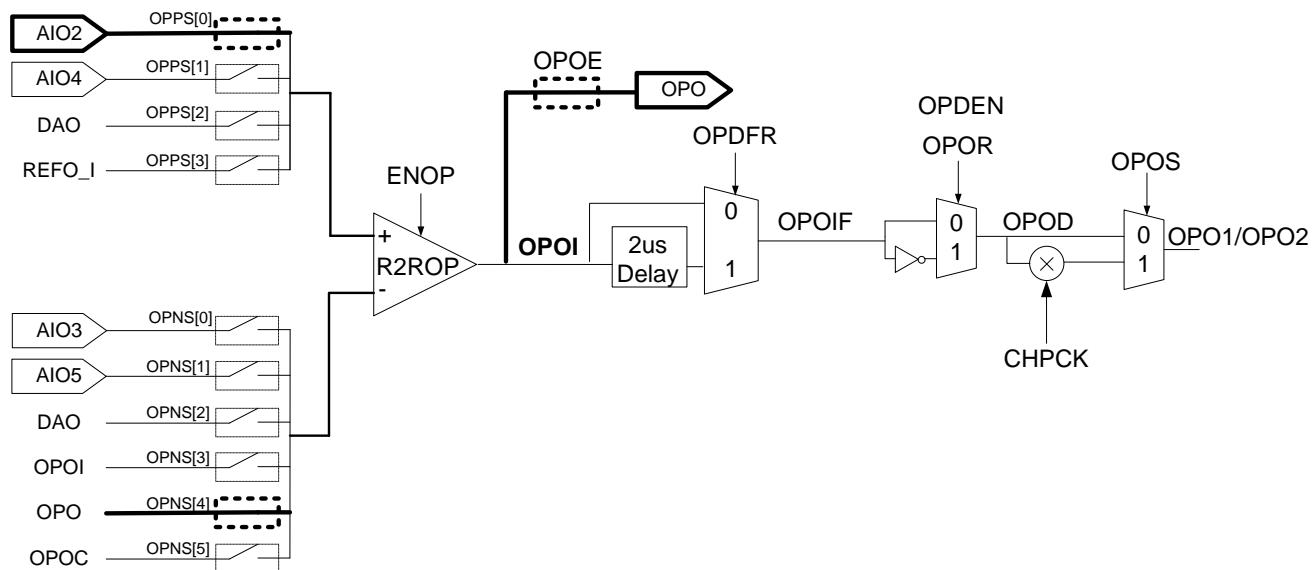


18.6 範常式說明

00		
01	#include "HY16F18X.h"	
02		
03	int main(void)	
04	{	
05	DrvPMU_VDDA_LDO_Ctrl(E_LDO);	//LDO ON
06	DrvPMU_VDDA_Voltage(E_VDDA3_0);	//VDDA=3.0
07	DrvPMU_REF0_Enable();	//REF0 ON
08		
09	DrvOP_Open();	
10		
11	DrvOP_PInput(0X08);	//OPA positive reference input selection REF0
12	DrvOP_NInput(0X10);	//OPA negative reference input selection OPOI
13		
14	DrvOP_OPOoutEnable();	//OP OutEnable
15		
16	while(1);	//while loop
17	}	
18		

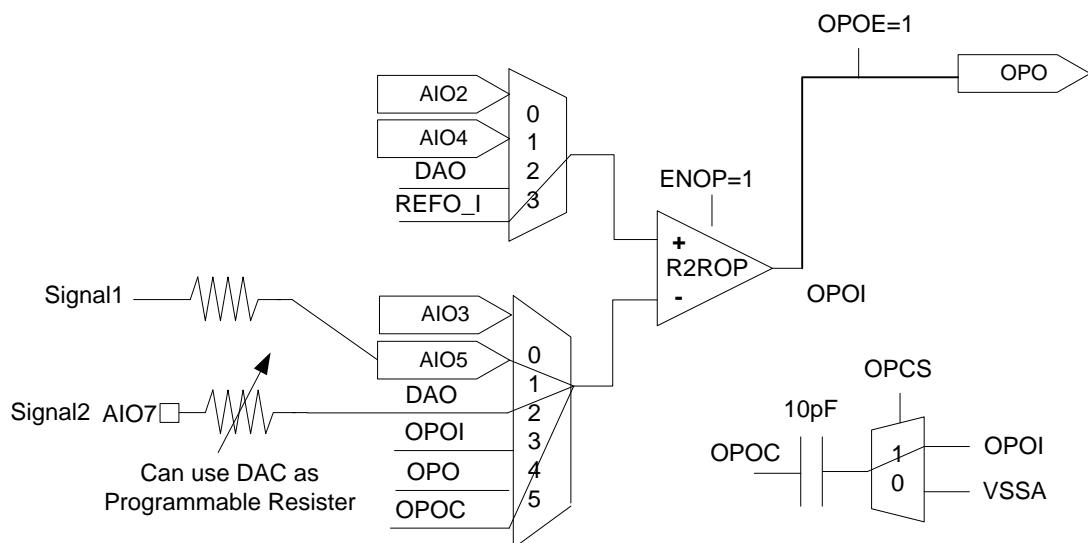
18.4 應用電路系統暫功能 01

Use as a Unit Gain Buffer



18.5 應用電路系統暫功能 02

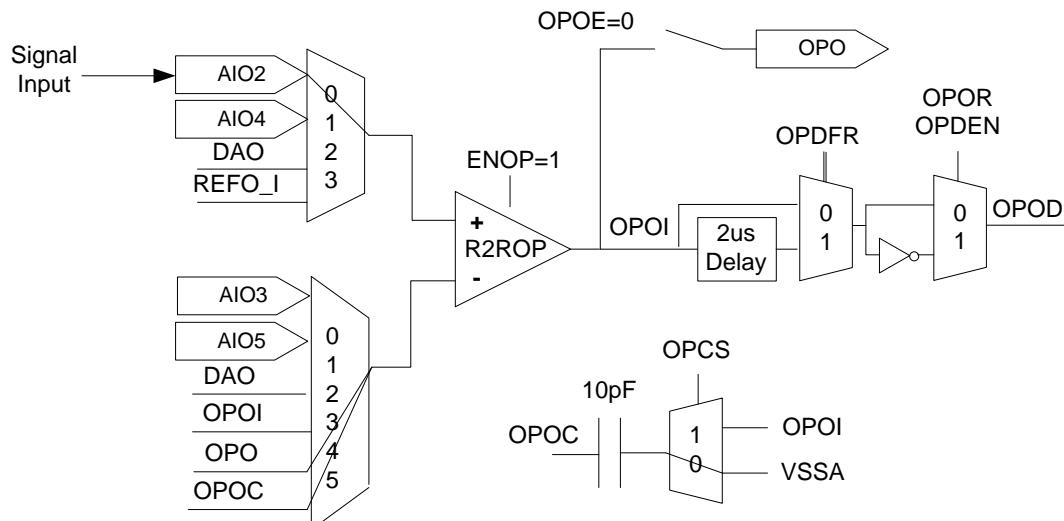
Use as an Integrator



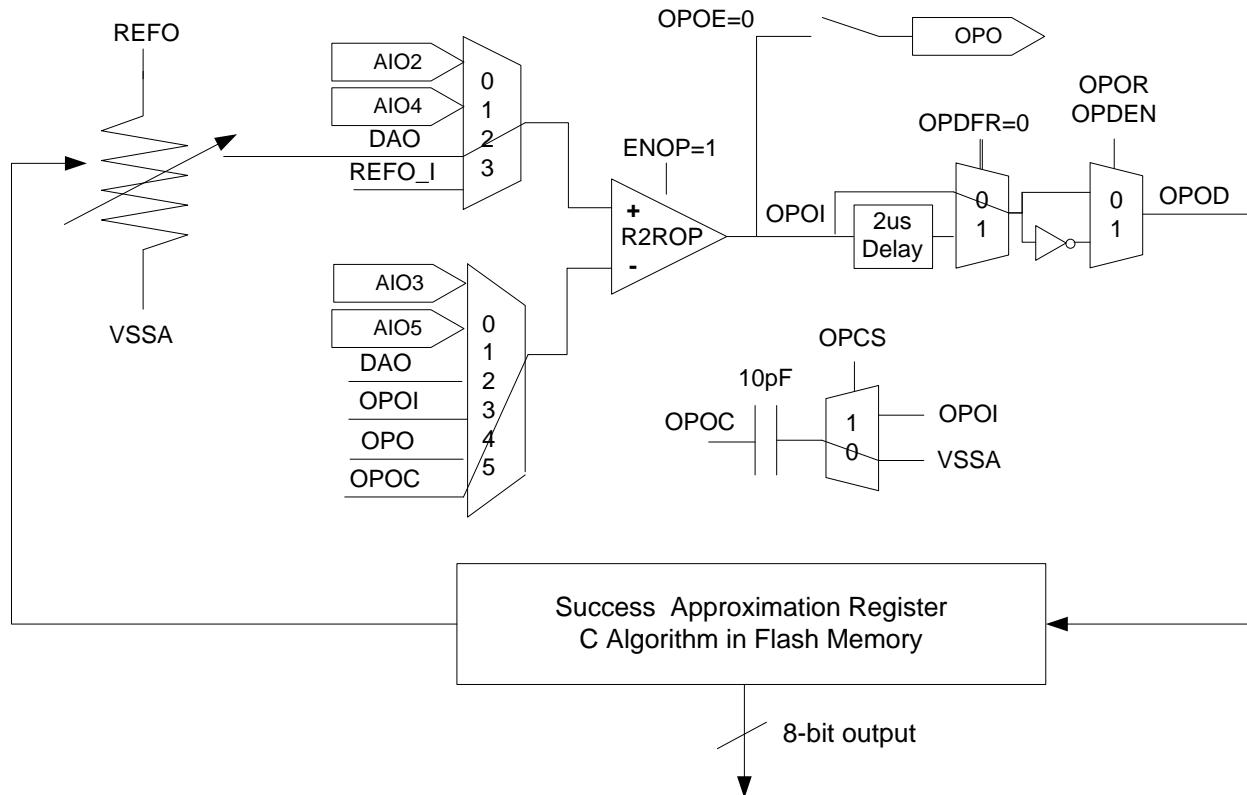
18.6 應用電路系統暫功能 03

Use as a 8-bit SAR ADC

Sample Phase



Conversion Phase



19. 8 bit Resistance Ladder

19.1 整體總說明

晶片內嵌一個 8 位元的數位電阻器。這個 8bit Resistance Ladder 模組是一個 8 位元的數位對類比轉換器。它是由一個保證單調性梯形電阻所構成。

8bit Resistance Ladder 特性包括:

- (1)8 位元的單調輸出
- (2)內部或外部基準的可編程選擇
- (3)可用來當作可編程電阻

8bit Resistance Ladder 的運作:

當 ENDA 是 0，則 8bit Resistance Ladder 會被關閉，就不會消耗電源。 V_{refp} 多工器被關閉，變成一個高阻抗節點。但 V_{refn} 仍然是在開起狀態，且連接到其中的一個來源。如果 DAOE 被設為 1，就會變成具有標量歐姆每一步驟的可編程電阻。

8bit Resistance Ladder 輸出:

DAO 依據儲存在 DABIT 和 $V_{DAC_Vrefp} - V_{DAC_Vrefn}$ 的數據來產生電壓輸出。

DABIT 是直二進制數據格式。圖 23-1 顯示傳輸功能圖。

$$DAO = (V_{DAC_Vrefp} - V_{DAC_Vrefn}) \times \frac{Dabit_in}{256} + V_{DAC_Vrefn}$$

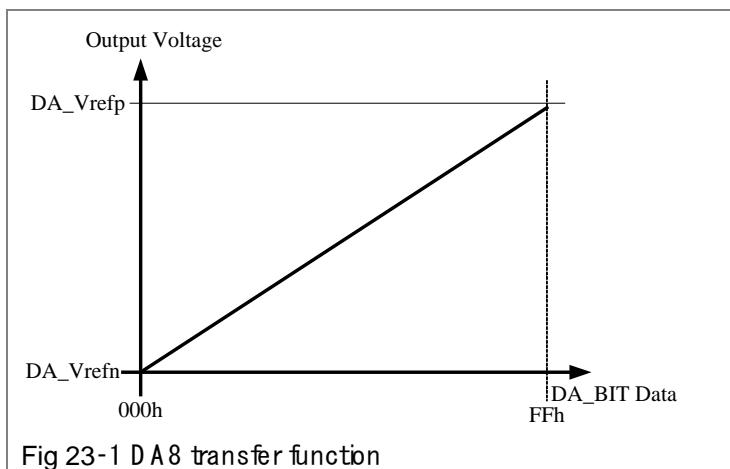


圖 8bit Resistance Ladder 數模轉換圖

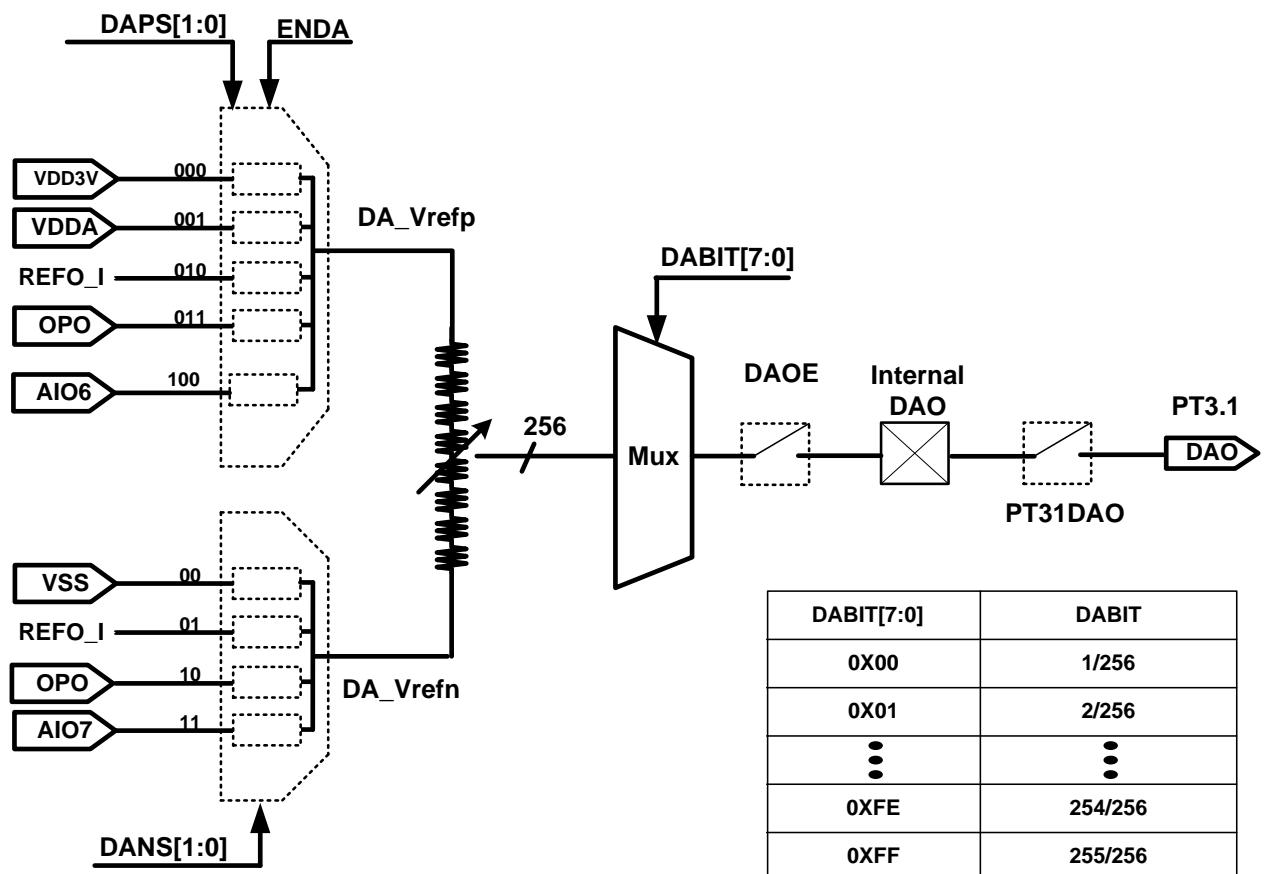


圖 19-1 8bit Resistance Ladder 功能方框圖

19.2 暫存器位址

8bit Resistance Ladder Register Address	31	24	23	16	15	8	7	0
8bit Resistance Ladder Base Address + 0X00 (0X41700)	-		-		MASK0		REG0	
8bit Resistance Ladder Base Address + 0X04 (0X41704)	-		-		MASK1		REG1	

-保留

19.3 暫存器功能

19.3.1 類比 8bit Resistance Ladder 暫存器 8bit Resistance Ladder 0

8bit Resistance Ladder Base Address + 0X00 (0X41700)						
Symbol	8bit Resistance Ladder0 (8bit Resistance Ladder Control Register 0)					
Bit	[31:16]					
名稱	RSV					
RW	R-0					
Bit	[15:8]	[7:6]	[5:4]	[3:2]	[1]	[0]
名稱	MASK	-	DAPS[1:0]	DANS1:0]	DAOE	ENDA
RW	R0W-0	-			RW-0	

位元	名稱	描述
Bit[5~4]	DAPS	DAC 正向輸入源選擇
		000 VDD3V
		001 VDDA
		010 REFO_I
		011 OPO
Bit[3~2]	DANS	DAC 負向輸入源選擇
		00 VSS
		01 REFO_I
		10 OPO
		11 AIO7
Bit[1]	DAOE	DAC 輸出開啟控制.
		0 關閉，處於高阻態
Bit[0]	ENDA	1 開啟，8bit Resistance Ladder 輸出相應電壓
		DAC 功能開啟控制
		0 關閉
		1 開啟

(注*1)

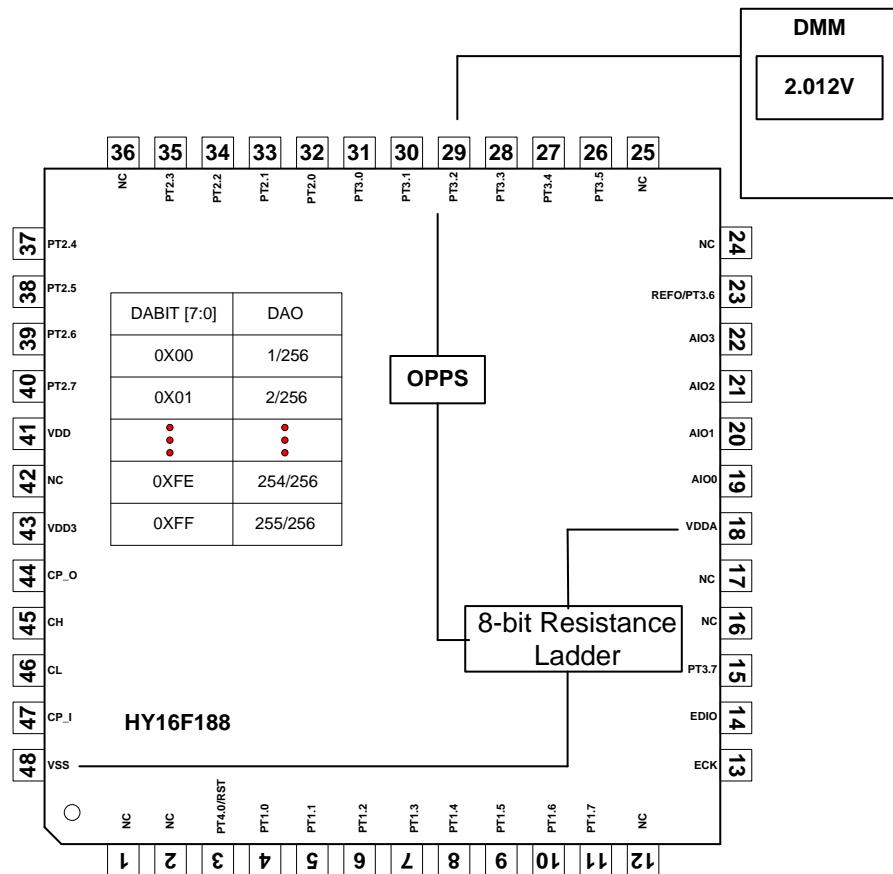
adc_04=0X1000FF00; //ADC1 寄存器 0X41104[28]=1 , 表示設置 AIO6 作為 DAC 正向輸入埠

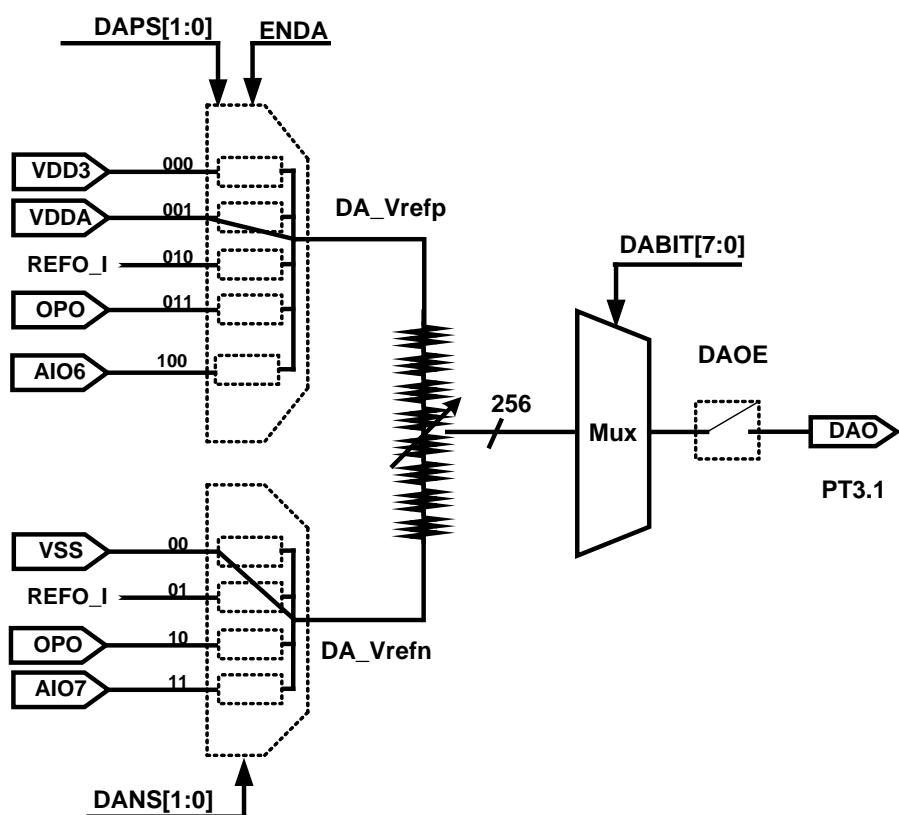
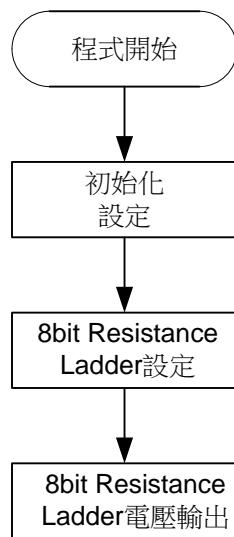
19.3.2 類比 8bit Resistance Ladder 暫存器 8bit Resistance Ladder1

8bit Resistance Ladder Base Address + 0X00 (0X41704)	
Symbol	8bit Resistance Ladder 1(8bit Resistance Ladder Control Register 1)
Bit	[31:16]
名稱	RSV
RW	R-0
Bit	[15:8]
名稱	MASK
RW	DABIT[7:0] R0W-0
	RW-0

位元	名稱	描述
Bit[7~0]	DAO	DAO[7:0] 輸出電壓的比例值設定，即是 DAO[7:0]/255

19.4 範常式式流程





19.5 範例程式功能

範例名稱:8bit Resistance Ladder 使用方式與說明

範例說明:

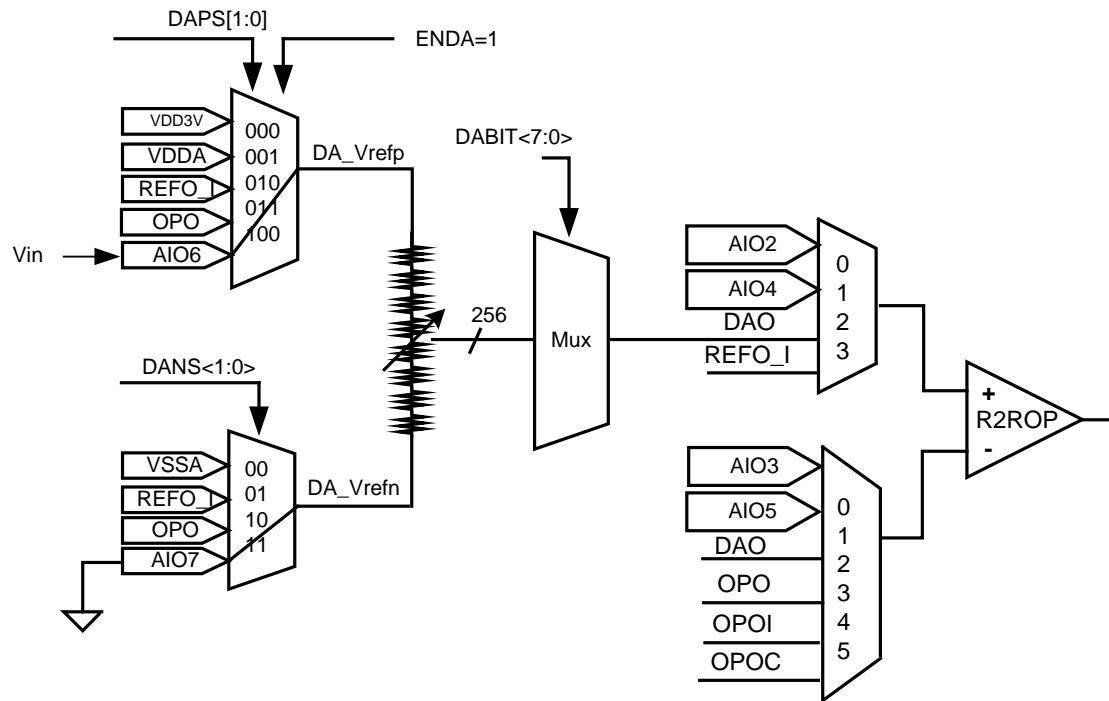
- (1) 將類比電壓與數位電壓做適當設定。
- (2) 將 REFO_I 接到 8bit Resistance Ladder 的正端。
- (3) DAC 的負端選擇 VSS，此時會透過 OP 的 AIO4。(AIO4 和 DAC 要同時選擇)
- (4) 透過適當的 8bit Resistance Ladder 網路設定，可在 AIO4 輸出端量到 256 階的 8bit Resistance Ladder 輸出。
- (5) 或採用 8bit Resistance Ladder 專用類比電壓輸出腳位 PT3.1

