



HY16F3910

User's Guide

高精度混合信号处理控制器

4x44 ~ 8x40 LCD Driver

32-bit 低功耗微控制器

21-bit ENOB $\Sigma\Delta$ ADC

128k Byte Flash ROM

Table of Contents

1. 芯片概述	13
1.1. 简介	13
1.2. 型态说明表	13
2. 功能概述	14
2.1. 方块图	14
2.2. 中央处理器核心方框图	15
3. 内存结构	16
3.1. 内存说明	16
3.2. 内存地址	17
3.3. 静态随机存取内存(SRAM)	17
3.4. 闪存(Flash ROM)	17
3.5. 总线接口单元	18
3.6. 开机内存(Boot ROM)	19
3.7. 嵌入式除错模块(EDM)	19
4. 系统寄存器	20
4.1. 整体总说明	20
4.2. 寄存器地址	20
4.3. 寄存器功能	20
4.3.1. SOC 寄存器	20
5. 电源管理	22
5.1. 整体总说明	22

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



5.1.1.	芯片工作电压 VDD5V 及 VDD15:	22
5.1.2.	VDDA 电压:	22
5.1.3.	芯片上电电压检测电路(BOR1、BOR2):	23
5.1.4.	参考电压和共模电压(REFO):	24
5.1.5.	LVD 低电压检测:	25
5.1.6.	各部份模块使用的电压源	26
5.2.	寄存器地址	27
5.3.	寄存器功能	27
5.3.1.	电源管理寄存器 0	27
5.3.2.	电源管理寄存器 1	28
5.3.3.	电源管理寄存器 2	29
6.	频率系统	30
6.1.	整体总说明	30
6.1.1.	外部振荡器	30
6.1.2.	内部 RC 振荡器 HAO 与 LPO	30
6.1.3.	工作频率源配置	31
6.2.	寄存器地址	33
6.3.	寄存器功能	33
6.3.1.	频率系统寄存器 0	33
6.3.2.	频率系统寄存器 1	34
6.3.3.	频率系统寄存器 2	35
6.3.4.	频率系统寄存器 3	37
6.3.5.	频率系统寄存器 4	38
6.3.6.	频率系统寄存器 5	39
6.3.7.	频率系统寄存器 6	40
6.3.8.	频率系统寄存器 7	41
6.3.9.	频率系统寄存器 8	41
7.	中断控制系统	42
7.1.	整体总说明	42
7.2.	寄存器地址	43
7.3.	寄存器功能	43

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



7.3.1.	中断控制寄存器 0	43
7.3.2.	中断控制寄存器 1	45
7.3.3.	中断控制寄存器 2	47
7.3.4.	中断控制寄存器 3	48
7.3.5.	中断控制寄存器 4	49
7.3.6.	中断控制寄存器 5	49
7.3.7.	中断控制寄存器 6	50
7.3.8.	中断控制寄存器 7	51
7.3.9.	中断控制寄存器 8	51
8.	看门狗 WDT	53
8.1.	整体总说明	53
8.1.1.	WDT 的设置说明	53
8.2.	寄存器地址	54
8.3.	寄存器功能	54
8.3.1.	WDT 寄存器	54
9.	定时器 TIMER A	55
9.1.	整体总说明	55
9.2.	寄存器地址	56
9.3.	寄存器功能	56
9.3.1.	Timer A 寄存器	56
10.	定时器 TIMER B	58
10.1.	整体总说明	58
10.1.1.	定时计数器模式	59
10.1.2.	系统 PWM 功能模式	64
10.2.	寄存器地址	76
10.3.	寄存器功能	76

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



10.3.1. Timer B 寄存器 0	76
10.3.2. Timer B 寄存器 1	78
10.3.3. Timer B 寄存器 2	78
10.3.4. Timer B 寄存器 3	78
11. 定时器 TIMER B2	79
11.1. 整体总说明	79
11.2. 寄存器地址	81
11.3. 寄存器功能	81
11.3.1. Timer B2 寄存器 0	81
11.3.2. Timer B2 寄存器 1	82
11.3.3. Timer B2 寄存器 2	82
11.3.4. Timer B2 寄存器 3	83
11.3.5. Timer B2 寄存器 4	83
12. 定时器 TIMER C	84
12.1. 整体总说明	84
12.2. 寄存器地址	86
12.3. 寄存器功能	86
12.3.1. Timer C 寄存器 0	86
12.3.2. Timer C 寄存器 1	87
13. 通用 GPIO PT1 管理	88
13.1. 整体总说明	88
13.2. 寄存器地址	89
13.3. 寄存器功能	90
13.3.1. PT1 寄存器 0	90
13.3.2. PT1 寄存器 1	90
13.3.3. PT1 寄存器 2	91
13.3.4. PT1 寄存器 3	91
14. 通用 GPIO PT2 管理	94
14.1. 整体总说明	94

HY16F3910 User's Guide

21-bit ENOB Σ ADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

14.2.	寄存器地址	95
14.3.	寄存器功能	96
14.3.1.	PT2 寄存器 0	96
14.3.2.	PT2 寄存器 1	96
14.3.3.	PT2 寄存器 2	97
14.3.4.	PT2 寄存器 3	97
15.	通用 GPIO PT3 管理.....	100
15.1.	整体总说明	100
15.2.	寄存器地址	102
15.3.	寄存器功能	102
15.3.1.	PT3 寄存器 0	102
15.3.2.	PT3 寄存器 1	102
15.3.3.	PT3 寄存器 2	103
15.3.4.	PT3 寄存器 3	103
15.4.	模拟数字复用功能切换注意事项.....	107
16.	通用 GPIO PT6 管理.....	108
16.1.	整体总说明	108
16.2.	寄存器地址	109
16.3.	寄存器功能	109
16.3.1.	PT6 寄存器 0	109
16.3.2.	PT6 寄存器 1	111
16.3.3.	PT6 寄存器 2	112
16.3.4.	PT6 寄存器 3	113
17.	通用 GPIO PT7 管理.....	115
17.1.	整体总说明	115
17.2.	寄存器地址	116
17.3.	寄存器功能	116

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



17.3.1. PT7 寄存器 0	116
17.3.2. PT7 寄存器 1	117
17.3.3. PT7 寄存器 2	118
17.3.4. PT7 寄存器 3	119
18. 通用 GPIO PT8 管理.....	121
18.1. 整体总说明	121
18.2. 寄存器地址	122
18.3. 寄存器功能	122
18.3.1. PT8 寄存器 0	122
18.3.2. PT8 寄存器 1	123
18.3.3. PT8 寄存器 2	124
18.3.4. PT8 寄存器 3	126
19. 通用 GPIO PT9 管理.....	128
19.1. 整体总说明	128
19.2. 寄存器地址	129
19.3. 寄存器功能	129
19.3.1. PT9 寄存器 0	129
19.3.2. PT9 寄存器 1	130
19.3.3. PT9 寄存器 2	131
19.3.4. PT9 寄存器 3	133
20. 通用 GPIO PT10 管理.....	135
20.1. 整体总说明	135
20.2. 寄存器地址	136
20.3. 寄存器功能	136
20.3.1. PT10 寄存器 0	136
20.3.2. PT10 寄存器 1	137
20.3.3. PT10 寄存器 2	138
20.3.4. PT10 寄存器 3	140
21. 通用 GPIO PT13 管理.....	142

HY16F3910 User's Guide

21-bit ENOB $\Sigma\Delta$ ADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



21.1.	整体总说明	142
21.2.	寄存器地址	143
21.3.	寄存器功能	143
21.3.1.	PT13 寄存器 0	143
21.3.2.	PT13 寄存器 1	144
21.3.3.	PT13 寄存器 2	145
21.3.4.	PT13 寄存器 3	146
22.	通用 GPIO 复用功能管理	148
22.1.	整体总说明	148
22.2.	寄存器地址	151
22.3.	寄存器功能	151
22.3.1.	GPIO 复用功能控制寄存器 0	151
22.3.2.	GPIO 复用功能控制寄存器 1	153
22.3.3.	GPIO 复用功能控制寄存器 2	154
22.3.4.	GPIO 复用功能控制寄存器 3	154
23.	$\Sigma\Delta$ 24 位模拟数字转换器 ADC	156
23.1.	整体总说明	156
23.1.1.	全差动信号输入端	158
23.1.2.	内置增益放大器	159
23.1.3.	参考电压输入通道	159
23.1.4.	输入信号输入偏压	160
23.1.5.	梳状滤波器	161
23.1.6.	温度传感器 TPS	161
23.1.7.	ADC 输入阻抗(R_{ADC})说明	164
23.1.8.	ADC 操作说明	165
23.2.	寄存器地址	166
23.3.	寄存器功能	166

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



23.3.1. ADC 寄存器 0	166
23.3.2. ADC 寄存器 1	167
23.3.3. ADC 寄存器 2	168
23.3.4. ADC 寄存器 3	169
23.3.5. ADC 寄存器 4	169
24. 串行通信 SPI	171
24.1. 整体总说明	171
24.2. 寄存器地址	176
24.3. 寄存器功能	176
24.3.1. SPI 寄存器 0	176
24.3.2. SPI 寄存器 1	177
24.3.3. SPI 寄存器 2	178
24.3.4. SPI 寄存器 3	178
25. 异步串行通信 UART	180
25.1. 整体总说明	180
25.1.1. Baud Rate 传输波特率	180
25.1.2. 自动波特率功能(Auto Baudrate detection)	180
25.1.3. 通信 IO 引脚	181
25.2. 寄存器地址	182
25.3. 寄存器功能	182
25.3.1. UART 寄存器 0	182
25.3.2. UART 寄存器 1	183
25.3.3. UART 寄存器 2	184
25.3.4. UART 寄存器 3	184
25.4. UART 使用说明	185
26. 异步串行通信 UART2	189
26.1. 整体总说明	189
26.2. 寄存器地址	189
26.3. 寄存器功能	189

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



26.3.1.	UART2 寄存器 0.....	189
26.3.2.	UART2 寄存器 1.....	191
26.3.3.	UART2 寄存器 2.....	191
26.3.4.	UART2 寄存器 3.....	192
26.4.	UART2 使用说明	192
27.	通用 I²C 通信接口	193
27.1.	整体总说明	193
27.1.1.	通信 I ² C 接口特性	193
27.1.2.	数据传输率的计算:.....	196
27.1.3.	超时控制 Time-out function (Time-Out):.....	196
27.1.4.	I ² C 通信引脚	197
27.1.5.	通信 I ² C 接口流程	197
27.1.6.	通信 I ² C Master TX 流程.....	199
27.1.7.	通信 I ² C Master RX 流程.....	200
27.1.8.	通信 I ² C Slaver TX 流程.....	201
27.1.9.	通信 I ² C Slaver RX 流程	202
27.1.10.	通信 I ² C General Call 流程.....	203
27.2.	寄存器地址	204
27.3.	寄存器功能	204
27.3.1.	I ² C 寄存器 0	204
27.3.2.	I ² C 寄存器 1	205
27.3.3.	I ² C 寄存器 2	206
27.3.4.	I ² C 寄存器 3	207
27.3.5.	I ² C 寄存器 4	209
27.3.6.	I ² C 寄存器 5	209
27.4.	I ² C 使用说明	211
27.4.1.	I ² C 初始化说明	211
27.4.2.	I ² C 操作流程说明	211
27.5.	I ² C General Call Mode	221
27.6.	10 Bit Addressing Mode	222

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



27.6.1.	I ² C 10 Bit Addressing Mode 说明	222
27.6.2.	10 Bit Addressing 数据写入流程说明	223
27.6.3.	10 Bit Addressing 数据读出流程说明	224
27.6.4.	10 Bit Addressing 数据被读出流程说明	225
27.7.	3 Byte Data Mode	226
27.7.1.	I ² C 3 Byte Date Mode 说明	226
27.7.2.	I ² C 3 Byte Date 连续读取流程说明	226
27.7.3.	I ² C 3 Byte Date 数据被读取流程说明	226
28.	硬件万年历 HW RTC	227
28.1.	整体总说明	227
28.2.	RTC 寄存器地址	229
28.3.	RTC 寄存器功能	229
28.3.1.	RTC 寄存器 0	229
28.3.2.	RTC 寄存器 1	231
28.3.3.	RTC 寄存器 2	232
28.3.4.	RTC 寄存器 3	233
28.3.5.	RTC 寄存器 4	235
28.3.6.	RTC 寄存器 5	237
28.3.7.	RTC 寄存器 6	238
28.3.8.	RTC 寄存器 7	240
29.	省电模式介绍	242
29.1.	整体总说明	242
30.	液晶驱动器 LCD	244
30.1.	整体总说明	244
30.2.	LCD 寄存器地址	245
30.3.	寄存器功能	245
30.3.1.	LCD 寄存器 0	245
30.3.2.	LCD 寄存器 1	246
30.3.3.	LCD 寄存器 2	247
30.4.	LCD RAM 功能	248
30.5.	LCD 省电功能	248

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



31. 修订记录	249
----------------	-----

1. 芯片概述

1.1. 简介

HY16F3910 是一款带液晶驱动电路(Liquid Crystal Display, LCD)、低功耗、高效能混合信号处理微控制器 (Mixed Signal Micro Controller, MCU) 适用于做精准的测量和控制,芯片可工作在 2.0V 到 5.5V 的宽电压范围,最高可运行到 16MHz,内建最大 128k Byte 的嵌入式闪存(Flash ROM),以及 8k Byte 静态随机存取内存 (SRAM)。HY16F3910 集成了高精度 24-bit 调变模拟转换器(ΣADC),硬件实现的实时时钟(Hardware RTC),提供高性能外围接口引脚,如 UART、SPI、I²C、GPIO,及内建的电源管理系统等功能。支持低电压检测,多个外部引脚唤醒功能。且支持 Andes C/C++开发平台(AndeSight)的 32-bit 微控制器。丰富的资源可以让设计者完成一个低电压低成本的混合信号处理系统。

这款控制器中的模拟前端电路包含了一个超低噪音 24-bit 模拟数字转换器(ADC)是嵌入式的。最大输出率为 31k SPS 且具 21-bit 有效位数(ENOB, Effective Number Of Bit),最小可解析输入信号为 65nV RMS Noise (Root- Mean- Square),ADC 前端内建可程序放大器 PGA,最大输入放大倍率高达 128 倍。

电源管理提供可选择的模拟电路调节电压,可做为电压基准源,以及传感器电源驱动。CPU 核工作电源也是由内部的线性稳压电源所提供。128k Byte 嵌入式闪存,可用来执行程序及储存数据。数据也可以在程序执行期间被储存在闪存中。同时内建 8k Byte 静态随机存取内存供系统使用。

采用 32-bit 高效能混合信号处理微控制器核心,可以使每一频率周期(Clock Cycle)执行一个指令,最高可达 16MIPS(Millions of Instructions Per Second),具低耗电量指标。纭康科技提供了一些容易使用的程序撰写工具,使用者可在 AndeSight 开发平台下撰写 C 语言或汇编语言的程序。芯片带有电路仿真功能,提供容易除错的环境。芯片可在 2.0V 到 5.5V 的数字工作电压, -40°C 到 85°C 的温度环境下工作。

1.2. 型态说明表

寄存器位型态说明表

设定型态	描述	初始值
-	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 1	
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
[]	Register length (寄存器长度)	
< >	Register value (寄存器内容)	
ABC[7:0]	ABC register had 0 to 7bit (ABC 寄存器总共有 8 位)	
ABC<111>	ABC register had 3bit and value had 111 of binary (ABC 寄存器总共有 3 位,内容为二进制 111)	
ABC<11x>	x : can be neglected, it can be set as 1 or 0 (ABC 寄存器总共有 3 位,内容为二进制。可为 110 或 111)	

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

2. 功能概述

2.1. 方块图

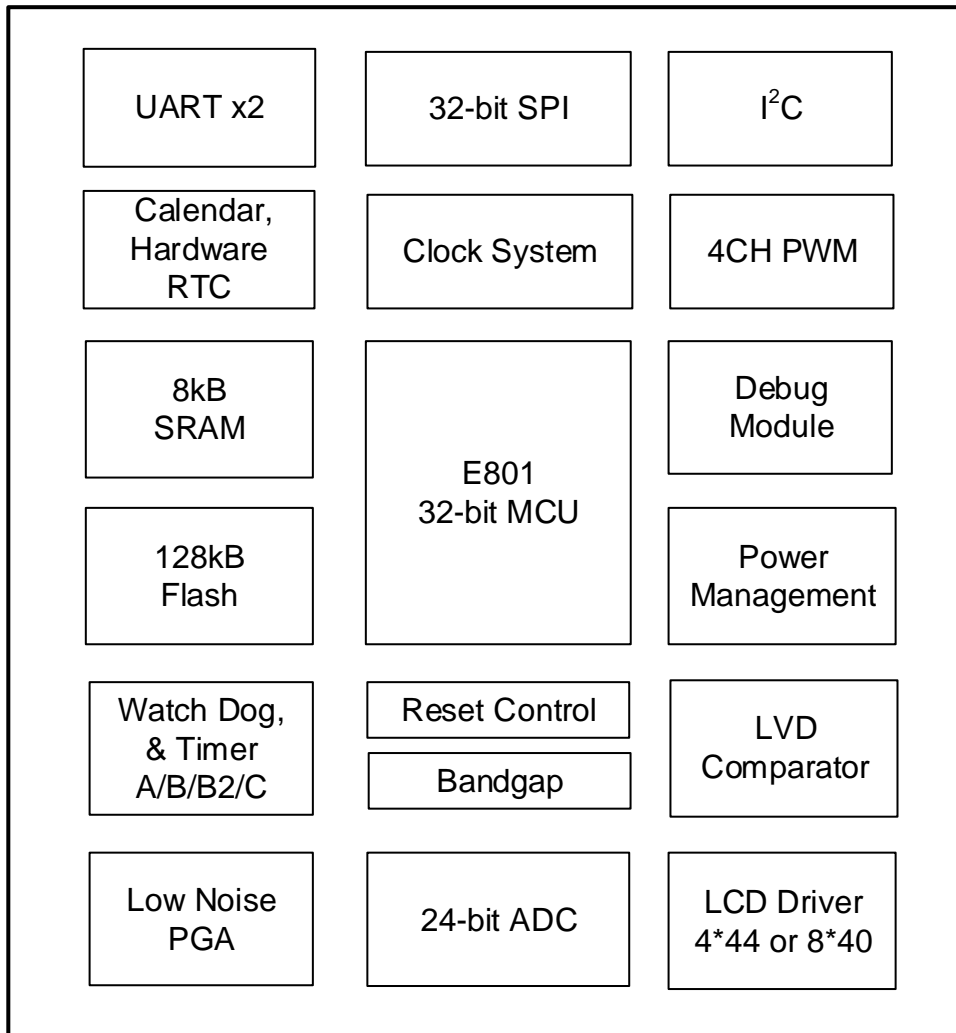


图 2-1 芯片功能架构图

2.2. 中央处理器核心方框图

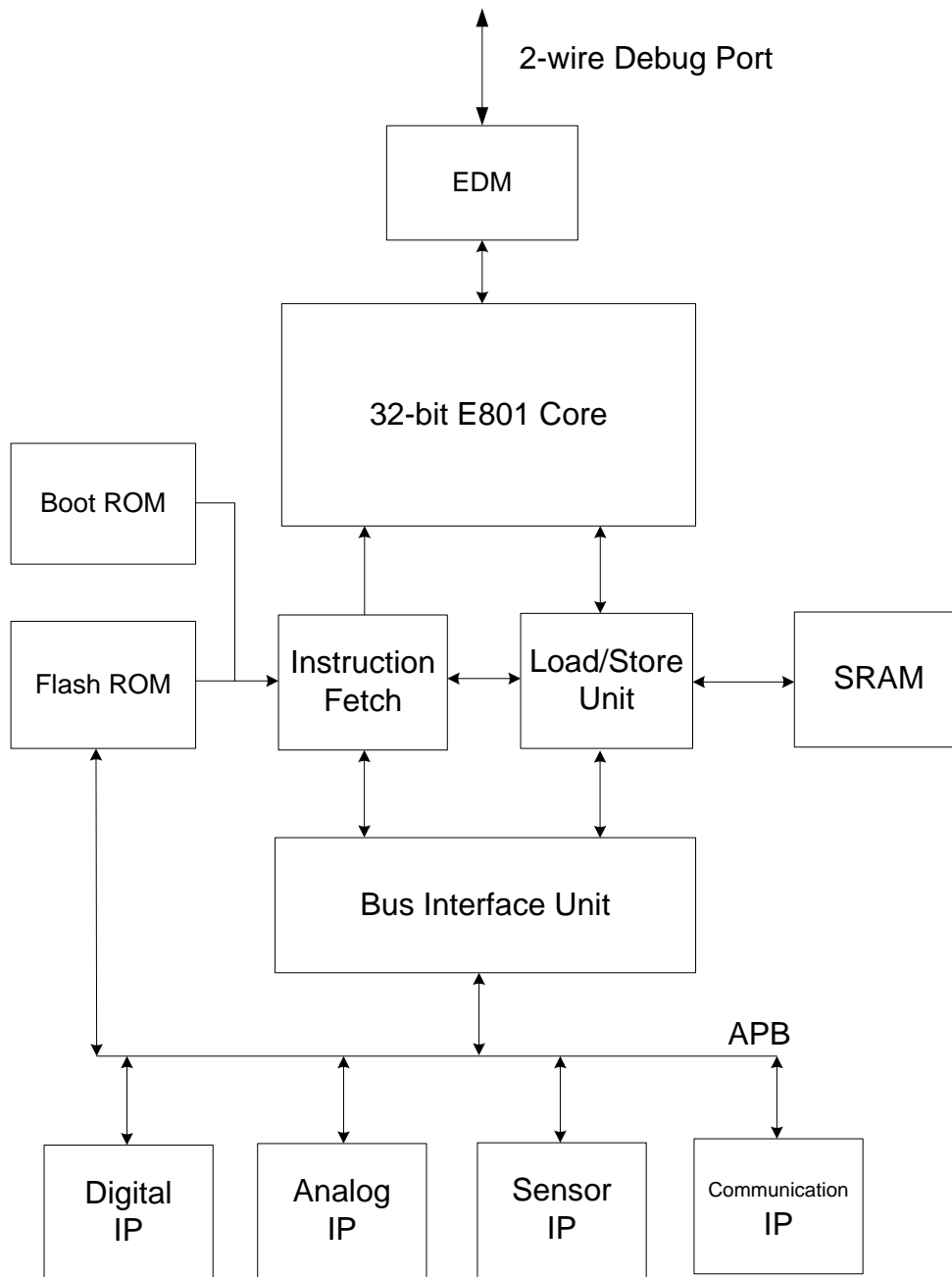


图 2-2 中央处理器核心方框图

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

3. 内存结构

3.1. 内存说明

HY16F3910 系列产品所使用的中央处理器(CPU)

内核是 Andes E801 32-bit CPU。微控制器的存储器地址分配如下：

0x00000 to 0x01FFFF 静态随机存取内存, SRAM (8k Byte)

0x40000 to 0x4FFFFF 系统控制寄存器, SOC Register

0x80000 to 0x81FFFF 开机区块内存, Boot ROM (8k Byte) : 可支持(Timeout Entry)4 线式与(CheckPin Entry) 5 线式 UART 传输 ROM ISP Bootloader 功能. 芯片预设不开启此功能, 需透过烧录器进行烧录设定或代烧程序才能开启此功能.

0x90000 to 0xAFFFFF 主程序区块闪存, Main Program Flash ROM (128k Byte)

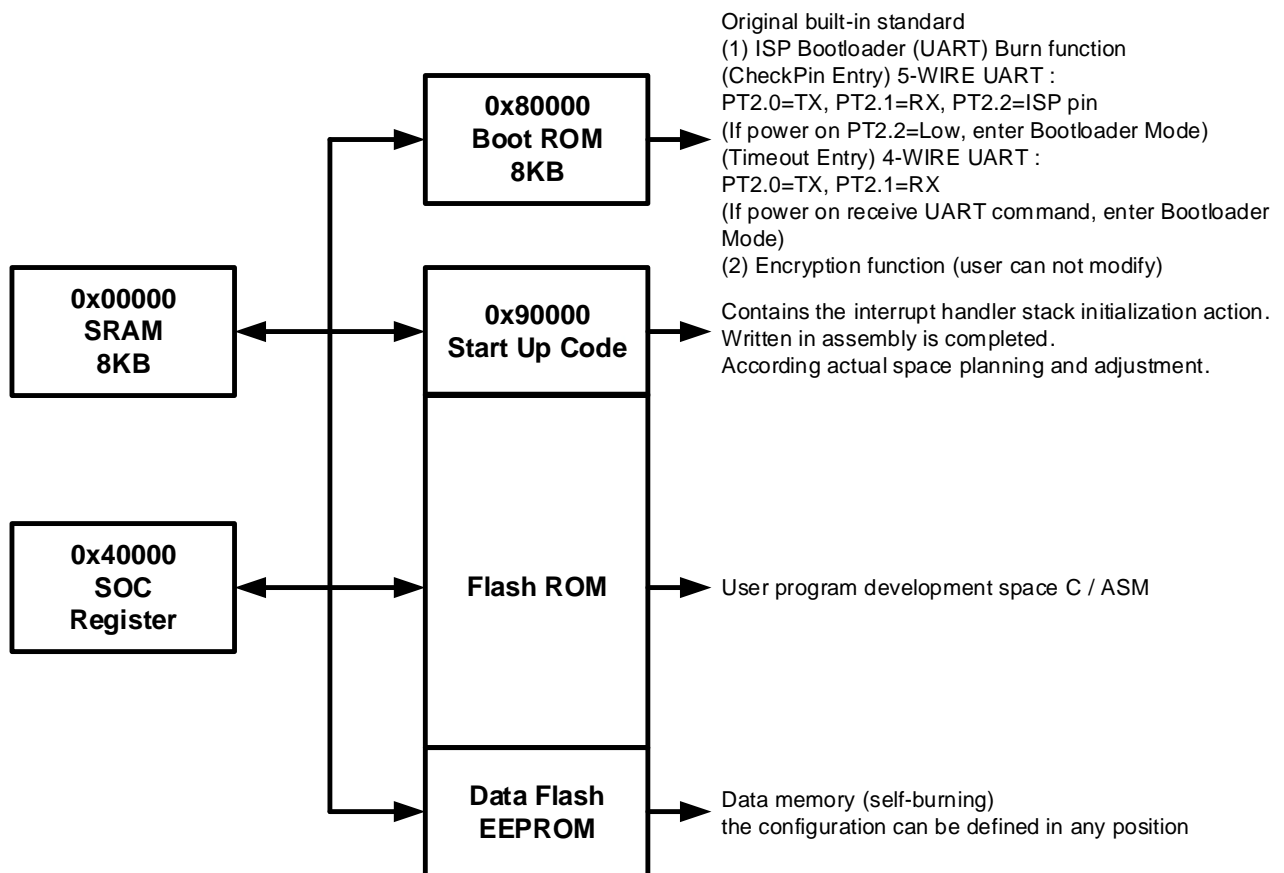


图 3-1 存储器地址分配图

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

3.2. 内存地址

微控制器详细的系统控制寄存器(SOC Register)地址分配如下表。

功能模块	描述	Base Address
INT	中断向量控制寄存器(Interrupt Control)	0x40000
SoC	系统控制寄存器(System)	0x40100
CLK	频率系统控制寄存器(Clock System)	0x40300
PMU	电源系统控制寄存器(Power Management)	0x40400
MC	内存控制寄存器(Memory Controller)	0x40600
PIO	通用型之输入输出端口控制寄存器(GPIO Port Control)	0x40800
TMR	计数器控制寄存器(Timer Register)	0x40C00
UART	UART 通信界面控制寄存器(UART Mode)	0x40E00
SPI	SPI 通信界面控制寄存器(SPI Mode)	0x40F00
I ² C	I ² C 通信界面控制寄存器(I ² C Mode)	0x41000
ADC	模拟数字转换器控制寄存器(Analog-to-Digital Module)	0x41100
RTC	硬件万年历 RTC 寄存器(Real Time Clock)	0x41A00
LCD	液晶显示驱动控制寄存器(LCD)	0x41B00

表 3-1 系统控制寄存器

部分重要寄存器都含有遮蔽位(MASK Bit)，如下图 3-3 说明。MASK 是用于使能对应控制位的写入，只有在与控制位对应的 MASK 位为<1>时，对应的控制位才能写入值，否则写入动作会无效，无法真正修改寄存器的值。如图 3-2 所示。

寄存器长度总共有 32 位，大部分有 16 位 MASK Bit。MASK Bit 分为两组 8 位，每 8 位 MASK 控制着相应 8 位控制寄存器位。根据寄存器的内容分布：BIT[31:24]控制着 BIT[23:16]，而 BIT[15:8]控制着 BIT[7:0]。只有在 MASK Bit 置<1>的情况下，对应的控制位才能写入有效值。

如：要对 BIT[5:0]写入 101010b，则寄存器操作方式写入值为：0011111100101010b。其中 00111111b 是 BIT[15:8]的 MASK BIT，可启用 BIT[5:0]对应控制位写入值有效，而 00101010b 就是对 BIT[5:0]写入的值。

INT Base Address + 0x10 (0x40010)									
Symbol	INTPT1 (PT1 Interrupt Control Register)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	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]
名称	MASK	PT17IF	PT16IF	PT15IF	PT14IF	PT13IF	PT12IF	PT11IF	PT10IF
RW	R0W-0	RW0-0							

表 3-2 寄存器的基本结构

3.3. 静态随机存取内存(SRAM)

HY16F3910 带有 8k Byte 静态随机存取内存。起始地址是从 0x0000 到 0x1FFF。MCU 可选择一个字节(One Byte)、两个字节(Half Word)、或四个字节(One Word)的存取。可以在一个 Clock Cycle 存取四个字节(One Word)数据。

3.4. 闪存(Flash ROM)

HY16F3910 带有 128k Byte 的嵌入式闪存。起始位置是从 0x90000 到 0xAFFFF。使用者可将程序代码储存在闪存。要编写闪存的程序代码，使用者需用 CPU 指令来读写快闪控制单元。使用者可以选择区块间的任何位置储存数据。

3.5. 总线接口单元

总线的结构中，寄存器的读写是由一个 32 位的高阶外围总线(Advanced Peripheral Bus, APB)所控制。可在一个 Clock Cycle 写入一个 32 位的数据。而为了防止数据写入期间去覆盖掉已写入的数据值，则可使用了 MASK 屏蔽的功能来完成。

如图 3-3 说明，原始寄存器中 BIT[7:0]数据为 10101010b，透过 MASK BIT 的控制使写入数据有效，当操作 BIT[15:0]，写入资料 0000111101010101b，则结果为 :000000010100101b。表示 MASK Bit 只可设置 1b，回读时皆为 0b，而 BIT[7:4]虽然要写入 0101b，但是 BIT[15:12]定义为 0000b，因此代表 BIT[7:4]的写入值是无效；而 BIT[3:0]要写入 0101b，且同时 MASK BIT[11:8]定义为 1111b，因此代表 BIT[3:0]的写入值可以有效。

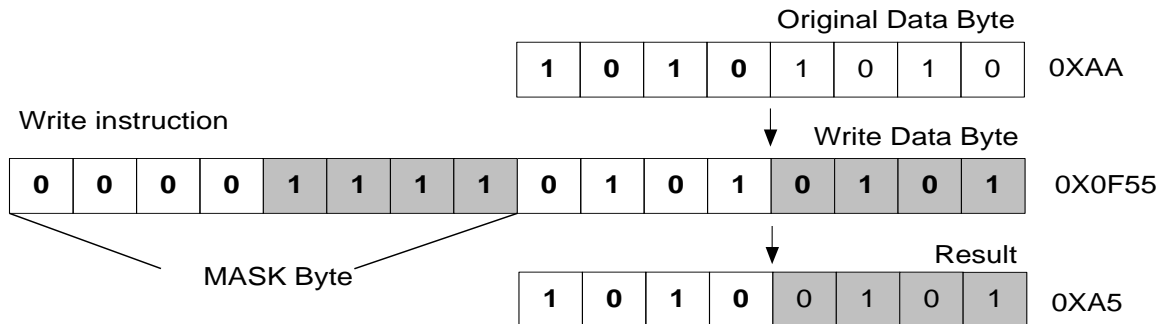


图 3-2 数据 MASK 结构

3.6. 开机内存(Boot ROM)

8k Byte 的开机内存。起始地址是从 0x80000 到 0x81FFF。该区块空间是给开机码、(Flash)快闪句柄、和安全保护码所使用。当芯片重置时，程序计数器会从 0x80000 开始。开机内存中的软件涵盖许多讯息，包含系统程序协议、和安全协议等。

3.7. 嵌入式除错模块(EDM)

嵌入式除错模块(Embedded Debug Module, EDM)为芯片在开发环境下可使用的除错接口。在芯片未设定安全性保护下，使用者可以透过 EDM 接口传送指令给 MCU，进而读取除错模式中的讯息。EDM 是芯片和计算机沟通的桥梁，透过 HY16F Mini Link(USB Control Board)连接 PC USB 以及芯片 EDM，仅使用两线协议接口。EDM 可以透过 APB 存取芯片的控制寄存器、一般用途的 GPR 寄存器、SRAM DLM、以及 Flash ROM ILM。

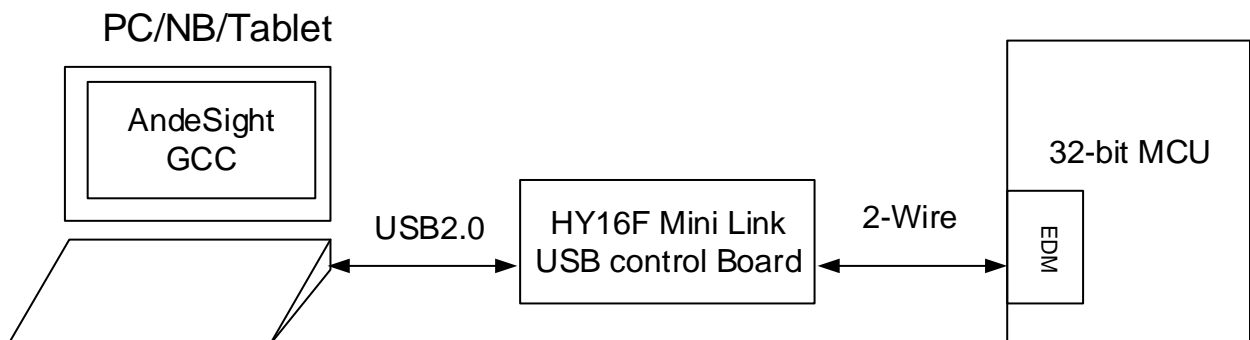


图 3-3 芯片开发连接图

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

4. 系统寄存器

4.1. 整体总说明

管理系统的工作模式及芯片的复位状态，如看门狗、外部复位、欠电压复位等。

4.2. 寄存器地址

SOC Status Register Address	31	24	23	16	15	8	7	0
SOC Status Base Address + 0x04(0x40104)	-	-	-	-	MASK0	-	REG0	-

-保留

4.3. 寄存器功能

操作该寄存器 0x40104[4]可以对系统工作模式设置为休眠模式(SLEEP)/待机模式(IDLE)。查询该寄存器 0x40104[3]可知芯片处于什么样的工作状态下。

4.3.1. SOC 寄存器

SOC Status Base Address + 0x04 (0x40104)									
Symbol	SOC Status Register								
Bit	[31:24]				[23:16]				
名称	ICE Configuration				SOC Configuration				
RW	R-0x0F				R-0x1A				
Bit	[15:8]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	V15OVD	V15PG	F _{CRst}	IDLE	F _{SLP/IDLE}	F _{WDog}	F _{RST}	F _{BOR1}
RW	R0W-0	R-0	R-0	R-0	R-0	RW0-0	R-0	R-0	RW0-1

位	名称	描述
Bit[07]	V15OVD	VDD15 过高电压 Flag (VDD15>12%) ,
		0 VDD15 电压正常
		1 VDD15 电压过高, 产生标志位;
Bit[06]	V15PG	VDD15 过低电压 Flag (VDD15<12%)
		0 VDD15 电压正常
		1 VDD15 电压过低, CPU Hold,等待电压正常后 CPU 才有 Clock 继续动作
Bit[05]	F _{CRST}	CPU Core 复位标志位
		0 正常
		1 ICP Core 曾经被触发
Bit[04]	IDLE	待机模式控制位
		0 休眠模式(Sleep Mode)
		1 待机模式(IDLE Mode)
Bit[03]	F _{SLP/IDLE}	休眠(Sleep)/待机(Idle Flag) (低电压复位或复位电路复位可清零该位)
		0 正常
		1 芯片进入休眠模式(Sleep Mode)或待机模式(Idle Mode)
Bit[02]	F _{WDT}	看门狗(WDT)标志位(低电压复位或外部复位可清零该位)

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

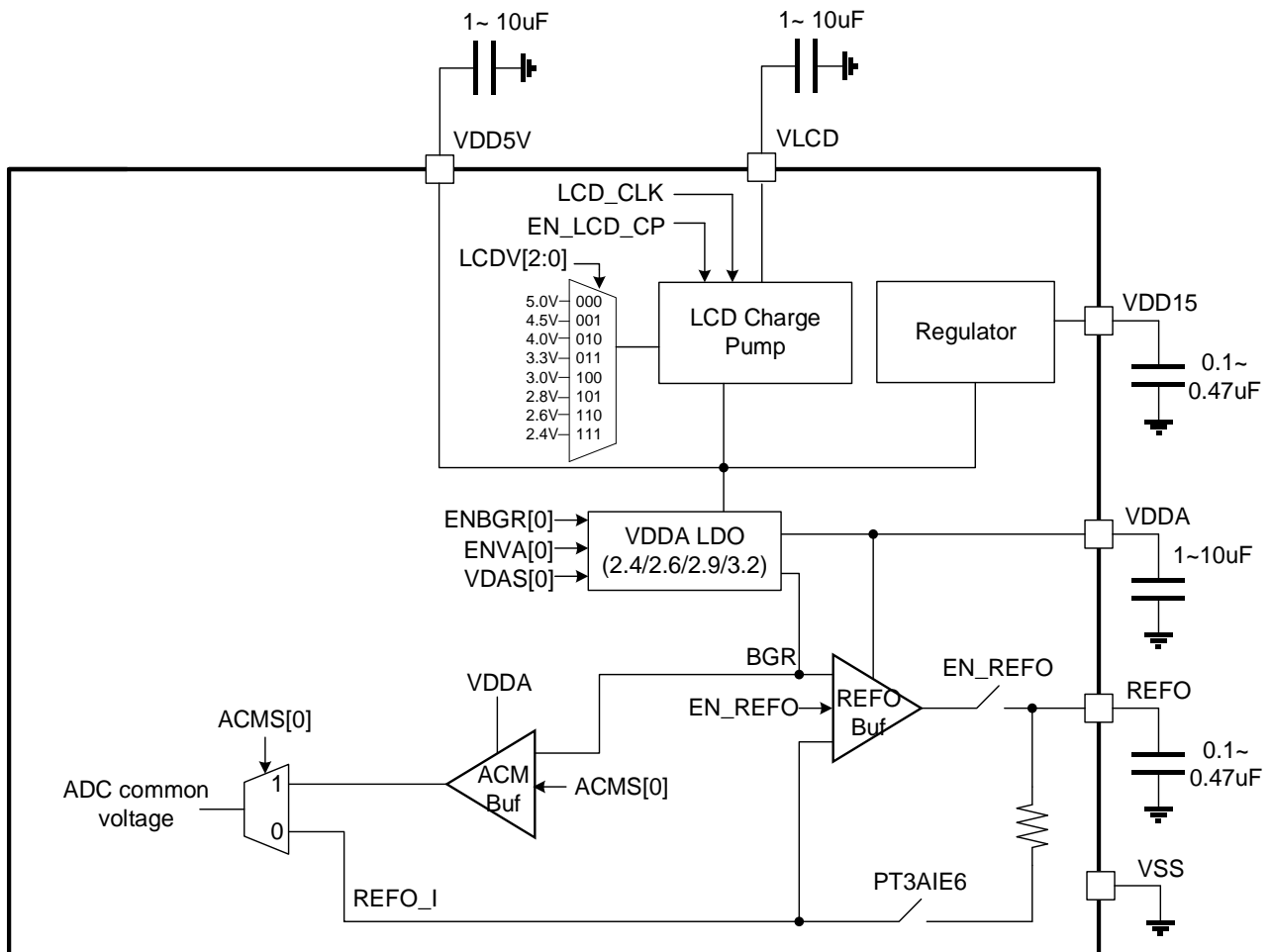
4 X 44~8 X 40 LCD Driver

位	名称	描述	
		0	正常
		1	看门狗产生复位或者中断
Bit[01]	F _{RST}	外部复位标志位(低电压复位(BOR)可清零该位)	
		0	正常
		1	Reset PIN 复位或者 ICP 软件复位已发生
Bit[00]	F _{BOR1}	BOR1 复位标志位	
		0	未发生过复位
		1	BOR1 复位已发生过 (使用者自行清除标志位)

5. 电源管理

5.1. 整体总说明

电源管理模块包括一个 Band Gap 基准(BGR)、一个 VDDA LDO、一个 VDD15 LDO、以及基准输出缓冲器。芯片 VDD5V 只需要一个电压源来运转，这运转电压源介于 2.0V 到 5.5V 之间。电源系统可分成三个部份：输入/输出电路、模拟电路、和数字电路。输入/输出电路的电源是由 VDD5V 所驱动。模拟电路电源是由内部的 VDDA LDO 所驱动。最后，CPU Core 的电源是由 VDD15 LDO 所驱动。当 MCU 在待机模式，芯片以最低功耗来维持寄存器和 SRAM 的数据存储操作。在待机模式中，BOR1、和 VDD15 LDO 是开启的。若需要自动唤醒模式，低速振荡器需要被开启。



5.1.1. 芯片工作电压 VDD5V 及 VDD15:

芯片工作电压是通过引脚 VDD5V 输入，电压范围是 2.0V~5.5V，且在引脚需要接一个 1uF~10uF 的对地电容，可使这 VDD5V 电压更稳定。VDD15 LDO 通过引脚 VDD15 输出稳压电压 1.5V，且引脚需要接入一个 0.47uF 的旁路电容器。

5.1.2. VDDA 电压:

芯片带有一个模拟电路使用的稳压电路 LDO：VDDA，使用 ADC 前必须开启 VDDA 电压。它可设置不同的操作模式和不同的输出电压。它有二个不同的操作模式，第一个模式是高阻态(High Z)，可从外部输入电压给

VDDA，但是外部输入的电压应该不可超过 VDD5V；第二个模式是可调节稳压模式 LDO，此模式 VDDA 可依设定输出不同的电压，如 2.4V、2.6V、2.9V、和 3.2V。要有更好的效能，VDD5V 应该要比 VDDA 高 0.2V 以上，且可驱动达到 10mA。它也需要在引脚 VDDA 接一个 1uF~10uF 的旁路电容器。注意，当 VDDA 接一个 1uF 对地电容时，VDDA 的稳定时间至少需要大于 0.5ms，当 VDDA 接一个 10uF 对地电容时，VDDA 的稳定时间至少需要大于 1ms。

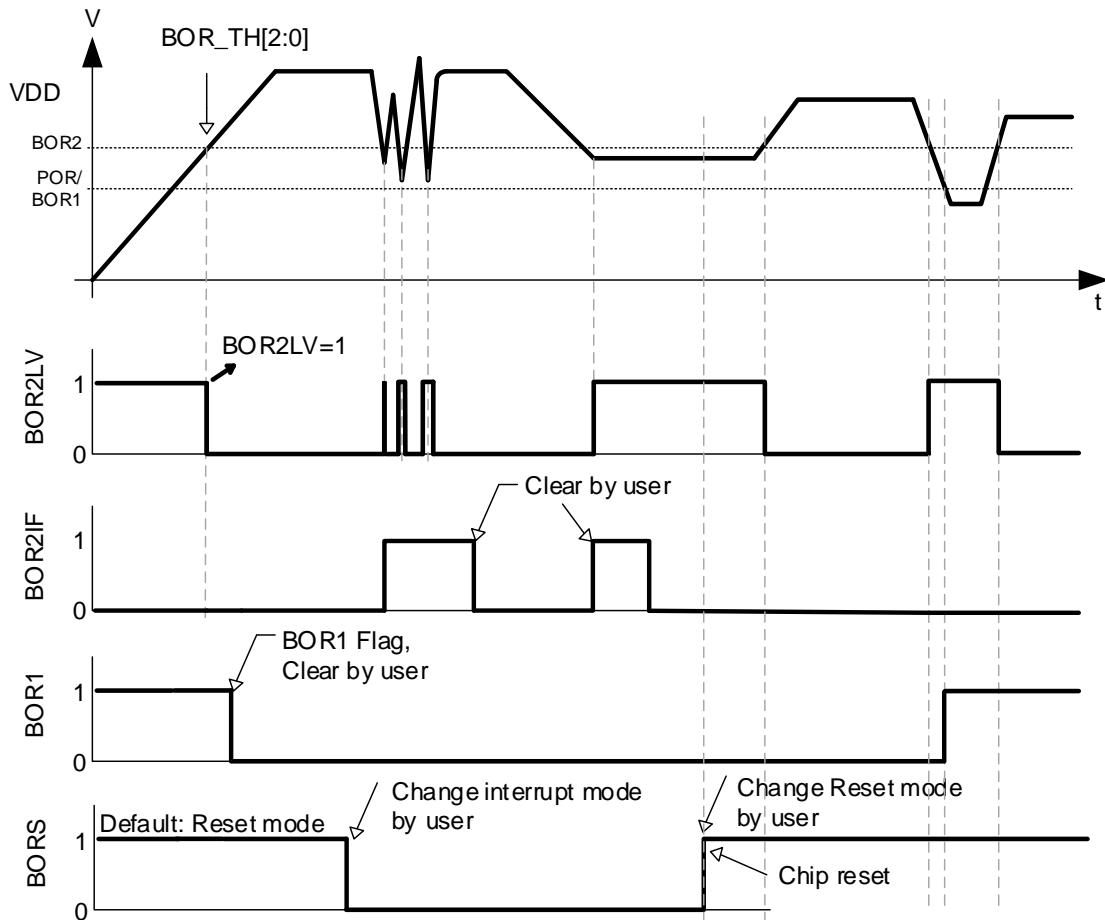
5.1.3. 芯片上电电压检测电路(BOR1、BOR2):

BOR1、BOR2 电路是用来监控电源系统和 MCU 的稳定度。当 VDD5V 的电压低于 BOR1 侦测电压就会被触发，系统因而被重设，芯片产生复位，直到芯片工作电压高于 BOR2 预设电压 1.7V 以上才会恢复正常工作。

BOR1 为省电 BOR 电路设计，BOR2 为精准 BOR 电路设计，芯片首次上电时，会同时触发 BOR1 以及 BOR2 线路，在高精度 BOR2 电压设计，可确保工作电压在 2.0V 时芯片就可以正常运作，满足低工作电压的要求。高精度 BOR2 电压设计下，不用外接 Reset IC 就可以满足各种上电波形条件下不当机的要求。当发生 BOR1 复位时，F_{BOR1}[0]标志位会被置<1>以记录发生的事件。BOR1 线路无法透过程序或其它设置方式使其关闭。

BOR2 预设会透过 BOR1 电路触发启动，并初始化 BOR2 电压为 1.7V 档位。BOR2 控制开关 ENBOR2[0]使用者可以透过软件操作来关闭达省电功能。BOR2 的触发行为可以设定为芯片 Reset 或是中断模式，中断模式可以反映 BOR2IF 标志位。在睡眠模式中可以关闭 BOR2 功能，满足较低的功耗设计需求。下图为 BOR1、BOR2 状态示意图。

BOR2 支持睡眠模式下自动启动功能，BOR2SLP 控制位预设状态为启动状态。当芯片从睡眠模式唤醒后，若 BOR2SLP 为启动状态，则 BOR2 会自动开启并切换成 Reset Mode 状态。若 BOR2SLP 为关闭状态，则不会改变 BOR2 原先状态。BOR2SLP 控制位启动可有效强化电源抗干扰能力。



5.1.4. 参考电压和共模电压(REFO):

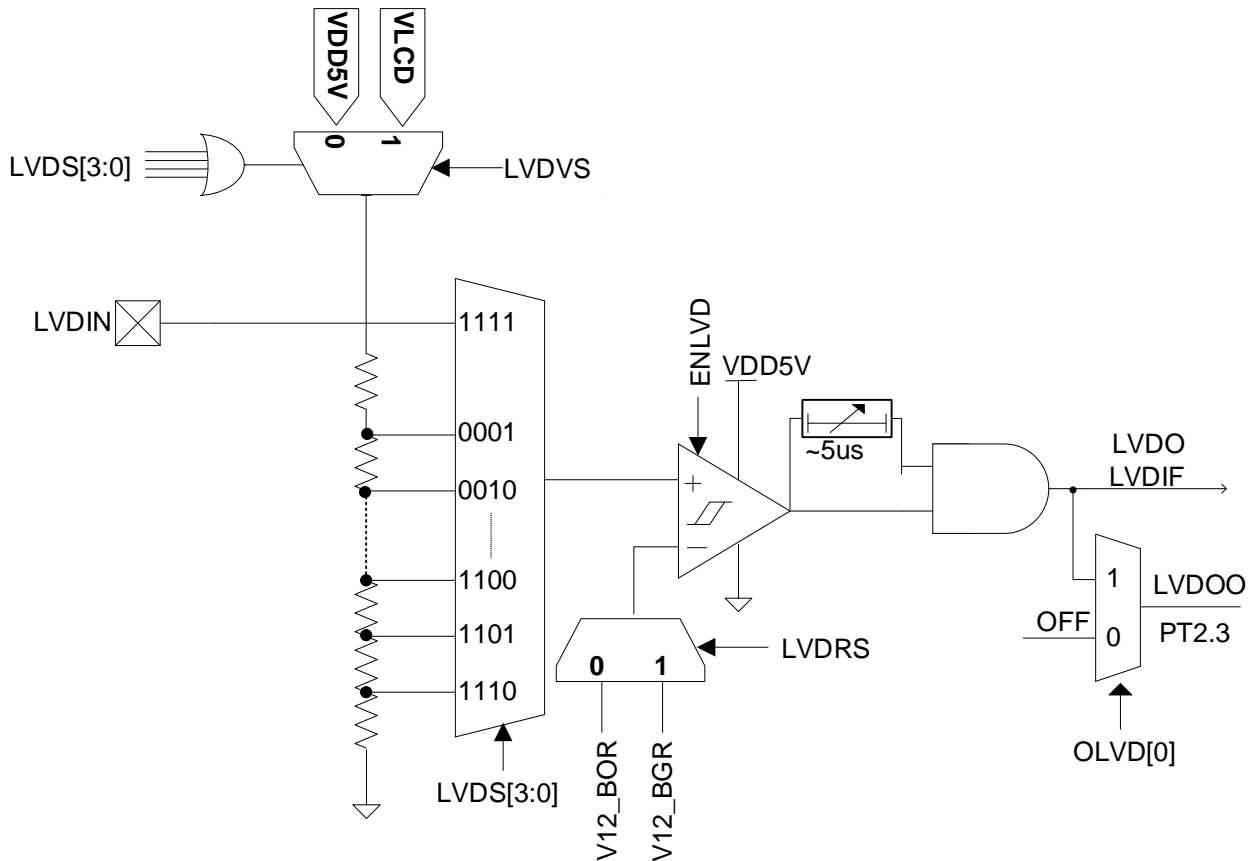
当 VDDA 高于 2.4V，模拟电路就可运转。但是，模拟电路要求提供 Band Gap 参考电压。因此，在启动模拟电路之前启动 Band Gap 参考电压，将寄存器 PMU[4](ENBGR)设定为 1 便可启动 Band Gap 参考电压。只有开启 Band Gap 参考电压后，共模电压(REFO)才有效输出 1.2V。要启动 ADC，需要为 ADC 提供一个共模电压(REFO)。这个共模电压可选择外部或内部的电源。若要选择内部电源，就要将 ACMS 设定为 1。若要选择外部电源，就要将 ACMS 设定为 0，同时需要输出一个共模电压(REFO)。使用者因应用会需要一个基准电压来驱动外部电路。就可以将 ENRFO 设定为 1，将共模电压输出至引脚，且 REFO 是具有缓冲器的 Band Gap 基准电压。REFO 引脚输出电压大约是 1.2V 且具有 +/-1mA 的推拉驱动能力。它可以驱动一个 22~1000nF 的大型电容器承载。如果使用了外部的 REFO 电压输出，此时 ADC 使用的共模电压则可用外部电源，将 ACMS 设定为 0，可节省电力。注意，当 REFO 接入一个 0.1uF 对地电容时，REFO 的稳定时间至少需要大于 0.1ms。

5.1.5. LVD 低电压检测:

LVD 低电压检测针对工作电压 VDD5V 与 VLCD 电压或外部输入电压 LVDIN，提供可设置的低电压侦测功能，当输入电压达到低电压侦测点时，则寄存器的 LVDO 位自动置 1；LVD 参考电压源为可选择 BOR 或 BGR(即 Band Gap 参考电压)。当输入电压为外部电压 LVDIN 时，低电压侦测点设置 LVDS 需设置为 0000b，此时低电压侦测点为 1.2V，即 BOR 或 BGR 实际电压大小；当输入电压为 VDD5V 或 VLCD 时，低电压侦测点 LVDS 设置范围为 0001b~1111b(2.0V~4.0V)。

设置流程:

- 选择输入电压，若为外部输入电压 LVDIN 则不需对寄存器 LVDS 设置，或透过 LVDVS 设置选择 VDD5V 或 VLCD
- 选择参考电压，透过 LVDRS 设置选择 V12_BOR 或 V12_BGR。当选择 V12_BGR 时，需要先开启 Band Gap 电压源才能使用(即 0x40400[4]=1b)，选择 V12_BOR 不需做任何额外电压源开启动作
- 设置低电压侦测点，若为外部输入电压 LVDIN 则 LVDS 设置为 0000b，若为 VDD5V 或 VLCD 则 LVDS 设置低电压侦测点 0001b~1111b(2.0V~4.0V)
- 使能低电压比较器 ENLVD，开始低电压侦测



HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

5.1.6. 各部份模块使用的电压源

Block 名称	电压源	Block 名称	电压源
32-bit CPU Core	VDD15	Band Gap/Reference	VDDA
8k Byte SRAM	VDD15	Hardware EUART	VDD15/VDD5V
128k Byte Flash ROM	VDD5V/ VDD15	32-bit Hardware SPI	VDD15/VDD5V
Clock System	VDD15	Hardware I2C	VDD15/VDD5V
Watch Dog Timer	VDD15	Timer A/B/C PWM	VDD15
Hardware RTC	VDD15	GPIO Port	VDD5V
BOR1/BOR2	VDD5V/ VDD15	24-bit SD ADC	VDDA

表 5-1 芯片电源分布

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



5.2. 寄存器地址

Power Management Register Address	31	24	23	16	15	8	7	0
PMU Base Address + 0x00 (0x40400)	MASK1		REG1		MASK0		REG0	
PMU Base Address + 0x08 (0x40408)	Rsv			REG3	MASK2		REG2	
PMU Base Address + 0x0C (0x4040C)	MASK5		REG5		MASK4		REG4	

5.3. 寄存器功能

5.3.1. 电源管理寄存器 0

Power Management Base Address + 0x00 (0x40400)									
Symbol	PMU0 (PMU Control Register 0)								
Bit	[31:24]	[23:20]			[19:18]		[17]		[16]
名称	MASK	-			VDAS		-		ENVA
RW	R0W-0	RW-0							
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	ENBGR	-	Rsv	ENRFO	V15LP
RW	R0W-0	RW-0	-		RW-0				

位	名称	描述
Bit[23:20]	-	保留请勿设定
Bit[19:18]	VDAS	VDDA 输出电压选择
		00 VDDA =2.4V
		01 VDDA =2.6V
		10 VDDA =2.9V
		11 VDDA =3.2V@VDD5V>=3.5V
Bit[16]	ENVA	VDDA LDO 电压源设置, 控制 VDDA 输出电压来源
		0 高阻抗(High Z), 可由 VDDA 引脚灌入电压
		1 VDDA LDO, VDDA 可调输出, 由 VDAS 决定
Bit[07]	-	保留请勿设定
Bit[04]	ENBGR	Band Gap 电压开启控制
		0 关闭
		1 开启
Bit[01]	ENRFO	参考电压(REFO)开启控制
		0 关闭
		1 开启
Bit[00]	V15LP	VDD15 LDO 低功耗控制
		0 正常(从休眠唤醒后该位需被置 0)
		1 低功耗(仅支持睡眠模式下启动低功耗设置) 此模式下, V15PG 与 V15OVD 标志位无效

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

5.3.2. 电源管理寄存器 1

Power Management Base Address + 0x08 (0x40408)						
Symbol	PMU1 (PMU Control Register 1)					
Bit	[31:19]			[18:17]		[16]
名称	Rsv			-		LVDO
RW	R-0			RW-0		R-X
Bit	[15:08]	[7:4]	[3]	[2]	[1]	[0]
名称	MASK	LVDS	LVDITT	LVD12	LVDVS	ENLVD
RW	ROW-0	RW-0	RW-0	RW-0		

位	名称	描述
Bit[18:17]	-	保留请勿设定
Bit[16]	LVDO	LVD Output
		0 当负端电压 > 正端电压时, LVDO=0
		1 当正端电压 > 负端电压时, LVDO=1
Bit[7:4]	LVDS	LVD 正端电压值
		0000 OFF
		0001 2.0V
		0010 2.1V
		0011 2.2V
		0100 2.3V
		0101 2.4V
		0110 2.5V
		0111 2.6V
		1000 2.7V
		1001 2.8V
		1010 2.9V
		1011 3.0V
		1100 3.3V
		1101 3.6V
1110 4.0V		
1111 1.2V, 外部输入电压 LVDIN.		
Bit[03]	LVDITT	LVDIF 中断源触发设定位
		0 当 LVDO 信号由 1 变为 0 时, 触发中断信号
		1 当 LVDO 信号由 0 变为 1 时, 触发中断信号
Bit[02]	LVD12	LVD 负端电压源
		0 V12_BOR
		1 V12_BGR(form bandgap output, precised)
Bit[01]	LVDVS	LVD 正端电压源
		0 VDD5V
		1 VLCD
Bit[00]	ENLVD	LVD Enable
		0 Disable
		1 Enable

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



5.3.3. 电源管理寄存器 2

Power Management Base Address + 0x0C (0x4040C)							
Symbol	PMU2 (PMU Control Register 2)						
Bit	[31:24]			[23:19]			[18:16]
名称	MASK			-			-
RW	R0W-0			WR-0			WR-100
Bit	[15:08]	[7]	[6:4]	[3]	[2]	[1]	[0]
名称	MASK	-	BOR2TH	BOR2SLP	BOR2LV	BOR2S	BOR2EN
RW	R0W-0	-	RW-0	RW-1	R-X	RW-1	

位	名称	描述
Bit[18:16]	-	保留请勿设定
Bit[6:4]	BOR2TH	BOR2 电压点设置位; BOR1 发生时,清除为<000>预设状态
		000 1.7V (预设)
		001 2.0V
		010 2.2V
		011 2.5V
		100 2.7V
		101 3.0V
		110 3.6V
		111 4.0V
Bit[03]	BOR2SLP	BOR2 自动启动功能 (只支持 Sleep mode 唤醒后自动切换功能).
		0 从睡眠模式唤醒时, BOR2 状态不变更. 1 从睡眠模式唤醒时,会自动启动 BOR2,并且切换为 Reset mode. (预设)
Bit[02]	BOR2LV	BOR2 信号状态实时反映
		0 VDD5V 电压 低于 BOR2 比较点 1 VDD5V 电压 高于 BOR2 比较点
Bit[01]	BOR2S	BOR2 工作模式设定
		0 中断模式; 1 Reset mode (预设). BOR1 发生时重置为 1.
Bit[00]	BOR2EN	BOR2 Enable
		0 Disable 1 Enable (预设). BOR1 发生时重置为 1.