19.6 範例程式說明

00		
01	#include "HY16F18X.h"	
02		
03	int main(void)	
04	{	
05	DrvPMU_VDDA_LDO_Ctrl(E_LDO);	//LDO ON
06	DrvPMU_VDDA_Voltage(E_VDDA3_0);	//VDDA=3.0
07		
08	//DrvOP_PInput(0X06);	//DAO output with AIO4(PIN29)
09		
10	DrvDAC_Enable();	//DAC IP enable
11	DrvDAC_EnableOutput();	//DAC output enable
12	DrvDAC_Open(E_DAC_VDDA,E_DAC_VSSA,0X80);	//DA_Vrefp= VDDA
13		//DA_Vrefn= VSS
14	dac_04=0xFF00+0X80;	//DAC BIT 0X80
15	pio_28=0X02020000;	//DAO output with PT3.1
16		
17	while(1);	//while loop
18	}	
19		

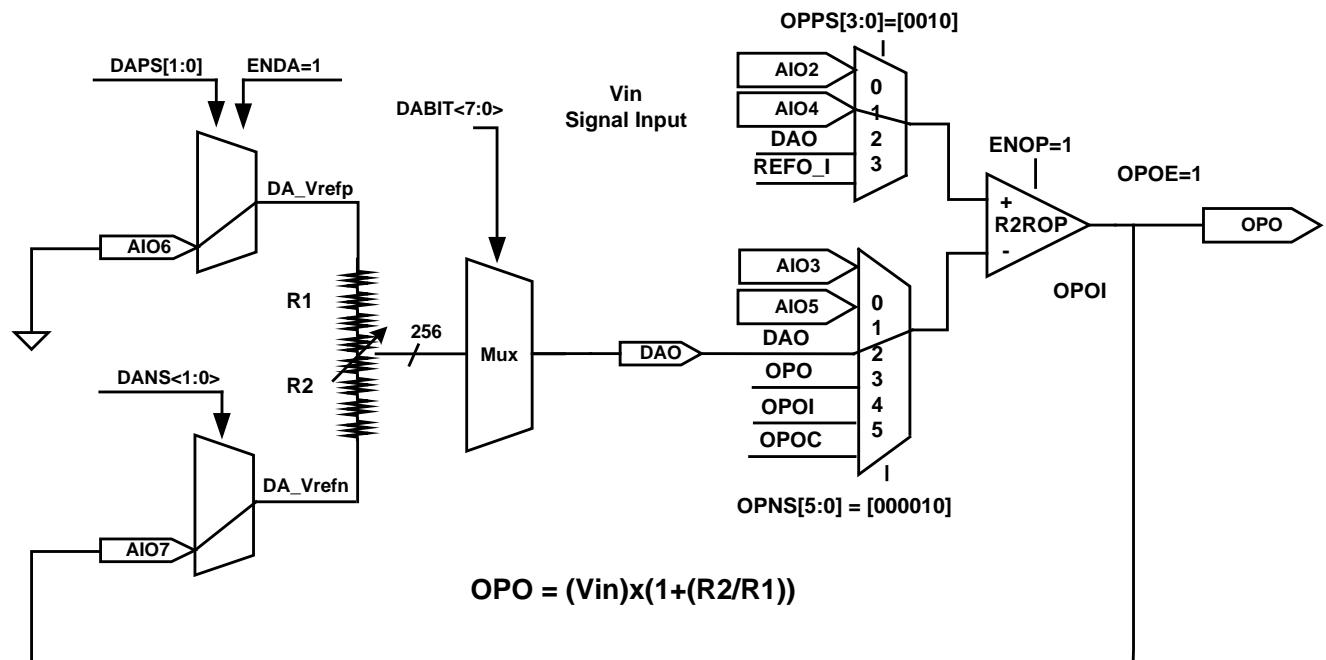
19.7 應用電路功能 01

Use DAO Output



19.8 應用電路功能 02

Use as Programmable Gain Amplifier



20. 多功能比較器 CMP

20.1 整體總說明

晶片內嵌一個低功耗、軌對軌輸入的多功能比較器 CMP，用來作類比訊號的比較。具有中斷功能，當產生比較結果時，可產生中斷信號，增加用戶的操作性。它可以有不同的組態設定來作不同的應用。

CPN 特性包括：

- (1) 軌對軌輸入範圍。
- (2) 低運作電流。
- (3) 2us 尖峰脈衝過濾器。
- (4) 不同比較集的 4 位元梯形電阻 DAC。
- (5) 觸控按鍵量測的充電和放電路徑。
- (6) 可產生中斷信號，屬於中斷向量 HW3。

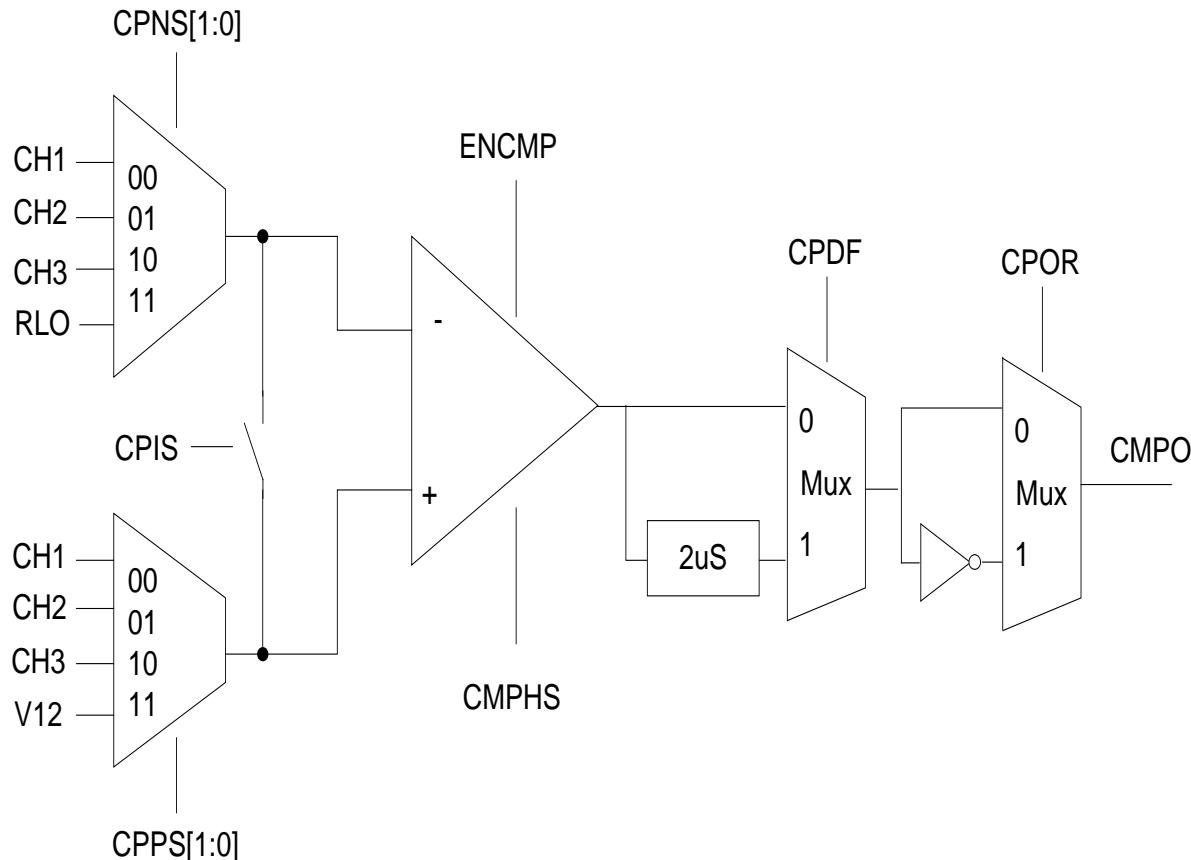


圖 20-1 CMP 網絡框圖

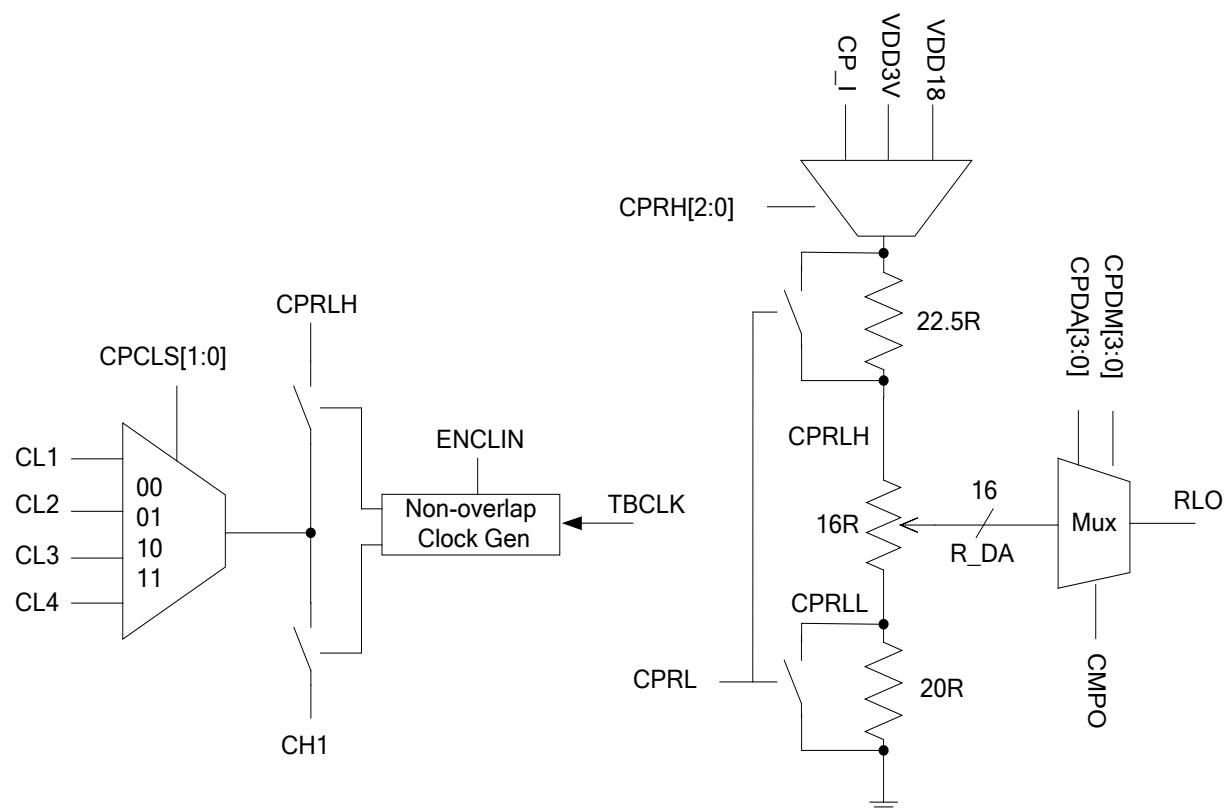


圖 20-2 CMP 網絡框圖-

20.1.1 多工輸入通道選擇器

比較器的輸入通道由兩部份組成，一為比較器的輸入通道，由控制器 CPPS[1:0]/CPNS[1:0]設置選擇，分別設置比較器的正向輸入通道與負向輸入通道；另一為觸摸按鍵輸入通道，由控制器 CPCCLS[1:0]設置選擇。透過正確的設置，兩部份輸入通道的組合使用，可以現實觸摸按鍵的應用。使用時可將控制位 CPIS 置`<1>`，使得比較器的正向輸入端與負向輸入端短路；反之，CPIS 被置`<0>`則不短路。

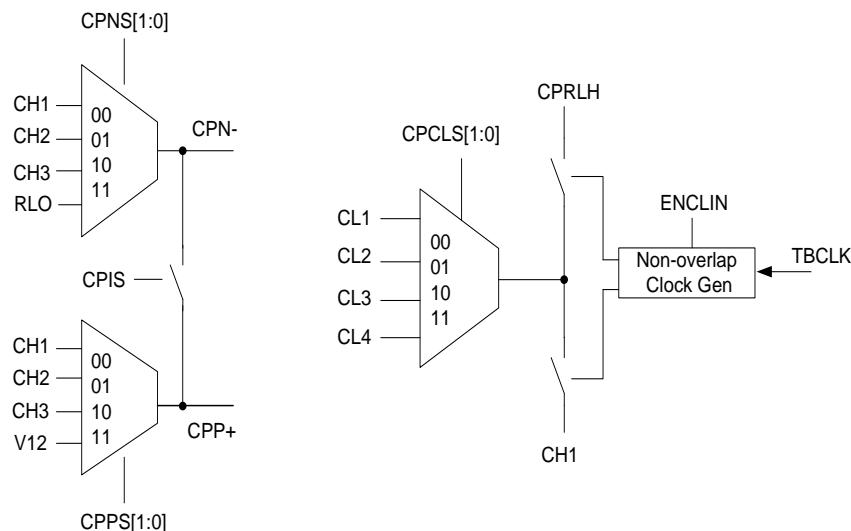


圖 20-3 CMP 輸入通道框架圖

20.1.2 內置多節點電阻器與電阻節點選擇器

比較器內建一個多節點電阻器，電阻分為三部份：22.5R、16R 與 20R。在 16R 電阻處接入一個 16 段的電阻節點選擇器，將 16R 的電阻等分為 16 個節點，可以透過控制器 CPDA[3:0]與 CPDM[3:0]的設置，可選擇不同的電阻節點，輸出不同的電壓值到比較器的輸入通道 RLO。而控制位 CPRLS 被置<1>，可使得 22.5R 與 20R 電阻被短路，可調節電阻節點電壓值。多節點電阻器的電壓源為 VDD18/VDD3V/CP_I，透過控制器 CPRH[1:0]設置選擇不同的電壓源，增加節點電壓的輸出範圍。

遲滯控制器 CPDM[3:0]是與節點選擇器 CPDA[3:0]是聯動的，遲滯控制器 CPDM[3:0]的每一位元對應控制著控制器 CPDA[3:0]的每一位元的遲滯功能的開啟與關閉。當遲滯控制器 CPDM[3:0]的對應位被置<1>，則節點控制器 CPDA[3:0]的對應位就會開啟遲滯功能，且該位的狀態值與比較器的輸出狀態值是一致，即 CPDA[X]=CMPO。這樣就會出項節點選擇器在兩個節點之間來回切換。

'u' 代表不改變

CPDM[3:0]	CMPO	CPDA[3:0]	CPDM[3:0]	CMPO	CPDA[3:0]
	輸出狀態	遲滯切換區間		輸出狀態	遲滯切換區間
0000	0	uuuu	1000	0	0uuu
	1	uuuu		1	1uuu
0001	0	uuu0	1001	0	0uu0
	1	uuu1		1	1uu1
0010	0	uu0u	1010	0	0u0u
	1	uu1u		1	1u1u
0011	0	uu00	1011	0	0u00
	1	uu11		1	1u11
0100	0	u0uu	1100	0	00uu
	1	u1uu		1	11uu
0101	0	u0u0	1101	0	00u0
	1	u1u1		1	11u1
0110	0	u00u	1110	0	000u
	1	u11u		1	111u
0111	0	u000	1111	0	0000
	1	u111		1	1111

表 20-1 遲滯控制 CPDM[3:0]配置與數值

20.1.3 比較器輸出

比較器輸出為數字輸出，且輸出至 IO 引腳 PT1.7，因此比較器的輸出需要設置 IO 作為輸出模式。比較器的輸出可以設置經過 2us 的低通濾波器，消除尖峰脈衝幹擾，控制位 CPDF 被置<1>時，則比較器輸出結果經過 2us 低通濾波器，若 CPDF 被置<0>，則不經過濾波器。比較器的輸出極性可透過控制位 CPOF 設置，當 CPOF 被置<1>，則比較器輸出反相，若 CPOF 被置<0>，則比較器輸出正常。

20.1.4 觸摸按鍵應用

比較器還有特別的功能：觸控按鍵的測量。主要原理是通過多節點電阻器設置比較電壓值輸入至 RLO，利用多節點電阻器提供電壓對觸控按鍵充電，再用觸控按鍵的電荷對負向輸入通道 CH1 的外接參考電容充電，TMB 計數 CH1 的電壓值大於 RLO 端電壓的充電時間，根據時間來區分觸控按鍵被觸摸與無觸摸的狀態。

控制對應觸控按鍵的充電與控制觸控按鍵對參考電容的充電就需要 2 個開關控制，且 2 個開關在任何時候都是保持一個閉合，一個斷開的狀態。比較器內建一個自動切換器來控制 2 個開關的一閉一開的狀態，自動切換器的工作頻率來源於 TMB 的工作時脈，所以此項功能需要打開 TMB 的計數功能及讀取與清零 TMB 的計數暫存器。

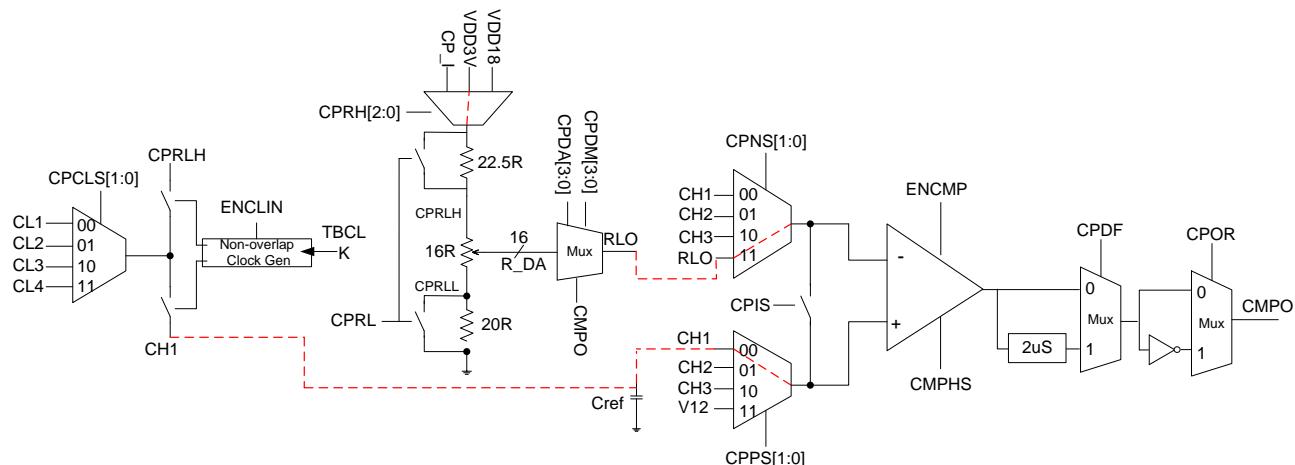


圖 20-4 觸控按鍵連接圖（其中一種配置）

20.1.5 比較器操作初始化

比較器的主要功能是比較輸入信號大小，但是根據不同的模塊組合，產生不同的應用，且要不同的初始化配置。

作為單純的信號比較器：

- ◆ 設置 **CMP** 工作模式，低功耗或正常工作模式。
 - ◆ 選擇輸入通道，包含正向輸入、負向輸入通道；
 - ◆ 若選擇 **RLO** 作為正向輸入通道，還需要設置多節電阻器的參考電壓源及電壓節點；
 - ◆ 使能比較器輸出功能；
 - ◆ 設置輸出是否經過低通濾波及是否輸出是否反相；
 - ◆ 若使用 **CMP** 比較中斷向量，還需要開啟比較器中斷功能；
 - ◆ 使能比較器功能。

觸控按鍵應用初始化：

- ◆ 設置 TMB，設置 TMB 的工作模式為模式 0，且設置觸發計數源為 CMPO，設置工作時脈及溢出值；
 - ◆ 設置 CMP 工作模式，低功耗或正常工作模式。
 - ◆ 選擇輸入通道，包含正向輸入、負向輸入通道；正向為 RLO，負向輸入為 CH1；
 - ◆ 設置多節電阻器的參考電壓源及電壓節點；及電阻器的電阻短路開關；
 - ◆ 使能比較器輸出功能；
 - ◆ 設置輸出是否經過低通濾波及是否輸出是否反相；
 - ◆ 若使用 CMP 比較中斷向量，還需要開啟比較器中斷功能；
 - ◆ 使能比較器功能；

充電前先將觸控按鍵上的電荷和參考電容的電荷釋放

- ◆ 先關閉自動切換器，再關閉電阻器的參考電壓源，使能電阻器的電阻短路開關；使能輸入端短路開關；通過電阻器讓 CH1 參考電容對地放電；
 - ◆ 將觸控按鍵的對應的 IO 引腳設為輸出模式，且輸出狀態 0，使得觸控按鍵對地放電；

再開啟充電功能

- ◆ 斷開輸入端短路開關，及斷開電阻器的電阻短路開關，開啟電阻器參考電壓源；
 - ◆ 關閉觸控按鍵的 IO 輸出模式；
 - ◆ 清零 TMB 的計數暫存器
 - ◆ 使能自動切換功能，並選擇待充電的觸控按鍵
 - ◆ 等待充電完成讀取 TMB 的計數值。

20.2 暫存器位址

CMP Register Address	31	24	23	16	15	8	7	0
CMP Base Address + 0X00 (0X41800)	-		REG0		MASK1		REG1	
CMP Base Address + 0X04 (0X41804)		MASK2		REG2		MASK3		REG3

-保留

20.3 暫存器功能

20.3.1 類比 CMP 暫存器 CMPCR0

CMP Base Address + 0X00 (0X41800) CMPCR0 (CMP Control Register 0)					
Symbol	CMPCR0 (CMP Control Register 0)				
Bit	[31:17]				[16]
名稱	-				CMPO
RW	-				R-0
Bit	[15:08]	[7:6]	[5:4]	[3]	[2:0]
名稱	MASK	CPPS[1:0]	CPNS[1:0]	CPRL	CPRH[2:0]
RW	R0W-0			RW-0	

位元	名稱	描述
Bit[16]	CMPO	比較器比較結果輸入轉出狀態
		0 負向輸入信號 > 正向輸入信號 1 正向輸入信號 > 負向輸入信號

Bit	名稱	描述
Bit[7~6]	CPPS	比較器正向輸入端選擇
		00 CH1
		01 CH2
		10 CH3
		11 V12
Bit[5~4]	CPNS	比較器負向輸入端選擇
		00 CH1
		01 CH2
		10 CH3
		11 RLO
Bit[3]	CPRL	比較器內建階梯電阻低節短路開關控制
		0 短路開關斷開，不短路低節電阻
		1 短路開關閉合，短路低節電阻，
Bit[2~0]	CPRH	比較器內建階梯電阻器電壓源選擇
		000 關閉，無電壓提供，處於高阻態
		001 CP_I，與電荷泵（Charge Pump）輸入電壓源一致
		010 VDD3V 晶片工作電壓源
		100 VDD18 (晶片內部 LDO 穩壓產生的 1.8V 數字電壓源

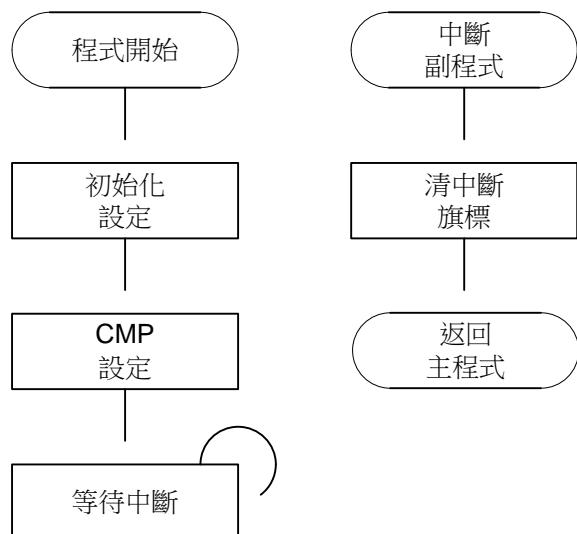
20.3.2 類比 CMP 暫存器 CMPCR1

CMP Base Address + 0X04 (0X41804)							
Symbol	CMPCR1 (CMP Control Register 1)						
Bit	[31:24]	[23:20]			[19:16]		
名稱	MASK	CPDM[3:0]			CPDA[3:0]		
RW	R0W-0				RW-0		
Bit	[15:08]	[7]	[6]	[5]	[4]	[3:2]	[1]
名稱	MASK	CPOR	CMPHS	CPIS	ENCLIN	CPCLS[1:0]	CPDF
RW	R0W-0				RW-0		