6. 频率系统

6.1. 整体总说明

频率控制系统为整个芯片提供频率,包括系统频率(CPU 频率、APB 频率)及所有外围外设工作频率(定时器、通信接口、RTC、模拟电路等外围设备)。各个功能模块的频率都带有时钟开关控制器,频率源选择及除频器。在休眠模式下,控制器始终关闭外部振荡器、内部振荡器及系统频率使系统功耗降到最低。工作频率源具有外部振荡器、内部 HAO 与 LPO 振荡器,透过除频器可弹性分配与管理 CPU 及外围设备的频率源调整芯片功耗,达到降低功耗目的。

6.1.1. 外部振荡器

外部振荡器分别为高速振荡器(HSXT)和低速振荡器(LSXT)。芯片带有高速和低速外部振荡器两组独立输入引脚,所以用户可以同时接上两组外部振荡器。外部振荡器需要并联一个电阻,否则即使焊接了振荡器,也无法正常启动振荡,同时必须要焊接两个 0~20pF 的对地电容,对地电容值大小需要视电路板布局产生的寄生电容效应评估。其振荡器接脚间的并联电阻(R1)、与振荡器各别引脚电容 C2/C1 参数随外接不同频率、不同品牌的振荡器及电路板布局方式而有所差异。下表列出 R1/C1/C2 参数与频率源配置建议值,供使用参考。

型式	符号	外部振荡器参数				指令执行状态	
		频率	R1/ Ω	C1	C2	Sleep Mode	Idle Mode
低速振荡	LSXT	32768Hz	10M	10pF	10pF	停止	可设置
高速振荡	HSXT	2~16MHz	1M	10pF	10pF	停止	可设置

表 6-1 外部振荡器配置建议

注意：外部振荡器引脚电容 C1/C2 大小,可依照实际 PCB 电路板布局与不同的振荡器视情况选择调整其电容值大小,建议范围为 0~20pF。

使用外部振荡器注意事项：

- 执行休眠 Sleep 指令后,外部振荡器皆会停止振荡。
- 外接振荡器时,需要注意引脚的输入/输出配置,使用时引脚设置不得配置内部上拉电阻,以免操作时发生异常。且外部电阻 R1 绝对不能缺省。
- 要使用外部振荡器(HSXT),必须选择 MCU Clock/2 以上的除频设计,可降低振荡器频率源受到干扰,强化抗干扰能力。

6.1.2. 内部 RC 振荡器 HAO 与 LPO

HAO 为芯片内部高速 RC 振荡器输出典型频率值为 4.147MHz/31.795MHz,且具有快速启动振荡、抗干扰能力强及低功耗的特性。HAO 输出频率值具有可校正性,用户可通过软件的方式对 HAO 的输出频率进行校正。

使用内部 RC 振荡器需要注意事项：

- 使用内部 RC 振荡器 HAO 31.795MHz 时,芯片工作电压须保持在高电压供电,且必须选择 MCU Clock/2 以上的除频设计,芯片才可以正常操作。
- 设置寄存器 HAOTR 0x40304[6:0]的值,可调整 HAO 输出频率值。

典型的 HAO 与 LPO 输出频率值如下表所示。

符号	频率	频率源配置			指令执行状态	
		ENHAO[1]	HAO[1:0]	CKHS[1]	Sleep	Idle
HAO	4.147MHz	1	01B	0	停止	可动作
	31.795MHz	1	11B	0	停止	可动作
LPO	32kHz	芯片上电后起振		CKLS=0	停止	振荡

表 6-2 内部 RC 振荡器配置

HAO 内部频率校正使用说明:

芯片 HAO 内部频率出厂时候会有 +/-10%左右的误差, 如果使用者想要有更精确的 HAO 工作频率, 可以使用 HAO 频率校正功能。HAO 频率校正部份, 可以使用宏康 C 函式库 DrvCLOCK_CalibrateHAO 这个函数, 可以透过这个函数的设定, 控制 HAO 振荡频率误差范围接近在 +/-2%以内, 详细正确 HAO 频率规格, 可以参考规格书 HY16F3910 说明, 函数使用说明可以参考如下:

-函数

```
void DrvCLOCK_CalibrateHAO(short int uMHZ)
```

-函数功能

按照芯片出厂时 HAO 的校正值来校正内部 RC 振荡器(HAO);使用时注意要与选定的 HAO 频率对应;设置寄存器 0x40304[6:0]的值

-输入参数

uMHz [in]待校正值的 HAO 频率模式选择

0 : Rsv. ; 1 : 校正 4.147MHz ; 2 : Rsv. ; 3 : 校正 31.795MHz ;

6.1.3. 工作频率源配置

内部/外部振荡器都可以为 CPU 提供频率源, 频率源经过除频器后供给 CPU。芯片可透过频率选择器 MCUCKS[1]选择 CPU 的频率源为 HS_CK 或 LS_CK, 通过除频器 ENMCD[1:0]对频率源进行除频。因而 CPU 的工作频率模式有多种可选择, 进而决定芯片的指令周期。周边外设工作频率源同样由外部或内部振荡器透过 HS_CK 或 LS_CK 经过除频器供给。或者由振荡器直接供给频率源, 如 WDT。由于周边外设工作性质不同而有不同的规划, 使用者可参考下图说明。

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

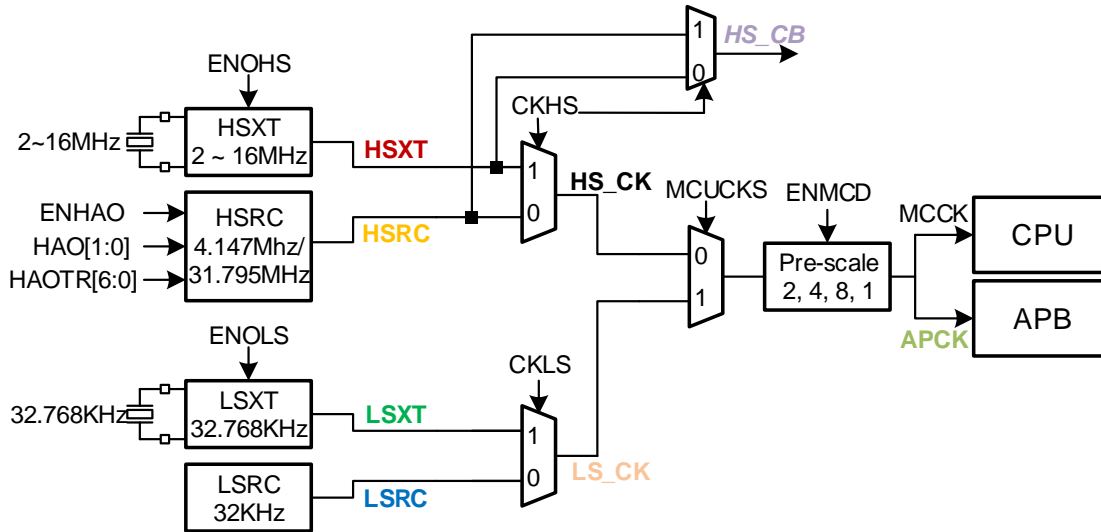


图 6-1 CPU 工作频率源配置图

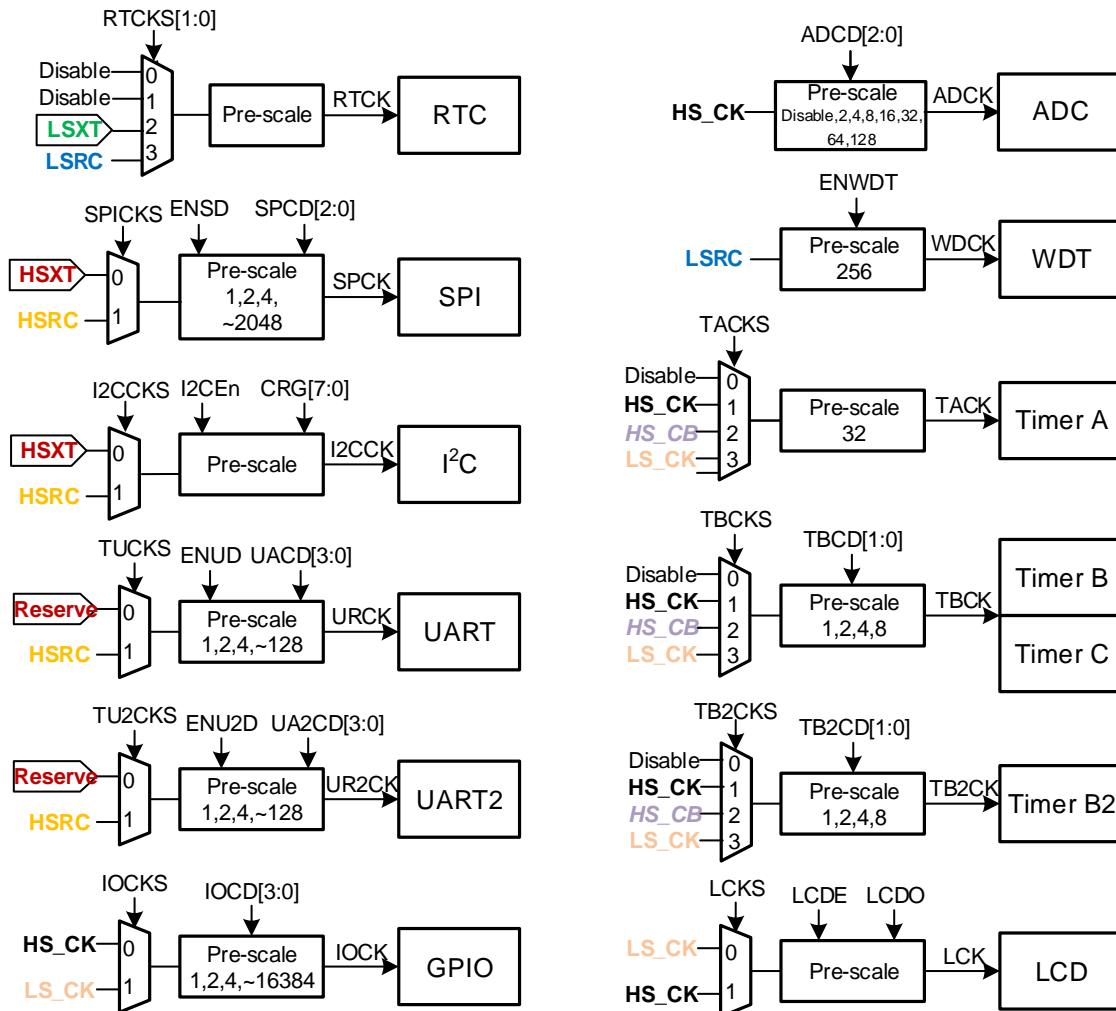


图 6-2 周边外设工作频率配置图

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



6.2. 寄存器地址

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	
CLK Base Address + 0x10 (0x40310)	MASK5		REG5		MASK6		REG6	
CLK Base Address + 0x14 (0x40314)	MASK7		REG7		MASK8		REG8	
CLK Base Address + 0x18 (0x40318)	-		-		-		-	
CLK Base Address + 0x18 (0x4031C)	-		-		-		-	
CLK Base Address + 0x20 (0x40320)	MASK9		REG9		MASK10		REG10	
CLK Base Address + 0x24 (0x40324)	REG11				MASK12		REG12	
CLK Base Address + 0x28 (0x40328)	REG13				-		REG14	

-保留

6.3. 寄存器功能

6.3.1. 频率系统寄存器 0

Clock Base Address + 0x00 (0x40300)								
Symbol	CLK0 (Clock Control Register 0)							
Bit	[31:16]					[18:16]		
名称	Rsv					-		
RW	R-0					RW-0		
Bit	[15:8]	[7]	[6]	[5]	[4:3]	[2]	[1]	[0]
名称	MASK	OHS_HS	CKLS	CKHS	HAO	ENOLS	ENOHS	ENHAO
RW	R0W-0	RW-0					RW-1	

位	名称	描述
Bit[18:16]	-	保留请勿设定
Bit[07]	OHS_HS	外部 高速 振荡器模式选择
		0 HSXT<4MHz 1 HSXT>4MHz
Bit[06]	CKLS	芯片 低速 频率源选择
		0 内部低速振荡器(OSC_LSRC) 1 外部低速振荡器(OSC_LSXT)
Bit[05]	CKHS	芯片 高速 频率源选择
		0 内部高速振荡器(OSC_HSRC) 1 外部高速振荡器(OSC_HSXT)
Bit[4:3]	HAO	内部高速振荡器(HAO)频率模式设置
		[00] Rsv.
		[01] 4.147MHz(预设)
		[10] Rsv. [11] 31.795MHz
Bit[02]	ENOLS	外部 低速 振荡器开启控制
		0 关闭

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述	
		1	开启
Bit[01]	ENOHS	外部高速振荡器开启控制	
		0	关闭
		1	开启
Bit[00]	ENHAO	内部高速振荡器开启控制	
		0	关闭
		1	开启

注意事项：

- HS_CK、LS_CK 频率源切换防呆控制：利用 CKHS 或 CKLS 切换 HS_CK 或 LS_CK 的频率源时，会判读相对应的振荡器是否开启，如未开启将不执行切换动作。
- 振荡器关闭防呆控制：如欲关闭某振荡器时，需先将 HS_CK 或 LS_CK 切换至另一组已开启之振荡器，避免振荡器关闭后使系统无频率源而当机。
- 高速振荡器关闭防呆控制：两组高速振荡可以同时关闭而不受前项防呆机制限制，但是需先将 CPU Core 频率源切换至低速频率源，否则无法同时关闭两组高速振荡器。

6.3.2. 频率系统寄存器 1

Clock Base Address + 0x04 (0x40304)			
Symbol	CLKCR1 (Clock Control Register 1)		
Bit	[31:16]		
名称	Reserved		
RW	R-0		
Bit	[15:8]	[7]	[6:0]
名称	Mask	-	HAOTR
RW	R0W-0	RW-0	RW-0x40

位	名称	描述
Bit[6:0]	HAOTR	内部高速振荡器校正控制寄存器 1*LSB.Step = 0.25%. Trim Range~ +/-16%. 000_0000b 是最慢速度 100_0000b 是预设速度 111_1111b 为最快速度

说明：HAO 频率校正部份，可以使用纘康 C 函式库 DrvCLOCK_CalibrateHAO 这个函数，可以透过这个函数的设定，把各频段 HAO 的 Trim 值填入在 HAOTR 寄存器，可控制 HAO 振荡频率误差范围接近在 +/-2% 以内

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



6.3.3. 频率系统寄存器 2

Clock Base Address + 0x08 (0x40308)						
Symbol	CLKCR2 (Clock Control Register 2)					
Bit	[31:24]	[23:22]	[21]	[20:19]	[18:16]	
名称	MASK	RTCKS	TUCKS	ENMCD	UACD	
RW	ROW-0	RW-0	RW-0	RW-0	RW-0	
Bit	[15:08]	[7:6]	[5:4]	[3:2]	[1]	[0]
名称	MASK	TBCKS	TBCD	TACKS	ENUD	MCUCKS
RW	ROW-0	RW-0				

位	名称	描述	
Bit[23:22]	RTCKS	RTC 频率源选择	
		00 关闭	
		01 关闭	
		10 LSXT(LSXT 需使能, 否则视为 Disable)	
		11 LSRC	
Bit[21]	TUCKS	EUART 频率源选择	
		0 Reserve	
		1 HSRC: 内部高速振荡器	
Bit[20:19]	ENMCD	MCU 输入频率除频设置	
		ENMCD[1:0]	Pre-scale
		00	MCU Clock/2 (预设)
		01	MCU Clock/4
		10	MCU Clock/8
11	MCU Clock/1		
Bit[18:16]	UACD	EUART 频率源 除频设置	
		UACD[2:0]	Pre-scale
		000	EUART 频率源/ 1
		001	EUART 频率源/ 2
		010	EUART 频率源/ 4
		011	EUART 频率源/ 8
		100	EUART 频率源/ 16
		101	EUART 频率源/ 32
		110	EUART 频率源/ 64
111	EUART 频率源/ 128		
Bit[7:6]	TBCKS	Timer B,C 频率源选择	
		TBCKS[1:0]	频率源
		00	关闭
		01	HS_CK
		10	HS_CB
11	LS_CK		
Bit[5:4]	TBCD	Timer B,C 频率除频设置	
		00 TBCK/1	

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述	
		01	TBCK/2
		10	TBCK/4
		11	TBCK/8
Bit[3:2]	TACKS	Timer A 频率源选择, 除频器固定为 1/32	
		0	关闭
		1	HS_CK
		2	HS_CB (如果 CPU 以 HAO 为频率源, TMA 则以 HSXT 为频率, 反之亦然)
		3	LS_CK
Bit[01]	ENUD	EUART 频率源开启控制	
		0	关闭
		1	开启
Bit[00]	MCUCKS	MCU 输入频率源选择	
		0	HS_CK
		1	LS_CK

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



6.3.4. 频率系统寄存器 3

Clock Base Address + 0x0C (0x4030C)					
Symbol	CLKCR3 (Clock Control Register 3)				
Bit	[31:24]	[23:21]	[20]	[19]	[18:16]
名称	MASK	-	-	-	-
RW	R0W-0	-	RW-0		
Bit	[15:08]	[7]	[6:4]	[3]	[2:0]
名称	MASK	-	ADCD	ENSD	SPCD
RW	R0W-0	R-0	RW-0		

位	名称	描述																		
Bit[18:16]	-	保留请勿设定																		
Bit[6:4]	ADCD	ADC 频率除频器设置																		
		<table border="1"> <thead> <tr> <th>ADCD[2:0]</th> <th>Pre-scale</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>Disable</td> </tr> <tr> <td>001</td> <td>~HS_CK / 2</td> </tr> <tr> <td>010</td> <td>~HS_CK / 4</td> </tr> <tr> <td>011</td> <td>~HS_CK / 8</td> </tr> <tr> <td>100</td> <td>~HS_CK / 16</td> </tr> <tr> <td>101</td> <td>~HS_CK / 32</td> </tr> <tr> <td>110</td> <td>~HS_CK / 64</td> </tr> <tr> <td>111</td> <td>~HS_CK / 128</td> </tr> </tbody> </table>	ADCD[2:0]	Pre-scale	000	Disable	001	~HS_CK / 2	010	~HS_CK / 4	011	~HS_CK / 8	100	~HS_CK / 16	101	~HS_CK / 32	110	~HS_CK / 64	111	~HS_CK / 128
		ADCD[2:0]	Pre-scale																	
		000	Disable																	
		001	~HS_CK / 2																	
		010	~HS_CK / 4																	
		011	~HS_CK / 8																	
		100	~HS_CK / 16																	
		101	~HS_CK / 32																	
110	~HS_CK / 64																			
111	~HS_CK / 128																			
Bit[03]	ENSD	SPI 频率开关																		
		0 关闭																		
		1 开启																		
Bit[2:0]	SPCD	SPI 频率除频器设置																		
		<table border="1"> <thead> <tr> <th>SPCD[2:0]</th> <th>Pre-scale</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>SPI 频率源/ 1</td> </tr> <tr> <td>001</td> <td>SPI 频率源/ 2</td> </tr> <tr> <td>010</td> <td>SPI 频率源/ 4</td> </tr> <tr> <td>011</td> <td>SPI 频率源/ 8</td> </tr> <tr> <td>100</td> <td>SPI 频率源/ 32</td> </tr> <tr> <td>101</td> <td>SPI 频率源/ 128</td> </tr> <tr> <td>110</td> <td>SPI 频率源/ 512</td> </tr> <tr> <td>111</td> <td>SPI 频率源/ 2048</td> </tr> </tbody> </table>	SPCD[2:0]	Pre-scale	000	SPI 频率源/ 1	001	SPI 频率源/ 2	010	SPI 频率源/ 4	011	SPI 频率源/ 8	100	SPI 频率源/ 32	101	SPI 频率源/ 128	110	SPI 频率源/ 512	111	SPI 频率源/ 2048
		SPCD[2:0]	Pre-scale																	
		000	SPI 频率源/ 1																	
		001	SPI 频率源/ 2																	
		010	SPI 频率源/ 4																	
		011	SPI 频率源/ 8																	
		100	SPI 频率源/ 32																	
		101	SPI 频率源/ 128																	
110	SPI 频率源/ 512																			
111	SPI 频率源/ 2048																			

注: 在设置 ADCD 的 ADC 频率除频器时, 最佳的 ADC 工作频率为 1MHz

建议设置 HS_CK=4.147MHz, 让 ADCD 工作频率为 HS_CK / 4 ~ 1MHz.

当选择 ADC 工作频率为 1MHz 时, 则 CPU 频率 MCCK 至少要设定为 4.147MHz 以上频率才可以获得最佳效能。

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



6.3.5. 频率系统寄存器 4

Clock Base Address + 0x10 (0x40310)						
Symbol	CLKCR4 (Clock Control Register 4)					
Bit	[31:24]	[23:22]	[21]	[20]	[19]	[18:16]
名称	MASK	LCDCPD	TU2CKS	ENU2D	-	UA2CD
RW	R0W-0	RW-0			-	RW-0
Bit	[15:08]	[7]	[6:4]	[3:1]	[0]	
名称	MASK	-	LCDO	LCDE	LCKS	
RW	R0W-0	-	RW-0			

位	名称	描述
Bit[22:23]	LCDCPD	LCD 电荷泵调器频率源选择 (这是设置 LCD internal charge pump 扫描频率, 可依实际显示效果及功耗来做不同类型选用)
		0 LS_CK / 1 or HS_CK/8 (LS_CK 或 HS_CK 由 LCKS 决定)
		1 LS_CK / 2 or HS_CK/16 (LS_CK 或 HS_CK 由 LCKS 决定)
		2 LS_CK / 4 or HS_CK/32 (LS_CK 或 HS_CK 由 LCKS 决定)
Bit[21]	TU2CKS	UART2 频率源选择
		0 Reserve
Bit[20]	ENU2D	UART2 频率源开启控制
		0 关闭
Bit[18:16]	UA2CD	UART2 频率源除频设置
		0 UART2 频率源/ 1
		1 UART2 频率源/ 2
		2 UART2 频率源/ 4
		3 UART2 频率源/ 8
		4 UART2 频率源/ 16
		5 UART2 频率源/ 32
		6 UART2 频率源/ 64
Bit[6:4]	LCDO	LCD 频率源 2 阶除频器设置
		0 LCD 频率源/ 1
		1 LCD 频率源/ 3
		2 LCD 频率源/ 5
		3 LCD 频率源/ 7
		4 LCD 频率源/ 9
		5 LCD 频率源/ 11
6 LCD 频率源/ 13		

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述
		7 LCD 频率源/ 15
Bit[3:1]	LCDE	LCD 频率源 1 阶除频器设置
		0 关闭
		1 LCD 频率源/ 1
		2 LCD 频率源/ 2
		3 LCD 频率源/ 4
		4 LCD 频率源/ 8
		5 LCD 频率源/ 16
		6 LCD 频率源/ 32
		7 关闭
Bit[00]	LCKS	LCD 频率源选择
		0 LS_CK(固定÷8)
		1 HS_CK(固定÷64)

6.3.6. 频率系统寄存器 5

Clock Base Address + 0x014 (0x40314)				
Symbol	CLKCR5 (Clock Control Register 5)			
Bit	[31:24]	[23:18]	[17]	[16]
名称	MASK	-	SPICKS	I2CCKS
RW	R0W-0	R-0	RW-0	RW-0
Bit	[15:8]	[7:6]	[5:4]	[3:0]
名称	MASK	TB2CKS	TM2CD	-
RW	R0W-0	RW-0		-

位	名称	描述
Bit[17]	SPICKS	SPI 频率源选择
		0 HSXT: 外部高速振荡器
		1 HSRC: 内部高速振荡器
Bit[16]	I2CCKS	I ² C 频率源选择
		0 HSXT: 外部高速振荡器
		1 HSRC: 内部高速振荡器
Bit[7:6]	TM2CKS	Timer B2 频率源选择
		0 关闭
		1 HS_CK
		2 HS_CB
		3 LS_CK
Bit[5:4]	TM2CD	Timer B2 频率源除频设置
		0 Timer B2 频率源/ 1
		1 Timer B2 频率源/ 2
		2 Timer B2 频率源/ 4

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



		3	Timer B2 频率源/ 8
--	--	---	-----------------

6.3.7. 频率系统寄存器 6

Clock Base Address + 0x020 (0x40320)				
CLKCR6 (Clock Control Register 6)				
Symbol				
Bit	[31:24]	[23:20]	[19]	[18:16]
名称	MASK	-	-	-
RW	R0W-0	RW-0		
Bit	[15:8]	[7:4]	[3]	[2:0]
名称	MASK	-	-	-
RW	R0W-0	R-0	RW-0	

位	名称	描述
Bit[23:20]	-	保留请勿设定
Bit[18:16]	-	保留请勿设定
Bit[7:4]	-	保留请勿设定
Bit[2:0]	-	保留请勿设定

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

6.3.8. 频率系统寄存器 7

Clock Base Address + 0x024 (0x40324)		
Symbol	CLKCR7 (Clock Control Register 7)	
Bit	[31:26]	[25:16]
名称	Reserved.	-
RW	R-0	RW-0
Bit	[15:8]	[7:0]
名称	MASK	-
RW	ROW-0	RW-0

位	名称	描述
Bit[25:16]	-	保留请勿设定
Bit[7:0]	-	保留请勿设定

6.3.9. 频率系统寄存器 8

Clock Base Address + 0x028 (0x40328)		
Symbol	CLKCR8 (Clock Control Register 8)	
Bit	[31:26]	[25:16]
名称	Reserved.	-
RW	R-0	R-X
Bit	[15:7]	[6:0]
名称	Reserved.	-
RW	R-0	R-X

位	名称	描述
Bit[25:16]	-	保留请勿设定
Bit[6:0]	-	保留请勿设定

7. 中断控制系统

7.1. 整体总说明

中断向量与中断优先权说明:

这个中断模块包含中断启动控制器、中断使能控制器和中断事件标志位寄存器，用于管理整体的中断服务，如通信功能中断、定时器中断、ADC 中断、IO 外部中断。

芯片提供 9 级中断源，同时也提供 4 级中断响应优先权级别，也就是中断 HW0~HW9 中断向量函数响应的优先级，如果当中断响应优先权级别都设置相同的时候，则中断响应优先权从高位到低位为 HW0、HW1 到 HW9。中断服务由中断事件标志位(INTF)，中断事件服务使能启动(INTE)与中断总控制 GIE 和向量地址 HW0~HW9 组成。当中断事件成立之后，则程序计数器 PC 在下一个指令周期会指向程序内存的中断服务向量地址 HW0~HW9 以执行中断服务程序。

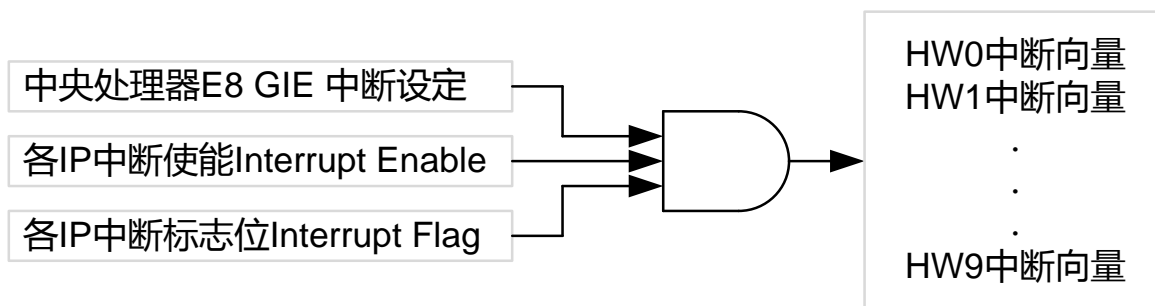


图 7-1 中断服务架构图

Interrupt Vector Address	Vector	Interrupt Function
INT Base Address + 0x00 (I2C/UART/SPI 通信接口)	HW0	void HW0_ISR(void)
INT Base Address + 0x04 (Timer ABC /WDT/ HW RTC)	HW1	void HW1_ISR(void)
INT Base Address + 0x08 (ADC)	HW2	void HW2_ISR(void)
INT Base Address + 0x0C (LVD/BOR2)	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)
INT Base Address + 0x18 (UART2)	HW7	void HW7_ISR(void)
INT Base Address + 0x1C (TMB2)	HW8	void HW8_ISR(void)
INT Base Address + 0x20 (PT3)	HW9	void HW9_ISR(void)

注意：INT HW6 是属于 SW INT

中断群组 HW0~HW9 具有优先权可以设定，提供 4 种优先权等级(0~3)。

0:优先权级别为最高级别

1:优先权级别为次高级别

2:优先权级别为低级级别

3:优先权级别为最低级别

系统预设 HW0~HW9 都设定为级别 0(优先权级别为最高级别)。

当优先权级别都设定为相同时候，则优先权为 HW0>HW1>HW2...>HW9。

举例说明：

设定 HW0 的优先权为级别 1，HW1 的优先权为级别 0，当两中断同时发生的时候，此时会因为优先权级别的设定关系，先进入 HW1 中断。如果设定 HW0 的优先权为级别 0，HW1 的优先权也为级别 0，当两中断同

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

时发生的时候，此时两中断级别虽然设定相同，但是会优先进入 HW0 中断。

操作细节说明:

用户置 1 或清 0 相对应中断使能位，可实现开启或关闭相应的中断功能，写 1 可以开启中断功能。中断事件发生后，会产生中断标志位，使用者可自行清零标志位以便取消中断请求。

必须开启全局中断使能位 GIE=1，否则无法响应任何中断。中断向量优先权决定在多个中断请求同时发生时，需先响应中断优先权高的中断向量。当进入中断向量服务程序，高级的中断向量可以终止当前中断服务转去执行高级中断服务。注意，当进入中断向量服务程序中，GIE 会自动被置为 0，所以需要先把 GIE 置为 1，满足高级中断服务条件成立即可进入高级中断服务程序，当高级中断服务执行完，程序会回到原本的中断服务程序，继续往下执行程序。芯片中断对应的中断向量程序进入地址如下表

7.2. 寄存器地址

Interrupt Register Address	31	24	23	16	15	8	7	0
INT Base Address + 0x00 (INTCOM) (0x40000)	MASK0		REG0		MASK1		REG1	
INT Base Address + 0x04 (INTTMR) (0x40004)	MASK2		REG2		MASK3		REG3	
INT Base Address + 0x08 (INTADC) (0x40008)	MASK4		REG4		MASK5		REG5	
INT Base Address + 0x0C (INTLVD) (0x4000C)	MASK6		REG6		MASK7		REG7	
INT Base Address + 0x10 (INTPT1) (0x40010)	MASK8		REG8		MASK9		REG9	
INT Base Address + 0x14 (INTPT2) (0x40014)	MASK10		REG10		MASK11		REG11	
INT Base Address + 0x18 (INTUART2) (0x40018)	MASK12		REG12		MASK13		REG13	
INT Base Address + 0x1C (INTTMB2) (0x4001C)	MASK14		REG14		MASK15		REG15	
INT Base Address + 0x20 (INTPT3) (0x40020)	MASK16		REG16		MASK17		REG17	

7.3. 寄存器功能

7.3.1. 中断控制寄存器 0

INT Base Address + 0x00 (0x40000)														
Symbol	INTCOM (Interrupt Control Register 0)													
Bit	[31:24]						[23:22]	[21]	[20]	[19]	[18]	[17]	[16]	
名称	MASK						-	I2CEIE	I2CIE	UTxIE	URxIE	STxIE	SRxIE	
RW	R0W-0						-	RW-0						
Bit	[15:14]	[13]	[12]	[11]	[10]	[09]	[08]	[07:06]	[05]	[04]	[03]	[02]	[01]	[00]
名称	MASK							-	I2CEIF	I2CIF	UTxIF	URxIF	STxIF	SRxIF
RW	R-0							-	RW0-0					

(对寄存器写时 Bit15~8 为 MASK，对寄存器读时 Bit15~8 为一般寄存器)

位	名称	描述
Bit[21]	I2CEIE	I ² C 错误中断使能控制
		0 关闭
		1 开启
Bit[20]	I2CIE	I2C 中断使能控制
		0 关闭
		1 开启
Bit[19]	UTxIE	UART 发送(TX)中断使能控制
		0 关闭

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述
		1 开启
Bit[18]	URxIE	UART 接收(RX)中断使能控制
		0 关闭
		1 开启
Bit[17]	STxIE	SPI 发送(TX)中断使能控制
		0 关闭
		1 开启
Bit[16]	SRxIE	SPI 接收(RX)中断使能控制
		0 关闭
		1 开启
Bit[13]	I2CEIR	I2C 错误中断请求
		0 正常
		1 中断
Bit[12]	I2CIR	I2C 中断请求
		0 正常
		1 中断
Bit[11]	UTxIR	UART TX 中断请求
		0 正常
		1 中断
Bit[10]	URxIR	UART RX 中断请求
		0 正常
		1 中断
Bit[09]	STxIR	SPI TX 中断请求
		0 正常
		1 中断
Bit[08]	SRxIR	SPI RX 中断请求
		0 正常
		1 中断
Bit[05]	I2CEIF	I2C 错误中断标志位(准位触发)
		0 正常
		1 I2C 错误发生中断
Bit[04]	I2CIF	I2C 中断标志位(准位触发)
		0 正常
		1 I2C 发生中断
Bit[03]	UTxIF	UART 发送(TX)中断标志位(准位触发)
		0 正常
		1 UART 发送(TX)发生中断

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述
Bit[02]	URxIF	UART 接收(RX)中断标志位(准位触发)
		0 正常
		1 UART 接收(RX)发生中断
Bit[01]	STxIF	SPI 发送(TX)中断标志位(准位触发)
		0 正常
		1 SPI 发送(TX)发生中断
Bit[00]	SRxIF	SPI 接收(RX)中断标志位(准位触发)
		0 正常
		1 SPI 接收(RX)发生中断

7.3.2. 中断控制寄存器 1

INT Base Address + 0x04 (0x40004)														
Symbol	INTTMR (Interrupt Control Register 1)													
Bit	[31:24]						[23:22]	[21]	[20]	[19]	[18]	[17]	[16]	
名称	MASK						-	RTCIE	WDTIE	TMC1IE	TMC0IE	TMBIE	TMAIE	
RW	ROW-0						-	RW-0						
Bit	[15:14]	[13]	[12]	[11]	[10]	[09]	[08]	[07:06]	[05]	[04]	[03]	[02]	[01]	[00]
名称	MASK							-	RTCIF	WDTIF	TMC1IF	TMC0IF	TMBIF	TMAIF
RW	-	RTCIR	WDTIR	TMC1IR	TMC0IR	TMBIR	TMAIR	-	RW0-0					

(对寄存器写时 Bit15~8 为 MASK，对寄存器读时 Bit15~8 为一般寄存器)

位	名称	描述
Bit[21]	RTCIE	实时时钟 RTC 中断使能控制
		0 关闭
		1 开启
Bit[20]	WDTIE	看门狗(WDT)中断使能控制
		0 关闭
		1 开启
Bit[19]	TMC1IE	TMC1 中断使能控制
		0 关闭
		1 开启
Bit[18]	TMC0IE	TMC0 中断使能控制
		0 关闭
		1 开启
Bit[17]	TMBIE	定时计数器 TMB 中断使能控制
		0 关闭
		1 开启
Bit[16]	TMAIE	定时计数器 TMA 中断使能控制

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述
		0 关闭
		1 开启
Bit[13]	RTCIR	RTC 中断请求
		0 正常
		1 中断
Bit[12]	WDTIR	看门狗中断请求
		0 正常
		1 中断
Bit[11]	TMC1IR	Timer C 通道 1 中断请求
		0 正常
		1 中断
Bit[10]	TMC0IR	Timer C 通道 0 中断请求
		0 正常
		1 中断
Bit[9]	TMBIR	TMB 中断请求
		0 正常
		1 中断
Bit[08]	TMAIR	TMA 中断请求
		0 正常
		1 中断
Bit[05]	RTCIF	时钟 RTC 中断标志位
		0 正常
		1 实时时钟 RTC 发生中断
Bit[04]	WDTIF	看门狗(WDT)中断标志位
		0 正常
		1 看门狗(WDT)发生中断
Bit[03]	TMC1IF	TMC1 中断标志位
		0 正常
		1 TMC1 发生中断
Bit[02]	TMC0IF	TMC0 中断标志位
		0 正常
		1 TMC0 发生中断
Bit[01]	TMBIF	定时计数器 TMB 中断标志位
		0 正常
		1 定时计数器 TMB 发生中断
Bit[00]	TMAIF	定时计数器 TMA 中断标志位
		0 正常

HY16F3910 User's Guide

21-bit ENOB $\Sigma\Delta$ ADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述
		1 定时计数器 TMA 发生中断

7.3.3. 中断控制寄存器 2

INT Base Address + 0x08 (0x40008)				
Symbol	INTADC (Interrupt Control Register 2)			
Bit	[31:24]	[23:17]	[16]	
名称	MASK	Rsv	ADCIE	
RW	ROW-0	R-0	RW-0	
Bit	[15:9]	[8]	[07:01]	[00]
名称	Rsv	ADCIR	Rsv	ADCIF
RW	R-0	R-0	R-0	R-0

(对寄存器写时 Bit15~8 为 MASK，对寄存器读时 8 为一般寄存器)

位	名称	描述
Bit[16]	ADCIE	ADC 转换器中断使能控制
		0 关闭
		1 开启
Bit[08]	ADCIR	ADC 中断请求
		0 正常
		1 中断
Bit[00]	ADCIF	ADC 转换器中断标志位
		0 正常
		1 ADC 转换器发生中断;读取 ADO(0x41108)时, ADIF 自动清零

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



7.3.4. 中断控制寄存器 3

INT Base Address + 0x0C (0x4000C)						
Symbol	INTLVD (Interrupt Control Register 3)					
Bit	[31:24]		[23:18]	[17]	[16]	
名称	MASK		Rsv	BOR2IE	LVDIE	
RW	R0W-0		R-0	RW-0	RW-0	
Bit	[15:10]	[09]	[08]	[07:02]	[01]	[00]
名称	MASK			Rsv	BOR2IF	LVDIF
	Rsv	BOR2IR	LVDIR			
RW	R-0		R-0	RW0-X	RW0-0	

(对寄存器写时 Bit15~8 为 MASK，对寄存器读时 Bit9~8 为一般寄存器)

位	名称	描述
Bit[17]	BOR2IE	BOR2 中断使能控制
		0 关闭
		1 开启
Bit[16]	LVDIE	低电压比较器输出(LVDO)中断使能控制
		0 关闭
		1 开启
Bit[09]	BOR2IR	BOR2 中断请求
		0 正常
		1 中断
Bit[08]	LVDIR	低电压比较器输出(LVDO)中断请求
		0 正常
		1 中断
Bit[01]	BOR2IF	BOR2 中断标志位
		0 正常
		1 BOR2 发生中断
Bit[00]	LVDIF	低电压比较器输出(LVDO)中断标志位
		0 正常
		1 低电压比较器输出(LVDO)发生中断

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



7.3.5. 中断控制寄存器 4

INT Base Address + 0x10 (0x40010)									
Symbol	INTPT1 (Interrupt Control Register 4)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	PT17IE	PT16IE	PT15IE	PT14IE	PT13IE	PT12IE	PT11IE	PT10IE
RW	R0W-0	RW-0							
Bit	[15:08]	[07]	[06]	[05]	[04]	[03]	[02]	[01]	[00]
名称	MASK PT1#IR	PT17IF	PT16IF	PT15IF	PT14IF	PT13IF	PT12IF	PT11IF	PT10IF
RW	R-0	RW0-0							

(对寄存器写时 Bit15~8 为 MASK，对寄存器读时 Bit15~8 为一般寄存器)

位	名称	描述
Bit[23:16]	PT1#IE	PT1#IE: PT1.#外部中断使能控制, #代表的是 7~0
		0 关闭
		1 开启
Bit[15:8]	PT1#IR	PT1# 中断请求, #代表的是 7~0
		0 正常
		1 中断
Bit[07:00]	PT1#IF	PT1#IF: PT1.#外部中断标志位, #代表的是 7~0
		0 正常
		1 PT1.# 外部发生中断

7.3.6. 中断控制寄存器 5

INT Base Address + 0x14 (0x40014)									
Symbol	INTPT2 (Interrupt Control Register 5)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	PT27IE	PT26IE	PT25IE	PT24IE	PT23IE	PT22IE	PT21IE	PT20IE
RW	R0W-0	RW-0							
Bit	[15:08]	[07]	[06]	[05]	[04]	[03]	[02]	[01]	[00]
名称	MASK PT2#IR	PT27IF	PT26IF	PT25IF	PT24IF	PT23IF	PT22IF	PT21IF	PT20IF
RW	R0W-0	RW0-0							

(对寄存器写时 Bit15~8 为 MASK，对寄存器读时 Bit15~8 为一般寄存器)

位	名称	描述
Bit[23:16]	PT2#IE	PT2#IE: PT2.#外部中断使能控制, #代表的是 7~0
		0 关闭
		1 开启
Bit[15:8]	PT2#IR	PT2.# 中断请求, #代表的是 7~0
		0 正常
		1 中断
Bit[07:00]	PT2#IF	PT2#IF: PT2.#外部中断标志位, #代表的是 7~0

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述
		0 正常
		1 PT2.# 外部发生中断

7.3.7. 中断控制寄存器 6

INT Base Address + 0x18 (0x40018)								
INTUART2 (Interrupt Control Register 6)								
Symbol	[31:24]			[23:20]	[19]	[18]	[17:16]	
名称	MASK			-	U2TxIE	U2RxIE	-	
RW	ROW-0			-	RW-0		-	
Bit	[15:12]	[11]	[10]	[09:08]	[07:04]	[03]	[02]	[01:00]
名称	MASK					U2TxIF	U2RxIF	-
	-	U2TxIR	U2RxIR	-	-			
RW	R-0			-		RW-0		

(对寄存器写时 Bit15~8 为 MASK，对寄存器读时 Bit11~10 为一般寄存器)

位	名称	描述
Bit[19]	U2TxIE	UART2 发送(TX)中断使能控制
		0 关闭 1 开启
Bit[18]	U2RxIE	UART2 接收(RX)中断使能控制
		0 关闭 1 开启
Bit[11]	U2TxIR	UART TX 中断请求
		0 正常 1 中断
Bit[10]	U2RxIR	UART RX 中断请求
		0 正常 1 中断
Bit[03]	U2TxIF	UART 发送(TX)中断标志位(准位触发)
		0 正常 1 UART 发送(TX)发生中断
Bit[02]	U2RxIF	UART 接收(RX)中断标志位(准位触发)
		0 正常 1 UART 接收(RX)发生中断

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



7.3.8. 中断控制寄存器 7

INT Base Address + 0x1C (0x4001C)					
Symbol	INTTMB2 (Interrupt Control Register 7)				
Bit	[31:24]		[23:18]	[17]	[16]
名称	MASK		-	TMB2IE	-
RW	R0W-0		-	RW-0	-
Bit	[15:10]	[09]	[08]	[07:02]	[01]
名称	MASK		-	TMB2IF	-
	-	TMB2IR			
RW	R-0		-	RW-0	-

(对寄存器写时 Bit15~8 为 MASK，对寄存器读时 Bit9 为一般寄存器)

位	名称	描述
Bit[17]	TMB2IE	Timer B2 中断使能控制
		0 关闭
		1 开启
Bit[09]	TMB2IR	Timer B2 中断请求
		0 正常
		1 中断
Bit[01]	TMB2IF	Timer B2 中断标志位(准位触发)
		0 正常
		1 TMB2 发送发生中断

7.3.9. 中断控制寄存器 8

INT Base Address + 0x20 (0x40020)									
Symbol	INTPT3 (Interrupt Control Register 8)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	PT37IE	PT36IE	PT35IE	PT34IE	PT33IE	PT32IE	PT31IE	PT30IE
RW	R0W-0	RW-0							
Bit	[15:08]	[07]	[06]	[05]	[04]	[03]	[02]	[01]	[00]
名称	MASK	PT37IF	PT36IF	PT35IF	PT34IF	PT33IF	PT32IF	PT31IF	PT30IF
	PT3#IR								
RW	R-0	RW0-0							

(对寄存器写时 Bit15~8 为 MASK，对寄存器读时 Bit15~8 为一般寄存器)

位	名称	描述
Bit[23:16]	PT3#IE	PT3#IE: PT3.#外部中断使能控制, #代表的是 7~0
		0 关闭
		1 开启
Bit[15:8]	PT3#IR	PT3.# 中断请求, #代表的是 7~0
		0 正常
		1 中断
Bit[07:00]	PT3#IF	PT3#IF : PT3.#外部中断标志位, #代表的是 7~0

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述	
		0	正常
		1	PT3.#外部发生中断

8. 看门狗 WDT

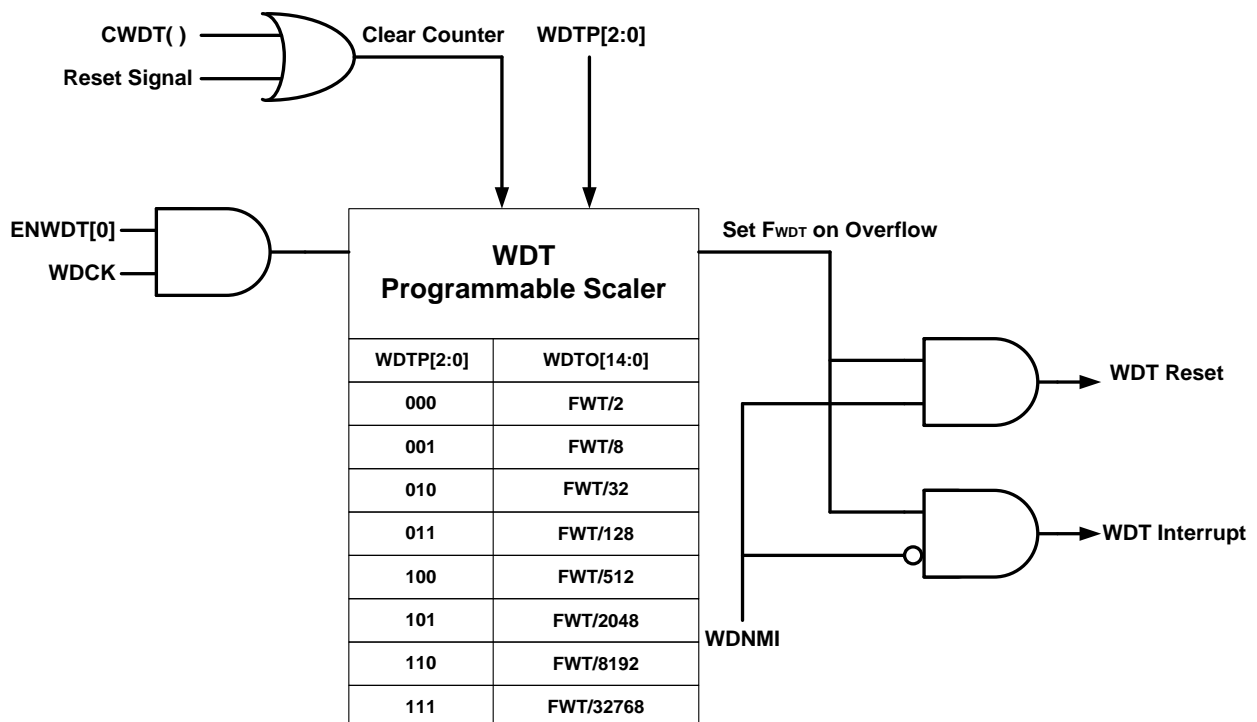
8.1. 整体总说明

看门狗 WDT 为顾名思义的为芯片的看守者，主要用于产生唤醒事件，或芯片发生不可预期当机后，能进行基础自动重置功能。

运行模式: 看门狗计数器溢位产生复位信号，重新启动芯片。可用软件清零定时计数器。

休眠模式: 看门狗 WDT 关闭，无法使用。

待机模式: 看门狗计数器溢位产生中断事件，唤醒芯片。



8.1.1. WDT 的设置说明：

设置除频器 WDT[2:0]可决定 WDT 计数器的工作频率及计数溢出值，计数溢出后可产生 WDT 复位信号或中断事件，控制位 WDNMI 0x40108[6]决定 WDT 计数溢出后产生复位信号还是中断请求信号，写入 0 时，WDT 会产生中断请求信号。作为中断模式请参考中断控制章节的介绍。WDT 必须在芯片运行模式下启动，将控制位 ENWDT 0x40108 [4]置<1>可启动 WDT。开启中断功能需要先使能全局中断控制位 GIE。WDT 的工作频率源为 LSRC，因而 WDT 的工作频率及计数溢出值的理论值计算方式如：

$$WDT = LSRC / 256 / WDT_PS \text{ (式 8-1)}$$

LSRC 为内部低速 RC 振荡器频率，WDT[2:0] 为除频器；

假设 LSRC=33.9kHz，WDT[2:0]=32768，则得到 WDT 工作频率为：

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

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



8.2. 寄存器地址

SOC Register Address	31	16	15	8	7	0
SOC Base Address + 0x08(0x40108)	WDTO0		MASK0		REG0	

8.3. 寄存器功能

8.3.1. WDT 寄存器

SOC Base Address + 0x08 (0x40108)								
Symbol	WDTCR (WDT Control Register)							
Bit	[31:16]							
名称	WDTO							
RW	-	R-0						
Bit	[15]	[14:08]	[07]	[06]	[05]	[04]	[03]	[02:00]
名称	-	MASK	-	WDNMI	CLRWDT	ENWDT	-	WDTP
RW	-	ROW-0	-	RW1-0	RW-0	RW1-0	-	RW-7

位	名称	描述
Bit[31:16]	WDTO	看门狗的计数寄存器
		0 置 0
		1 置 1
Bit[06]	WDNMI	看门狗中断工作模式选择
		0 定时计数模式(Timer mode)
		1 复位模式(Reset Mode) (只要设置 Reset Mode 后,将无法再切换 Timer Mode)
Bit[05]	CLRWDT	看门狗计数器归零控制
		0 关闭
		1 启动
Bit[04]	ENWDT	看门狗开启控制
		0 关闭
		1 开启 (只要设置开启后,将无法关闭)
Bit[2:0]	WDTP	看门狗计数溢出值设置
		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

9. 定时器 Timer A

9.1. 整体总说明

定时器 A 为 16 位递增型计数器，可工作于运行模式和待机模式 (Idle Mode) 和等待模式 (Wait Mode)，可用于产生不同输出频率。

主要特性:

- (01) 递增型计数器
- (02) 总共 16 阶溢位值选择
- (03) 计数溢出产生中断事件
- (04) 可读取计数器的值

定时器 A (TMA) 的设置说明:

TMA 是一个 16 位的递增型计数器。它的输入频率源选择为 TACK。且在 TACK 的每一个上升沿进行计数，输入频率源的频率由频率系统管理模块控制。控制位 ENTA 0x40C00[5] 被置 <1> 或写 <0>，可开启或关闭 TMA 功能。TMA 的计数溢出值通过除频器 TMAS 0x40C00[3:0] 进行设置，用户可通过改变除频器 TMAS 的值来改变计数溢出值，达到不同频率的计数值。控制位 TACL R 0x40C00[4] 被置 <1>，TMA 发生复位，计数寄存器变为 0，计数器寄存器被清零后，控制位 TACL R 自动为 0。TMA 计数溢出后，便会产生中断请求，TMA 中断标志位 TMAIF 0x40004[0] 被置 <1>，若开启 TMA 的中断功能，且全局中断控制位又被置 <1>，芯片就会响应 TMA 的中断请求进入 TMA 中断服务事件。清零 TMA 中断标志位，可取消 TMA 中断请求，此时芯片便不响应 TMA 中断。注意，TMAIF 中断标志位虽可被置为 <0>，但是 TMA 在计数溢出后，因为会发生中断请求，此时 TMAIF 中断标志位还是会自动被置为 <1>。在待机模式下，TMA 中断可用于唤醒。在休眠模式下，TMA 中断不可用。TMA 拥有 16 阶除频设置，使得 TMA 具有比较宽的计数范围。

启动 TMA 后 or 清除 TMAR 后，第一次中断必须忽略不计，因为是从 0000H 开始计数，时间不同于设定值。

TMA 计数溢出值计算方式：

- $TAR[15:0] = 1 / (TACK / 32 / TMAS[3:0])$ (式 9-1)
- TACK 是 TMA 输入频率源，TMAS[3:0] 为除频值；
- 假设 TMA 选择 LS_CK，LS_CK 来自 LPO，则 $TACK = 32\text{kHz}$ ， $TMAS[3:0] = 1001B = 1024$ ，
- 则 Timer A 计数溢出理论值为：
 $35000\text{Hz} / 32 / TMAS (1024) = 35000\text{Hz} / 32 / 1024 = 1.068\text{Hz}$

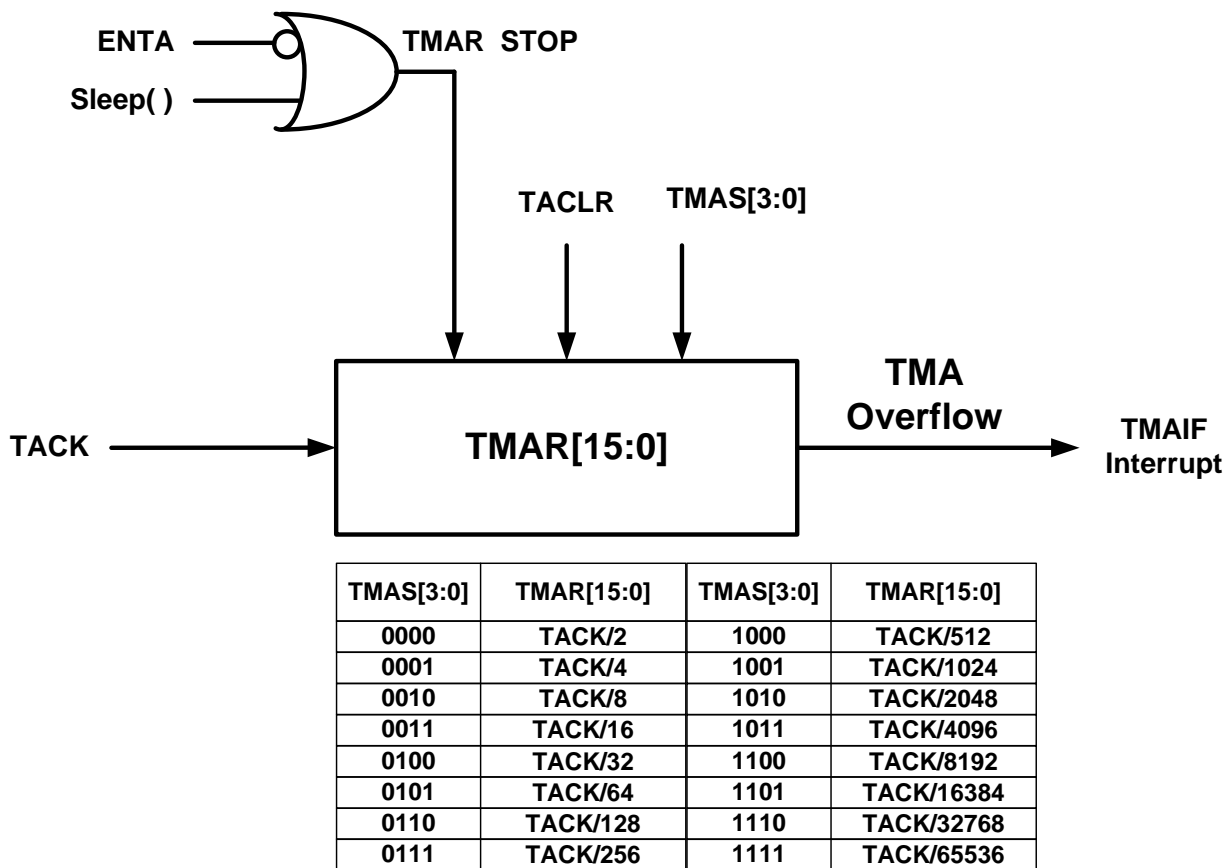


图 9-1 定时计数器 A 方块图

9.2. 寄存器地址

TMA Register Address	31	24	23	16	15	8	7	0
TMA Base Address + 0x00(0x40C00)	TMAR1		TMAR0		MASK0		REG0	

9.3. 寄存器功能

9.3.1. Timer A 寄存器

TMA Base Address + 0x00 (0x40C00)					
TMACR(TMA Control Register)					
Symbol	[31:16]				
Bit	[31:16]				
名称	TMAR				
RW	R-0				
Bit	[15:8]	[07:06]	[05]	[04]	[03:00]
名称	MASK	-	ENTA	TACL	TMAS
RW	ROW-0	-	RW-0		RW-0xF

位	名称	描述
Bit[31:16]	TMAR	Timer A Counter 计数值 TAR[31:16] 是 16-bit Timer A 计数寄存器的计数值，输出值从 MSB 至 LSB
Bit[05]	ENTA	开启定时计数器 Timer A 0 关闭

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述	
		1	开启
Bit[04]	TACLR	清除 Timer A 计数值	
		0	正常
		1	清零(清零完成后该位自动变为 0)
Bit[3:0]	TMAS	Timer A 除频器设定	
		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		

10. 定时器 Timer B

10.1. 整体总说明

定时计数器 B 是一个 16 位的计数器，可用于时间计数或时间控制，时钟产生、延迟时间等功能。它在计数溢出时发生中断信号，程序可读取 TMB 当前计数值；TMB 还用于 PWM 波形的产生。它可工作于运行模式、等待模式和待机模式。定时计数器 B 的 16 位计数寄存器可拆分为两个独立的 8 位计数寄存器，因而 TMB 具有四种计数方式：

- 16 位递增计数方式，可产生中断信号；
- 16 位计数方式，先递增至溢出值，然后从溢出值递减至 0；可产生中断信号
- 两组独立的 8 位递增计数方式，最高计数值为 0xFF，可产生中断信号
- 8+8 位递增计数方式，低 8 位计数器溢出，高 8 位计数器才自动加 1，可产生中断信号

同时 TMB 具有三个计数溢出控制器：TBC0、TBC1、TBC2，TMB 可复用为 PWM 波形发生器，具有两路 PWM 波形 PWM0/PWM1，且每路 PWM 都具有多种工作模式，可适应不同的 PWM 波形输出需求，工作模式如下：PWMA/PWMB/PWMC/PWMD/PWME/PWMF/PWMG

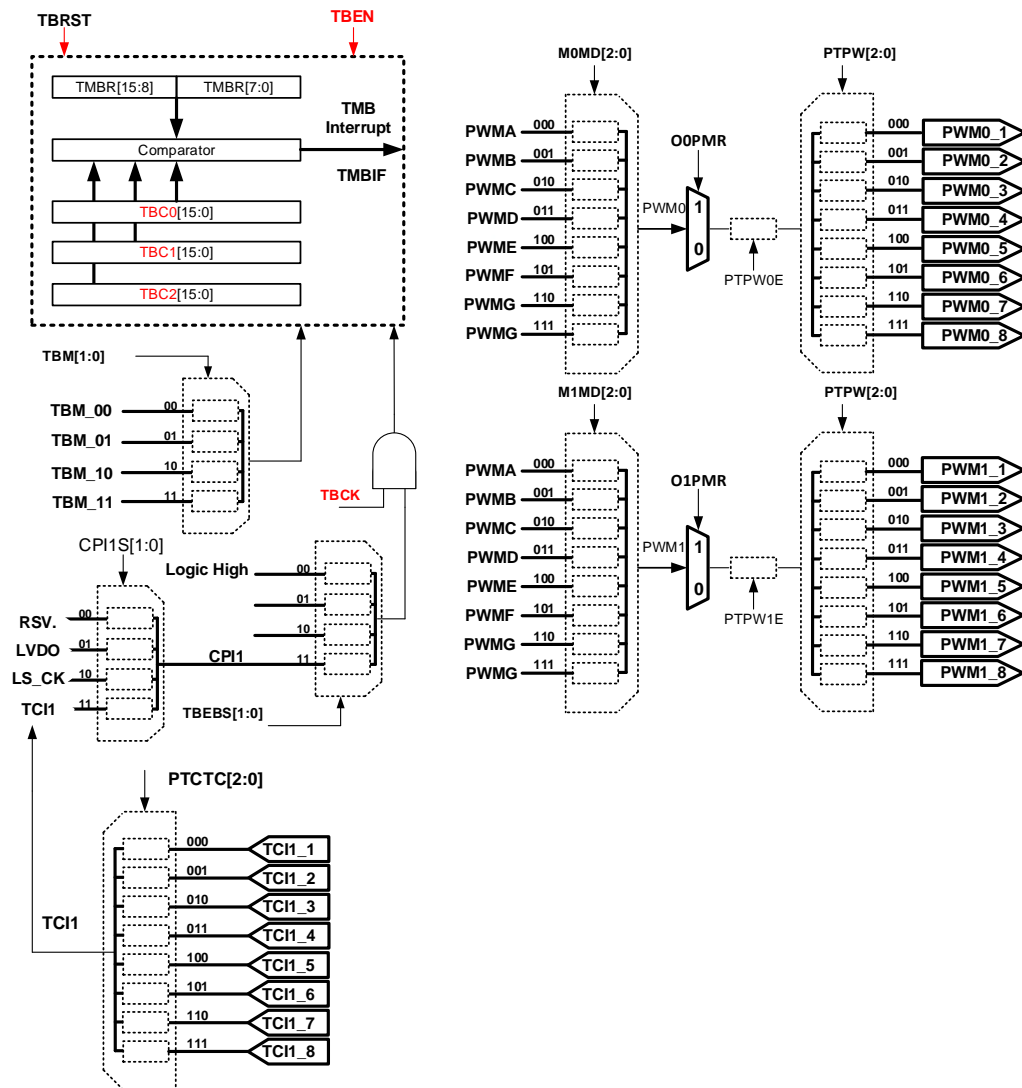


图 10-1 定时计数器 B 方框图

10.1.1. 定时计数器模式

定时器 B(TMB)是一个 16 位递增型计数器,可用于脉冲宽度调制(PMW)波形的产生。可执行时间计数或时间控制、时间计算、时钟产生等功能,且可在计数溢出时产生中断信号。TMB 可工作于运行模式和待机模式(Idle Mode)和等待模式(Wait Mode)。

具有 4 种不同计数方式,可产生不同频率的定时计数值:

- A) 16 位递增计数方式,可产生中断信号
- B) 16 位计数方式,先递增至溢出值,然后从溢出值递减至 0;可产生中断信号
- C) 两组独立的 8 位递增计数方式,最高计数值为 0xFF,可产生中断信号
- D) 8+8 位递增计数方式,低 8 位计数器溢出,高 8 位计数器才自动加 1,可产生中断信号

具有 2 种不同的触发计数信号源,可适用于不同事件计数:

- (1) 总是启用,连续计数方式
- (2) CPI1 输入高电位触发

TMB 的工作频率源为 HS_CK 或 LS_CK,经过除频器产生频率源 TBCK,提供给 TMB 工作频率。提供设置除频器 **TBCD** 0x40308[5:4],可设置 TMB 不同计数周期。在频率系统控制模块可设置 TMB 的频率源。TMBR 是个 16 位的定时/计数寄存器,可拆分为两个独立 8 位的定时/计数寄存器,以适应 TMB 四种不同的计数方式。TMBR 在 TBCK 的上升沿进行递增或递减,在不同的计数方式下,TMBR 以不同的条件发生递增或递减。TMBR 可通过设置控制 TBRST 0x40C04[4]位为<1>,自动清零,并且清零后控制位 TBRST 自动为 0。程序上亦可读取 TMBR 当前计数值,以作为其它用途。TBEN 是 TMB 的使能控制信号,该位被写 1,开启 TMB 计数功能,被写 0 则关闭 TMB 计数功能。TBEBS 0x40C04[1:0]是 TMB 的触发计数信号源控制器,透过该控制器可设置 4 种不同的触发计数信号源。TBM 0x40C04[3:2]是 TMB 的计数方式控制器,透过设置该控制器可设置 4 种不同的计数方式。

TMB 作为定时计数器操作初始化设置:

- 选择 TMB 工作频率源为 HS_CK 或 LS_CK(控制位 TMCKS 0x40308[7]),并且做频率源除频设置与开启动作(控制位 ENTD 0x40308[6]与控制位 TBCD 0x40308[5:4])
- 选择计数模式,设置寄存器控制位 TBM 0x40C04[3:2]
- 选择触发计数信号源,设置寄存器控制位 TBEBS 0x40C04[1:0],作为定时器,可设置为<00>,即总是启用,连续计数;
- 设置定时计数溢出值,设置寄存器控制位 TBC0 0x40C0C[15:0];
- 设置寄存器 0x40C04[4]=1,即控制位 TBRST 置<1>,清零计数寄存器;
- 设置寄存器 0x40C04[5]=1,即控制位 TBEN 被置<1>,使能 TMB。
- TMB 开始计数,TMB 计数溢出后,便会产生中断请求,TMB 中断标志位寄存器 TMBIF 0x40004[1]被置<1>,若开启 TMB 的中断功能,即寄存器控制位 TMBIE 0x40004[17]被置<1>,且全局中断控制位(GIE)又被置<1>,芯片就会响应 TMB 的中断请求进入 TMB 中断服务事件。清零 TMB 中断标志位,可取消 TMB 中断请求,此时芯片便不响应 TMB 中断。

注意,TMBIF 中断标志位虽可被置为<0>,但是 TMB 在计数溢出后,因为会发生中断请求,此时 TMBIF 中断标志位还是会置为<1>。在待机模式下,TMB 中断可用于唤醒。在休眠模式下,TMB 中断不可用。

Timer B 溢出理论值计算方式:

$$T = TBC0 * 1 / TBCK; TBCK = HS_CK(或 LS_CK) / TBCD : (式 10-1)$$

$$T = TBC0 * TBCD / HS_CK(或 LS_CK) ; (式 10-2)$$

TMB 具有四种不同的计数方式，不同的计数方式具有不同的溢出条件，以下做详细介绍。

10.1.1.1. TMB 计数方式 0 :

当寄存器控制位 TBM 0x40C04[3:2]=00b，寄存器的控制位 TMBR 0x40C08[15:0]作为 16 位递增型计数器。TMB 工作于该模式，TMBR 会在 TBCK 的每一个上升沿自动加 1。若 TMBR 计数值大于寄存器的控制位 0x40C0C TBC0[15:0]，TMBR 就会在 TBCK 的下一个上升沿变成 0，且定时器中断标志位 TMBIF 被置<1>，(即 0x40004[1]=1)，若此时开启 TMB 的中断功能及使能全局中断功能，芯片就会响应 TMB 中断。然后，TMBR 重新开始递增计数。该模式的计数波形示意图如下图所示。

此模式下 TMB 的计数周期计算方式： $T = TBC0 * TBCD / HS_CK(或 LS_CK)$

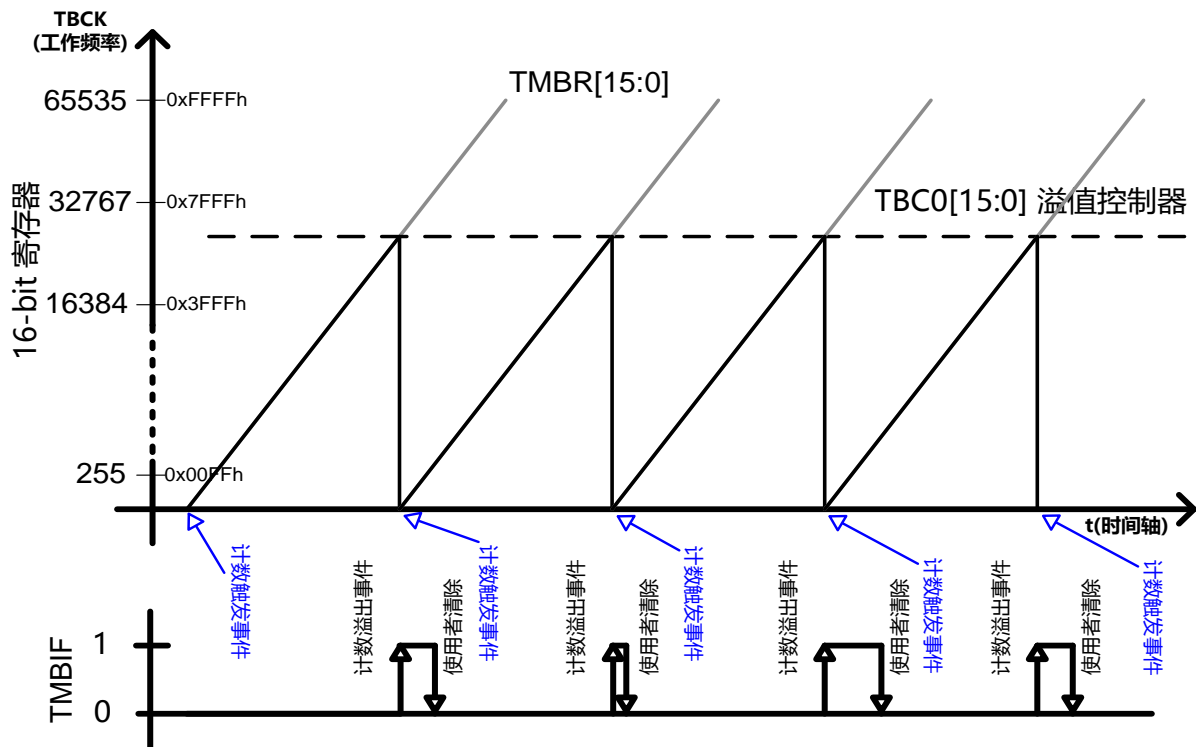


图 10-2 计数方式 0 的计数波形示意图

10.1.1.2. TMB 计数方式 1 :

当 TBM 0x40C04[3:2]=01b，TMB 工作于先递增后递减的计数方式，TMBR 是 16 位计数器。启动 TMB 后，TMBR 先递增计数，TMBR 会在 TBCK 的每一个上升沿自动加 1。当 TMBR 等于 TBC0，TMBR 就会改变成向下模式，但是中断标志位 TMBIF 还是为 0，在 TBCK 的下一个上升沿开始，TMBR 变为递减计数，直到 TMBR 递减至 0 时，发生中断请求，中断标志位 TMBIF 被置<1>，且 TMBR 在 TBCK 的下一个上升沿开始递增计数模式，如此循环运行。该模式的计数波形示意图如下图所示。

此模式下 TMB 的计数周期计数方式为： $T=2 \cdot TBC0 \cdot TBCD / HS_CK$ (或 LS_CK)

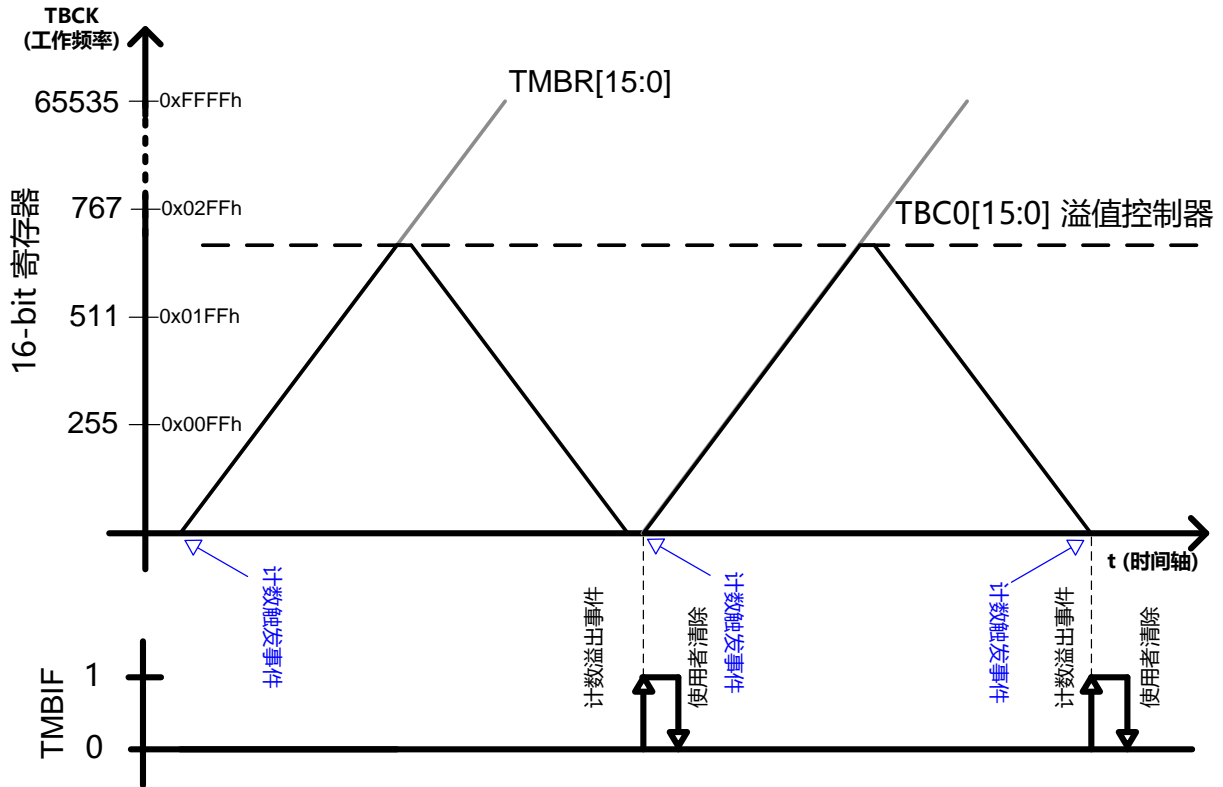


图 10-3 计数方式 1 的计数波形示意图

10.1.1.3. TMB 计数方式 2 :

当 TBM 0x40C04[3:2] = 10b ,TMB 工作与递增计数方式 ,但 TMBR 被拆分为两个 8 位的独立计数器 :TMBR [15:8]和 TMBR [7:0]。且两个独立的 8 位是同时递增计数的。TMBR [15:8]的溢出值由 TBC0 [15:8]所控制,而 TMBR [7:0]的溢出值由 TBC0 [7:0]所控制。这两个计数器在 TBCK 的每一个上升沿自动加 1,若 TMBR [15:8]等于 TBC0 [15:8],则 TBCK 的下一个上升沿会将 TMBR [15:8]变成 0,但中断标志位 TMBIF 仍维持为 0;若 TMBR [7:0]等于 TBC0 [7:0],则会在 TBCK 的下一个上升沿将 TMBR [7:0]变成 0,且同时将中断标志位 TMBIF 置<1>,此时若开启 TMB 中断功能及全局中断使能,芯片就会响应 TMB 中断。该模式是由计数器 TMBR[7:0]控制中断请求,因而在使用该模式时,需要注意设置 TBC0[7:0]的值,以便控制 TMB 中断向量。该模式的计数波形示意图如下图所示。

此模式的中断方式计数周期计算方式为： $T = TBC0[7:0] * TBCD / HS_CK(或 LS_CK)$;