位元	名稱	描述
Bit[23]	CPDM[3]	CPDA[3]輸出遲滯開啟控制，且 CPDA[3]的值受 CMPO 控制，並且保持一致
		0 關閉
		1 開啟，CPDA[3]=CMPO
Bit[22]	CPDM[2]	CPDA[2]輸出遲滯開啟控制，且 CPDA[2]的值受 CMPO 控制，並且保持一致
		0 關閉
		1 開啟，CPDA[2]=CMPO
Bit[21]	CPDM[1]	CPDA[1]輸出遲滯開啟控制，且 CPDA[1]的值受 CMPO 控制，並且保持一致
		0 關閉
		1 開啟，CPDA[1]=CMPO
Bit[20]	CPDM[0]	CPDA[0]輸出遲滯開啟控制，且 CPDA[0]的值受 CMPO 控制，並且保持一致
		0 關閉
		1 開啟，CPDA[0]=CMPO
Bit[19~16]	CPDA	比較器內建多點電阻器的分壓節點設置.
		0000 0
		0001 1/16 (CPRLH – CPRLL)
		0010 2/16 (CPRLH – CPRLL)
		0011 3/16 (CPRLH – CPRLL)
		0100 4/16 (CPRLH – CPRLL)
		0101 5/16 (CPRLH – CPRLL)
		0110 6/16 (CPRLH – CPRLL)
		0111 7/16 (CPRLH – CPRLL)
		1000 8/16 (CPRLH – CPRLL)
		1001 9/16 (CPRLH – CPRLL)
		1010 10/16 (CPRLH – CPRLL)
		1011 11/16 (CPRLH – CPRLL)
		1100 12/16 (CPRLH – CPRLL)
		1101 13/16 (CPRLH – CPRLL)
		1110 14/16 (CPRLH – CPRLL)
		1111 15/16 (CPRLH – CPRLL)

位元	名稱	描述
Bit[7]	CPOR	比較器數字輸出相位控制
		0 正常輸出 1 反相輸出
Bit[6]	CMPHS	比較器高速模式開啟控制
		0 低功耗模式 1 高速模式
Bit[5]	CPIS	比較器輸入端短路開關控制
		0 短路開關斷開 1 短路開關閉合
Bit[4]	ENCLIN	比較器內建自動切換 (non-overlapp) 功能開啟控制. 自動切換器時鐘源是 TBCLK
		0 關閉 1 開啟，使用 TBCLK 作為自動切換器的驅動時鐘源
Bit[3-2]	CPCLS	比較器觸摸鍵功能正向輸入源選擇
		00 CL1 01 CL2 10 CL3 11 CL4
		比較器輸出低通濾波器開啟控制
		0 關閉，比較器輸出不經過 2us 低通濾波器 1 開啟，比較器輸出經過 2us 低通濾波器
Bit[0]	ENCMP	比較器功能開啟控制
		0 關閉(輸出狀態為 0) 1 開啟

20.4 範常式式流程

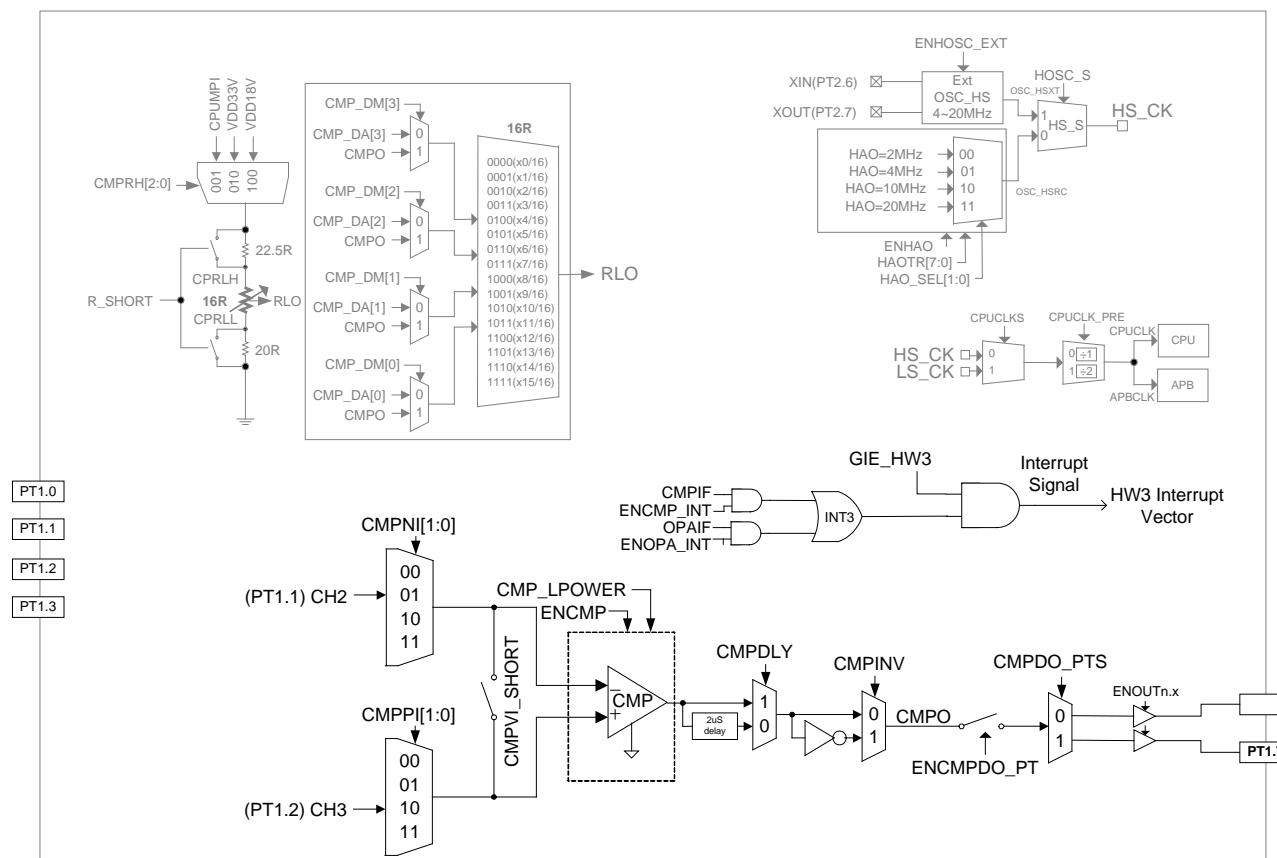


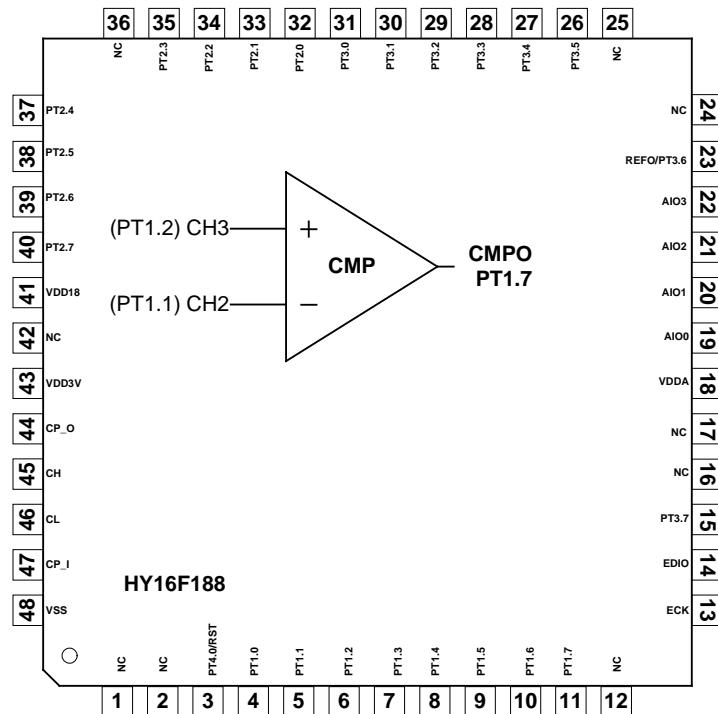
20.5 範常式式功能

範例名稱: CMP 使用方式說明

範例說明:

- (1)透過適當的 CMP 暫存器與 IO 設定。
- (2)CH3(PT1.2)接到 CMP 的正端。
- (3)CH2(PT1.1)接到 CMP 的負端。
- (4)若 CH3 正端電壓大於 CH2 負端電壓，則 CMPO 數位輸出 High。
- (5)若 CH3 正端電壓小於 CH2 負端電壓，則 CMPO 數位輸出 Low。



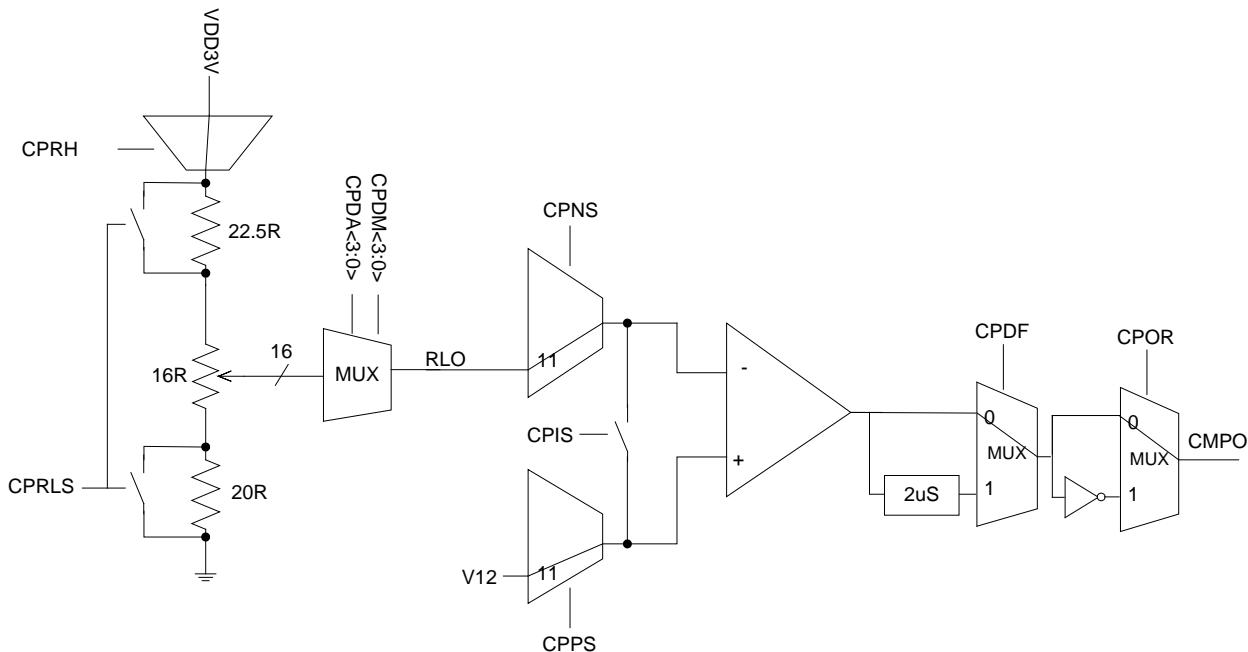


20.6 範常式式說明

00	
01	#include "HY16F18X.h"
02	
03	int main(void)
04	{
05	DrvGPIO_Open(E_PT1,0X80,E_IO_OUTPUT); //PT1.7 set Output
06	DrvCMP_PInput(2); //CMP positive input CH3
07	DrvCMP_NInput(1); //CMP negative input CH2
08	
09	DrvCMP_Enable(); //CMP enable
10	DrvCMP_OutputPinEnable(0); //Enable CMP digital output to port
11	//0:PT1.7
12	DrvCMP_ClearIntFlag(); //Clear CMP interrupt flag
13	DrvCMP_EnableInt(); //Enable CMP Interrupt
14	
15	
16	
17	SYS_EnableGIE(7); //Enable GIE (Global Interrupt Enable)
18	while(1); //while loop
19	}
20	void HW3_ISR(void)
21	{
22	DrvCMP_ClearIntFlag(); //Clear CMP interrupt flag
23	}
24	

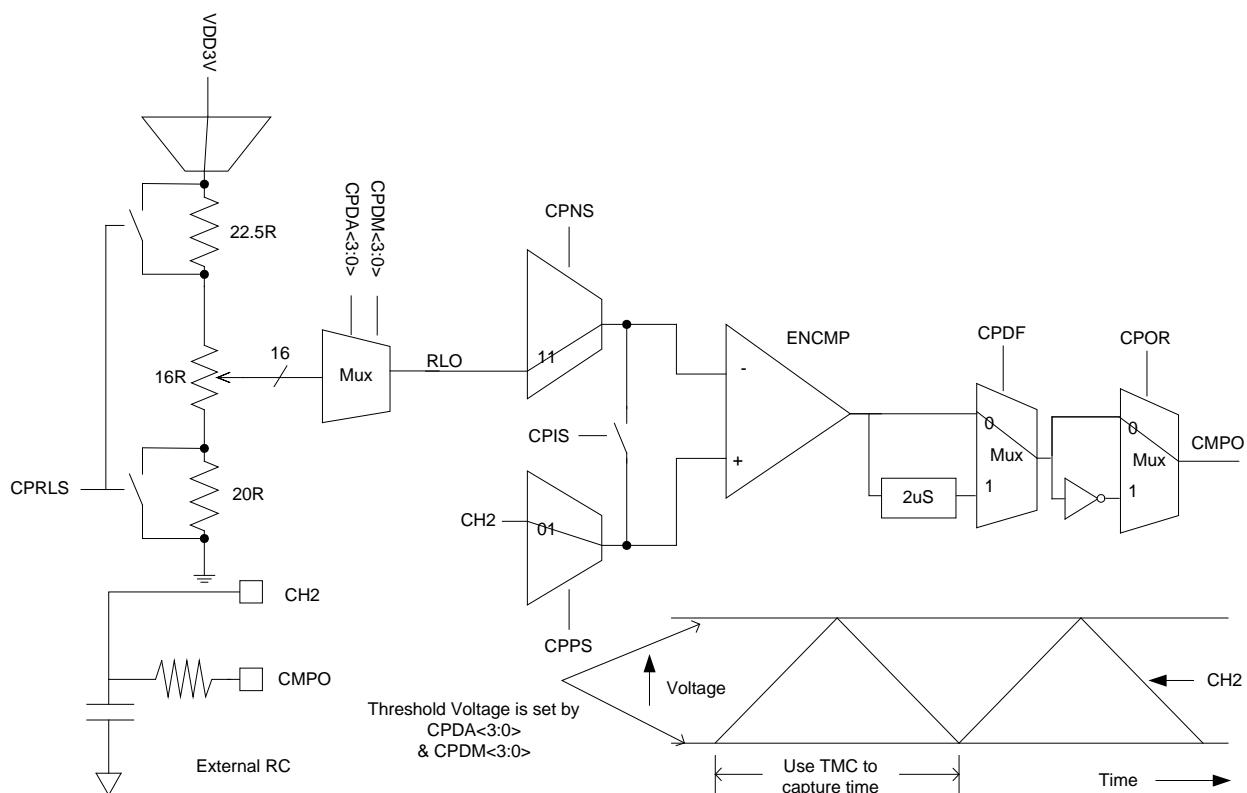
20.7 應用電路功能 01

用來當作一個低電壓偵測器。



20.8 應用電路功能 02

用來做電容測量。



21.串列通信 SPI

21.1 整體總說明

HY16F18X 有 1 個串列週邊介面 Serial Peripheral Interface (SPI)。這個 SPI 使用同步串列數據通訊協定，並使用全雙工模式來運作。它與 4-線雙向介面做通訊，且可在主/從端模式下運作。在主端模式下，它有數個組態來執行不同的從端裝置。

功能：

- (1)全雙工同步傳輸。
- (2)支援主端模式或從端模式運作。
- (3)支援 MSB 或 LSB 最先傳輸。
- (4)傳輸框格式為 4~32 位元可程式化設定 BIT 長度。
- (5)高速 SPI 汇流排忙碌狀態旗標。
- (6)可編程時鐘脈衝率。
- (7)支援高/低電位從機端選擇。
- (8)可編程時鐘極性及相位。

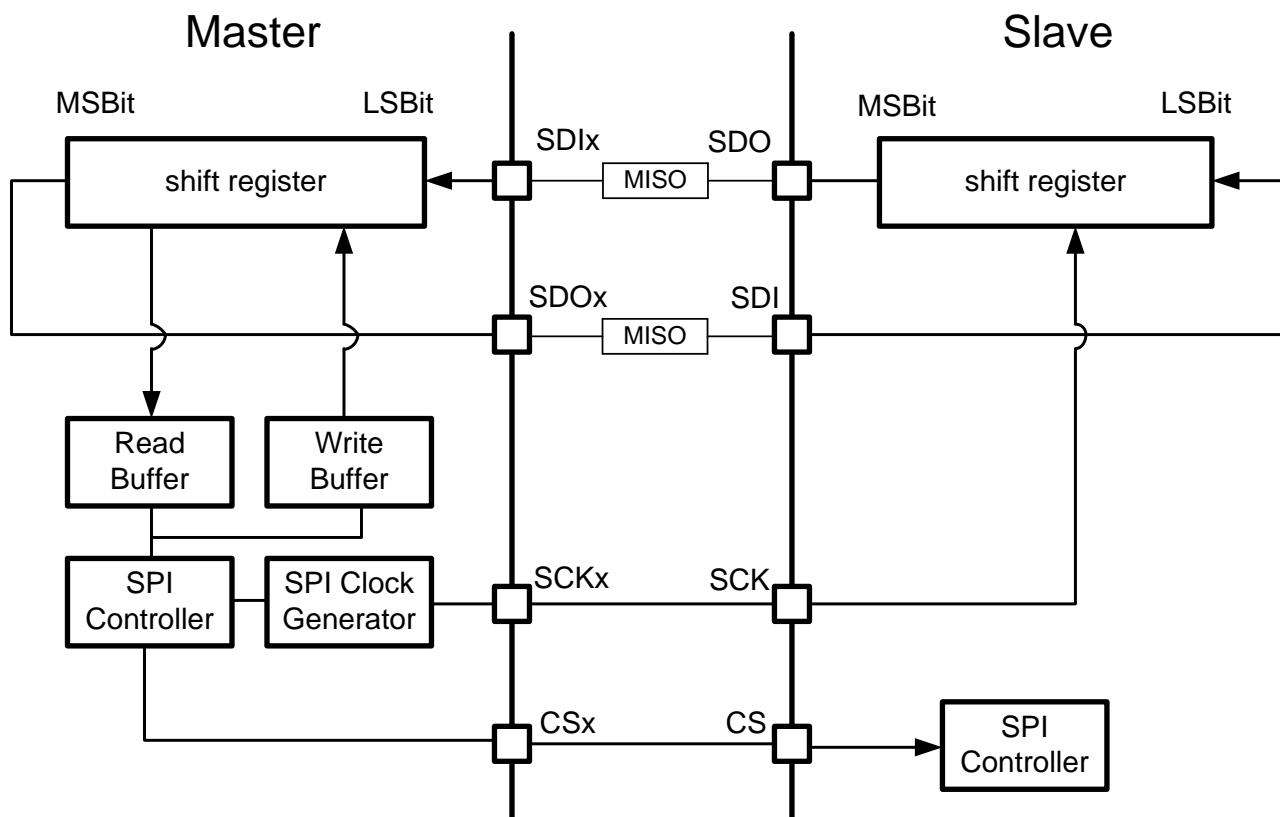


圖 21-1 串列通訊 SPI 架構圖

這 MISO 接腳是主端裝置的輸入和從端裝置的輸出。MOSI 接腳是主端裝置的輸出和從端裝置的輸入。SCK 接腳是來自主端裝置的串口通訊時鐘輸出。

這個 CS 接腳是來自主端裝置的晶片選擇，以啟動從端裝置的 SPI 通訊。
這些主端裝置或從端裝置的 MOSI/MISO/SCK/CS 接腳被連接在一起以便執行工作。

通訊永遠是由主端裝置所啟動。主端裝置經由 MOSI 接腳傳送資料給從端裝置，而從端裝置經由 MISO 接腳回應。所以，這是全雙工通訊，數據進出同步，使用相同的時鐘源。

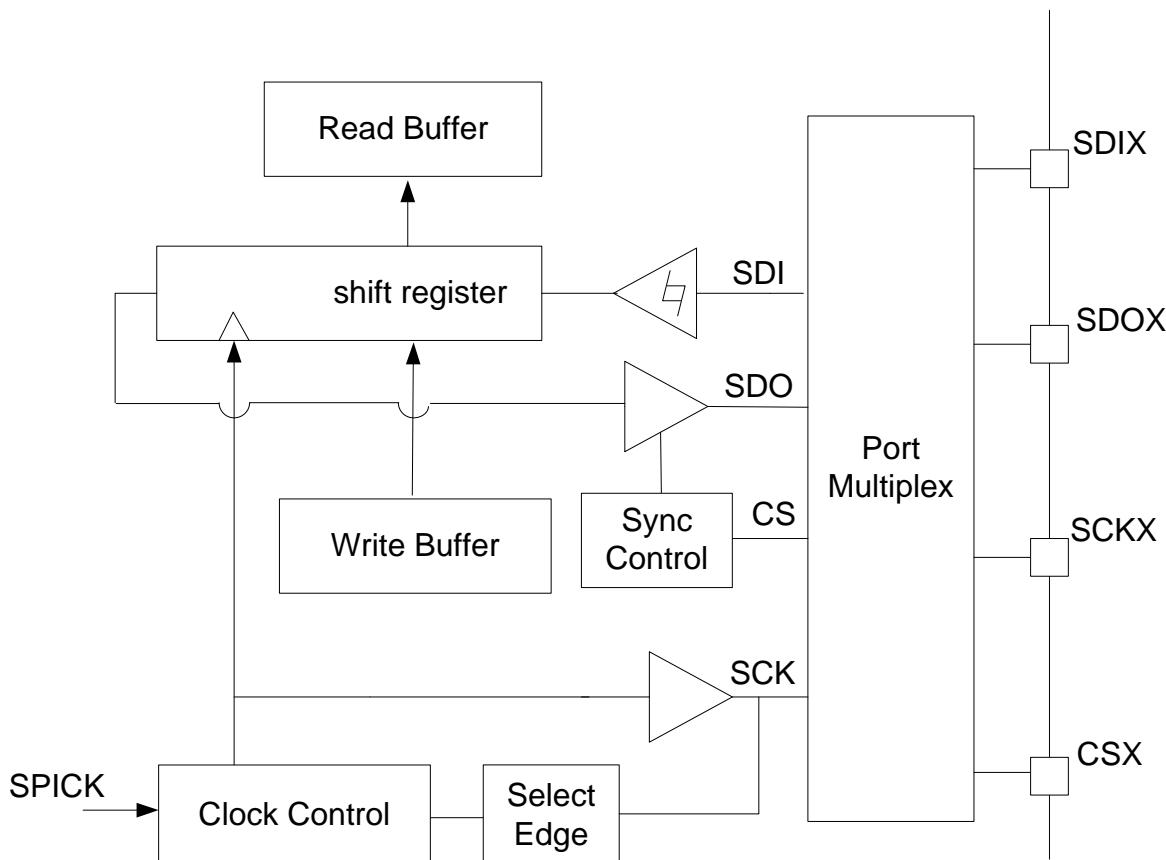


圖 21-2 SPI 的 IO 引腳圖

功能描述:I/O 接腳設定:

這些 SPI 接腳可用程式編寫給不同的 I/O 接腳。

時鐘相位與時鐘極性:

可用軟體來組成四個不同的時序方案，並由 CPOL 和 CPHA 暫存器來控制。

CPOL(時鐘極性)是在沒有資料傳輸的情況下，控制時鐘的穩定狀態值。

它可用在主端模式和從端模式中。如果 CPOL 為 1(高電位)，則當 SPI 處於閒置模式時，

SCK 就會是 1。另一方面，如果 CPOL 是 0(低電位)，則當 SPI 處於閒置狀態時，SCK 就會是 0(低電位)。

CPHA(時鐘相位)控制 SCK 的數據時脈沿捕捉。如果 CPHA 為 1(高電位)，

SCK 接腳的第二個時脈沿(如果 CPOL 為 1 則是下降沿；為 0 則是上升沿)就會捕捉到 MSB 數據。

這個數據會被拴在第二個 SCK 時脈沿。另一方面，如果 CPHA 是 0(低電位)，

SCK 接腳上的第一個時脈沿(如果 CPOL 為 1 則是上升沿；為 0 則是下降沿)就會捕捉到 MSB 數據。

這個數據會被拴在第一個 SCK 時脈沿。因此，CPOL 和 CPHA 暫存器的組合控制數據捕捉和時脈沿輸出。

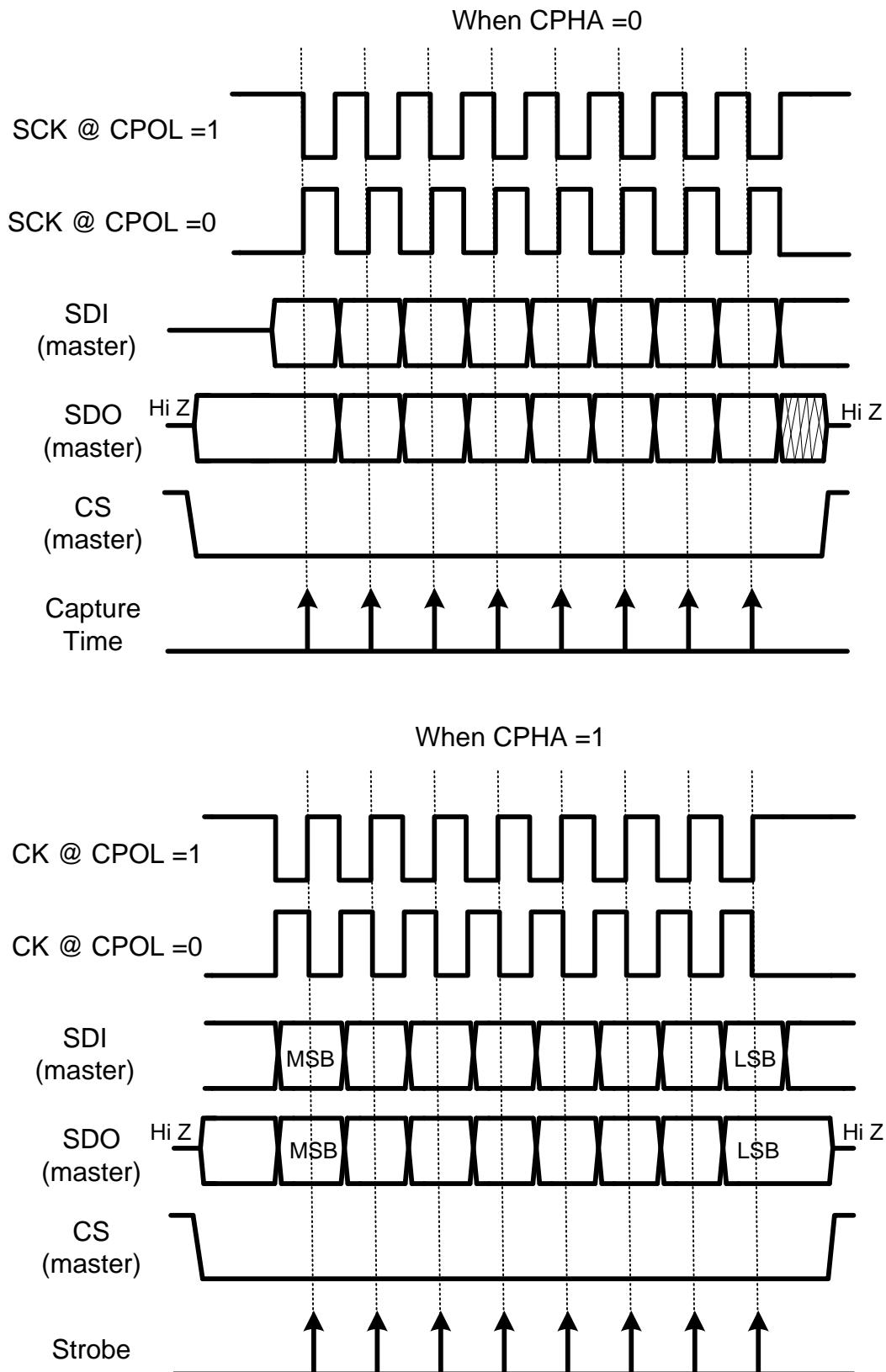


圖 21-3 SPI 的主動模式時序圖

注意事項：SPI Interface 工作於 Master Mode 時，SCK 工作頻率為 SPICK/2。

SPI 控制暫存器 1 (SPI Control Register 1,):