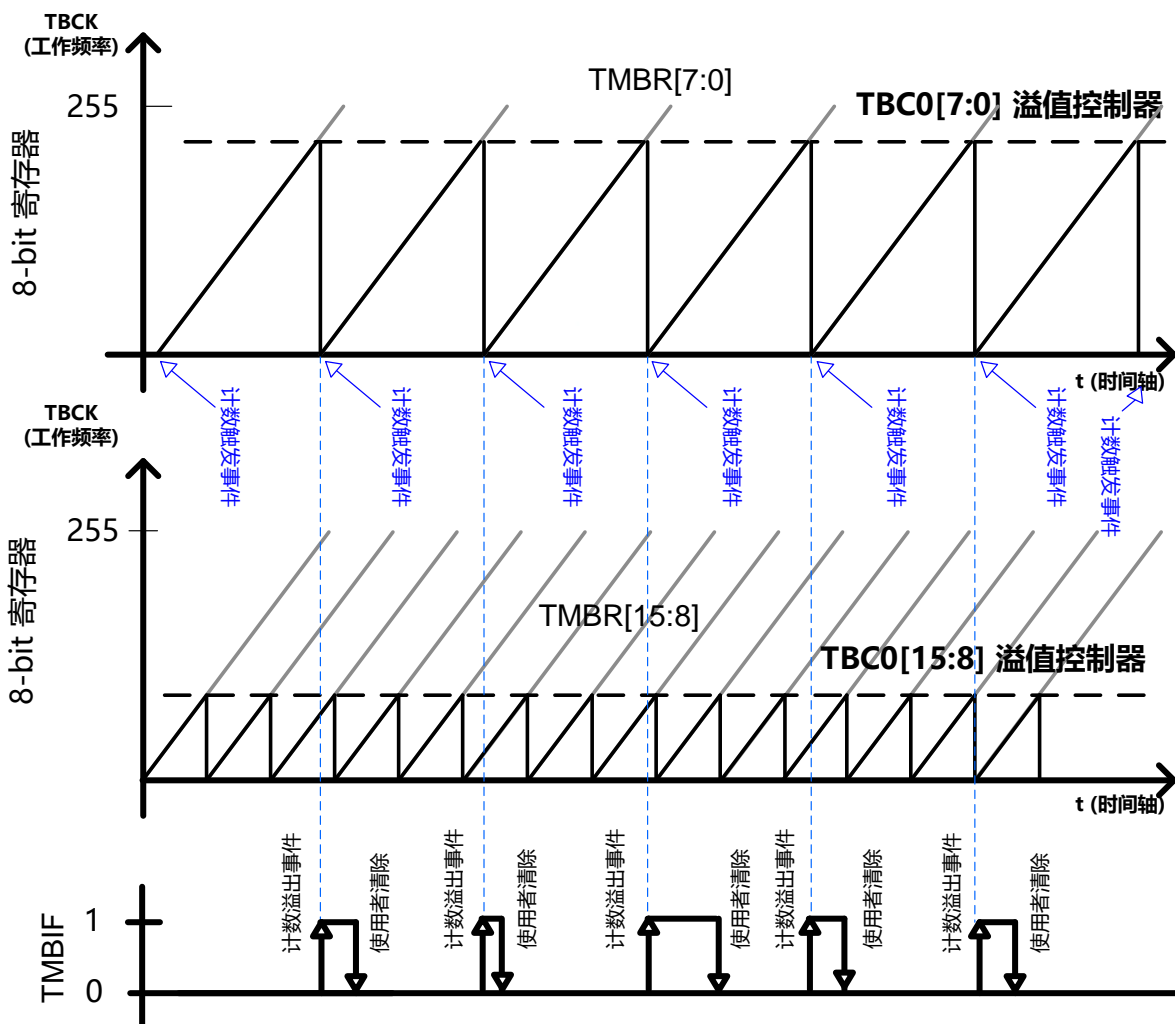


图 10-4 计数方式 2 的计数波形示意

10.1.1.4. TMB 计数方式 3 :

当 TBM 0x40C04[3:2] = 11b ,TMB 工作于递增计数模式 ,TMBR 会分解为两个计数器 :TMBR [15:8]和 TMBR

[7:0]，两者都是递增计数模式。TMBR [7:0]的溢出值由 TBC0 [7:0]所控制。

TMBR [7:0]会在 TBCK 的每一个上升沿自动加 1，若 TMBR [7:0]等于 TBC0 [7:0]，

则 TMBR 会在 TBCK 的下一个上升沿变成 0。此外，TMBIF 会变成 1 且 TMBR [15:8]会增加 1。此时若开启 TMB 中断功能及使能全局中断，芯片便响应中断请求。

该模式的计数波形示意图如下图所示。

此模式中中断方式的计数周期为： $T = TBC0[7:0] * TBCD / HS_CK(或 LS_CK)$ ；

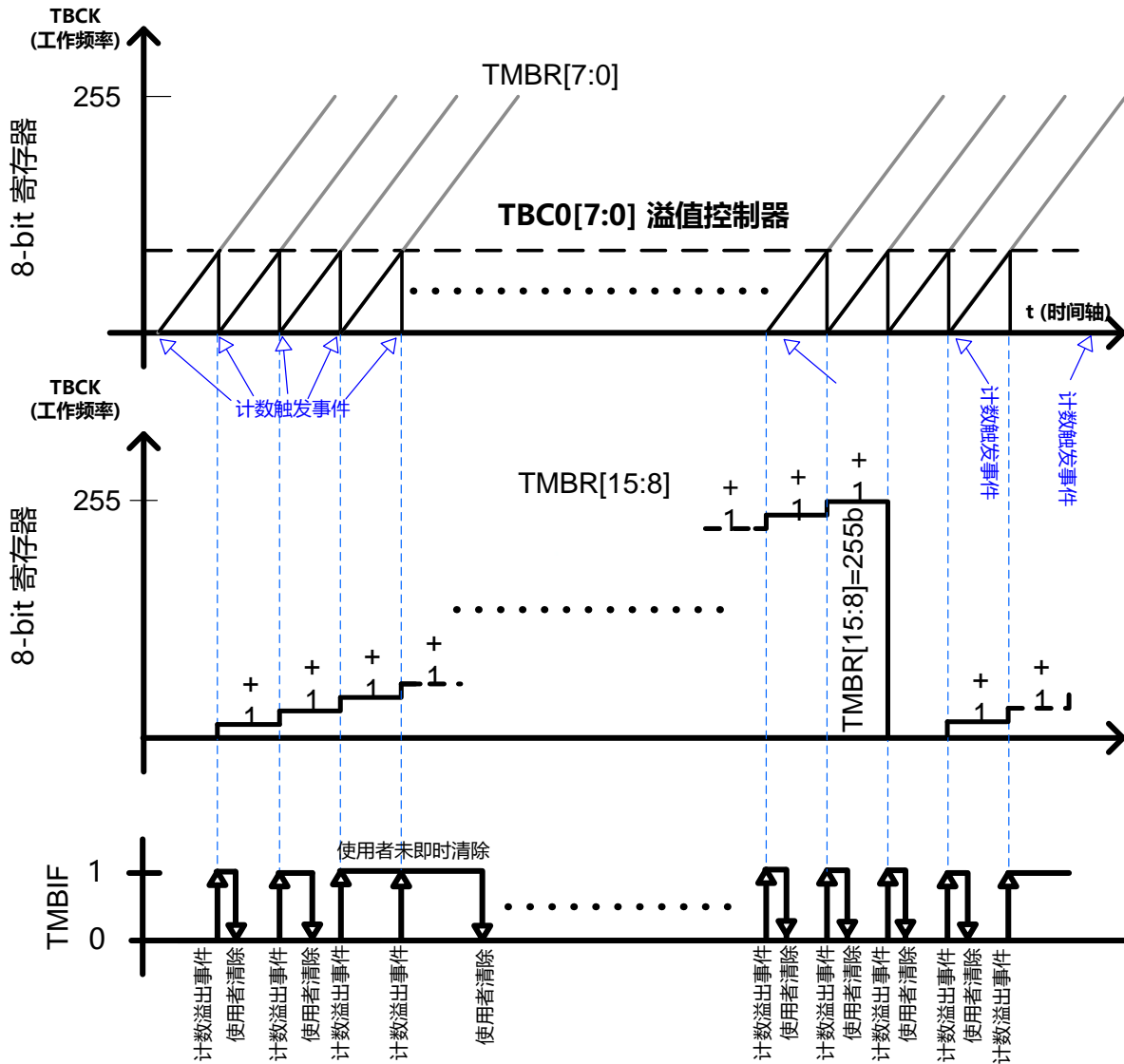


图 10-5 计数模式 3 的计数波形示意图

10.1.2. 系统 PWM 功能模式

定时器 B 工作于 PWM 模式，当 TMB 不同的计数方式与不同脉冲宽度调制(PWM)模式选择器组合使用，可产生多种形式的 PWM 波形。芯片只有两路 PWM：PWM0/PWM1；可简单的认为这是两个 PWM 波形发生器，根据 TMB 不同的计数方式，及 PWM 不同的工作模式，两者组合便可产生多种形式 PWM 波形。芯片为 PWM 输出提供多个输出引脚，每一个 PWM 波形发生器对应 4 个输出 IO 引脚，所以 PWM 的使用及输出比较灵活。但是该功能需要搭配 TMB 使用，也即是需要开启 TMB 及设置 TMB 的计数周期值。

两路 PWM 波形(PWM0/PWM1)都具有多种工作模式：PWMA、PWMB、PWMC、PWMD、PWME、PWMF、PWMG;通过设置寄存器的控制 O0MD 0x40C04[18:16]、O1MD 0x40C04[22:20]，分别设置 PWM0、PWM1 的工作模式。寄存器的控制位 O1PMR 0x40C04[23]、O0PMR 0x40C04[19]的设置，可改变 PWM 输出波形的相位。可透过 PWM 工作模式标志位寄存器 0x40C08[21:16]查看当前的 PWM 工作模式，标志位为 1 标明该工作模式被使能。寄存器的控制位 TBC1 0x40C10[15:0]/TBC2 0x40C10[31:16]分别为 PWM0/PWM1 的占空比控制器，设置 TBC1/TBC2 的值，可调节输出 PWM 的占空比。

芯片为每一路 PWM 提供 4 个输出 IO 引脚对应引脚分布在各个 PTPORT，由寄存器控制位 PTPW 0x40840[4:2]、PTPW1E 0x40840[1]、PTPW0E 0x40840[0]控制 PWM1、PWM0 的输出引脚的选择与开启。控制 PWM 输出引脚的开启与关闭，可以控制到 PWM 的输出与关闭，若想完全关闭 PWM 功能，只能关闭 TMB。PWM 输出引脚如下表所示。

序号 PTPW[2:0]	PWM0 输出引脚	PWM1 输出引脚	序号 PTPW[2:0]	PWM0 输出引脚	PWM1 输出引脚
000	PT1.0	PT1.1	100	PT6.0	PT6.1
001	PT1.4	PT1.5	101	PT7.4	PT7.5
010	PT2.0	PT2.1	110	PT9.0	PT9.1
011	PT2.4	PT2.5	111	PT8.0	PT8.1

表 10-1 PWM0、PWM1 输出引脚分布

PWM 初始化操作说明：

- 选择 PWM 工作模式及占空比、输出波形相位，即设置寄存器 0x40C04 控制位 O0MD /O0PMR、O1MD/O1PMR，写入定时计数溢出值至寄存器 0x40C10 控制位 TBC1/TBC2。
- 控制输出 IO 的使能与关闭，可控制 PWM 的输出与关闭，若想要完全关闭 PWM，就必须关闭 TMB。
- 由寄存器 0x40840 控制位 PTPW、PTPW1E、PTPW0E 控制 PWM1、PWM0 的输出引脚的选择与开启。
- 选择 TMB 工作频率源为 HS_CK 或 LS_CK(控制位 TMCKS 0x40308[7])，并且做频率源除频设置与开启动作(控制位 ENTD 0x40308[6]与控制位 TBCD 0x40308[5:4])。
- 选择计数模式，设置寄存器控制位 TBM 0x40C04[3:2]。
- 选择触发计数信号源，设置寄存器控制位 TBEBS 0x40C04[1:0]，作为定时器，可设置为 00b，即总是启用，连续计数。
- 设置定时计数溢出值，设置寄存器控制位 TBC0 0x40C0C[15:0]。
- 设置寄存器 0x40C04[4]=1，即控制位 TBRST 置<1>，清零计数寄存器。
- 设置寄存器 0x40C04[5]=1，即控制位 TBEN 被置<1>，使能 TMB。

PWM 波形是由 TMBR、TBC0、TBC1、TBC2 组合使用产生的，且工作模式有 7 种不同的模式，因而每种模式的工作条件有所差异，以下针对 7 中模式做出详细介绍。两个独立的 PWM：PWM0 和 PWM1，使

用条件与控制是一样的，所以不做区分介绍。

10.1.2.1. PWMA 模式

PWMA 模式是 16 位 PWM，TMBR 计数值与 TBC1 做比较，PWM 波形周期由 TBC0 控制。

PWM 输出状态控制条件：

PWM = 1，当 TMBR [15:0] >= TBC1 [15:0]；

PWM = 0，当 TMBR [15:0] < TBC1 [15:0]；

PWM 的周期：

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

PWMA 频率与工作周期计算公式：

$$\text{PWMA Frequency} = \frac{\text{TBCCK}}{\text{TBC0}[15:0] + 1}$$

$$\text{PWMA Duty Cycle} = \frac{(\text{TBC0}[15:0] + 1) - \text{TBC1}[15:0]}{\text{TBC0}[15:0] + 1}$$

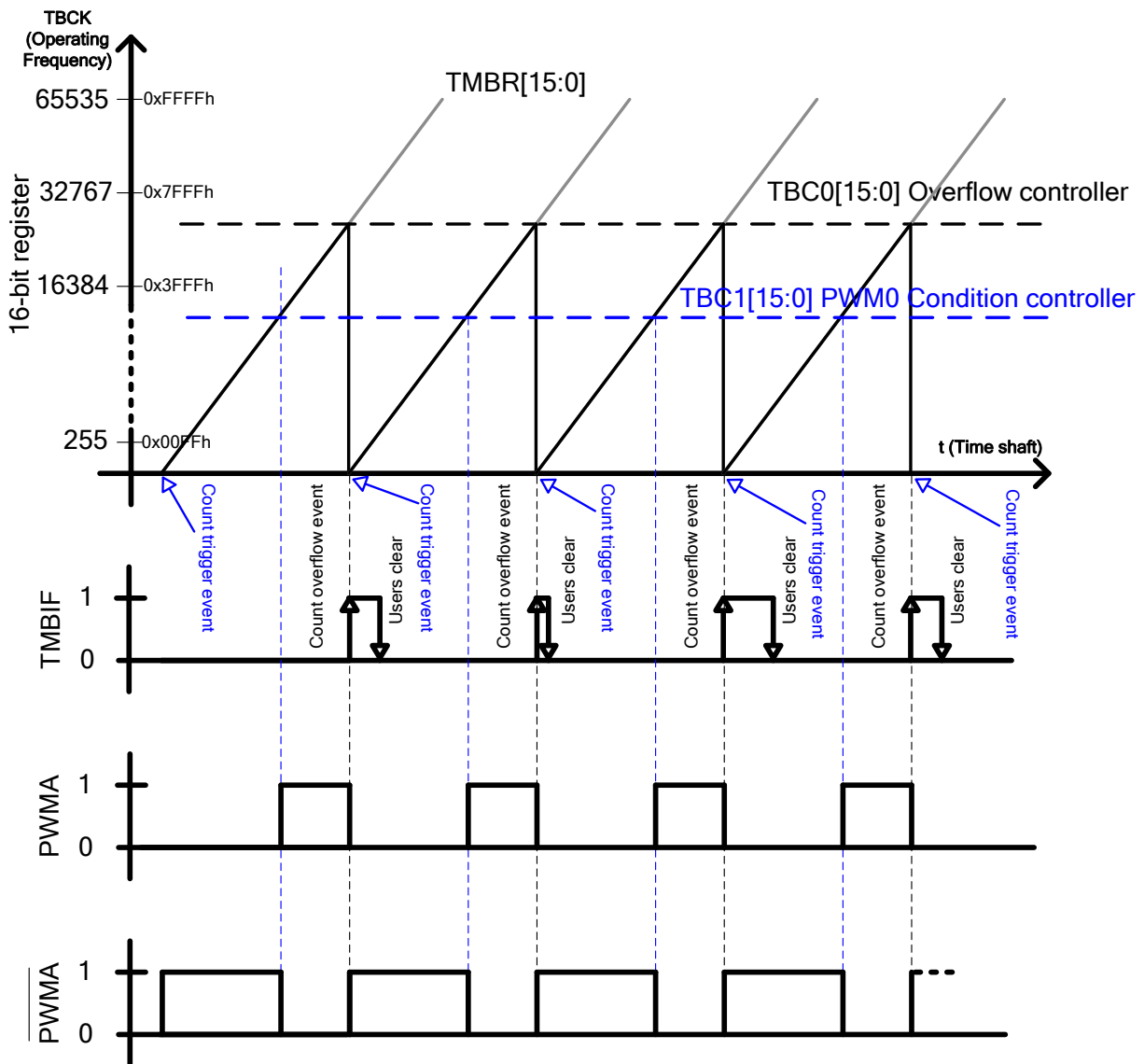


图 10-6 PWM 模式 A 波形示意图及计数波形示意图

10.1.2.2. PWMB 模式

PWMB 模式是 16 位 PWM，TMBR 计数值与 TBC2 做比较，PWM 波形周期值由 TBC0 控制。

PWM 输出状态控制条件：

PWM = 1，当 TMBR [15:0] >= TBC2 [15:0]；

PWM = 0，当 TMBR [15:0] < TBC2 [15:0]；

PWM 的周期：

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

PWMB 频率与工作周期计算公式：

$$\text{PWMB Frequency} = \frac{\text{TBCK}}{\text{TBC0}[15:0] + 1}$$

$$\text{PWMB Duty Cycle} = \frac{(\text{TBC0}[15:0] + 1) - \text{TBC2}[15:0]}{\text{TBC0}[15:0] + 1}$$

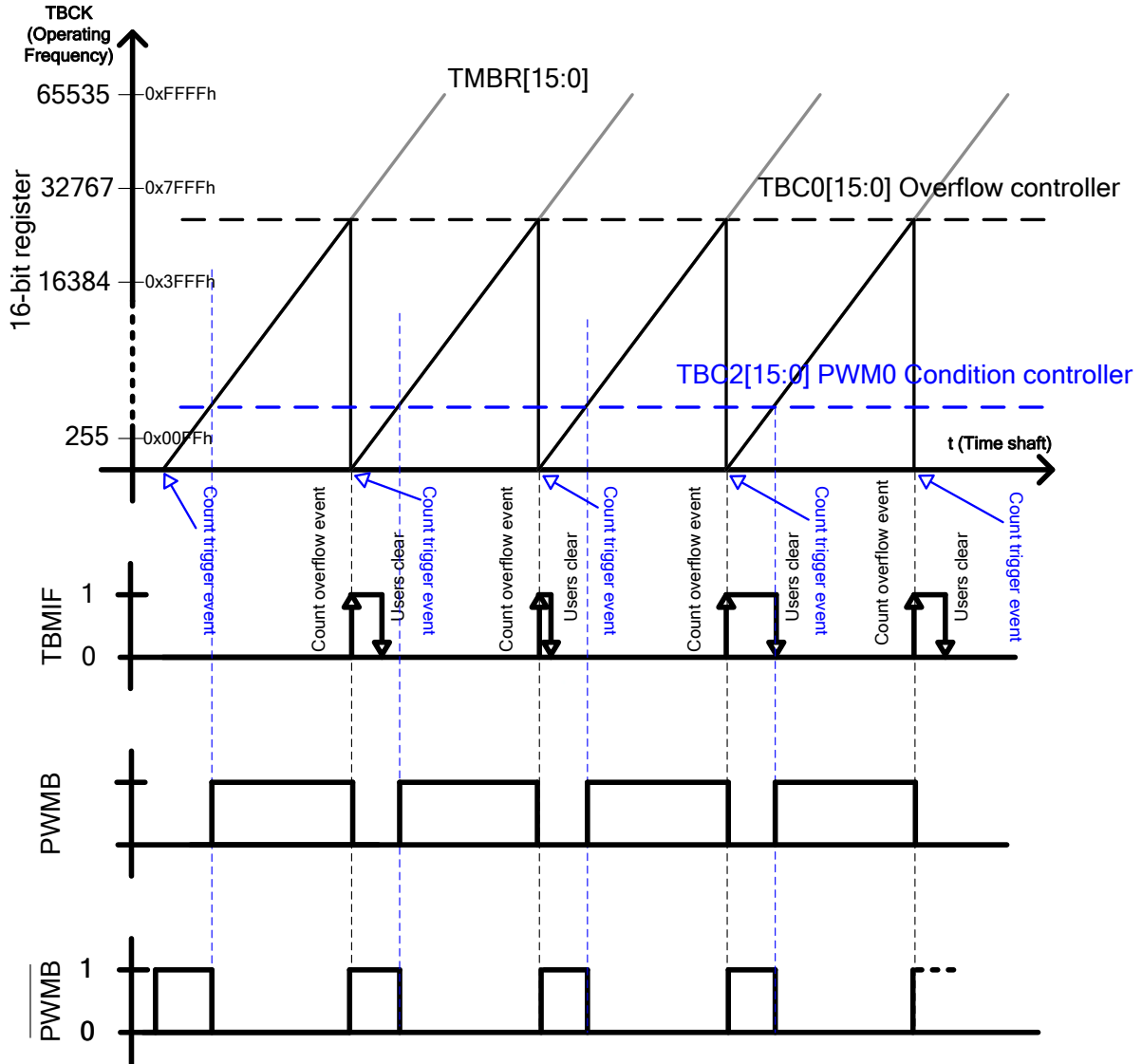


图 10-7 PWM 模式 B 波形图及计数波形示意图

10.1.2.3. PWMC 模式

PWMC 是一个 8 位 PWM，TMBR 计数值与 TBC1 [7:0]做比较，在周期值 TBC0 范围内，出现多个 PWM 波形。

PWM 输出状态控制条件：

PWM = 1，当 TMBR [7:0] >= TBC1 [7:0]。

PWM = 0，当 TMBR [7:0] < TBC1 [7:0]。

PWM 的周期：

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

PWMC 频率与工作周期计算公式：

$$\text{PWMC Frequency} = \frac{\text{TBCK}}{\text{TBC0}[7:0] + 1}$$

$$\text{PWMC Duty Cycle} = \frac{(\text{TBC0}[7:0] + 1) - \text{TBC1}[7:0]}{\text{TBC0}[7:0] + 1}$$

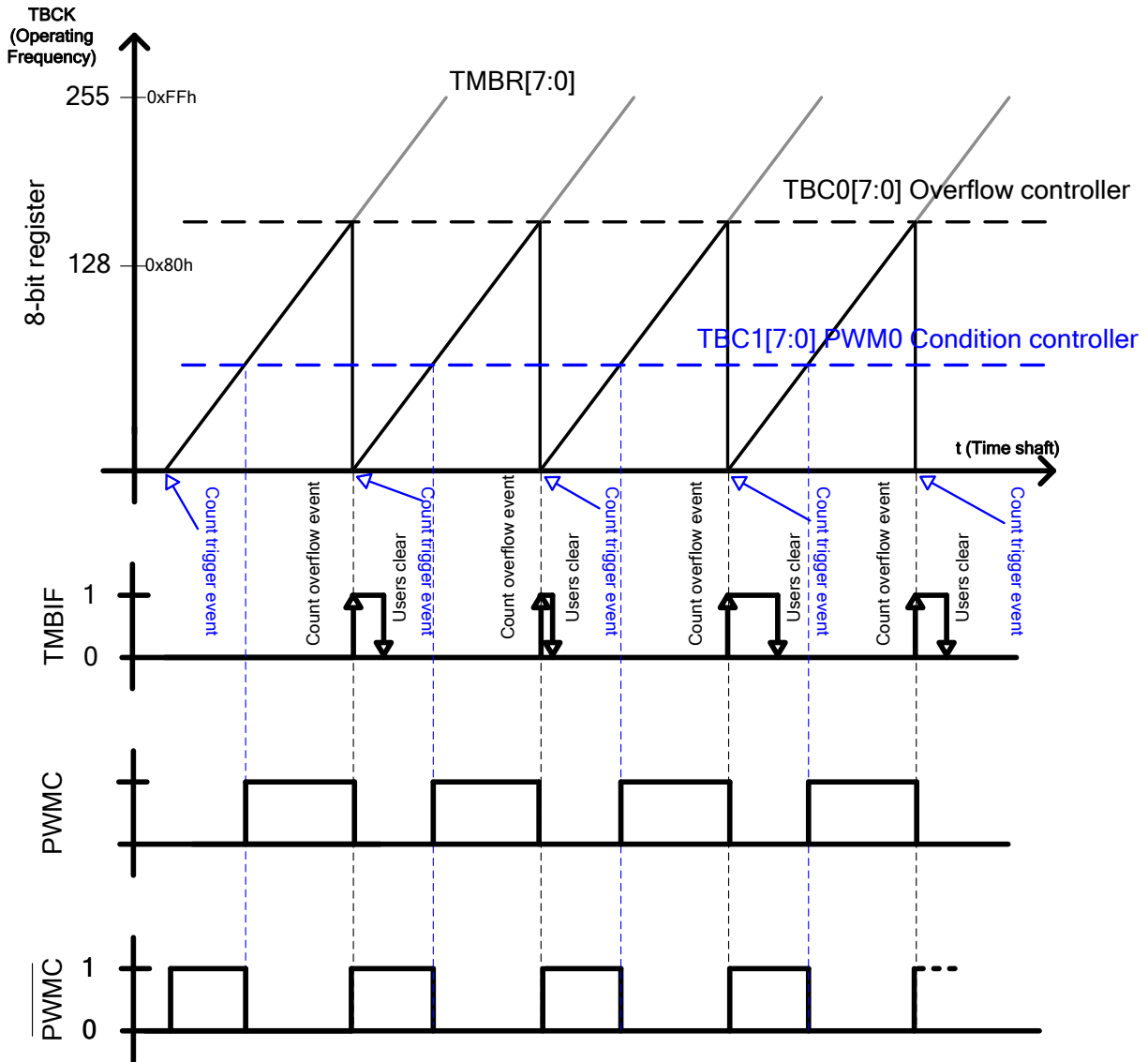


图 10-8 PWM 模式 C 波形图及计数波形示意图

10.1.2.4. PWMD 模式

PWMD 是一个 8 位 PWM，TMBR 计数值与 TBC2 [7:0]做比较，在周期值 TBC0 范围内，出现多个 PWM 波形。

PWM 输出状态控制条件：

PWM = 1，当 TMBR [15:8] >= TBC2 [7:0]；

PWM = 0，当 TMBR[15:8] < TBC2[7:0]；

PWM 的周期：

$$\text{PWM Period} = \text{TMBR}[15:8] * \text{TBCD} / \text{HS_CK(或 LS_CK)} ;$$

PWMD 频率与工作周期计算公式：

$$\text{PWMD Frequency} = \frac{\text{TBCK}}{\text{TBC0}[15 : 8] + 1}$$

$$\text{PWMD Duty Cycle} = \frac{(\text{TBC0}[15 : 8] + 1) - \text{TBC2}[7 : 0]}{\text{TBC0}[15 : 8] + 1}$$

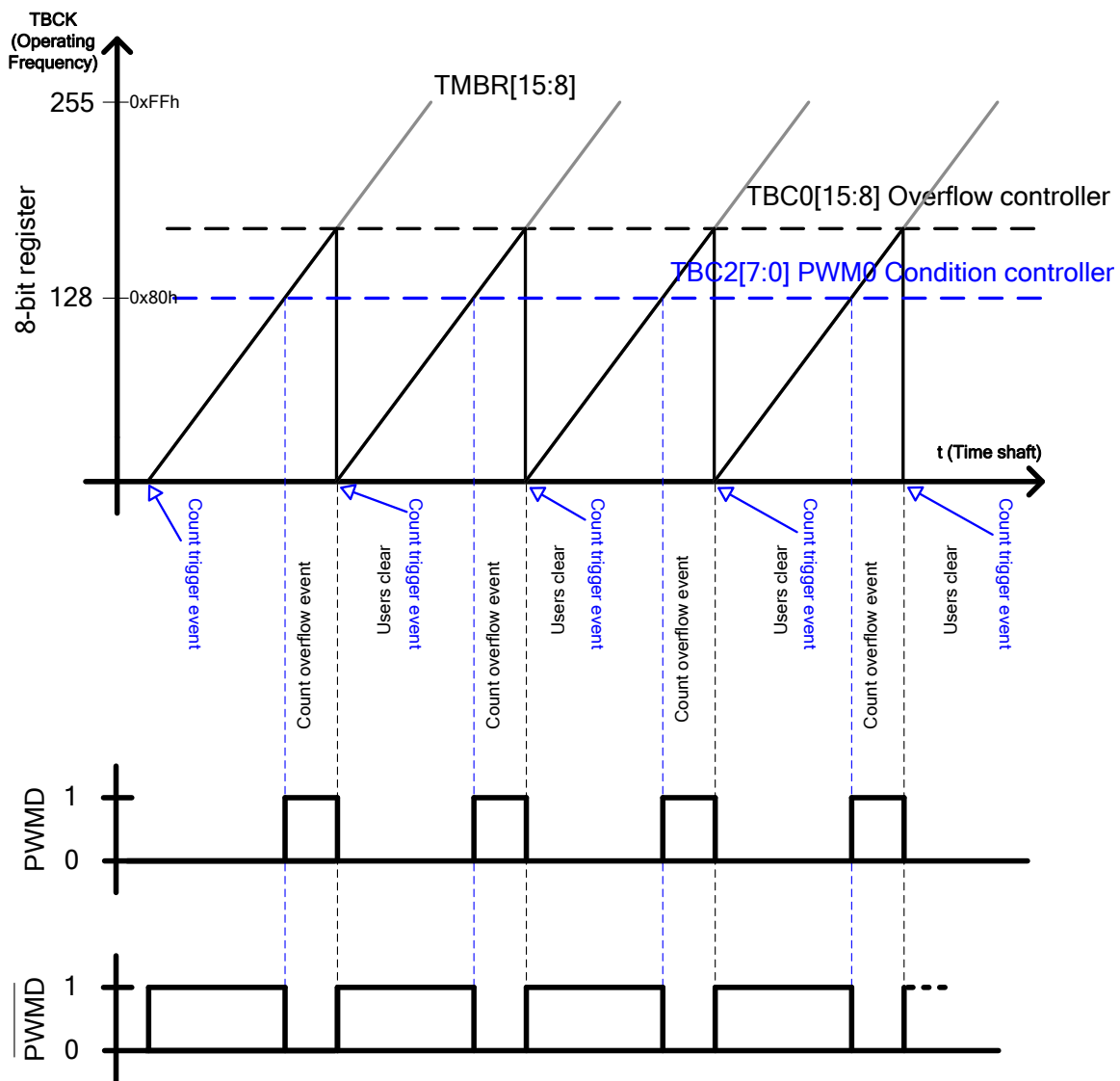


图 10-9 PWM 模式 D 波形图及计数波形示意图

10.1.2.5. PWME 模式(8+8-bit PWM)

将 TMB 计数器设置在 8+8-bit 模式且 PWM 输出波形选择 PWME 则可得到 8+8bit PWM 输出。8+8-bit PWM 由 TMBR[7:0]、TMBR[15:8]、TBC0[7:0]、TBC1[7:0]与 TBC2[7:0]等控制寄存器以及内部数字电路组成。其中 TMBR[7:0]为累进计数器,TBC0[7:0]为 PWM 频率控制器且当 TMBR[7:0]数至 TBC0[7:0]时 TMBR[15:8]累加 1、

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



TBC1[7:0]为 PWM 工作周期控制器、TBC2[7:0]为 8+8-bit PWM 工作周期微调器。

(※以下波形说明，皆以 O1PMR 或 O0PMR 设定为<0>，反相输出作为说明)

8+8-bit PWM 工作周期微调器 TBC2[7:0]设置与说明，如 所示。

设置 加权量	TBC2[7:0]							
	01H	02H	04H	08H	10H	20H	40H	80H
PWM 工作周期 (Duty Cycle)微调	1/2	1/4	1/8	1/16	1/32	1/64	1/128	1/256
说明	2次 TMB 溢出， 一次 N+1 一次 N	4次 TMB 溢出中， 会有 3 个 N 及 1 个 N+1	8次 TMB 溢出中， 会有 7 个 N 及 1 个 N+1	16次 TMB 溢出中， 会有 15 个 N 及 1 个 N+1	32次 TMB 溢出中， 会有 31 个 N 及 1 个 N+1	64次 TMB 溢出中， 会有 63 个 N 及 1 个 N+1	128次 TMB 溢出中， 会有 127 个 N 及 1 个 N+1	256次 TMB 溢出中， 会有 255 个 N 及 1 个 N+1

表 10-1 工作周期微调器设置表

工作周期微调器 TBC2[7:0]说明，其中 N 为工作周期的宽度(注：N = TBC1[7:0])

(A)基本型

(01)TBC2[7:0]设置 01h，使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 2 个输出周期为一组的波形，其中 1 个输出 N 后接着输出 1 个 N+1。

(02)TBC2[7:0]设置 02h，使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 4 个输出周期为一组的波形，其中连续输出 3 个 N 后接着输出 1 个 N+1。

(03)TBC2[7:0]设置 04h，使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 8 个输出周期为一组的波形，其中连续输出 7 个 N 后接着输出 1 个 N+1。

(04)TBC2[7:0]设置 08h，使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 16 个输出周期为一组的波形，其中连续输出 15 个 N 后接着输出 1 个 N+1。

(05)TBC2[7:0]设置 10h，使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 32 个输出周期为一组的波形，其中连续输出 31 个 N 后接着输出 1 个 N+1。

(06)TBC2[7:0]设置 20h，使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 64 个输出周期为一组的波形，其中连续输出 63 个 N 后接着输出 1 个 N+1。

(07)TBC2[7:0]设置 40h，使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 128 个输出周期为一组的波形，其中连续输出 127 个 N 后接着输出 1 个 N+1。

(08)TBC2[7:0]设置 80h，使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 256 个输出周期为一组的波形，其中连续输出 255 个 N 后接着输出 1 个 N+1。

(B)逻辑运算 OR 迭合型

(1/2+1/4,1/2+1/8,-,1/2+1/4+1/8+1/16+1/32+1/64+1/128,1/2+1/4+1/8+1/16+1/32+1/64+1/256 说明与表示)

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



- (01)TBC2[7:0]设置 03h(1/2+1/4), 使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 4 个输出周期为一组的波形, 其中会有 1 个输出 N 另 3 个则为 N+1。
- (02)TBC2[7:0]设置 05h(1/2+1/8), 使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 8 个输出周期为一组的波形, 其中会有 3 个输出 N 另 5 个则为 N+1。
- (03)TBC2[7:0]设置 09h(1/2+1/16), 使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 16 个输出周期为一组的波形, 其中会有 7 个输出 N 另 9 个则为 N+1。
- (04)TBC2[7:0]设置 11h(1/2+1/32), 使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 32 个输出周期为一组的波形, 其中会有 15 个输出 N 另 17 个则为 N+1。
- (05)TBC2[7:0]设置 21h(1/2+1/64), 使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 64 个输出周期为一组的波形, 其中会有 31 个输出 N 另 33 个则为 N+1。
- (06)TBC2[7:0]设置 41h(1/2+1/128), 使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 128 个输出周期为一组的波形, 其中会有 63 个输出 N 另 67 个则为 N+1。
- (07)TBC2[7:0]设置 81h(1/2+1/256), 使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 256 个输出周期为一组的波形, 其中会有 127 个输出 N 另 129 个则为 N+1。
- (08)TBC2[7:0]设置 07h(1/2+1/4+1/8), 使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 8 个输出周期为一组的波形, 其中会有 1 个输出 N 另 7 个则为 N+1。
- (09)TBC2[7:0]设置 0Fh(1/2+1/4+1/8+1/16), 使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 16 个输出周期为一组的波形, 其中会有 1 个输出 N 另 15 个则为 N+1。
- (10)TBC2[7:0]设置 1Fh(1/2+1/4+1/8+1/16+1/32), 使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 32 个输出周期为一组的波形, 其中会有 1 个输出 N 另 31 个则为 N+1。
- (11)TBC2[7:0]设置 3Fh(1/2+1/4+1/8+1/16+1/32+1/64), 使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 64 个输出周期为一组的波形, 其中会有 1 个输出 N 另 63 个则为 N+1。
- (12)TBC2[7:0]设置 7Fh(1/2+1/4+1/8+1/16+1/32+1/64+1/128), 使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 128 个输出周期为一组的波形, 其中会有 1 个输出 N 另 127 个则为 N+1。
- (13)TBC2[7:0]设置 FFh(1/2+1/4+1/8+1/16+1/32+1/64+1/128+1/256), 使得 PWM 工作周期的波形发生 N+1 与 N 输出。即是产生以 256 个输出周期为一组的波形, 其中会有 1 个输出 N 另 255 个则为 N+1。

下表 10-2、图 10-10 部分列出 TBC2[7:0]在不同设置下, 8+8-bit PWM 波形变化以供使用者参考。

8+8bit PWM			TBN 溢位次数																		
型态	TB C2 [7:0]	逻辑运算	0	1	2	3	4	5	6	7	8	9	10	~	1 2 7	1 2 8	~	2 5 2	2 5 3	2 5 4	2 5 5
基本波形	0x01	1/2	N	N+1	N	N+1	N	N+1	N	N+1	N	N+1	N	~	N+1	N	~	N	N+1	N	N+1
	0x02	1/4	N	N	N+1	N	N	N	N+1	N	N	N	N+1	~	N	N	~	N	N	N+1	N
	0x04	1/8	N	N	N	N	N+1	N	N	N	N	N	N	~	N	N	~	N+1	N	N	N
	0x08	1/16	N	N	N	N	N	N	N	N	N	N	N	~	N	N	~	N	N	N	N
	0x10	1/32	N	N	N	N	N	N	N	N	N	N	N	~	N	N	~	N	N	N	N
	0x20	1/64	N	N	N	N	N	N	N	N	N	N	N	~	N	N	~	N	N	N	N
	0x40	1/128	N	N	N	N	N	N	N	N	N	N	N	~	N	N	~	N	N	N	N
	0x80	1/256	N	N	N	N	N	N	N	N	N	N	N	~	N	N+1	~	N	N	N	N
逻辑	0x03	3/4	N	N+1	N+1	N+1	N	N+1	N+1	N+1	N	N+1	N+1	~	N+1	N	~	N	N+1	N+1	N+1

辑 运 算 迭 何 形	0x05	5/8	N	N+1	N	N+1	N+1	N+1	N	N+1	N	N+1	N	~	N+1	N	~	N+1	N+1	N	N+1
	0x07	7/8	N	N+1	N+1	N+1	N+1	N+1	N+1	N+1	N+1	N+1	N	~	N+1	N	~	N+1	N+1	N+1	N+1
	0x0F	15/16	N	N+1	N+1	N+1	N+1	N+1	N+1	N+1	N+1	N+1	N	~	N+1	N	~	N+1	N+1	N+1	N+1
	0x85	161/256	N	N+1	N	N+1	N+1	N+1	N	N	N	N+1	N	~	N+1	N+1	~	N+1	N+1	N	N+1
	0x8F	241/256	N	N+1	N+1	N+1	N+1	N+1	N+1	N+1	N+1	N+1	N	~	N+1	N+1	~	N+1	N+1	N+1	N+1
	0xFF	255/256	N	N+1	N+1	N+1	N+1	N+1	N+1	N+1	N+1	N+1	N+1	~	N+1	N+1	~	N+1	N+1	N+1	N+1

表 10-2 PWME 输出波形示意表

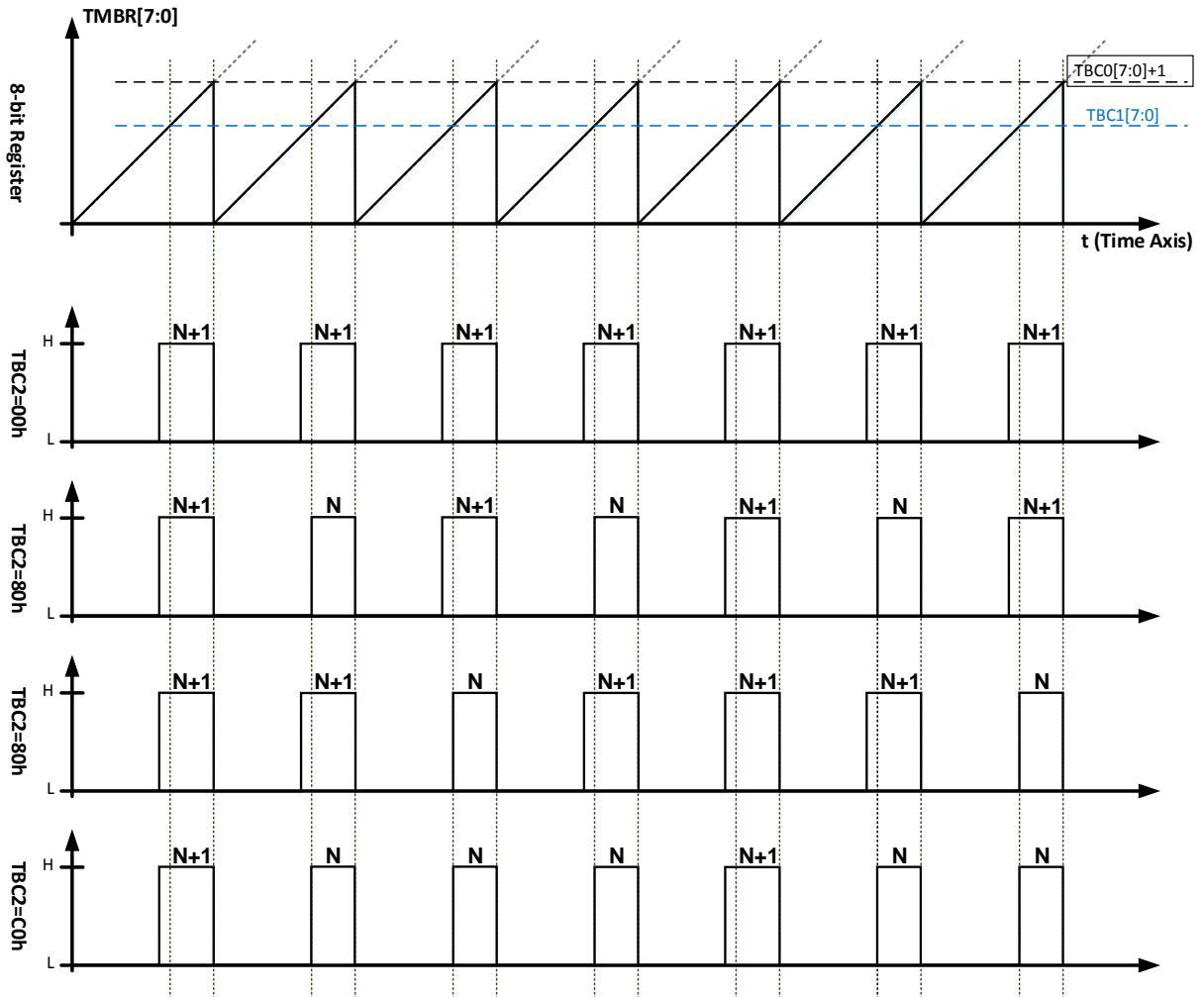


图 10-10 PWME 输出波形示意图

PWME 输出操作说明

(A) 初始化 PWME 频率与工作周期设置：

- (01) 设置寄存器的控制位 TMCKS 0x40308[7]可选择 TMB 的工作频率源，设置控制位 TBCD 0x40308[5:4]以决定 TMB 工作频率。
- (02) 设置寄存器的控制位 TBM 0x40C04[3:2]设置为<11>，将 TMB 规划为 8+8-bit 计数器。
- (03) 设置寄存器的控制位 O0MD 0x40C04[18:16]或控制位 O1MD 0x40C04[22:20]设置为<100>使输出波形为 PWME。
- (04) 将 TBEBS 0x40C04[1:0]设置<00>以选择触发计数信号为总是启用(Always Enable)，即循环计数。

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

- (05)写入数据至 TBC0[7:0]，以决定 PWM 之频率。
- (06)写入数据至 TBC1[7:0]，以决定 PWM 之工作周期(Duty Cycle)。
- (07)写入数据至 TBC2[7:0]，以决定 PWM 之工作周期(Duty Cycle)微调方式。
- (08)将 TBEN 0x40C04[5]设置<1>启用计数器。

(B) 产生 PWME 波形:

- (01)当 TMBR[7:0]计数数值至等于 TBC1[7:0]时，使得 PWME 状态由 0→1。
- (02)当 TMBR[7:0]再计数数值至等于 TBC0[7:0]时，使得 PWME 状态由 1→0；
- (2.1)并产生溢位事件使得 TMBIF 0x40004[1]置<1>并归零重新递增计数，此时 TMBIE 0x40004[17]设置<1>则会产生中断事件服务。
- (2.2)此时，TBC2[7:0]所设置的数据，将使调整 PWME 输出为 N+1 与 N，其中 N = TBC1[7:0]。

(C) PWM 输出控制：

- (01)设置 O0PMR 0x40C04[19]或 O1PMR 0x40C04[23]以决定引脚输出波形是否反相。将 PTPW0E 0x40840[0]或 PTPW1E 0x40840[1]设置为<1>，将 PWM 波形之引脚为输出状态，且 PTPW 0x40840 [4:2]选择适当的 PWM 波形输出脚位将 TBEN 0x40C04[5]设置<0>则关闭计数器与 PWM 输出。

PWME 频率与工作周期计算公式：

$$\text{PWME Frequency} = \frac{TBCK}{TBC0[7:0] + 1}$$

$$\text{PWME Duty Cycle} = \frac{(TBC0[7:0] + 1) - TBC1[7:0] - TBC2[7:0]/256}{TBC0[7:0] + 1}$$

- (02)微调即是当工作周期出现 N+1 情况时有效，公式如下：<X>代表 TBC2 的每个位
微调=< 0 >X128+< 1 >X64+< 2 >X32+< 3 >X16+< 4 >X8+< 5 >X4+< 6 >X2+< 7 >

10.1.2.6. PWMF 模式

PWMF 是一个 16 位 PWM, TMBR 计数值与 TBC1 及 TBC2 做比较, 且 TBC2 的值必须大于 TBC1 的值, TMBR 不断递增至溢出。

PWM 输出状态控制条件：

PWM = 1, 当 $TBC1[15:0] \leq TMBR[15:0] \leq TBC2[15:0]$;

PWM = 0, 当 $TMBR[15:0] > TBC2[15:0]$ 或 $TMBR[15:0] \leq TBC1[15:0]$;

PWM=1 的时间为 $t = t_{clock} \times (TBC2 - TBC1)$;

PWM 的周期值：

PWM Period = $TMBR[15:0] \times TBCD / HS_CK$ (或 LS_CK) ;

PWMF 频率与工作周期计算公式：

由于 PWMF 是工作于特殊波形, 频率与工作周期计算公式在此不描述；

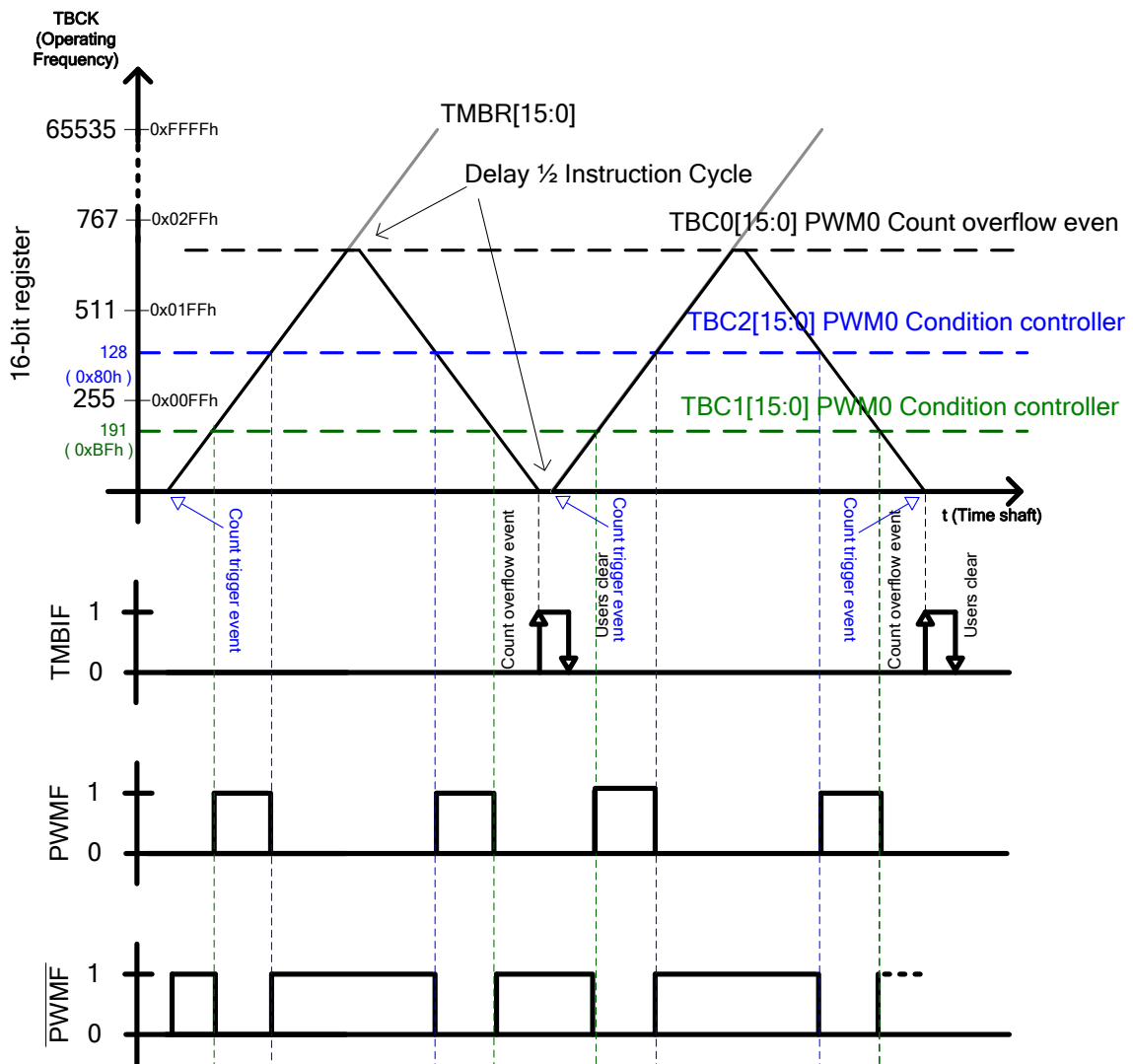


图 10-11 PWM 模式 F 波形图及计数波形示意图

10.1.2.7. PWMG 模式

PWMG 是 16 位的 PWM 模式，且输出波形占空比为 50%，即是输出 PFD 波形。TMBR 计数值不与 TBC1/TBC2 做比较，且输出波形的周期值至于 TBC0 相关。

PWM 的周期值：

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

PWMG 频率与工作周期计算公式：

$$\text{PWMG Frequency} = \frac{\text{TBCK}}{\text{TBC0}[15:0] + 1} \div 2$$

PWMG Duty Cycle = 50%

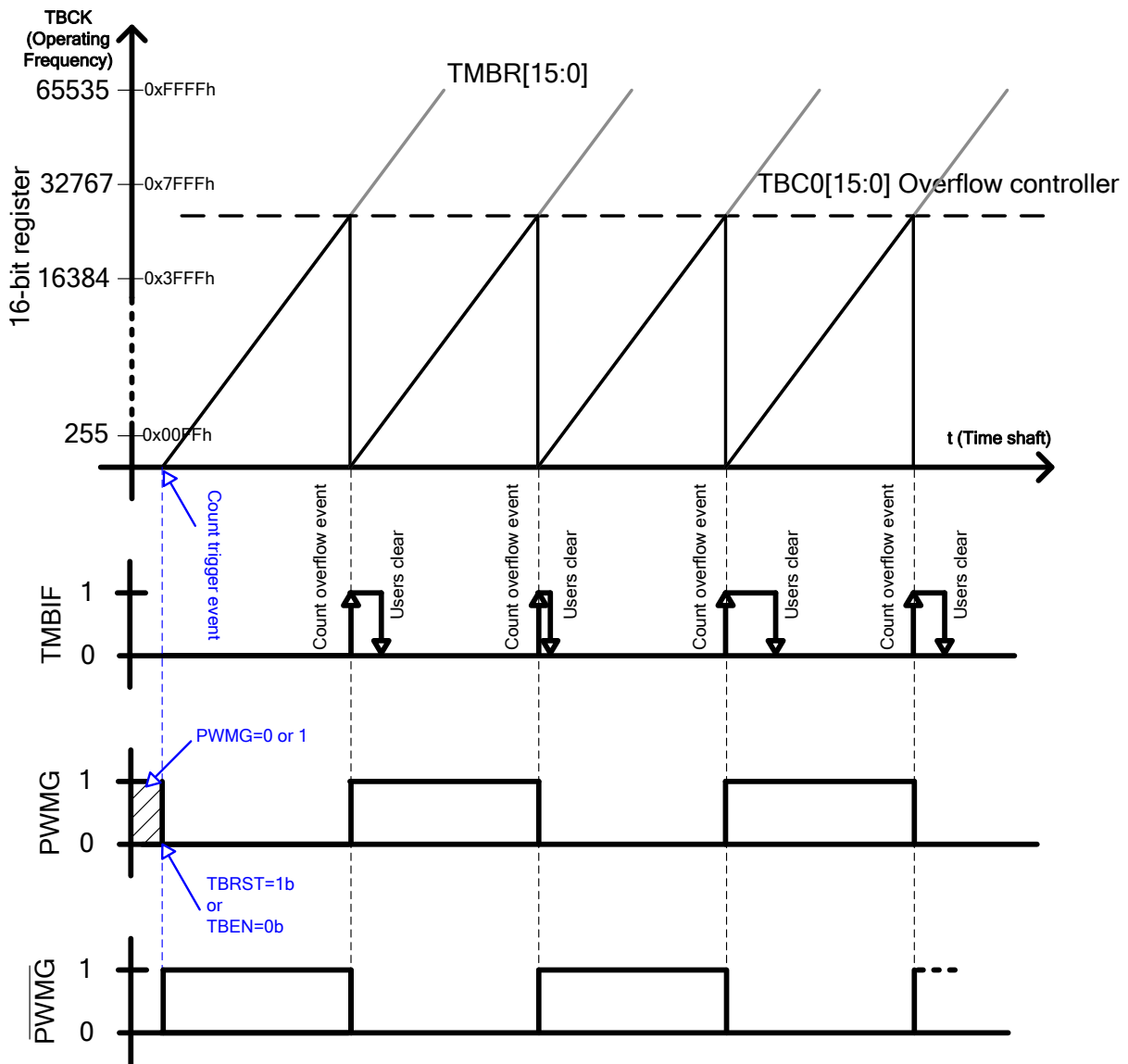


图 10-12 PWM 模式 G 波形图及计数波形示意图

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



10.2. 寄存器地址

TMB Register Address	31	24	23	16	15	8	7	0
TMA Base Address + 0x04(0x40C04)	MASK1		REG1		MASK0		REG0	
TMA Base Address + 0x08(0x40C08)	-		REG2		TBCR		TBCR	
TMA Base Address + 0x0C(0x40C0C)	-		-		TBC0		TBC0	
TMA Base Address + 0x10(0x40C10)	TBC2		TBC2		TBC1		TBC1	

- 保留

10.3. 寄存器功能

10.3.1. Timer B 寄存器 0

TMB Base Address + 0x04 (0x40C04)						
Symbol	TMBCR0(TMB Control Register 0)					
Bit	[31:24]	[23]	[22:20]	[19]	[18:16]	
名称	MASK	O1PMR	O1MD	O0PMR	O0MD	
RW	R0W-0	RW-0				
Bit	[15:08]	[7:6]	[05]	[04]	[03:02]	[01:00]
名称	MASK	-	TBEN	TBRST	TBM	TBEBS
RW	R0W-0	-	RW-0			

位	名称	描述
Bit[23]	O1PMR	PWM1 波形输出相位控制
		0 反相输出
		1 正常输出
Bit[22:20]	O1MD	PWM1 工作模式选择
		000 PWMA
		001 PWMB
		010 PWMC
		011 PWMD
		100 PWME
		101 PWMF
		110 PWMG
		111 PWMG
Bit[19]	O0PMR	PWM0 波形输出相位控制
		0 反相输出
		1 正常输出
Bit[18:16]	O0MD	PWM0 工作模式选择
		000 PWMA
		001 PWMB
		010 PWMC
		011 PWMD
		100 PWME
		101 PWMF
		110 PWMG
		111 PWMG
Bit[05]	TBEN	Timer B 开启控制
		0 关闭

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述	
		1	开启
Bit[04]	TBRST	Timer B 复位	
		0	正常
		1	清零 Timer B 计数寄存器 TMBR ，完成后自动置为 0
Bit[03:02]	TBM	Timer B 计数模式选择	
		00	16-bit 递增计数器，锯齿波类型的计数方式，以步长为 1 递增至最大值 TBC0
		01	16-bit 递增递减计数器，三角波类型的计数方式，以步长为 1 先递增至最大值 TBC0 后再由最大值递减至 0
		10	2 个独立 8Bit 递增计数器 TMBR[15:8]及 TMBR[7:0]，锯齿波类型的计数方式，两个计数器以步长为 1，同时递增计数至最大值 TBC0[15:8]及 TBC0[7:0]
		11	2 个 8Bit 递增计数器 TMBR[15:8]及 TMBR[7:0]，步长为 1 的锯齿波类型计数方式，当计数器 TMBR[7:0]递增计数溢出后，计数器 TMBR[15:8]才自动加 1，且 TMBR[7:0]又从 0 开始递增计数
Bit[01:00]	TBEBS	Timer B 计数触发模式选择	
		00	1 总是启用，连续计数方式
		01	Rsv
		10	Rsv.
		11	CPI1 选择. (由 CPI1S[1:0]所定义)

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

10.3.2. Timer B 寄存器 1

TMB Base Address + 0x08 (0x40C08)							
Symbol	TMBCR1(TMB Control Register 1)						
Bit	[31:22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	-	PWMF	PWME	PWMD	PWMC	PWMB	PWMA
RW	-	R-X					
Bit	[15:00]						
名称	TMBR						
RW	R-X						

位	名称	描述
Bit[21:16]	PWM Flag	PWM A/B/C/D/E/F 工作模式状态标志位
		0 正常
		1 启用
Bit[15:00]	TMBR	Timer B 16-bit 计数值

10.3.3. Timer B 寄存器 2

TMB Base Address + 0x0C (0x40C0C)	
Symbol	TMBCOD(TMB Counter Overflow Condition Register)
Bit	[31:16]
名称	-
RW	-
Bit	[15:00]
名称	TBC0 :Timer B Overflow Condition
RW	RW-0xFFFF

位	名称	描述
Bit[15:00]	TBC0	Timer B 计数溢出门限值

10.3.4. Timer B 寄存器 3

TMB Base Address + 0x10 (0x40C10)	
Symbol	PWMDOD(PWM Counter Overflow Condition Control Register)
Bit	[31:16]
名称	TBC2 : PWM1 占空比 计数溢出值
RW	RW-0xFFFF
Bit	[15:00]
名称	TBC1 : PWM0 占空比 计数溢出值
RW	RW-0xFFFF

位	名称	描述
Bit[31:16]	TBC2	PWM1 占空比 计数溢出值
Bit[15:00]	TBC1	PWM0 占空比 计数溢出值

11. 定时器 Timer B2

11.1. 整体总说明

Timer B2 为 HY16F3910 的第二组 Timer B，操作方法与 Timer B 完全相同，详细使用方法可参考 TimerB 章节。

芯片为每一路 PWM 提供 4 个输出 IO 引脚对应引脚分布在各个 PTPORT，由寄存器控制位 PTPW2、PTPW2E、PTPW3E 控制 PWM2、PWM3 的输出引脚的选择与开启。控制 PWM 输出引脚的开启与关闭，可以控制到 PWM 的输出与关闭，若想完全关闭 PWM 功能，只能关闭 TMB2。PWM 输出引脚如表 11-1 所示。

序号 PTPW2[2:0]	PWM2 输出引脚	PWM3 输出引脚
000	PT1.2	PT1.3
001	PT1.6	PT1.7
010	PT2.2	PT2.3
011	PT2.6	PT2.7
100	PT6.2	PT6.3
101	PT7.6	PT7.7
110	PT9.2	PT9.3
111	PT8.2	PT8.3

表 11-1 PWM2、PWM3 输出引脚分布

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

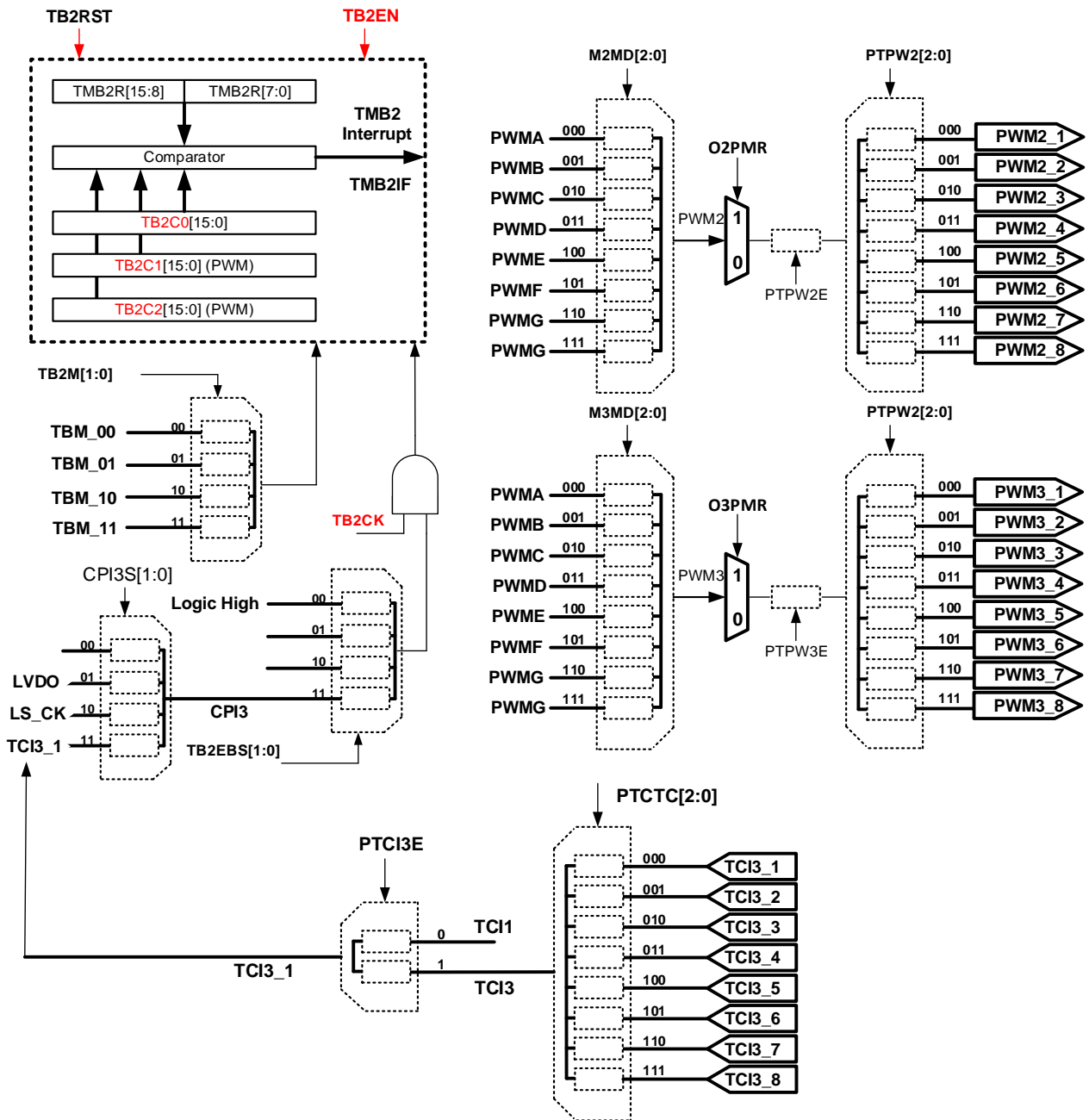


图 11-1 定时计数器 B2 方框图

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



11.2. 寄存器地址

TMB2 Register Address	31	24	23	16	15	8	7	0
TMA Base Address + 0x24(0x40C24)	MASK1		REG1		MASK0		REG0	
TMA Base Address + 0x28(0x40C28)	-		REG2		TB2CR		TB2CR	
TMA Base Address + 0x2C(0x40C2C)	-		-		TB2C0		TB2C0	
TMA Base Address + 0x30(0x40C30)	TB2C2		TB2C2		TB2C1		TB2C1	
TMA Base Address + 0x34(0x40C34)	-		TB2CR2		-		-	

11.3. 寄存器功能

11.3.1. Timer B2 寄存器 0

TMB2 Base Address + 0x24 (0x40C24)							
Symbol	TMB2CR0(TMB2 Control Register 0)						
Bit	[31:24]	[23]	[22:20]	[19]	[18:16]		
名称	MASK	O3PMR	O3MD	O2PMR	O2MD		
RW	R0W-0	RW-0					
Bit	[15:8]	[7]	[6]	[5]	[4]	[3:2]	[1:0]
名称	MASK	-	-	TB2EN	TB2RST	TB2M	TB2EBS
RW	R0W-0	-	RW-0				

位	名称	描述	
Bit[23]	O3PMR	PWM3 波形输出相位控制	
		0	反相输出
		1	正常输出
Bit[22:20]	O3MD	PWM3 工作模式选择	
		000	PWMA
		001	PWMB
		010	PWMC
		011	PWMD
		100	PWME
		101	PWMF
		110	PWMG
		111	PWMG
Bit[19]	O2PMR	PWM2 波形输出相位控制	
		0	反相输出
		1	正常输出
Bit[18:16]	O2MD	PWM2 工作模式选择	
		000	PWMA
		001	PWMB
		010	PWMC
		011	PWMD
		100	PWME
		101	PWMF
		110	PWMG
		111	PWMG

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述	
Bit[05]	TB2EN	Timer B2 开启控制	
		0	关闭
		1	开启
Bit[04]	TB2RST	Timer B2 复位	
		0	正常
		1	清零 Timer B2 计数寄存器 TB2R, 完成后自动置为 0
Bit[03:02]	TB2M	Timer B2 计数模式选择	
		00	16-bit 递增计数器, 锯齿波类型的计数方式, 以步长为 1 递增至最大值 TB2C0
		01	16-bit 递增递减计数器, 三角波类型的计数方式, 以步长为 1 先递增至最大值 TB2C0 后再由最大值递减至 0
		10	2 个独立 8Bit 递增计数器 TB2R[15:8]及 TB2R[7:0], 锯齿波类型的计数方式, 两个计数器以步长为 1, 同时递增计数至最大值 TB2C0[15:8]及 TB2C0[7:0]
		11	2 个 8Bit 递增计数器 TB2R[15:8]及 TB2R[7:0], 步长为 1 的锯齿波类型计数方式, 当计数器 TB2R[7:0]递增计数溢出后, 计数器 TB2R[15:8]才自动加 1, 且 TB2R[7:0]又从 0 开始递增计数
Bit[01:00]	TB2EBS	Timer B2 计数触发模式选择	
		00	1 总是启用, 连续计数方式
		01	Rsv
		10	Rsv
		11	CPI3 选择. (由 CPI3S[1:0]所定义)

11.3.2. Timer B2 寄存器 1

TMB2 Base Address + 0x28 (0x40C28)							
Symbol	TMB2CR1(TMB2 Control Register 1)						
Bit	[31:22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	-	PWMF	PWME	PWMD	PWMC	PWMB	PWMA
RW	-	R-X					
Bit	[15:00]						
名称	TMB2R						
RW	R-X						