(BL 控制位) 數據框格式：

用來傳送和接收的交易字(transaction word)位元長度可在暫存器的控制位 BL 0x40F04[4:0]中予以定義。最小的位元長度為 4 個位元，最大長度為 32 個位元。數據在移位暫存器中，其傳輸格式可為 MSB 位元先被傳送或 LSB 位元先被傳送，且由暫存器的控制位 LBF 0x40F04[18]所定義。如果 LBF 是<0>，則數據傳輸格式為在移位暫存器中的 MSB 位元先被傳輸。然後，第二個 MSB 被傳送，最後才是 LSB 位元。如果 LBF 是<1>，則數據傳輸格式為 LSB 位元先被傳送。

(CSL 控制位) 從端裝置晶片選擇級別：

適用於 SPI 四線式主端與從端模式。這個 CS 接腳可被定義為 0 或 1(低電位或高電位)以啟動從端裝置。這是由暫存器的控制位 CSL 0x40F04[19]所控制。如果在主端裝置中的 CSL 是<0>，則 CS 接腳就會輸出 0(低電位)以啟動從端裝置。另一方面，如果主端裝置中的 CSL 為<1>，則 CS 接腳就會輸出 1(高電位)以啟動從端裝置。如果在從端裝置中的 CSL 為<0>，則從端裝置會在收到一個 CS 的輸入 0(低電位)後被啟動。另一方面，如果在從端裝置中的 CSL 為<1>，則從端裝置會在收到一個 CS 的輸入 1(高電位)後被啟動。

注意事項：SPI Interface 工作於四線式 Master mode 時，CS 腳位的控制是屬於半自動控制的方式，例如：當 CSL 設定為<1>時，此時 CS 腳位會被拉到低電位，當 SPI Master 要寫資料到終端 SPI Device 的時候，CS 腳位會自動拉到高電位，待資料傳送完成之後，會再自動回覆到低電位，即是 Idle 時候為 Low，Active 時候才為 High。

(CSO 控制位)

這個控制位只有在三線式 SPI Slave mode 才會使用到。此腳位的功能為晶片內部喚醒 CS 信號模擬器控制。當 SPI Master 要寫資料給 SPI Slave，SPI Slave 要接收資料之前，則需要先設定 CSO=<0>才能正確接收資料。當資料接收完成後，要把資料從 RXB Buffer 讀出前，則需要先設定 CSO=<1>後，才能正確讀取所接收資料，在資料讀取後則需要設定 CSO=<0>才能準備接收下一筆資料。而當 SPI Slave 要回傳資料給 SPI Master 的時候，同樣需要先設定 CSO=<1>後，將需要傳送資料寫入 TXB Buffer 之後，再設定回 CSO=<0>，這樣才可以將資料回傳給 Master。

注意事項：當使用三線式的 SPI 傳輸時，如果 SPI Slave 端已經先完成了初始化，並且設定 CSO=0，此時如果 SPI Master 端才上電做初始化的設定，會有可能造成 SPI Master 在初始化的過程中因為 GPIO 的初始變化，造成 SPI Slave 端誤判，接收到一筆錯誤的資料，因而引起 SPI Slave 端有第一筆資料誤接收的可能性。因為三線式的 SPI 傳輸中，本身並沒有 CS 腳位做同步的動作，使用者應該在 SPI Master 與 Slave 端初始化過程中建立握手協定(Handshake Protocol)，在確認雙方都初始化完成之後才開始做資料傳輸。

SPI 控制暫存器 0 (SPI Control Register 0) :

(OVF 控制位):

OVF 是 SPI 的溢位旗標。當傳輸期間有額外的 SCK 時脈沿輸入時，它就會是在高電位(1)。例如，如果一個交易字(Word)的位元長度是 16 個位元，且在 CS 改變為高電位前(在此例，CSL 為<0>)，有 17 個來自主端裝置的時脈緣，而當 OVF 收到第 17 個時脈沿時，它的值是 1。這表示此一傳輸有錯誤發生。如果第 17 個時脈沿已發生，代表第一個被傳輸的數據遺失了。

(ABF 控制位):

ABF 是 SPI 中止旗標，只用在從端模式中。在傳輸期間，當 SCK 時脈沿輸入不足時，它就會是在高電位(1)。例如，如果一個交易字(word)的位元長度是 16 個位元，有 15 個來自主端裝置的時脈沿，且 CS 改變為高電位(在此例，CSL 為<0>)，則 ABF 為<1>。這表示此一傳輸有錯誤發生。交易未完成，傳輸的數據被更新到讀取暫存器中。傳輸被中止，且會遺失掉。

(BUF 控制位):

BUF 是 SPI 的忙碌旗標。當 SPI 在傳輸或接受數據時，它是在高電位(1)。在主端裝置中，當 SPI 開始數據傳輸時，它是在高電位(1)。一旦 SPI 停止數據傳輸或傳輸字已完成傳輸時，它就會自動被清除。在從端模式中，當 SPI 準備好要與主端裝置通訊時，則 BUF 是<1>。一旦 SPI 停止數據傳輸或傳輸字已完成傳輸時，它會自動被清除。

SPI 中斷旗標控制位:

- (1)STxIF:旗標 STxIF 是 SPI 的傳輸中斷(interrupt)。當寫入暫存器被載入到移位暫存器時，它會被設定為<1>。
(2)SRxIF:旗標 SRxIF 是 SPI 的接收中斷。當移位暫存器被載入到讀取暫存器時，它會被設定為<1>。

21.2 暫存器位址

SPI Register Address	31	24	23	16	15	8	7	0
SPI Base Address + 0X00(0X40F00)	SPIC2M		SPIC2		SPIC1M		SPIC1	
SPI Base Address + 0X04(0X40F04)	SPIC0M		SPIC0		-		BL	
SPI Base Address + 0X08(0X40F08)	RXB3		RXB2		RXB1		RXB0	
SPI Base Address + 0X0C(0X40F0C)	TXB3		TXB2		TXB1		TXB0	

21.3 暫存器功能

21.3.1 通信 SPI 暫存器 SPICR0

SPI Base Address + 0X00 (0X40F00)									
Symbol	SPICR0 (SPI Control Register 0)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名稱	MASK	-	RXF	OVF	ABF	BUF	DCF	TXBF	RXBF
RW	R0W-0	-	R-0	RW0-0				R-0	
Bit	[15:08]	[7:4]			[3]	[2]	[1]	[0]	
名稱	MASK	-			CPHA	CPOL	M/S	EN	
RW	R0W-0	-						RW-0	

位元	名稱	描述
Bit[22]	RXF	接收(Rx)暫存器值更新旗標
		0 正常
		1 接收(RX)暫存器的數據有更新，此時不能讀取接收暫存器
Bit[21]	OVF	SPI 總線的數據長度過長旗標
		0 正常
		1 接收到的數據長度大於自行設置的數據長度 BL[4:0].寫入 0 可清除 OVF 旗標
Bit[20]	ABF	SPI 總線的數據長度偏少旗標
		0 正常
		1 接收到的數據長度小於自行設置的數據長度 BL[4:0].寫入 0 可清除 ABF 旗標
Bit[19]	BUF	SPI 總線繁忙旗標
		0 SPI 總線介面空閒待機狀態
		1 SPI 總線介面繁忙狀態
Bit[18]	DCF	數據丟失旗標
		0 正常
		1 接收暫存器已滿而繼續接收數據，舊的數據會丟失，讀取接收暫存器可清零該位
Bit[17]	TXBF	TX 發送暫存器已滿旗標
		0 TX 發送暫存器為空，可發送數據
		1 TX 發送暫存器已滿，繼續寫入數據將會覆蓋舊的數據
Bit[16]	RXBF	Rx 接收暫存器已滿旗標
		0 RX 接收暫存器為空
		1 RX 接收暫存器已滿(讀取接收暫存器可以清零該位)

位元	名稱	描述
Bit[03]	CPHA	SPI 總線捕捉數據的時脈相位設置
		0 在 SCK 第一個時脈沿捕捉數據
		1 在 SCK 第二個時脈沿捕捉數據
Bit[02]	CPOL	SPI 總線工作頻率極性控制
		0 SCK 低電位為空閒
		1 SCK 高電位為空閒
Bit[01]	M/S	SPI 工作模式設置
		0 被動模式
		1 主動模式
Bit[00]	EN	SPI 功能開啟控制
		0 關閉
		1 開啟

21.3.2 通信 SPI 暫存器 SPICR1

SPI Base Address + 0X04 (0X40F04)						
Symbol	SPI CR1(SPI Control Register 1)					
Bit	[31:24]	[23:21]	[20]	[19]	[18]	[17:16]
名稱	MASK	-	CSO	CSL	LBF	MD
RW	R0W-0	-			RW-0	
Bit	[15:5]			[4:0]		
名稱	-			BL		
RW	-			RW-0		

位元	名稱	描述
Bit[20]	CSO	晶片內部喚醒 (CS) 信號模擬器控制，適用於 3 線模式
		0 CS 信號模擬器工作 1 CS 信號模擬器待機
Bit[19]	CSL	CS 信號極性設置，用於啟動器件
		0 低電位啟動 1 高電位啟動
Bit[18]	LBF	數據發送順序
		0 先發送高字節 1 先發送低字節
Bit[17-16]	MD	SPI 介面工作模式設置
		00 SPI 標準 4 線通訊介面模式
		01 SPI 通用 3 線介面模式
		10 TI 模式
		11 TI 模式

位元	名稱	描述	
Bit[4~0]	SPIBL	SPI 發送一個字的數據長度設置	
		00000	8 bits length
		00001	16 bits length
		00010	24 bits length
		00011	4 bits length
		00100	5 bits length
		00101	6 bits length
		00110	7 bits length
		00111	8 bits length
		01000	9 bits length
		01001	10 bits length
		01010	11 bits length
		01011	12 bits length
		01100	13 bits length
		01101	14 bits length
		01110	15 bits length
		01111	16 bits length
		10000	17 bits length
		10001	18 bits length
		10010	19 bits length
		10011	20 bits length
		10100	21 bits length
		10101	22 bits length
		10110	23 bits length
		10111	24 bits length
		11000	25 bits length
		11001	26 bits length
		11010	27 bits length
		11011	28 bits length
		11100	29 bits length
		11101	30 bits length
		11110	31 bits length
11111	32 bits length		

當 MD 被設定為 3 線模式時，原本的 CS 接腳就會變成 GPIO 模式。
你可用使用者隨機波浪型來產生所需的或使用在其他的應用上，
但仍需以 CS SPIC 介面做為同步訊號。

主端模式下的 3 線模式：
內部的同步電路本身沒有任何特別的處理。

從端模式下的 3 線模式：
需要使用 CSO 位元元類比 SPI 匯流排上的 CS 訊號，
並使用 CSO CS 訊號來模擬，且必須類似標準的 4 線模式。

21.3.3 通信 SPI 暫存器 SPICR2

SPI Base Address + 0X08 (0X40F08)	
Symbol	SPICR2 (SPI Control Register2)
Bit	[31:16]
名稱	RXB31-16
RW	R-X
Bit	[15:0]
名稱	RXB15-0
RW	RW-X

位元	名稱	描述
Bit[31~0]	SPIRB	SPIRB[31:0] 是 32 位的接收暫存器

以 LBF 位元來設定 LSB 或 MSB 先被傳輸。
當 LSB 被設定為先傳輸，就會影響到數據儲存的位置，
RXB 有效數據會，被向右對齊。例如，BL 被設定為 8 位元元模式時，
接收到的數據就會在 RXB [7:0]；設定為 9 位元元模式時，
接收到的數據就會在 RXB [8:0]，以此類推。當設定 MSB 先被傳輸時，
RXB 有效數據會被向左對齊。例如，BL 被設定為 8 位元元模式時，
接收到的數據就會在 RXB [31:24]；設定為 9 位元元模式時，
接收到的數據就會在 RXB [31:23]，以此類推。

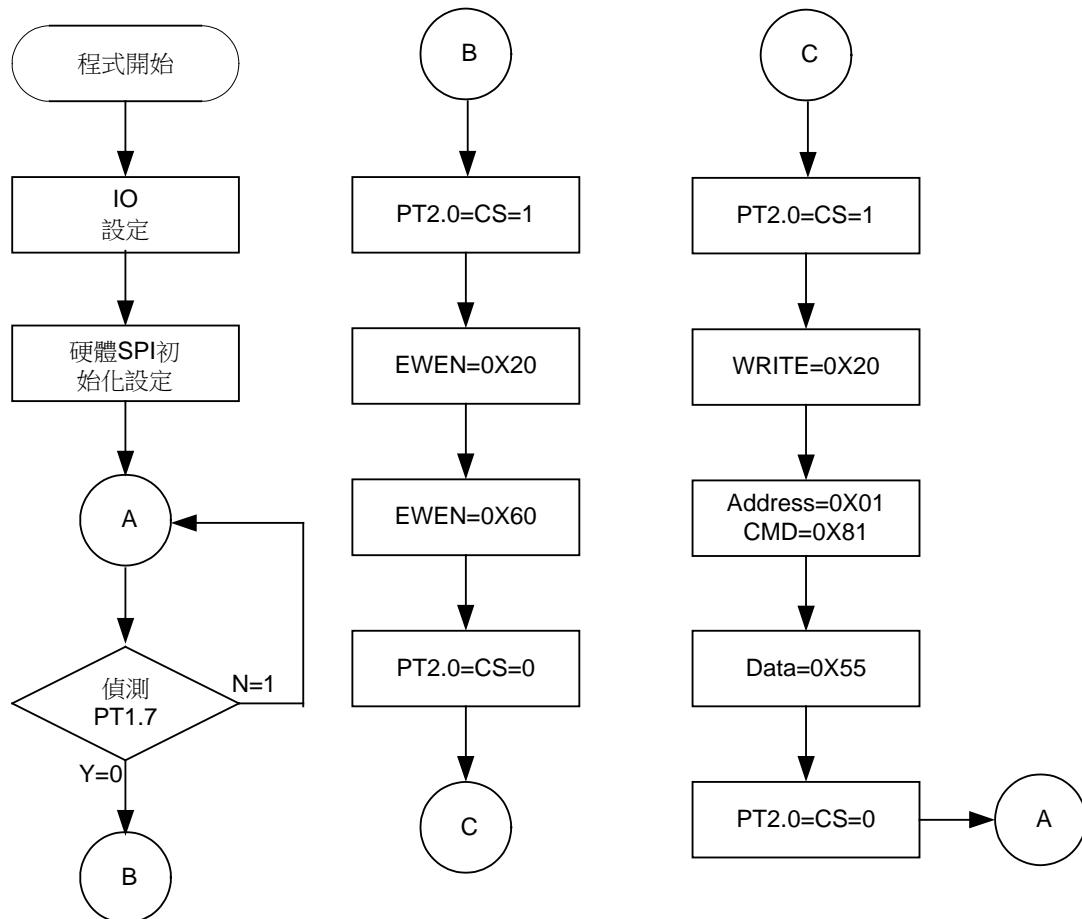
25.3.4 通信 SPI 暫存器 SPICR3

SPI Base Address + 0X0C (0X40F0C)	
Symbol	SPICR3 (SPI Control Register 3)
Bit	[31:16]
名稱	TXB31-16
RW	R-X
Bit	[15:0]
名稱	TXB15-0
RW	RW-X

位元	名稱	描述
Bit[31~0]	SPITB	SPITB[31:0] 是 32 位的發送暫存器

以 LBF 位元來設定 LSB 或 MSB 先被傳輸。
當 LSB 被設定為先傳輸，就會影響到數據儲存的位置，
TXB 有效數據會被向右對齊。例如，BL 被設定為 8 位元元模式時，
接收到的數據就會儲存在 TXB [7:0]；設定為 9 位元元模式時，
接收到的數據就應儲存在 TXB [8:0]，以此類推。當設定 MSB 先被傳輸時，
TXB 有效數據會被向左對齊。例如，BL 被設定為 8 位元元模式時，
接收到的數據就會儲存在 TXB [31:24]；設定為 9 位元元模式時，
接收到的數據就會儲存在 TXB [31:23]，以此類推。

21.4 範常式式流程

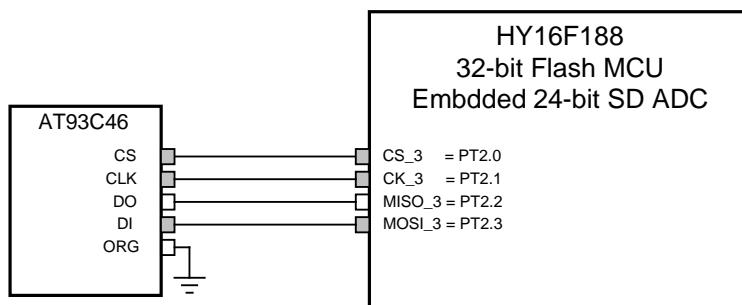


21.5 範常式式功能

範例名稱:通信 SPI 中斷設定

範例說明:

- (1)透過 SPI 腳位與 SPI 暫存器設定
- (2)測試硬體 SPI 對 AT93C46 寫入功能



21.6 範常式式說明

(主程式)

00		
01	#include "HY16F18X.h"	
02		
03		
04	volatile unsigned int i,a;	
05		
06	void InitalSPI32(void);	
07	void Delay(unsigned int num);	
08		
09	int main(void)	
11	{	
12	SPI32_INI();	
13		
14	while(1)	
15	{	
16	i=DrvGPIO_GetBit(E_PT1,7);	
17	while(i==0)	
18	{	
19		//Write enable
20	DrvGPIO_SetBit(E_PT2,0);	//PT2.0 CS=1
21	DrvSPI32_Write(0X02000000);	//AT93C56=0X09
22	Delay(0X20);	
23	DrvSPI32_Write(0X60000000);	//AT93C56=0X80
24	Delay(0X50);	
25	DrvGPIO_ClrBit(E_PT2,0);	//PT2.0 CS=0
26		
27	Delay(0X20);	
28		
29		
30		//Write CMD
31	DrvGPIO_SetBit(E_PT2,0);	//PT2.0 CS=1
32	DrvSPI32_Write(0X02000000);	//AT93C56=0X0A
33	Delay(0X20);	
34	DrvSPI32_Write(0X81000000);	//AT93C56 Address 0X01
35	Delay(0X20);	
36	DrvSPI32_Write(0X55000000);	//AT93C56 Data 0X55
37	Delay(0X50);	
38	DrvGPIO_ClrBit(E_PT2,0);	//PT2.0 CS=0
39		
40	Delay(0X20);	
41	i=DrvGPIO_GetBit(E_PT1,7);	
42	}	
43	Delay(0X200);	
44	}	
45	return 0;	
46	}	
47		
48		

(副程式)

00		
01	void HW0_ISR(void)	
02	{	
03	int_00=0XFF00FF00;	
04	}	
05		
06	void InitalSPI32(void)	
07	{	
08	//Set SPI input pin	
09	DrvGPIO_Open(E_PT2,0X0B,E_IO_OUTPUT);	//CS=PT2.0 output
11	DrvGPIO_Open(E_PT2,0X04,E_IO_INPUT);	
12	DrvGPIO_SetPortBits(E_PT2,0X01);	
13		
14	pio_44=0XFF50;	
15	spi_00=0XFF00FF03;	
16	spi_04=0XFF010007;	
17	clk_0c=0XFF00FF0C;	
18	}	
19		
20		
21	void Delay(unsigned int num)	//Software Delay Subroutines
22	{	
23	volatile unsigned int d;	
24	for(d=0;d<=num;d++);asm("NOP");	
25	}	
26		

22.非同步串列通訊 EUART

22.1 整體總說明

晶片嵌入一個增強型的 UART(Enhanced Universal Asynchronous Receiver Transmit)管理。這個 EUART 的週邊通常稱作串列通訊介面(SCI)。EUART 可設定為全雙工異步系統，其週邊通訊裝置包括 LCD/LED 螢幕終端機和 PC/NB/Tablet/Smart Phone。它也可被設定為半雙工同步系統，其週邊通訊裝置則包括 ADC 或 DAC 整合電路、串列 EEPROM/Flash 等。加強型的 EUART 具有額外的特性，包括數據框錯誤偵測和自動位址辨識。數據框錯誤偵測可以決定一個數據框是否為有效或是否未經過框停止位元。自動位址辨識功能可將位址框內容與單晶片位址做比較；而串列中斷只能在這兩者符合時才能被產生。

22.1.1 傳輸串列傳輸速率

Baud Rate Generator 是一個專用的 13bit 發生器，支援 EUART 的非同步模式。下表是串列串列傳輸速率的計算公式，但是僅適用於主控模式。且在給定目標串列傳輸速率及工作頻率為 OSC HAO 的情況下，可以使用下表的公式來計算 Baud Rate Generator 的近似整數值，從而可以確定串列傳輸串列傳輸速率誤差。且建議在切換工作頻率後，需要重置串列傳輸速率或使用自動串列傳輸速率功能，重新校正 Baud Rate Generator 的值。

Baud Rate Generator/EUART MODE	串列傳輸串列傳輸速率計算公式
13 bit/非同步	$OSC\ HAO \div [4x(n+1)]$
OSC HAO=工作頻率; n=BRGRH:BRGRL 暫存器的值；	

例如：工作在非同步模式下，其工作頻率為 OSC HAO(4MHZ),而目標串列傳輸速率 9600bps，可計算 Baud Rate Generator 的值。

$$\begin{aligned} \text{根據公式: Baud Rate Generator} &= ((OSC_HAO \div \text{目標串列傳輸速率}) \div 4) - 1 \\ &= ((4000000 \div 9600) \div 4) - 1 \\ &= 103.1667 \\ &\approx 103 \end{aligned}$$

而根據上述計算 Baud Rate Generator 值來計算串列傳輸速率值：串列傳輸速率 = $4000000 \div (4x(103+1)) = 9615.38$;所以存在一定誤差，該誤差的計算方式為：

$$\begin{aligned} \text{誤差率} &= (\text{實際計算串列傳輸速率} - \text{目標串列傳輸速率}) / \text{目標串列傳輸速率} \\ &= (9615 - 9600) / 9600 \\ &= 0.16\% \end{aligned}$$

22.1.2 通訊 IO 引腳

EUART 通訊總線只用兩根線 TX/RX，晶片為 EUART 模塊配置了 8 組通訊 IO 引腳(每一組包含 TX/RX 線，方便用戶在使用上設計的自由性。但這個是 IO 口的複用功能，透過控制器 PTUR[2:0]/PTURE 可方便的選擇及開啟 EUART 的通訊 IO 引腳，因而在使用 EUART 功能同時，需要先開啟 IO 通訊引腳，且對應的 IO 引腳需要被設置為輸入或輸出模式。EUART 通訊 IO 引腳分佈如下表所示。

PTUR[2:0]	PTURE	TX	RX	PTUR[2:0]	PTURE	TX	RX
000	1	PT1.0	PT1.1	100	1	PT2.0	PT2.1
001	1	PT1.2	PT1.3	101	1	PT2.2	PT2.3
010	1	PT1.4	PT1.5	110	1	PT2.4	PT2.5
011	1	PT1.6	PT1.7	111	1	PT2.6	PT2.7