位	名称	描述	
Bit[21:16]	PWM Flag	PWM A/B/C/D/E/F 工作模式状态标志位	
		0	正常
		1	启用
Bit[15:00]	TMB2R	Timer B2 16-bit 计数值	

11.3.3. Timer B2 寄存器 2

TMB2 Base Address + 0x2C (0x40C2C)	
Symbol	TMB2COD(TMB2 Counter Overflow Condition Register)
Bit	[31:16]

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

名称	-
RW	-
Bit	[15:00]
名称	TB2C0:Timer B Overflow Condition
RW	RW-0xFFFF

位	名称	描述
Bit[15:0]	TB2C0	Timer B2 计数溢出限值

11.3.4. Timer B2 寄存器 3

TMB2 Base Address + 0x30 (0x40C30)	
Symbol	PWM2DOD(PWM Counter Overflow Condition Control Register)
Bit	[31:16]
名称	TB2C2: PWM3 占空比 计数溢出值
RW	RW-0xFFFF
Bit	[15:00]
名称	TB2C1: PWM2 占空比 计数溢出值
RW	RW-0xFFFF

位	名称	描述
Bit[31:16]	TB2C2	PWM3 占空比 计数溢出值
Bit[15:00]	TB2C1	PWM2 占空比 计数溢出值

11.3.5. Timer B2 寄存器 4

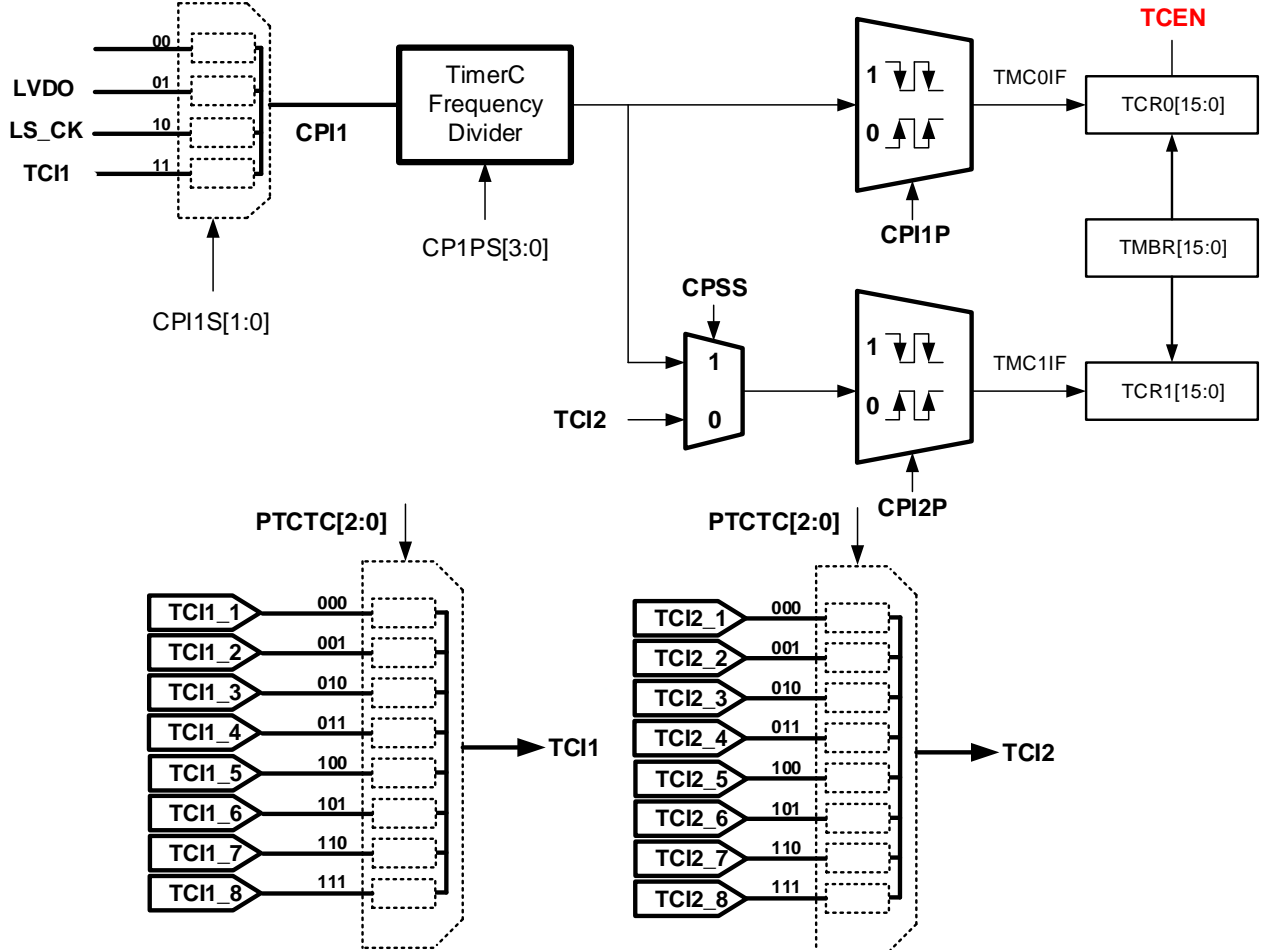
TMB2 Base Address + 0x34 (0x40C34)					
Symbol	TMB2CR1(TMB2 Control Register 2)				
Bit	[31:24]	[23]	[22]	[21:20]	[19:16]
名称	-	CPI3R	RSV	CPI3S	RSV
RW	-	RW-0	-	RW-0	-
Bit	[15:00]				
名称	RSV				
RW	R-0				

位	名称	描述
Bit[23]	CPI3R	Timer B2 TCI3 Input Mode Control
		0 Level Trigger
		1 上升源触发
Bit[21:20]	CPI3S	Timer C Channel3 触发源控制
		00 -
		01 LVDO
		10 LS_CK 低速振荡
		11 TCI3_1, 由 PTCI3E 决定。

12. 定时器 Timer C

12.1. 整体总说明

定时器 C 是设计用来做捕捉的功能，可用于执行频率测量、事件计数、间隔时间测量等功能，可以在计数溢出时产生中断信号。在使用时需要配合 TMB 计数寄存器一起使用。



CP1PS[3:0]	CPI1 Divider	CP1PS[3:0]	CPI1 Divider
0000	CPI1/1	1000	CPI1/256
0001	CPI1/2	1001	CPI1/512
0010	CPI1/4	1010	CPI1/1024
0011	CPI1/8	1011	CPI1/2048
0100	CPI1/16	1100	CPI1/4096
0101	CPI1/32	1101	CPI1/8192
0110	CPI1/64	1110	CPI1/16384
0111	CPI1/128	1111	CPI1/32768

图 12-1 TMC 功能方框图

TMC 频率源选择

TMC 的频率源与 TMB 一致 都是由 HS_CK 或 LS_CK 经过除频器产生频率源 TBCK。使能控制位 TCEN[0] 的设置，可开启或关闭 TMC 捕捉功能。

TMC 捕捉计数值

TMC 的捕捉计数值是由 TMB 来计数器寄存器的控制位 TMBR 0x40C08[15:0]完成，当 Timer B 启动后

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

TMBR 即开始计数，而 CPI1P 发生触发后将 TMBR 的值放置 TCR0 并发生中断(TMC0IF)，CPI2P 发生触发后将 TMBR 的值放置 TCR1 并发生中断(TMC1IF)。

捕捉比较器 1

捕捉比较器 1 有 4 个捕捉信号输入源，透过选择器 CPI1S 0x40C14[21:20]设置输入信号源；且输入信号还需经过除频器 CP1PS 0x40C14[19:16]，除频器的的设置对输入信号源进行除频，可以将输入信号减慢，这样可以测量频率较快的输入信号。同时可以透过控制器 CPI1P 0x40C14[1]的设置，设置捕捉信号的触发沿为上升沿或下降沿捕捉。当捕捉事件完成后，可产生中断信号，中断标志位 TMC0IF 0x40004[2]被置<1>。

捕捉比较器 1 的捕捉信号输入源：

输入信号源符号	功能描述
LVDO	低电压比较器的输出状态
LS_CK	芯片低速频率源
TCI1	从 IO 引脚输入

捕捉比较器 1 输入 IO (当控制位 CPI1S 0x40C14[21:20]=11b 时候)：

序号	TCI1	TCI2	序号	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

捕捉比较器 1 的初始化操作：

- (1) 选择 TMC 工作频率源 TBCK。
- (2) 设置捕捉信号输入源及输入信号源除频值，即设置 CPI1S、CP1PS 的值。
- (3) 设置捕捉信号触发沿，即是设置 CPI1P 的值。
- (4) 若是选择 TCI1 作为捕捉信号输入源，需要设置输入 IO 引脚配置对应 IO 作为输入模式。
- (5) 若使用中断功能，需使能 TMC0IE 0x40004[18]=<1>;并使能全局中断 GIE=<1>。
- (6) 启动 TMC 功能，使能 TCEN 0x40C14[0]=<1>。

捕捉比较器 2

捕捉比较器 2 具有 2 个捕捉信号输入源，透过选择器 CPSS 0x40C14[22]设置不同输入信号源，但其输入信号无须经过除频器。透过控制器 CPI2P 0x40C14[2]可设置信号捕捉触发沿为上升沿或下降沿捕捉。当捕捉事件完成后，可产生中断信号，中断标志位 TMC1IF 0x40004[3]被置<1>。

捕捉比较器 2 的捕捉信号输入为：从 IO 引脚输入；与捕捉比较器 1 的输入源一致；

捕捉比较器 2 的初始化操作：

选择 TMC 工作频率源 TBCK。

设置捕捉信号输入源，即设置 CPSS 0x40C14[22]的值。

设置捕捉信号触发沿，即是设置 CPI2P 0x40C14[2]的值。

若是选择 TCI2 作为捕捉信号输入源，需要设置输入 IO 引脚配置对应 IO 作为输入模式。

若使用中断功能，需使能 TMC1IE 0x40004[19]=<1>；且使能全局中断 GIE=<1>。

启动 TMC 功能，使能 TCEN 0x40C14[0]=<1>。

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



12.2. 寄存器地址

TMC Register Address	31	24	23	16	15	8	7	0
TMC Base Address + 0x14(0x40C14)	MASK1		REG1		MASK0		REG0	
TMC Base Address + 0x18 (0x40C18)	TCR1		TCR1		TCR0		TCR0	

12.3. 寄存器功能

12.3.1. Timer C 寄存器 0

TMC Base Address + 0x14 (0x40C14)							
Symbol	TMCCR0(TMC Control Register 0)						
Bit	[31:24]	[23]	[22]	[21:20]	[19:16]		
名称	MASK	CPI1R	CPSS	CPI1S	CP1PS		
RW	R0W-0	RW-0					
Bit	[15:08]	[7:3]			[2]	[1]	[0]
名称	MASK	-			CPI2P	CPI1P	TCEN
RW	R0W-0	-			RW-0		

位	名称	描述
Bit[23]	CPI1R	Timer B TCI1 输入模式控制
		0 Level Trigger
		1 上升源触发
Bit[22]	CPSS	Capture 2 (Timer C Channel 2)捕捉触发源选择
		0 TCI2 来自 GPIO 引脚的输入
		1 与 Capture 1 (Channel 1)一样的捕捉触发源
Bit[21:20]	CPI1S	Capture 1 (Timer C Channel 1)捕捉触发源选择
		00 Rsv
		01 LVDO 输出
		10 低频频率源 LS_CK
		11 TCI1 来自 GPIO 的输入
Bit[19:16]	CP1PS	Capture1 触发源的除频器设置
		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		
Bit[02]	CPI2P	Capture2 触发源设置

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述	
		0	上升源触发
		1	下降源触发
Bit[01]	CPI1P	Capture1 触发源设置	
		0	上升源触发
		1	下降源触发
Bit[00]	TCEN	Timer C 开启控制	
		0	关闭(但不清零 TCR1 及 TCR2)
		1	开启

12.3.2. Timer C 寄存器 1

TMA Base Address + 0x18 (0x40C18)	
Symbol	TMCCR1(TMC Control Register 1)
Bit	[31:16]
名称	TCR1
RW	R-X
Bit	[15:00]
名称	TCR0
RW	R-X

位	名称	描述
Bit[31:16]	TCR1	Capture2 频率捕捉计数器
Bit[15:00]	TCR0	Capture1 频率捕捉计数器

13. 通用 GPIO PT1 管理

13.1. 整体总说明

PT1 具有 8 个 IO 引脚，可以作为通用的普通 IO 引脚，亦可复用为 SPI、I²C、UART、PWM、外部中断输入等功能模块的输入或输出 IO 引脚。针对不同的复用，需要做不同的设置。

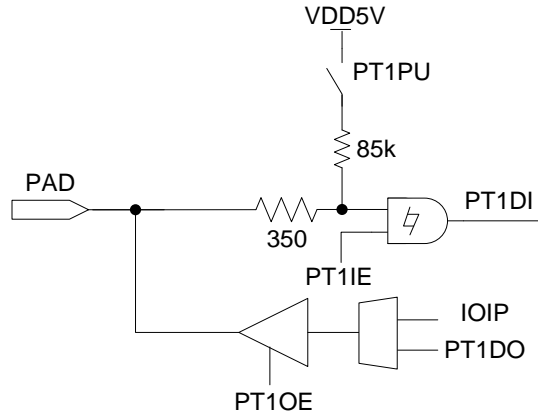


图 13-1 PT1 功能方框图

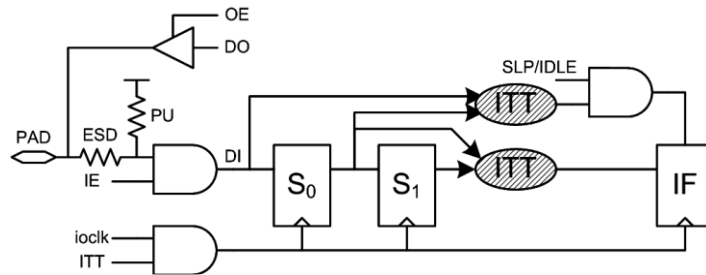


图 13-2 PT1 中断功能方框图

PT1 具有输入、输出、内部上拉电阻及作为外部中断输入引脚的功能，且分别有不同的控制器来设置。

内部上拉电阻

控制器 PT1PU 可设置每个 IO 引脚的内部上拉电阻的开启与关闭，每一位对应一个 IO 引脚。当 IO 引脚对应位被置<1>，则开启内部上拉电阻，若被置<0>，则关闭内部上拉电阻。IO 引脚作为输入模式时，若外部没有上拉电阻，必须开启内部上拉电阻，尤其在低功耗模式时，可以防止漏电，而增加功耗。

输出模式

控制器 PT1OE 可设置每个 IO 引脚输出模式的开启与关闭，每一位对应一个 IO 引脚。当 IO 引脚对应位被置<1>，则开启对应 IO 引脚输出模式；若被置<0>，则关闭输出模式。透过控制位 PT1DO 来控制对应 IO 引脚的输出状态为 1 或 0。在低功耗模式下，若 IO 必须开启输出模式，可根据外围电路来设置输出状态，来减低芯片功耗。此模式下不能开启 IO 内部上拉电阻，且不能同时开启输入、输出模式，因而在开启输出模式时，需要关闭 IO 引脚的输入模式。

输入模式

控制器 PT1IE 可设置每个 IO 引脚输入模式的开启与关闭，每一位对应一个 IO 引脚。当控制器对应位被置<1>，则开启对应 IO 引脚的输入模式；若被置<0>，则关闭输入模式。透过控制器 PT1DI 可读取当前对应 IO 引脚的输入状态为 1 或 0。当 IO 被设置为输入模式，若芯片没有接入外部上拉电阻，则必须开启芯片内部上拉电阻，不能允许 IO 引脚出现浮接状态，以免造成芯片产生漏电现象；特别是在低功耗模式下，建议将 IO 引脚设置为输入模式。作为模拟信号输入引脚时，不用设置对应 IO 引脚为输入模式。在开启输入模式前，需要关闭对应 IO 引脚的输出模式。

外部中断输入

PT1 具有的 8 个 IO 引脚都可复用为外部中断输入引脚。此模式需要将 IO 引脚设置为输入模式且使能内部上拉电阻。需要透过控制器 PT1#ITT 设置外部中断触发沿，并使能控制位 PT1IDF，以使能中断触发沿有效。透过控制器 INTPT1 使能对应 IO 引脚的中断响应功能，当外部中断信号产生时，对应 IO 引脚的中断标志位被置 1。在使能全局中断 GIE 及开启 IO 外部中断功能的条件下，芯片就马上暂停当前程序转去执行 IO 外部中断程序。

13.2. 寄存器地址

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)	PT1IDF		PT1#ITT		PT1#ITT		PT1#ITT	

-保留

说明：上列表中#代表 0~7

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



13.3. 寄存器功能

13.3.1. PT1 寄存器 0

GPIO Base Address + 0x00 (0x40800)									
Symbol	PT1CR0 (PT1 Control Register 0)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	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]
名称	MASK	PT1OE7	PT1OE6	PT1OE5	PT1OE4	PT1OE3	PT1OE2	PT1OE1	PT1OE0
RW	R0W-0	RW-0							

位	名称	描述
Bit[23:16]	PT1PU#	PT1.# 内部上拉控制(PT1 Pull High Enable), #代表的是 7~0
		0 关闭内部上拉
		1 开启内部上拉
Bit[07:00]	PT1OE#	PT1.# PAD 输出模式开启控制(PT1 Output Enable), #代表的是 7~0
		0 关闭输出模式
		1 开启输出模式

13.3.2. PT1 寄存器 1

GPIO Base Address + 0x04 (0x40804)									
Symbol	PT1CR1 (PT1 Control Register 1)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	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]
名称	MASK	PT1DO7	PT1DO6	PT1DO5	PT1DO4	PT1DO3	PT1DO2	PT1DO1	PT1DO0
RW	R0W-0	RW-0							

位	名称	描述
Bit[23:16]	PT1IE#	PT1.# PAD 输入模式开启控制(PT1 Input Enable), #代表的是 7~0
		0 关闭输入模式
		1 开启输入模式
Bit[7:00]	PT1DO#	PT1.# PAD 输出状态值(PT1 Output Data), #代表的是 7~0
		0 输出低电位
		1 输出高电位

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



13.3.3. PT1 寄存器 2

GPIO Base Address + 0x08 (0x40808)									
Symbol	PT1CR2 (PT1 Control Register 2)								
Bit	[31:16]								
名称	-								
RW	-								
Bit	[15:8]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	-	PT1DI[7]	PT1DI[6]	PT1DI[5]	PT1DI[4]	PT1DI[3]	PT1DI[2]	PT1DI[1]	PT1DI[0]
RW	-	R-0							

位	名称	描述
Bit[7:0]	PT1DI#	PT1.# PAD 输入状态值(PT1DI: PT1 Data Input), #代表的是 7~0
		0 输入低电位
		1 输入高电位

13.3.4. PT1 寄存器 3

GPIO Base Address + 0x1C (0x4080C)						
Symbol	PT1CR3 (PT1 Control Register 3)					
Bit	[31:24]	[23:21]	[21:18]	[17:16]		
名称	PT17IDF~ PT10IDF	PT17ITT	PT16ITT	PT15ITT		
RW	R-0		RW-0			
Bit	[15]	[14:12]	[11:9]	[8:6]	[5:3]	[2:0]
名称	PT15ITT	PT14ITT	PT13ITT	PT12ITT	PT11ITT	PT10ITT
RW	RW-0					

位	名称	描述	
Bit[31]	PT17IDF	PT1.7 中断条件标志位 (举例 :可在进入 Sleep Mode 之前判断该 bit 为 0b 或 1b. 如果为 1b 代表中断条件可成立, 则进入 Sleep Mode 之后, 可以透过 PT1.7 唤醒, 如果为 0b, 则无法透过 PT1.7 唤醒)	
		When PT17ITT=0	Always 0. 说明 : 当 PT17ITT 设定为 000, 则 Bit[31]=0b
		When PT17ITT=1	Inverse DI. : 说明 :进入 Sleep Mode 之前, 当 PT1.7=Low 状态时候, 则此时 Bit[31]=1b
		When PT17ITT=2	Same as DI. 说明: 进入 Sleep Mode 之前, 当 PT1.7=High 状态时候, 则此时 Bit[31]=1b
		When PT17ITT=3	Same as S1. 说明: PT1.7 有电位变化, 即触发产生中断
		When PT17ITT=4	Same as DI. 说明: 进入 Sleep Mode 之前, 当 PT1.7=High 状态时候, 则此时 Bit[31]=1b
		When PT17ITT=5	Inverse DI. 说明: 进入 Sleep Mode 之前, 当 PT1.7=Low 状态时候, 则此时 Bit[31]=1b
		When PT17ITT=6	Same as DI. 说明: 进入 Sleep Mode 之前, 当 PT1.7=High 状态时候, 则此时 Bit[31]=1b

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述	
		When PT17ITT=7	Inverse DI. 说明: 进入 Sleep Mode 之前 当 PT1.7=Low 状态时候, 则此时 Bit[31]=1b
Bit[30]	PT16IDF	PT1.6 中断条件标志位	
		When PT16ITT=0	Always 0
		When PT16ITT=1	Inverse DI
		When PT16ITT=2	Same as DI
		When PT16ITT=3	Same as S1
		When PT16ITT=4	Same as DI
		When PT16ITT=5	Inverse DI
		When PT16ITT=6	Same as DI
		When PT16ITT=7	Inverse DI
Bit[29]	PT15IDF	PT1.5 中断条件标志位	
		When PT15ITT=0	Always 0.
		When PT15ITT=1	Inverse DI.
		When PT15ITT=2	Same as DI.
		When PT15ITT=3	Same as S1.
		When PT15ITT=4	Same as DI.
		When PT15ITT=5	Inverse DI.
		When PT15ITT=6	Same as DI.
		When PT15ITT=7	Inverse DI. 说明:
Bit[28]	PT14IDF	PT1.4 中断条件标志位	
		When PT14ITT=0	Always 0
		When PT14ITT=1	Inverse DI
		When PT14ITT=2	Same as DI
		When PT14ITT=3	Same as S1
		When PT14ITT=4	Same as DI
		When PT14ITT=5	Inverse DI
		When PT14ITT=6	Same as DI
		When PT14ITT=7	Inverse DI
Bit[27]	PT13IDF	PT1.3 中断条件标志位	
		When PT13ITT=0	Always 0
		When PT13ITT=1	Inverse DI
		When PT13ITT=2	Same as DI
		When PT13ITT=3	Same as S1
		When PT13ITT=4	Same as DI
		When PT13ITT=5	Inverse DI
		When PT13ITT=6	Same as DI
		When PT13ITT=7	Inverse DI
Bit[26]	PT12IDF	PT1.2 中断条件标志位	
		When PT12ITT=0	Always 0
		When PT12ITT=1	Inverse DI
		When PT12ITT=2	Same as DI
		When PT12ITT=3	Same as S1
		When PT12ITT=4	Same as DI
		When PT12ITT=5	Inverse DI
		When PT12ITT=6	Same as DI
		When PT12ITT=7	Inverse DI
Bit[25]	PT11IDF	PT1.1 中断条件标志位	
		When PT11ITT=0	Always 0
		When PT11ITT=1	Inverse DI
		When PT11ITT=2	Same as DI

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述	
		When PT11ITT=3	Same as S1
		When PT11ITT=4	Same as DI
		When PT11ITT=5	Inverse DI
		When PT11ITT=6	Same as DI
		When PT11ITT=7	Inverse DI
Bit[24]	PT110DF	PT1.0 中断条件标志位	
		When PT10ITT=0	Always 0
		When PT10ITT=1	Inverse DI
		When PT10ITT=2	Same as DI
		When PT10ITT=3	Same as S1
		When PT10ITT=4	Same as DI
		When PT10ITT=5	Inverse DI
		When PT10ITT=6	Same as DI
Bit[23:00]	PT1#ITT	PT10# 选择中断触发方式. #代表的是 0~7	
		000	关闭 GPIO 中断触发, 不能响应中断
		001	上升沿触发
		010	下降沿触发
		011	电位变化触发
		100	低电位触发
		101	高电位触发
		110	低电位触发
		111	高电位触发

14. 通用 GPIO PT2 管理

14.1. 整体总说明

PT2 具有 8 个 IO 引脚，可以作为通用的普通 IO 引脚，亦可复用为 SPI、I²C、UART、PWM、外部振荡器输入及外部中断输入等功能模块的输入或输出 IO 引脚。针对不同的复用，需要做不同的设置。

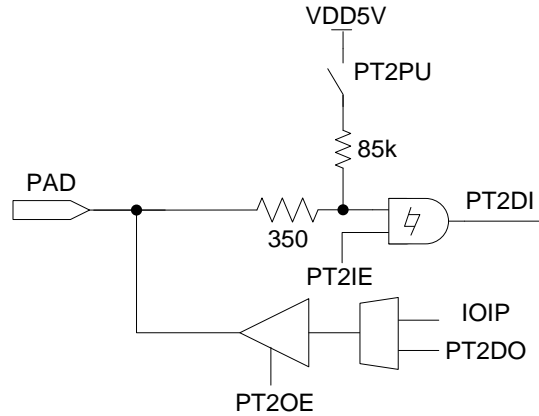


图 14-1 PT2 功能方框图

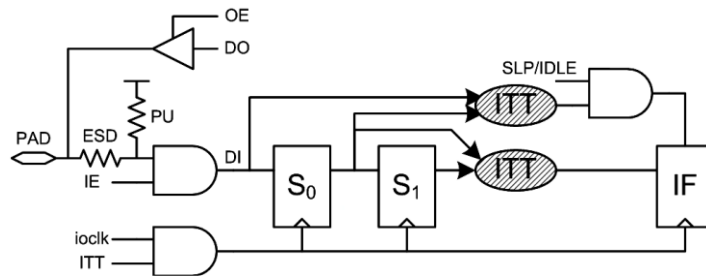


图 14-2 PT2 中断功能方框图

PT2 具有输入、输出、内部上拉电阻及作为外部中断输入引脚的功能，且分别有不同的控制器来设置。

内部上拉电阻

控制器 PT2PU 可设置每个 IO 引脚的内部上拉电阻的开启与关闭，每一位对应一个 IO 引脚。当 IO 引脚对应位被置<1>，则开启内部上拉电阻，若被置<0>，则关闭内部上拉电阻。IO 引脚作为输入模式时，若外部没有上拉电阻，必须开启内部上拉电阻，尤其在低功耗模式时，可以防止漏电，而增加功耗。作为模拟信号输入引脚及外部振荡器输入引脚时，不用开启内部上拉电阻。注意：PT2.4~PT2.7 作为外部振荡器输入引脚时，不可开启内部上拉电阻，否则振荡器不能正常启动振荡。

输出模式

控制器 PT2OE 可设置每个 IO 引脚输出模式的开启与关闭，每一位对应一个 IO 引脚。当 IO 对应位被置<1>，则开启对应 IO 引脚输出模式；若被置<0>，则关闭输出模式。透过控制位 PT2DO 来控制对应 IO 引脚的输出状态为 1 或 0。在低功耗模式下，若 IO 必须开启输出模式，可根据外围电路来设置输出状态，来减低芯片功耗。此模式下不能开启 IO 内部上拉电阻，且不能同时开启输入、输出模式，因而在开启输出模式时，需要关闭 IO 引脚的输入模式。注意：PT2.4~PT2.7 作为外部振荡器输入引脚时，必须关闭输出模式。

输入模式

控制器 PT2IE 可设置每个 IO 引脚输入模式的开启与关闭，每一位对应一个 IO 引脚。当控制器对应位被置

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

<1>，则开启对应 IO 引脚的输入模式；若被置<0>，则关闭输入模式。透过控制器 PT2DI 可读取当前对应 IO 引脚的输入状态为 1 或 0。当 IO 被设置为输入模式，若芯片没有接入外部上拉电阻，则必须开启芯片内部上拉电阻，不能允许 IO 引脚出现浮接状态，以免造成芯片产生漏电现象；特别是在低功耗模式下，建议将 IO 引脚设置为输入模式。作为模拟信号输入引脚时，不用设置对应 IO 引脚为输入模式。在开启输入模式前，需要关闭对应 IO 引脚的输出模式。

外部中断输入

PT2 具有的 8 个 IO 引脚都可复用为外部中断输入引脚。此模式需要将 IO 引脚设置为输入模式且使能内部上拉电阻。需要透过控制器 PT2#ITT 设置外部中断触发沿，并使能控制位 PT2IDF，以使能中断触发沿有效。透过控制器 INTPT2 使能对应 IO 引脚的中断响应功能，当外部中断信号产生时，对应 IO 引脚的中断标志位被置 1。在使能全局中断 GIE 及开启 IO 外部中断功能的条件下，芯片就马上暂停当前程序转去执行 IO 外部中断程序。

14.2. 寄存器地址

GPIO Register Address	31	24	23	16	15	8	7	0
GPIO Base Address + 0x10(0x40810)	MASK1		PT2PU		MASK0		PT2OE	
GPIO Base Address + 0x14 (0x40814)	MASK3		PT2IE		MASK2		PT2DO	
GPIO Base Address + 0x18(0x40818)	-		-		-		PT2DI	
GPIO Base Address + 0x1C (0x4081C)	PT2IDF		PT2#ITT		PT2#ITT		PT2#ITT	

-保留

说明：上列表中#代表 0~7

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

14.3. 寄存器功能

14.3.1. PT2 寄存器 0

GPIO Base Address + 0x10 (0x40810)									
Symbol	PT2CR0 (PT2 Control Register 0)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	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]
名称	MASK	PT2OE7	PT2OE6	PT2OE5	PT2OE4	PT2OE3	PT2OE2	PT2OE1	PT2OE0
RW	R0W-0	RW-0							

位	名称	描述
Bit[23:16]	PT2PU#	PT2 内部上拉控制(PT2PU: PT2 Pull High Enable), #代表的是 7~0
		0 关闭内部上拉
		1 开启内部上拉
Bit[07:00]	PT2OE#	PT2 PAD 输出模式开启控制(PT2OE: PT2 Output Enable), #代表的是 7~0
		0 关闭输出模式
		1 开启输出模式

14.3.2. PT2 寄存器 1

GPIO Base Address + 0x14 (0x40814)									
Symbol	PT2CR1 (PT2 Control Register 1)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	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]
名称	MASK	PT2DO7	PT2DO6	PT2DO5	PT2DO4	PT2DO3	PT2DO2	PT2DO1	PT2DO0
RW	R0W-0	RW-0							

位	名称	描述
Bit[23:16]	PT2IE#	PT2.# PAD 输入模式开启控制(PT2IE: PT2 Input Enable), #代表的是 7~0
		0 关闭输入模式
		1 开启输入模式
Bit[7:00]	PT2DO#	PT2.# PAD 输出状态值(PT2DO: PT2 Output Data), #代表的是 7~0
		0 输出低电位
		1 输出高电位

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



14.3.3. PT2 寄存器 2

GPIO Base Address + 0x18 (0x40818)									
Symbol	PT2CR2 (PT2 Control Register 2)								
Bit	[31:16]								
名称	-								
RW	-								
Bit	[15:8]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	-	PT2DI7	PT2DI6	PT2DI5	PT2DI4	PT2DI3	PT2DI2	PT2DI1	PT2DI0
RW	-	R-0							

位	名称	描述
Bit[7:0]	PT2DI#	PT2.# PAD 输入状态值(PT2DI: PT2 Data Input), #代表的是 7~0
		0 输入低电位
		1 输入高电位

14.3.4. PT2 寄存器 3

GPIO Base Address + 0x1C (0x4081C)						
Symbol	PT2CR3 (PT2 Control Register 3)					
Bit	[31:24]	[23:21]	[21:18]	[17:16]		
名称	PT27IDF~ PT20IDF	PT27ITT	PT26ITT	PT25ITT		
RW	R-0		RW-0			
Bit	[15]	[14:12]	[11:9]	[8:6]	[5:3]	[2:0]
名称	PT25ITT	PT24ITT	PT23ITT	PT22ITT	PT21ITT	PT20ITT
RW	RW-0					

位	名称	描述	
Bit[31]	PT27IDF	PT2.7 中断条件标志位 (举例 :可在进入 Sleep Mode 之前判断该 bit 为 0b 或 1b. 如果为 1b 代表中断条件可成立, 则进入 Sleep Mode 之后, 可以透过 PT2.7 唤醒, 如果为 0b, 则无法透过 PT2.7 唤醒)	
		When PT27ITT=0	Always 0. 说明 : 当 PT27ITT 设定为 000, 则 Bit[31]=0b
		When PT27ITT=1	Inverse DI. : 说明 :进入 Sleep Mode 之前, 当 PT2.7=Low 状态时候, 则此时 Bit[31]=1b
		When PT27ITT=2	Same as DI. 说明: 进入 Sleep Mode 之前, 当 PT2.7=High 状态时候, 则此时 Bit[31]=1b
		When PT27ITT=3	Same as S1. 说明: PT2.7 有电位变化, 即触发产生中断
		When PT27ITT=4	Same as DI. 说明: 进入 Sleep Mode 之前, 当 PT2.7=High 状态时候, 则此时 Bit[31]=1b
		When PT27ITT=5	Inverse DI. 说明: 进入 Sleep Mode 之前, 当 PT2.7=Low 状态时候, 则此时 Bit[31]=1b

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述	
		When PT27ITT=6	Same as DI. 说明: 进入 Sleep Mode 之前, 当 PT2.7=High 状态时候, 则此时 Bit[31]=1b
		When PT27ITT=7	Inverse DI. 说明: 进入 Sleep Mode 之前 当 PT2.7=Low 状态时候, 则此时 Bit[31]=1b
Bit[30]	PT26IDF	PT2.6 中断条件标志位	
		When PT26ITT=0	Always 0
		When PT26ITT=1	Inverse DI
		When PT26ITT=2	Same as DI
		When PT26ITT=3	Same as S1
		When PT26ITT=4	Same as DI
		When PT26ITT=5	Inverse DI
		When PT26ITT=6	Same as DI
		When PT26ITT=7	Inverse DI
Bit[29]	PT25IDF	PT2.5 中断条件标志位	
		When PT25ITT=0	Always 0.
		When PT25ITT=1	Inverse DI.
		When PT25ITT=2	Same as DI.
		When PT25ITT=3	Same as S1.
		When PT25ITT=4	Same as DI.
		When PT25ITT=5	Inverse DI.
		When PT25ITT=6	Same as DI.
		When PT25ITT=7	Inverse DI. 说明:
Bit[28]	PT24IDF	PT2.4 中断条件标志位	
		When PT24ITT=0	Always 0
		When PT24ITT=1	Inverse DI
		When PT24ITT=2	Same as DI
		When PT24ITT=3	Same as S1
		When PT24ITT=4	Same as DI
		When PT24ITT=5	Inverse DI
		When PT24ITT=6	Same as DI
		When PT24ITT=7	Inverse DI
Bit[27]	PT23IDF	PT2.3 中断条件标志位	
		When PT23ITT=0	Always 0
		When PT23ITT=1	Inverse DI
		When PT23ITT=2	Same as DI
		When PT23ITT=3	Same as S1
		When PT23ITT=4	Same as DI
		When PT23ITT=5	Inverse DI
		When PT23ITT=6	Same as DI
		When PT23ITT=7	Inverse DI
Bit[26]	PT22IDF	PT2.2 中断条件标志位	
		When PT22ITT=0	Always 0
		When PT22ITT=1	Inverse DI
		When PT22ITT=2	Same as DI
		When PT22ITT=3	Same as S1
		When PT22ITT=4	Same as DI
		When PT22ITT=5	Inverse DI
		When PT22ITT=6	Same as DI
		When PT22ITT=7	Inverse DI
Bit[25]	PT21IDF	PT2.1 中断条件标志位	

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述
		When PT21ITT=0 Always 0
		When PT21ITT=1 Inverse DI
		When PT21ITT=2 Same as DI
		When PT21ITT=3 Same as S1
		When PT21ITT=4 Same as DI
		When PT21ITT=5 Inverse DI
		When PT21ITT=6 Same as DI
		When PT21ITT=7 Inverse DI
Bit[24]	PT20DF	PT2.0 中断条件标志位
		When PT20ITT=0 Always 0
		When PT20ITT=1 Inverse DI
		When PT20ITT=2 Same as DI
		When PT20ITT=3 Same as S1
		When PT20ITT=4 Same as DI
		When PT20ITT=5 Inverse DI
		When PT20ITT=6 Same as DI
When PT20ITT=7 Inverse DI		
Bit[23:00]	PT2#ITT	PT2.# 选择中断触发方式. #代表的是 0~7
		000 关闭 GPIO 中断触发, 不能响应中断
		001 上升沿触发
		010 下降沿触发
		011 电位变化触发
		100 低电位触发
		101 高电位触发
		110 低电位触发
		111 高电位触发

15. 通用 GPIO PT3 管理

15.1. 整体总说明

PT3 具有 8 个 IO 引脚，可以作为通用的普通 IO 引脚，亦可复用为低电压比较器输入引脚、ICE 接口通信引脚(ECK、EDIO)，以及 ADC 转换器模块的输入或输出 IO 引脚。针对不同的复用，需要做不同的设置。

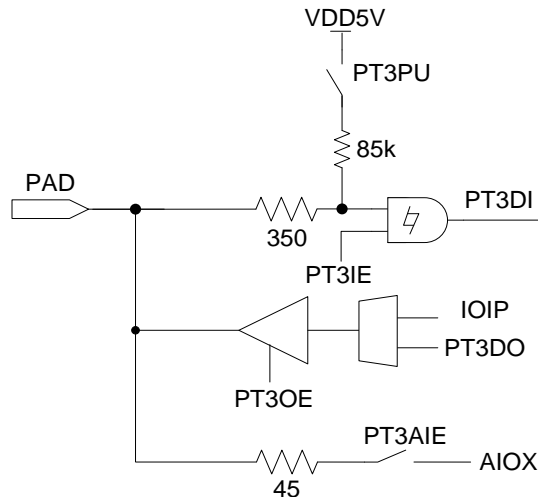


图 15-1 PT3 功能方框图

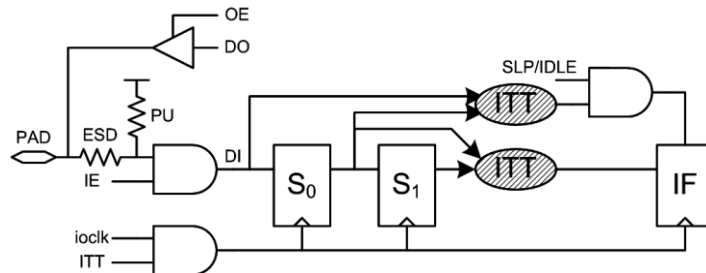


图 15-2 PT3 中断功能方框图

PT3 具有输入、输出、内部上拉电阻的功能，且分别有不同的控制器来设置。PT3.0、PT3.1 带有 ICE 通信接口功能(ECK、EDIO)，请勿将该脚位设计为大电流输出控制脚位，避免影响 ICE 接口功能。当芯片上电时，在 BOR1/BOR2 触发时间之内，ICE 接口功能才会生效；在完成上电时间之内，若未透过特殊命令触发 ICE 通信时，当上电时间完成后，择 ICE 接口将会关闭。

内部上拉电阻

控制器 PT3PU 可设置每个 IO 引脚的内部上拉电阻的开启与关闭，每一位对应一个 IO 引脚。当 IO 对应位被置<1>，则开启内部上拉电阻，若被置<0>，则关闭内部上拉电阻。IO 引脚作为输入模式时，若外部没有上拉电阻，必须开启内部上拉电阻，尤其在低功耗模式时，可以防止漏电，而增加功耗。作为模拟信号输入引脚时，不用开启内部上拉电阻。

数字输出模式

控制器 PT3OE 可设置每个 IO 引脚输出模式的开启与关闭，每一位对应一个 IO 引脚。当 IO 引脚对应位被置<1>，则开启对应 IO 引脚输出模式；若被置<0>，则关闭输出模式。透过控制位 PT3DO 来控制对应 IO 引脚的输出状态为 1 或 0。在低功耗模式下，若 IO 必须开启输出模式，可根据外围电路来设置输出状态，来减低芯片功耗。

数字输入模式

控制器 PT3IE 可设置每个 IO 引脚输入模式的开启与关闭，每一位对应一个 IO 引脚。当控制器对应位被置 <1>，则开启对应 IO 引脚的数字输入模式；若被置 <0>，则关闭数字输入模式。透过控制器 PT3DI 可读取当前对应 IO 引脚的输入状态为 1 或 0。当 IO 被设置为数字输入模式，若芯片没有接入外部上拉电阻，则必须开启芯片内部上拉电阻，不能允许 IO 引脚出现浮接状态，以免造成芯片产生漏电现象；特别是在低功耗模式下，建议将 IO 引脚设置为数字输入模式。在开启数字输入模式前，需要关闭对应 IO 引脚的输出模式。

模拟输入输出模式

控制器 PT3AIE 可设置每个 IO 引脚模拟输入输出模式的开启与关闭，每一位对应一个 IO 引脚。当控制器对应位被置 <1>，则开启对应 IO 引脚的模拟模式；若被置 <0>，则关闭模拟模式。当 IO 被设置为模拟模式时，禁止启动内部上拉电阻开关，也不允许启动数字输入或是数字输出功能，以免造成芯片产生漏电现象影响模拟功能测试；

外部中断输入

PT3 具有的 8 个 IO 引脚都可复用为外部中断输入引脚。此模式需要将 IO 引脚设置为输入模式且使能内部上拉电阻。需要透过控制器 PT3#ITT 设置外部中断触发沿，并使能控制位 PT3IDF，以使能中断触发沿有效。透过控制器 INTPT3 使能对应 IO 引脚的中断响应功能，当外部中断信号产生时，对应 IO 引脚的中断标志位被置 1。在使能全局中断 GIE 及开启 IO 外部中断功能的条件下，芯片就马上暂停当前程序转去执行 IO 外部中断程序。

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



15.2. 寄存器地址

GPIO Register Address	31	24	23	16	15	8	7	0
GPIO Base Address + 0x20(0x40820)	MASK1		PT3PU		MASK0		PT3OE	
GPIO Base Address + 0x24(0x40824)	MASK3		PT3IE		MASK2		PT3DO	
GPIO Base Address + 0x28(0x40828)	MASK5		PT3AIE		MASK4		PT3DI	
GPIO Base Address + 0x2C(0x4082C)	PT3IDF		PT3#ITT		PT3#ITT		PT3#ITT	

-Reserved

15.3. 寄存器功能

15.3.1. PT3 寄存器 0

GPIO Base Address + 0x20 (0x40820)									
Symbol	PT3CR0 (PT3 Control Register 0)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	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]
名称	MASK	PT3OE7	PT3OE6	PT3OE5	PT3OE4	PT3OE3	PT3OE2	PT3OE1	PT3OE0
RW	R0W-0	RW-0							

位	名称	描述
Bit[23:16]	PT3PU#	PT3.# 内部上拉开启控制(PT3PU: PT3 Pull High Enable), #代表的是 7~0
		0 关闭内部上拉
		1 开启内部上拉
Bit[07:00]	PT3OE#	PT3.# PAD 输出模式开启控制(PT3OE: PT3 Output Enable), #代表的是 7~0
		0 关闭输出模式, 启动模拟输入功能。
		1 开启输出模式

15.3.2. PT3 寄存器 1

GPIO Base Address + 0x24 (0x40824)									
Symbol	PT3CR1 (PT3 Control Register 1)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	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]
名称	MASK	PT3DO7	PT3DO6	PT3DO5	PT3DO4	PT3DO3	PT3DO2	PT3DO1	PT3DO0
RW	R0W-0	RW-0							

位	名称	描述
Bit[23:16]	PT3IE#	PT3.# PAD 输入模式开启控制(PT3IE: PT3 Input Enable), #代表的是 7~0
		0 关闭数字输入模式

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述	
		1	开启数字输入模式
Bit[07:00]	PT3DO#	PT3.# PAD 输出状态值(PT3DO: PT3 Output Data), #代表的是 7~0	
		0	输出低电位
		1	输出高电位

15.3.3. PT3 寄存器 2

GPIO Base Address + 0x28 (0x40828)											
Symbol	PT3CR2 (PT3 Control Register 2)										
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]		
名称	MASK	PT3AIE7	PT3AIE6	PT3AIE5	PT3AIE4	PT3AIE3	PT3AIE2	PT3AIE1	PT3AIE0		
RW	R0W-0	R-0									
Bit	[15:09]	[8]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]	
名称	-	PT310OEOK	PT3DI[7]	PT3DI[6]	PT3DI[5]	PT3DI[4]	PT3DI[3]	PT3DI[2]	PT3DI[1]	PT3DI[0]	
RW	-	R-0	R-0								

位	名称	描述
Bit[23:16]	PT3AIE	Port 3 PAD 模拟输入输出模式开启控制
		0 关闭模拟模式
		1 开启模拟模式
Bit[8]	PT310OEOK	PT3.0/PT3.1 可控制状态标志位
		0 CPU 还无法控制 PT3.0/3.1
Bit[7:0]	PT3DI#	PT3.0/PT3.1 可以设计为数字输入输出使用. Note: 1.上电后 250ms 之后, SMBus, EDM 界面关闭;此时状态标志位转为 1.之后 PT3.0~3.1 才可以让 CPU 控制! 2.在 Debug mode 下, PT3.0/PT3.1 让 ICP 接口占用,因此该状态标志位为会恒为 0.
		0 输入低电位
		1 输入高电位

15.3.4. PT3 寄存器 3

GPIO Base Address + 0x1C (0x4082C)						
Symbol	PT3CR3 (PT3 Control Register 3)					
Bit	[31:24]	[23:21]	[20:18]	[17:16]		
名称	PT37IDF~ PT30IDF	PT37ITT	PT36ITT	PT35ITT		
RW	RW-0					
Bit	[15]	[14:12]	[11:9]	[8:6]	[5:3]	[2:0]
名称	PT35ITT	PT34ITT	PT33ITT	PT32ITT	PT31ITT	PT30ITT
RW	RW-0					

位	名称	描述
---	----	----

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述
Bit[31]	PT37IDF	PT3.7 中断条件标志位 (举例 :可在进入 Sleep Mode 之前判断该 bit 为 0b 或 1b. 如果为 1b 代表中断条件可成立, 则进入 Sleep Mode 之后, 可以透过 PT3.7 唤醒, 如果为 0b, 则无法透过 PT3.7 唤醒)
		When PT37ITT=0 Always 0. 说明 : 当 PT37ITT 设定为 000, 则 Bit[31]=0b
		When PT37ITT=1 Inverse DI. : 说明 :进入 Sleep Mode 之前, 当 PT3.7=Low 状态时候, 则此时 Bit[31]=1b
		When PT37ITT=2 Same as DI. 说明: 进入 Sleep Mode 之前, 当 PT3.7=High 状态时候, 则此时 Bit[31]=1b
		When PT37ITT=3 Same as S1. 说明: PT3.7 有电位变化, 即触发产生中断
		When PT37ITT=4 Same as DI. 说明: 进入 Sleep Mode 之前, 当 PT3.7=High 状态时候, 则此时 Bit[31]=1b
		When PT37ITT=5 Inverse DI. 说明: 进入 Sleep Mode 之前, 当 PT3.7=Low 状态时候, 则此时 Bit[31]=1b
		When PT37ITT=6 Same as DI. 说明: 进入 Sleep Mode 之前, 当 PT3.7=High 状态时候, 则此时 Bit[31]=1b
		When PT37ITT=7 Inverse DI. 说明: 进入 Sleep Mode 之前 当 PT3.7=Low 状态时候, 则此时 Bit[31]=1b
Bit[30]	PT36IDF	PT3.6 中断条件标志位
		When PT36ITT=0 Always 0
		When PT36ITT=1 Inverse DI
		When PT36ITT=2 Same as DI
		When PT36ITT=3 Same as S1
		When PT36ITT=4 Same as DI
		When PT36ITT=5 Inverse DI
		When PT36ITT=6 Same as DI
		When PT36ITT=7 Inverse DI
Bit[29]	PT35IDF	PT3.5 中断条件标志位
		When PT35ITT=0 Always 0
		When PT35ITT=1 Inverse DI
		When PT35ITT=2 Same as DI
		When PT35ITT=3 Same as S1
		When PT35ITT=4 Same as DI
		When PT35ITT=5 Inverse DI
		When PT35ITT=6 Same as DI
		When PT35ITT=7 Inverse DI
Bit[28]	PT34IDF	PT3.4 中断条件标志位
		When PT34ITT=0 Always 0
		When PT34ITT=1 Inverse DI
		When PT34ITT=2 Same as DI
		When PT34ITT=3 Same as S1
		When PT34ITT=4 Same as DI
		When PT34ITT=5 Inverse DI
		When PT34ITT=6 Same as DI
		When PT34ITT=7 Inverse DI
Bit[27]	PT33IDF	PT3.3 中断条件标志位

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述			
		When PT33ITT=0	Always 0		
		When PT33ITT=1	Inverse DI		
		When PT33ITT=2	Same as DI		
		When PT33ITT=3	Same as S1		
		When PT33ITT=4	Same as DI		
		When PT33ITT=5	Inverse DI		
		When PT33ITT=6	Same as DI		
		When PT33ITT=7	Inverse DI		
Bit[26]	PT32IDF	PT3.2 中断条件标志位			
		When PT32ITT=0	Always 0		
		When PT32ITT=1	Inverse DI		
		When PT32ITT=2	Same as DI		
		When PT32ITT=3	Same as S1		
		When PT32ITT=4	Same as DI		
		When PT32ITT=5	Inverse DI		
		When PT32ITT=6	Same as DI		
Bit[25]	PT31IDF	PT3.1 中断条件标志位			
		When PT31ITT=0	Always 0		
		When PT31ITT=1	Inverse DI		
		When PT31ITT=2	Same as DI		
		When PT31ITT=3	Same as S1		
		When PT31ITT=4	Same as DI		
		When PT31ITT=5	Inverse DI		
		When PT31ITT=6	Same as DI		
Bit[24]	PT30IDF	PT3.0 中断条件标志位			
		When PT30ITT=0	Always 0		
		When PT30ITT=1	Inverse DI		
		When PT30ITT=2	Same as DI		
		When PT30ITT=3	Same as S1		
		When PT30ITT=4	Same as DI		
		When PT30ITT=5	Inverse DI		
		When PT30ITT=6	Same as DI		
Bit[23:21]	PT37ITT	PT3.7 选择中断触发方式			
		000	关闭 GPIO 中断触发，不能响应中断		
		001	上升沿触发	101	高电位触发
		010	下降沿触发	110	低电位触发
		011	电位变化触发	111	高电位触发
		100	低电位触发		
Bit[20:18]	PT36ITT	PT3.6 选择中断触发方式			
		000	关闭 GPIO 中断触发，不能响应中断		
		001	上升沿触发	101	高电位触发
		010	下降沿触发	110	低电位触发
		011	电位变化触发	111	高电位触发
		100	低电位触发		
Bit[17:15]	PT35ITT	PT3.5 选择中断触发方式			

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述			
		000	关闭 GPIO 中断触发，不能响应中断		
		001	上升沿触发	101	高电位触发
		010	下降沿触发	110	低电位触发
		011	电位变化触发	111	高电位触发
		100	低电位触发		
Bit[14:12]	PT34ITT	PT3.4 选择中断触发方式			
		000	关闭 GPIO 中断触发，不能响应中断		
		001	上升沿触发	101	高电位触发
		010	下降沿触发	110	低电位触发
		011	电位变化触发	111	高电位触发
		100	低电位触发		
Bit[11:09]	PT33ITT	PT3.3 选择中断触发方式			
		000	关闭 GPIO 中断触发，不能响应中断		
		001	上升沿触发	101	高电位触发
		010	下降沿触发	110	低电位触发
		011	电位变化触发	111	高电位触发
		100	低电位触发		
Bit[08:06]	PT32ITT	PT3.2 选择中断触发方式			
		000	关闭 GPIO 中断触发，不能响应中断		
		001	上升沿触发	001	上升沿触发
		010	下降沿触发	010	下降沿触发
		011	电位变化触发	011	电位变化触发
		100	低电位触发	100	低电位触发
Bit[05:03]	PT31ITT	PT3.1 选择中断触发方式			
		000	关闭 GPIO 中断触发，不能响应中断		
		001	上升沿触发	101	高电位触发
		010	下降沿触发	110	低电位触发
		011	电位变化触发	111	高电位触发
		100	低电位触发		
Bit[02:00]	PT30ITT	PT3.0 选择中断触发方式			
		000	关闭 GPIO 中断触发，不能响应中断		
		001	上升沿触发	101	高电位触发
		010	下降沿触发	110	低电位触发
		011	电位变化触发	111	高电位触发
		100	低电位触发		

15.4. 模拟数字复用功能切换注意事项

PT3.2~PT3.7 除了可以当一般数字功能使用，也可以设置做为模拟复用功能，而在做模拟数字复用功能切换时候，应该注意到相关的寄存器设定，以避免影响到该引脚的正常功能使用。整体而言 PT3OE 决定是数字输出选择，PT3AIE 决定模拟输入输出引脚选择，PT3PU 决定是否启动内部上升电阻，PT3IE 决定是否启动数字输入引脚功能。以下针对 PT3 设定进行说明：

PT3.6/REFO 复用引脚:

- 设计为模拟脚 REFO 电压输出：
控制寄存器 ENRFO 0x40400[1]=1b, PT3PU6=PT3OE6=PT3IE6=0b, PT3AIE6=1b
- 设计为模拟脚 REFO 电压由外部输入：
控制寄存器 ENRFO 0x40400[1]=0b, PT3PU6=PT3OE6=PT3IE6=0b, PT3AIE6=1b
- 设计为 PT3.6 数字脚输入与输出功能：
控制寄存器 ENRFO 0x40400[1]=0b, PT3OE6=1b, PT3IE6=1b, PT3AIE6=0b
- 设计为 PT3.6 数字脚输入功能：
控制寄存器 ENRFO 0x40400[1]=0b, PT3OE6=0b, PT3IE6=1b, PT3AIE6=0b
- 设计为 PT3.6 数字脚输出功能：
控制寄存器 ENRFO 0x40400[1]=0b, PT3OE6=1b, PT3IE6=0b, PT3AIE6=0b

其它 GPIO 的使用方式: (PT3.2~PT3.5, PT3.7 皆同下说明)

举例 PT3.5/AIO7 复用引脚:

- 设计为 AIO7 模拟输入功能：PT3PU5=PT3OE5=PT3IE5=0b, PT3AIE5=1b
- 设计为 PT3.5 数字输入与输出功能：PT3IE5=1b, PT3PU5=0b, PT3OE5=1b, PT3AIE5=0b
- 设计为 PT3.5 数字输出功能：PT3IE5=0b, PT3PU5=0b, PT3OE5=1b, PT3AIE5=0b
- 设计为 PT3.5 数字输入功能：PT3IE5=1b, PT3PU5=1b, PT3OE5=0b, PT3AIE5=0b. (PT3PU5=1b 的设定是让输入端不浮接)

16. 通用 GPIO PT6 管理

16.1. 整体总说明

PT6 具有 8 个 IO 引脚，可以作为通用的普通 IO 引脚，PWM 与通信引脚外，亦可复用为 LCD 功能输出引脚。针对不同的复用，需要做不同的设置。IO 引脚不支持内建上拉电阻功能，也不支持中断输入功能。

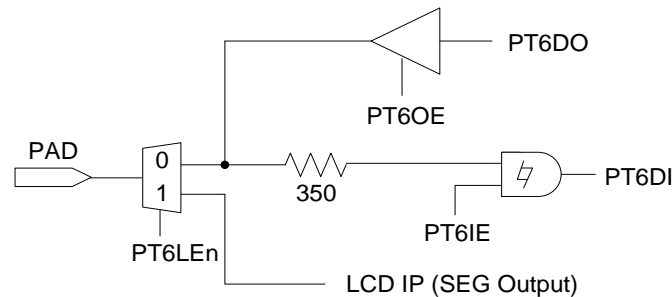


图 16-1 PT6 功能方框图

PT6 具有输入、输出的功能，且分别有不同的控制器来设置。

输出模式

控制器 PT6xOE 可设置每个 IO 引脚输出模式的开启与关闭，每一位对应一个 IO 引脚。当 IO 对应位被置 <1>，则开启对应 IO 输出模式；若被置 <0>，则关闭输出模式。

透过控制位 PT6xDO 来控制对应 IO 引脚的输出状态为 1 或 0。在低功耗模式下，若 IO 必须开启输出模式，可根据外围电路来设置输出状态，来减低芯片功耗。此模式下不能同时开启输入、输出模式，因而在开启输出模式时，需要关闭 IO 的输入模式。

说明：上述 x 代表的是 0~7，对应到 PT6.0~PT6.7。

输入模式

控制器 PT6xIE 可设置每个 IO 引脚输入模式的开启与关闭，每一位对应一个 IO 引脚。当控制器对应位被置 <1>，则开启对应 IO 引脚的输入模式；若被置 <0>，则关闭输入模式。透过控制器 PT6xDI 可读取当前对应 IO 引脚的输入状态为 1 或 0。当 IO 被设置为输入模式，需要接入外部上拉电阻，不能允许 IO 引脚出现浮接状态，以免造成芯片产生漏电现象；特别是在低功耗模式下，建议将 IO 引脚设置为输入模式。在开启输入模式前，需要关闭对应 IO 引脚的输出模式。

说明：上述 x 代表的是 0~7，对应到 PT6.0~PT6.7。

LCD 模式

控制器 SEGx[7:0] 决定 LCD SEGMENT 输出数据，

若 LCD 为 1/8 Duty 模式，则 SEGx[7:0] 则决定 1/8 Duty 数据内容；

若 LCD 为 1/7 Duty 模式，则 SEGx[6:0] 则决定 1/7 Duty 数据内容；

若 LCD 为 1/6 Duty 模式，则 SEGx[5:0] 则决定 1/6 Duty 数据内容；

若 LCD 为 1/5 Duty 模式，则 SEGx[4:0] 则决定 1/5 Duty 数据内容；

若 LCD 为 1/4 Duty 模式，则 SEGx[3:0] 则决定 1/4 Duty 数据内容；

若 LCD 为 1/3 Duty 模式，则 SEGx[2:0] 则决定 1/3 Duty 数据内容；

说明：上述 x 代表的是 2~9，对应到 SEG2~SEG9。

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



16.2. 寄存器地址

GPIO Mode Register Address	31	24	23	16	15	8	7	0
GPIO Base Address + 0x50(0x40850)	MASK1		PT61CFG		MASK0		PT60CFG	
GPIO Base Address + 0x54(0x40854)	MASK3		PT63CFG		MASK2		PT62CFG	
GPIO Base Address + 0x58(0x40858)	MASK5		PT65CFG		MASK4		PT64CFG	
GPIO Base Address + 0x5C(0x4085C)	MASK7		PT67CFG		MASK6		PT66CFG	

LCD Mode Register Address	31	24	23	16	15	8	7	0
GPIO Base Address + 0x50(0x40850)	MASK1		SEG3		MASK0		SEG2	
GPIO Base Address + 0x54(0x40854)	MASK3		SEG5		MASK2		SEG4	
GPIO Base Address + 0x58(0x40858)	MASK5		SEG7		MASK4		SEG6	
GPIO Base Address + 0x5C(0x4085C)	MASK7		SEG9		MASK6		SEG8	

LCD Register Address 0x41B04 可以决定设定为 GPIO Mode 或是 LCD Mode.

16.3. 寄存器功能

16.3.1. PT6 寄存器 0

When GPIO Mode.

GPIO Base Address + 0x50 (0x40850)									
Symbol	PT60CFG/ PT61CFG (PT6 Control Register 0)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT61OE	PT61IE	PT61DO	PT61DI
RW	R0W-0	RW-0						RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT60OE	PT60IE	PT60DO	PT60DI
RW	R0W-0	RW-0						RW-1	

位	名称	描述
Bit[19]	PT61OE	PT6.1 Output Enable
		0 关闭
		1 开启
Bit[18]	PT61IE	PT6.1 Input Enable
		0 关闭
		1 开启
Bit[17]	PT61DO	PT6.1 Output Data
		0 Output Low
		1 Output High
Bit[16]	PT61DI	PT6.1 Input Data
		0 Input Low
		1 Input High
Bit[03]	PT60OE	PT6.0 Output Enable
		0 关闭
		1 开启
Bit[02]	PT60IE	PT6.0 Input Enable

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述	
		0	关闭
		1	开启
Bit[01]	PT60DO	PT6.0 Output Data	
		0	Output Low
		1	Output High
Bit[00]	PT60DI	PT6.0 Input Data	
		0	Input Low
		1	Input High

When LCD Mode

GPIO Base Address + 0x50 (0x40850)									
Symbol	SEG2/SEG3 (PT6 Control Register 0)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	SEG3 Data							
RW	R0W-0	RW-0							RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	SEG2 Data							
RW	R0W-0	RW-0							RW-1

位	名称	描述
Bit[23:16]	SEG 3 Data	LCD Segment 3 Data
		Segment Data
Bit[07:00]	SEG 2 Data	LCD Segment 2 Data
		Segment Data

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



16.3.2. PT6 寄存器 1

When GPIO Mode.

GPIO Base Address + 0x54 (0x40854)									
Symbol	PT62CFG/ PT63CFG (PT6 Control Register 1)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT63OE	PT63IE	PT63DO	PT63DI
RW	R0W-0	RW-0						RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT62OE	PT62IE	PT62DO	PT62DI
RW	R0W-0	RW-0						RW-1	

位	名称	描述	
Bit[19]	PT63OE	PT6.3 Output Enable	
		0	关闭
		1	开启
Bit[18]	PT63IE	PT6.3 Input Enable	
		0	关闭
		1	开启
Bit[17]	PT63DO	PT6.3 Output Data	
		0	Output Low
		1	Output High
Bit[16]	PT63DI	PT6.3 Input Data	
		0	Input Low
		1	Input High
Bit[03]	PT62OE	PT6.2 Output Enable	
		0	关闭
		1	开启
Bit[02]	PT62IE	PT6.2 Input Enable	
		0	关闭
		1	开启
Bit[01]	PT62DO	PT6.2 Output Data	
		0	Output Low
		1	Output High
Bit[00]	PT62DI	PT6.2 Input Data	
		0	Input Low
		1	Input High

When LCD Mode.

GPIO Base Address + 0x54 (0x40854)									
Symbol	SEG4/SEG5 (PT6 Control Register 1)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	SEG5 Data							
RW	R0W-0	RW-0						RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	SEG4 Data							
RW	R0W-0	RW-0						RW-1	

位	名称	描述
Bit[23:16]	SEG 5 Data	LCD Segment 5 Data

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



		Segment Data
Bit[07:00]	SEG 4 Data	LCD Segment 4 Data
		Segment Data

16.3.3. PT6 寄存器 2

When GPIO Mode

GPIO Base Address + 0x58 (0x40858)									
Symbol	PT64CFG/ PT65CFG (PT6 Control Register 2)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT65OE	PT65IE	PT65DO	PT65DI
RW	R0W-0	RW-0						RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT64OE	PT64IE	PT64DO	PT64DI
RW	R0W-0	RW-0						RW-1	

位	名称	描述	
Bit[19]	PT65OE	PT6.5 Output Enable	
		0	关闭
		1	开启
Bit[18]	PT65IE	PT6.5 Input Enable	
		0	关闭
		1	开启
Bit[17]	PT65DO	PT6.5 Output Data	
		0	Output Low
		1	Output High
Bit[16]	PT65DI	PT6.5 Input Data	
		0	Input Low
		1	Input High
Bit[03]	PT64OE	PT6.4 Output Enable	
		0	关闭
		1	开启
Bit[02]	PT64IE	PT6.4 Input Enable	
		0	关闭
		1	开启
Bit[01]	PT64DO	PT6.4 Output Data	
		0	Output Low
		1	Output High
Bit[00]	PT64DI	PT6.4 Input Data	
		0	Input Low
		1	Input High

When LCD Mode

GPIO Base Address + 0x58 (0x40858)									
Symbol	SEG6/SEG7 (PT6 Control Register 2)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	SEG7 Data							
RW	R0W-0	RW-0						RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	SEG6 Data							

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



RW	R0W-0	RW-0	RW-1
----	-------	------	------

位	名称	描述
Bit[23:16]	SEG 7 Data	LCD Segment 7 Data Segment Data
Bit[07:00]	SEG 6 Data	LCD Segment 6 Data Segment Data

16.3.4. PT6 寄存器 3

When GPIO Mode

GPIO Base Address + 0x5C (0x4085C)									
Symbol	PT66CFG/ PT67CFG (PT6 Control Register 3)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT67OE	PT67IE	PT67DO	PT67DI
RW	R0W-0	RW-0						RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT66OE	PT66IE	PT66DO	PT66DI
RW	R0W-0	RW-0						RW-1	

位	名称	描述
Bit[19]	PT67OE	PT6.7 Output Enable
		0 关闭
		1 开启
Bit[18]	PT67IE	PT6.7 Input Enable
		0 关闭
		1 开启
Bit[17]	PT67DO	PT6.7 Output Data
		0 Output Low
		1 Output High
Bit[16]	PT67DI	PT6.7 Input Data
		0 Input Low
		1 Input High
Bit[03]	PT66OE	PT6.6 Output Enable
		0 关闭
		1 开启
Bit[02]	PT66IE	PT6.6 Input Enable
		0 关闭
		1 开启
Bit[01]	PT66DO	PT6.6 Output Data
		0 Output Low
		1 Output High
Bit[00]	PT66DI	PT6.6 Input Data
		0 Input Low
		1 Input High

When LCD Mode

GPIO Base Address + 0x5C (0x4085C)									
Symbol	SEG8/SEG9 (PT6 Control Register 3)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

名称	MASK	SEG9 Data								
RW	R0W-0	RW-0								RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]	
名称	MASK	SEG8 Data								
RW	R0W-0	RW-0								RW-1

位	名称	描述
Bit[23:16]	SEG 9 Data	LCD Segment 9 Data
		Segment Data
Bit[07:00]	SEG 8 Data	LCD Segment 8 Data
		Segment Data

17. 通用 GPIO PT7 管理

17.1. 整体总说明

PT7 具有 8 个 IO 引脚，可以作为通用的普通 IO 引脚 PWM 与通信脚位外，亦可复用为 LCD 功能输出引脚。针对不同的复用，需要做不同的设置。IO 引脚不支持内建上拉电阻功能，也不支持中断输入功能。

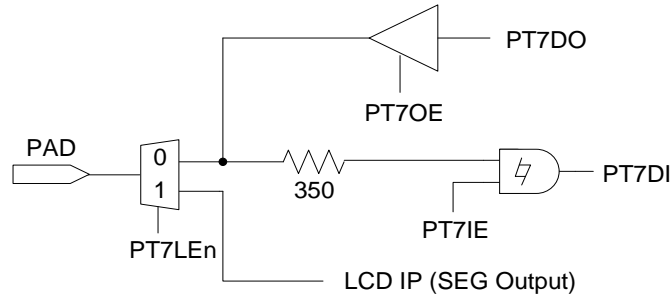


图 17-1 PT7 功能方框图

PT7 具有输入、输出的功能，且分别有不同的控制器来设置。

输出模式

控制器 PT7xOE 可设置每个 IO 输出模式的开启与关闭，每一位对应一个 IO 引脚。当 IO 对应位被置<1>，则开启对应 IO 输出模式；若被置<0>，则关闭输出模式。透过控制位 PT7xDO 来控制对应 IO 引脚的输出状态为 1 或 0。在低功耗模式下，若 IO 必须开启输出模式，可根据外围电路来设置输出状态，来减低芯片功耗。此模式下不能同时开启输入、输出模式，因而在开启输出模式时，需要关闭 IO 的输入模式。

说明：上述 x 代表的是 0~7，对应到 PT7.0~PT7.7。

输入模式

控制器 PT7xIE 可设置每个 IO 引脚输入模式的开启与关闭，每一位对应一个 IO 引脚。当控制器对应位被置<1>，则开启对应 IO 引脚的输入模式；若被置<0>，则关闭输入模式。透过控制器 PT7xDI 可读取当前对应 IO 引脚的输入状态为 1 或 0。当 IO 被设置为输入模式，需要接入外部上拉电阻，不能允许 IO 引脚出现浮接状态，以免造成芯片产生漏电现象；特别是在低功耗模式下，建议将 IO 引脚设置为输入模式。在开启输入模式前，需要关闭对应 IO 引脚的输出模式。

说明：上述 x 代表的是 0~7，对应到 PT7.0~PT7.7。

LCD 模式

控制器 SEGx[7:0]决定 LCD SEGMENT 输出数据，

若 LCD 为 1/8 Duty 模式，则 SEGx[7:0]则决定 1/8 Duty 数据内容；

若 LCD 为 1/7 Duty 模式，则 SEGx[6:0]则决定 1/7 Duty 数据内容；

若 LCD 为 1/6 Duty 模式，则 SEGx[5:0]则决定 1/6 Duty 数据内容；

若 LCD 为 1/5 Duty 模式，则 SEGx[4:0]则决定 1/5 Duty 数据内容；

若 LCD 为 1/4 Duty 模式，则 SEGx[3:0]则决定 1/4 Duty 数据内容；

若 LCD 为 1/3 Duty 模式，则 SEGx[2:0]则决定 1/3 Duty 数据内容；

说明：上述 x 代表的是 10~17，对应到 SEG10~SEG17。

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



17.2. 寄存器地址

GPIO Mode Register Address	31	24	23	16	15	8	7	0
GPIO Base Address + 0x60(0x40860)	MASK1			PT71CFG	MASK0		PT70CFG	
GPIO Base Address + 0x64(0x40864)	MASK3			PT73CFG	MASK2		PT72CFG	
GPIO Base Address + 0x68(0x40868)	MASK5			PT75CFG	MASK4		PT74CFG	
GPIO Base Address + 0x6C(0x4086C)	MASK7			PT77CFG	MASK6		PT76CFG	

LCD Mode Register Address	31	24	23	16	15	8	7	0
GPIO Base Address + 0x60(0x40860)	MASK1			SEG11	MASK0		SEG10	
GPIO Base Address + 0x64(0x40864)	MASK3			SEG13	MASK2		SEG12	
GPIO Base Address + 0x68(0x40868)	MASK5			SEG15	MASK4		SEG14	
GPIO Base Address + 0x6C(0x4086C)	MASK7			SEG17	MASK6		SEG16	

LCD Register Address 0x41B04 可以决定设定为 GPIO Mode 或是 LCD Mode.