表 22-1 EUART 通訊 IO 引腳分佈

22.1.3 自動串列傳輸速率功能與設置

EUART 模組支援自動檢測和校正串列傳輸速率的功能，稱之為自動串列傳輸速率功能。

其 Auto-baud rate 設置流程如下：

1. UART 初始化設置：包含 UART TX, RX Port 設定.TX 和 RX 對應到的 GPIO 腳位需要設置對應 TX 為 Output 和 RX 為 Input, 並且設置 ENSP=1b。
2. 設置 WUE=0b，並將自動串列傳輸速率控制位 ENABD=1b 才能啟動自動串列傳輸速率功能。
3. 做自動串列傳輸速率功能之前，先清除 Baud Rate Generator 暫存器內容，並且清除 RX Data Buffer 與 RX Flag 設置，清除完後，等待接收資料 0x55。
4. 開始進行自動串列傳輸速率檢測功能（接收資料為 0x55），在自動檢測和校正完成後便將計算結果寫入 BRGRH[4:0]與 BRGRL[7:0]寄存器。
5. 當 Baud Rate Generator 寄存器計算滿出時即其內容由 0FFFH 到 0000H 產生溢出，則自動串列傳輸速率溢出旗標位 ABDOVF 被置 1，使用者可以利用軟體方式將 ABDOVF 置 0 或透過設置 ENABD=0 使 ABDOVF=0。在 ABDOVF=1 後，使能位元 ENABD 的狀態仍會保持為 1。
6. 完成 Auto-baud rate 設置。最後建議可在做完 Auto-baud rate 之後，增加 Hand hark process，目的是確認 auto-baud rate 得正確性。

22.1.4 EUART 非同步模式

此模式使用標準的‘不歸零’(Non-Return-To-Zero, NRZ)格式，即是 1 個起始位加上 8 bit 或 9 bit 資料，最後為 1 bit 停止位元，最常用的資料格式為 8 bit 的。鏡片上專用 13 bit 串列串列傳輸速率發生器，可借助工作頻率振盪器產生標準的串列串列傳輸速率。

再者，EUART 首先發送和接收的資料是最低的有效位元，發送和接收在功能上是獨立的，但採用相同的通訊資料格式和串列傳輸速率。且其更支援硬體奇偶校驗功能，校驗位元會保存在資料的第 9 bit 中。

22.1.5 EUART TX 發送模式

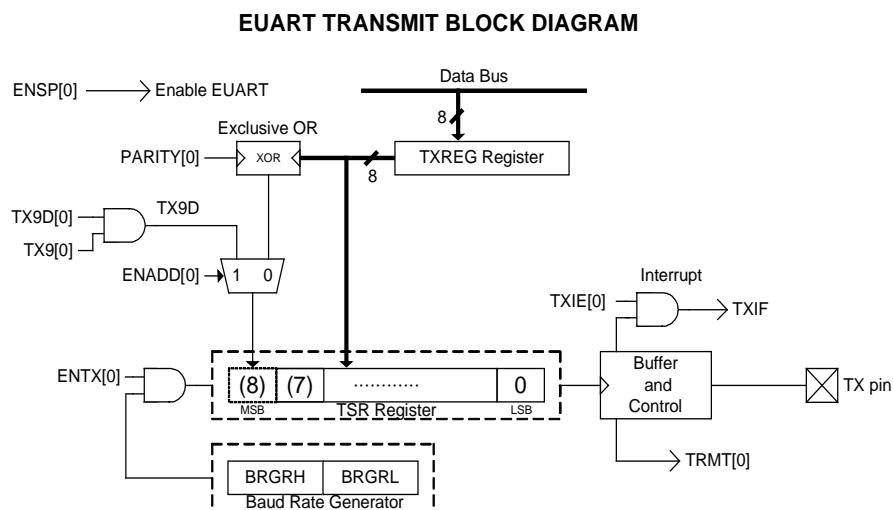


圖 22-1 EUART 發送模組方框圖

圖 22-1 為 EUART 發送模組方框圖，發送器的核心是以串列方式發送移位寄存器 (Transmit Shift Register, TSR) 內資料，而使用者是無法讀寫 TSR。

TSR 是從讀/寫發送緩存寄存器 TXREG[7:0]中獲取資料。TXREG 寄存器中資料通過軟體方式寫入，在前一次載入的停止位發送完之前，都不會再向 TSR 寄存器寫入資料；一旦停止位元被發送完，TXREG 寄存器中的資料就會被存入 TSR，此時 TXREG 的寄存器變為空（當前是沒有再次軟體寫入資料的動作），與此同時 UTXIF 由 0 變為 1 (當使能 ENTX, UTXIF 為 1)。而 UTXIF 會在重新寫入資料後的第 2 個指令週期被置 0，且 UTXIF 被置 0

後，會在一個指令週期後再被置 1。可以通過設置 UTXIE 為 1 或 0 來允許或禁止中斷；而且不管 TXIE 的狀態如何，只要中斷發生，UTXIF 都會由 1 變為 0，並且在一個指令週期後再被置 1。若此時 TSR 寄存器資料尚未發送完畢，TXREG 寄存器又被寫入資料，則在寫入新資料的 2 個指令週期後 TXIF 會被清零並維持到 STOP BIT 發送時才會被置 1。

因此在 TXREG 寫入新資料後立即查詢 UTXIF，得到的值是不可靠的。UTXIF 表示的是當前 TXREG 寄存器的狀態，UTXIF=1 表示 TXREG 為空，UTXIF=0 表示 TXREG 滿載。而 Bit TRMT 則是表示 TSR 寄存器的狀態，TRMT=1 表示 TSR 為空，TRMT=0 表示 TSR 寄存器滿載。TRMT 與 EUART 的中斷無直接的關係，所以用戶需要確定能否再次寫入資料，可以通過查詢 TRMT Bit 的狀態來確定。

EUART 的 TX 模式的功能使用，TX 功能可以通過查詢或中斷的方式實現。

使用時必須注意以下的幾點：

- ✓ UART 動作除了讀寫外，與 CPU 指令週期無關
- ✓ TXIF、RXIF 表示中斷標誌位元，與其他任何時間無關

要用 CPU 去見識周邊元件時，須先注意相對操作速度

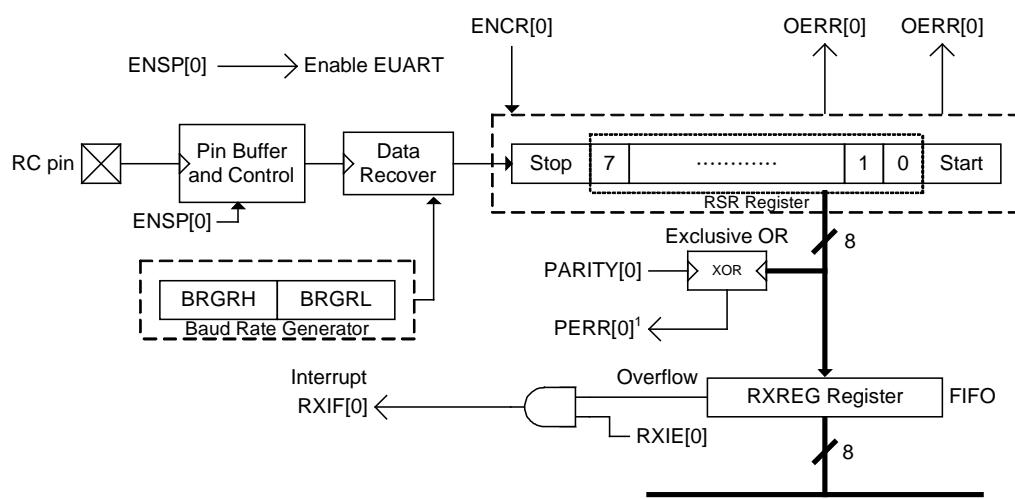
- ✓ 1. reset (ENTX=0) default 狀態：TRMT=1，TXTIP=0，UTXIF=1
- ✓ 2.等到 write TXREG 之後：TRMT=0，TXTIP=1，UTXIF=0
- ✓ 3. 傳輸 TSR
- ✓ 4. 傳完 8/9 bit，TRMT=(是否寫過 TXREG)的反相，TXTIP=0，UTXIF=1
- ✓ 5. 回到第 2 點，繼續下一筆 data 傳輸；或是設 ENTX=0，ENSP=0，結束 UART TX.

中斷方式實現 TX 模式功能：由於 UTXIF 不僅是中斷標誌位元，還是 TXREG 的狀態標誌位元，在上電(reset default)時 UTXIF=1，所以一旦 UTXIE=1 就會進入中斷向量並相應中斷，但是 UTXIF 無法軟體清零，就會產生一直進入中斷的現象。因此使用 TX 功能需要注意，在需要發送資料時才將 UTXIE 置 1，然後進入中斷後再向 TXREG 寫入資料並禁止 TX 中斷即 UTXIE=0。按照此操作方式重複發送資料，便可以實現通過中斷方式來發送資料。

查詢方式實現 TX 模式功能相對來說簡單多，主要在寫入資料後，對 TRMT Bit 進行查詢，查詢狀態為：TRMT=1 為空寫入資料，TRMT=0 為滿載，等待發送完畢。反復操作便可將資料如數發送。

22.1.6 EUART RX 接收模式

EUART 8-BITs RECEIVE BLOCK DIAGRAM



¹Don't care PERR[0] state of 8-bits receive mode

圖 22-2 EUART 8-bits 接收模式方框圖

EUART 9-BITs RECEIVE BLOCK DIAGRAM

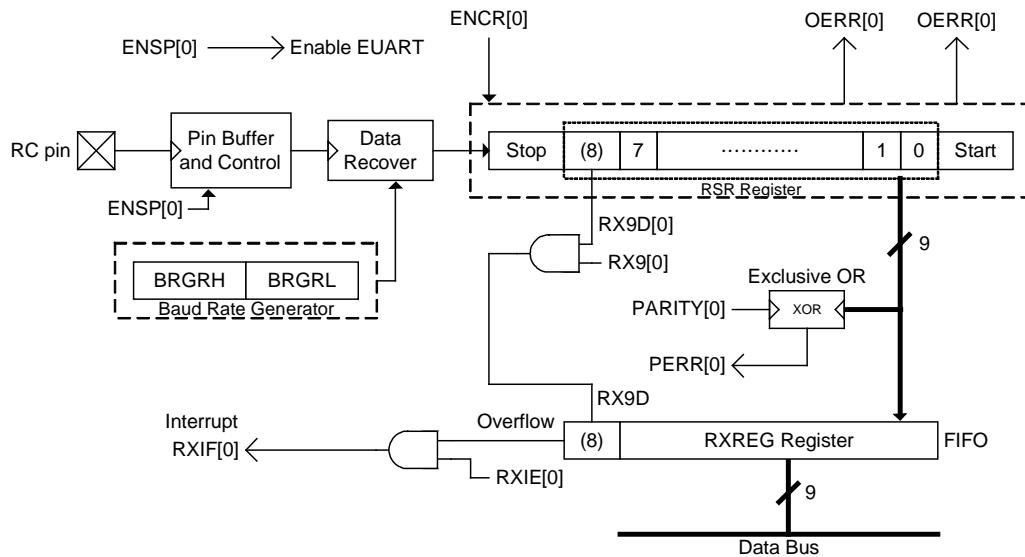


圖 22-3 EUART 9-bit 接收模式方框圖

圖 22-2、圖 22-3 所示 EUART 的接收模式功能方框圖。在 RC 引腳上接收資料，並驅動資料恢復電路。資料恢復電路實際上是一個以 13 bit 串列傳輸串列傳輸速率為工作頻率的高速移位元器，而主接收串列移位元器的工作頻率等於串列傳輸速率或 OSC_RC2M(此模式通常用於 RS-232 系統)。

若 RC 引腳接收資料時未收到完整的位元組【起始位元、資料位元、停止位元】，FERR 位將被設置為 1，可透過清除 ENCR 位來清除 FERR 位。

當 RC 引腳接收到 2 筆完整位元組資料後（皆未從 RCREG 寄存器讀取），OERR 位元將被置 1，可透過清除 ENCR 位來清除 OERR 位。

當完整資料接收完成時，RX 的中斷標誌位元 URXIF 將會被置 1，而 URXIF 是不能軟體清零，只能通過執行讀取 RCREG 寄存器的動作來清除 URXIF。

RCIDL 位元是反應是否處於接收狀態，使用者可通過查詢此位元判斷資料是否接收完成。

資料接收時，硬體將會針對接收到 8bit 資料進行互斥或運算，若 RC9 設置為 1 時，將針對包含接收到的 RC9D 資料進行互斥或運算。運算後再與使用者設定的 PARITY 位再次進行互斥或運算，並將運算結果顯示在 PERR 位元。過接收到資料正確，則 PERR 為 0；若接收到資料錯誤，則 PERR 為 1，PERR 位元無法使用軟體方式清零，只能在下一筆資料接收正確時，PERR 才會被置 0。

RX 的中斷操作方式建議：進入 RX 中斷後，先讀取 RCREG 寄存器，再去清除 URXIF。

22.1.7 自動喚醒功能

在休眠 (sleep) 或待機 (idle) 模式下，EUART 的所有時鐘源都會被暫停。因此串列串列傳輸速率發生器處於非啟動狀態 (idle 模式下 EUART 會動作)，並且無法進行正確的資料接收。自動喚醒功能允許在 RC 線上有事件發生時喚醒控制器，該功能需要 EUART 工作在非同步模式下通過將位元 WUE 置 1，使能並啟動自動喚醒功能。該功能啟動後，將禁止 RC 上的接收操作，且 EUART 保持在空閒狀態並監視喚醒事件。

喚醒事件是指在 RC 線上發生由高電位到低電位的轉換。在喚醒事件後，模組會產生 URXIF 中斷，在正常工作模式下會與 Q 時鐘同步產生；如果晶片處於休眠或等待模式下，則兩者不同步。通過讀取 RCREG 寄存器可清零 RXIF 位。

喚醒事件發生後，當 RC 線上出現由低電平到高電平的轉換時，WUE 位自動清零。此時 EUART 模組將從空閒狀態返回正常工作模式。

注：使用自動喚醒功能注意事項

✓ 因為自動喚醒功能是通過檢測 RC 上的上升沿跳變實現的，所以在停止位前該引腳上任何的狀態改變都有可能會產生錯誤的結束信號並導致資料或幀錯誤，因此為了保證正確的傳輸，必須首先發送全 0 字元。對於標準

的 RS-232 晶片，這可以是 00h。

✓ 另外還必須考慮振盪器起振時間，尤其在採用延遲較長的振盪器應用中更要注意這一點。或喚醒信號字元必須足夠長，並且跟有足夠長的時間間隔，以便是選定振盪器有充足的時間起振並保證 EUART 正確初始化。

注：使用 WUE 位的注意事項

✓ WUE 和 URXIF 時間的時序來判斷接收資料的有效性可能會引起混淆。如前所述，將 WUE 位置 1 會使 EUART 進入空閒模式。喚醒事件會產生一個接收中斷，並將 URXIF 位置 1。此後當 RC 出現上升沿時 WUE 位被清零。然後通過讀取 RCREG 寄存器來清除中斷條件。

✓ 一般情況下，此時喚醒後 RCREG 中的資料是無效的，應該丟棄。WUE 位清零（或仍然置 1）且 URXIF 位置 1 並不能表明 RCREG 中資料接收完整的。用戶應該考慮使用固件驗證是否完整地接收資料。要確保沒有丟失有效資料，應檢查 RCIDL 位元來驗證是否還在接收資料。如果不在進行接收，則可將 WUE 位置 1，使晶片立即進入睡眠模式。

22.2 暫存器位址

UART Register Address	31	24	23	16	15	8	7	0
UART Base Address + 0X00(0X40E00)	-		-		URCONM	URCON		
UART Base Address + 0X04(0X40E04)	-		-		-	URSTA		
UART Base Address + 0X08(0X40E08)	-		-		BACONM	BACON		
UART Base Address + 0X0C(0X40E0C)	-		-		-	BRGRH		
UART Base Address + 0X10(0X40E10)	-		-		-	BRGRL		
UART Base Address + 0X14(0X40E14)	-		-		-	TXREG		
UART Base Address + 0X18(0X40E18)	-		-		-	RCREG		

-保留

22.3 暫存器功能

22.3.1 通信 UART 暫存器 UARTCR0

UART Base Address + 0X00 (0X40E00)								
Symbol	UARTCR0 (UART Control Register 0)							
Bit	[31:16]							
名稱	RSV.							
RW	R-0							
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2:1]	[0]
名稱	MASK	ENSP	ENTX	TX9	TX9D	PARITY	-	WUE
RW	R0W-0	RW-0						- RW-0

位元	名稱	描述
Bit[7]	ENSP	UART 功能開啟控制
		0 關閉
		1 開啟
Bit[6]	ENTX	UART 發送功能開啟控制
		0 關閉
		1 開啟
Bit[5]	TX9	發送數據長度控制 (8 位或 9 位)
		0 發送 8 位的數據
		1 發送 9 位的數據
Bit[4]	TX9D	設置發送數據的第 9 位狀態值

		0	發送數據的第 9 位是 0
		1	發送數據的第 9 位是 1
Bit[3]	PARITY	數據校驗設置(無校驗/偶校驗/奇校驗)	
		0	偶校驗
Bit[0]	WUE	喚醒功能開啟控制	
		0	關閉
		1	開啟

22.3.2 通信 UART 暫存器 1

UART Base Address + 0X04 (0X40E04)									
Symbol	UART1 (UART Control Register 1)								
Bit	[31:16]								
名稱	RSV.								
RW	R-0								
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名稱	RSV	-	RC9D	PERR	FERR	OERR	RCIDL	TRMT	ABDOCF
RW	R-0	-	R-0						

位元	名稱	描述
Bit[6]	RC9D	接收到 9 位的數據，數據的第 9 位狀態值，且第 9 位數據是存儲在這裡
		0 數據的第 9 位是 0
		1 數據的第 9 為時 1
Bit[5]	PERR	校驗錯誤旗標
		0 未發生校驗湊無
Bit[4]	FERR	校驗錯誤產生，無法軟件清零，待下筆數據正確發送時自動清零
		UART 帧錯誤旗標
		0 未發生帧錯誤
Bit[3]	OERR	1 帧錯誤發生
		溢出錯誤旗標
		0 未發生溢出錯誤
Bit[2]	RCIDL	1 發生溢出
		接收數據狀態旗標
		0 空閒
Bit[1]	TRMT	1 正在接收數據
		發送移位暫存器狀態旗標
		0 發送移位暫存器為滿
Bit[0]	ABDOVF	1 發送移位暫存器為空
		自動串列傳輸速率計算滿位旗標
		0 計算未發生滿位
		1 計算發生滿位；由軟件清零

22.3.3 通信 UART 暫存器 UARTCR2

UART Base Address + 0X08 (0X40E08)									
Symbol	UARTCR2 (UART Control Register 2)								
Bit	[31:16]								
名稱	RSV.								

RW	R-0					
Bit	[15:8]	[7:4]	[3]	[2]	[1]	[0]
名稱	MASK	-	ENCR	RC9	ENADD	ENABD
RW	R0W-0	-			RW-0	

位元	名稱	描述					
Bit[3]	ENCR	Clear FERR Control					
		0	Normal				
		1	Clear FERR, then return 0 automatically				
Bit[2]	RC9	接收 9 位或 8 位數據控制					
		0	接收 8 位數據				
		1	接收 9 位數據				
Bit[1]	ENADD	地址檢測開啟控制器(RC9 = 1 時候才成立)					
		0	關閉地址檢測功能, 接收到所有數據的第 9-bit 作為奇偶校驗功能				
		1	開啟地址檢測功能, 允許中斷與使用接收緩衝器. TX9D, RC9D 可有資料做判讀.				
Bit[0]	ENABD	自動串列傳輸速率自動檢測開啟控制					
		0	關閉自動檢測或檢測已完成				
		1	開啟自動串列傳輸速率功能；檢測數據 0X55，完成後自動清零				

22.3.4 通信 UART 暫存器 UARTCR3

UART Base Address + 0X0C (0X40E0C)					
Symbol	UARTCR3 (UART Control Register 3)				
Bit	[31:16]				
名稱	-				
RW					
Bit	[15:8]				[4:0]
名稱	MASK	-			BRGR12-8
RW	R0W-0	-			RW-0

位元	名稱	描述		
Bit[4~0]	BRGR12-8	串列傳輸速率發生器高 5 位計數值		

22.3.5 通信 UART 暫存器 UARTCR4

UART Base Address + 0X10 (0X40E10)					
Symbol	UARTCR4 (UART Control Register 4)				
Bit	[31:16]				
名稱	RSV				
RW					
Bit	[15:8]				[7:0]
名稱	MASK	-			BRGR7-0
RW	R0W-0	-			RW-0

位元	名稱	描述		
Bit[7~0]	BRGR7-0	串列傳輸速率發生器低 8 位計數值		

22.3.6 通信 UART 暫存器 UARTCR5

UART Base Address + 0X14(0X40E14)		
Symbol	UARTCR5 (UART Control Register 5)	
Bit	[31:16]	
名稱	RSV	
RW	R-0	
Bit	[15:8]	[7:0]
名稱	RSV	TXREG
RW	R-0	RW-0

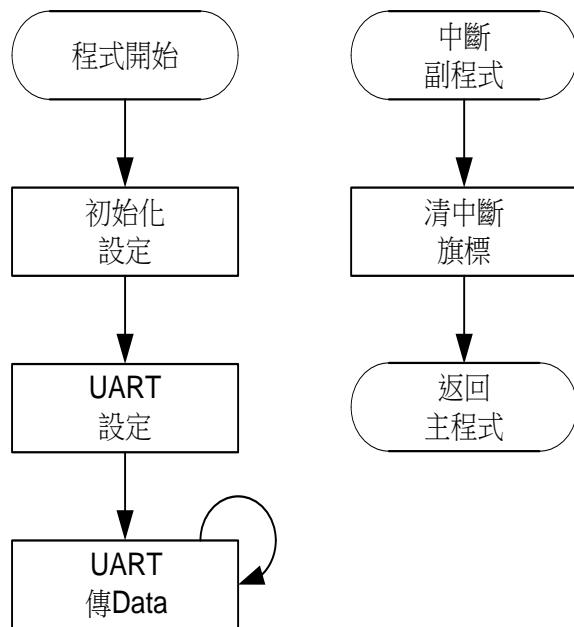
位元	名稱	描述
Bit[7~0]	TXREG	TXREG[7:0]是發送緩衝器

22.3.7 通信 UART 暫存器 UARTCR6

UART Base Address + 0X14(0X40E18)		
Symbol	UARTCR6 (UART Control Register 6)	
Bit	[31:16]	
名稱	RSV	
RW	R-0	
Bit	[15:8]	[7:0]
名稱	RSV	RCREG
RW	R-0	RW-0

位元	名稱	描述
Bit[7~0]	RCREG	RCREG[7:0]是接收緩衝器

22.4 範常式流程



22.5 範常式功能

範例名稱:通信協定 UART 中斷設定

範例說明

(1) 將 TX 與 RX PIN 連接 RS232 相關電路

26.6 範常式說明

00		
01	#include "HY16F18X.h"	
02	#include "DrvREG32.h"	
03		
04	void Delay (unsigned int num);	
05	void All_in_One_Initial(void);	
06		
07	unsigned int temp,Buffer_Start,PT_1D7,i;	//
08		
09		
10	int main(void)	
11	{	
12	All_in_One_Initial();	Initial UART and others
13	PT_1D7=DrvGPIO_GetBit(E_PT1,7);	//
14	SYS_EnableGIE(7);	
15	Buffer_Start=0x00;	
16		
17	while(1)	
18	{	
19	i=DrvGPIO_GetBit(E_PT1,7);	//read PT1.7 pin
20	if(i==0)	//high or low

21	{	//If PT1.7=0
22	int_00=0x08080c00;	//UART INT set 0X08080C00
23	DrvUART_Write(Buffer_Start++);	
24	Delay(0x8000);	
25	}	
26	Delay(0x8000);	
27	}	
28		
29	return 0;	
30	}	
31		
32	void HW0_ISR(void)	//UART Interrupt
33	{	
34	int_00=0x0e000c00;	//Clear UART interrupt flag
35	temp=DrvUART_Read();	
36	int_00=0x08000c00;	//Clear UART interrupt flag
37	}	
39		

40		
41	void Delay(unsigned int num)	
42	{	
43	volatile unsigned int d;	
44	for(d=0;d<=num;d++);asm("NOP");	
45	}	
46		
47	void All_in_One_Initial(void)	
48	{	
49	DrvGPIO_Open(E_PT1,0x80,E_IO_INPUT);	//set PT1_7 INPUT
50	DrvGPIO_Open(E_PT1,0x80,E_IO_PullHigh);	//enable PT1_7
51		//pull high R
52	DrvGPIO_Open(E_PT2,0x04,E_IO_OUTPUT);	//PT2.2 output TX.5
53	DrvGPIO_Open(E_PT2,0x08,E_IO_INPUT);	//PT2.3 input RX.5
54		
55	DrvCLOCK_EnableHighOSC(E_EXTERNAL);	//select HSXT 4MHz
56	DrvUART_Open(4,B9600,DRVUART_PARITY_NONE,DRVUART_DATABITS_8,5);	
57		
58	DrvUART_DisableAutoBaudrate();	
59	clk_08=0xff10ff00;	
60		
61	asm volatile("sethi \$r0, 0xc0000");	//N801 GIE setting1
62	asm volatile("ori \$r0, \$r0, 0x003f");	
63	asm volatile("mtsr \$r0, \$INT_MASK");	
64	asm volatile("movi \$r0, 0x70009");	//N801 GIE setting2
65	asm volatile("mtsr \$r0, \$PSW");	
66	}	
67		

23.I2C 通信介面

23.1 整體總說明

通訊介面(CI)