17.3. 寄存器功能

17.3.1. PT7 寄存器 0

When GPIO Mode

GPIO Base Address + 0x60 (0x40860)									
PT70CFG/ PT71CFG (PT7 Control Register 0)									
Symbol	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT71OE	PT71IE	PT71DO	PT71DI
RW	R0W-0	RW-0						RW-1	
Symbol	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT70OE	PT70IE	PT70DO	PT70DI
RW	R0W-0	RW-0						RW-1	

位	名称	描述
Bit[19]	PT71OE	PT7.1 Output Enable
		0 关闭
		1 开启
Bit[18]	PT71IE	PT7.1 Input Enable
		0 关闭
		1 开启
Bit[17]	PT71DO	PT7.1 Output Data
		0 Output Low
		1 Output High
Bit[16]	PT71DI	PT7.1 Input Data
		0 Input Low
		1 Input High
Bit[03]	PT70OE	PT7.0 Output Enable
		0 关闭
		1 开启
Bit[02]	PT70IE	PT7.0 Input Enable
		0 关闭
		1 开启

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

		1	开启
Bit[01]	PT70DO	PT7.0 Output Data	
		0	Output Low
		1	Output High
Bit[00]	PT70DI	PT7.0 Input Data	
		0	Input Low
		1	Input High

When LCD Mode

GPIO Base Address + 0x60 (0x40860)									
Symbol	SEG10/SEG11 (PT7 Control Register 0)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	SEG11 Data							
RW	R0W-0	RW-0							RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	SEG10 Data							
RW	R0W-0	RW-0							RW-1

位	名称	描述
Bit[23:16]	SEG 11 Data	LCD Segment 11 Data Segment Data
Bit[7:0]	SEG 10 Data	LCD Segment 10 Data Segment Data

17.3.2. PT7 寄存器 1

When GPIO Mode

GPIO Base Address + 0x64 (0x40864)									
Symbol	PT72CFG/ PT73CFG (PT7 Control Register 1)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT73OE	PT73IE	PT73DO	PT73DI
RW	R0W-0	RW-0							RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT72OE	PT72IE	PT72DO	PT72DI
RW	R0W-0	RW-0							RW-1

位	名称	描述	
Bit[19]	PT73OE	PT7.3 Output Enable	
		0	关闭
		1	开启
Bit[18]	PT73IE	PT7.3 Input Enable	
		0	关闭
		1	开启
Bit[17]	PT73DO	PT7.3 Output Data	
		0	Output Low
		1	Output High
Bit[16]	PT73DI	PT7.3 Input Data	
		0	Input Low
		1	Input High
Bit[03]	PT72OE	PT7.2 Output Enable	
		0	关闭

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述	
		1	开启
Bit[02]	PT72IE	PT7.2 Input Enable	
		0	关闭
		1	开启
Bit[01]	PT72DO	PT7.2 Output Data	
		0	Output Low
		1	Output High
Bit[00]	PT72DI	PT7.2 Input Data	
		0	Input Low
		1	Input High

When LCD Mode

GPIO Base Address + 0x64 (0x40864)									
Symbol	SEG12/SEG13 (PT7 Control Register 1)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	SEG13 Data							
RW	R0W-0	RW-0							RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	SEG12 Data							
RW	R0W-0	RW-0							RW-1

位	名称	描述
Bit[23:16]	SEG 13 Data	LCD Segment 13 Data Segment Data
Bit[7:0]	SEG 12 Data	LCD Segment 12 Data Segment Data

17.3.3. PT7 寄存器 2

When GPIO Mode

GPIO Base Address + 0x68 (0x40868)									
Symbol	PT74CFG/ PT75CFG (PT7 Control Register 2)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT75OE	PT75IE	PT75DO	PT75DI
RW	R0W-0	RW-0							RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT74OE	PT74IE	PT74DO	PT74DI
RW	R0W-0	RW-0							RW-1

位	名称	描述	
Bit[19]	PT75OE	PT7.5 Output Enable	
		0	关闭
		1	开启
Bit[18]	PT75IE	PT7.5 Input Enable	
		0	关闭
		1	开启
Bit[17]	PT75DO	PT7.5 Output Data	
		0	Output Low
		1	Output High

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述
Bit[16]	PT75DI	PT7.5 Input Data
		0 Input Low
		1 Input High
Bit[3]	PT74OE	PT7.4 Output Enable
		0 关闭
		1 开启
Bit[2]	PT74IE	PT7.4 Input Enable
		0 关闭
		1 开启
Bit[1]	PT74DO	PT7.4 Output Data
		0 Output Low
		1 Output High
Bit[0]	PT74DI	PT7.4 Input Data
		0 Input Low
		1 Input High

When LCD Mode

GPIO Base Address + 0x68 (0x40868)									
Symbol	SEG14/SEG15 (PT7 Control Register 2)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	SEG15 Data							
RW	R0W-0	RW-0							RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	SEG14 Data							
RW	R0W-0	RW-0							RW-1

位	名称	描述
Bit[23:16]	SEG 15 Data	LCD Segment 15 Data Segment Data
Bit[7:0]	SEG 14 Data	LCD Segment 14 Data Segment Data

17.3.4. PT7 寄存器 3

When GPIO Mode

GPIO Base Address + 0x6C (0x4086C)									
Symbol	PT76CFG/ PT77CFG (PT7 Control Register 3)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT77OE	PT77IE	PT77DO	PT77DI
RW	R0W-0	RW-0							RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT76OE	PT76IE	PT76DO	PT76DI
RW	R0W-0	RW-0							RW-1

位	名称	描述
Bit[19]	PT77OE	PT7.7 Output Enable
		0 关闭
		1 开启
Bit[18]	PT77IE	PT7.7 Input Enable

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



		0	关闭
		1	开启
Bit[17]	PT77DO	PT7.7 Output Data	
		0	Output Low
		1	Output High
Bit[16]	PT77DI	PT7.7 Input Data	
		0	Input Low
		1	Input High
Bit[03]	PT76OE	PT7.6 Output Enable	
		0	关闭
		1	开启
Bit[02]	PT76IE	PT7.6 Input Enable	
		0	关闭
		1	开启
Bit[01]	PT76DO	PT7.6 Output Data	
		0	Output Low
		1	Output High
Bit[00]	PT76DI	PT7.6 Input Data	
		0	Input Low
		1	Input High

When LCD Mode

GPIO Base Address + 0x6C (0x4086C)									
Symbol	SEG16/SEG17 (PT7 Control Register 3)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	SEG17 Data							
RW	R0W-0	RW-0							RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	SEG16 Data							
RW	R0W-0	RW-0							RW-1

位	名称	描述
Bit[23:16]	SEG 17 Data	LCD Segment 17 Data Segment Data
Bit[7:0]	SEG 16 Data	LCD Segment 16 Data

18. 通用 GPIO PT8 管理

18.1. 整体总说明

PT8 具有 8 个 IO 引脚，可以作为通用的普通 IO 引脚亦可复用为 LCD 功能输出引脚。针对不同的复用，需要做不同的设置。IO 引脚不支持内建上拉电阻功能，也不支持中断输入功能。

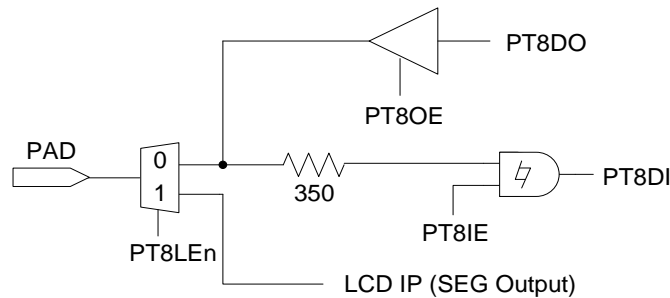


图 18-1 PT8 功能方框图

PT8 具有输入、输出的功能，且分别有不同的控制器来设置。

输出模式

控制器 PT8xOE 可设置每个 IO 输出模式的开启与关闭，每一位对应一个 IO 引脚。当 IO 对应位被置<1>，则开启对应 IO 输出模式；若被置<0>，则关闭输出模式。

透过控制位 PT8xDO 来控制对应 IO 引脚的输出状态为 1 或 0。在低功耗模式下，若 IO 必须开启输出模式，可根据外围电路来设置输出状态，来减低芯片功耗。此模式下不能同时开启输入、输出模式，因而在开启输出模式时，需要关闭 IO 的输入模式。

说明：上述 x 代表的是 0~7，对应到 PT8.0~PT8.7。

输入模式

控制器 PT8xIE 可设置每个 IO 引脚输入模式的开启与关闭，每一位对应一个 IO 引脚。当控制器对应位被置<1>，则开启对应 IO 引脚的输入模式；若被置<0>，则关闭输入模式。透过控制器 PT8xDI 可读取当前对应 IO 引脚的输入状态为 1 或 0。

当 IO 被设置为输入模式，需要接入外部上拉电阻，不能允许 IO 引脚出现浮接状态，以免造成芯片产生漏电现象；特别是在低功耗模式下，建议将 IO 引脚设置为输入模式。在开启输入模式前，需要关闭对应 IO 引脚的输出模式。

说明：上述 x 代表的是 0~7，对应到 PT8.0~PT8.7。

LCD 模式

控制器 SEGx[7:0]决定 LCD SEGMENT 输出数据，

若 LCD 为 1/8 Duty 模式，则 SEGx[7:0]则决定 1/8 Duty 数据内容；

若 LCD 为 1/7 Duty 模式，则 SEGx[6:0]则决定 1/7 Duty 数据内容；

若 LCD 为 1/6 Duty 模式，则 SEGx[5:0]则决定 1/6 Duty 数据内容；

若 LCD 为 1/5 Duty 模式，则 SEGx[4:0]则决定 1/5 Duty 数据内容；

若 LCD 为 1/4 Duty 模式，则 SEGx[3:0]则决定 1/4 Duty 数据内容；

若 LCD 为 1/3 Duty 模式，则 SEGx[2:0]则决定 1/3 Duty 数据内容；

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



说明：上述 x 代表的是 18~25，对应到 SEG18~SEG25。

18.2. 寄存器地址

GPIO Mode Register Address	31	24	23	16	15	8	7	0
GPIO Base Address + 0x70(0x40870)	MASK1			PT81CFG		MASK0		PT80CFG
GPIO Base Address + 0x74(0x40874)	MASK3			PT83CFG		MASK2		PT82CFG
GPIO Base Address + 0x78(0x40878)	MASK5			PT85CFG		MASK4		PT84CFG
GPIO Base Address + 0x7C(0x4087C)	MASK7			PT87CFG		MASK6		PT86CFG

LCD Mode Register Address	31	24	23	16	15	8	7	0
GPIO Base Address + 0x70(0x40870)	MASK1			SEG19		MASK0		SEG18
GPIO Base Address + 0x74(0x40874)	MASK3			SEG21		MASK2		SEG20
GPIO Base Address + 0x78(0x40878)	MASK5			SEG23		MASK4		SEG22
GPIO Base Address + 0x7C(0x4087C)	MASK7			SEG25		MASK6		SEG24

LCD Register Address 0x41B04 决定设定为 GPIO Mode 或是 LCD Mode.

18.3. 寄存器功能

18.3.1. PT8 寄存器 0

When GPIO Mode

GPIO Base Address + 0x70 (0x40870)										
Symbol	PT80CFG/ PT81CFG (PT8 Control Register 0)									
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]	
名称	MASK	-	-	-	-	PT81OE	PT81IE	PT81DO	PT81DI	
RW	R0W-0	RW-0						RW-1		RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]	
名称	MASK	-	-	-	-	PT80OE	PT80IE	PT80DO	PT80DI	
RW	R0W-0	RW-0						RW-1		RW-1

位	名称	描述	
Bit[19]	PT81OE	PT8.1 Output Enable	
		0	关闭
		1	开启
Bit[18]	PT81IE	PT8.1 Input Enable	
		0	关闭
		1	开启
Bit[17]	PT81DO	PT8.1 Output Data	
		0	Output Low
		1	Output High
Bit[16]	PT81DI	PT8.1 Input Data	
		0	Input Low
		1	Input High
Bit[03]	PT80OE	PT8.0 Output Enable	
		0	关闭
		1	开启

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述	
Bit[02]	PT80IE	PT8.0 Input Enable	
		0	关闭
		1	开启
Bit[01]	PT80DO	PT8.0 Output Data	
		0	Output Low
		1	Output High
Bit[00]	PT80DI	PT8.0 Input Data	
		0	Input Low
		1	Input High

When LCD Mode

GPIO Base Address + 0x70 (0x40870)									
Symbol	SEG18/SEG19 (PT8 Control Register 0)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	SEG19 Data							
RW	R0W-0	RW-0							RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	SEG18 Data							
RW	R0W-0	RW-0							RW-1

位	名称	描述
Bit[23:16]	SEG 19 Data	LCD Segment 19 Data Segment Data
Bit[07:00]	SEG 18 Data	LCD Segment 18 Data Segment Data

18.3.2. PT8 寄存器 1

When GPIO Mode

GPIO Base Address + 0x74 (0x40874)									
Symbol	PT82CFG/ PT83CFG (PT8 Control Register 1)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT83OE	PT83IE	PT83DO	PT83DI
RW	R0W-0	RW-0							RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT82OE	PT82IE	PT82DO	PT82DI
RW	R0W-0	RW-0							RW-1

位	名称	描述	
Bit[19]	PT83OE	PT8.3 Output Enable	
		0	关闭
		1	开启
Bit[18]	PT83IE	PT8.3 Input Enable	
		0	关闭
		1	开启
Bit[17]	PT83DO	PT8.3 Output Data	
		0	Output Low
		1	Output High
Bit[16]	PT83DI	PT8.3 Input Data	

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述	
		0	Input Low
		1	Input High
Bit[03]	PT82OE	PT8.2 Output Enable	
		0	关闭
		1	开启
Bit[02]	PT82IE	PT8.2 Input Enable	
		0	关闭
		1	开启
Bit[01]	PT82DO	PT8.2 Output Data	
		0	Output Low
		1	Output High
Bit[00]	PT82DI	PT8.2 Input Data	
		0	Input Low
		1	Input High

When LCD Mode

GPIO Base Address + 0x74 (0x40874)									
Symbol	SEG20/SEG21 (PT8 Control Register 1)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	SEG21 Data							
RW	R0W-0	RW-0							RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	SEG20 Data							
RW	R0W-0	RW-0							RW-1

位	名称	描述
Bit[23:16]	SEG 21 Data	LCD Segment 21 Data Segment Data
Bit[7:0]	SEG 20 Data	LCD Segment 20 Data Segment Data

18.3.3. PT8 寄存器 2

When GPIO Mode

GPIO Base Address + 0x78 (0x40878)									
Symbol	PT84CFG/ PT85CFG (PT8 Control Register 2)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT85OE	PT85IE	PT85DO	PT85DI
RW	R0W-0	RW-0							RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT84OE	PT84IE	PT84DO	PT84DI
RW	R0W-0	RW-0							RW-1

位	名称	描述	
Bit[19]	PT85OE	PT8.5 Output Enable	
		0	关闭
		1	开启
Bit[18]	PT85IE	PT8.5 Input Enable	
		0	关闭
		1	开启

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述	
		1	开启
Bit[17]	PT85DO	PT8.5 Output Data	
		0	Output Low
		1	Output High
Bit[16]	PT85DI	PT8.5 Input Data	
		0	Input Low
		1	Input High
Bit[03]	PT84OE	PT8.4 Output Enable	
		0	关闭
		1	开启
Bit[02]	PT84IE	PT8.4 Input Enable	
		0	关闭
		1	开启
Bit[01]	PT84DO	PT8.4 Output Data	
		0	Output Low
		1	Output High
Bit[00]	PT84DI	PT8.4 Input Data	
		0	Input Low
		1	Input High

When LCD Mode

GPIO Base Address + 0x78 (0x40878)									
Symbol	SEG22/SEG23 (PT8 Control Register 2)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	SEG23 Data							
RW	R0W-0	RW-0							RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	SEG22 Data							
RW	R0W-0	RW-0							RW-1

位	名称	描述
Bit[23:16]	SEG 23 Data	LCD Segment 23 Data
		Segment Data
Bit[7:0]	SEG 22 Data	LCD Segment 22 Data
		Segment Data

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



18.3.4. PT8 寄存器 3

When GPIO Mode

GPIO Base Address + 0x7C (0x4087C)									
Symbol	PT86CFG/ PT87CFG (PT8 Control Register 3)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT87OE	PT87IE	PT87DO	PT87DI
RW	R0W-0	RW-0						RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT86OE	PT86IE	PT86DO	PT86DI
RW	R0W-0	RW-0						RW-1	

位	名称	描述	
Bit[19]	PT87OE	PT8.7 Output Enable	
		0	关闭
		1	开启
Bit[18]	PT87IE	PT8.7 Input Enable	
		0	关闭
		1	开启
Bit[17]	PT87DO	PT8.7 Output Data	
		0	Output Low
		1	Output High
Bit[16]	PT87DI	PT8.7 Input Data	
		0	Input Low
		1	Input High
Bit[03]	PT86OE	PT8.6 Output Enable	
		0	关闭
		1	开启
Bit[02]	PT86IE	PT8.6 Input Enable	
		0	关闭
		1	开启
Bit[01]	PT86DO	PT8.6 Output Data	
		0	Output Low
		1	Output High
Bit[00]	PT86DI	PT8.6 Input Data	
		0	Input Low
		1	Input High

When LCD Mode

GPIO Base Address + 0x7C (0x4087C)									
Symbol	SEG24/SEG25 (PT8 Control Register 3)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	SEG25 Data							
RW	R0W-0	RW-0						RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	SEG24 Data							
RW	R0W-0	RW-0						RW-1	

位	名称	描述
---	----	----

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



Bit[23:16]	SEG 25 Data	LCD Segment 25 Data Segment Data
Bit[7:0]	SEG 24 Data	LCD Segment 24 Data

19. 通用 GPIO PT9 管理

19.1. 整体总说明

PT9 具有 8 个 IO 引脚，可以作为通用的普通 IO 引脚 PWM 或是通信引脚功能，亦可复用为 LCD 功能输出引脚，针对不同的复用，需要做不同的设置。IO 引脚不支持内建上拉电阻功能，也不支持中断输入功能。

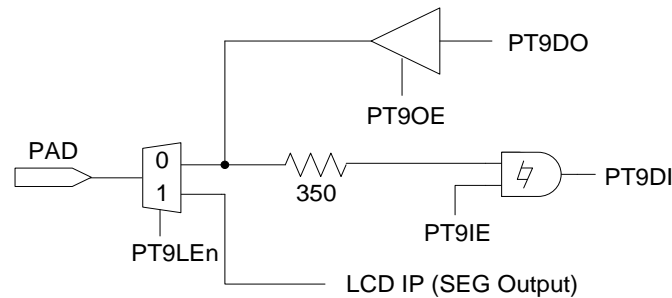


图 19-1 PT9 功能方框图

PT9 具有输入、输出的功能，且分别有不同的控制器来设置。

输出模式

控制器 PT9xOE 可设置每个 IO 输出模式的开启与关闭，每一位对应一个 IO 引脚。当 IO 对应位被置<1>，则开启对应 IO 输出模式；若被置<0>，则关闭输出模式。透过控制位 PT9xDO 来控制对应 IO 引脚的输出状态为 1 或 0。在低功耗模式下，若 IO 必须开启输出模式，可根据外围电路来设置输出状态，来减低芯片功耗。此模式下不能同时开启输入、输出模式，因而在开启输出模式时，需要关闭 IO 的输入模式。

说明：上述 x 代表的是 0~7，对应到 PT9.0~PT9.7。

输入模式

控制器 PT9xIE 可设置每个 IO 引脚输入模式的开启与关闭，每一位对应一个 IO 引脚。当控制器对应位被置<1>，则开启对应 IO 引脚的输入模式；若被置<0>，则关闭输入模式。透过控制器 PT9xDI 可读取当前对应 IO 引脚的输入状态为 1 或 0。当 IO 被设置为输入模式，需要接入外部上拉电阻，不能允许 IO 引脚出现浮接状态，以免造成芯片产生漏电现象；特别是在低功耗模式下，建议将 IO 引脚设置为输入模式。在开启输入模式前，需要关闭对应 IO 引脚的输出模式。

说明：上述 x 代表的是 0~7，对应到 PT9.0~PT9.7。

LCD 模式

控制器 SEGx[7:0]决定 LCD SEGMENT 输出数据，

若 LCD 为 1/8 Duty 模式，则 SEGx[7:0]则决定 1/8 Duty 数据内容；

若 LCD 为 1/7 Duty 模式，则 SEGx[6:0]则决定 1/7 Duty 数据内容；

若 LCD 为 1/6 Duty 模式，则 SEGx[5:0]则决定 1/6 Duty 数据内容；

若 LCD 为 1/5 Duty 模式，则 SEGx[4:0]则决定 1/5 Duty 数据内容；

若 LCD 为 1/4 Duty 模式，则 SEGx[3:0]则决定 1/4 Duty 数据内容；

若 LCD 为 1/3 Duty 模式，则 SEGx[2:0]则决定 1/3 Duty 数据内容；

说明：上述 x 代表的是 26~33，对应到 SEG26~SEG33。

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



19.2. 寄存器地址

GPIO Mode Register Address	31	24	23	16	15	8	7	0
GPIO Base Address + 0x80(0x40880)	MASK1		PT91CFG		MASK0		PT90CFG	
GPIO Base Address + 0x84(0x40884)	MASK3		PT93CFG		MASK2		PT92CFG	
GPIO Base Address + 0x88(0x40888)	MASK5		PT95CFG		MASK4		PT94CFG	
GPIO Base Address + 0x8C(0x4088C)	MASK7		PT97CFG		MASK6		PT96CFG	

LCD Mode Register Address	31	24	23	16	15	8	7	0
GPIO Base Address + 0x80(0x40880)	MASK1		SEG27		MASK0		SEG26	
GPIO Base Address + 0x84(0x40884)	MASK3		SEG29		MASK2		SEG28	
GPIO Base Address + 0x88(0x40888)	MASK5		SEG31		MASK4		SEG30	
GPIO Base Address + 0x8C(0x4088C)	MASK7		SEG33		MASK6		SEG32	

LCD Register Address 0x41B04 决定设定为 GPIO Mode 或是 LCD Mode.

19.3. 寄存器功能

19.3.1. PT9 寄存器 0

When GPIO Mode

GPIO Base Address + 0x80 (0x40880)									
Symbol	PT90CFG/ PT91CFG (PT9 Control Register 0)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT91OE	PT91IE	PT91DO	PT91DI
RW	R0W-0	RW-0						RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT90OE	PT90IE	PT90DO	PT90DI
RW	R0W-0	RW-0						RW-1	

位	名称	描述
Bit[19]	PT91OE	PT9.1 Output Enable
		0 关闭
		1 开启
Bit[18]	PT91IE	PT9.1 Input Enable
		0 关闭
		1 开启
Bit[17]	PT91DO	PT9.1 Output Data
		0 Output Low
		1 Output High
Bit[16]	PT91DI	PT9.1 Input Data
		0 Input Low
		1 Input High
Bit[03]	PT90OE	PT9.0 Output Enable
		0 关闭
		1 开启
Bit[02]	PT90IE	PT9.0 Input Enable

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述	
		0	关闭
		1	开启
Bit[01]	PT90DO	PT9.0 Output Data	
		0	Output Low
		1	Output High
Bit[00]	PT90DI	PT9.0 Input Data	
		0	Input Low
		1	Input High

When LCD Mode

GPIO Base Address + 0x80 (0x40880)										
Symbol	SEG26/SEG27 (PT9 Control Register 0)									
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]	
名称	MASK	SEG27 Data								
RW	R0W-0	RW-0							RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]	
名称	MASK	SEG26 Data								
RW	R0W-0	RW-0							RW-1	

位	名称	描述
Bit[23:16]	SEG 27 Data	LCD Segment 27 Data Segment Data
Bit[7:0]	SEG 26 Data	LCD Segment 26 Data Segment Data

19.3.2. PT9 寄存器 1

When GPIO Mode.

GPIO Base Address + 0x84 (0x40884)										
Symbol	PT92CFG/ PT93CFG (PT9 Control Register 1)									
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]	
名称	MASK	-	-	-	-	PT93OE	PT93IE	PT93DO	PT93DI	
RW	R0W-0	RW-0							RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]	
名称	MASK	-	-	-	-	PT92OE	PT92IE	PT92DO	PT92DI	
RW	R0W-0	RW-0							RW-1	

位	名称	描述	
Bit[19]	PT93OE	PT9.3 Output Enable	
		0	关闭
		1	开启
Bit[18]	PT93IE	PT9.3 Input Enable	
		0	关闭
		1	开启
Bit[17]	PT93DO	PT9.3 Output Data	
		0	Output Low
		1	Output High
Bit[16]	PT93DI	PT9.3 Input Data	
		0	Input Low

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述
		1 Input High
Bit[3]	PT92OE	PT9.2 Output Enable
		0 关闭
		1 开启
Bit[2]	PT92IE	PT9.2 Input Enable
		0 关闭
		1 开启
Bit[1]	PT92DO	PT9.2 Output Data
		0 Output Low
		1 Output High
Bit[0]	PT92DI	PT9.2 Input Data
		0 Input Low
		1 Input High

When LCD Mode

GPIO Base Address + 0x84 (0x40884)									
Symbol	SEG28/SEG29 (PT9 Control Register 1)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	SEG29 Data							
RW	R0W-0	RW-0							RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	SEG28 Data							
RW	R0W-0	RW-0							RW-1

位	名称	描述
Bit[23:16]	SEG 29 Data	LCD Segment 29 Data Segment Data
Bit[7:0]	SEG 28 Data	LCD Segment 28 Data Segment Data

19.3.3. PT9 寄存器 2

When GPIO Mode

GPIO Base Address + 0x88 (0x40888)									
Symbol	PT94CFG/ PT95CFG (PT9 Control Register 2)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT95OE	PT95IE	PT95DO	PT95DI
RW	R0W-0	RW-0							RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT94OE	PT94IE	PT94DO	PT94DI
RW	R0W-0	RW-0							RW-1

位	名称	描述
Bit[19]	PT95OE	PT9.5 Output Enable
		0 关闭
		1 开启
Bit[18]	PT95IE	PT9.5 Input Enable
		0 关闭

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述	
		1	开启
Bit[17]	PT95DO	PT9.5 Output Data	
		0	Output Low
		1	Output High
Bit[16]	PT95DI	PT9.5 Input Data	
		0	Input Low
		1	Input High
Bit[03]	PT94OE	PT9.4 Output Enable	
		0	关闭
		1	开启
Bit[02]	PT94IE	PT9.4 Input Enable	
		0	关闭
		1	开启
Bit[01]	PT94DO	PT9.4 Output Data	
		0	Output Low
		1	Output High
Bit[00]	PT94DI	PT9.4 Input Data	
		0	Input Low
		1	Input High

When LCD Mode

GPIO Base Address + 0x88 (0x40888)									
Symbol	SEG30/SEG31 (PT9 Control Register 2)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	SEG31 Data							
RW	R0W-0	RW-0							RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	SEG30 Data							
RW	R0W-0	RW-0							RW-1

位	名称	描述
Bit[23:16]	SEG 31 Data	LCD Segment 31 Data
		Segment Data
Bit[7:0]	SEG 30 Data	LCD Segment 30 Data
		Segment Data

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



19.3.4. PT9 寄存器 3

When GPIO Mode

GPIO Base Address + 0x8C (0x4088C)									
Symbol	PT96CFG/ PT97CFG (PT9 Control Register 3)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT97OE	PT97IE	PT97DO	PT97DI
RW	R0W-0	RW-0						RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT96OE	PT96IE	PT96DO	PT96DI
RW	R0W-0	RW-0						RW-1	

位	名称	描述	
Bit[19]	PT97OE	PT9.7 Output Enable	
		0	关闭
		1	开启
Bit[18]	PT97IE	PT9.7 Input Enable	
		0	关闭
		1	开启
Bit[17]	PT97DO	PT9.7 Output Data	
		0	Output Low
		1	Output High
Bit[16]	PT97DI	PT9.7 Input Data	
		0	Input Low
		1	Input High
Bit[03]	PT96OE	PT9.6 Output Enable	
		0	关闭
		1	开启
Bit[02]	PT96IE	PT9.6 Input Enable	
		0	关闭
		1	开启
Bit[01]	PT96DO	PT9.6 Output Data	
		0	Output Low
		1	Output High
Bit[00]	PT96DI	PT9.6 Input Data	
		0	Input Low
		1	Input High

When LCD Mode

GPIO Base Address + 0x8c (0x4088c)									
Symbol	SEG32/SEG33 (PT9 Control Register 3)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	SEG33 Data							
RW	R0W-0	RW-0						RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	SEG32 Data							
RW	R0W-0	RW-0						RW-1	

位	名称	描述
---	----	----

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



Bit[23:16]	SEG 33 Data	LCD Segment 33 Data Segment Data
Bit[7:0]	SEG 32 Data	LCD Segment 32 Data Segment Data

20. 通用 GPIO PT10 管理

20.1. 整体总说明

PT10 具有 8 个 IO 引脚，可以作为通用的普通 IO 引脚 PWM 或是通信引脚功能，亦可复用为 LCD 功能输出引脚，针对不同的复用，需要做不同的设置。IO 引脚不支持内建上拉电阻功能，也不支持中断输入功能。

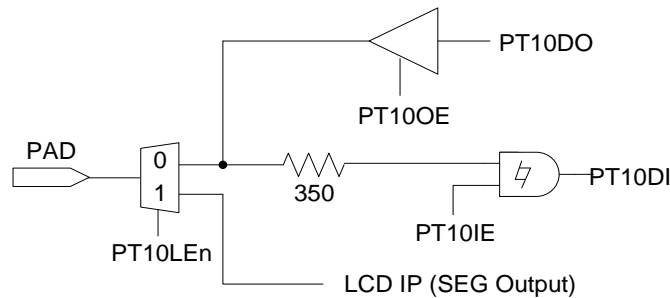


图 20-1 PT10 功能方框图

PT10 具有输入、输出的功能，且分别有不同的控制器来设置。

输出模式

控制器 PT10xOE 可设置每个 IO 输出模式的开启与关闭，每一位对应一个 IO 引脚。当 IO 对应位被置<1>，则开启对应 IO 输出模式；若被置<0>，则关闭输出模式。透过控制位 PT10xDO 来控制对应 IO 引脚的输出状态为 1 或 0。在低功耗模式下，若 IO 必须开启输出模式，可根据外围电路来设置输出状态，来减低芯片功耗。此模式下不能同时开启输入、输出模式，因而在开启输出模式时，需要关闭 IO 的输入模式。

说明：上述 x 代表的是 0~7，对应到 PT10.0~PT10.7。

输入模式

控制器 PT10xIE 可设置每个 IO 引脚输入模式的开启与关闭，每一位对应一个 IO 引脚。当控制器对应位被置<1>，则开启对应 IO 引脚的输入模式；若被置<0>，则关闭输入模式。透过控制器 PT10xDI 可读取当前对应 IO 引脚的输入状态为 1 或 0。当 IO 被设置为输入模式，需要接入外部上拉电阻，不能允许 IO 引脚出现浮接状态，以免造成芯片产生漏电现象；特别是在低功耗模式下，建议将 IO 引脚设置为输入模式。在开启输入模式前，需要关闭对应 IO 引脚的输出模式。

说明：上述 x 代表的是 0~7，对应到 PT10.0~PT10.7。

LCD 模式

控制器 SEGx[7:0]决定 LCD SEGMENT 输出数据，

若 LCD 为 1/8 Duty 模式，则 SEGx[7:0]则决定 1/8 Duty 数据内容；

若 LCD 为 1/7 Duty 模式，则 SEGx[6:0]则决定 1/7 Duty 数据内容；

若 LCD 为 1/6 Duty 模式，则 SEGx[5:0]则决定 1/6 Duty 数据内容；

若 LCD 为 1/5 Duty 模式，则 SEGx[4:0]则决定 1/5 Duty 数据内容；

若 LCD 为 1/4 Duty 模式，则 SEGx[3:0]则决定 1/4 Duty 数据内容；

若 LCD 为 1/3 Duty 模式，则 SEGx[2:0]则决定 1/3 Duty 数据内容；

说明：上述 x 代表的是 34~41，对应到 SEG34~SEG41。

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



20.2. 寄存器地址

GPIO Mode Register Address	31	24	23	16	15	8	7	0
GPIO Base Address + 0x90(0x40890)	MASK1			PT101CFG		MASK0		PT100CFG
GPIO Base Address + 0x94(0x40894)	MASK3			PT103CFG		MASK2		PT102CFG
GPIO Base Address + 0x98(0x40898)	MASK5			PT105CFG		MASK4		PT104CFG
GPIO Base Address + 0x9C(0x4089C)	MASK7			PT107CFG		MASK6		PT106CFG

LCD Mode Register Address	31	24	23	16	15	8	7	0
GPIO Base Address + 0x90(0x40890)	MASK1			SEG35		MASK0		SEG34
GPIO Base Address + 0x94(0x40894)	MASK3			SEG37		MASK2		SEG36
GPIO Base Address + 0x98(0x40898)	MASK5			SEG39		MASK4		SEG38
GPIO Base Address + 0x9C(0x4089C)	MASK7			SEG41		MASK6		SEG40

LCD Register Address 0x41B04 决定设定为 GPIO Mode 或是 LCD Mode.

20.3. 寄存器功能

20.3.1. PT10 寄存器 0

When GPIO Mode

GPIO Base Address + 0x90 (0x40890)									
PT100CFG/ PT101CFG (PT10 Control Register 0)									
Symbol	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT101OE	PT101IE	PT101DO	PT101DI
RW	R0W-0	RW-0						RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT100OE	PT100IE	PT100DO	PT100DI
RW	R0W-0	RW-0						RW-1	

位	名称	描述
Bit[19]	PT101OE	PT10.1 Output Enable
		0 关闭
		1 开启
Bit[18]	PT101IE	PT10.1 Input Enable
		0 关闭
		1 开启
Bit[17]	PT101DO	PT10.1 Output Data
		0 Output Low
		1 Output High
Bit[16]	PT101DI	PT10.1 Input Data
		0 Input Low
		1 Input High
Bit[03]	PT100OE	PT10.0 Output Enable
		0 关闭
		1 开启
Bit[02]	PT100IE	PT10.0 Input Enable

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述	
		0	关闭
		1	开启
Bit[01]	PT100DO	PT10.0 Output Data	
		0	Output Low
		1	Output High
Bit[00]	PT100DI	PT10.0 Input Data	
		0	Input Low
		1	Input High

When LCD Mode

GPIO Base Address + 0x90 (0x40890)										
Symbol	SEG34/SEG35 (PT10 Control Register 0)									
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]	
名称	MASK	SEG35 Data								
RW	R0W-0	RW-0							RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]	
名称	MASK	SEG34 Data								
RW	R0W-0	RW-0							RW-1	

位	名称	描述
Bit[23:16]	SEG 35 Data	LCD Segment 35 Data Segment Data
Bit[7:0]	SEG 34 Data	LCD Segment 34 Data Segment Data

20.3.2. PT10 寄存器 1

When GPIO Mode.

GPIO Base Address + 0x94 (0x40894)										
Symbol	PT102CFG/ PT103CFG (PT10 Control Register 1)									
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]	
名称	MASK	-	-	-	-	PT103OE	PT103IE	PT103DO	PT103DI	
RW	R0W-0	RW-0							RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]	
名称	MASK	-	-	-	-	PT102OE	PT102IE	PT102DO	PT102DI	
RW	R0W-0	RW-0							RW-1	

位	名称	描述	
Bit[19]	PT103OE	PT10.3 Output Enable	
		0	关闭
		1	开启
Bit[18]	PT103IE	PT10.3 Input Enable	
		0	关闭
		1	开启
Bit[17]	PT103DO	PT10.3 Output Data	
		0	Output Low
		1	Output High
Bit[16]	PT103DI	PT10.3 Input Data	
		0	Input Low

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述
		1 Input High
Bit[3]	PT102OE	PT10.2 Output Enable
		0 关闭
		1 开启
Bit[2]	PT102IE	PT10.2 Input Enable
		0 关闭
		1 开启
Bit[1]	PT102DO	PT10.2 Output Data
		0 Output Low
		1 Output High
Bit[0]	PT102DI	PT10.2 Input Data
		0 Input Low
		1 Input High

When LCD Mode

GPIO Base Address + 0x94 (0x40894)									
Symbol	SEG36/SEG37 (PT10 Control Register 1)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	SEG37 Data							
RW	R0W-0	RW-0							RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	SEG36 Data							
RW	R0W-0	RW-0							RW-1

位	名称	描述
Bit[23~16]	SEG 37 Data	LCD Segment 37 Data Segment Data
Bit[7~0]	SEG 36 Data	LCD Segment 36 Data Segment Data

20.3.3. PT10 寄存器 2

When GPIO Mode

GPIO Base Address + 0x98 (0x40898)									
Symbol	PT104CFG/ PT105CFG (PT10 Control Register 2)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT105OE	PT105IE	PT105DO	PT105DI
RW	R0W-0	RW-0							RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT104OE	PT104IE	PT104DO	PT104DI
RW	R0W-0	RW-0							RW-1

位	名称	描述
Bit[19]	PT105OE	PT10.5 Output Enable
		0 关闭
		1 开启
Bit[18]	PT105IE	PT10.5 Input Enable
		0 关闭

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述	
		1	开启
Bit[17]	PT105DO	PT10.5 Output Data	
		0	Output Low
		1	Output High
Bit[16]	PT105DI	PT10.5 Input Data	
		0	Input Low
		1	Input High
Bit[03]	PT104OE	PT10.4 Output Enable	
		0	关闭
		1	开启
Bit[02]	PT104IE	PT10.4 Input Enable	
		0	关闭
		1	开启
Bit[01]	PT104DO	PT10.4 Output Data	
		0	Output Low
		1	Output High
Bit[00]	PT104DI	PT10.4 Input Data	
		0	Input Low
		1	Input High

When LCD Mode

GPIO Base Address + 0x98 (0x40898)									
Symbol	SEG38/SEG39 (PT10 Control Register 2)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	SEG39 Data							
RW	R0W-0	RW-0							RW-1
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	SEG38 Data							
RW	R0W-0	RW-0							RW-1

位	名称	描述
Bit[23:16]	SEG 39 Data	LCD Segment 39 Data
		Segment Data
Bit[7:0]	SEG 38 Data	LCD Segment 38 Data
		Segment Data

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



20.3.4. PT10 寄存器 3

When GPIO Mode

GPIO Base Address + 0x9C (0x4089C)									
Symbol	PT106CFG/ PT107CFG (PT10 Control Register 3)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT107OE	PT107IE	PT107DO	PT107DI
RW	R0W-0	RW-0						RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT106OE	PT106IE	PT106DO	PT106DI
RW	R0W-0	RW-0						RW-1	

位	名称	描述
Bit[19]	PT107OE	PT10.7 Output Enable
		0 关闭
		1 开启
Bit[18]	PT107IE	PT10.7 Input Enable
		0 关闭
		1 开启
Bit[17]	PT107DO	PT10.7 Output Data
		0 Output Low
		1 Output High
Bit[16]	PT107DI	PT10.7 Input Data
		0 Input Low
		1 Input High
Bit[03]	PT106OE	PT10.6 Output Enable
		0 关闭
		1 开启
Bit[02]	PT106IE	PT10.6 Input Enable
		0 关闭
		1 开启
Bit[01]	PT106DO	PT10.6 Output Data
		0 Output Low
		1 Output High
Bit[00]	PT106DI	PT10.6 Input Data
		0 Input Low
		1 Input High

When LCD Mode

GPIO Base Address + 0x9C (0x4089C)									
Symbol	SEG40/SEG41 (PT10 Control Register 3)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	SEG41 Data							
RW	R0W-0	RW-0						RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	SEG40 Data							
RW	R0W-0	RW-0						RW-1	

位	名称	描述
---	----	----

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



Bit[23:16]	SEG 41 Data	LCD Segment 41 Data Segment Data
Bit[7:0]	SEG 40 Data	LCD Segment 40 Data Segment Data

21. 通用 GPIO PT13 管理

21.1. 整体总说明

PT13 具有 8 个 IO 引脚，可以作为通用的普通 IO 引脚亦可复用为 LCD 功能输出引脚。不支持内部上拉电阻，也不支持外部中断功能。针对不同的复用，需要做不同的设置。

PT13 具有输入、输出的功能，且分别有不同的控制器来设置。

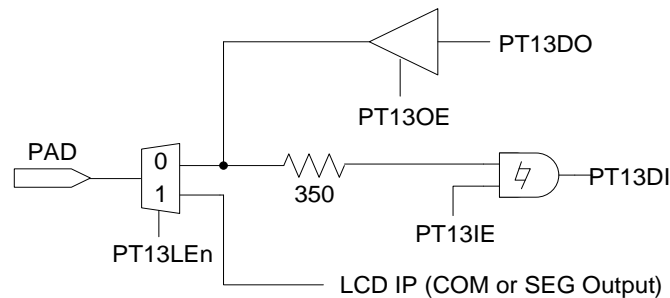


图 21-1 PT13 功能方框图

输出模式

控制器 PT13xOE 可设置每个 IO 输出模式的开启与关闭，每一位对应一个 IO 接脚。当 IO 对应位被置<1>，则开启对应 IO 输出模式；若被置<0>，则关闭输出模式。透过控制位 PT13xDO 来控制对应 IO 引脚的输出状态为 1 或 0。在低功耗模式下，若 IO 必须开启输出模式，可根据外围电路来设置输出状态，来减低芯片功耗。此模式下不能同时开启输入、输出模式，因而在开启输出模式时，需要关闭 IO 的输入模式。

说明：上述 x 代表的是 0~7，对应到 PT13.0~PT13.7

输入模式

控制器 PT13xIE 可设置每个 IO 引脚输入模式的开启与关闭，每一位对应一个 IO 引脚。当控制器对应位被置<1>，则开启对应 IO 引脚的输入模式；若被置<0>，则关闭输入模式。透过控制器 PT13xDI 可读取当前对应 IO 引脚的输入状态为 1 或 0。当 IO 被设置为输入模式，要接入外部上拉电阻，不能允许 IO 引脚出现浮接状态，以免造成芯片产生漏电现象；特别是在低功耗模式下，建议将 IO 引脚设置为输入模式。在开启输入模式前，需要关闭对应 IO 引脚的输出模式。

说明：上述 x 代表的是 0~7，对应到 PT13.0~PT13.7。

LCD 模式

控制器 SEGx[7:0]决定 LCD SEGMENT 输出数据，

若 LCD 为 1/8 Duty 模式，则 PT13 全部设计为 COM PORT 使用；

SEG0 只支持 1/3 duty 以及 1/4duty，

SEG1 只支持 1/3 duty、1/4 duty 以及 1/5duty，

SEG42 只支持 1/3 duty、1/4 duty、1/5duty 以及 1/6duty，

SEG43 只支援 1/3 duty、1/4 duty、1/5duty、1/6duty、以及 1/7 duty。

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



21.2. 寄存器地址

GPIO Mode Register Address	31	24	23	16	15	8	7	0
GPIO Base Address + 0xC0(0x408C0)	MASK1		PT131CFG		MASK0		PT130CFG	
GPIO Base Address + 0xC4(0x408C4)	MASK3		PT133CFG		MASK2		PT132CFG	
GPIO Base Address + 0xC8(0x408C8)	MASK5		PT135CFG		MASK4		PT134CFG	
GPIO Base Address + 0xCC(0x408CC)	MASK5		PT137CFG		MASK4		PT136CFG	

LCD Mode Register Address	31	24	23	16	15	8	7	0
GPIO Base Address + 0xC8(0x408C8)	MASK5		SEG1		MASK4		SEG0	
GPIO Base Address + 0xCC(0x408CC)	MASK5		SEG43		MASK4		SEG42	

LCD Register Address 0x41B08 决定设定为 GPIO Mode 或是 LCD Mode.

21.3. 寄存器功能

21.3.1. PT13 寄存器 0

When GPIO Mode

GPIO Base Address + 0xC0 (0x408C0)									
PT130CFG/ PT131CFG (PT13 Control Register 0)									
Symbol	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT131OE	PT131IE	PT131DO	PT131DI
RW	R0W-0	RW-0						RW-1	
Symbol	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT130OE	PT130IE	PT130DO	PT130DI
RW	R0W-0	RW-0						RW-1	

位	名称	描述
Bit[19]	PT131OE	PT13.1 Output Enable
		0 关闭
		1 开启
Bit[18]	PT131IE	PT13.1 Input Enable
		0 关闭
		1 开启
Bit[17]	PT131DO	PT13.1 Output Data
		0 Output Low
		1 Output High
Bit[16]	PT131DI	PT13.1 Input Data
		0 Input Low
		1 Input High
Bit[03]	PT130OE	PT13.0 Output Enable
		0 关闭
		1 开启
Bit[02]	PT130IE	PT13.0 Input Enable
		0 关闭
		1 开启

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述	
Bit[01]	PT130DO	PT13.0 Output Data	
		0	Output Low
		1	Output High
Bit[00]	PT130DI	PT13.0 Input Data	
		0	Input Low
		1	Input High

When LCD Mode

PT13.0=COM0、PT13.1=COM1。

21.3.2. PT13 寄存器 1

When GPIO Mode

GPIO Base Address + 0xC4 (0x408C4)									
Symbol	PT132CFG/ PT133CFG (PT13 Control Register 1)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT133OE	PT133IE	PT133DO	PT133DI
RW	R0W-0	RW-0						RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT132OE	PT132IE	PT132DO	PT132DI
RW	R0W-0	RW-0						RW-1	

位	名称	描述	
Bit[19]	PT133OE	PT13.3 Output Enable	
		0	关闭
		1	开启
Bit[18]	PT133IE	PT13.3 Input Enable	
		0	关闭
		1	开启
Bit[17]	PT133DO	PT13.3 Output Data	
		0	Output Low
		1	Output High
Bit[16]	PT133DI	PT13.3 Input Data	
		0	Input Low
		1	Input High
Bit[03]	PT132OE	PT13.2 Output Enable	
		0	关闭
		1	开启
Bit[02]	PT132IE	PT13.2 Input Enable	
		0	关闭
		1	开启
Bit[01]	PT132DO	PT13.2 Output Data	
		0	Output Low
		1	Output High
Bit[00]	PT132DI	PT13.2 Input Data	
		0	Input Low
		1	Input High

When LCD Mode

PT13.2=COM2、PT13.3=COM3。

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



21.3.3. PT13 寄存器 2

When GPIO Mode

GPIO Base Address + 0xC8 (0x408C8)									
Symbol	PT134CFG/ PT135CFG (PT13 Control Register 2)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT135OE	PT135IE	PT135DO	PT135DI
RW	R0W-0	RW-0						RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT134OE	PT134IE	PT134DO	PT134DI
RW	R0W-0	RW-0						RW-1	

位	名称	描述
Bit[19]	PT135OE	PT13.5 Output Enable
		0 关闭
		1 开启
Bit[18]	PT135IE	PT13.5 Input Enable
		0 关闭
		1 开启
Bit[17]	PT135DO	PT13.5 Output Data
		0 Output Low
		1 Output High
Bit[16]	PT135DI	PT13.5 Input Data
		0 Input Low
		1 Input High
Bit[03]	PT134OE	PT13.4 Output Enable
		0 关闭
		1 开启
Bit[02]	PT134IE	PT13.4 Input Enable
		0 关闭
		1 开启
Bit[01]	PT134DO	PT13.4 Output Data
		0 Output Low
		1 Output High
Bit[00]	PT134DI	PT13.4 Input Data
		0 Input Low
		1 Input High

When LCD Mode

GPIO Base Address + 0xC8 (0x408C8)									
Symbol	SEG0/SEG1 (PT13 Control Register 2)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-				SEG1 Data			
RW	R0W-0	RW-0						RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-			SEG0 Data			
RW	R0W-0	RW-0						RW-1	

位	名称	描述
---	----	----

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



Bit[20:16]	SEG 1 Data	LCD Segment 1 Data (support 1/3 or 1/4 or 1/5 duty mode) Segment Data
Bit[03:00]	SEG 0 Data	LCD Segment 0 Data (support 1/3 or 1/4 duty mode) Segment Data

21.3.4. PT13 寄存器 3

When GPIO Mode

GPIO Base Address + 0xCC (0x408CC)									
Symbol	PT136CFG/ PT137CFG (PT13 Control Register 3)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	PT137OE	PT137IE	PT137DO	PT137DI
RW	R0W-0	RW-0						RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
名称	MASK	-	-	-	-	PT136OE	PT136IE	PT136DO	PT136DI
RW	R0W-0	RW-0						RW-1	

位	名称	描述
Bit[19]	PT137OE	PT13.7 Output Enable
		0 关闭
		1 开启
Bit[18]	PT137IE	PT13.7 Input Enable
		0 关闭
		1 开启
Bit[17]	PT137DO	PT13.7 Output Data
		0 Output Low
		1 Output High
Bit[16]	PT137DI	PT13.7 Input Data
		0 Input Low
		1 Input High
Bit[03]	PT136OE	PT13.6 Output Enable
		0 关闭
		1 开启
Bit[02]	PT136IE	PT13.6 Input Enable
		0 关闭
		1 开启
Bit[01]	PT136DO	PT13.6 Output Data
		0 Output Low
		1 Output High
Bit[00]	PT136DI	PT13.6 Input Data
		0 Input Low
		1 Input High

When LCD Mode

GPIO Base Address + 0xCC (0x408CC)									
Symbol	SEG42/SEG43 (PT13 Control Register 3)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	SEG43 Data						
RW	R0W-0	RW-0						RW-1	
Bit	[15:08]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



名称	MASK	-	-	SEG42 Data	
RW	R0W-0	RW-0			RW-1

位	名称	描述
Bit[22:16]	SEG 43 Data	LCD Segment 43 Data (support 1/3 or 1/4 or 1/5 or 1/6 or 1/7 duty mode) Segment Data
Bit[05:00]	SEG 42 Data	LCD Segment 42 Data (support 1/3 or 1/4 or 1/5 or 1/6 duty mode) Segment Data

22. 通用 GPIO 复用功能管理

22.1. 整体总说明

芯片具有多组通用 IO 引脚大多具有复用功能，需要寄存器控制复用功能的开启与关闭。本章节介绍 IO 复用功能的控制。IO 都具有多个复用功能，但是一般同一时间只能设置某个功能有效，所以在不需用到复用功能时，请记得关闭复用功能，以便作为其它功能使用。也有一些是多个复用功能组合使用，如 PT2，在被设置为外部中断输入时，同样也可配置为 I²C、SPI、UART 等串行引脚，切记要同样是输入才行，这样可以通过通信信号配合外部中断功能，实现由通信产生的外部中断功能。复用功能通常都以各群组为单位才能使用其功能，而且每次只能选择其中一组使用。如欲使用 SPI 功能时，CS_3、CK_3、MISO_3、MOSI_3 为第一组模块，CS_4、CK_4、MISO_4、MOSI_4 为第二组模块，其它模块依此类推。而 SPI 的功能使用上，可以依照使用者设定需求，选择第一组或是第二组使用，但是同一时间上只能设定一组使用。当使用者需要同时使用通信 SPI、I²C、UART 等功能时，则可以设定成 SPI 使用第一组模块(CS_3、CK_3、MISO_3、MOSI_3)，I²C 使用第三组模块(SCL_7、SDA_7)，UART 使用第四组模块(Tx2_4、Rx2_4)，如此便可以透过不同引脚设定达到同时支持效果。下表列出所有 IO 引脚的复用功能，及复用功能的优先权级别，0 代表最高级别，6 代表最低级别。

Function	INT	GPIO	Timer C Capture	Special Function	SPI	I ² C	UART	Analog	Timer B/B2 PWM
Output Priority	I/P	I/P	I/P	0	1	2	3	4	5
PT1.0	INT1.0	DIO	TC11_1		CS_1	SCL_1	Tx_1		PWM0_1
PT1.1	INT1.1	DIO	TC12_1		CK_1	SDA_1	Rx_1		PWM1_1
PT1.2	INT1.2	DIO	TC11_2		MISO_1	SCL_2	Tx2_1		PWM2_1
PT1.3	INT1.3	DIO	TC12_2		MOSI_1	SDA_2	Rx2_1		PWM3_1
PT1.4	INT1.4	DIO	TC11_3		CS_2	SCL_3	Tx_2		PWM0_2
PT1.5	INT1.5	DIO	TC12_3		CK_2	SDA_3	Rx_2		PWM1_2
PT1.6	INT1.6	DIO	TC11_4		MISO_2	SCL_4	Tx2_2		PWM2_2
PT1.7	INT1.7	DIO	TC12_4		MOSI_2	SDA_4	Rx2_2		PWM3_2
PT2.0	INT2.0	DIO	TC11_5		CS_3	SCL_5	Tx_3		PWM0_3
PT2.1	INT2.1	DIO	TC12_5		CK_3	SDA_5	Rx_3		PWM1_3
PT2.2	INT2.2	DIO	TC11_6		MISO_3	SCL_6	Tx2_3		PWM2_3
PT2.3	INT2.3	DIO	TC12_6	LVDOO	MOSI_3	SDA_6	Rx2_3		PWM3_3
PT2.4	INT2.4	DIO	TC11_7	LS_XOUT	CS_4	SCL_7	Tx_4		PWM0_4
PT2.5	INT2.5	DIO	TC12_7	LS_XIN	CK_4	SDA_7	Rx_4		PWM1_4
PT2.6	INT2.6	DIO	TC11_8	HS_XIN	MISO_4	SCL_8	Tx2_4		PWM2_4
PT2.7	INT2.7	DIO	TC12_8	HS_XOUT	MOSI_4	SDA_8	Rx2_4		PWM3_4
PT3.0	INT3.0	DIO		ECK					
PT3.1	INT3.1	DIO		EDIO					
PT3.2	INT3.2	DIOAI						AIO4	
PT3.3	INT3.3	DIOAI						AIO5	
PT3.4	INT3.4	DIOAI						AIO6	
PT3.5	INT3.5	DIOAI						AIO7	

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



Function	INT	GPIO	Timer C Capture	Special Function	SPI	I ² C	UART	Analog	Timer B/B2 PWM
Output Priority	I/P	I/P	I/P	0	1	2	3	4	5
PT3.6	INT3.6	DIOAIO						REFO	
PT3.7	INT3.7	DIOAI						AIO8/LVDIN	
AIO0		AI						AIO0	
AIO1		AI						AIO1	
AIO2		AI						AIO2	
AIO3		AI						AIO3	
PT13.0		DIOAO		COM 0					
PT13.1		DIOAO		COM 1					
PT13.2		DIOAO		COM 2					
PT13.3		DIOAO		COM 3					
PT13.4		DIOAO		COM 4/SEG 0					
PT13.5		DIOAO		COM 5/SEG 1					
PT13.6		DIOAO		COM 6/SEG 42					
PT13.7		DIOAO		COM 7/SEG 43					
PT6.0		DIOAO	TCI3_1	SEG 2	CS_5		Tx_5		PWM0_5
PT6.1		DIOAO		SEG 3	CK_5		Rx_5		PWM1_5
PT6.2		DIOAO	TCI3_2	SEG 4	MISO_5		Tx2_5		PWM2_5
PT6.3		DIOAO		SEG 5	MOSI_5		Rx2_5		PWM3_5
PT6.4		DIOAO		SEG 6					
PT6.5		DIOAO		SEG 7					
PT6.6		DIOAO		SEG 8					
PT6.7		DIOAO		SEG 9					
PT7.0		DIOAO		SEG 10					
PT7.1		DIOAO		SEG 11					
PT7.2		DIOAO		SEG 12					
PT7.3		DIOAO		SEG 13					
PT7.4		DIOAO	TCI3_3	SEG 14	CS_6		Tx_6		PWM0_6
PT7.5		DIOAO		SEG 15	CK_6		Rx_6		PWM1_6
PT7.6		DIOAO	TCI3_4	SEG 16	MISO_6		Tx2_6		PWM2_6
PT7.7		DIOAO		SEG 17	MOSI_6		Rx2_6		PWM3_6
PT8.0		DIOAO		SEG 18	CS_8		Tx_8		PWM0_8
PT8.1		DIOAO		SEG 19	CK_8		Rx_8		PWM1_8
PT8.2		DIOAO		SEG 20	MISO_8		Tx2_8		PWM2_8
PT8.3		DIOAO		SEG 21	MOSI_8		Rx2_8		PWM3_8
PT8.4		DIOAO		SEG 22					

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



Function	INT	GPIO	Timer C Capture	Special Function	SPI	I ² C	UART	Analog	Timer B/B2 PWM
Output Priority	I/P	I/P	I/P	0	1	2	3	4	5
PT8.5		DIOAO		SEG 23					
PT8.6		DIOAO		SEG 24					
PT8.7		DIOAO		SEG 25					
PT9.0		DIOAO	TCI3_5	SEG 26	CS_7		Tx_7		PWM0_7
PT9.1		DIOAO		SEG 27	CK_7		Rx_7		PWM1_7
PT9.2		DIOAO	TCI3_6	SEG 28	MISO_7		Tx2_7		PWM2_7
PT9.3		DIOAO		SEG 29	MOSI_7		Rx2_7		PWM3_7
PT9.4		DIOAO		SEG 30					
PT9.5		DIOAO		SEG 31					
PT9.6		DIOAO		SEG 32					
PT9.7		DIOAO		SEG 33					
PT10.0		DIOAO		SEG 34					
PT10.1		DIOAO		SEG 35					
PT10.2		DIOAO		SEG 36					
PT10.3		DIOAO		SEG 37					
PT10.4		DIOAO	TCI3_7	SEG 38					
PT10.5		DIOAO		SEG 39					
PT10.6		DIOAO	TCI3_8	SEG 40					
PT10.7		DIOAO		SEG 41					

表 22-1 IO 引脚复用功能分布及优先级别

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



22.2. 寄存器地址

GPIO Register Address	31	24	23	16	15	8	7	0
GPIO Base Address + 0x40(0x40840)	MASK1		GPIOMCR1		MASK0		GPIOMCR0	
GPIO Base Address + 0x44(0x40844)	MASK3		GPIOMCR3		MASK2		GPIOMCR2	
GPIO Base Address + 0x48(0x40848)	MASK5		GPIOMCR5		MASK4		GPIOMCR4	
GPIO Base Address + 0x4C(0x4084C)	MASK7		GPIOMCR7		MASK6		GPIOMCR6	

22.3. 寄存器功能

22.3.1. GPIO 复用功能控制寄存器 0

GPIO Base Address + 0x40 (0x40840)							
Symbol	GPIOMCR0/ GPIOMCR1 (GPIO multiplex Control Register 0)						
Bit	[31:24]	[23:21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	-	-	-	-	Rsv
RW	R0W-0	-	-	-	-	-	RW-0
Bit	[15:08]	[07:05]	[04:02]		[01]		[00]
名称	MASK	PTCTC	PTPW		PTPW1E		PTPW0E
RW	R0W-0	RW-0					

位	名称	描述																																				
Bit[7:5]	PTCTC	捕捉比较器信号输入端 IO 选择																																				
		<table border="1"> <thead> <tr> <th>PTCTC[2:0]</th> <th>TCI1</th> <th>TCI2</th> <th>TCI3</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>PT1.0</td> <td>PT1.1</td> <td>PT6.0</td> </tr> <tr> <td>001</td> <td>PT1.2</td> <td>PT1.3</td> <td>PT6.2</td> </tr> <tr> <td>010</td> <td>PT1.4</td> <td>PT1.5</td> <td>PT7.4</td> </tr> <tr> <td>011</td> <td>PT1.6</td> <td>PT1.7</td> <td>PT7.6</td> </tr> <tr> <td>100</td> <td>PT2.0</td> <td>PT2.1</td> <td>PT9.0</td> </tr> <tr> <td>101</td> <td>PT2.2</td> <td>PT2.3</td> <td>PT9.2</td> </tr> <tr> <td>110</td> <td>PT2.4</td> <td>PT2.5</td> <td>PT10.4</td> </tr> <tr> <td>111</td> <td>PT2.6</td> <td>PT2.7</td> <td>PT10.6</td> </tr> </tbody> </table>	PTCTC[2:0]	TCI1	TCI2	TCI3	000	PT1.0	PT1.1	PT6.0	001	PT1.2	PT1.3	PT6.2	010	PT1.4	PT1.5	PT7.4	011	PT1.6	PT1.7	PT7.6	100	PT2.0	PT2.1	PT9.0	101	PT2.2	PT2.3	PT9.2	110	PT2.4	PT2.5	PT10.4	111	PT2.6	PT2.7	PT10.6
		PTCTC[2:0]	TCI1	TCI2	TCI3																																	
		000	PT1.0	PT1.1	PT6.0																																	
		001	PT1.2	PT1.3	PT6.2																																	
		010	PT1.4	PT1.5	PT7.4																																	
		011	PT1.6	PT1.7	PT7.6																																	
		100	PT2.0	PT2.1	PT9.0																																	
		101	PT2.2	PT2.3	PT9.2																																	
110	PT2.4	PT2.5	PT10.4																																			
111	PT2.6	PT2.7	PT10.6																																			
Bit[4:2]	PTPW	PWM 输出端 IO 选择																																				
		<table border="1"> <thead> <tr> <th>PTPW[2:0]</th> <th>PWM0</th> <th>PWM1</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>PT1.0</td> <td>PT1.1</td> </tr> <tr> <td>001</td> <td>PT1.4</td> <td>PT1.5</td> </tr> <tr> <td>010</td> <td>PT2.0</td> <td>PT2.1</td> </tr> <tr> <td>011</td> <td>PT2.4</td> <td>PT2.5</td> </tr> <tr> <td>100</td> <td>PT6.0</td> <td>PT6.1</td> </tr> <tr> <td>101</td> <td>PT7.4</td> <td>PT7.5</td> </tr> <tr> <td>110</td> <td>PT9.0</td> <td>PT9.1</td> </tr> <tr> <td>111</td> <td>PT8.0</td> <td>PT8.1</td> </tr> </tbody> </table>	PTPW[2:0]	PWM0	PWM1	000	PT1.0	PT1.1	001	PT1.4	PT1.5	010	PT2.0	PT2.1	011	PT2.4	PT2.5	100	PT6.0	PT6.1	101	PT7.4	PT7.5	110	PT9.0	PT9.1	111	PT8.0	PT8.1									
		PTPW[2:0]	PWM0	PWM1																																		
		000	PT1.0	PT1.1																																		
		001	PT1.4	PT1.5																																		
		010	PT2.0	PT2.1																																		
		011	PT2.4	PT2.5																																		
		100	PT6.0	PT6.1																																		
		101	PT7.4	PT7.5																																		
110	PT9.0	PT9.1																																				
111	PT8.0	PT8.1																																				
Bit[01]	PTPW1E	PWM 1 IO 输出开启控制																																				
		<table border="1"> <tbody> <tr> <td>0</td> <td>关闭(IO 没有输出)</td> </tr> <tr> <td>1</td> <td>开启(输出由 PTPW 设置)</td> </tr> </tbody> </table>	0	关闭(IO 没有输出)	1	开启(输出由 PTPW 设置)																																
0	关闭(IO 没有输出)																																					
1	开启(输出由 PTPW 设置)																																					
Bit[00]	PTPW0E	PWM 0 IO 输出开启控制																																				

HY16F3910 User's Guide

21-bit ENOB $\Sigma\Delta$ ADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



		0	关闭(IO 没有输出)
		1	开启(输出有 PTPW 设置)

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



22.3.2. GPIO 复用功能控制寄存器 1

GPIO Base Address + 0x44 (0x40844)				
Symbol	GPIOMCR2/ GPIOMCR3 (GPIO Multiplex Control Register 1)			
Bit	[31:24]	[23:20]	[19:17]	[16]
名称	MASK	-	I2CPTS	I2CPTEn
RW	R0W-0	-	RW-0	
Bit	[15:08]	[07:05]	[04]	[03:01]
名称	MASK	PTCSP	PTSPE	PTUR
RW	R0W-0	RW-0		

位	名称	描述																																													
Bit[19:17]	I2CPTS	I ² C 通信 IO 选择																																													
		<table border="1"> <thead> <tr> <th>I2CPTS[2:0]</th> <th>SCL</th> <th>SDA</th> </tr> </thead> <tbody> <tr><td>000</td><td>PT1.0</td><td>PT1.1</td></tr> <tr><td>001</td><td>PT1.2</td><td>PT1.3</td></tr> <tr><td>010</td><td>PT1.4</td><td>PT1.5</td></tr> <tr><td>011</td><td>PT1.6</td><td>PT1.7</td></tr> <tr><td>100</td><td>PT2.0</td><td>PT2.1</td></tr> <tr><td>101</td><td>PT2.2</td><td>PT2.3</td></tr> <tr><td>110</td><td>PT2.4</td><td>PT2.5</td></tr> <tr><td>111</td><td>PT2.6</td><td>PT2.7</td></tr> </tbody> </table>	I2CPTS[2:0]	SCL	SDA	000	PT1.0	PT1.1	001	PT1.2	PT1.3	010	PT1.4	PT1.5	011	PT1.6	PT1.7	100	PT2.0	PT2.1	101	PT2.2	PT2.3	110	PT2.4	PT2.5	111	PT2.6	PT2.7																		
		I2CPTS[2:0]	SCL	SDA																																											
		000	PT1.0	PT1.1																																											
		001	PT1.2	PT1.3																																											
		010	PT1.4	PT1.5																																											
		011	PT1.6	PT1.7																																											
		100	PT2.0	PT2.1																																											
		101	PT2.2	PT2.3																																											
110	PT2.4	PT2.5																																													
111	PT2.6	PT2.7																																													
Bit[16]	I2CPTEn	I ² C 通信 IO 复用功能开启控制																																													
		0 关闭 (无信号输出)																																													
		1 开启 (IO 复用为 I ² C 通信引脚, IO 由 I2CPTS 设置)																																													
Bit[7:5]	PTCSP	SPI 通信 IO 选择																																													
		<table border="1"> <thead> <tr> <th>PTCSP[2:0]</th> <th>CS</th> <th>CK</th> <th>MISO</th> <th>MOSI</th> </tr> </thead> <tbody> <tr><td>000</td><td>PT1.0</td><td>PT1.1</td><td>PT1.2</td><td>PT1.3</td></tr> <tr><td>001</td><td>PT1.4</td><td>PT1.5</td><td>PT1.6</td><td>PT1.7</td></tr> <tr><td>010</td><td>PT2.0</td><td>PT2.1</td><td>PT2.2</td><td>PT2.3</td></tr> <tr><td>011</td><td>PT2.4</td><td>PT2.5</td><td>PT2.6</td><td>PT2.7</td></tr> <tr><td>100</td><td>PT6.0</td><td>PT6.1</td><td>PT6.2</td><td>PT6.3</td></tr> <tr><td>101</td><td>PT7.4</td><td>PT7.5</td><td>PT7.6</td><td>PT7.7</td></tr> <tr><td>110</td><td>PT9.0</td><td>PT9.1</td><td>PT9.2</td><td>PT9.3</td></tr> <tr><td>111</td><td>PT8.0</td><td>PT8.1</td><td>PT8.2</td><td>PT8.3</td></tr> </tbody> </table>	PTCSP[2:0]	CS	CK	MISO	MOSI	000	PT1.0	PT1.1	PT1.2	PT1.3	001	PT1.4	PT1.5	PT1.6	PT1.7	010	PT2.0	PT2.1	PT2.2	PT2.3	011	PT2.4	PT2.5	PT2.6	PT2.7	100	PT6.0	PT6.1	PT6.2	PT6.3	101	PT7.4	PT7.5	PT7.6	PT7.7	110	PT9.0	PT9.1	PT9.2	PT9.3	111	PT8.0	PT8.1	PT8.2	PT8.3
		PTCSP[2:0]	CS	CK	MISO	MOSI																																									
		000	PT1.0	PT1.1	PT1.2	PT1.3																																									
		001	PT1.4	PT1.5	PT1.6	PT1.7																																									
		010	PT2.0	PT2.1	PT2.2	PT2.3																																									
		011	PT2.4	PT2.5	PT2.6	PT2.7																																									
		100	PT6.0	PT6.1	PT6.2	PT6.3																																									
		101	PT7.4	PT7.5	PT7.6	PT7.7																																									
110	PT9.0	PT9.1	PT9.2	PT9.3																																											
111	PT8.0	PT8.1	PT8.2	PT8.3																																											
MISO: Master input mode, Slave output mode. MOSI: Master output mode, Slave input mode.																																															
Bit[04]	PTSPE	SPI 通信 IO 复用功能开启控制																																													
		0 关闭(只作为普通 IO)																																													
		1 开启(IO 复用为 SPI 通信引脚, 通信 IO 由 PTCSP 设置)																																													
Bit[3:1]	PTUR	UART 通信 IO 选择																																													
		<table border="1"> <thead> <tr> <th>PTUR[2:0]</th> <th>TX</th> <th>RX</th> </tr> </thead> <tbody> <tr><td>000</td><td>PT1.0</td><td>PT1.1</td></tr> <tr><td>001</td><td>PT1.4</td><td>PT1.5</td></tr> <tr><td>010</td><td>PT2.0</td><td>PT2.1</td></tr> <tr><td>011</td><td>PT2.4</td><td>PT2.5</td></tr> <tr><td>100</td><td>PT6.0</td><td>PT6.1</td></tr> <tr><td>101</td><td>PT7.4</td><td>PT7.5</td></tr> <tr><td>110</td><td>PT9.0</td><td>PT9.1</td></tr> <tr><td>111</td><td>PT8.0</td><td>PT8.1</td></tr> </tbody> </table>	PTUR[2:0]	TX	RX	000	PT1.0	PT1.1	001	PT1.4	PT1.5	010	PT2.0	PT2.1	011	PT2.4	PT2.5	100	PT6.0	PT6.1	101	PT7.4	PT7.5	110	PT9.0	PT9.1	111	PT8.0	PT8.1																		
		PTUR[2:0]	TX	RX																																											
		000	PT1.0	PT1.1																																											
		001	PT1.4	PT1.5																																											
		010	PT2.0	PT2.1																																											
		011	PT2.4	PT2.5																																											
		100	PT6.0	PT6.1																																											
		101	PT7.4	PT7.5																																											
110	PT9.0	PT9.1																																													
111	PT8.0	PT8.1																																													

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



Bit[00]	PTURE	EUART 通信 IO 复用功能开启控制	
		0	关闭(只作为普通 IO)
		1	开启(IO 引脚复用为 UART 通信引脚, 通信 IO 由 PTUR 设置)

22.3.3. GPIO 复用功能控制寄存器 2

GPIO Base Address + 0x48 (0x40848)						
Symbol	GPIOMCR4/GPIOMCR5 (GPIO Multiplex Control Register 2)					
Bit	[31:16]					
名称	-					
RW	-					
Bit	[15:08]	[7:6]	[5]	[4:2]	[1]	[0]
名称	MASK	-	PTCI3E	PTPW2	PTPW3E	PTPW2E
RW	R0W-0	-			RW-0	

位	名称	描述		
Bit[05]	PTCI3E	TCI3 模式控制		
		0	TCI3 与 TCI1 相同	
		1	TCI3 配置透过 PTCTC	
Bit[04:02]	PTPW2	PWM 输出 IO 选择		
		PTPW2[2:0]	PWM2	PWM3
		000	PT1.2	PT1.3
		001	PT1.6	PT1.7
		010	PT2.2	PT2.3
		011	PT2.6	PT2.7
		100	PT6.2	PT6.3
		101	PT7.6	PT7.7
		110	PT9.2	PT9.3
111	PT8.2	PT8.3		
Bit[01]	PTPW3E	GPIO PWM3 控制开关		
		0	关闭	
		1	开启	
Bit[00]	PTPW2E	GPIO PWM2 控制开关		
		0	关闭	
		1	开启	

22.3.4. GPIO 复用功能控制寄存器 3

GPIO Base Address + 0x4C (0x4084C)				
Symbol	GPIOMCR6/GPIOMCR7 (GPIO Multiplex Control Register 3)			
Bit	[31:16]			
名称	-			
RW	-			
Bit	[15:08]	[7:4]	[3:1]	[0]
名称	MASK	-	PTUR2	PTUR2E
RW	R0W-0	-		RW-0

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述		
Bit[3:1]	PTUR2	UART2 通信 IO 选择		
		PTUR2[2:0]	TX2	RX2
		000	PT1.2	PT1.3
		001	PT1.6	PT1.7
		010	PT2.2	PT2.3
		011	PT2.6	PT2.7
		100	PT6.2	PT6.3
		101	PT7.6	PT7.7
		110	PT9.2	PT9.3
111	PT8.2	PT8.3		
Bit[0]	PTUR2E	GPIO UART2 控制开关		
		0	关闭	
		1	开启	

注意事项：

- PTSPE 与 PTCSP 相关之 SPI I/O Port 具有最高优先权，当相关 I/O Port 被选择为 SPI 用途后，其它 IP 与 GPIO 设定均为无效。
- I2CPTEn 与 I2CPTS 相关之 I²C I/O Port 具有次高优先权，当相关 I/O Port 被选择为 I²C 用途后，除了 SPI 之其它 IP 与 GPIO 设定均为无效。
- 只限定 I/O Port 的输出部分，UART 为第三优先权，ADC 为第四优先权，PWM 为第五优先权，GPIO 为最低优先权。

23. $\Sigma\Delta$ 24 位模拟数字转换器 ADC

23.1. 整体总说明

芯片带有一个嵌入式的高效 Sigma Delta 24 位模拟数字转换器(24-bit $\Sigma\Delta$ ADC)。ADC 前置一个低噪声, 可编程增益放大器(Low Noise PGA), 被用来放大输入信号。ADC 增益设定范围为 1 ~ 128。ADC 的采样频率可通过寄存器编程设定, 所设计的采样频率建议最高为 1MHz。它有一个三阶调节器用于过滤调节器的量子化噪声。ADC 的过采样率可编程范围是 64~32768。它是设计用来测量输出信号非常小的感应器, 例如应变计、压力表、和工业处理控制。注意, 虽然 ADC 采样率可以藉由设置 HS_CK 频率源来做更高的采样率选择, ADC 频率源为 HS_CK;如 HAO 在 4.147MHz 与除频器 4 为规格书中 ENOB 规格之测试条件, 较高的 ADC 频率可能造成低于规格的结果, 建议除频后, 采样率为 250kHz~1MHz 左右。

特性:

可设置采样频率 250kHz~1MHz

分辨率高达 21 位的有效位数(ENOB);

最低输入噪音为 65nV RMS;

可设置超采样频率为 64~32768;

最高输出率为 15kHz;

内建低噪音可编程增益放大器, 增益倍数为 1~128;

内建温度感应器;

内建 4 位 DAC 来调整偏移;

三阶梳状滤波器;

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

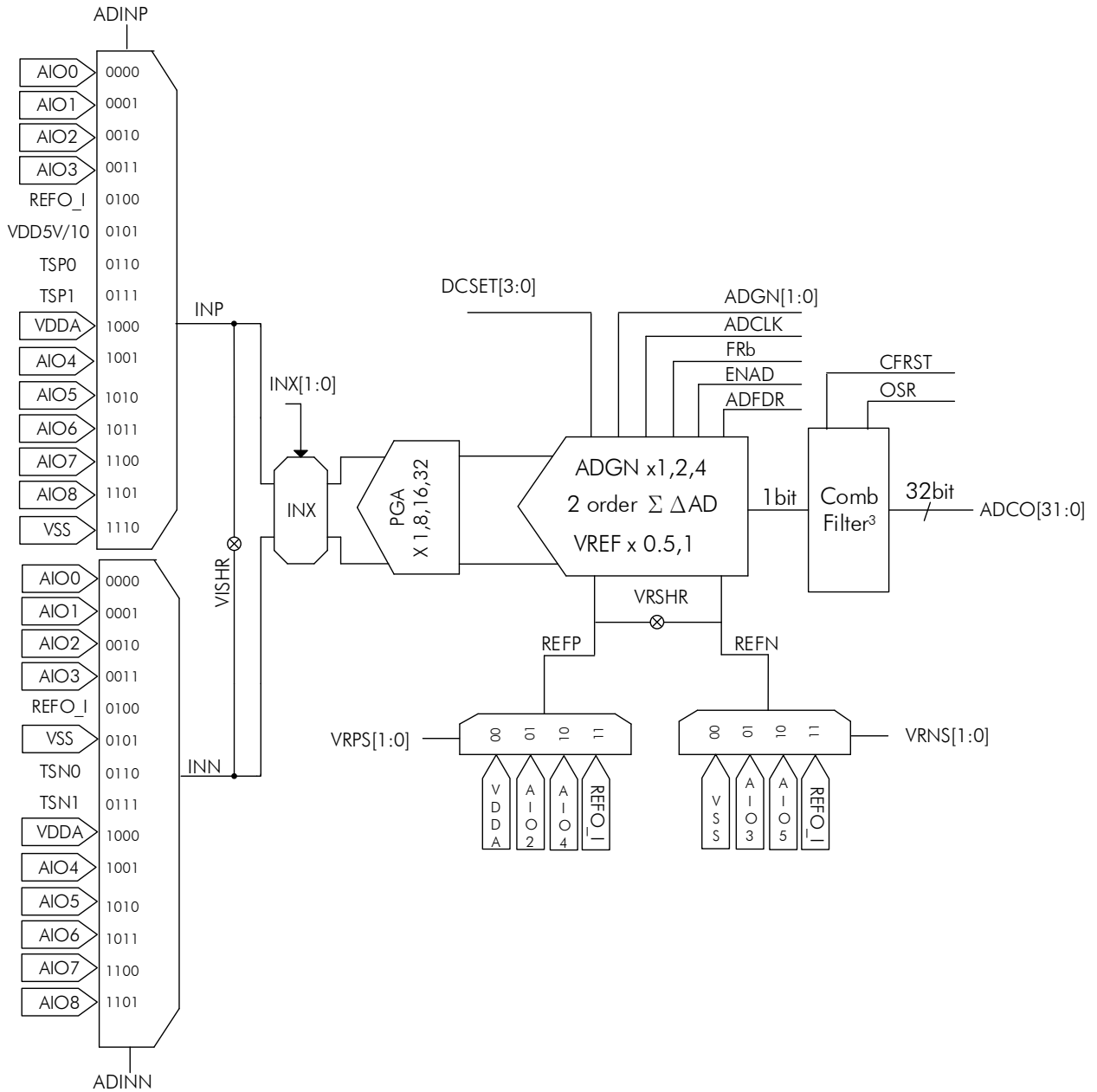


图 23-1 ADC 功能方框图

23.1.1. 全差动信号输入端

ADC 的输入信号为全差动输入模式，即输入端由正向输入端和负向输入端构成。正向与负向的信号输入通道都包含多个内部信号输入通道，在 ADC 取样频率 1MHz，ADC 放大倍率为一倍的情况下，ADC 信号输入端的输入阻抗约为 100K。透过控制器 ADINP[3:0]、ADINN[3:0]选择正向、负向的信号输入通道，但是正向输入端在同一时间只能选择一路信号输入通道，负向输入端在同一时间只能选择一路信号输入通道。正向与负向可选同样的输入通道，这样差分信号值接近零，只残留 Offset 偏移量。下图列出正向端与负向端的信号输入通道。

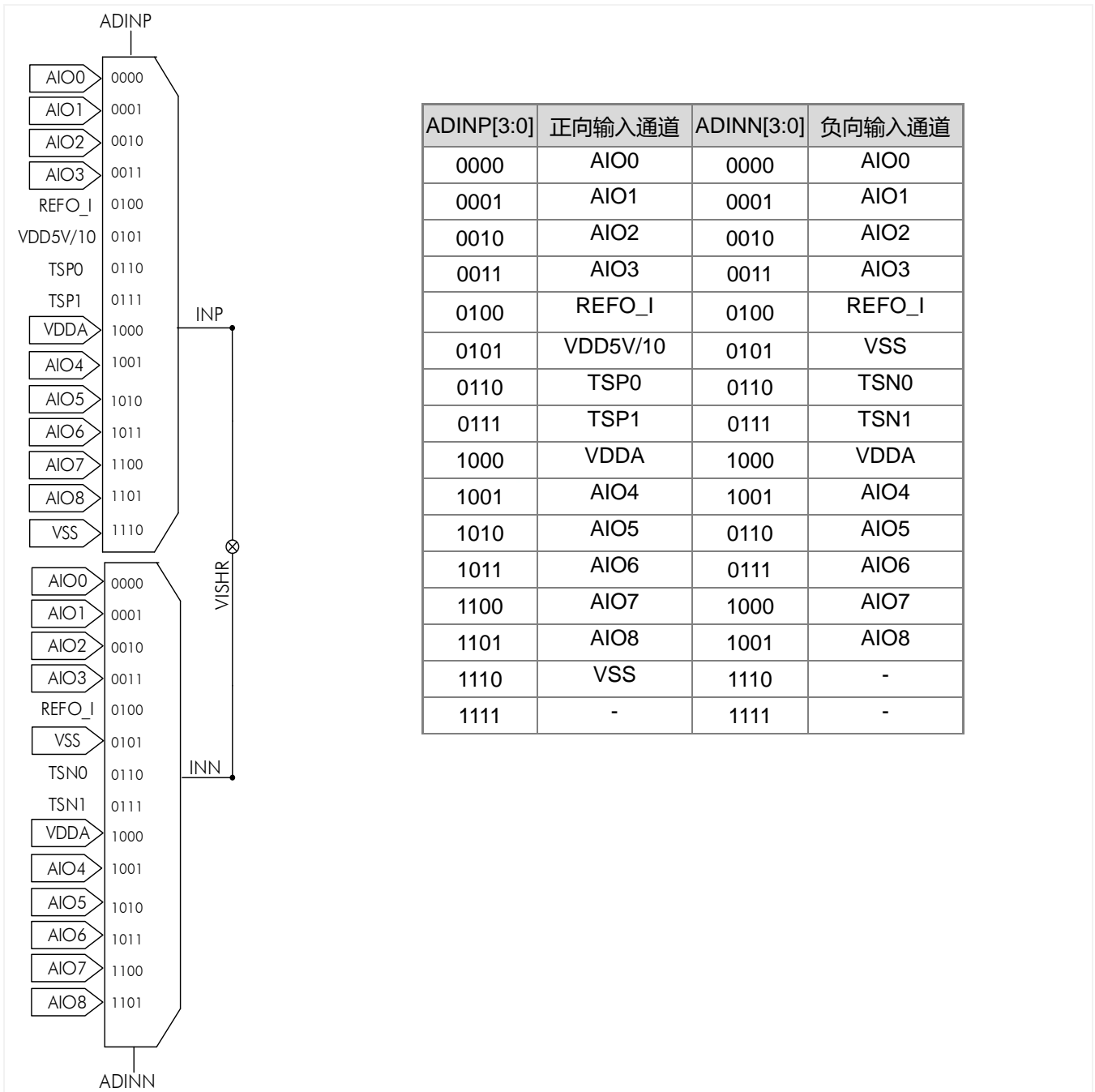


图 23-2 ADC 信号输入通道

输入信号经由内部增大放大再进行转换，所以对于输入信号的电压范围亦有限制，为了 ADC 输出能够得到较高的分辨率及线性度，建议输入信号的差分电压值 $\Delta SI = \pm 0.9 * \Delta VREF$ ($\Delta SI = INP - INN$)。输入信号电压如下表所示。

外部输入通道	电压输入范围
ADINP	$VSS - 0.2V \leq INP \leq VDDA$
ADINN	$VSS - 0.2V \leq INN \leq VDDA$

表 23-1 输入信号电压范围表

23.1.2. 内置增益放大器

ADC 内置两个增益放大器：一个低噪声、低温度系数的可编程增益放大器 PGA，放大倍数为 8/16/32；一个是可编程增益放大器 Σ AD，放大倍数为 1、2、4。因此两个增益放大器组合使用内部最大放大倍数为 128。但是放大倍数与 ADC 输出值有效位数(ENOB)成反比列的，放大倍数越大，ENOB 的值越小。所以在设置放大倍数需要根据实际需要来配置。透过控制器 PGA[2:0]可选择 PGA 增益器的放大倍率，PGA 的放大倍率选择如下表；透过控制器 ADGN[1:0]可选择 ADC Modulator 的增益倍率，ADC Modulator 放大倍率选择如下表所示

C type PGA				
PGA[2:0]	000	001	011	111
放大倍率	X1	X 8	X16	X32

ADC Modulator				
ADGN[1:0]	00	01	10	11
放大倍率	X1	X2	-	X4

表 23-2 内部增益放大倍率

23.1.3. 参考电压输入通道

ADC 参考电压输入属于全差动输入模式，即参考电压输入端由正向输入端与负向输入端构成。正向与负向输入端都包含 2 个外部输入通道和 2 个内部输入通道。透过控制器 VRPS[1:0]、VRNS[1:0]可分别设置参考电压的正向输入通道、负向输入通道。正向输入端在同一时间只能选择一路输入通道，负向输入端在同一时间只能选择一路输入通道。

参考电压由 VREFP 与 VREFN 输入后产生的 $\Delta VREF$ 电压差，在经过可编程参考电压衰减器后作为 ADC 的参考电压值。控制器 FRb[0]可设置参考电压衰减倍率，参考电压衰减倍率如下表所示。

参考电压计算如下：

$$\Delta VREF = VREFP - VREFN \quad (\text{式 23-1})$$

$$VREF = \text{Gain} \times \Delta VREF \quad (\text{式 23-2})$$

$\Delta VREF$: 参考电压的电压差值； $VREF$: ADC 内部参考电压值

$VREFP/VREFN$: 输入参考电压值

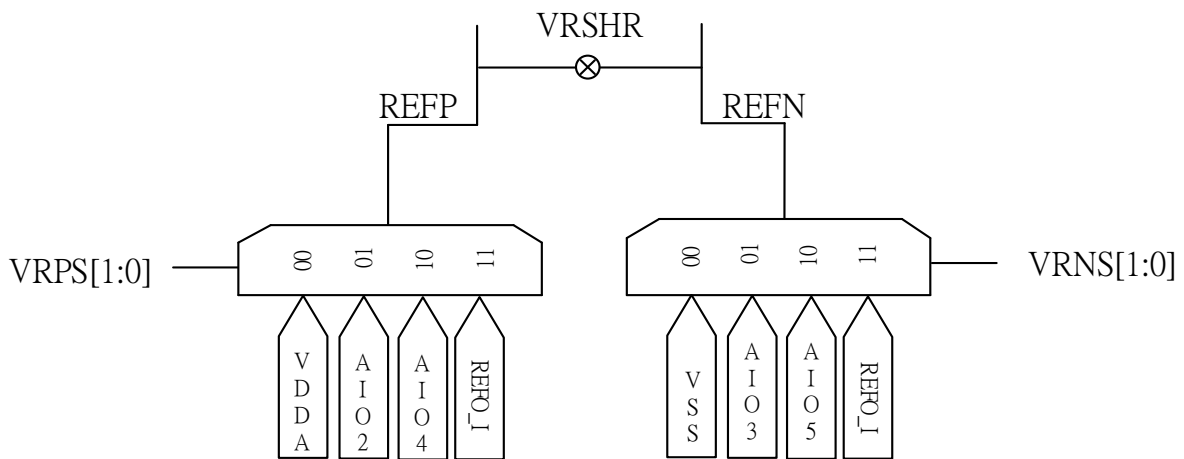


图 23-3 参考电压输入通道

参考电压衰减倍率		
FRb[0]	0	1
Gain	1	1/2

表 23-3 参考电压衰减倍率

参考电压正负输入通道的输入阻抗为 500kΩ，且 $VREFP$ 或 $VREFN$ 的输入电压不可小于 VSS ，也不可超过 $VDDA$ ；透过控制器设置为外部输入通道，可增加输入阻抗，但也许注意外部输入通道的电压值范围。为了是 ADC 输出得到较高的分辨率及线性度，估建议参考电压 $\Delta VREF = 0.8V \sim 1.8V$ 。

外部输入通道	电压输入范围
AIO2 / AIO4	$VREFN \leq VREFP \leq VDDA$
AIO3 / AIO5	$VSS \leq VREFN \leq VREFP$

表 23-4 参考电压外部输入信道电压输入范围

23.1.4. 输入信号输入偏压

ADC 具有零点偏压平移控制器，零点偏压平移控制器 $DCSET[3:0]$ 透过改变输入信号零点的位置，以避免因为输入信号电压过大而导致超出最大测量范围溢出。待测信号经过 ADC Modulator 增益放大及零点偏压平移调整后，等效的待测信号 ΔSI_I 的计算公式如下：

$$\Delta SI_I = ADGN \times \Delta SI_{\pm} + (DCSET \times \Delta VREF) \quad (\text{式 23-3})$$

设置值	DCSET[3:0]							
	0000	0001	0010	0011	0100	0101	0110	0111
平移量	0*VREF	+1/8* VREF	+1/4* VREF	+3/8* VREF	+1/2* VREF	+5/8* VREF	+3/4* VREF	+7/8* VREF

HY16F3910 User's Guide

21-bit ENOB $\Sigma\Delta$ ADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



设置值	1000	1001	1010	1011	1100	1101	1110	1111
平移量	0*VREF	-1/8*VREF	-1/4*VREF	-3/8*VREF	-1/2*VREF	-5/8*VREF	-3/4*VREF	-7/8*VREF

表 23-5 待测输入信号零点偏压设置对照表

23.1.5. 梳状滤波器

$\Sigma\Delta$ ADC 采用三阶的梳状滤波器设置,透过控制器 OSR[3:0]的设置及与 ADC 的采样频率组合,可得到不同的过采样频率,实现不同的 ADC 转换值输出频率。OSR[3:0]设置参数如表所示。

OSR[3:0]											
设置值	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010
除频值	32768	16384	8192	4096	2048	1024	512	256	128	64	Rsv

表 23-6 过采样频率除频表

模拟数字转换结果存放于寄存器 ADCO[23:0],最高位为符号位,所以转换结果与输入信号的关系如表 23-7 所示。

	等效待测信号	ADCO[23:0]	
		十六进制	二进制
两极性输出 二补码格式	ΔVR	7F FF FF	0111-1111 1111-1111 1111-1111
	$\Delta VR \times \frac{1}{2^{23}}$	00 00 01	0000-0000 0000-0000 0000-0001
		00 00 00	0000-0000 0000-0000 0000-0000
	$-\Delta VR \times \frac{1}{2^{23}}$	FF FF FF	1111-1111 1111-1111 1111-1111
	ΔVR	80 00 00	1000-0000 0000-0000 0000-0000

表 23-7 ADCO[23:0]与输入信号关系表

梳状滤波器提供复位控制功能,当控制位 CFRST 被置<0>时 梳状滤波器进行复位 接着设置 CFRST=<1>,启动梳状滤波器,这样 $\Sigma\Delta$ ADC 就会自动丢弃前 2 笔数据,用户等待中断发生时,读取到的第一笔 ADC 转换值就为有效的 ADC 值。

23.1.6. 温度传感器 TPS

温度传感器由二极管(BJT)组成,其电压信号对温度的变化为已通过 0K 曲线,其具有以下特色:
 温度传感器在环境温度为 0K 时其输出的电压值 VTPS@0K =0V ;
 透过测量方式可使得模拟数字转换器 ADC 的偏移电压(VADC-OFFSET)与 BJT 之不对称性自动抵消;
 校正温度仅需单点校正即可满足 $\pm 2^{\circ}\text{C}$ 误差;

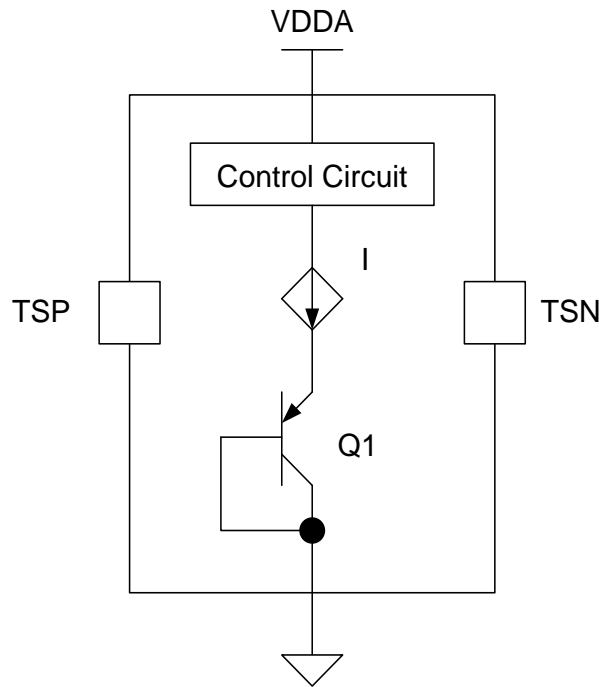


图 23-8 温度传感器应用方块图

TPS 初始化设置与计算方式如下操作：

启用 ADC 则 TPS 的功能随即被自动启用。

固定 ADC 与系统工作频率相关设置，TPS 校正时设置与 TPS 测量时的设置需相同。

在同一温度 $T_a(^{\circ}\text{C})$ 下，测量得 ADCTPS0 与 ADCTPS1 的数值后，将两数先相减再除 2 即可求得在温度 T_a 下测得 TPS 相对应的电压值 $V_{TS@T_a}$ 。

测量 ADCTPS0 时，ADINP[3:0]设置<0110>且 ADINN[3:0]设置<0110>

测量 ADCTPS1 时，ADINP[3:0]设置<0111>且 ADINN[3:0]设置<0111>

将 ADCTPS0 的数值与 ADCTPS1 数值先相减再除 2 即可得到 $ADCTPS@T_A$

TPS 的输出值 V_{TPS} 对温度变化为一线性曲线，故可推导出其增益值 G_{TPS} (或称斜率)。

$$G_{TPS} = \frac{ADC_{TPS@T_A}}{(273.15 + T_{offset} + T_A)K} \dots\dots\dots (式 23-4)$$

G_{TPS} : 温度计算斜率 $\frac{ADC \text{ count}}{K}$

$ADC_{TPS@T_A}$: 校正温度下所测得的 ADC 值

K : $^{\circ}\text{C}+273.15$

T_{offset} : 由于 TPS 在温度单位转换上的不理想，故会有一偏差值

TPS 在温度转换上的不理想，因此实际上并非于 $^{\circ}\text{C}=\text{K}-273.15$

而是 $^{\circ}\text{C}=\text{K}+\text{KT}=\text{K}+(-273.15-\text{Toffset})$

其中的 KT 值请参考 IC Data sheet ADC 章节内 TPS 规格。

HY16F3910 KT 值为 -283, $^{\circ}\text{C}=\text{K}-283$, $\text{K}=\text{^{\circ}\text{C}}+283$

TPS 范例说明

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

假设将于 25°C 进行 TPS 的校正。校正后将 IC 移动置一较高温环境(65°C)，测试该环境下的温度。

(1) ADINP[3:0]设置<0110>且 ADINN[3:0]设置<0110>，ADC 量测得到一个数字码 ADCTPS0=5897634。

(2) ADINP[3:0]设置<0111>且 ADINN[3:0]设置<0111>，ADC 量测得到一个数字码 ADCTPS1=-5827679。

(3) 计算 $ADCTPS@25=(ADCTPS0 - ADCTPS1)/2=5862656$ 。此动作可消除 Temperature Sensor 的 Offset。

(4) 计算 G_{TPS} ：

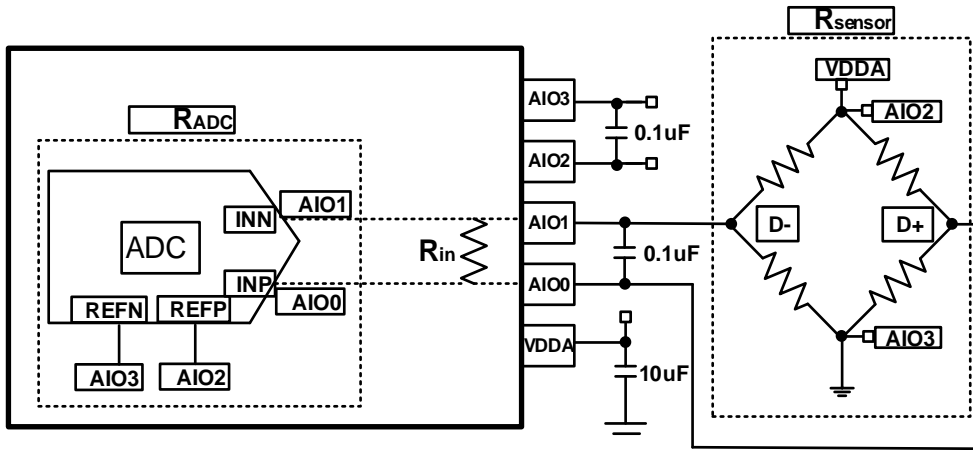
$$G_{TPS} = \frac{ADC_{TPS@T_A}}{(273.15 + T_{offset} + T_A)K} = \frac{5862656}{(283 + 25)K} = 19034.60$$

(5) 将 IC 移置高温环境(65°C)后一段时间后，参考步骤(1)~(3) 再次测得 $ADCTPS@65 : 6630103$

$$T_x = \frac{ADC_{TPS@65}}{G_{TPS}} - [273.15 + T_{offset}] = \frac{6630103}{19034.60} - 283 = 65.32^\circ C$$

23.1.7. ADC 输入阻抗(R_{ADC})说明

下图为 ADC 输入阻抗(R_{ADC})与 Sensor 输出阻抗(R_{sensor})与实际进入到芯片的输入阻抗(R_{in})的示意图. 使用者可以依照 Sensor 特性来评估是否可以直接把 Sensor 与 ADC 输入通道连接, 避免产生量测上的阻抗效应.



(R_{in})与(R_{ADC})与(R_{sensor})的关系式子为: $R_{in} = R_{sensor} // R_{ADC}$

(R_{in}): R_{in} 为 R_{sensor} 并联 R_{ADC}

(R_{ADC}):代表 ADC 输入阻抗

(R_{sensor}):代表 Sensor 输出阻抗

注意 : (R_{ADC})不等于实际 HY16F ADC 可连接的 Sensor 最大输出阻抗. 在 PGA 与 ADGN=1 倍时候, $R_{ADC}=830k\Omega$, 但该数值不同可连接 Sensor 的最大输出阻抗数值(R_{sensor}). 一般参考建议在 PGA 与 ADGN=1 倍时候, 可接 Sensor 最大输出阻抗为 $100k\Omega$.

ADC 输入阻抗(R_{ADC})表

$R_{ADC}(\Omega) @ ADCK= 1MHz$			
PGA	ADGN=1	ADGN=2	ADGN=4
1	830k Ω	415k Ω	207k Ω
8	40k Ω	40k Ω	40k Ω
16	20k Ω	20k Ω	20k Ω
32	10k Ω	10k Ω	10k Ω

Sensor 输出阻抗(R_{sensor})表

$R_{sensor}(\Omega) @ ADCK= 1MHz$			
PGA	ADGN=1	ADGN=2	ADGN=4
1	100k Ω	50k Ω	25k Ω
8	5k Ω	5k Ω	5k Ω
16	2.5k Ω	2.5k Ω	2.5k Ω
32	2k Ω	2k Ω	2k Ω

23.1.8. ADC 操作说明

ADC 是 24 位分辨率的 Δ - Σ 架构。要启用 ADC 功能,就需要正确设定一些外围电路。ADC 的电源是 VDDA 电压。因此, VDDA 需要高于 2.4V。要获得较好的 ADC 效能就需有一个稳定的 VDDA 电源供应。因为 VDDA 需要一些时间来就绪,ADC 须等待 VDDA 就绪后才开始量测的动作。将 ENBGR 设为 <1> 来开启偏移和 BandGap 电压。然后需要一个 1.2V 共模电压以启动 ADC。这个共模电压电压可经由外部或内部选择。ADC 也需要有一个 ADCK 频率输入,这输入的频率应该被设定为 1MHz。

详细的 ADC 初始化配置操作如下:

- (1) 选择 ADC 待测信号输入通道,包括正向输入通道 ADINP 0x41104[7:4]、负向输入通道 ADINN 0x41104[3:0]。
- (2) 配置 ADC 内部增益放大倍率 ADGN 0x41104[18:16],根据实际情况设置,让 Δ SI 在 $0.9 \cdot V_{REF}$ 范围内。
- (3) 设置零点偏压 DCSET 0x41104[27:24],若不需要,请设置 $0 \cdot V_{REF}$ 。
- (4) 选择 ADC 参考电压输入信道 VRPS 0x41100[19:18] 与 VRNS 0x41100[17:16],并选择参考电压衰减率 FRb 0x41104[19]。
- (5) 设置 ADC 转换值的输出频率 OSR 0x41100[5:2],需要根据实际 ENOB 需要来设置。
- (6) 启动梳状滤波器, CFRST 0x41100[1]=<1>;设置该位硬件可以自动丢弃前 2 笔数据。
- (7) 配置并启动 ADC 工作频率源(寄存器 0x4030C[6:4]),建议将 ADC 采样频率设置在 1MHz 左右。
- (8) 开启 VDDA 电压 VDAS 0x40400[19:18]与设置 VDDA 稳压电压输入源 ENVA 0x40400[16]及 BandGap 参考电压 ENBGR 0x40400[4]=<1>,开启共模参考电压 ENRFO 0x40400[1]=<1>及模拟地来源 ACMS 0x40400[3]=<1>,并等待电压稳定时间。
- (9) 根据需要开启 ADC 中断功能 ADCIE 0x40008[16]=<1>,并使能全局中断 GIE=<1>。
- (10) 开启 ADC 功能 ENADC 0x41100[0]=<1>,等待第一次 ADC 中断信号发生时,即可取样 ADC 输出数据,读取寄存器 ADCO 0x41108[31:8]。

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



23.2. 寄存器地址

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)			ADO0				0x00	
ADC Base Address + 0x0C (0x4110C)			ADO1				0x00	
ADC Base Address + 0x10 (0x41110)	MASK5		REG5		MASK6		REG6	

23.3. 寄存器功能

23.3.1. ADC 寄存器 0

ADC Base Address + 0x00 (0x41100)						
Symbol	ADCCR0 (ADC Control Register 0)					
Bit	[31:24]	[23:22]	[21:20]		[19:18]	[17:16]
名称	MASK	-	-		VRPS	VRNS
RW	R0W-0	-	RW-0			
Bit	[15:08]	[7]	[6]	[5:2]	[1]	[0]
名称	MASK	ACMS	-	OSR	CFRST	ENADC
RW	R0W-0	RW-0				

位	名称	描述		
Bit[21:20]	-	保留请勿设定		
Bit[19:18]	VRPS	参考电压正向输入源选择		
		00 VDDA		
		01 AIO2		
		10 AIO4		
Bit[17:16]	VRNS	参考电压负向输入源选择		
		00 VSS		
		01 AIO3		
		10 AIO5		
Bit[7]	ACMS	ADC 模拟地输入源选择		
		0 ERFO_I		
		1 1.2V		
Bit[5:2]	OSR	ADC 过采样输出频率设置		
		OSR[3:0]	过采样率	数据输出频率(以 ADC 频率源为 1MHz 说明)
		0000	32768	31sps
		0001	16384	61sps
		0010	8192	122sps
		0011	4096	244sps
		0100	2048	488sps
		0101	1024	977sps
		0110	512	1953sps
		0111	276	3906sps
		1000	128	7813sps
1001	64	15625sps		

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述	
		1010	保留
		1011	保留
		1100	保留
		1101	保留
		1110	保留
		1111	保留
Bit[01]	CFRST	梳状滤波器开启控制	
		0	复位 (准位复位)
		1	开启复位功能, 完成复位后自动变为 0
Bit[00]	ENADC	ADC 开启控制	
		0	关闭
		1	开启

23.3.2. ADC 寄存器 1

ADC Base Address + 0x04 (0x41104)						
Symbol	ADCCR1 (ADC Control Register 1)					
Bit	[31:28]	[27:24]	[23:22]	[21:20]	[19]	[18:16]
名称	-	DCSET	INX	ADGN	FRb	PGA
RW	-	RW-0	RW-0	RW-0		
Bit	[15:08]	[07:04]		[03:00]		
名称	MASK	ADINP		ADINN		
RW	R0W-0	RW-0				

位	名称	描述			
Bit[27:24]	DCSET	DC 零点平移输入电压选择 (VREF = REFP-REFN)			
		DCSET[3:0]	Offset	DCSET[3:0]	Offset
		0000	0 VREF	1000	0 VREF
		0001	+1/8 VREF	1001	-1/8 VREF
		0010	+1/4 VREF	1010	-1/4 VREF
		0011	+3/8 VREF	1011	-3/8 VREF
		0100	+1/2 VREF	1100	-1/2 VREF
		0101	+5/8 VREF	1101	-5/8 VREF
		0110	+3/4 VREF	1110	-3/4 VREF
		0111	+7/8 VREF	1111	-7/8 VREF
Bit[23:22]	INX	SI±输入信号转置器			
		INX[1:0]	SI±输入信号		
		00	INP = ADH, INN = ADL		
		01	INP 浮接, INN = ADH & ADL		
		10	INP = ADH & ADL, INN 浮接		
		11	INP = ADL, INN = ADH		

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述																																				
																																						
Bit[21:20]	ADGN	ADC 输入信号放大倍数 ADC Gain 调整器设置 <table border="1"> <thead> <tr> <th>ADGN[1:0]</th> <th>Gain</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>Gain = 1</td> </tr> <tr> <td>01</td> <td>Gain = 2</td> </tr> <tr> <td>10</td> <td>保留</td> </tr> <tr> <td>11</td> <td>Gain = 4</td> </tr> </tbody> </table>	ADGN[1:0]	Gain	00	Gain = 1	01	Gain = 2	10	保留	11	Gain = 4																										
ADGN[1:0]	Gain																																					
00	Gain = 1																																					
01	Gain = 2																																					
10	保留																																					
11	Gain = 4																																					
Bit[19]	FRb	参考电压值范围选择 <table border="1"> <tbody> <tr> <td>0</td> <td>满量程参考电压输入, 即 VREF*1</td> </tr> <tr> <td>1</td> <td>1/2 倍参考电压输入, 即 VREF*1/2</td> </tr> </tbody> </table>	0	满量程参考电压输入, 即 VREF*1	1	1/2 倍参考电压输入, 即 VREF*1/2																																
0	满量程参考电压输入, 即 VREF*1																																					
1	1/2 倍参考电压输入, 即 VREF*1/2																																					
Bit[18:16]	PGA	ADC 输入信号放大倍数 PGA 调整器设置 <table border="1"> <thead> <tr> <th>PGA[2:0]</th> <th>Gain</th> <th>PGA[2:0]</th> <th>Gain</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>Gain = 1</td> <td>100</td> <td>保留, 不开放</td> </tr> <tr> <td>001</td> <td>Gain = 8</td> <td>101</td> <td>保留, 不开放</td> </tr> <tr> <td>010</td> <td>保留, 不开放</td> <td>110</td> <td>保留, 不开放</td> </tr> <tr> <td>011</td> <td>Gain = 16</td> <td>111</td> <td>Gain = 32</td> </tr> </tbody> </table>	PGA[2:0]	Gain	PGA[2:0]	Gain	000	Gain = 1	100	保留, 不开放	001	Gain = 8	101	保留, 不开放	010	保留, 不开放	110	保留, 不开放	011	Gain = 16	111	Gain = 32																
PGA[2:0]	Gain	PGA[2:0]	Gain																																			
000	Gain = 1	100	保留, 不开放																																			
001	Gain = 8	101	保留, 不开放																																			
010	保留, 不开放	110	保留, 不开放																																			
011	Gain = 16	111	Gain = 32																																			
Bit[7:4]	ADINP	ADC 正向信号输入端选择 <table border="1"> <thead> <tr> <th>ADINP[3:0]</th> <th>ADC 输入正通道</th> <th>ADINP[3:0]</th> <th>ADC 输入正通道</th> </tr> </thead> <tbody> <tr> <td>0000</td> <td>AIO0</td> <td>1000</td> <td>VDDA</td> </tr> <tr> <td>0001</td> <td>AIO1</td> <td>1001</td> <td>AIO4</td> </tr> <tr> <td>0010</td> <td>AIO2</td> <td>1010</td> <td>AIO5</td> </tr> <tr> <td>0011</td> <td>AIO3</td> <td>1011</td> <td>AIO6</td> </tr> <tr> <td>0100</td> <td>REFO_I</td> <td>1100</td> <td>AIO7</td> </tr> <tr> <td>0101</td> <td>VDD5V/10</td> <td>1101</td> <td>AIO8</td> </tr> <tr> <td>0110</td> <td>TSP0</td> <td>1110</td> <td>VSS</td> </tr> <tr> <td>0111</td> <td>TSP1</td> <td>1111</td> <td>-</td> </tr> </tbody> </table>	ADINP[3:0]	ADC 输入正通道	ADINP[3:0]	ADC 输入正通道	0000	AIO0	1000	VDDA	0001	AIO1	1001	AIO4	0010	AIO2	1010	AIO5	0011	AIO3	1011	AIO6	0100	REFO_I	1100	AIO7	0101	VDD5V/10	1101	AIO8	0110	TSP0	1110	VSS	0111	TSP1	1111	-
ADINP[3:0]	ADC 输入正通道	ADINP[3:0]	ADC 输入正通道																																			
0000	AIO0	1000	VDDA																																			
0001	AIO1	1001	AIO4																																			
0010	AIO2	1010	AIO5																																			
0011	AIO3	1011	AIO6																																			
0100	REFO_I	1100	AIO7																																			
0101	VDD5V/10	1101	AIO8																																			
0110	TSP0	1110	VSS																																			
0111	TSP1	1111	-																																			
Bit[3:0]	ADINN	ADC 负向信号输入端选择 <table border="1"> <thead> <tr> <th>ADINN[3:0]</th> <th>ADC 输入负通道</th> <th>ADINN[3:0]</th> <th>ADC 输入负通道</th> </tr> </thead> <tbody> <tr> <td>0000</td> <td>AIO0</td> <td>1000</td> <td>VDDA</td> </tr> <tr> <td>0001</td> <td>AIO1</td> <td>1001</td> <td>AIO4</td> </tr> <tr> <td>0010</td> <td>AIO2</td> <td>1010</td> <td>AIO5</td> </tr> <tr> <td>0011</td> <td>AIO3</td> <td>1011</td> <td>AIO6</td> </tr> <tr> <td>0100</td> <td>REFO_I</td> <td>1100</td> <td>AIO7</td> </tr> <tr> <td>0101</td> <td>VSS</td> <td>1101</td> <td>AIO8</td> </tr> <tr> <td>0110</td> <td>TSN0</td> <td>1110</td> <td>-</td> </tr> <tr> <td>0111</td> <td>TSN1</td> <td>1111</td> <td>-</td> </tr> </tbody> </table>	ADINN[3:0]	ADC 输入负通道	ADINN[3:0]	ADC 输入负通道	0000	AIO0	1000	VDDA	0001	AIO1	1001	AIO4	0010	AIO2	1010	AIO5	0011	AIO3	1011	AIO6	0100	REFO_I	1100	AIO7	0101	VSS	1101	AIO8	0110	TSN0	1110	-	0111	TSN1	1111	-
ADINN[3:0]	ADC 输入负通道	ADINN[3:0]	ADC 输入负通道																																			
0000	AIO0	1000	VDDA																																			
0001	AIO1	1001	AIO4																																			
0010	AIO2	1010	AIO5																																			
0011	AIO3	1011	AIO6																																			
0100	REFO_I	1100	AIO7																																			
0101	VSS	1101	AIO8																																			
0110	TSN0	1110	-																																			
0111	TSN1	1111	-																																			

23.3.3. ADC 寄存器 2

ADC Base Address + 0x08 (0x41108)

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

Symbol		ADCCR2 (ADC Control Register 2)	
Bit		[31:16]	
名称		ADCO	
RW		R-0	
Bit	[15:8]		[7:0]
名称	ADCO		0x00
RW	R-0		R-0

ADCO[31:0] ADC 转换值输出寄存器，只有高 24-bit 数据有效

DAFM[0]=0 时，ADCO[31:8]数据 = COMB Filter 原始数据

注意事项:

- (1) 芯片程序读取 ADO 将自动清除 ADC 中断信号 (0x40008 : ADCIF)，ICE 开发接口读取 ADO 并不会触发 ADCIF 被清除 (EDM)。
- (2) 对 ADO[0] 写 0 可以触发 ADCIF 清除，供开发接口使用。

23.3.4. ADC 寄存器 3

Symbol		ADC Base Address + 0x0C (0x4110C) ADCCR3 (ADC Control Register 3)	
Bit		[31:16]	
名称		ADCO1	
RW		R-0	
Bit	[15:8]		[7:0]
名称	ADCO1		0x00
RW	R-0		R-0

ADCO1[31:0] ADC 转换值输出寄存器，只有高 24-bit 数据有效

DAFM[0]=0 时，ADCO1[31:8]数据= COMB Filter 前一笔数据

注意事项:

- (1) 芯片程序读取 ADO1 将自动清除 ADC 中断信号 (0x40008 : ADCIF)，ICE 开发接口读取 ADO1 并不会触发 ADCIF 被清除 (EDM)。
- (2) 对 ADO1[0] 写 0 可以触发 ADCIF 清除，供开发接口使用。

23.3.5. ADC 寄存器 4

Symbol		ADC Base Address + 0x10 (0x41110) ADCCR4 (ADC Control Register 4)		
Bit	[31:24]		[23:16]	
名称	MASK		-	
RW	R0W-0		RW-0	
Bit	[15:08]		[7-1]	[0]
名称	MASK		-	-
RW	R0W-0		RW-0	R-X

位	名称	描述
Bit[23:16]	-	保留请勿设定

HY16F3910 User's Guide

21-bit ENOB $\Sigma\Delta$ ADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述
Bit[7:1]	-	保留请勿设定
Bit[0]	-	保留请勿设定

24. 串行通信 SPI

24.1. 整体总说明

HY16F3910 有 1 个串行外围接口 Serial Peripheral Interface (SPI)。这个 SPI 使用同步串行数据通信协议，并使用全双工模式来运作。它与 4-线双向接口做通信，且可在主/从端模式下运作。在主端模式下，它有数个组态来执行不同的从端装置。

功能：

全双工同步传输。

支持主端模式或从端模式运作。

支持 MSB 或 LSB 最先传输。

传输框格式为 4~32 位可程序化设定 BIT 长度。

高速 SPI 总线忙碌状态标志位。

可编程时钟脉冲率。

支持高/低电位从机端选择。

可编程时钟极性及其相位。

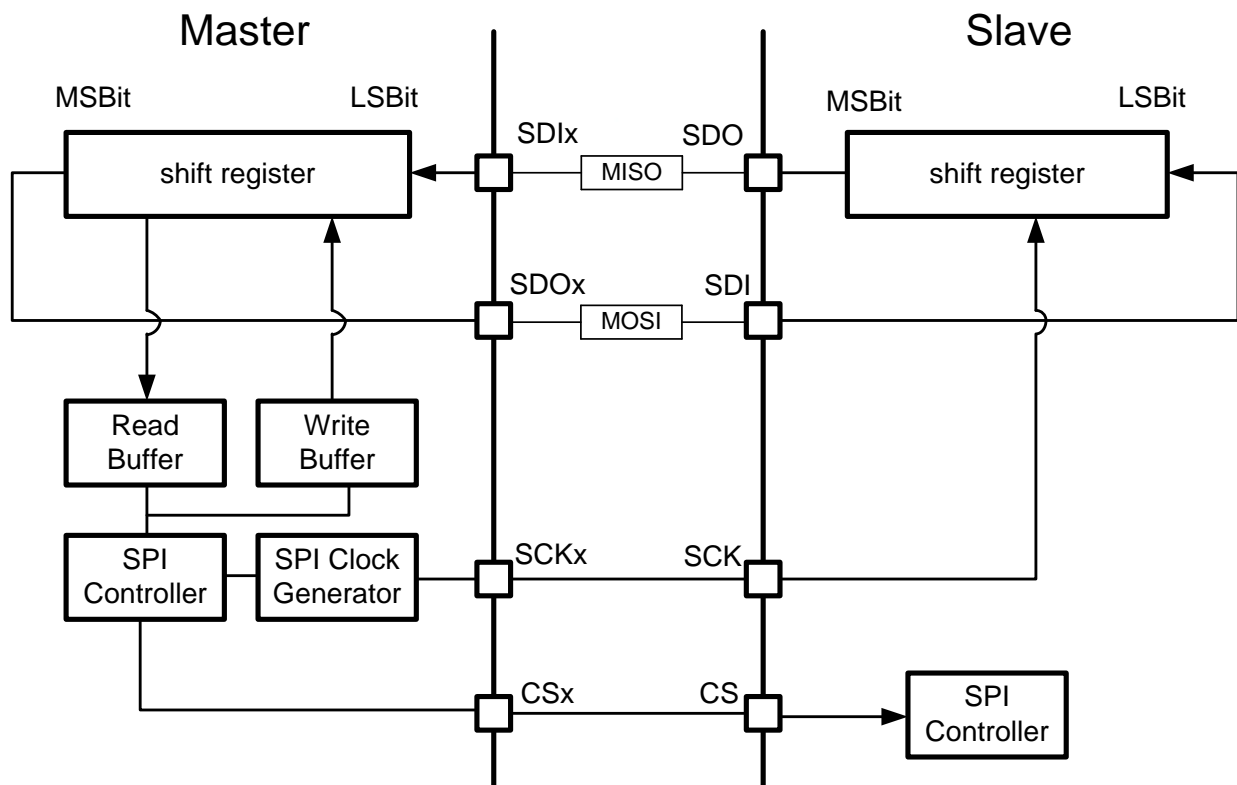


图 24-1 串行通信 SPI 架构图

MISO 引脚是主端装置的输入和从端装置的输出。MOSI 引脚是主端装置的输出和从端装置的输入。SCK 引脚是来自主端装置的串行通信时钟输出。CS 引脚是来自主端装置的芯片选择，以启动从端装置的 SPI 通信。这些主端装置或从端装置 MOSI/MISO/SCK/CS 引脚被连接在一起以便执行工作。通信永远是由主端装置所启动。主端装置经由 MOSI 引脚传送数据给从端装置，而从端装置经由 MISO 引脚回应。所以，这是全双工通

信，数据进出同步，使用相同的时钟源。

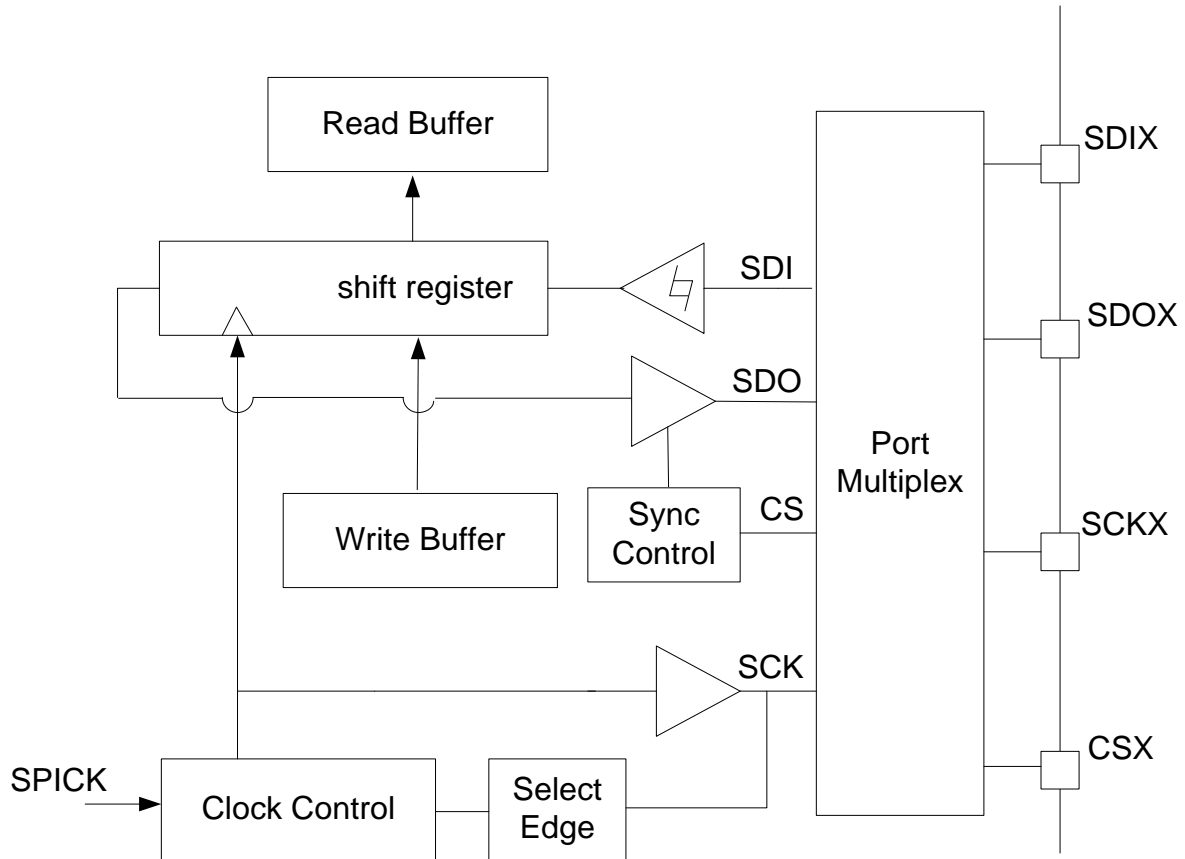


图 24-2 SPI 的 IO 引脚图

功能描述:I/O 引脚设定:

这些 SPI 引脚可用程序编写给不同的 I/O 引脚。

时钟相位与时钟极性:

可用软件来组成四个不同的时序方案，并由 CPOL 和 CPHA 寄存器来控制。

CPOL(时钟极性)是在没有数据传输的情况下，控制时钟的稳定状态值。

它可用在主端模式和从端模式中。如果 CPOL 为 1(高电位)，则当 SPI 处于闲置模式时，

SCK 就会是 1。另一方面，如果 CPOL 是 0(低电位)，则当 SPI 处于闲置状态时，SCK 就会是 0(低电位)。

CPHA(时钟相位)控制 SCK 的数据频率沿捕捉。如果 CPHA 为 1(高电位)，

SCK 引脚的第二个频率沿(如果 CPOL 为 1 则是上升沿；为 0 则是下降沿)就会捕捉到 MSB 数据。这个数据会被拴在第二个 SCK 频率沿。另一方面，如果 CPHA 是 0(低电位)，

SCK 引脚上的第一个频率沿(如果 CPOL 为 1 则是下降沿；为 0 则是上升沿)就会捕捉到 MSB 数据。这个数据会被拴在第一个 SCK 频率沿。因此，CPOL 和 CPHA 寄存器的组合控制数据捕捉和频率沿输出。

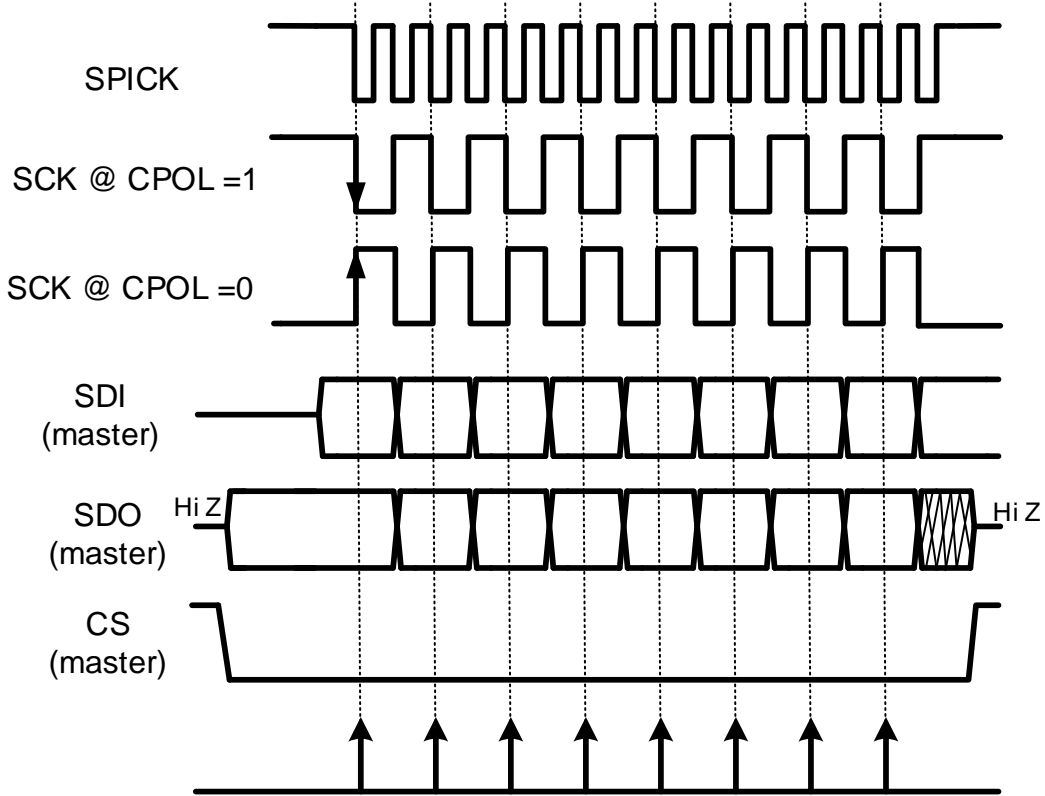


图 24-3 SPI 的主动模式时序图(CPHA=0)

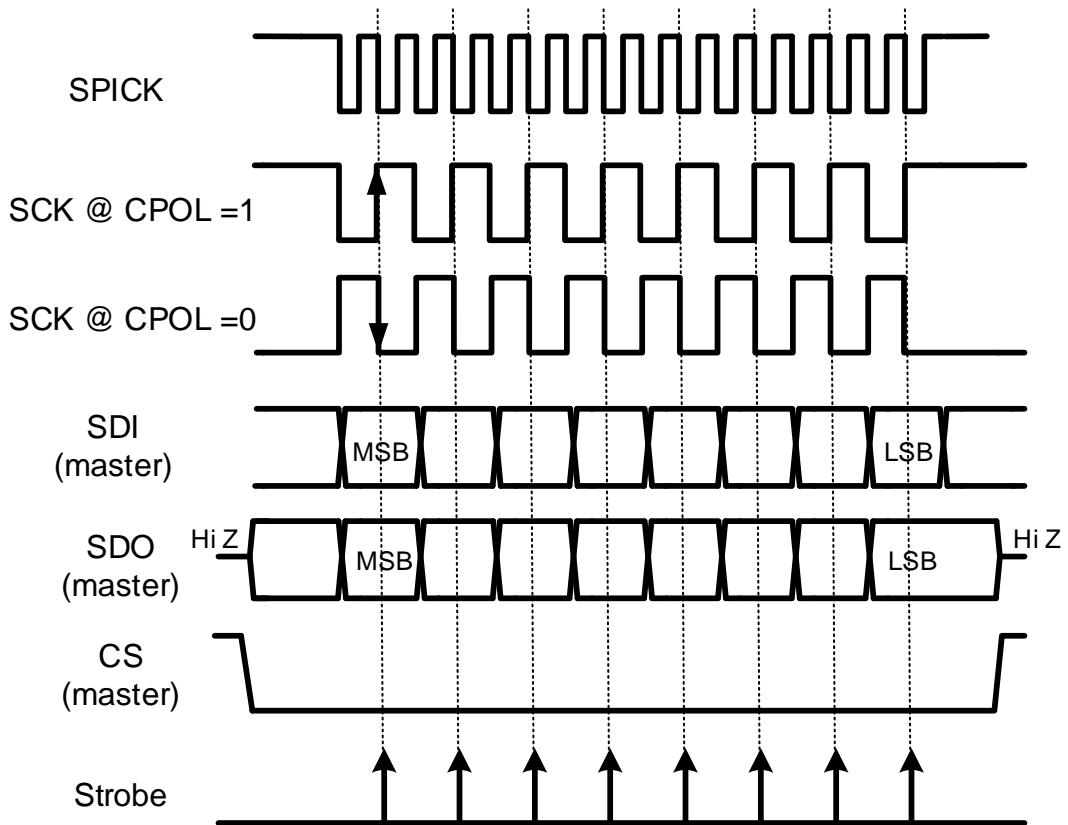


图 24-4 SPI 的主动模式时序图(CPHA=1)

SPI 控制寄存器 1 (SPI Control Register 1,):

(BL 控制位)数据框格式:

用来传送和接收的交易字(transaction word)位长度可在寄存器的控制位 BL 0x40F04[4:0]中予以定义。最小的位长度为 4 个位，最大长度为 32 个位。数据在移位寄存器中，其传输格式可为 MSB 位先被传送或 LSB 位先被传送，且由寄存器的控制位 LBF 0x40F04[18]所定义。如果 LBF 是<0>，则数据传输格式为在移位寄存器中的 MSB 位先被传输。然后，第二个 MSB 被传送，最后才是 LSB 位。如果 LBF 是<1>，则数据传输格式为 LSB 位先被传送。

(CSL 控制位)从端装置芯片选择级别:

适用于 SPI 四线式主端与从端模式。这个 CS 引脚可被定义为 0 或 1(低电位或高电位)以启动从端装置。这是由缓存器的控制位 CSL 0x40F04[19]所控制。如果在主端装置中的 CSL 是<0>，则 CS 引脚就会输出 0(低电位)以启动从端装置。另一方面，如果主端装置中的 CSL 为<1>，则 CS 引脚就会输出 1(高电位)以启动从端装置。如果在从端装置中的 CSL 为<0>，则从端装置会在收到一个 CS 的输入 0(低电位)后被启动。另一方面，如果在从端装置中的 CSL 为<1>，则从端装置会在收到一个 CS 的输入 1(高电位)后被启动。

注意事项： SPI Interface 工作于四线式 Master mode 时，CS 脚位的控制是属于半自动控制的方式，例如：当 CSL 设定为<1>时，此时 CS 脚位会被拉到低电位，当 SPI Master 要写数据到终端 SPI Device 的时候，CS 脚位会自动拉到高电位，待数据传送完成之后，会再自动回复到低电位，即是 Idle 时候为 Low，Active 时候才为 High。

(CSO 控制位)

这个控制位只有在三线式 SPI Slave mode 才会使用到。此脚位的功能为芯片内部唤醒 CS 信号仿真器控制。当 SPI Master 要写数据给 SPI Slave，SPI Slave 要接收数据之前，则需要先设定 CSO=<0>才能正确接收数据。当数据接收完成后，要把数据从 RXB Buffer 读出前，则需要先设定 CSO=<1>后，才能正确读取所接收数据，在数据读取后则需要设定 CSO=<0>才能准备接收下一笔数据。而当 SPI Slave 要回传数据给 SPI Master 的时候，同样需要先设定 CSO=<1>后，将需要传送数据写入 TXB Buffer 之后，再设定回 CSO=<0>，这样才可以将数据回传给 Master。

注意事项： 当使用三线式的 SPI 传输时，如果 SPI Slave 端已经先完成了初始化，并且设定 CSO=0，此时如果 SPI Master 端才上电做初始化的设定，会有可能造成 SPI Master 在初始化的过程中因为 GPIO 的初始变化，造成 SPI Slave 端误判，接收到一笔错误的的数据，因而引起 SPI Slave 端有第一笔数据误接收的可能性。因为三线式的 SPI 传输中，本身并没有 CS 脚位做同步的动作，使用者应该在 SPI Master 与 Slave 端初始化过程中建立握手协议(Handshake Protocol)，在确认双方都初始化完成之后才开始做数据传输。

SPI 控制寄存器 0 (SPI Control Register 0) :

(OVF 控制位):

OVF 是 SPI 的溢位标志位。当传输期间有额外的 SCK 频率沿输入时，它就会是在高电位(1)。例如，如果一个交易字(Word)的位长度是 16 个位，且在 CS 改变为高电位前(在此例，CSL 为<0>)，有 17 个来自主端装置的频率缘，而当 OVF 收到第 17 个频率沿时，它的值是 1。这表示此一传输有错误发生。如果第 17 个频率沿已发生，代表第一个被传输的数据遗失了。

(ABF 控制位):

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

ABF 是 SPI 中止标志位，只用在从端模式中。在传输期间，当 SCK 频率沿输入不足时，它就会是在高电位(1)。例如，如果一个交易字(word)的位长度是 16 个位，有 15 个来自主端装置的频率沿，且 CS 改变为高电位(在此例，CSL 为<0>)，则 ABF 为<1>。这表示此一传输有错误发生。交易未完成，传输的数据被更新到读取寄存器中。传输被中止，且会遗失掉。

(BUF 控制位):