I2C 串列通訊的主要類別。
I2C 串列介面(跨集成電路串列介面)
I2C 通訊介面包含主(Master)和從(Slave)兩個運作模式。

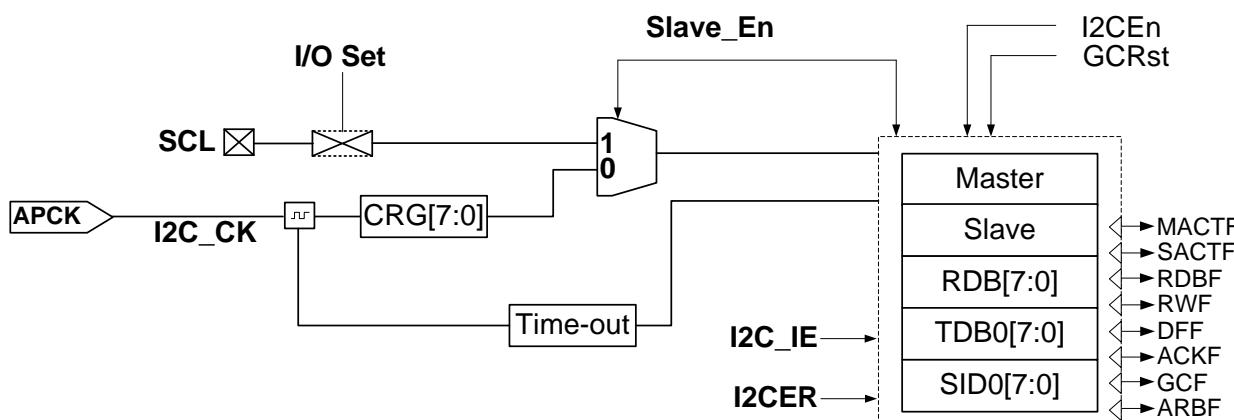


圖 23-1 I2C 通訊架構圖

23.1.1 通信 I2C 介面特性:

標準 I2C 串列介面包括 2 接腳的串列數據(SDA)和串列時鐘(SCL)。接腳是開放式漏極開路輸出結構，需要外部上拉電阻以確保高階輸出。標準 I2C 串列介面可以組態設定為主(Master)模式、從(Slave)模式、或主/從(master/slave)模式。可編程時鐘允許調整 I2C 的傳輸速率。在主從之間，數據以雙向傳輸。I2C 允許大的運作電壓範圍。I2C 的參考設計使用一個 7 位元的長位址空間，但保留了 16 個位址，以處理一群匯流排及最多可達 112 個節點的通訊。

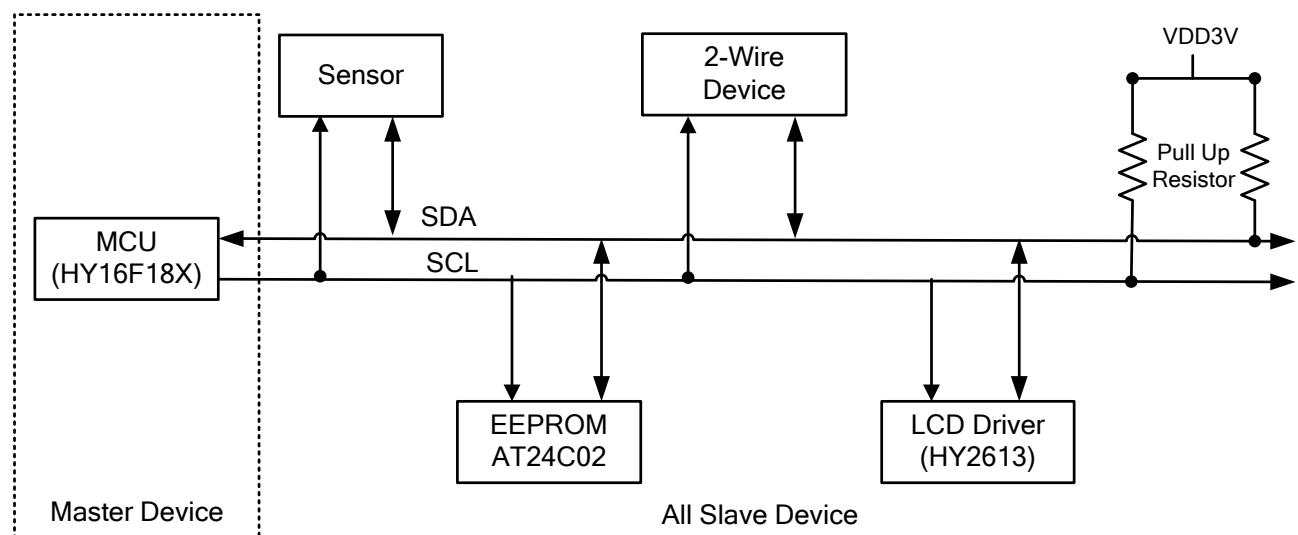


圖 23-2 I2C 總線掛接器件圖

23.1.2 通信 I2C 介面訊號

開始訊號(START):

主端模式 SCL 為高電位元。從高電位發出 SDA 到低電位以啟動數據傳輸。

數據(DATA)和位址(ADDRESS)訊號：

I2C 串列介面協定只有當 SCL 是低電位元時才需要，SDA 只能依據數據來改變。

回應訊號(Acknowledge)：

接收數據(Slave)是自最初的 8 個位元之後才開始。

傳送數據到一個裝置(Host)是送一個低電位，表示數據已經被收到。

停止訊號(STOP)：

主端模式 SCL 是高電位元。從低電位發出 SDA 到高電位以結束一個數據傳輸。

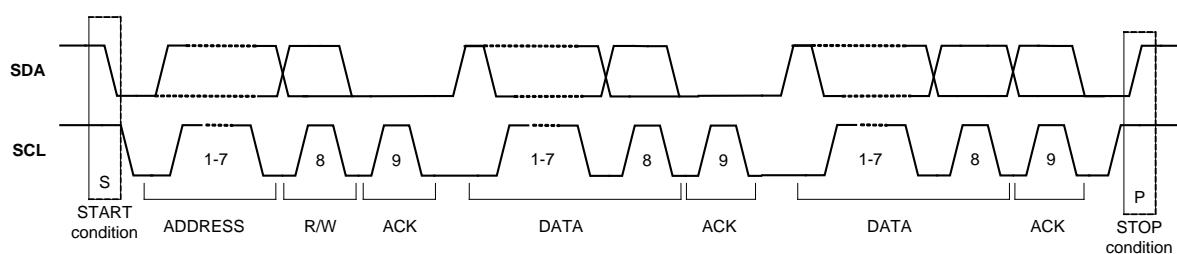


圖 23-3 I2C 總線時序圖

數據傳輸率的計算：

I2C 內部暫存器 CRG [7:0]可控制主端模式的數據傳輸率。內部計數器的 CRG [7:0]值經由 SCL 接腳訊號產生一個主端模式，使得數據傳輸率可以 I2C 時鐘源 2C_CK 頻率為依據。

I2C 汇流排 SCL 接腳上的數據傳輸是時脈訊號，係由 I2C 電路時鐘源頻率 I2CLK 的 CRG 依據 SCL 接腳的時脈率，以下列公式來決定：

$$\text{Data Baud Rate} = (\text{I2CLK}) / [4X (\text{CRG} + 1)] \quad (\text{式 23-1})$$

超時控制 Timing function (Time-Out):

超時控制是要避免 I2C 控制器在 I2C 運作時將 I2C 通訊匯流排鎖住，以便提供足夠的時間來處理 MCU I2C 控制器。因此，I2C 控制器對每一個位元的回應要在 SCL 降到低電位時才會發生；此時，主端無法聽到下一個時脈訊號，也就是說，產生了一個通訊延遲(Clock Stretching)。但當 MCU 太過忙碌時，或其他原因無法回應 I2C 控制器，則 SCL I2C 通訊匯流排就有可能在低電壓被鎖住。

為了避免這種情況的發生，超時控制器會依據運作頻率分頻器 DI2C [2:0]和時間條件控制器 I2CTLT [3:0]來決定 SCL 狀態是否為低電位超時情況。

條件處理有以下幾種狀態：

當機器偵測到 SCL 被拉到低電位並符合條件時，就會強迫 SCL I2C 控制器放手，且會送出一個中斷事件給 CPU。

當 SCL 未符合超時條件且處於高電位時，超時控制器的內部計數器就會被重設，且會將下一個 SCL 拉向低電位並重新計算。

I2C 通訊引腳

I2C 總線只有兩根線，但晶片給 I2C 模塊配置 8 組通訊 IO 引腳(一組 IO 引腳包含 SCL/SDA)，是 IO 口的複用功能。方便用戶可以自由的設置選擇不同通訊引腳。透過控制器 I2CPTS[2:0]、I2CPTEN 選擇及開啟相應的通訊引腳。在使用 I2C 功能時，必須先開啟通訊 IO 引腳，對應的 IO 引腳需設置為輸入或輸出模式。下表為通訊引腳分佈表。

I2CPTS[2:0]	I2CPTEN	SCL	SDA	I2CPTS[2:0]	I2CPTEN	SCL	SDA
000	1	PT1.0	PT1.1	100	1	PT2.0	PT2.1
001	1	PT1.2	PT1.3	101	1	PT2.2	PT2.3
010	1	PT1.4	PT1.5	110	1	PT2.4	PT2.5
011	1	PT1.6	PT1.7	111	1	PT2.6	PT2.7

表 23-1 I2C 通訊 IO 引腳分佈

23.1.3 通信 I2C 介面流程

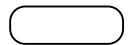
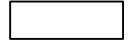
I2C 串列介面術語

(SPIA):代表對 Action 控制暫存器所下達指令, S 為 Start 指令, P 為 Stop 指令,
I 為中斷旗標, A 為 Acknowledge 指令。

SPIA:代表讀取 Action 控制暫存器之值，可以用於判讀中斷旗標或其他指令是否運作完成。

STA:讀取 Status 暫存器之值，用以表示目前 I2C 電路運作狀態。

下列流程圖會以圖所示之(灰底圓框)、(白底圓框)、(純白方框)
分別表示 I2C 介面之狀態：

-  Status With IRQ
-  Status Without IRQ
-  Action

灰底圓框:表示中斷旗標已被設立之 I2C 狀態。

白底圓框:表示中斷旗標未被設立，需由 MCU 主動讀取之 I2C 狀態。

純白方框:表示需由 MCU 對 I2C 下達指令。

23.1.3.1 串口 Master TX 流程

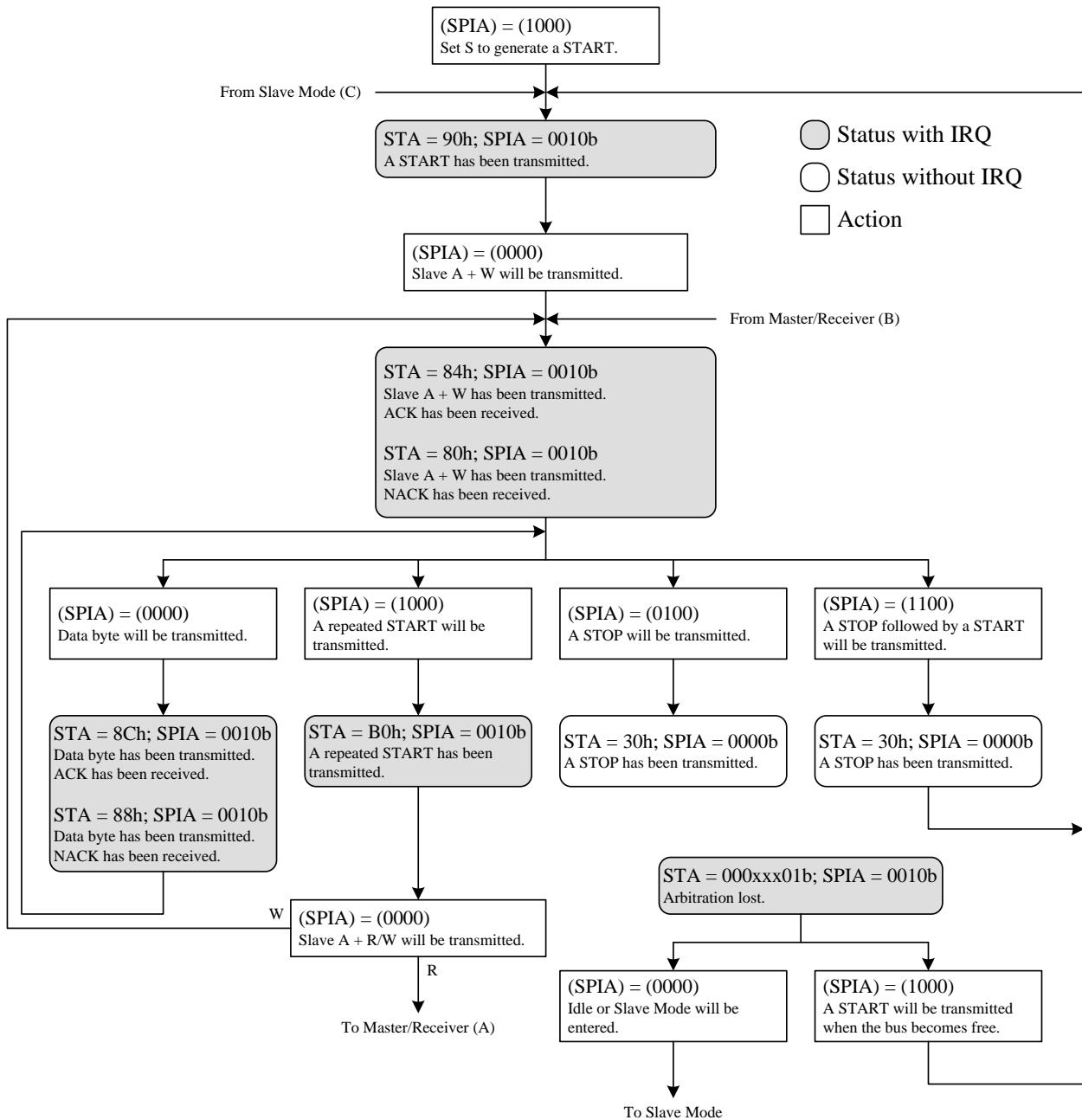


圖 23-4 Master Transmitter Mode

23.1.3.2 串口 Master RX 流程

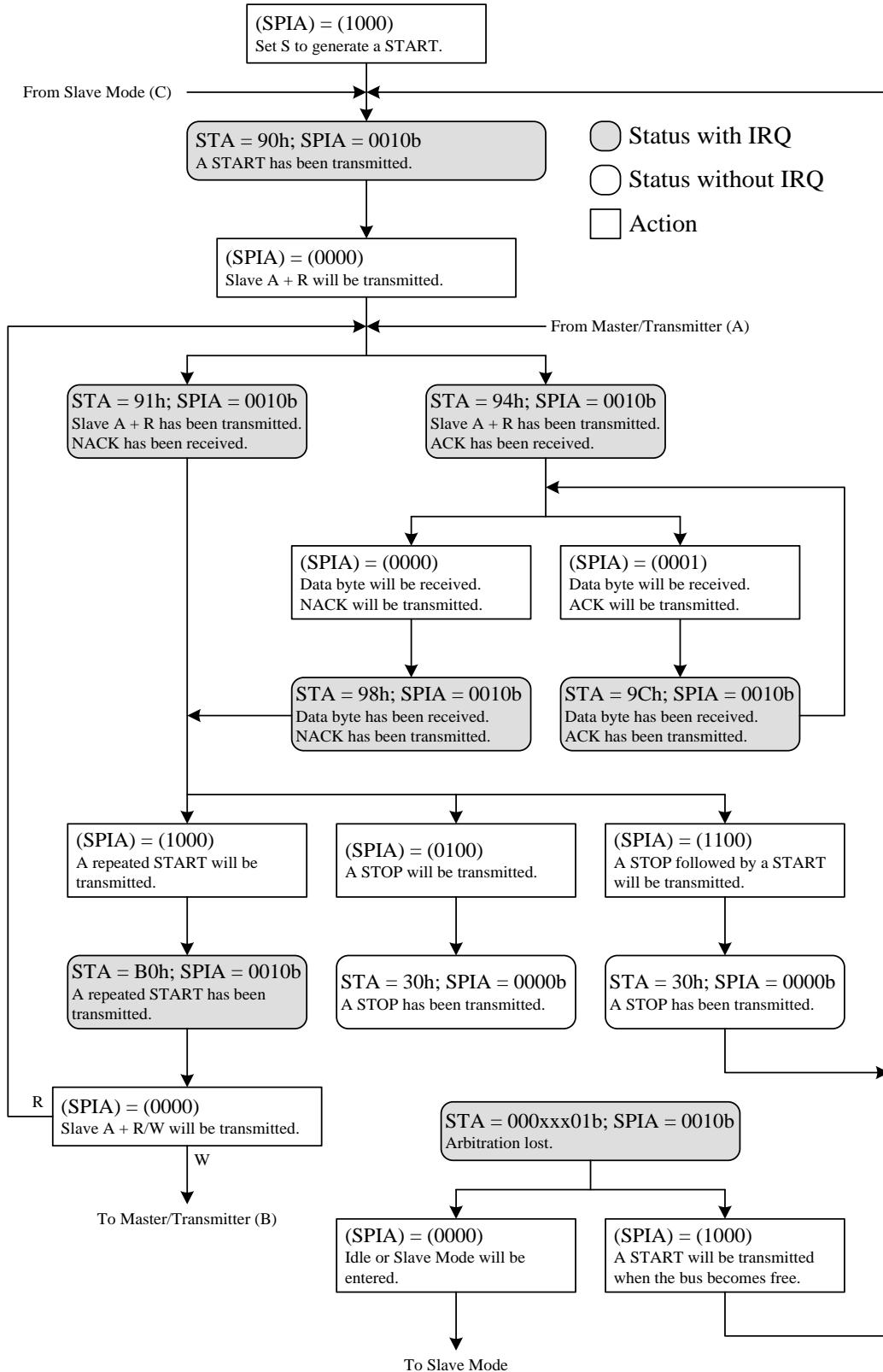


圖 23-5 Master Receiver Mode

23.1.3.3 串口 Slaver TX 流程

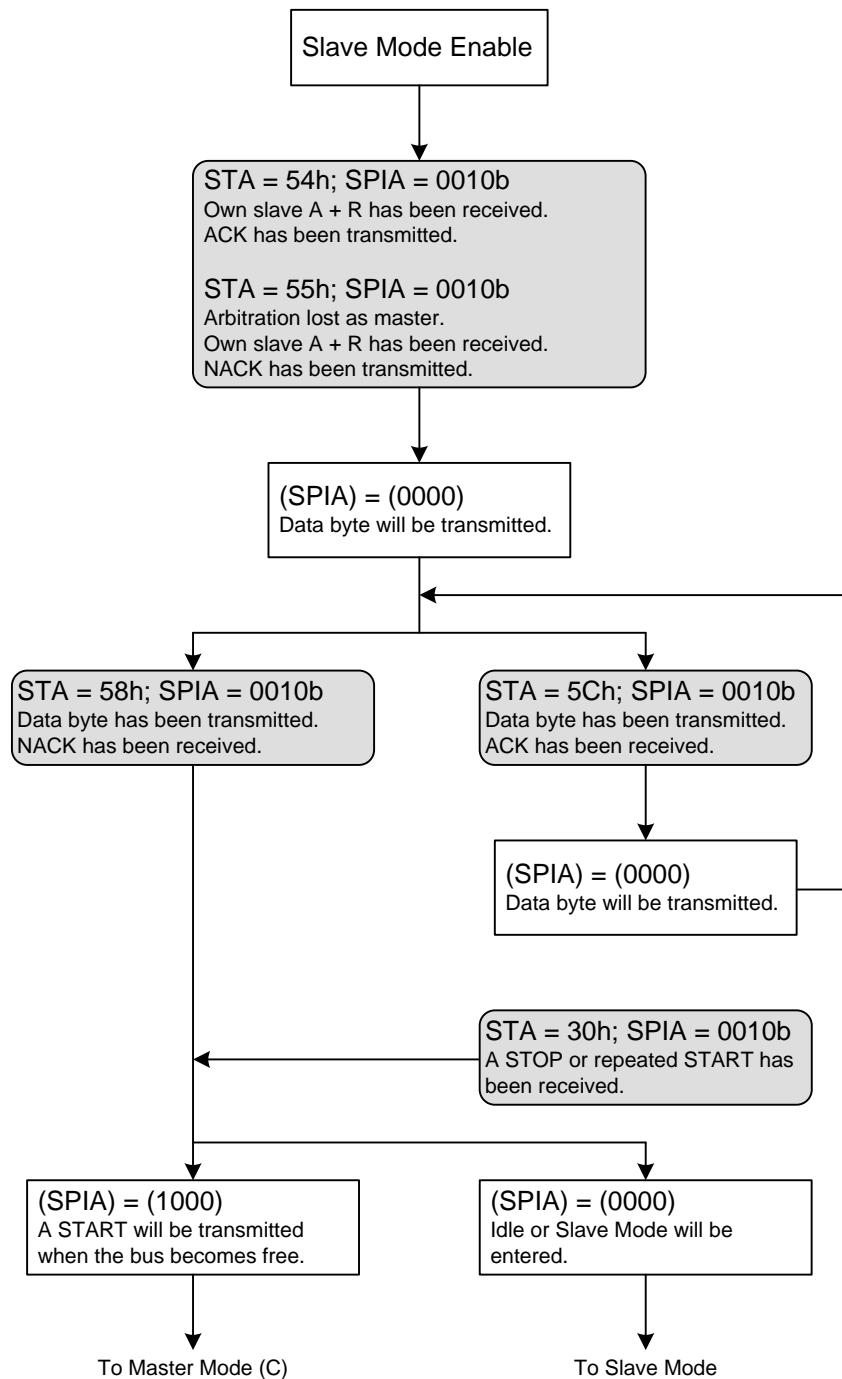


圖 23-6 Slave Transmitter Mode

23.1.3.4 串口 Slaver RX 流程

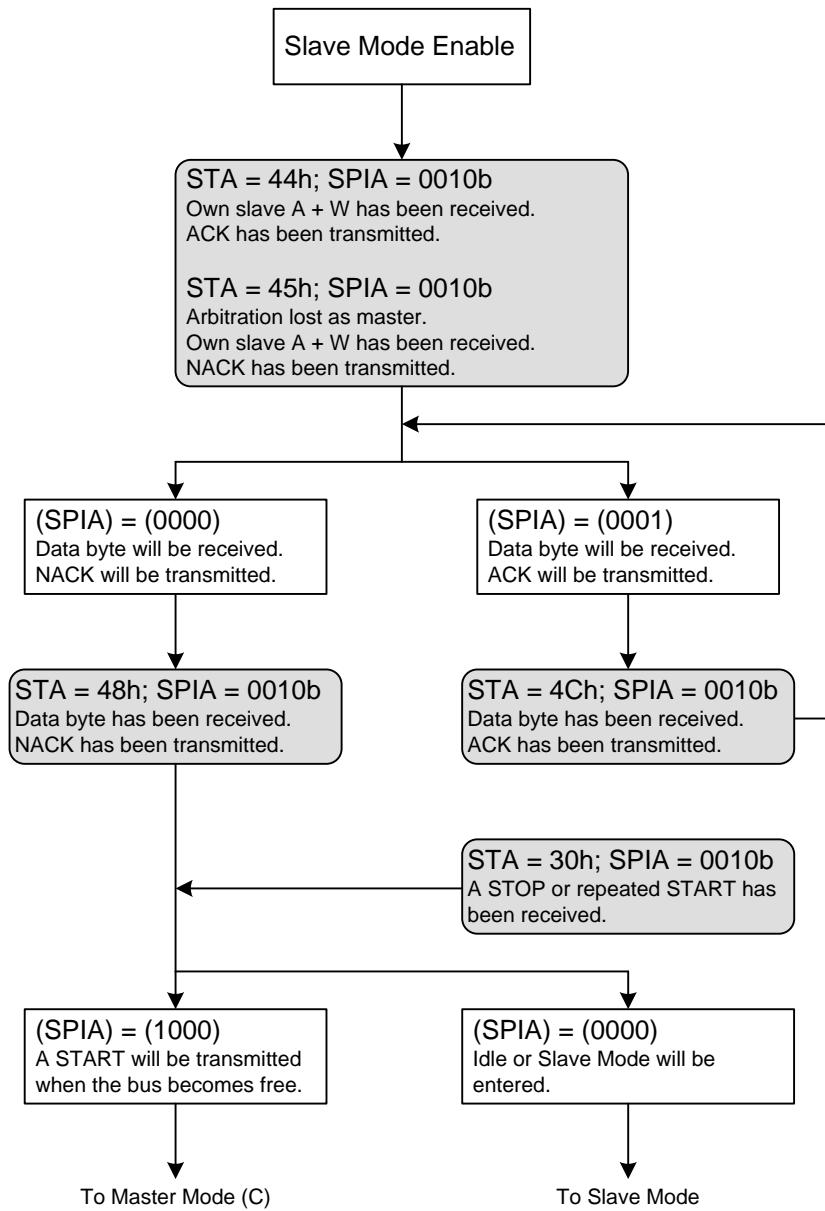


圖 23-7 Slave Receiver Mode

23.1.3.5 串口 General Call 流程

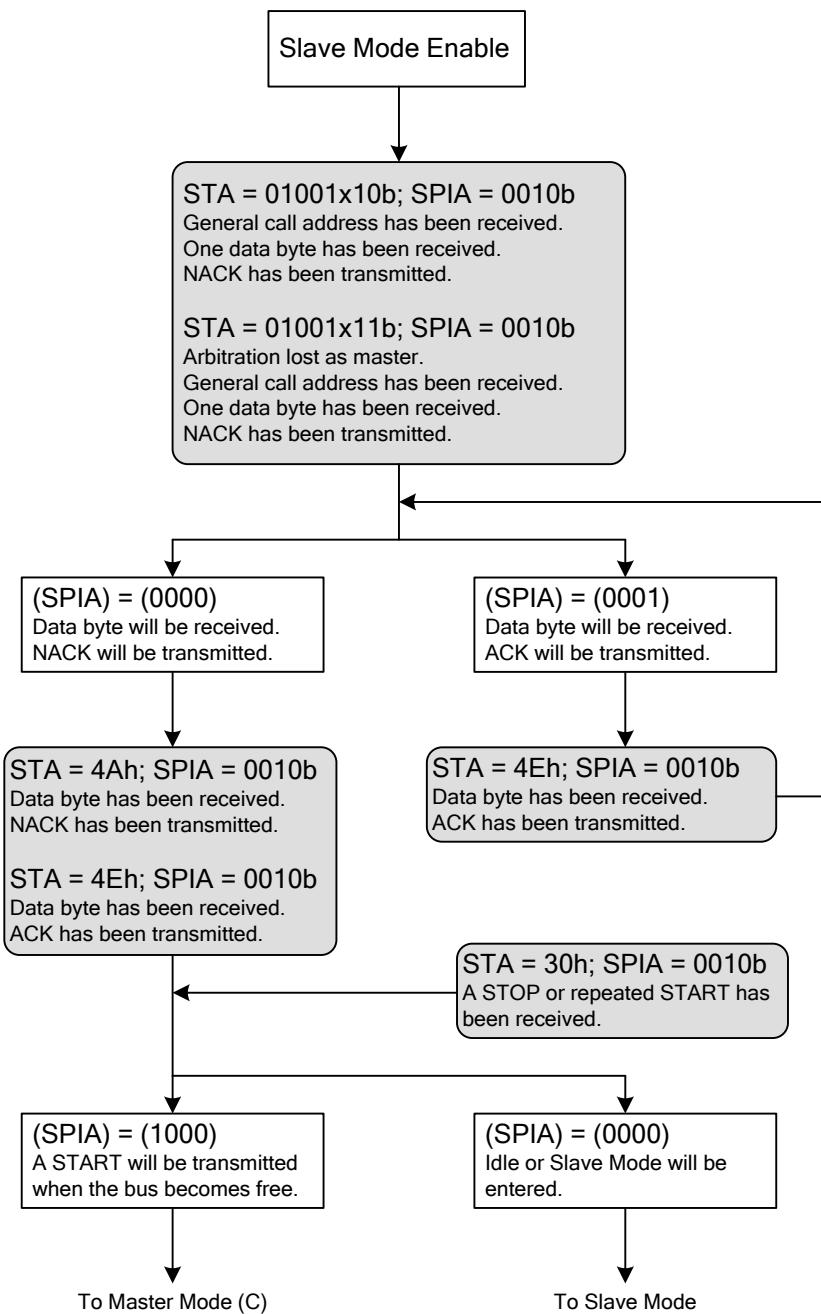


圖 23-8 General Call Mode

23.2 暫存器位址

I2C Register Address	31	24	23	16	15	8	7	0
I2C Base Address + 0X00 (0X41000)	-	-		MASK0	I2C_CON0			
I2C Base Address + 0X04 (0X41004)	MASK1		I2C_CON1		MASK2	I2C_CON2		
I2C Base Address + 0X08 (0X41008)	MASK3		I2C_CON3		MASK4	I2C_CON4		
I2C Base Address + 0X0C (0X4100C)	MASK5		MASK6		I2C_CON5	I2C_CON6		
I2C Base Address + 0X10 (0X41010)	-	-		-		I2C_CON7		
I2C Base Address + 0X14 (0X41014)	-	-		-		I2C_CON8		