BUF 是 SPI 的忙碌标志位。当 SPI 在传输或接受数据时，它是在高电位(1)。在主端装置中，当 SPI 开始数据传输时，它是在高电位(1)。一旦 SPI 停止数据传输或传输字已完成传输时，它就会自动被清除。在从端模式中，当 SPI 准备好要与主端装置通信时，则 BUF 是<1>。一旦 SPI 停止数据传输或传输字已完成传输时，它会自动被清除。

SPI 中断标志位控制位:

- (1) STxIF:标志位 STxIF 是 SPI 的传输中断(interrupt)。当写入寄存器被加载到移位寄存器时，它会被设定为<1>。
- (2) SRxIF:标志位 SRxIF 是 SPI 的接收中断。当移位寄存器被加载到读取寄存器时，它会被设定为<1>。

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



24.2. 寄存器地址

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	

24.3. 寄存器功能

24.3.1. SPI 寄存器 0

SPI Base Address + 0x00 (0x40F00)									
Symbol	SPICR0 (SPI Control Register 0)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	-	RxF	OVF	ABF	BUF	DCF	TxBF	RxBF
RW	R0W-0	-	R-0	RW0-0		R-0			
Bit	[15:08]	[07:04]			[03]	[02]	[01]	[00]	
名称	MASK	-			CPHA	CPOL	M/S	En	
RW	R0W-0	-			RW-0				

位	名称	描述
Bit[22]	RxF	接收(RX)寄存器值更新标志位
		0 正常
		1 接收(RX)寄存器的数据有更新, 此时不能读取接收寄存器
Bit[21]	OVF	SPI 总线的数据长度过长标志位
		0 正常
		1 接收到的数据长度大于自行设置的数据长度 BL[4:0].写入 0 可清除 OVF 标志位
Bit[20]	ABF	SPI 总线的数据长度偏少标志位
		0 正常
		1 接收到的数据长度小于自行设置的数据长度 BL[4:0].写入 0 可清除 ABF 标志位
Bit[19]	BUF	SPI 总线繁忙标志位
		0 SPI 总线界面空闲待机状态
		1 SPI 总线界面繁忙状态
Bit[18]	DCF	数据丢失标志位
		0 正常
		1 接收寄存器已满而继续接收数据, 旧的数据会丢失, 读取接收寄存器可清零该位
Bit[17]	TxBF	TX 发送寄存器已满标志位
		0 TX 发送寄存器为空, 可发送数据
		1 TX 发送寄存器已满, 继续写入数据将会覆盖旧的数据
Bit[16]	RxBF	Rx 接收寄存器已满标志位
		0 RX 接收寄存器为空
		1 RX 接收寄存器已满(读取接收寄存器可以清零该位)

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述
Bit[03]	CPHA	SPI 总线捕捉数据的频率相位设置
		0 在 SCK 第一个频率沿捕捉数据
		1 在 SCK 第二个频率沿捕捉数据
Bit[02]	CPOL	SPI 总线工作频率极性控制
		0 SCK 低电位为空闲
		1 SCK 高电位为空闲
Bit[01]	M/S	SPI 工作模式设置
		0 被动模式
		1 主动模式
Bit[00]	EN	SPI 功能开启控制
		0 关闭
		1 开启

24.3.2. SPI 寄存器 1

SPI Base Address + 0x04 (0x40F04)						
Symbol	SPI CR1(SPI Control Register 1)					
Bit	[31:24]	[23:21]	[20]	[19]	[18]	[17:16]
名称	MASK	-	CSO	CSL	LBF	MD
RW	R0W-0	-	RW-0			
Bit	[15:05]	[04:00]				
名称	-	BL				
RW	-	RW-0				

位	名称	描述	
Bit[20]	CSO	芯片内部唤醒(CS)信号仿真器控制, 适用于 3 线模式	
		0 CS 信号仿真器工作	
		1 CS 信号仿真器待机	
Bit[19]	CSL	CS 信号极性设置, 用于启动器件, 适用于 4 线主端与从端模式	
		0 低电位启动	
		1 高电位启动	
Bit[18]	LBF	数据发送顺序	
		0 MSB 先发送	
		1 LSB 先发送	
Bit[17:16]	MD	SPI 接口工作模式设置	
		MD[1:0]	工作模式
		00	SPI 标准 4 线通信接口模式
		01	SPI 通用 3 线接口模式
		10	TI 模式
11	TI 模式		

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述			
Bit[4:0]	BL	SPI 发送一个字的数据长度设置			
		BL[4:0]	数据长度	BL[4:0]	数据长度
		00000	8 bits length	10000	17 bits length
		00001	16 bits length	10001	18 bits length
		00010	24 bits length	10010	19 bits length
		00011	4 bits length	10011	20 bits length
		00100	5 bits length	10100	21 bits length
		00101	6 bits length	10101	22 bits length
		00110	7 bits length	10110	23 bits length
		00111	8 bits length	10111	24 bits length
		01000	9 bits length	11000	27 bits length
		01001	10 bits length	11001	26 bits length
		01010	11 bits length	11010	27 bits length
		01011	12 bits length	11011	28 bits length
		01100	13 bits length	11100	29 bits length
		01101	14 bits length	11101	30 bits length
		01110	15 bits length	11110	31 bits length
01111	16 bits length	11111	32 bits length		

当 MD 被设定为 3 线模式时，原本的 CS 引脚就会变成 GPIO 模式。

24.3.3. SPI 寄存器 2

SPI Base Address + 0x08 (0x40F08)	
Symbol	SPICR2 (SPI Control Register2)
Bit	[31:16]
名称	RXB31_16
RW	R-X
Bit	[15:0]
名称	RXB15_00
RW	RW-X

位	名称	描述
Bit[31:0]	SPIRB	SPIRB[31:00] 是 32 位的接收寄存器

以 LBF 位来设定 LSB 或 MSB 先被传输。

当 LSB 被设定为先传输，就会影响到数据储存的位置，RXB 有效数据会，被向右对齐。

例如，BL 被设定为 8 位模式时，接收到的数据就会在 RXB [7:0]；

设定为 9 位模式时，接收到的数据就会在 RXB [8:0]，以此类推。

当设定 MSB 先被传输时，RXB 有效数据会被向左对齐。

例如，BL 被设定为 8 位模式时，接收到的数据就会在 RXB [31:24]；

设定为 9 位模式时，接收到的数据就会在 RXB [31:23]，以此类推。

24.3.4. SPI 寄存器 3

SPI Base Address + 0x0C (0x40F0C)	
Symbol	SPICR3 (SPI Control Register 3)
Bit	[31:16]

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

名称	TXB31-16
RW	R-X
Bit	[15:0]
名称	TXB15-0
RW	RW-X

位	名称	描述
Bit[31:0]	SPITB	SPITB[31:0] 是 32 位的发送寄存器

以 LBF 位来设定 LSB 或 MSB 先被传输。

当 LSB 被设定为先传输，就会影响到数据储存的置，TXB 有效数据会被向右对齐。

例如，BL 被设定为 8 位模式时，接收到的数据就会储存在 TXB [7:0]；

设定为 9 位模式时，接收到的数据就应储存在 TXB [8:0]，以此类推。

当设定 MSB 先被传输时，TXB 有效数据会被向左对齐。

例如，BL 被设定为 8 位模式时，接收到的数据就会储存在 TXB [31:24]；

设定为 9 位模式时，接收到的数据就会储存在 TXB [31:23]，以此类推。

25. 异步串行通信 UART

25.1. 整体总说明

HY16F3910 有 2 组异步串行通信为 UART 和 UART2。此为增强型 EUART (Enhanced Universal Asynchronous Receiver Transmit)。这个 UART 的外围通常称作串行通信接口(SCI)。UART 可设定为全双工异步系统，其外围通信装置包括 LCD/LED 屏幕终端机和 PC/NB/Tablet/Smart Phone。它也可被设定为半双工同步系统，其外围通信装置则包括 ADC 或 DAC 整合电路、串行 EEPROM/Flash 等。加强型的 UART 具有额外的特性，包括数据框错误侦测和自动地址辨识。数据框错误侦测可以决定一个数据框是否为有效或是否未经过框停止位。自动地址辨识功能可将地址框内容与单芯片地址做比较；而串行中断只能在这两者符合时才能被产生。

25.1.1. Baud Rate 传输波特率

寄存器 UARTCR2[15:0]是一个专用的 16bit baud rate 发生器，支持 EAURT 的异步模式。下表是串行波特率的计算公式，但是仅适用于主控模式。且在给定目标波特率及工作频率(UART_CK)为 HSXT 或 HSRC 的情况下，可以使用下表的公式来计算 Baud Rate 的近似整数，从而可以确定串行传输波特率误差。且建议在切换工作频率后，需要重置波特率或使用自动波特率功能，重新校正 Baud Rate 的值。

Baud Rate/EUART MODE	串行传输波特率计算公式
16 bit/异步	$UART_CK \div [4x(n+1)]$
	n= UARTCR2 寄存器的值；

例如：工作在异步模式下，其工作频率假设为 4.147MHZ，而目标波特率 9600bps，可计算 Baud Rate 的值。

$$\begin{aligned} \text{根据公式: Baud Rate} &= ((UART_CK \div \text{目标波特率}) \div 4) - 1 \\ &= ((4147000 \div 9600) \div 4) - 1 \\ &= 106.995 \\ &\approx 107 \end{aligned}$$

而根据上述计算 Baud Rate 值来计算波特率值：

波特率 = $4147000 \div (4 \times (107 + 1)) = 9599.5$;所以存在一定误差，该误差的计算方式为：

$$\begin{aligned} \text{误差率} &= (\text{实际计算波特率} - \text{目标波特率}) / \text{目标波特率} \\ &= (9599.5 - 9600) / 9600 \\ &= 0.0052\% \end{aligned}$$

25.1.2. 自动波特率功能(Auto Baudrate detection)

UART 模块支持自动检测和校正波特率的功能，称之为自动波特率功能。自动波特率必须在控制位 RxEn=1b 和 RxABDEn =1b 时才有效。在接收到开始状态后，即开始进行自动波特率检测功能(接收数据需要为 0x55)，在自动检测和校正完成后便将计算结果写入寄存器 UARTCR2[15:0]。

UART Auto-Baud rate 自动波特率设置流程：

- UART 初始化设置：包含 UART TX, RX Port 设定。TX 和 RX 对应到的 GPIO 脚位需要设置对应 TX 为 Output 和 RX 为 Input。
- Auto Baudrate 初始化设置：预先清除 UARTCR2[15:0]寄存器内容，关闭 RX 的 GPIO Input 设定，等待 RX IRQ(URxIF)中断标志位产生，当收到 RX IRQ(URxIF)之后，再重新设定 RX 对应到的 GPIO port 为 Input。设置完成后，要再清除 UART 状态标志位寄存器与清除 UART RX Data Buffer 和 RX IRQ(URxIF)，即完成 Auto Baudrate 初始化设置。

- C. 设置 Auto-baud Enable and Detection : 开启 Auto baud rate 功能 RxABDEn =1b , 并且等待 0x55。
当收到 0x55 之后, 寄存器 UARTCR2[15:0]会自动填入目标波特率, 完成 Auto-baud rate 设置。最后
建议可在做完 Auto-baud rate 之后, 增加 Hand shark process , 目的是确认 auto-baud rate 得正确性。

25.1.3. 通信 IO 引脚

UART 通信总线只用两根线 TX/RX , 芯片为 UART 模块配置了 6 组通信 IO 引脚(每一组包含 TX/RX 线) , 方便用户在使用上设计的自由性。但这个是 IO 的复用功能, 透过 GPIO 复用功能控制器 0x40844 的控制位 PTUR 与 PTURE 方便的选择及开启 UART 的通信 IO 引脚, 注意, 在使用 UART 功能同时, 需要先开启 IO 通信引脚, 且对应的 IO 引脚需要被设置为输入或输出模式。UART 通信 IO 引脚分布如下表所示。

UART Port1(简称 UART)				UART Port2(简称 UART2)			
PTUR[2:0]	PTURE	TX	RX	PTUR2[2:0]	PTUR2E	TX2	RX2
000	1	PT1.0	PT1.1	000	1	PT1.2	PT1.3
001	1	PT1.4	PT1.5	001	1	PT1.6	PT1.7
010	1	PT2.0	PT2.1	010	1	PT2.2	PT2.3
011	1	PT2.4	PT2.5	011	1	PT2.6	PT2.7
100	1	PT6.0	PT6.1	100	1	PT6.2	PT6.3
101	1	PT7.4	PT7.5	101	1	PT7.6	PT7.7
110	1	PT9.0	PT9.1	110	1	PT9.2	PT9.3
111	1	PT8.0	PT8.1	111	1	PT8.2	PT8.3

表 25-1 UART1/2 通信 IO 引脚分布

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



25.2. 寄存器地址

UART Register Address	31	24	23	16	15	8	7	0
UART Base Address + 0x00(0x40E00)	MASK0		REG0		MASK1		REG1	
UART Base Address + 0x04(0x40E04)	-		-		MASK2		REG2	
UART Base Address + 0x08(0x40E08)	-		-		Baud Rate			
UART Base Address + 0x0C(0x40E0C)	-		TX		-		RX	

-保留

25.3. 寄存器功能

25.3.1. UART 寄存器 0

UART Base Address + 0x00 (0x40E00)									
Symbol	UARTCR0 (UART Control Register 0)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	MASK	OErr	NErr	FErr	PErr	TxBusy	TxBF	RxBusy	RxBF
RW	R0W-0	RW0-0				R-0			
Bit	[15:08]	[07:06]		[05:04]		[03]	[02]	[01]	[00]
名称	MASK	PLen		DLen		RxIT	RxEn	TxIT	TxEn
RW	R0W-0	RW-1		RW-2		RW-0			

位	名称	描述	
Bit[23]	OErr	RX Buffer over run error flag	
		0 Normal 1 Over run	
Bit[22]	NErr	RX Noise detected flag	
		0 Normal 1 Noise detected	
Bit[21]	FErr	RX Frame check erroe flag	
		0 Normal 1 Frame check error	
Bit[20]	PErr	RX Parity check erroe	
		0 Normal 1 Parity check error	
Bit[19]	TxBusy	TX Busy falg	
		0 Idle 1 Busy	
Bit[18]	TxBF	TX Buffer Full flag	
		0 Empty 1 Full	
Bit[17]	RxBusy	RX Busy flag	
		0 Idle 1 Busy	
Bit[16]	RxBF	RX Buffer Full flag	
		0 Empty 1 Full	
Bit[7:6]	PLen	TX 停止位长度控制	
		PLen[1:0]	停止位长度
		00	0.5Bit
		01	1Bit
		10	1.5Bit
		11	2 Bit

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述		
Bit[5:4]	DLen	TX/RX 数据长度		
		DLen[1:0]	Normal Mode	Parity Check Mode
		00	6 Bit Mode	5 Bit Mode
		01	7 Bit Mode	6 Bit Mode
		10	8 Bit Mode	7 Bit Mode
		11	9 Bit Mode	8 Bit Mode
Bit[03]	RxIT	RX 中断方式选择		
		0 当 RX Data Buffer 有数据时发出中断，读取数据后中断消失 注意:必须透过读取 Rx Data Buffer 寄存器的动作后，再下清除 URxIF=0b 的动作，则才可以正确清除中断标志位，若没有读取 Rx Data 寄存器的动作，透过指令仍无法清除 URxIF 状态 1 当 RX 接收完一笔数据后发出中断；手动清零中断标志位		
Bit[02]	RxEn	UART RX 控制开关		
		0 关闭 1 开启		
Bit[01]	TxIT	TX 中断方式选择		
		0 当 TX Data Buffer 空闲时发出中断，写入资料后中断消失 1 当 TX 传送完一笔数据后发出中断；手动清零中断标志位		
Bit[00]	TxEn	UART TX 控制开关		
		0 关闭 1 开启		

25.3.2. UART 寄存器 1

UART Base Address + 0x04 (0x40E04)							
Symbol	UARTCR1 (UART Control Register 1)						
Bit	[31:16]						
名称	-						
RW	-						
Bit	[15:08]	[07:05]	[04]	[03]	[02]	[01]	[00]
名称	Mask	-	RxABDF	RxABDEn	RxWUEn	PrtEn	PrtODD
RW	R0W-0	-					RW-0

位	名称	描述
Bit[04]	RxABDF	自动波特率侦测错误标志位
		0 正常 1 发生错误
Bit[03]	RxABDEn	自动侦测波特率开关
		0 关闭 1 开启
Bit[02]	RxWUEn	自动唤醒模式
		0 关闭

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述	
		1	开启
Bit[01]	PrtEn	奇偶校正开关	
		0	关闭
		1	开启
Bit[00]	PrtODD	选择奇同位、偶同位校准	
		0	偶同位校准
		1	奇同位校准

25.3.3. UART 寄存器 2

UART Base Address + 0x08 (0x40E08)	
Symbol	UARTCR2 (UART Control Register 2)
Bit	[31:16]
名称	-
RW	-
Bit	[15:0]
名称	Baud Rate
RW	RW-X

位	名称	描述
Bit[15~0]	Baud Rate	UART 波特率设定

25.3.4. UART 寄存器 3

UART Base Address + 0x0C (0x40E0C)		
Symbol	UARTCR3 (UART Control Register 3)	
Bit	[31:25]	[24:16]
名称	-	Tx Data
RW	-	W-X
Bit	[15:9]	[8:0]
名称	-	Rx Data
RW	-	R-X

※ 注意事项 ※

HY16F3910 对 0x40E0C~0x40E0F 任一字节的读取，都会触发 Rx Data Buffer 被读取而清除 Rx Data Buffer。

位	名称	描述
Bit[24:16]	Tx Data	TX Data Buffer
Bit[08:00]	Rx Data	RX Data Buffer

25.4. UART 使用说明

HY16F3910 有两组 UART 与 UART2 可供使用者使用,使用 UART 做串行传输控制流程如下,首先是 UART I/O 脚位的初始化设置,在初始化 UART 的设置部分需要注意到,当选择好 TX/RX IO 通信引脚之后,需要先做开启 IO 通信引脚动作,并且对应的 IO 引脚需要使用 GPIO 来设置为输入或输出模式。第二点为 UART 频率源选择控制, UART 频率源可以选择使用内部振荡器或者外部振荡器, UART 频率源的选择与 UART 的除频会决定 UART 传输速度。以上两点设定完之后也要设定 UART 传输协议,包含波特率设定与传输位等选择,最后在 UART 设定完成之后,需要先做一小段的 Delay 时间设置,此为 IO 初始化稳定时间,当 IO 初始化达到稳定之后,即可做 UART 使能动作,完成 UART 初始化动作。UART 做传输数据使用上,建议所有的数据接收和传送都是在(Interrupt)中断事件里面做处理,如果是使用 UART,则是在 INT HW0 做中断处理,如果是使用 UART2,是在 INT HW7 做中断处理。在 UART 完成初始化并且开启 TX 与 RX 中断使能之后,即可以开始等待中断条件成立并做 UART 串行数据传输。

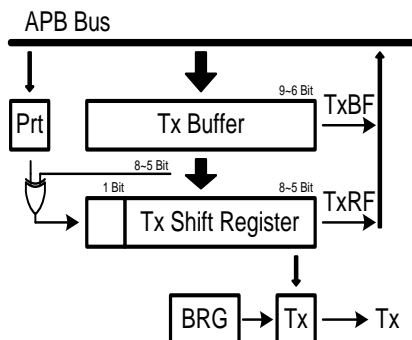
UART 中断说明 :

以下为 URxIF, URxIR, IRxIE 使用关系说明。

INT Base Address + 0x00 (0x40000)														
INTCOM (Interrupt Control Register 0)														
Symbol	[31:24]						[23:22]	[21]	[20]	[19]	[18]	[17]	[16]	
名称	MASK						-	I2CEIE	I2CIE	UTxIE	URxIE	STxIE	SRxIE	
RW	R0W-0						-	RW-0						
Bit	[15:14]	[13]	[12]	[11]	[10]	[09]	[08]	[07:06]	[05]	[04]	[03]	[02]	[01]	[00]
名称	MASK							-	I2CEIF	I2CIF	UTxIF	URxIF	STxIF	SRxIF
	-	I2CEIR	I2CIR	UTxIR	URxIR	STxIR	SRxIR							
RW	R-0							-	RW0-0					

- URxIE =0b, UART RX 接收发生中断时, URxIR=0b. URxIF=1b, 但是芯片不会进到中断子程序 HW0 中。
- URxIE =1b, UART RX 接收发生中断时, URxIR=1b. URxIF=1b, 芯片进到中断子程序 HW0 中。
- 清除 URxIF=0b 动作时,同时 URxIR=0b。
- 目前函式库对于中断标志位的清除动作等,是藉由操作 URxIF 来控制。

UART TX Interface 说明 :

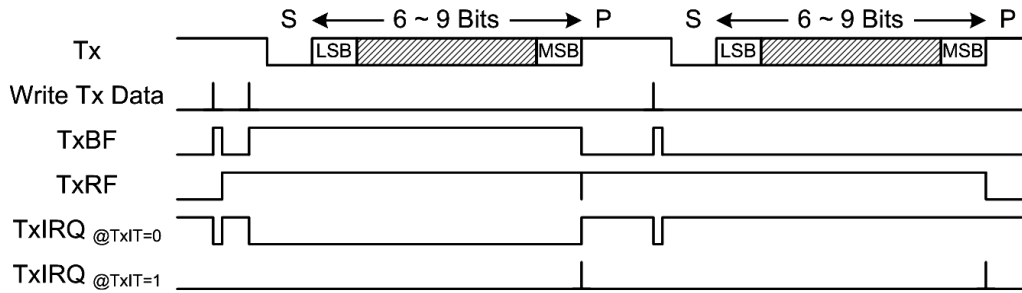


UART Transmit Block Diagram

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

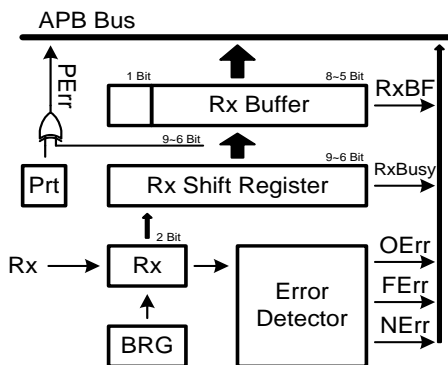
4 X 44~8 X 40 LCD Driver



动作说明:

- TXRF, 此为 Tx Shift Register 的状态。
- 当写数据到 TX Data 寄存器后, TxBF=1b, 代表 Tx Buffer 不为空。之后会将数据移位到 Tx shift Register 内, 这时 Tx Buffer 即为空, TxBF=0b。
- 当 Tx 数据还未全部传送出去时, 此时又写数据到 TX Data 寄存器时, 则 TxBF=1b, 代表 Tx Buffer 不为空。直到 Tx shift Register 内的数据全部传送出去后, Tx Buffer 已经将数据移位到 Tx shift Register 内, 则 TxBF=0b。
- 当 TxBF=1b 情况下, 又写数据到 TX Data 寄存器时, 此时新的数据会将 Tx Buffer 内的值覆盖过去。使用者需要判断, 避免数据被覆盖掉。
- TxIT 的设定会影响 UTxIF 中断产生的方式(图中以 TxIRQ 描述)。当 TxIT=0b 时, 其 UTxIF 产生的方式, 相同于 HY16F188 系列产品使用方式。而在 HY16F3910 系列产品使用上, 增加了 TxIT=1b 新功能设定。
- TxIT=0b, 当 TX Buffer 空闲时发出中断, 写入资料后中断消失; UTxIF 的标志位则与 TxBF 标志位反向。只要 Tx Buffer 为空时, UTxIF=1b。因此这个状态下, 若程序一开始就开启 UTxIE=1b, 则就会不停进入中断。
- TxIT=1b, 当 TX 传送完一笔数据后发出中断; 当一笔数据完整输出到 STOP 发生时, 才会产生中断标志位 UTxIF=1b。使用者可以自行透过指令清除 UTxIF=0b。这个做法会方便使用者想要知道数据何时完整输出后, 而发生中断通知。

UART RX Interface 说明 :

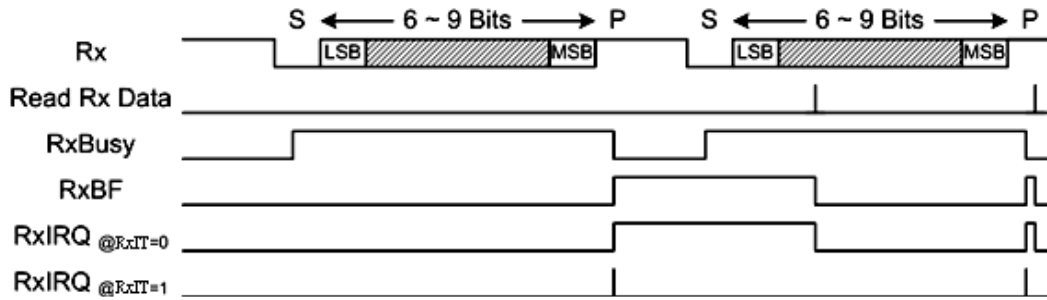


UART Receive Block Diagram

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



动作说明:

-UART 开始接收数据时，当接收到 STAR 后的半个 UART CLOCK 时，RxBusy=1b，RX Buffer 内还没接收完整数据，RX Buffer 为空，因此 RxBF=0b。当数据接收完成，发生 STOP 时，RxBusy=0b，RX Buffer 已经有数据，因此 RxBF=1b。

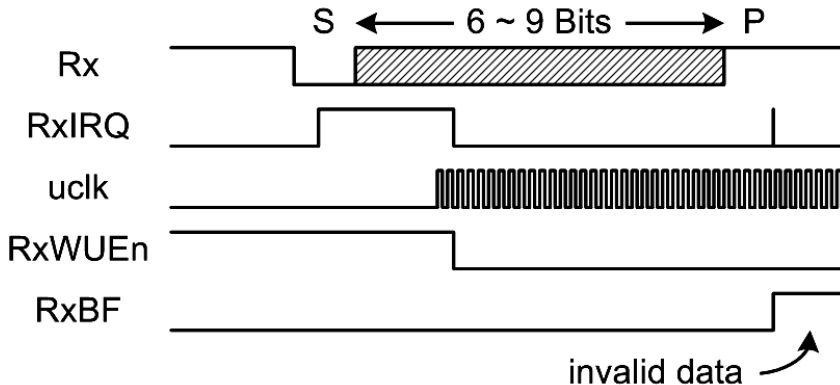
-RxIT 的设定会影响 URxIF 中断产生的方式(图中以 RxIRQ 描述)。

-RxIT=0b, 当 RX Buffer 有数据时发出中断, 读取数据后中断消失; 当数据接收完成后, RxBusy=0b, RxBF=1b, 则中断标志位发生 URxIF=1b。此时必须透过读取 Rx Data 寄存器的动作后, 再下清除 URxIF=0b 的动作, 则才可以正确清除中断标志位, 若没有读取 Rx Data 寄存器的动作, 透过指令仍无法清除 URxIF 状态。

-RxIT=1b, 当 RX 接收完一笔数据后发出中断; 当数据接收完成后, RxBusy=0b, RxBF=1b, 则中断标志位发生 URxIF=1b。此时不需要读取 Rx Data 寄存器, 都可以透过指令方式直接清除 URxIF=0b 的动作。

UART Auto WakeUp 使用说明：

当 HY16F3910 芯片进入省电模式(Sleep 或 Idle Mode)，可以设计使用 UART 的 RX 引脚来做唤醒动作。当进入省电模式时候，RxIRQ 所收到的第一笔数据为唤醒芯片用，需避免作为相关运算使用。以下说明 UART WakeUp 设置流程。



1. UART 初始化设置：包含 UART 的 TX 与 RX Port 设定，TX 和 RX 对应到的 GPIO 脚位需要设置对应 TX 为 Output 和 RX 为 Input。需注意: RX 引脚状态需要设定为内部 Pull High 状态或由外部线路使其 RX 引脚为 Pull High 状态。
2. 开启 UART WakeUp 功能，即设置寄存器 0x40E04[2]=RxWUEn=1b，并且开启 RX Interrupt，与使能全局中断 GIE=1。
3. 设置芯片进入省电模式(Sleep 或 Idle Mode)。 需注意:进入省电模式之前，需要先把 CPU 工作频率切换到内部低频 LPO，并且再把 CPU 高频 HAO 做关闭动作，这样才可以达到符合规格书预期的省电模式状态。
4. 等待 Host 端的 TX 传送信号到 HY16F3910 做唤醒芯片动作。当 HY16F3910 接收到 Host 端传送的 TX 信号，会先进入到 UART 中断，在 RxBF Flag 之后，清除此笔无效数据及相关 Interrupt Flag，并且重新再开启内部 HAO 高频，把 CPU 工作频率切换到 HAO，离开 UART 中断子程序返回到主程序。

需注意：在 RxBF 出现前的 Interrupt Flag 为唤醒芯片用，需避免作为相关运算用。如果是从 Sleep mode 唤醒至少要等 64msec(max: < 100msec)，芯片才可以开始动作，等待过程中，Host 端送出到 HY16F3910 芯片的 UART command 是无效的。

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

26. 异步串行通信 UART2

26.1. 整体总说明

设计方式相同于 UART 章节说明。

26.2. 寄存器地址

UART2 Register Address	31	24	23	16	15	8	7	0
UART2 Base Address + 0x00(0x40E10)	MASk0			REG0	MASk1		REG1	
UART2 Base Address + 0x04(0x40E14)	-			-	MASk2		REG2	
UART2 Base Address + 0x08(0x40E18)	-			-	Baud Rate			
UART2 Base Address + 0x0C(0x40E1C)	-			TX2	-		RX2	

-保留

26.3. 寄存器功能

26.3.1. UART2 寄存器 0

UART2 Base Address + 0x10 (0x40E10)									
Symbol	UART2CR0 (UART2 Control Register 0)								
Bit	[31:24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]
名称	Mask	OErr	NErr	FErr	PErr	TxBusy	TxBF	RxBusy	RxBF
RW	R0W-0	RW0-0				R-0			
Bit	[15:08]	[07:06]		[05:04]		[03]	[02]	[01]	[00]
名称	MASK	PLen		DLen		RxIT	RxEn	TxIT	TxEn
RW	R0W-0	RW-1		RW-2		RW-0			

位	名称	描述
Bit[23]	OErr	RX Buffer over run error flag
		0 Normal 1 Over run
Bit[22]	NErr	RX Noise detected flag
		0 Normal 1 Noise detected
Bit[21]	FErr	RX Frame check error flag
		0 Normal 1 Frame check error
Bit[20]	PErr	RX Parity check error
		0 Normal 1 Parity check error
Bit[19]	TxBusy	TX Busy flag
		0 Idle 1 Busy
Bit[18]	TxBF	TX Buffer Full flag
		0 Empty 1 Full
Bit[17]	RxBusy	RX Busy flag
		0 Idle 1 Busy
Bit[16]	RxBF	RX Buffer Full flag
		0 Empty 1 Full

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述		
Bit[7:6]	PLen	TX 停止长度控制		
		PLen[1:0]	停止位长度	
		00	0.5Bit	
		01	1Bit	
		10	1.5Bit	
		11	2 Bit	
Bit[5~:]	DLen	TX/RX 数据长度		
		DLen[1:0]	Normal Mode	Parity Check Mode
		00	6 Bit Mode	5 Bit Mode
		01	7 Bit Mode	6 Bit Mode
		10	8 Bit Mode	7 Bit Mode
		11	9 Bit Mode	8 Bit Mode
Bit[03]	RxIT	RX 中断方式选择		
		0 当 RX Data Buffer 有数据时发出中断，读取数据后中断消失		
		1 当 RX 接收完一笔数据后发出中断；手动清除中断标志位		
Bit[02]	RxEn	UART RX 控制开关		
		0 关闭		
		1 开启		
Bit[01]	TxIT	TX 中断方式选择		
		0 当 TX Data Buffer 空闲时发出中断，写入资料后中断消失		
		1 当 TX 传送完一笔数据后发出中断；手动清除中断标志位		
Bit[00]	TxEn	UART TX 控制开关		
		0 关闭		
		1 开启		

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

26.3.2. UART2 寄存器 1

UART2 Base Address + 0x14 (0x40E14)							
Symbol	UART2CR1 (UART2 Control Register 1)						
Bit	[31:16]						
名称	-						
RW	-						
Bit	[15:08]	[07:05]	[04]	[03]	[02]	[01]	[00]
名称	Mask	-	RxABDF	RxABDEn	RxWUEn	PrtEn	PrtODD
RW	R0W-0	-	RW-0				

位	名称	描述
Bit[04]	RxABDF	自动波特率侦测错误标志位
		0 正常
		1 发生错误
Bit[03]	RxABDEn	自动侦测波特率开关
		0 关闭
		1 开启
Bit[02]	RxWUEn	自动唤醒模式
		0 关闭
		1 开启
Bit[01]	PrtEn	奇偶校正开关
		0 关闭
		1 开启
Bit[00]	PrtODD	选择奇同位、偶同位校准
		0 偶同位校准
		1 奇同位校准

26.3.3. UART2 寄存器 2

UART2 Base Address + 0x18 (0x40E18)	
Symbol	UART2CR2 (UART2 Control Register 2)
Bit	[31:16]
名称	RSV.
RW	R-0
Bit	[15:00]
名称	Baud Rate
RW	RW-X

位	名称	描述
Bit[15:0]	Baud Rate	UART 波特率设定

26.3.4. UART2 寄存器 3

UART2 Base Address + 0x1C (0x40E1C)		
Symbol	UART2CR3 (UART2 Control Register 3)	
Bit	[31:25]	[24:16]
名称	-	Tx Data
RW	-	W-X
Bit	[15:09]	[08:00]
名称	-	Rx Data
RW	-	R-X

位	名称	描述
Bit[24:16]	Tx Data	TX Data Buffer
Bit[08:00]	Rx Data	RX Data Buffer

26.4. UART2 使用说明

UART2 与 UART 不同的地方在于控制位置寄存器与 IO 引脚配置与中断向量的不同，UART 为 INT HW0，而 UART2 为 INT HW7，其余控制方式皆相同。

27. 通用 I²C 通信接口

27.1. 整体总说明

HY16F3910 有 1 个通信接口(I²C), 包含主(Master)和从(Slave)两个运作模式如下图所示。主机模式可以根据系统的需求结合传送控制器(Transmission Controller, Tx Controller)传送 I²C 封包格式的信号至 I²C Bus, 并以 Clock Generator 决定所需的传送速率。而 Slave Controller 可以接收 I²C Bus 上的信号, 以(Slave)从机模式接受 Bus 上的(Master)主机之通信需求, 并结合传送控制器回传主机所需要的数据。

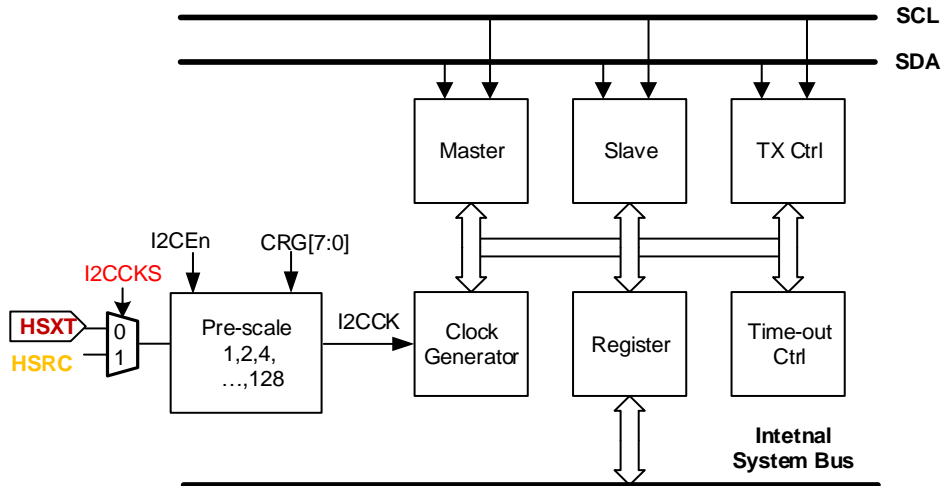


图 27-1 I²C 通信架构图

27.1.1. 通信 I²C 接口特性

标准 I²C 串行接口包括 2 引脚的串行数据(SDA)和串行频率(SCL)。引脚是开放式漏极开路输出结构, 需要外部上拉电阻以确保高电平输出。标准 I²C 串行接口可以组态设定为主(Master)模式、从(Slave)模式、或主/从(master/slave)模式。可编程时钟允许调整 I²C 的传输速率。在主从之间, 数据以双向传输。I²C 允许大的运作电压范围。

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

● I²C 串行接口功能特性：

- 标准 I²C 串行接口包含 2 个引脚，为串行数据(SDA)、串行频率(SCL)。
- 引脚为 Open Drain 输出结构，需要外部提升电阻，确保高电位输出。
- 标准 I²C 串行接口可配置为主机(Master)、从机(Slave)或主/从机模式。
- 可程序频率，允许调整 I²C 传输速率。
- 主机和从机之间的数据传输为双向的。
- I²C 允许相当大的工作电压范围。
- I²C 的参考设计使用一个 7 位长度的地址空间但保留了 16 个地址 所以在 一组总线最多可和 112 个节点通信。保留地址如下表，参考 I²C-bus specification and user manual。

Slave address	R/W bit	Description
0000 000	0	general call address ^[1]
	1	START byte ^[2]
0000 001	X	CBUS address ^[3]
0000 010	X	reserved for different bus format ^[4]
0000 011	X	reserved for future purposes
0000 1XX	X	Hs-mode master code
1111 1XX	1	device ID
1111 0XX	X	10-bit slave addressing

X = don't care; 1 = HIGH; 0 = LOW.

^[1] The general call address is used for several functions including software reset.

^[2] No device is allowed to acknowledge at the reception of the START byte.

^[3] The CBUS address has been reserved to enable the inter-mixing of CBUS compatible and I²C-bus compatible devices in the same system. I²C-bus compatible devices are not allowed to respond on reception of this address.

^[4] The address reserved for a different bus format is included to enable I²C and other protocols to be mixed. Only I²C-bus compatible devices that can work with such formats and protocols are allowed to respond to this address.

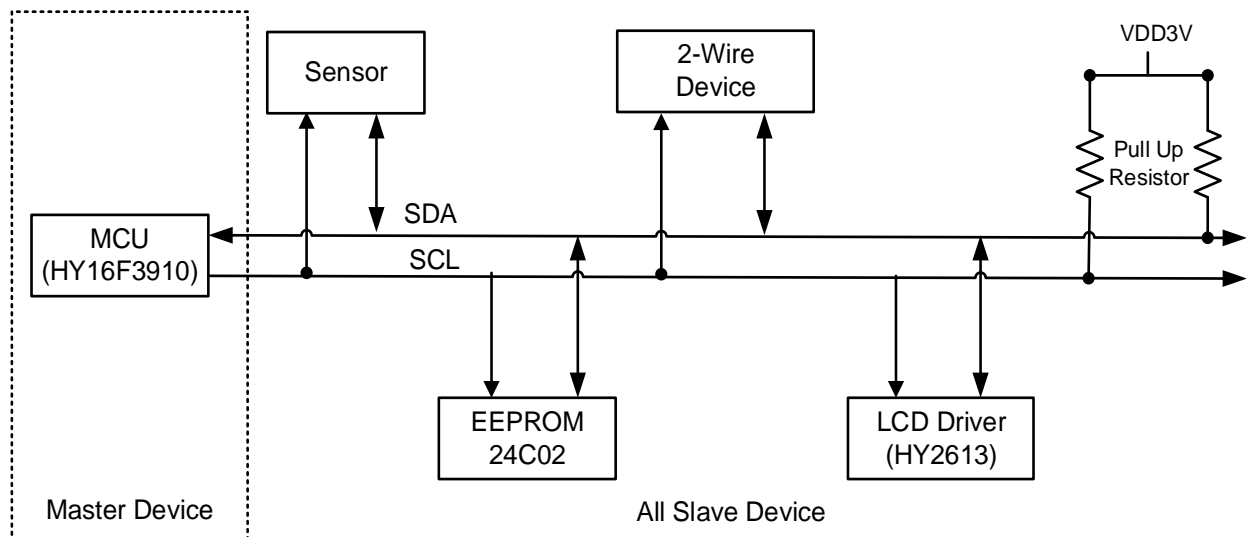


图 27-2 I²C 总线挂接器件图

- 通信 I²C 接口信号

- 开始信号(START): 主端模式 SCL 为高电位。从高电位发出 SDA 到低电位以启动数据传输。
- 数据(DATA)和地址(ADDRESS)信号: I²C 串行接口协议只有当 SCL 是低电位时才需要, SDA 只能依据数据来改变。
- 响应信号(Acknowledge): 接收数据(Slave)是自最初的 8 个位之后才开始。传送数据到一个装置 (Host)是送一个低电位, 表示数据已经被收到。
- 停止信号(STOP): 主端模式 SCL 是高电位。从低电位发出 SDA 到高电位以结束一个数据传输。

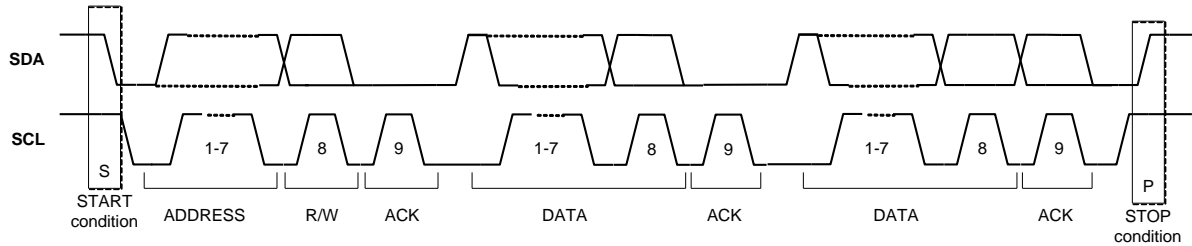


图 27-3 I²C 总线时序图

27.1.2. 数据传输率的计算:

◆ Master Mode

I²C 内部寄存器 CRG[7:0]可以控制主端模式的数据传输率。I²C 总线 SCL 引脚上的串行数据传输速度决定于 I²C 的电路频率源(Clock 可以选择 HSXT 或是 HSRC)和波特率寄存器 CRG[7:0]的数值设定，I²C 串行数据传输率可以由以下公式来决定：

$$\text{Data BuardRate(I2CCK)} = \frac{\text{Clock}}{[4 \times (\text{CRG}[7:0] + 1)]} \quad (\text{式 27-1})$$

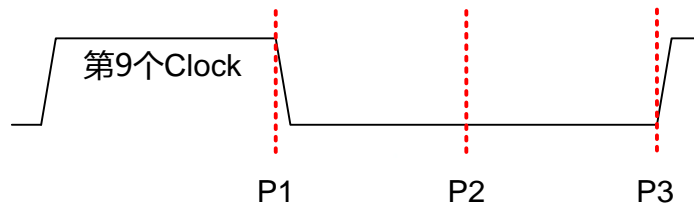
※注意：

上拉电阻大小会影响 Data BuardRate，因芯片等到 High 时才开始计算 High 周期。

I²C Master Mode 与 I²C Slave Mode 之下，SCL 最高能够支持速度为 400kHz。

◆ Slave Mode

当 Master 端使用的是标准硬件 I²C 或者是会判断 SCL 状态才做动的装置时，CRG[7:0]建议填值为 01H。如果 Master 端是使用 I/O 仿真而成的装置，如需配合第九个 Clock 长度可依实际情况调整。公式计算为下



P1~P2的时间决定于ISR中何时将I2C中断标志位写0

P2~P3的时间由CRG[7:0]决定

Min=(CRG+1)*(CPU_CK周期)

Max=2*(CRG+1)*(CPU_CK周期)

27.1.3. 超时控制 Time-out function (Time-Out):

- ◆ Time-out 控制是为了避免 I²C 控制器将 I²C Bus 死锁，I²C Device 在运作的过程中为了提供 MCU 足够的时间处理 I²C 控制器的需求，因此 I²C 控制器在每一个响应位之后都会将 SCL 拉为 Low，使 Master 无法传出下一个频率信号。但是当 MCU 过于忙碌或任何原因无法响应 I²C 控制器的需求时，I²C Bus 的 SCL 将有可能被死锁于 Low。为了避免此情况发生，Time-out 控制器可以根据程序员所设定的 Time-out 条件，侦测 SCL 被本机拉为 Low 的时间，当 Time-out 被判定确立后，I²C 控制器会将 SCL 放掉并且发出中断至 MCU。
- ◆ Time-out 控制电路是以 I²C 频率源 I2CCK 为计数条件，I2CCK 先以 TOPS 所设定之值最多可以作 128 的 Pre-scale，再根据 TOLimit 计数 SCL 被本机设为 Low 的时间(在此先订为 SCLo)，如果 SCLo 于尚未达到 Time-out 时间被释放为 High，则 Time-out 控制器内部的计数器将被重置，并于下次 SCLo 在被拉为 Low 时重新计数；如果 SCLo 超出 Time-out 时间后依然被拉为 Low，则 Time-out 标志位 TOFlag 将被设立，并发出中断信号要求 MCU 处理。
- ◆ Time-out 标志位被设立后，稍后的传送过程中一律响应 NACK 于 I²C Bus。MCU 要清除 Time-out 标志位后方能再度正常使用 I²C 控制电路，Time-out 标志位的清除必须将 TOEn 关闭再开启，使 Time-out 控制电路回复初始状态。

27.1.4. I²C 通信引脚

I²C 总线只有两根线，但芯片给 I²C 模块配置 4 组通信 IO 引脚(一组 IO 引脚包含 SCL/SDA)，是 IO 的复用功能。方便用户可以自由的设置选择不同通信引脚。透过控制器 I2CPTS 0x40844[19:17]、I2CPTEn 0x40844[16]选择及开启相应的通信引脚。在使用 I²C 功能时，必须先开启通信 IO 引脚，对应的 IO 引脚需设置为输入或输出模式。

下表为通信引脚分布表。

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

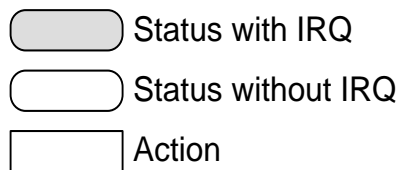
表 27-1 I²C 通信 IO 引脚分布

注意：HY16F3910 产品系列的 I²C 应用，初始化 GPIO 引脚功能为输入或输出模式为选择性设置，使用者在 I²C 初始化流程上可以省略此步骤流程。

27.1.5. 通信 I²C 接口流程

- I²C 串行接口术语

- (SPIA) : 代表对 Action Register(ACT)控制寄存器所下达指令, S 为 Start 指令, P 为 Stop 指令, I 为中断标志位, A 为 Acknowledge 指令。
- SPIA(所对应的寄存器位置为 0x41004[3:0]) : 代表读取 Action Register(ACT)控制寄存器之值, 可以用于判读中断标志位或其它指令是否运作完成。
- STA(所对应的寄存器位置为 0x41004[23:16]) : 读取 Status 寄存器之值, 用以表示目前 I²C 电路运作状态。
- 下列流程图会以图所示之(灰底圆框)、(白底圆框)、(纯白方框), 分别表示 I²C 界面之状态:
灰底圆框 : 表示中断标志位已被设立之 I²C 状态。
白底圆框 : 表示中断标志位未被设立, 需由 MCU 主动读取之 I²C 状态。
纯白方框 : 表示需由 MCU 对 I²C 下达指令。



27.1.6. 通信 I²C Master TX 流程

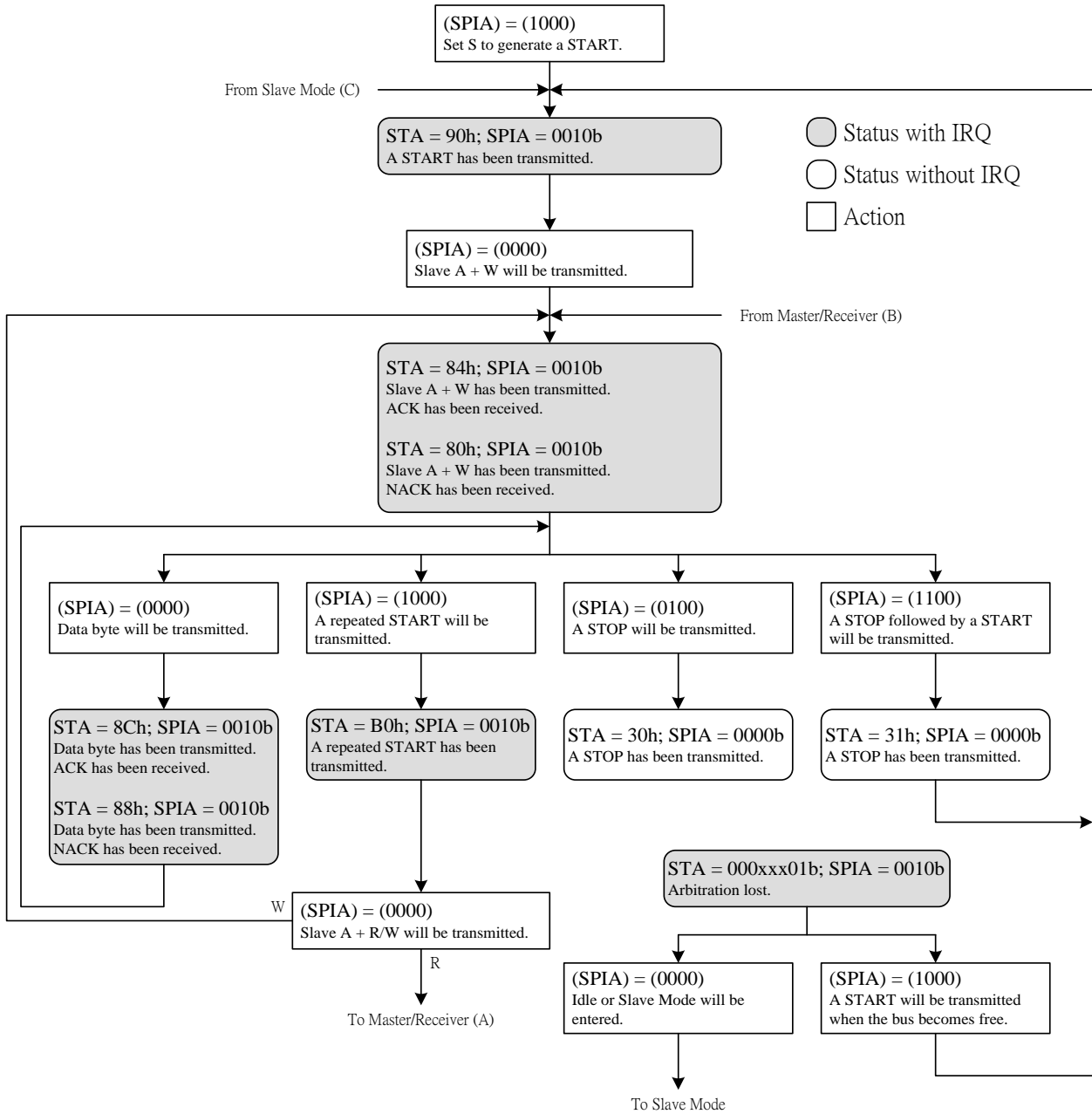


图 27-4 Master Transmitter Mode

27.1.7. 通信 I²C Master RX 流程

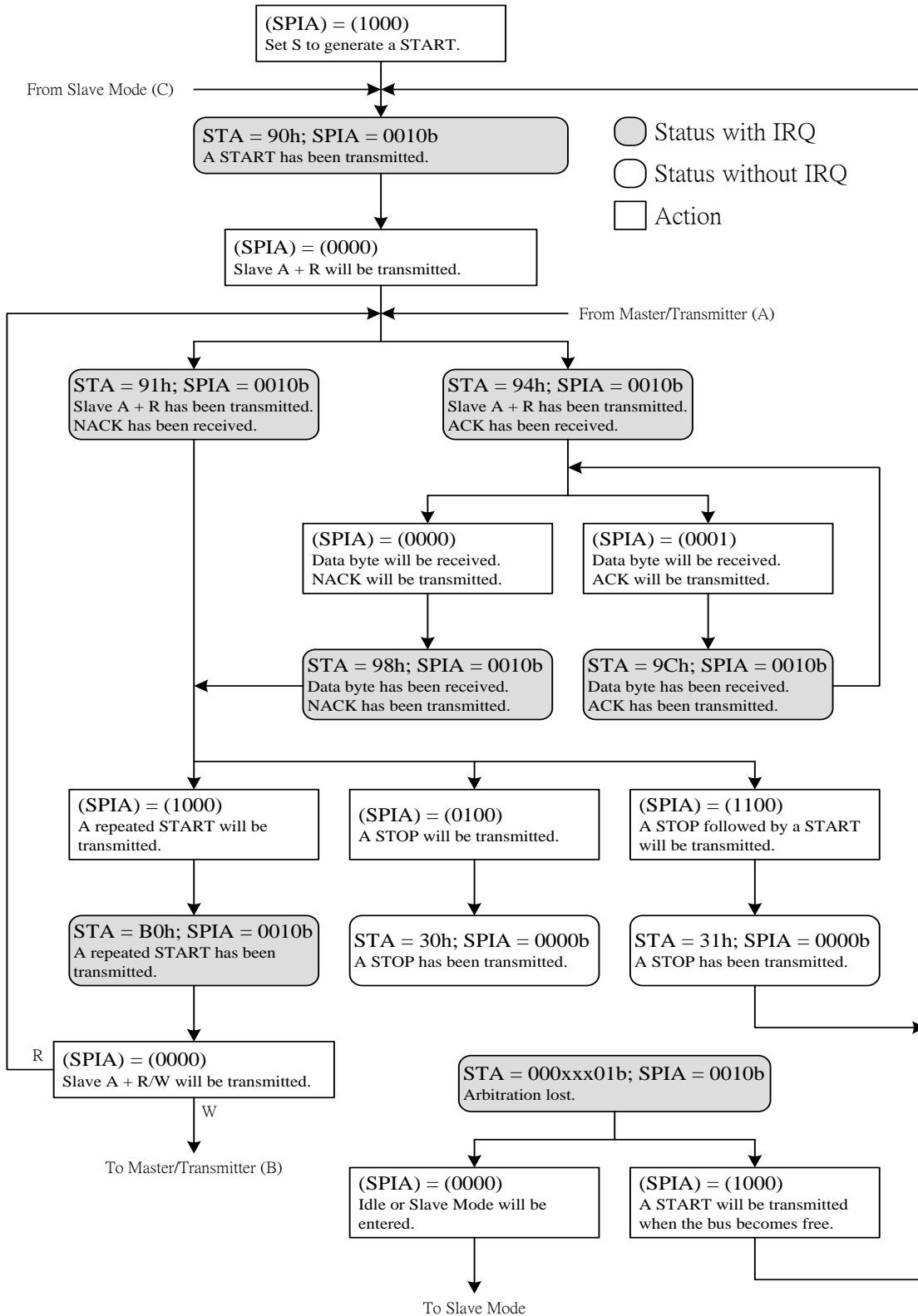


图 27-5 Master Receiver Mode

27.1.8. 通信 I²C Slaver TX 流程

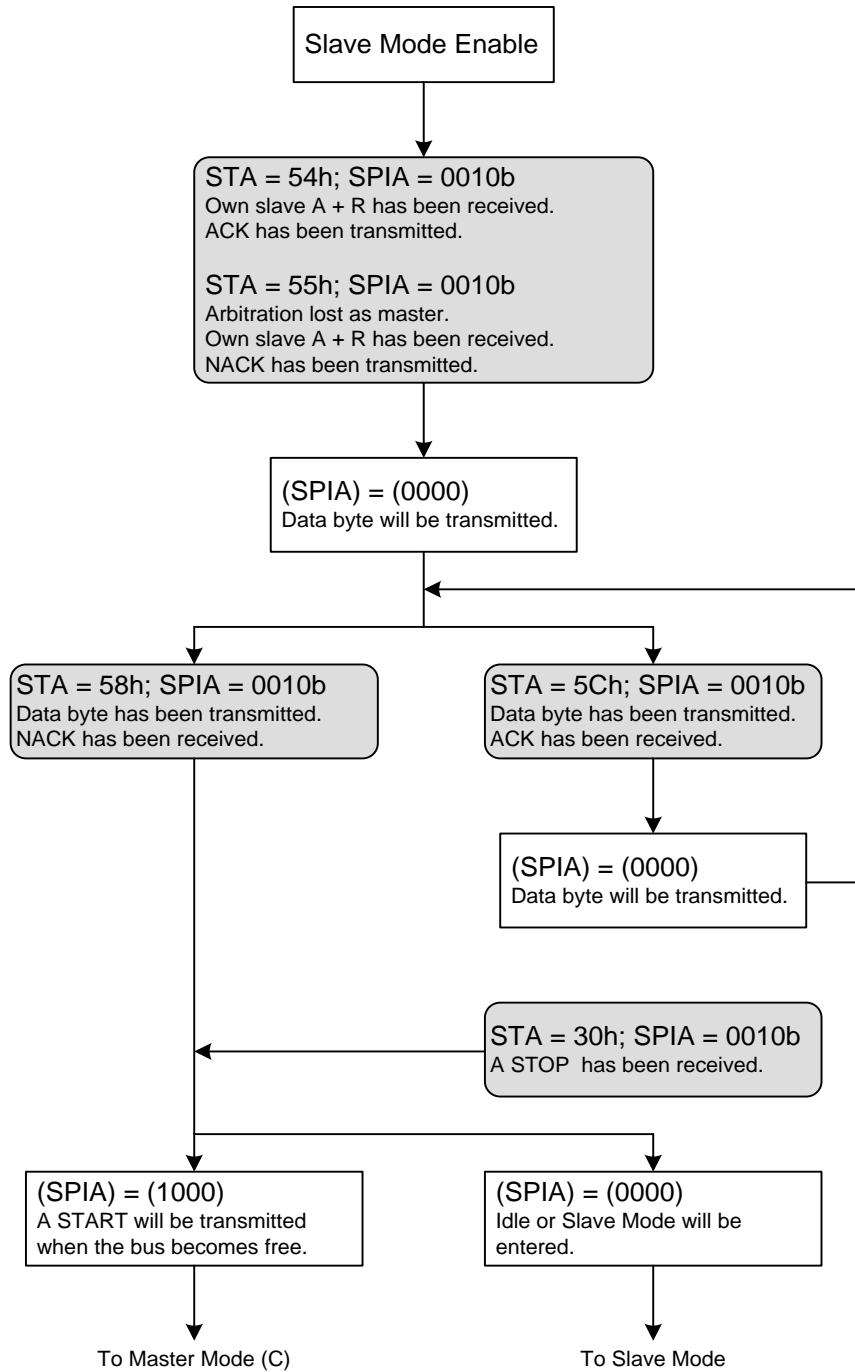


图 27-6 Slave Transmitter Mode

27.1.9. 通信 I²C Slaver RX 流程

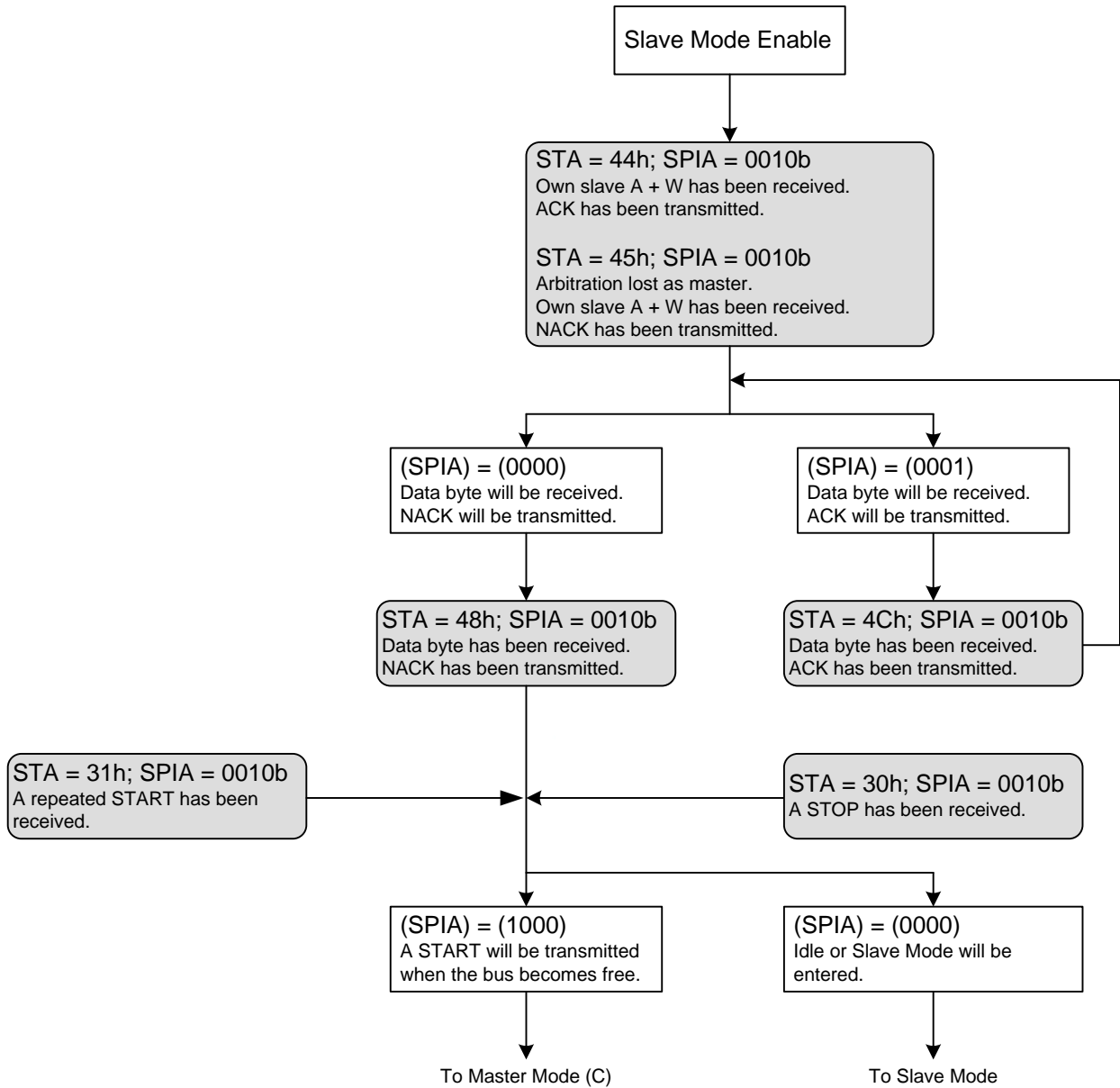


图 27-7 Slave Receiver Mode

27.1.10. 通信 I²C General Call 流程

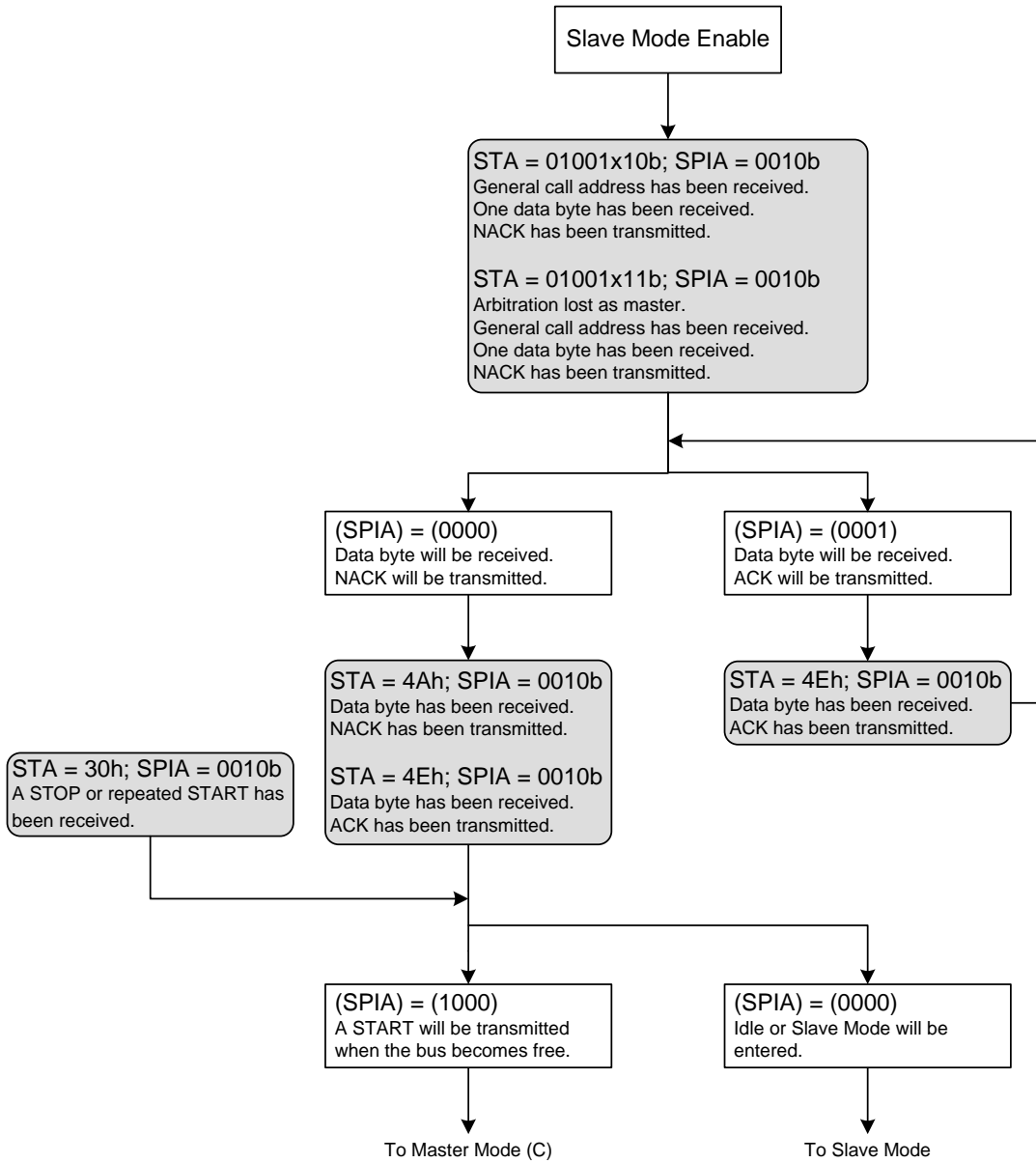


图 27-8 General Call Mode

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

27.2. 寄存器地址

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

27.3. 寄存器功能

27.3.1. I2C 寄存器 0

I2C Base Address + 0x00 (0x41000)							
Symbol	I2CCR0 (I ² C Control Register 0)						
Bit	[31:16]						
名称	RSV						
RW	R-0						
Bit	[15:08]	[07:06]	[05]	[04:03]	[02]	[01]	[00]
名称	MASK	-	TOCKS	-	GCRst	TOEn	I2CEn
RW	R0W-0	-	RW-0	-	RW-0		

Configuration Register (CFG)

位	名称	描述
Bit[05]	TOCKS	Timeout Base Clock Control
		0 Base on I ² C IP Clock
		1 Base on CRG Clock
Bit[02]	GCRst	全呼复位使能控制(详细说明请参考 0 I2C)
		0 关闭
		1 开启
Bit[01]	TOEn	超时复位功能开启控制
		0 关闭
		1 开启
Bit[00]	I2CEn	I2C 功能开启控制
		0 关闭
		1 开启

注意事项：当 I2CEn 关闭时，将关闭 I²C 内部的 Clock，除了 Configuration Register 可以进行写入动作，其余寄存器将无法写入数据。