-保留

23.3 暫存器功能

23.3.1 通信 I2C 暫存器 I2CCR0

I2C Base Address + 0X00 (0X41000)					
Symbol	I2CCR0 (I2C Control Register 0)				
Bit	[31:16]				
名稱	RSV.				
RW	R-0				
Bit	[15:8]	[7:3]	[2]	[1]	[0]
名稱	MASK	RSV.	GCRst	TOEn	I2CEn
RW	ROW-0	R-0		RW-0	

位元	名稱	描述
Bit[02]	GCRst	全呼復位使能控制
		0 關閉
		1 開啟
Bit[01]	TOEn	超時復位功能開啟控制
		0 關閉
		1 開啟
Bit[00]	I2CEn	I2C 功能開啟控制
		0 關閉
		1 開啟

23.3.3 通信 I2C 暫存器 I2CCR1

Action Register (ACT)

I2C Base Address + 0X04 (0X41004)										
Symbol	I2CCR1 (I2C Control Register 1)									
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]	
名稱	MASK	M _{Act}	S _{Act}	R _x P/S _r	R/W	DF	A/NA	GC	ARB	
RW	R0W-0	R-0								
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]	
名稱	MASK	SEn	10bEn	3BEn	EIRQFlag	START	STOP	IRQFlag	A/NA	
RW	R0W-0	RW-0								

位元	名稱	描述	
Bit[23]	M _{Act}	主機模式啟用旗標	
		0	未啟用
		1	啟用
Bit[22]	SAct	從機模式啟用旗標	
		0	未啟用
		1	啟用
Bit[21]	Rx P/Sr	接收停止或重新開始旗標	
		0	正常
		1	接收停止或重新開始旗標已被發送或接收.
Bit[20]	R/W	讀寫狀態旗標	
		0	寫命令已被發送或接收
		1	讀命令已被發送或接收
Bit[19]	DF	資料旗標	
		0	正常
		1	I2C 資料已被發送或接收
Bit[18]	A/NA	應答信號(ACK)狀態旗標	
		0	應答信號 (ACK) 未被發送或接收
		1	應答信號(ACK)已被發送或接收
Bit[17]	GC	全呼狀態旗標	
		0	正常
		1	當前正進行全呼操作
Bit[16]	ARB	仲裁漏失旗標	
		0	正常
		1	仲裁漏失

位元	名稱	描述		
Bit[07]	SEn	從機模式開啟控制		
		<table border="1"> <tr> <td>0</td><td>關閉</td></tr> <tr> <td>1</td><td>開啟</td></tr> </table>	0	關閉
0	關閉			
1	開啟			
Bit[06]	10bEn	從機 10 位元位址碼模式開啟控制		
		<table border="1"> <tr> <td>0</td><td>關閉</td></tr> <tr> <td>1</td><td>開啟 10 位元位址碼模式</td></tr> </table>	0	關閉
0	關閉			
1	開啟 10 位元位址碼模式			
Bit[05]	3BEn	從機發送 3 筆數據功能開啟控制		
		<table border="1"> <tr> <td>0</td><td>關閉</td></tr> <tr> <td>1</td><td>開啟從機連續發送 3 筆數據功能</td></tr> </table>	0	關閉
0	關閉			
1	開啟從機連續發送 3 筆數據功能			
Bit[04]	EIRQFlag	錯誤旗標，與錯誤中斷旗標 I2CEIF 相關聯，先清零該位才能清零 I2CEIF		
		<table border="1"> <tr> <td>0</td><td>正常</td></tr> <tr> <td>1</td><td>發生超時或接收到意外的起始（停止）信號或仲裁失敗</td></tr> </table>	0	正常
0	正常			
1	發生超時或接收到意外的起始（停止）信號或仲裁失敗			
Bit[03]	START(S)	起始信號(S)控制位		
		<table border="1"> <tr> <td>0</td><td>正常</td></tr> <tr> <td>1</td><td>於 I2C 總線上產生起始信號</td></tr> </table>	0	正常
0	正常			
1	於 I2C 總線上產生起始信號			
Bit[02]	STOP(P)	停止信號(P)控制位		
		<table border="1"> <tr> <td>0</td><td>正常</td></tr> <tr> <td>1</td><td>於 I2C 總線上產生停止信號</td></tr> </table>	0	正常
0	正常			
1	於 I2C 總線上產生停止信號			
Bit[01]	IRQFlag(I)	(I) 器件狀態控制位		
		<table border="1"> <tr> <td>0</td><td>正常；</td></tr> <tr> <td>1</td><td>響應中斷，器件接收到 9 個 clock 後響應中斷，並將 SCL 拉低直到該位被清零後釋放 SCL 信號線；寫入 0 將會清零器件狀態控制位，使 I2C 往下一個狀態執行。</td></tr> </table>	0	正常；
0	正常；			
1	響應中斷，器件接收到 9 個 clock 後響應中斷，並將 SCL 拉低直到該位被清零後釋放 SCL 信號線；寫入 0 將會清零器件狀態控制位，使 I2C 往下一個狀態執行。			
Bit[00]	A/NA(A)	應答信號(ACK)回覆控制位		
		<table border="1"> <tr> <td>0</td><td>未回覆 ACK 或回覆 NACK</td></tr> <tr> <td>1</td><td>應答信號(ACK)已回覆。</td></tr> </table>	0	未回覆 ACK 或回覆 NACK
0	未回覆 ACK 或回覆 NACK			
1	應答信號(ACK)已回覆。			

23.3.4 通信 I2C 暫存器 I2CCR2

I2C Base Address + 0X08 (0X41008)				
Symbol	I2CCR2 (I2C Control Register 2)			
Bit	[31:24]		[23:16]	
名稱	MASK		CRG	
RW	R0W-0		RW-0	
Bit	[15:08]	[7]	[6:4]	[3:0]
名稱	MASK	TOFlag	TOPS	TOLimit
RW	R0W-0	R-0	RW-0	

位元	名稱	描述
Bit[23~16]	CRG	I2C 總線數據串列傳輸速率控制暫存器
		0 置 0
		1 置 1

I2C 總線的數據串列傳輸速率決定於 I2C 的時脈源和串列傳輸速率控制暫存器 CRG 的值；I2C 總線數據串列傳輸速率可由以下公式計算得到：

$$\text{Data Baud Rate} = (\text{I2CLK}) / [4 \times (\text{CRG} + 1)]$$

位元	名稱	描述
Bit[7]	TOFlag	超時旗標
		0 正常
		1 I2C 總線時鐘線高低電位控制超時
Bit[6~4]	TOPS	超時時鐘除頻器設置
		111 $\text{CLK}_{\text{PS}} = \text{I}^2\text{CLK} / 128$
		110 $\text{CLK}_{\text{PS}} = \text{I}^2\text{CLK} / 64$
		101 $\text{CLK}_{\text{PS}} = \text{I}^2\text{CLK} / 32$
		100 $\text{CLK}_{\text{PS}} = \text{I}^2\text{CLK} / 16$
		011 $\text{CLK}_{\text{PS}} = \text{I}^2\text{CLK} / 8$
		010 $\text{CLK}_{\text{PS}} = \text{I}^2\text{CLK} / 4$
		001 $\text{CLK}_{\text{PS}} = \text{I}^2\text{CLK} / 2$
		000 $\text{CLK}_{\text{PS}} = \text{I}^2\text{CLK} / 1$

位元	名稱	描述
Bit[3~0]	TOLimit	超時上限值設置
		1111 16x CLK_{PS} Cycle
		1110 15x CLK_{PS} Cycle
		1101 14x CLK_{PS} Cycle
		1100 13x CLK_{PS} Cycle
		1011 12x CLK_{PS} Cycle
		1010 11x CLK_{PS} Cycle
		1001 10x CLK_{PS} Cycle
		1000 9x CLK_{PS} Cycle
		0111 8x CLK_{PS} Cycle
		0110 7x CLK_{PS} Cycle
		0101 6x CLK_{PS} Cycle
		0100 5x CLK_{PS} Cycle
		0011 4x CLK_{PS} Cycle
		0010 3x CLK_{PS} Cycle
		0001 2x CLK_{PS} Cycle
		0000 1x CLK_{PS} Cycle

23.3.5 通信 I2C 暫存器 I2CCR3

Slave ID0 (SID0)

I2C Base Address + 0X0C (0X4100C)			
Symbol	I2CCR3 (I2C Control Register 3)		
Bit	[31:24]		[23:16]
名稱	SID ₁ MASK		SID ₀ MASK
RW	R0W-0		R0W-0
Bit	[15:9]	[8]	[7:1] [0]
名稱	SID ₁	VD1	SID ₀
RW	RW-0		RW-0

位元	名稱	描述
Bit[31~24]	SID1 MASK	SID1 MASK
		0 關閉 1 開啟
Bit[23~16]	SID0 MASK	SID0 MASK
		0 關閉 1 開啟
Bit[15~9]	SID1	SID1 從機地址碼設置
		0 置 0 1 置 1
Bit[08]	Valid1	從機位址碼有效控制位，寫入位址碼時該位必須為 1
		0 從機地址碼無效 1 從機位址碼有效
Bit[7~1]	SID0	SID0 從機地址碼設置
		0 置 0 1 置 1
Bit[00]	Valid0	從機位址碼有效控制位，寫入位址碼時該位必須為 1
		0 從機地址碼無效 1 從機位址碼有效

23.3.6 通信 I2C 暫存器 I2CCR4

I2C Base Address + 0X10 (0X41010)			
Symbol	I2CCR4 (I2C Control Register 4)		
Bit	[31:16]		
名稱	RSV.		
RW	R-0		
Bit	[15:8]	[7:1]	[0]
名稱	RSV	Rx A7-1/D7-1	RW/D0
RW	R-0	R-X	

位元	名稱	描述
Bit[7~1]	Rx A7-1/D7-1	接收位址或數據暫存器 RX[7:0]
		0 置 0
		1 置 1
Bit[0]	RW/D0	接收到的數據是讀寫命令或數據的第 0 位值
		0 置 0
		1 置 1

23.3.7 通信 I2C 暫存器 5

Transmitter Data Buffer 0 (TXAD)

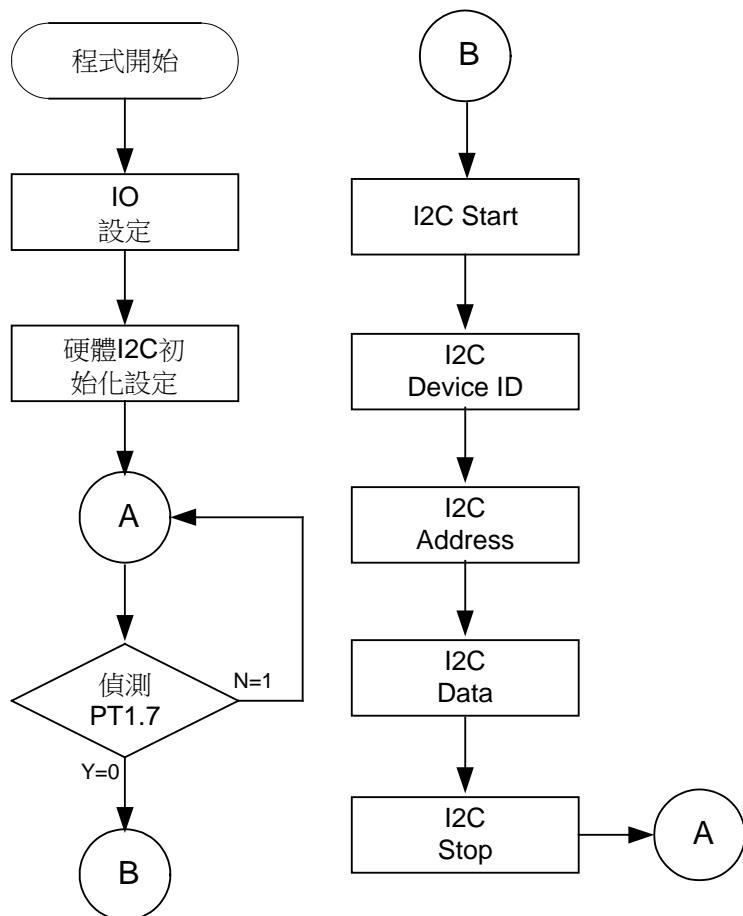
I2C Base Address + 0X14 (0X41014)			
Symbol	I2C 5 (I2C Control Register 5)		
Bit	[31:24]	[23:17]	[16]
名稱	RSV.	TX2 A7-1/D7-1	Flag/D0
RW	R-0	RW-X	
Bit	[15:08]	[7:1]	[0]
名稱	TX1 A7-0/D7-0	TX0 A7-1/D7-1	RW/D0
RW	RW-X		

位元	名稱	描述
Bit[23~17]	TX2 A7-1/D7-1	發送暫存器 2，發送地址或數據 data[7:1]的值
		0 置 0
		1 置 1
Bit[16]	Flag/D0	發送暫存器 2，發送讀寫命令或數據 data[0]的值
		0 置 0
		1 置 1
Bit[15~8]	TX1 A7-0/D7-0	發送暫存器 1，發送地址或數據 data[7:0]的值
		0 置 0
		1 置 1
Bit[7~1]	TX0 A7-1/D7-1	發送暫存器 0，發送地址或數據 data[7:1]的值
		0 置 0
		1 置 1
Bit[00]	RW/D0	發送暫存器 0，發送讀寫命令或數據 data[0]的值
		0 置 0
		1 置 1

注意:

- (1)在通訊過程中，當沒有要傳輸時，數據傳輸寄存器必須被設置為 0XFF。
因為最低位元，常期拉 Low 容易讓 SDA 總線鎖住在 0。

23.4 範常式式流程



23.5 範常式式功能

範例名稱:測試硬體 I2C

範例說明:

- (1)透過 I2C 腳位與 I2C 暫存器設定
(2)測試硬體 I2C 對 AT24C02 寫入功能

23.6 範常式說明

(主程式)

00	
01	#include "HY16F18X.h"
02	
03	
04	volatile unsigned int i,a;
05	
06	
07	int main(void)
08	{
09	I2C_INI();
11	
12	while(1)
13	{
14	i=(pio_08 & 0x80)>>7;a=i;
15	while(a==0)
16	{
17	I2C_Start();
18	//HY16F188 Hardware I2C Start
19	DrvI2C_WriteData(0XA0);
20	//AT24C02 Device ID
21	I2C_NACK();
22	DrvI2C_WriteData(0X00);
23	//AT24C02 Address=0X00
24	I2C_NACK();
25	DrvI2C_WriteData(0X55);
26	//Data1=0X55 @ Address=0X00
27	I2C_NACK();
28	DrvI2C_WriteData(0XAA);
29	//Data2=0XAA @ Address=0X01
30	I2C_NACK();
31	I2C_Stop();
32	//HY16F188 Hardware I2C Stop
33	i=(pio_08 & 0x80)>>7;a=i;
34	}
35	Delay(0x50);
36	}
37	return 0;
38	}

(副程式)

00		
01	void I2C_INI(void)	//Hardware I2C Initial
02	{	
03	pio_00=0xFF800000;	//PT1.7 input
04	pio_04=0xFF800000;	
05	pio_10=0xFF00FF03;	
06	pio_14=0xFF00FF03;	
07		
08	pio_44=0X0909FF00;	
09	i2c_00=0xFF00FF00;	//0X41000//I2C OFF
11	i2c_14=0X0000FFFF;	//0X44014
12	}	
13		
14		
15	void I2C_Start(void)	//Hardware I2C Start
16	{	
17	DrvI2C_Ctrl(1,0,0,0);	//SPIA(1000)
18	I2C_NOP();	
19	}	
20		
21	void I2C_Stop(void)	//Hardware I2C Stop
22	{	
23	DrvI2C_Ctrl(0,1,0,0);	//SPIA(0100)
24	I2C_NOP();	
25	}	
26		
27	void I2C_Write(unsigned int I2C_Data)	//Hardware I2C Write
28	{	
29	DrvI2C_WriteData(I2C_Data);	
30	i2c_04=0xFF01;	//0X41004
31	I2C_NOP();	
32	}	
33		
34	void I2C_NACK(void)	//Hardware I2C NACK
35	{	
36	DrvI2C_ClearIRQ();	
37	DrvI2C_ClearEIRQ();	
38	while(!(i2c_04 & 0X00000002));	
39	}	
40		
41	void I2C_NOP(void)	//Software I2C NOP
42	{	
43	volatile unsigned int n;	
44	for(n=0;n<=0x10;n++);asm("NOP");//0x10	
45	}	
46		

24.硬體萬年曆 HW RTC

24.1 整體總說明

實時時鐘(RTC)控制器提供使用者實時時鐘和日曆。

RTC 的時鐘源來自於連接 I/O 埠的外部 32.768 KHz 晶體或內部的 33kHz LPO 震盪器。

這個 RTC 控制器以二進制編碼的十進制(BDC)格式來呈現時/分/秒的時間資訊，以及年/月/日/星期的日期資訊。在這控制器中，有一個可編程警報中斷程式和一個定期可編程喚醒中斷程式，使系統得以自動喚醒處理低電力模式。另有一個 6 位元的數位定時晶體振盪器偏差補償機制。

功能：時間資訊(時/分/秒)以及日曆資訊(年/月/日/星期)儲存在暫存器中。

- (1)警報暫存器(年/月/日期/時/分/秒)。
- (2)所有的時間和日曆資訊皆以 BCD 格式呈現。
- (3)閏年自動補償(年度：2012~2099)。
- (4)星期計數器。
- (5)6 位元數位定時晶體振盪器偏差補償。
- (6)支援定期將 CPU 從閒置模式喚醒。
- (7)支援 8 個定期喚醒期間選項：1/128, 1/64, 1/32, 1/16, 1/8, 1/4, 1/2, and 1。
- (8)支援 12/24 小時制的時間模式。

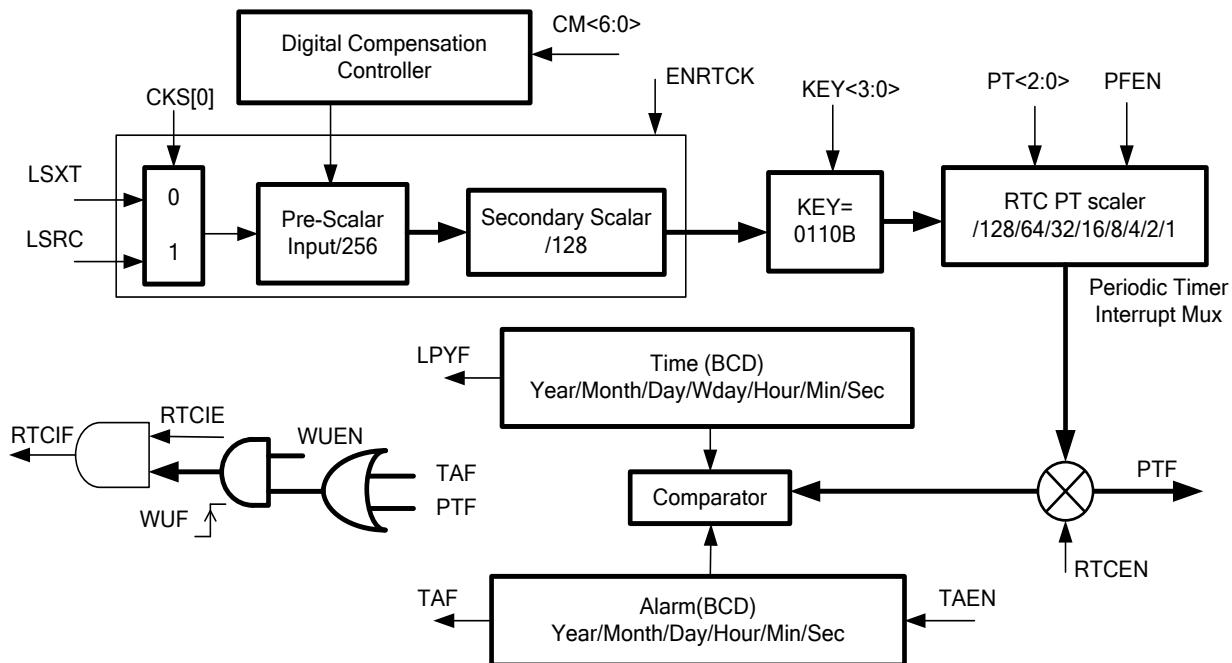


圖 24-1 RTC 架構圖

對 RTC 暫存器的存取：

因為 RTC 時鐘和系統時鐘的頻率不同，在使用者寫入一個新的資料到暫存器後，

經過兩個 RTC 時鐘脈衝後，暫存器會被更新。RTC 數據應該要經常更新。

有一個針對寫入 RTC 暫存器的保護鍵。要對 RTC 暫存器作寫入的動作時，

這個 RTKEY 鍵須被設定為`<0110>`，其他的 RTKEY 鍵值是無法對 RTC 暫存器作寫入的動作。

但要注意，因為 RTC 不會檢查對暫存器寫入的資料格式，使用者必須很小心這個寫入動作。

啟動 RTC：

要寫入一個值到 RTC 暫存器，就必須先將`<0110>`寫入到 RTKEY。

要啟動 RTC，使用者首先要檢查 LPOSC 或 LS_CLK 是否可使用。然後將 RTCEN 設定為 1。

頻率補償：

RTC 允許對時鐘輸入的數位補償。RTC 中央頻率為 32768Hz。

任何的不完美可導致頻率偏差。可用數位補償來降低頻率偏差。

補償方式為每一步驟執行 $+/-2\text{ppm}$ ，可允許最大 $+126\text{ppm}$ 及最小 -126ppm 的頻率變動。

最大的輸入頻率為 32772Hz，最小的輸入頻率為 32763Hz。

使用者可以輸入最大到 10MHz 的基準頻率，在製造期間來測量 RTC 時鐘。

測量值被計算以求得補償值。然後，這個補償值被儲存在快閃記憶體中。

一旦系統被開機後，這個補償值就會被載入到 RTCO`<6:0>`。

時間資訊：

時間儲存在 RTTHR/RTMII/RTSE 這些暫存器中。它們使用 BCD 格式。

使用者可將時間設定為 24 小時制或 12(AM/PM)制。

時間的預設值為 00:00:00 (時/分/秒)，且為 24 小制。

日曆資訊：

日曆資訊儲存在 RTYE/RTMO/RTDA/RTWDA 這些暫存器中。

它們使用 BCD 格式。閏年的演算法係由硬體來完成，

有效年份期間為 2012~2099。當 RTLPYF 為 1，就表示為閏年。

年份是使用兩個數字來表示，代表 20XX 年。系統重設後的預設日期為 12/1/1 星期天(2012 年 1 月 2 日)。

最大年度為 99，且在 99/12/31 之後年度會變成 00/1/1，但此時閏年補償就會無效。

星期計數器：

RTC 控制器提供星期的資訊。這個 WDA 值被定義為從 0 到 6，分別代表星期天到星期六。

警報中斷：

當 RTYE/RTMO/RTDA/RTHR/RTMII/RTSE

這些暫存器與 RCYE/RCMO/RCDA/RCHR/RCMI/ RCSE 這些暫存器相符合，

且 RTAEN 為 1 時，RTTAF 中斷旗標會定設為 1 紙 MCU。

定期計時器中斷：

這個定期計時器中斷有 8 個定期選項：1/128、1/64、1/32、1/16、1/8、1/4、1/2 和 1 秒。

將 RTPFEN 設定為 1 以啟動這個定期計時器中斷。這些定期的選項係由 RTPT`<2:0>`所控制。

系統喚醒中斷：

當 MCU 進入閒置模式後，它可經由系統喚醒中斷程式被喚醒。

有兩個來源可喚醒 MCU：定期計時器中斷和警告中斷。將 RTWFEN 設定為 1 以啟動這個中斷程式。

24.2 暫存器位址

RTC Register Address	31	24	23	16	15	8	7	0
RTC Base Address + 0X00 (0X41A00)	RTKEY		RTCC1		RTCC0M		RTCC0	
RTC Base Address + 0X04 (0X41A04)	RTCOM		RTCO		RTCPTM		RTPT	
RTC Base Address + 0X08 (0X41A08)	-		-		RTHRM		RTHR	
RTC Base Address + 0X0C (0X41A0C)	RTMIM		RTMI		RTSEM		RTSE	
RTC Base Address + 0X00 (0X41A10)	RTYEM		RTYE		RTMOM		RTMO	
RTC Base Address + 0X04 (0X41A14)	RTDAM		RTDA		RTWDM		RTWDA	
RTC Base Address + 0X08 (0X41A18)	-		RCHR		RCMI		RCSE	
RTC Base Address + 0X0C (0X41A1C)	-		RCYE		RCMO		RCDA	

-保留

24.3 暫存器功能

24.3.1 硬體 RTC 暫存器 RTCCR0

RTC Base Address + 0X00 (0X41A00)							
Symbol	RTCCR0 (RTC Control Register 0)						
Bit	[31:24]	[23:20]		[19]	[18]	[17]	[16]
名稱	MASK	KEY		LPYF	PTF	WUF	TAF
RW	R0W-0	RW-0		R-0	RW0-0	R-0	RW0-0
Bit	[15:08]	[7:6]	[5]	[4]	[3]	[2]	[1]
名稱	MASK	-	PTEn	WUEn	TAEn	HRF	CKS
RW	R0W-0	-			RW-0		RTCEn

位元	名稱	描述
Bit[23-20]	KEY	RTC 寄存器的密匙，上鎖後可保護寄存器，防止被寫入
		0110 寫入密匙，解除寄存器保護，只有解鎖後才能對寄存器寫入操作
		Others 上鎖，保護寄存器，不能進行寫入操作
Bit[19]	LPYF	閏年旗標
		0 當前年份不是閏年 1 當前年份是閏年
Bit[18]	PTF	定時器喚醒中斷旗標
		0 正常 1 定時器喚醒已觸發
Bit[17]	WUF	喚醒中斷旗標
		0 正常 1 喚醒中斷已觸發
Bit[16]	TAF	鬧鐘狀態旗標
		0 正常 1 鬧鐘中斷觸發
Bit[05]	PFEN	RTC 定時器計時功能開啟控制
		0 關閉 1 開啟
Bit[04]	WUFEN	RTC 哸醒功能開啟控制
		0 關閉 1 開啟
Bit[03]	TAEN	RTC 鬧鐘功能開啟控制
		0 關閉 1 開啟
Bit[02]	HRF	RTC 小時格式設置 (24/12)
		0 24 小時制 1 12 小時制(PM/AM)
Bit[01]	CKS	RTC 時脈源輸入選擇
		0 外部低速晶震源 1 內部低速晶震源
Bit[00]	RTCEN	RTC 功能開啟控制

		0	關閉 RTC 功能
		1	開啟 RTC 功能

注意事項：

- (1) RTC Clck Source Selection “CKS” 具有防呆保護,如果 CKS 選擇 LSXT 但是 LSXT 沒有 Enable 的情況下,電路會自動切換至 LSRC 作為 Clck Source。
- (2) 當 RTC 設定工作於 24 小時制的時候,RTC 的小時(Hour)單位計數範圍是 0~23 循環計數,當 RTC 設定工作於 12 小時制的時候,RTC 的小時(Hour)單位計數範圍是 0~11 循環計數
- (3) 當 HRF 控制位設定為<1>的時候,也就是工作在 12 小時制,此時如果要對 RTC 時間做寫入動作,在小時(Hour)的單位,如果超過數字 12 以上會造成 RTC 寫入無效動作。
- (4) RTC 暫存器資料寫入時需注意,如果在 HRF 控制位設定為<0>的時候,也就是工作在 24 小時制,此時寫入時間如果是大於 12 小時,資料可以正常寫入 RTC 暫存器內。而這時如果再把 HRF 控制位設定為<1>的時候,則會造成 RTC 暫存器的小時單位不斷的往上計數上去,此時,即使是設定工作在 12 小時制,小時單位計數也不會是 0~11 的循環計數,會有異常狀況發生。

24.3.2 硬體 RTC 暫存器 RTCCR1

RTC Base Address + 0X04 (0X41A04)					
Symbol	RTCCR1 (RTC Control Register 1)				
Bit	[31:24]	[23]	[22:16]		
名稱	MASK	-	CM		
RW	R0W-0	-	RW-0		
Bit	[15:08]	[7:5]	[4]	[3]	[2:0]
名稱	MASK	-	CKH	-	PT
RW	R0W-0	-	RW-0	-	RW-0

位元	名稱	描述
Bit[22~16]	CM	RTC 時脈頻率補償值設置
		0111111 +126 PPM 的晶震頻率補償 (最大值)
		0111110 +124 PPM 的晶震頻率補償
		... 遞增步長: +2 PPM 的晶震頻率補償
		0000001 +2 PPM 的晶震頻率補償
		0000000 0 PPM 的晶震頻率補償
		1000000 0 PPM 的晶震頻率補償
		1000001 -2 PPM 的晶震頻率補償
		... 遞減步長: -2 PPM 的晶震頻率補償
		1111110 -124 PPM 的晶震頻率補償
		1111111 -126 PPM 的晶震頻率補償(最小值)
Bit[4]	CKH	RTC 高速時脈源開啟控制,建議在測試模式下才開啟。
		0 關閉, 使用低速時脈源
		1 開啟, 使用高速時脈源
Bit[2~0]	PT	定時器定時喚醒時間設置
		000 1/128 s
		001 1/64 s
		010 1/32 s
		011 1/16 s
		100 1/8 s
		101 1/4 s
		110 1/2 s
		111 1 s

24.3.3 硬體 RTC 暫存器 RTCHRC

RTC Base Address + 0X08 (0X41A08)						
Symbol	RTCHRC (RTC Hour Control Register For calendar)					
Bit	[31:16]					
名稱	RSV					
RW	R-0					
Bit	[15:08]	[7]	[6]	[5:4]	[3:0]	
名稱	MASK	-	PM	10HR	1HR	
RW	R0W-0	-	RW-0			

位元	名稱	描述
Bit[6]	HRPM	萬年曆小時格式 am/pm
		0 AM 或 24 小時制
		1 PM (必須置 1, 若 HRF 被置 1)
Bit[5~4]	10HR	小時十位數的值(BCD 碼格式)
		00 0
		01 1
		10 2 (HRF=1) / HRF=0 時無效
		11 無效
Bit[3~0]	1HR	小時個位數的值(BCD 碼格式)
		0000 0
		0001 1
		0010 2
		0011 3
		0100 4
		0101 5
		0110 6
		0111 7
		1000 8
		1001 9
		其他值 無效

24.3.4 硬體 RTC 暫存器 RTCSMC

RTC Base Address + 0X0C (0X41A0C)				
Symbol	RTCSMC (RTC seconds and min Control Register For calendar)			
Bit	[31:24]	[23]	[22:20]	[19:16]
名稱	MASK	-	10MIN	1MIN
RW	R0W-0	-	RW-0	RW-0
Bit	[15:08]	[07]	[06:04]	[03:00]
名稱	MASK	-	10SEC	1SEC
RW	R0W-0	-	RW-0	RW-0

位元	名稱	描述
Bit[22~20]	10MIN	分鐘十位數的值(BCD 碼格式)
		000 0
		001 1
		010 2
		011 3
		100 4
		101 5
		110 6
		111 無效
		分鐘個位數的值(BCD 碼格式)
Bit[19~16]	1MIN	0000 0
		0001 1
		0010 2
		0011 3
		0100 4
		0101 5
		0110 6
		0111 7
		1000 8
		1001 9
Bit[6~4]	10SEC	其他值 無效
		秒鐘十位數的值(BCD 碼格式)
		000 0
		001 1
		010 2
		011 3
		100 4
		101 5
		110 6
		111 無效
Bit[3~0]	1SEC	秒鐘個位數的值(BCD 碼格式)
		0000 0
		0001 1
		0010 2
		0011 3
		0100 4
		0101 5
		0110 6
		0111 7
		1000 8
		1001 9
		其他值 無效

24.3.5 硬體 RTC 暫存器 RTCYMC

RTC Base Address + 0X10 (0X41A10)			
Symbol	RTCYMC (RTC Year and Month Control Register For Calendar)		
Bit	[31:24]	[23:20]	[19:16]
名稱	MASK	10YEAR	1YEAR
RW	R0W-0	RW-1	RW-2
Bit	[15:08]	[07:05]	[04]
名稱	MASK	-	10MO
RW	R0W-0	-	RW-0
			RW-1

位元	名稱	描述
Bit[23~20]	10YEAR	年份十位數的值 (BCD 碼格式)
		0000 0
		0001 1
		0010 2
		0011 3
		0100 4
		0101 5
		0110 6
		0111 7
		1000 8
Bit[19~16]	1YEAR	年份個位數的值 (BCD 碼格式)
		0000 0
		0001 1
		0010 2
		0011 3
		0100 4
		0101 5
		0110 6
		0111 7
		1000 8
Bit[4]	10MO	月份十位數的值(BCD 碼格式)
		0 0
		1 1
Bit[3~0]	1MO	月份個位數的值(BCD 碼格式)
		0000 0
		0001 1
		0010 2
		0011 3
		0100 4
		0101 5
		0110 6
		0111 7
		1000 8
		1001 9
		其他值 無效

24.3.6 硬體 RTC 暫存器 RTCDWC

RTC Base Address + 0X14 (0X41A14)				
Symbol	RTCDWC (RTC Date and week Control Register For calendar)			
Bit	[31:24]	[23:22]	[21:20]	[19:16]
名稱	MASK	-	10DAT	1DAT
RW	R0W-0	-	RW-0	RW-1
Bit	[15:08]	[07:03]		[02:00]
名稱	MASK	-		WDA
RW	R0W-0	-		RW-0

位元	名稱	描述	
Bit[21~20]	10DAT	日期十位數的值 (BCD 碼格式)	
		00	0
		01	1
		10	2
		11	3
Bit[19~16]	1DAT	日期個位數的值 (BCD 碼格式)	
		0000	0
		0001	1
		0010	2
		0011	3
		0100	4
		0101	5
		0110	6
		0111	7
		1000	8
		1001	9
		其他值	無效
Bit[2~0]	WDA	星期的值 (BCD 碼格式)	
		000	Sunday 星期日
		001	Monday 星期一
		010	Tuesday 星期二
		011	Wednesday 星期三
		100	Thursday 星期四
		101	Friday 星期五
		110	Saturday 星期六
		111	無效

24.3.7 硬體 RTC 暫存器 RTCHRA

RTC Base Address + 0X18(0X41A18)					
Symbol	RTCHRA (RTC Hour and min and seconds Control Register for alarm)				
Bit	[31:24]	[23]	[22]	[21:20]	[19:16]
名稱	RSV	-	CPM	10CHR	1CHR
RW	R-0	-		RW-0	
Bit	[15]	[14:12]	[11:8]	[7]	[6:4]
名稱	-	10CMI	1CM1	-	10CSE
RW	-	RW-0		-	RW-0

位元	名稱	描述
Bit[22]	CPM	鬧鐘顯示的格式為 am/pm
		0 AM 或者 24 小時制
		1 PM (當 HRF=1 時，該位元需要被置 1)
Bit[21~20]	10CHR	鬧鐘模式下小時十位元數的值 (BCD 碼格式)
		00 0
		01 1
		10 2 (HRF=1) / HRF=0 時無效
		11 無效
Bit[19~16]	1CHR	鬧鐘模式下小時個位元數的值 (BCD 碼格式)
		0000 0
		0001 1
		0010 2
		0011 3
		0100 4
		0101 5
		0110 6
		0111 7
		1000 8
		1001 9
		其他值 無效
Bit[14~12]	10CMI	鬧鐘模式下分鐘十位元數的值 (BCD 碼格式)
		000 0
		001 1
		010 2
		011 3
		100 4
		101 5
		110 6
		111 無效

位元	名稱	描述
Bit[11~8]	1CMI	鬧鐘模式下分鐘的個位數的值 (BCD 碼格式)
		0000 0
		0001 1
		0010 2
		0011 3
		0100 4
		0101 5
		0110 6
		0111 7
		1000 8
Bit[6~4]	10CSE	鬧鐘模式下秒鐘十位元數的值(BCD 碼格式)
		000 0
		001 1
		010 2
		011 3
		100 4
		101 5
		110 6
		111 無效
		鬧鐘模式下秒鐘個位元數的值 (BCD 碼格式)
Bit[3~0]	1CSE	0000 0
		0001 1
		0010 2
		0011 3
		0100 4
		0101 5
		0110 6
		0111 7
		1000 8
		1001 9
其他值 無效		

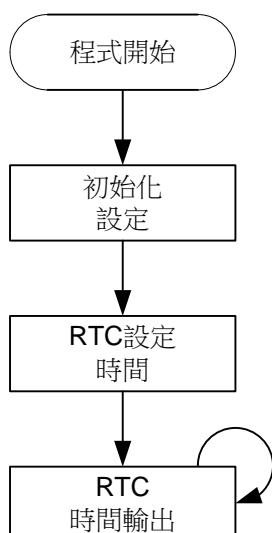
24.3.8 硬體 RTC 暫存器 RTCYMDA

RTC Base Address + 0X1C(0X41A1C) RTCYMDA (RTC Year /month/date Control Register For alarm)					
Symbol	[31:24]	[23:20]	[19:16]		
Bit	RSV	10CYE	1CYE		
RW	R-0	RW-1	RW-2		
Bit	[15:13]	[12]	[11:8]	[07]	[06:04]
名稱	-	10CMO	1CMO	-	10CDAT
RW	-	RW-0	RW-1	-	RW-0
					RW-1

位元	名稱	描述	
Bit[23~20]	10CYE	鬧鐘模式下年份十位元數的值 (BCD 碼格式)	
		0000	0
		0001	1
		0010	2
		0011	3
		0100	4
		0101	5
		0110	6
		0111	7
		1000	8
		1001	9
		其他值	無效
Bit[19~16]	1CYE	鬧鐘模式下年份個位元數的值 (BCD 碼格式)	
		0000	0
		0001	1
		0010	2
		0011	3
		0100	4
		0101	5
		0110	6
		0111	7
		1000	8
		1001	9
		其他值	無效
Bit[12]	10CMO	鬧鐘模式下月份十位元數的值 (BCD 碼格式)	
		0	0
		1	1

位元	名稱	描述	
Bit[11~8]	1CMO	鬧鐘模式下月份個位元數的值 (BCD 碼格式)	
		0000	0
		0001	1
		0010	2
		0011	3
		0100	4
		0101	5
		0110	6
		0111	7
		1000	8
Bit[5~4]	10CDAT	鬧鐘模式下日期十位元數的值 (BCD 碼格式)	
		00	0
		01	1
		10	2
		11	3
Bit[3~0]	1CDAT	鬧鐘模式下日期個位元數的值 (BCD 碼格式)	
		0000	0
		0001	1
		0010	2
		0011	3
		0100	4
		0101	5
		0110	6
		0111	7
		1000	8

24.4 範常式式流程



24.5 範常式式功能

範例名稱:測試硬體 RTC

範例說明:測試硬體 RTC 萬年曆功能

24.6 範常式式說明

(主程式)

00		
01	#include "HY16F18X.h"	
02	unsigned int sec,min,hour,week,day,month,year;	
03		
04	void Delay(unsigned int num);	
05	void RTC_Initial(void);	
06		
07	int main(void)	
08	{	
09	RTC_Initial();	//RTC 初始化(包含時間設定)
11		
12	while(1)	
13	{	
14	asm("NOP");	
15	S_DRVRTC_TIME_DATA_T sCurTime;	//RTC 回讀時間設定
16	DrvRTC_Read(DRVRTC_CURRENT_TIME,&sCurTime);	//RTC 回讀時間資料
17		
18	sec=sCurTime.u32cSecond;	//回讀_秒
19	min=sCurTime.u32cMinute;	//回讀_分
20	hour=sCurTime.u32cHour;	//回讀_小時
21	week=sCurTime.u32cDayOfWeek;	//回讀_星期
22	day=sCurTime.u32cDay;	//回讀_日期
23	month=sCurTime.u32cMonth;	//回讀_月份
24	year=sCurTime.u32Year;	//回讀_年份
25	asm("NOP");	
26	}	
27	return 0;	
28	}	
29		

(副程式)

00		
01	void RTC_Initial(void)	
02	{	
03	//RTC CLK;	
04	clk_08=0x8080ff00;	
05	//RTC KEY;	
06	rtc_00=0xff60ff00;	
07	asm("NOP");	
08		
09	DrvRTC_WriteEnable();	
11	//DrvRTC_ClockSource(E_INTERNAL_CLOCK);//35KHz	
12	DrvRTC_ClockSource(E_EXTERNAL_CLOCK);//32768Hz	
13		
14	DrvRTC_PeriodicTimeEnable(0);//set 1/128	
15	DrvRTC_Enable();	
16	DrvRTC_HourFormat(0);	
17		
18	asm("NOP");	
19	S_DRVRTC_TIME_DATA_T sCurTime;//setting start	
20	DrvRTC_Read(DRVRTC_CURRENT_TIME,&sCurTime);	
21	sCurTime.u8cClockDisplay=1;	
22	sCurTime.u8cAmPm=0;	
23	sCurTime.u32cSecond=19;	//設定_秒
24	sCurTime.u32cMinute=50;	//設定_分
25	sCurTime.u32cHour=10;	//設定_小時
26	sCurTime.u32cDayOfWeek=5;	//設定_星期
27	sCurTime.u32cDay=9;	//設定_日期
28	sCurTime.u32cMonth=8;	//設定_月份
29	sCurTime.u32Year=2013;	//設定_年份
30	sCurTime.u8IsEnableWakeUp=0;	
31	DrvRTC_Write(DRVRTC_CURRENT_TIME,&sCurTime);	
32	asm("NOP");	
33	}	
34		

25.省電模式介紹

25.1 整體總說明

本節說明不同的電源模式，能夠開啟哪些功能模塊。

活動模式是所有的週邊電路，都能被打開，且 MCU 是 HS_CK 或 LS_CK 時脈；在此模式下，系統能自由地跳到其他模式，且回應時間最短。

低電力模式是所有的類比電路都能被打開，且 MCU 是 LS_CK 時脈；在此模式下，MCU 以最低的頻率運作，系統可經由指令的執行跳到其他的模式。

有 4 種省電模式，可讓 MCU 停止執行指令。

這些模式可由中斷(interrupt)來跳脫。一旦 interrupt 被觸發，2MHz 的內部時脈源就會開啟。然後，MCU 就會依據這個時脈源來執行，且系統離開這些省電模式。

晶片進入省電模式前，必須開啟任何一個中斷向量，否則無法達到省電的功效。尤其需要注意在不同的省電模式下，只能開啟一些功能模塊，只有一些功能才能將晶片從省電模式喚醒，如在休眠模式下，定時器中斷是無效的，且只能通過通訊中斷、IO 口外部中斷及復位來喚醒晶片。

25.2 暫存器位址

IP	運行模式	等待模式
	全速運行	等待
MCU clock	一直有效	可設置
Internal Low OSC	一直有效	一直有效
Other OSC	可設置	可設置
Other IPS	可設置	可設置
Pin Reset	可設置	可設置
Pin Interrupt	可設置	可設置
SPI Slave	可設置	可設置
I2C Slave	可設置	可設置
BOR/POR Reset	一直有效	一直有效

IP	待機模式	休眠模式
	待機	休眠
MCU clock	可設置	關閉
Internal Low OSC	一直有效	關閉
Other OSC	可設置	關閉
Other IPS	可設置	關閉
Pin Reset	可設置	可設置
Pin Interrupt	可設置	可設置
SPI Slave	可設置	可設置
I2C Slave	可設置	可設置
BOR/POR Reset	一直有效	一直有效
Operation Current(uA)	5uA	2.5uA
Wake Up to Full Speed Time	50us	64ms

25.3 暫存器功能

Interrupt/Reset Mode	Sleep Mode	Idle Mode	Wait Mode	Note
	Wakeup	Wakeup	Wakeup	
Power On Reset	V	V	V	Reset
PT 4.0 Reset	V	V	V	Reset
WDT Reset Type			V	WDT Reset Type
I2C TX IRQ		V	V	I2CIE
I2C RX IRQ	V	V	V	I2CIE
I2C Error IRQ			V	I2CEIE
UART TX IRQ			V	UTXIE
UART RX IRQ	V	V	V	URXIE
SPI TX IRQ			V	STXIE
SPI RX IRQ	V	V	V	SRXIE
RTC IRQ		V	V	RTCIE
WDT IRQ			V	WDTIE
TMA IRQ			V	TMAIE
TMB IRQ			V	TMBIE
TMC IRQ			V	TMCIE
ADC IRQ			V	ADCIE
CMP IRQ	V	V	V	CPIE
OPAMP IRQ			V	OPOIE
PT1 IRQ	V	V	V	PT1IE
PT2 IRQ	V	V	V	PT2IE
Debug Exception			V	EDM

SYS Base Address + 0X04 (0X40104)								
Symbol	SYS0 (SYS Control Register 0)							
Bit	[31:08]	[7:6]	[5]	[4]	[3]	[2]	[1]	[0]
名稱	MASK	-	F1	IDLE	F2	F3	F4	F5
RW	R0W-0	-			RW0-0			RW0-1

F1~F5 at CH6 System Register

位元	名稱	描述
Bit[4]	IDLE	待機模式(IDLE)使能控制
		0 休眠模式 (sleep mode)
		1 待機模式 (IDLE mode)

Mode	Setting	描述
Wait Mode	sys_04=0xFF10; asm("standby 0");	//Idle Set //Wait Mode
Idle Mode	sys_04=0xFF10; asm("standby 1");	//Idle Set //Idle Mode
Sleep Mode	sys_04=0xFF00; asm("standby 1");	//Sleep Set //Sleep Mode

SYS_04 Address = 0X40104

26. HYCON Note 32

Type	描述	Initial
-	No Use	
RSV.	Reserve	
X	Unknown	
W	Write	
R	Read	
R0	Only Read 0	
R1	Only Read 1	
W0	Only Write 0	
W1	Only Write 0	
RW-0	Read/ Write	Initial 0
RW-1	Read/ Write	Initial 1
R0W-0	Read 0/ Write	Initial 0
R1W-1	Read 1/ Write	Initial 1
R-X	Read	Initial 1 or 0 Unknown

27.修訂歷史

下表顯示版本修訂為全面性的，標點符號和字體不包含在修訂範圍內。

Version	Page	Summary of Changes	Date
V0.1	ALL	First Edition	20130520
V0.2	ALL	The New First Edition	20130720
V0.3	ALL	ADD Wait Mode @ CH4	20130930
V0.4	ALL	中文化	20140310
V0.5	ALL	DAC 統一更正名稱為 8-bit Resistance Ladders(數位電阻器)	20150609
	ALL	HSXT 外部震盪器修正最高範圍為 16MHz	
	CH5	圖 5.1 功能方塊圖內容描述 REFOI 更正名稱 REFO_I	
	CH7	針對中斷向量優先權級別加強說明	
	CH17	ADC 網路輸入端的 OPO 更正為 OPOI, REFO 更正為 REFO_I	
	CH18	OPAMP 網路 OPNS[3]更正為 OPOI, OPNS[4]更正為 OPO, R2ROP 輸出描述修正	
	CH19	8-bit Resistance Ladders 網路輸入端的 REFO 更正為 REFO_I	
V0.6	CH22	Parity 位元功能描述修正. 針對 ENADD 位元功能描述增加補充說明	20150911
	CH24	修正 RTC 章節的 HRF 控制位描述, 並新增 RTC 使用注意事項	
V0.7	CH21	新增與修正 SPI 章節暫存器內容描述	20151116
V0.8	ALL	移除 GPIO PT4.0 第 15 章節相關描述, 並且移除文章內有關 PT4.0 功能描述.	20160322
	CH22.1.1 &CH22.1.3&CH22.3.4	修正與新增 UART 使用 Auto BaudRate 判斷機制說明 原 BRGRH 描述為 4bits, 更正為 5bits	
V0.9	P8	更新晶片說明方塊圖	2016/05/24
	ALL	移除 HSRC 20MHz 選項, HSRC 最高支援頻率為 10MHz, 修正 LSRC 頻率為 33kHz	
	P140	修正 TRMT 控制位描述, 正確為: 0 : 發送移位暫存器為滿 1 : 發送移位暫存器為空	