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



27.3.2. I2C 寄存器 1

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]
名称	MASK	MAct	SAct	Rx P/Sr	R/W	DF	A/NA	GC	ARB
RW	R0W-0	R-0							
Bit	[15:08]	[07]	[06]	[05]	[04]	[03]	[02]	[01]	[00]
名称	MASK	SEn	10bEn	3BEn	EIRQFlag	START	STOP	IRQFlag	A/NA
RW	R0W-0	RW-0							

位	名称	描述	
Bit[23]	MAct	主机模式启用标志位	
		0	未启用
		1	启用
Bit[22]	SAct	从机模式启用标志位	
		0	未启用
		1	启用
Bit[21]	Rx P/Sr	接收停止或重新开始标志位	
		0	正常
		1	接收停止或重新开始标志位已被发送或接收。
Bit[20]	R/W	读写状态标志位	
		0	写命令已被发送或接收
		1	读命令已被发送或接收
Bit[19]	DF	资料标志位	
		0	正常
		1	I ² C 数据已被发送或接收
Bit[18]	A/NA	应答信号(ACK)状态标志位	
		0	应答信号(ACK)未被发送或接收
		1	应答信号(ACK)已被发送或接收
Bit[17]	GC	全呼状态标志位	
		0	正常
		1	当前正进行全呼操作
Bit[16]	ARB	仲裁漏失标志位	
		0	正常
		1	仲裁漏失
Bit[07]	SEn	(I ² C Slave)从机模式开启控制	
		0	关闭
		1	开启
Bit[06]	10bEn	(I ² C Slave)从机 10 位地址码模式开启控制	

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述	
		0	关闭
		1	开启 10 位地址码模式
Bit[05]	3BEn	(I ² C Slave)从机发送 3 笔数据功能开启控制	
		0	关闭
		1	开启从机连续发送 3 笔数据功能
Bit[04]	EIRQFlag	错误标志位，与错误中断标志位 I2CEIF 相关联，先清零该位才能清零 I2CEIF	
		0	正常
		1	发生超时或接收到意外的起始(停止)信号或仲裁失败
Bit[03]	START(S)	起始信号控制位(S)	
		0	正常
		1	于 I ² C 总在线产生起始信号
Bit[02]	STOP(P)	停止信号控制位(P)	
		0	正常
		1	于 I ² C 总在线产生停止信号
Bit[01]	IRQFlag(I)	中断控制位(I)	
		0	正常；
		1	响应中断，器件接收到 9 个 clock 后响应中断，并将 SCL 拉低直到该位被清零后释放 SCL 信号线；写入 0 将会清零器件状态控制位，使 I ² C 往下一个状态执行。
Bit[00]	A/NA(A)	应答信号回复控制位(A)	
		0	未回复 ACK 或回复 NACK
		1	应答信号(ACK)已回复。

27.3.3. I2C 寄存器 2

I2C Base Address + 0x08 (0x41008)				
Symbol	I2CCR2 (I2C Control Register 2)			
Bit	[31:24]	[23:16]		
名称	MASK	CRG		
RW	R0W-0	RW-0		
Bit	[15:08]	[07]	[06:04]	[03:00]
名称	MASK	TOFlag	TOPS	TOLimit
RW	R0W-0	R-0	RW-0	

位	名称	描述
Bit[23:16]	CRG	I ² C 总线数据波特率控制寄存器
		0~255 可设定
		I ² C 总线的数据波特率决定于 I ² C 的频率源和波特率控制寄存器 CRG 的值； I ² C 总线数据波特率可由以下公式计算得到：

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



		$\text{Data BuardRate}(I2CCK) = \frac{\text{Clock}}{[4 \times (\text{CRG}[7:0] + 1)]}$ <p>Note: Clock 可以选择为 HSXT 或是 HSRC 频率源。</p>																																					
Bit[07]	TOFlag	超时标志位 <table border="1"> <tr> <td>0</td> <td>正常</td> </tr> <tr> <td>1</td> <td>I²C 总线时钟线高低电位控制超时</td> </tr> </table>		0	正常	1	I ² C 总线时钟线高低电位控制超时																																
0	正常																																						
1	I ² C 总线时钟线高低电位控制超时																																						
Bit[6:4]	TOPS	超时时钟除频器设置 <table border="1"> <thead> <tr> <th>TOPS[2:0]</th> <th>Pre-scale</th> <th>TOPS[2:0]</th> <th>Pre-scale</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>/ 1</td> <td>100</td> <td>/ 16</td> </tr> <tr> <td>001</td> <td>/ 2</td> <td>101</td> <td>/ 32</td> </tr> <tr> <td>010</td> <td>/ 4</td> <td>110</td> <td>/ 64</td> </tr> <tr> <td>011</td> <td>/ 8</td> <td>111</td> <td>/ 128</td> </tr> </tbody> </table> <p>IF TOCKS=0b, CLKPS = I2CCK / TOPS Else IF TOCKS=1b, CLKPS = I2CCK / TOPS / (CRG+1)</p>		TOPS[2:0]	Pre-scale	TOPS[2:0]	Pre-scale	000	/ 1	100	/ 16	001	/ 2	101	/ 32	010	/ 4	110	/ 64	011	/ 8	111	/ 128																
TOPS[2:0]	Pre-scale	TOPS[2:0]	Pre-scale																																				
000	/ 1	100	/ 16																																				
001	/ 2	101	/ 32																																				
010	/ 4	110	/ 64																																				
011	/ 8	111	/ 128																																				
Bit[3:0]	TOLimit	超时上限值设置 <table border="1"> <thead> <tr> <th>TOLimit[3:0]</th> <th>Limit</th> <th>TOLimit[3:0]</th> <th>Limit</th> </tr> </thead> <tbody> <tr> <td>0000</td> <td>1 x CLKPS Cycle</td> <td>1000</td> <td>9 x CLKPS Cycle</td> </tr> <tr> <td>0001</td> <td>2 x CLKPS Cycle</td> <td>1001</td> <td>10 x CLKPS Cycle</td> </tr> <tr> <td>0010</td> <td>3 x CLKPS Cycle</td> <td>1010</td> <td>11 x CLKPS Cycle</td> </tr> <tr> <td>0011</td> <td>4 x CLKPS Cycle</td> <td>1011</td> <td>12 x CLKPS Cycle</td> </tr> <tr> <td>0100</td> <td>5 x CLKPS Cycle</td> <td>1100</td> <td>13 x CLKPS Cycle</td> </tr> <tr> <td>0101</td> <td>6 x CLKPS Cycle</td> <td>1101</td> <td>14 x CLKPS Cycle</td> </tr> <tr> <td>0110</td> <td>7 x CLKPS Cycle</td> <td>1110</td> <td>15 x CLKPS Cycle</td> </tr> <tr> <td>0111</td> <td>8 x CLKPS Cycle</td> <td>1111</td> <td>16 x CLKPS Cycle</td> </tr> </tbody> </table> <p>Timeout 的发生是以 CLKPS 计数 TOLimit + 1 次后触发</p>		TOLimit[3:0]	Limit	TOLimit[3:0]	Limit	0000	1 x CLKPS Cycle	1000	9 x CLKPS Cycle	0001	2 x CLKPS Cycle	1001	10 x CLKPS Cycle	0010	3 x CLKPS Cycle	1010	11 x CLKPS Cycle	0011	4 x CLKPS Cycle	1011	12 x CLKPS Cycle	0100	5 x CLKPS Cycle	1100	13 x CLKPS Cycle	0101	6 x CLKPS Cycle	1101	14 x CLKPS Cycle	0110	7 x CLKPS Cycle	1110	15 x CLKPS Cycle	0111	8 x CLKPS Cycle	1111	16 x CLKPS Cycle
TOLimit[3:0]	Limit	TOLimit[3:0]	Limit																																				
0000	1 x CLKPS Cycle	1000	9 x CLKPS Cycle																																				
0001	2 x CLKPS Cycle	1001	10 x CLKPS Cycle																																				
0010	3 x CLKPS Cycle	1010	11 x CLKPS Cycle																																				
0011	4 x CLKPS Cycle	1011	12 x CLKPS Cycle																																				
0100	5 x CLKPS Cycle	1100	13 x CLKPS Cycle																																				
0101	6 x CLKPS Cycle	1101	14 x CLKPS Cycle																																				
0110	7 x CLKPS Cycle	1110	15 x CLKPS Cycle																																				
0111	8 x CLKPS Cycle	1111	16 x CLKPS Cycle																																				

TimeOut 可以设定为 4.147MHz/128/76/15=35.2ms 等级

27.3.4. I2C 寄存器 3

Slave ID0 (SID0)

I2C Base Address + 0x0C (0x4100C)				
I2CCR3 (I2C Control Register 3)				
Symbol	[31:24]		[23:16]	
名称	SID1 MASK		SID0 MASK	
RW	ROW-0		ROW-0	
Bit	[15:09]	[08]	[07:01]	[00]
名称	SID1	VD1	SID0	VD0
RW	RW-0		RW-0	

位	名称	描述
Bit[31:24]	SID1 MASK	SID1 MASK
		0 关闭
		1 开启
Bit[23:16]	SID0 MASK	SID0 MASK
		0 关闭
		1 开启
Bit[15:9]	SID1	SID1 从机地址码设置

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述	
		0	置 0
		1	置 1
Bit[08]	VD1	从机地址码有效控制位，写入地址码时该位必须为 1	
		0	从机地址码无效
		1	从机地址码有效
Bit[7:1]	SID0	SID0 从机地址码设置	
		0	置 0
		1	置 1
Bit[00]	VD0	从机地址码有效控制位，写入地址码时该位必须为 1	
		0	从机地址码无效
		1	从机地址码有效

注意事项：

当 I²C 工作于从机(Slave)模式时，提供两组 Slave ID Register，可以应用于同时存在两组从机模式。例如：VD0 设定为 0x30，VD1 设定为 0x32。也可以利用两组 Slave ID 组合出 10 位从机寻址模式。

I²C Slave 内建 Slave ID Comparator 用以比较由 I²C Bus 上所接收的 Slave ID 是否与 Slave ID Register 上所设定的 ID 一致性。当比较结果一致时就会发出 Slave Mode 中断信号通知系统准备服务主机的需求，并且将 I²C Bus 的频率信号 SCL 拉为 Low 使主机等待本机系统有所响应。而 Slave Controller 将等待系统有所响应才会将 SCL 的控制权交还主机。因此为了避免本机系统异常而长时间占用 I²C Bus，使用者必须适当的设定超时控制器(Time-out Controller)，以便于系统太久无响应时，由 Slave Controller 自行解除 SCL 的控制，并且发出错误状态中断信号。

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

27.3.5. I2C 寄存器 4

I2C Base Address + 0x10 (0x41010)			
Symbol	I2CCR4 (I2C Control Register 4)		
Bit	[31:16]		
名称	RSV		
RW	R-0		
Bit	[15:08]	[07:01]	[00]
名称	-	Rx A7-1/D7-1	RW/D0
RW	-	R-X	

Receiver Data Buffer (RxAD)

位	名称	描述
Bit[7:1]	Rx A7-1/D7-1	接收地址或数据寄存器 RX[7:0]
		0 置 0
		1 置 1
Bit[00]	RW/D0	接收到的数据是读写命令或数据的第 0 位值
		0 置 0
		1 置 1

27.3.6. I2C 寄存器 5

I2C Base Address + 0x14 (0x41014)			
Symbol	I2C 5 (I2C Control Register 5)		
Bit	[31:24]	[23:17]	[16]
名称	RSV.	TX2 A7-1/D7-1	Flag/D0
RW	R-0	RW-X	
Bit	[15:08]	[7:1]	[0]
名称	TX1 A7-0/D7-0	TX0 A7-1/D7-1	RW/D0
RW	RW-X		

Transmitter Data Buffer 0 (TxAD)

位	名称	描述
Bit[23:17]	TX2 A7-1/D7-1	发送寄存器 2, 发送地址或数据 data[7:1]的值
		0 置 0
		1 置 1
Bit[16]	Flag/D0	发送寄存器 2, 发送读写命令或数据 data[0]的值
		0 置 0
		1 置 1
Bit[15:8]	TX1 A7-0/D7-0	发送寄存器 1, 发送地址或数据 data[7:0]的值
		0 置 0
		1 置 1
Bit[7:1]	TX0 A7-1/D7-1	发送寄存器 0, 发送地址或数据 data[7:1]的值

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述	
		0	置 0
		1	置 1
Bit[00]	RW/D0	发送寄存器 0, 发送读写命令或数据 data[0]的值	
		0	置 0
		1	置 1

注意事项：

在通信过程中，当没有要传输时，数据传输寄存器必须被设置为 0xFF。

因为最低位，长期拉 Low 容易让 SDA 总线锁住在 Low(0)。

当 I²C 工作于从机模式(Slave)，如果工作于标准 1 byte 模式，是利用 TX0 A7-1/D7-1 进行单笔数据传送，如果工作 3 bytes 模式，是利用 TX0 A7-1/D7-1，TX1 A7-0/D7-0, TX2 A7-1/D7-1 进行数据传送。

TX1 A7-0/D7-0 : Transmitter 2nd Data Buffer Bit7-0 for 3 Byte Mode Only
于 3 byte 模式数据传送完成后，此寄存器自动设为 FFh。

TX2 A7-0/D7-0 : Transmitter 3rd Data Buffer bit7-1 for 3 Byte mode only.
Flag/D0 : Transmitter 3rd Flag or Data buffer bit 0 for 3 byte mode only
于 3 byte 模式数据传送完成后，此寄存器自动设为 FFh。

27.4. I²C 使用说明

27.4.1. I²C 初始化说明

I²C Master Mode Initial

- 前置作业：配置 I2CCK，配置 GPIO 为 I²C 模式。
- 配置 I2CEn 使能 I²C 与 I²C 内部频率电路。
- 配置 CRG 使 I²C 运作于所需之传输率。
- 配置 TOPS 与 TOLimit 设定 Clock Stretching 时间限制。
- 配置 I2CEn 与 TOEn 使能 I²C 与 Time-out 控制电路。
- I²C Master Mode Initial 完成，使用者可以利用 ACT,STA,RxAD 与 TxAD 等寄存器完成所需数据传送。

I²C Slave Mode Initial

- 前置作业：配置 I2CCK，配置 GPIO 为 I²C 模式。
- 配置 I2CEn 使能 I²C 与 I²C 内部频率电路。
- 配置 TOPS 与 TOLimit 设定 Clock Stretching 时间限制。
- 配置 Slave IDx 与 Slave IDx Mask 做为 Slave Mode ID 之比较用。
- 配置 SEn 使能 Slave Mode 电路。
- 配置 I2CEn 与 TOEn 使能 I²C 与 Time-out 控制电路。
- I²C Slave Mode Initial 完成，使用者可以利用 ACT,STA,RxAD 与 TxAD 等寄存器完成所需数据传送。

10 Bit Addressing I²C Slave Mode Initial

- 前置作业：配置 I2CCK，配置 GPIO 为 I²C 模式。
- 配置 I2CEn 使能 I²C 与 I²C 内部频率电路。
- 配置 TOPS 与 TOLimit 设定 Clock Stretching 时间限制。
- 配置 Slave IDx 与 Slave IDx Mask 做为 Slave Mode ID 之比较用。
- 配置 SEn 与 10bEn 使能 Slave Mode 电路与 10 Bit Addressing 电路。
- 配置 I2CEn 与 TOEn 使能 I²C 与 Time-out 控制电路。
- I²C Slave Mode Initial 完成，使用者可以利用 ACT,STA,RxAD 与 TxAD 等寄存器完成所需数据传送。

3 Byte I²C Slave Mode Initial

- 前置作业：配置 I2CCK，配置 GPIO 为 I²C 模式。
- 配置 I2CEn 使能 I²C 与 I²C 内部频率电路。
- 配置 TOPS 与 TOLimit 设定 Clock Stretching 时间限制。
- 配置 Slave IDx 与 Slave IDx Mask 做为 Slave Mode ID 之比较用。
- 配置 SEn 与 3BEn 使能 Slave Mode 电路与 3 Byte TX 控制电路。
- 配置 I2CEn 与 TOEn 使能 I²C 与 Time-out 控制电路。
- I²C Slave Mode Initial 完成，使用者可以利用 ACT,STA,RxAD,TX0,TX1,TX2 等寄存器完成所需数据传送。

27.4.2. I²C 操作流程说明

下图为一个常见的 I²C EEPROM(24C02)数据控制格式。

以下提供 I²C Master Write & Read 来读取与写入 EEPROM 的控制解说来让使用者认识 I²C 操作流程。

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

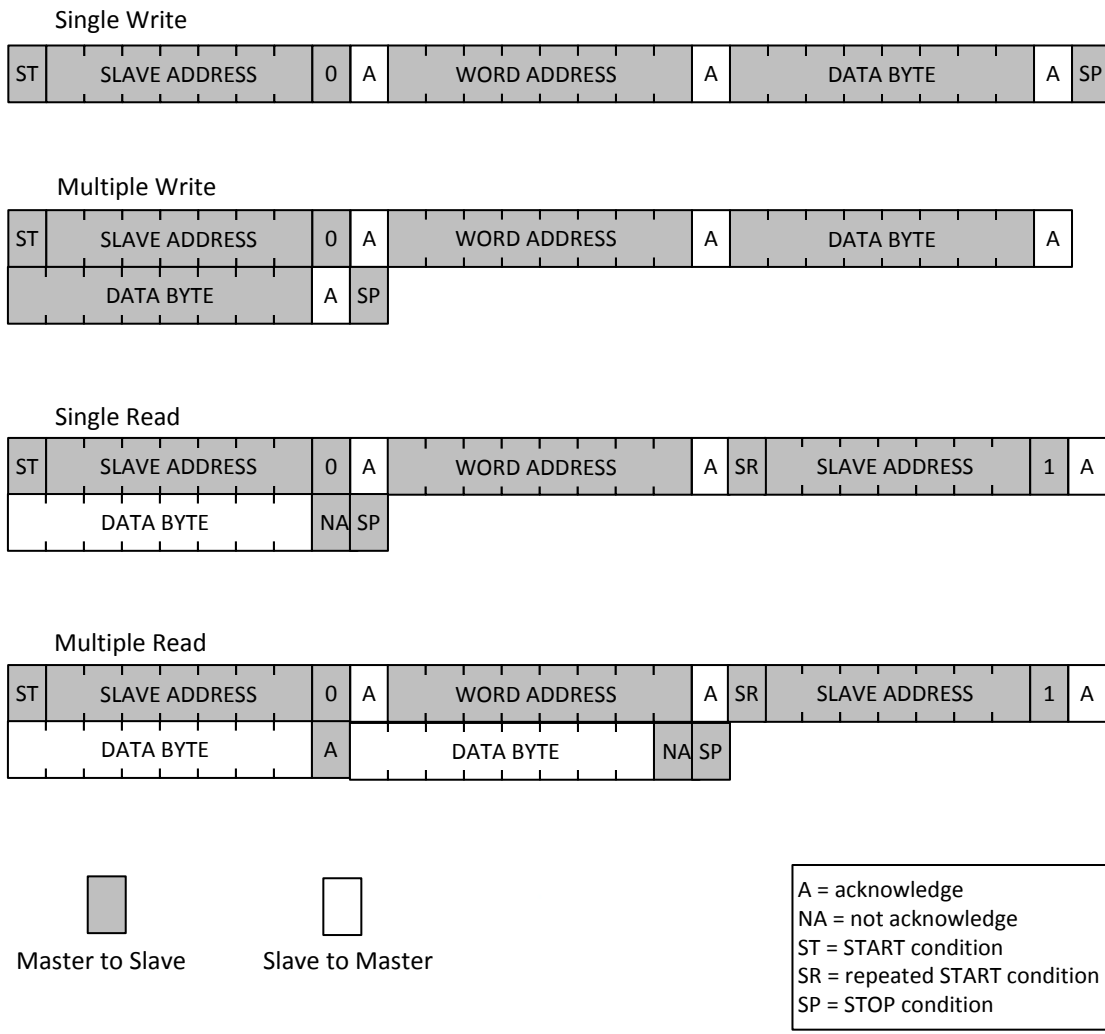


图 27-9 I²C EEPROM 数据控制格式

I²C Master TX 端流程说明 :

可以参考以下文字说明，清楚了解到 I²C Master TX 端 I²C Single Write 的操作流程。

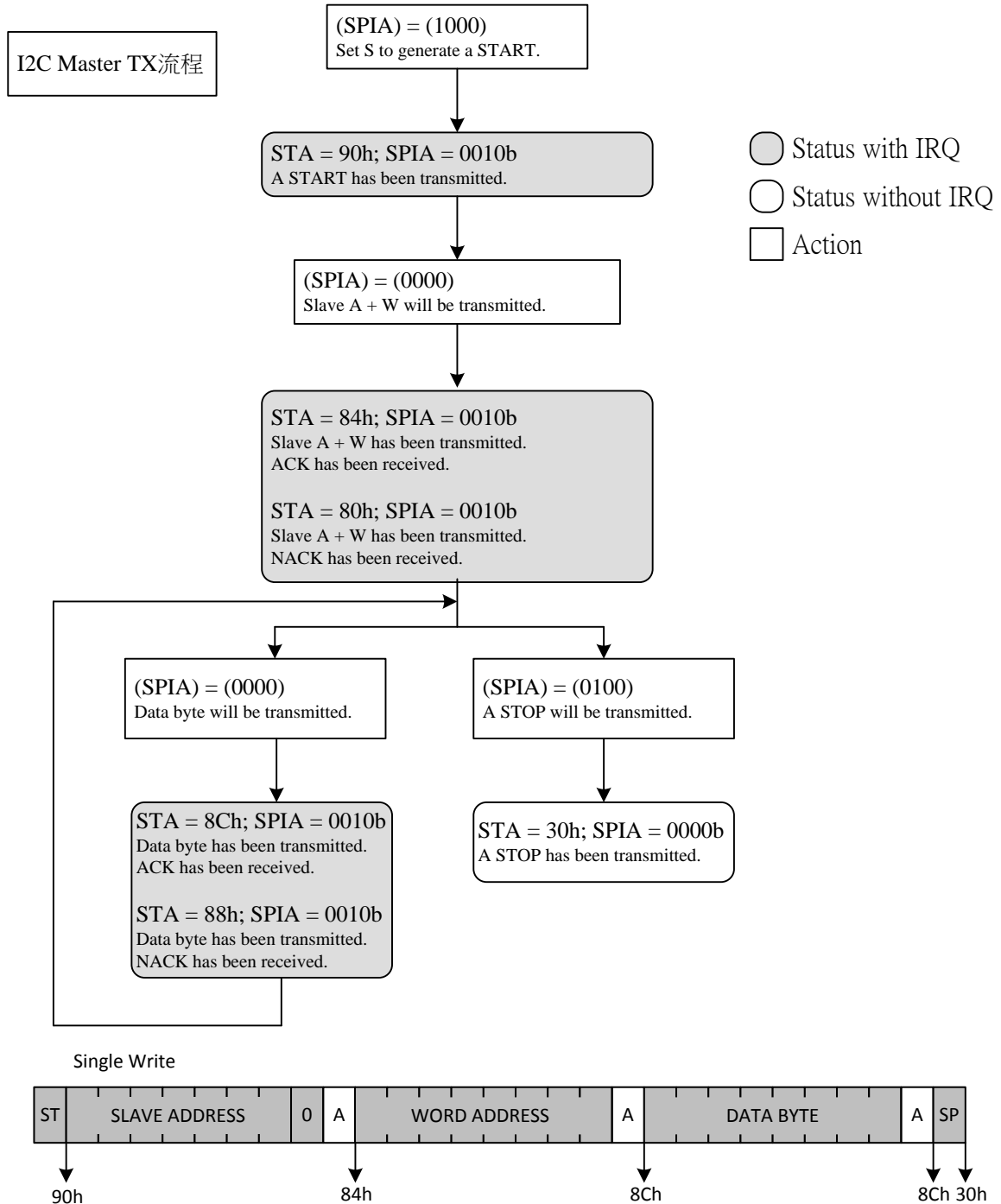


图 27-10 I²C Master TX 状态之流程图

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

1. 首先使用纉康 C 函式库指令 `DrvI2C_Ctrl(1,0,0,0)` 设定(SPIA)=(1000) 完成 START condition , 当执行完指令 `DrvI2C_Ctrl(1,0,0,0)` , 可以使用示波器从 SCL 和 SDA 脚位观察出 I²C START 波形信号已经送出。
2. 使用纉康 C 函式库指令 `DrvI2C_GetStatusFlag` 确认 STA 状态是否为 90h , 如果为 90h , 进入 90h 状态。在 90h 状态内使用指令 `DrvI2C_WriteData` 填入 SLAVE ADDRESS 并且使用指令 `DrvI2C_Ctrl(0,0,0,0)` 设定(SPIA)=(0000) , 当在 90h 状态内执行完 `DrvI2C_Ctrl(0,0,0,0)` , 可以使用示波器从 SCL 和 SDA 脚位观察 I²C SLAVE ADDRESS 波形信号已经送出。
3. 使用纉康 C 函式库指令 `DrvI2C_GetStatusFlag` 确认 STA 状态是否为 84h , 如果为 84h , 代表 Slave 已经回 ACK , 进入 84h 状态。 在此状态内使用指令 `DrvI2C_WriteData` 填入 WORD ADDRESS 并且使用指令 `DrvI2C_Ctrl(0,0,0,0)` 设定(SPIA)=(0000) , 当在 84h 状态内执行完 `DrvI2C_Ctrl(0,0,0,0)` , 可以从 SCL 和 SDA 脚位观察 I²C WORD ADDRESS 波形信号已经送出。
4. 使用纉康 C 函式库指令 `DrvI2C_GetStatusFlag` 确认 STA 状态是否为 8Ch , 如果为 8Ch , 代表 Slave 已经回 ACK , 进入 8Ch 状态。 在此状态内 , 使用指令 `DrvI2C_WriteData` 填入 DATA BYTE 并且使用指令 `DrvI2C_Ctrl(0,0,0,0)` 设定(SPIA)=(0000) , 当在 8Ch 状态内执行完 `DrvI2C_Ctrl(0,0,0,0)` , 可以从 SCL 和 SDA 脚位观察 I²C DATA BYTE 波形信号已经送出。
5. 使用纉康 C 函式库指令 `DrvI2C_GetStatusFlag` 确认 STA 状态是否为 8Ch , 如果为 8Ch , 代表 Slave 已经回 ACK。此时还是进入 8Ch 状态 在此状态内 因为已经不做任何数据传送 所以可以使用指令 `DrvI2C_Ctrl(0,1,0,0)` 设定(SPIA)=(0100) , 当在 8Ch 状态内执行完 `DrvI2C_Ctrl(0,1,0,0)` , 可以从 SCL 和 SDA 脚位观察 I²C STOP 波形信号已经产生。
6. 使用纉康 C 函式库指令 `DrvI2C_GetStatusFlag` 确认 STA 状态是否为 30h , 如果为 30h , 代表第一笔数据传送已经完成。

I²C Master TX&RX 端流程说明：

可以参考以下文字说明，清楚了解到 I²C Master TX&RX 与 I²C Single Read 的操作流程。

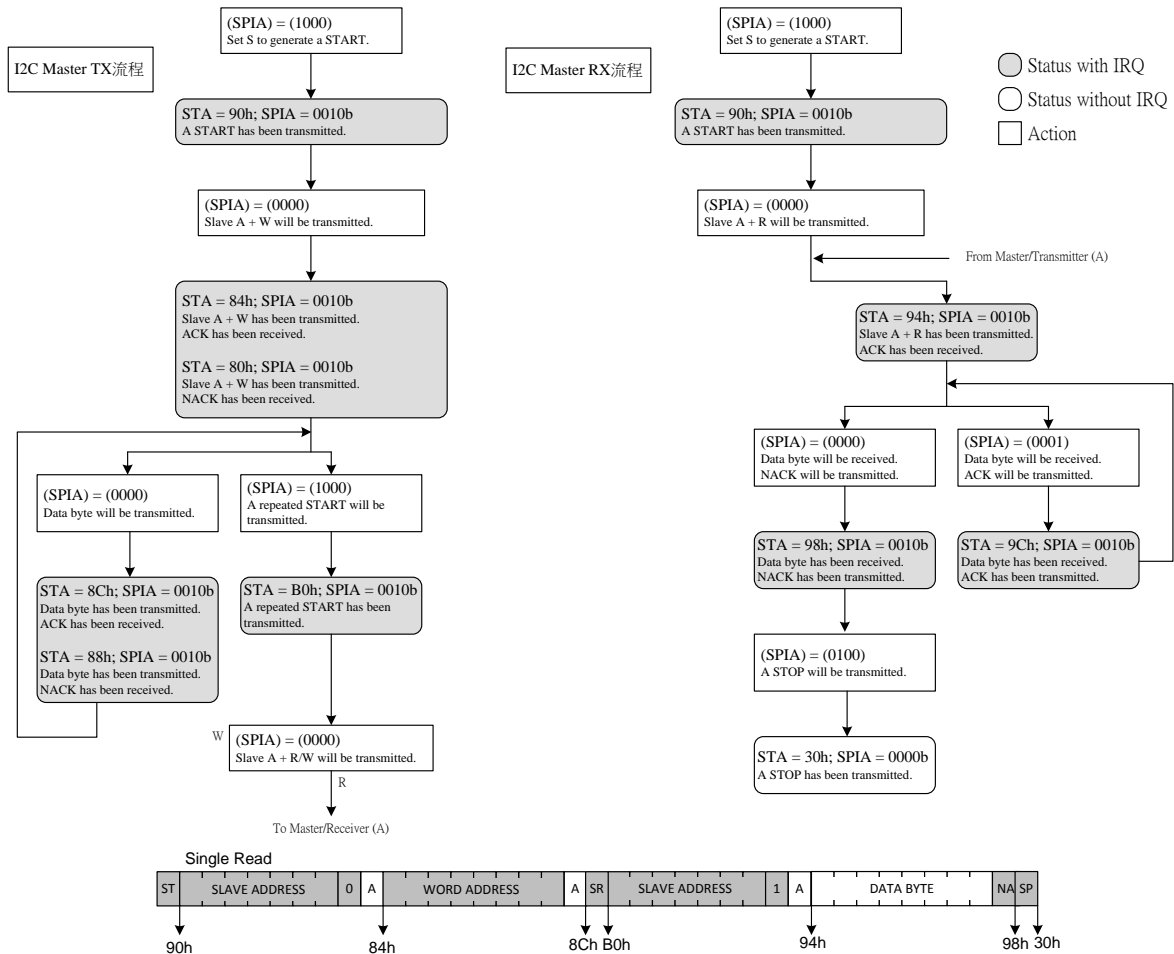


图 27-11 I²C Master TX&RX 状态之流程图

1. 在执行 I²C Master RX 流程之前，还是先需要做 I²C Master TX 流程。首先使用宏康 C 函式库指令 DrvI2C_Ctrl(1,0,0,0)设定(SPIA)=(1000) START condition，当执行完指令 DrvI2C_Ctrl(1,0,0,0)，可以从 SCL 和 SDA 脚位观察出 I²C START 波形信号已经送出。
2. 使用宏康 C 函式库指令 DrvI2C_GetStatusFlag 确认 STA 状态是否为 90h，如果为 90h，进入 90h 状态。在 90h 状态内使用指令 DrvI2C_WriteData 填入 SLAVE ADDRESS 并且使用指令 DrvI2C_Ctrl(0,0,0,0)设定 (SPIA)=(0000)，当在 90h 状态内执行完 DrvI2C_Ctrl(0,0,0,0)，可以从 SCL 和 SDA 脚位观察 I2C SLAVE ADDRESS 波形信号已经送出。
3. 使用宏康 C 函式库指令 DrvI2C_GetStatusFlag 确认 STA 状态是否为 84h，如果为 84h，代表 Slave 已经回 ACK，进入 84h 状态，在此状态内使用指令 DrvI2C_WriteData 填入 WORD ADDRESS 并且使用指令 DrvI2C_Ctrl(0,0,0,0)设定(SPIA)=(0000)，当在 84h 状态内执行完 DrvI2C_Ctrl(0,0,0,0)，可以从 SCL 和 SDA 脚位观察 I²C WORD ADDRESS 波形信号已经送出。
4. 使用宏康 C 函式库指令 DrvI2C_GetStatusFlag 确认 STA 状态是否为 8Ch，如果为 8Ch，代表 Slave 已经回

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



ACK, 进入 8Ch 状态。在此状态内使用指令 Drvl2C_Ctrl(1,0,0,0)设定(SPIA)=(1000), 当在 8Ch 状态内执行完 Drvl2C_Ctrl(1,0,0,0), 可以从 SCL 和 SDA 脚位观察 I²C repeated start 波形信号已经送出。

5. 使用彙康 C 函式库指令 Drvl2C_GetStatusFlag 确认 STA 状态是否为 B0h, 如果为 B0h, 代表 repeated start 信号已经送出, 使用指令 Drvl2C_WriteData 填入 SLAVE ADDRESS+1 并且使用指令 Drvl2C_Ctrl(0,0,0,0)设定(SPIA)=(0000), 当在 B0h 状态内执行完 Drvl2C_Ctrl(0,0,0,0), 可以从 SCL 和 SDA 脚位观察 I²C SLAVE ADDRESS+1 波形信号已经送出, 此时进入 I²C Master RX 流程。

6. 使用彙康 C 函式库指令 Drvl2C_GetStatusFlag 确认 STA 状态是否为 94h, 如果为 94h, 代表 Master 端已经收到 SLAVE ADDRESS+1 所回的 ACK, 进入 94h 状态, 使用指令 Drvl2C_Ctrl(0,0,0,0)设定(SPIA)=(0000), 当在 94h 状态内执行完 Drvl2C_Ctrl(0,0,0,0), 可以从 SCL 和 SDA 脚位观察 I²C Slave DATA BYTE 波形信号已经送出。

7. 使用彙康 C 函式库指令 Drvl2C_GetStatusFlag 确认 STA 状态是否为 98h, 如果为 98h, 代表 Master 端已经收到 Slave 端所回的 DATA BYTE 数据并且 Master 已经送出 NACK 信号给 Slave 端。使用指令 Drvl2C_ReadData 读回 Slave 端所送出的 DATA BYTE 数据并且设定(SPIA)=(0100), 送出 Stop 结束信号。

8. 使用彙康 C 函式库指令 Drvl2C_GetStatusFlag 确认 STA 状态是否为 30h, 如果为 30h, 代表第一笔数据读取已经完成。

I²C Slave RX 端流程说明 :

可以参考以下文字说明，清楚了解到 I²C Slave RX 端与 I²C Single Write 的操作流程。

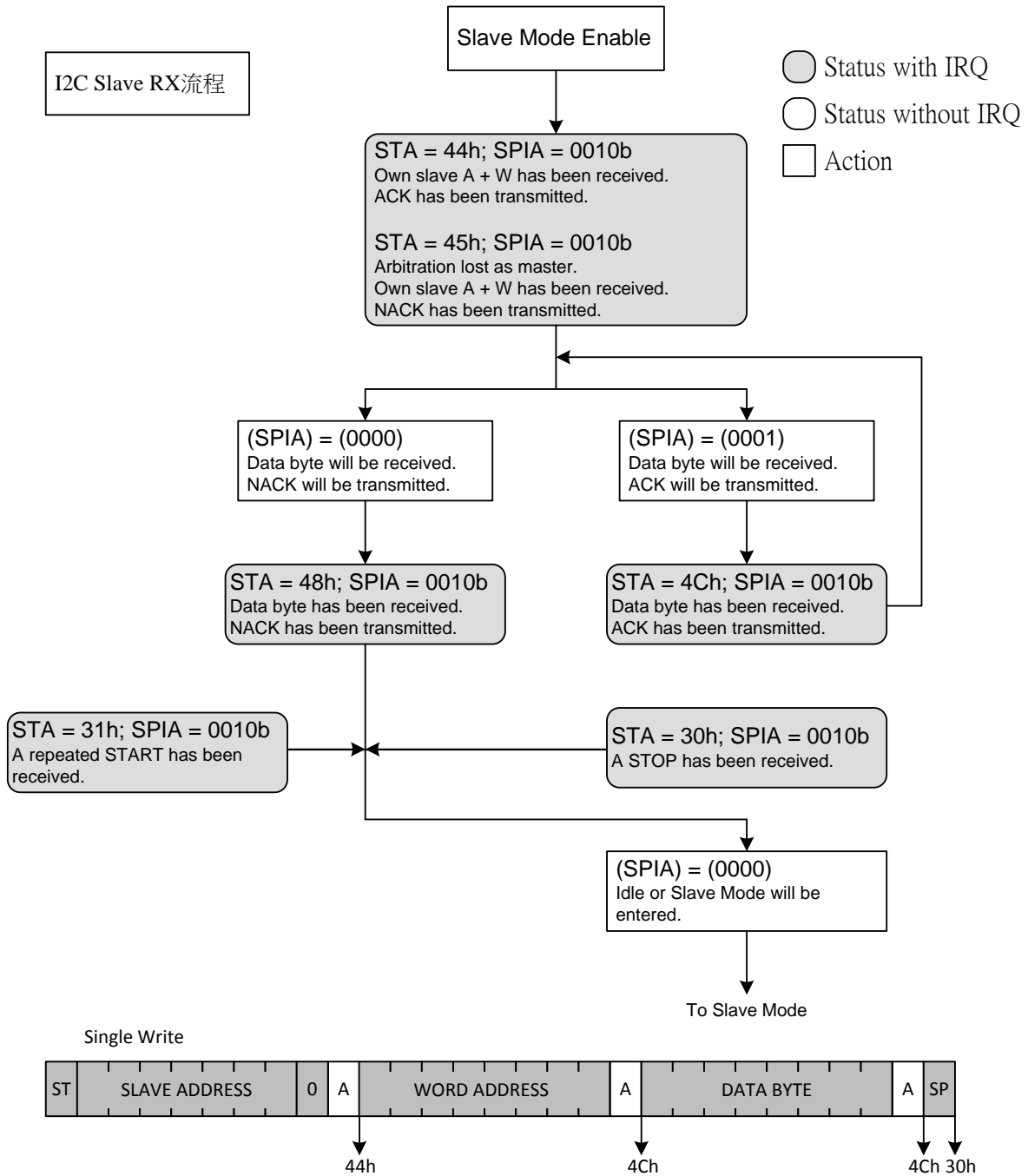


图 27-12 I²C Slave RX 状态之流程图

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

1. 在完成 I²C Slave 初始化之后，首先 I²C Master 端先送出 SLAVE ADDRESS 数据给 I²C Slave 端，如果 I²C Slave 端有正确回 ACK，此时 I²C Slave 端会进入 0x44 的状态，这时候如果使用示波器观察 SCL 与 SDA 脚位状态，可以看出 I²C Master 已经送出 SLAVE ADDRESS 数据给 I²C Slave，并且 I²C Slave 端也已经做了第一次的 ACK 信号回复。
2. 当 I²C Slave 端进入 0x44 状态，此时代表 I²C Slave 已经做了第一次的 ACK 回复，这时候 I²C Master 端如果送出 WORD ADDRESS 的数据给 I²C Slave 端，当 I²C Slave 端使用指令 DrvI2C_Ctrl(0,0,0,1)设定 (SPIA)=(0001)，当执行完(SPIA)=(0001)的控制状态，代表 I²C slave 端已经收到 WORD ADDRESS 数据并且做 ACK 的回复，这时候如果使用示波器观察 SCL 与 SDA 脚位状态，可以看出 I²C Slave 已经做了第二次的 ACK 信号回复，I²C Slave 端进入 0x4C 状态。
3. I²C Slave 端进入 0x4C 状态，此时 I²C Slave 端使用指令 DrvI2C_ReadData 收下 WORD ADDRESS 数据，这时候 I²C Master 端会再送出 DATA BYTE 的资料给 I²C Slave 端，当 I²C Slave 端使用指令 DrvI2C_Ctrl(0,0,0,1)设定 (SPIA)=(0001)，当执行完(SPIA)=(0001)的控制状态，代表 I²C Slave 已经收到 DATA BYTE 数据并且做 ACK 的回复，这时候如果使用示波器观察 SCL 与 SDA 脚位状态，可以看出 I²C Slave 端已经做了第三次的 ACK 信号回复，I²C Slave 端还是进入 0x4C 状态。
4. I²C Slave 端进入 0x4C 状态，此时 I²C Slave 端使用指令 DrvI2C_ReadData 收下 DATA BYTE 数据，当 I²C Slave 端使用指令 DrvI2C_Ctrl(0,0,0,1)设定 (SPIA)=(0001)之后，I²C Master 端会送出 STOP 信号给 I²C Slave 端，这时候如果使用示波器观察 SCL 与 SDA 脚位状态，可以看出 I²C Master 端送出 STOP 信号给 I²C Slave 端，I²C Slave 端进入 0x30 状态。
5. I²C Slave 端进入 0x30 状态，代表 I²C Slave 端已经收到 I²C Master 端所送出 STOP 信号，此时使用指令 DrvI2C_Ctrl(0,0,0,0)设定 (SPIA)=(0000)，让 I²C Slave 端重新进入初始状态，等待下一次的 I²C Master 端信号发送信号。

I²C Slave TX&RX 流程说明：

可以参考以下文字说明，清楚了解到 I²C Slave TX&RX 与 I²C Single Read 的操作流程。

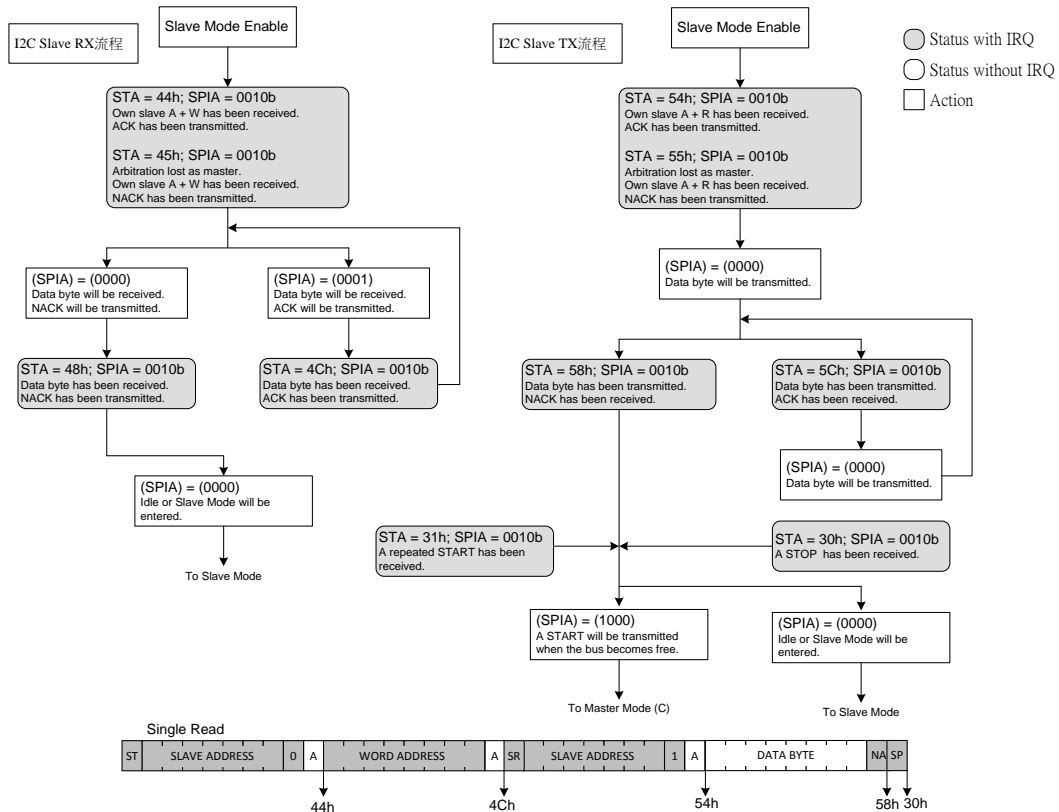


图 27-13 I²C Slave TX&RX 状态之流程图

1. 在完成 I²C Slave 初始化之后，首先 I²C Master 端先送出 SLAVE ADDRESS 数据给 I²C Slave 端，如果 I²C Slave 端有正确回 ACK，此时 I²C Slave 端会进入 0x44 的状态，这时候如果使用示波器观察 SCL 与 SDA 脚位状态，可以看出 I²C Master 已经送出 SLAVE ADDRESS 数据给 I²C Slave，并且 I²C Slave 端也已经做了第一次的 ACK 信号回复。
2. 当 I²C Slave 端进入 0x44 状态，此时代表 I²C Slave 已经做了第一次的 ACK 回复，这时候 I²C Master 端如果送出 WORD ADDRESS 的数据给 I²C Slave 端，当 I²C Slave 端使用指令 DrvI2C_Ctrl(0,0,0,1)设定 (SPIA)=(0001)，当执行完(SPIA)=(0001)的控制状态，代表 I²C slave 端已经收到 WORD ADDRESS 数据并且做 ACK 的回复，这时候如果使用示波器观察 SCL 与 SDA 脚位状态，可以看出 I²C Slave 已经做了第二次的 ACK 信号回复，I²C Slave 端进入 0x4C 状态。
3. I²C Slave 端进入 0x4C 状态，此时 I²C Slave 端使用指令 DrvI2C_ReadData 收下 WORD ADDRESS 数据，这时候 I²C Master 端会送出 SLAVE ADDRESS+1 数据给 I²C Slave 端，当 I²C Slave 端使用指令 DrvI2C_Ctrl(0,0,0,1)设定(SPIA)=(0001)，当执行完(SPIA)=(0001)的控制状态，代表 I²C Slave 已经收到 SLAVE ADDRESS+1 数据并且做 ACK 的回复，这时候如果使用示波器观察 SCL 与 SDA 脚位状态，可以看出 I²C Slave 端已经做了第三次的 ACK 信号回复，I²C Slave 端进入 0x54 状态。
4. I²C Slave 端进入 0x54 状态，此时 I²C Slave 端使用指令 DrvI2C_WriteData 填入想要回传给 I²C Master 端

的 DATA BYTE 数据，当 I²C Slave 端使用指令 DrvI2C_Ctrl(0,0,0,0)设定(SPIA)=(0000)之后，这时候如果使用示波器观察 SCL 与 SDA 脚位状态，可以看出 I²C Master 端已经收到 I²C Slave 端送出的 DATA BYTE 信号，并且 I²C Master 已经送出 NACK 信号给 I²C Slave 端，I²C Slave 端进入 0x58 状态。

5. I²C Slave 端进入 0x58 状态，代表 I²C Slave 端已经收到 I²C Master 端所送出 NACK 信号，此时 I²C Slave 端使用指令 DrvI2C_WriteData 把最高位 MSB 设定为 High 并且使用指令 DrvI2C_Ctrl(0,0,0,0)设定 (SPIA)=(0000)，在执行完(SPIA)=(0000)之后，这时候如果使用示波器观察 SCL 与 SDA 脚位状态，可以看出 I²C Master 端送出 STOP 信号，I²C Slave 端进入 0x30 状态。
6. I²C Slave 端进入 0x30 状态，代表 I²C Slave 端已经收到 I²C Master 端所送出 STOP 信号，此时使用指令 DrvI2C_Ctrl(0,0,0,0)设定(SPIA)=(0000)，让 I²C Slave 端重新进入初始状态，等待下一次的 I²C Master 端信号发送信号。

27.5. I²C General Call Mode

HY16F3910 的 I²C Slave 支持 I²C 特殊工作模式 General Call Function。

当 I²C Slave 的 General Call Function 模式启动之后，此时 I²C Master 可以用广播模式来对 I²C Slave Controller 做控制，I²C Slave 工作在广播模式的时候 Slave Address ID 为 00h，I²C Slave 的 General Call Function 主要区分 General Call 和 General Call Reset 两种。

General Call :

当本机的 General Call 被呼叫时，I²C Slave Controller 会在多等待接收一笔数据后才会发出中断信号，而不是一般的收到 Slave ID 吻合就会立刻发出中断信号，而在响应位的处理上，当 I²C Slave Controller 的从机模式开启时接收 General Call ID 为 00h 时一律自动发出 "ACK"信号至主机，并继续自动接收下一笔数据后一律发出 "NACK"信号与中断信号至本机处理器。下图为 General Call 控制指令，当 SEn 与 GCRst 功能同时使能(Enable) 时候即可使用 General Call 来对 I²C Slave 装置做控制。控制流程为当 I²C Slave 端收到 I²C Master 所送出的第一笔数据 General Call "00h"回复 ACK 与控制命令回复 NACK 的时候会发出第一笔中断信号，读取 STA 为 4Ah 或 4Eh 代表 I²C Slave 已经工作在 General Call 模式，且第一笔数据已存于 RX 接收数据寄存器内，I²C Slave 端可以读取 RX 接收数据寄存器并且判断为何种命令并且做出相对应动作。

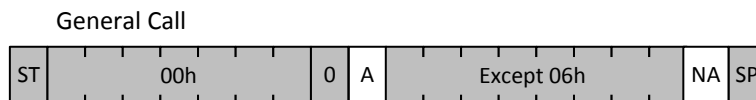


图 27-14 I²C General Call

General Call Reset :

I²C Slave Controller 亦支持 General Call Reset 功能，当 SEn 与 GCRst 功能同时被开启时，如果 I²C Controller 接收 General call ID 00h 并且第一笔资料为"06h"即为 General Call Reset 条件成立，此时原本会发送至本机处理器的中断信号(Interrupt)将被重置信号(Reset)取代，提供外部主机可以经由 I²C Bus 重置本机芯片之功能。



图 27-15 I²C General Call Reset

27.6. 10 Bit Addressing Mode

27.6.1. I²C 10 Bit Addressing Mode 说明

10 位寻址(10-bit addressing)模式为 I²C 原有 7 位寻址的扩充功能，10 位与 7 位寻址模式可以并存于现有之 I²C 架构上。10 位寻址模式是在 START 后的最前面两个字节传送从机地址，I²C Bus 标准亦针对从机地址的格式做了定义，如下图所示，第一个字节必须为 11110xxb，也就是第一个字节必定为“写入”指令，其 Bit2~1 为从机地址的 Bit9~8，第二个字节则为从机地址的 Bit7~0，第三个字节开始才是数据的部分。所以当主机传送第一个字节时，可能会同时有数个从机发出响应。当主机所传送的前两个字节均有收到响应，才代表有从机准备与其通信，并由第三个字节开始传送数据。

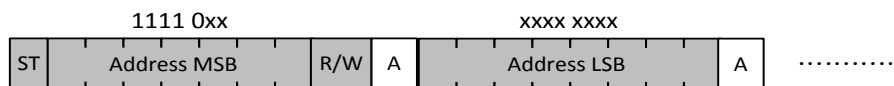


图 27-16 I²C 10 Bit Addressing Mode

Master Transmitter :

主机为传送数据的运作下，主机在使用 I²C 控制器与 7 位寻址模式并无差异，只需要在前面两个字节传送从机地址与写入指令，便可以开始传送数据，但是将 I²C 控制器使用于(Slave)从机模式时需要对寄存器做部份设定。首先必须设定 SID0 为 Address MSB，SID1 为 Address LSB，如下图所示。并且将 SID0 VD0 位设为 High 使能地址比较电路，另外再将 ACT 寄存器内的 SEn 与 10bEn 位设为 High，开启从机模式与 10 位寻址模式便能使 I²C 控制器操作于 10 位从机模式。

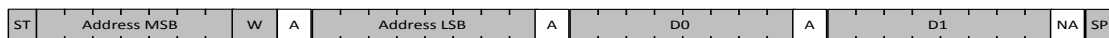


图 27-17 Master Transmitter

Master Receiver :

主机为数据读取的运作下，主机需先以传送“写入”与从机地址的方式使能相对应的从机，再透过 Repeat Start 切换至“读取”指令，当然在主机送出 Repeat Start 前依然可以先写入部份数据再切换致读取模式读取所需数据。而从机在 START 后的从机地址符合后会被使能于写入模式，稍后如果接收 Repeat Start 后只需要比对第一个字节的地址符合即代表主机依然是与本从机通信，也就是说，下图内 START 后的 Address MSB 与 Repeat Start 后的 Address MSB 必须相同，才能使从机进入读取模式，如果不同，从机将会退出此次的通信，而 Repeat Start 后的 Address MSB 将会有其它 7 位从机认定为另一次通信的开始进行地址比较与响应。

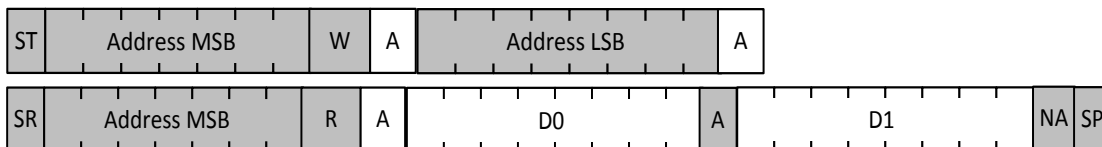


图 27-18 Master Receiver

Slave ID Mask :

从机地址(Slave ID, SID)提供本机运作于从机模式使用，透过 SID 寄存器的设定，本机可以任意设定 Slave ID 并透过本机处理器，将芯片应用于各式各样以 I²C 为传输界面的应用上。而从机地址屏蔽(Slave ID Mask, SIDM)可以更进一步的增加 Slave 模式的应用范例，SID Mask 个别的位被设定为 High 之后，Slave ID 与 I²C Bus 上的 Slave Address 其相对应位将一律是为“吻合”，因此 SID Mask 将可以使 SID 由单一地址扩充至区段式的范围比较。

27.6.2. 10 Bit Addressing 数据写入流程说明

10 位寻址模式数据写入流程：

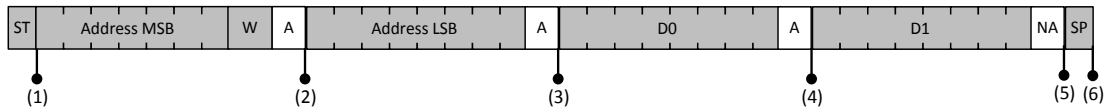


图 27-19 10 位寻址模式数据写入流程

- 10 Bit Address Slave 数据被写入流程(可参照 I²C Slave Receiver Mode 流程图)。
 - A. 于初始化阶段设立 SEn 与 10bEn 使能 10 Bit Addressing Slave 模式，并等待中断信号。(3)
 - B. 读取 STA 为 44h 代表 Bus 上之 Slave ID 为本机 ID，且为被写入状况。
 - C. 设定 A/NA 寄存器为 High，于此笔数据传输完成后传送 ACK 至 Master，代表本机将继续接收下一笔数据，清除中断标志位 IRQFlag 触发被写入程序，并等待下一个中断信号。(4)
 - D. 读取 STA 为 4Ch 确认 ACK 已被传送，如果 Slave 尚可接收数据写入，则重复上一步骤，若无则进行下一步骤。
(上述两步骤只在写入单一笔数据的情况可以忽略)
 - E. 设定 A/NA 寄存器为 Low，于此笔数据传输完成后传送 NACK 至 Master，代表本机将不在接收下一笔数据，清除中断标志位 IRQFlag 触发被写入程序，并等待下一中断信号。(5)
 - F. 读取 STA 为 48h 确认 NACK 已被传送，并准备结束此次被写入程序。
 - G. 清除中断标志位 IRQFlag 等待主机发出之 STOP 信号，并等待中断信号。(6)
 - H. 读取 STA 为 30h 代表主机已经结束写入程序。
 - I. 清除中断标志位 IRQFlag 已进入下一程序，可以设立 START 寄存器以进入主机模式试图取得 Bus 控制权，或是只清中断标志位 IRQFlag 持续维持从机模式。

27.6.3. 10 Bit Addressing 数据读出流程说明

10 位寻址模式数据读出流程(10 Bit Addressing Master) :

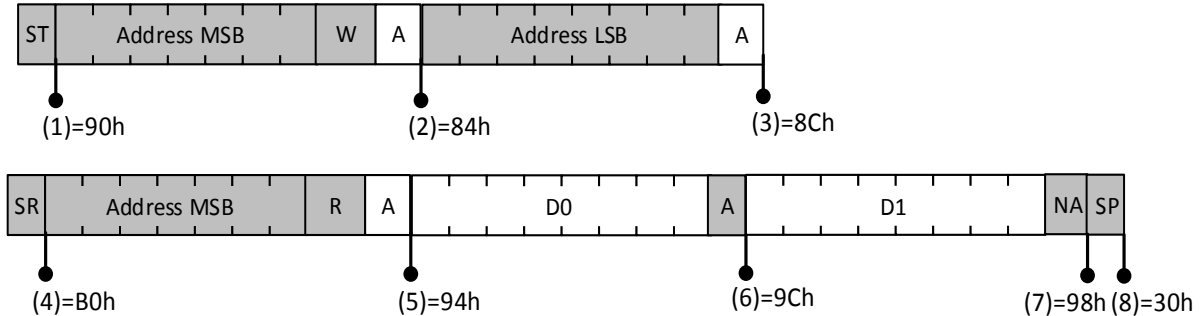


图 27-20 10 位寻址模式数据读出流程

- 10 Bit Addressing Master 数据读取流程 (可先参照 I²C Master Transmitter 流程 再参照 I²C Master Receiver Mode 流程图)
 - A. 设立 START 寄存器发出 Start Bit，并等待中断信号。(1)
 - B. 读取 STA 为 90h 代表 Start 以成功发出并取得 Bus 控制权。
 - C. 将目标 Slave 之地址 Address MSB 填写于 TxAD 寄存器内。
 - D. 清除中断标志位 IRQFlag 触发位只传送程序，并等待中断信号。(2)
 - E. 读取 STA 为 80h 代表 Bus 上无此 Slave 或 Slave 无法响应(可能过于忙碌或已经当机)。如读取 STA 为 84h 代表 Slave 已发出响应。在 10 Bit Addressing 模式下可能会有数个 Slave 同时对 Address MSB 做出响应。
 - F. 将目标 Slave 之地址 Address MSB 填写于 TxAD 寄存器内。
 - G. 清除中断标志位 IRQFlag 触发位只传送程序，并等待中断信号。(3)
 - H. 读取 STA 为 88h 代表 Bus 上无此 Slave 或 Slave 无法响应(可能过于忙碌或已经当机)。如读取 STA 为 8CH 代表 Slave 已发出响应，并等待数据写入。
(如果主机有数据或命令要传送到从机，可以在此步骤以后执行送出数据的流程)
 - I. 设立 START 寄存器，并清除中断标志位 IRQFlag 触发 Repeat Start Bit 传送程序，并等待中断信号，以进行对 10 Bit Addressing Slave 的数据读取。(4)
 - J. 读取 STA 为 B0h 代表 Repeat Start 以成功发出并取得 Bus 控制权。
 - K. 将目标 Slave 之地址 Address MSB 填写于 TxAD 寄存器内。
 - L. 清除中断标志位 IRQFlag 触发地址传送程序，并等待中断信号。(5)
 - M. 读取 STA 为 91h 代表 Bus 上无此 Slave 或 Slave 无法响应(可能过于忙碌或已经当机)。如读取 STA 为 94h 代表 Slave 已发出响应，并等待数据读取。
 - N. 设定 A/NA 寄存器为 High 于此笔数据传输完成后传送 ACK 至 Slave 代表后续尚有数据需传输。清除中断标志位 IRQFlag 触发数据读取程序，并等待下一中断信号。(6)
 - O. 读取 RxAD 内由 Slave 回传之数据，并读取 STA 为 9Ch 确认 ACK 已被传送，如果再次由 Slave 读取数据后尚有数据需要读取则重复上一步骤，若无则进行下一步骤。(上述两步骤只在读取单一笔数据的情况可以忽略)
 - P. 设定 A/NA 寄存器为 Low 于此笔数据传输完成后传送 NACK 至 Slave 代表传输流程即将结束，清除中

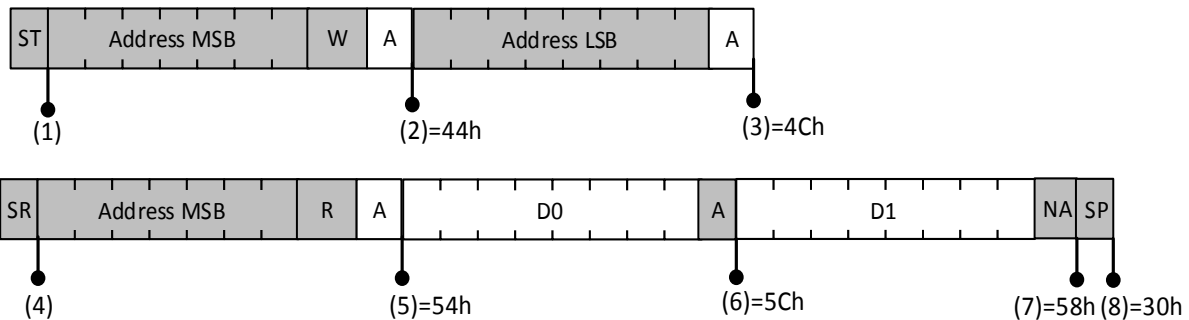
断标志位 IRQFlag 触发数据读取程序，并等待下一中断信号。(7)

Q. 读取 RxAD 内由 Slave 回传之数据; 读取 STA 为 98h 代表 NACK 以传出。

R. 设立 STOP 寄存器，并清除中断标志位 IRQFlag 触发 STOP Bit 传送程序，已结束本次数据传输。

27.6.4. 10 Bit Addressing 数据被读出流程说明

10 位寻址模式数据被读出流程(10 Bit Addressing Slave) :



- 10 Bit Addressing Slave 数据被读取流程(请先参照 I²C Slave Receiver 流程再参照 I²C Slave Transmitter Mode 流程)

- A. 于初始化阶段设立 SEn 与 10bEn 使能 10 Bit Addressing Slave 模式并等待中断信号。(3)
- B. 读取 STA 为 44h 代表 Bus 上之 Slave ID 为本机 ID，且为被写入状况。
- C. 设定 A/NA 寄存器为 High，于此笔数据传输完成后传送 Ack 至 Master，代表本机将继续接收下一笔数据，清除中断标志位 IRQFlag 触发被写入程序，并等待下一中断信号。
- D. 读取 STA 若为 4Ch 代表 Master 尚处于对 Slave 写入数据或命令之行程。此时之中断信号为 10 位寻址模式数据写入流程的第(4)号中断，并请以改该流程继续执行。
- E. 如果 Master 已经发出 Repeat Start Bit，此时之中断为本 case 之第(4)号中断，读取 STA 为 70h，因为此时 Master 持续再传送新的 Slave ID，STA 为 70h 只存在 Bit7 的 SCL 负缘之前，负缘之后所读之 STA 可能为 50h 或 54h，清除中断标志位 IRQFlag 结束先前的写入程序，并等待下一中断信号。(5)
- F. 读取 STA 为 54h 代表 Bus 上之 Slave ID 为本机 ID，且为被读取状况。请注意，如果本机来不及处理上一步之中断信号，将有可能被本步骤之中断信号覆盖。
- G. 将欲被读取之数据存入 TxAD 内，清除中断标志位 IRQFlag 触发被读取程序，并等待中断信号。(6)(7)
- H. 读取 STA 如果是 58h 代表主机即将结束读取程序，如果是 5Ch 则表示主机要读取数据，Slave 需准备相关数据并重复上一步骤。
- I. 清除中断标志位 IRQFlag 等待主机发出之 Stop Bit 信号，并等待中断信号。(8)
- J. 读取 STA 为 30h 代表主机已经结束读取程序。
- K. 清除中断标志位 IRQFlag 已进入下一程序，可以设立 START 寄存器已进入主机模式，试图取得 Bus 控制权，或是只清中断标志位 IRQFlag 持续维持从机模式。

27.7.3 Byte Data Mode

27.7.1. I²C 3 Byte Date Mode 说明

三字节数据传送模式(3 Byte Data Mode)提供连续传送三个字节的的数据,本机处理器可以将三个字节的的数据同时填入 TX0, TX1 与 TX2 三个寄存器内, 并启动三字节传送模式, I²C Controller 将会自动传送三个字节的的数据后才会发出中断通知本机处理器进行下一步的控制。此一功能被设计来减少 I²C Controller 的中断发生频率, 以及减少本机处理器需服务中断呼叫的次数, 用以提升整体芯片的运作效能。

当 3Byte Data Mode 已经启动后, 本机处理器已经将数据填入传送寄存器后, 在下达数据传输命令的同时将 3BEn 位设立, 使 I²C Controller 开始执行三个字节的的数据传输。使用此模式需要特别注意的是, 只有本机处与从机模式且需要传送数据时才能开启 3BEn 位, 当数据传送行程结束必须关闭 3BEn 位, 以免造成 I²C Controller 动作异常, 另外要注意的是 3Byte Data Mode 在每一次的传输行程只能使用一次, 也就是说当每一次的 Start Bit 后本机进行从机数据传输模式时, 只能利用三字节模式自动传送三笔数据, 之后的数据传输只能以一般的单笔数据传送模式进行。

27.7.2. I²C 3 Byte Date 连续读取流程说明

3 Byte Data 连续读取模式 :

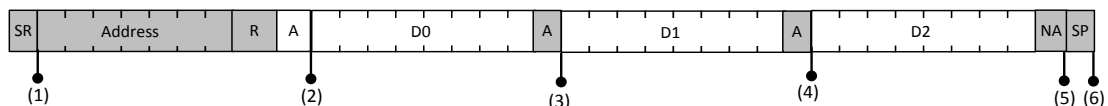


图 27-21 3Byte Data 连续读取模式

3 Byte Master 数据读取流程 (请参照 I²C Master Receiver 流程图)

Master 于此模式是以一般的读取流程进行, 相关中断信号为(1~5)号中断。

27.7.3. I²C 3 Byte Date 数据被读取流程说明

3 Byte Slave 数据被读取流程 (请参照 I²C Slave Transmitter 流程图)

- 于初始化阶段设立 SEn 使能 Slave 模式, 并等待中断信号。(2)
- 读取 STA 为 54h 代表 Bus 上之 Slave ID 为本机 ID, 且为被读取状况。
- 将欲被读取之数据存入 TX0, TX1, TX2 内, 设立 3BEn 并清除中断标志位 IRQFlag 触发被读取程序, 并等待中断信号。(5)
- 读取 STA 为 58h 代表主机即将结束读取程序。
- 清除 3BEn 与中断标志位 IRQFlag 等待主机发出之 Stop Big 信号, 并等待中断信号。(4)
- 读取 STA 为 30h 代表主机已结束读取程序。
- 清除中断标志位 IRQFlag 以进入下一程序, 可以设立 START 寄存器已进入主机模式试图取得 Bus 控制权, 或是只清中断标志位 IRQFlag 持续维持从机模式。

28. 硬件万年历 HW RTC

28.1. 整体总说明

实时时钟(RTC)控制器提供使用者实时时钟和日历。RTC 的时钟源来自于连接 I/O 口的外部 32.768kHz 振荡器或内部的 32kHz LPO 振荡器。这个 RTC 控制器以二进制编码的十进制(BCD)格式来呈现时/分/秒的时间信息，以及年/月/日/星期的日期信息。在这控制器中，有一个可编程警报中断程序和一个定期可编程唤醒中断程序，使系统得以自动唤醒处理低电力模式。另有一个 6 位的数字定时振荡器偏差补偿机制。

功能：时间信息(时/分/秒)以及日历信息(年/月/日/星期)储存在寄存器中。

警报寄存器(年/月/日期/时/分/秒)。

所有的时间和日历信息皆以 BCD 格式呈现。

闰年自动补偿(年度：2012~2099)。

星期计数器。

6 位数字定时振荡器偏差补偿。

支持定期将 CPU 从闲置模式唤醒。

支持 8 个定期唤醒期间选项：1/128, 1/64, 1/32, 1/16, 1/8, 1/4, 1/2, and 1。

支持 12/24 小时制的时间模式。

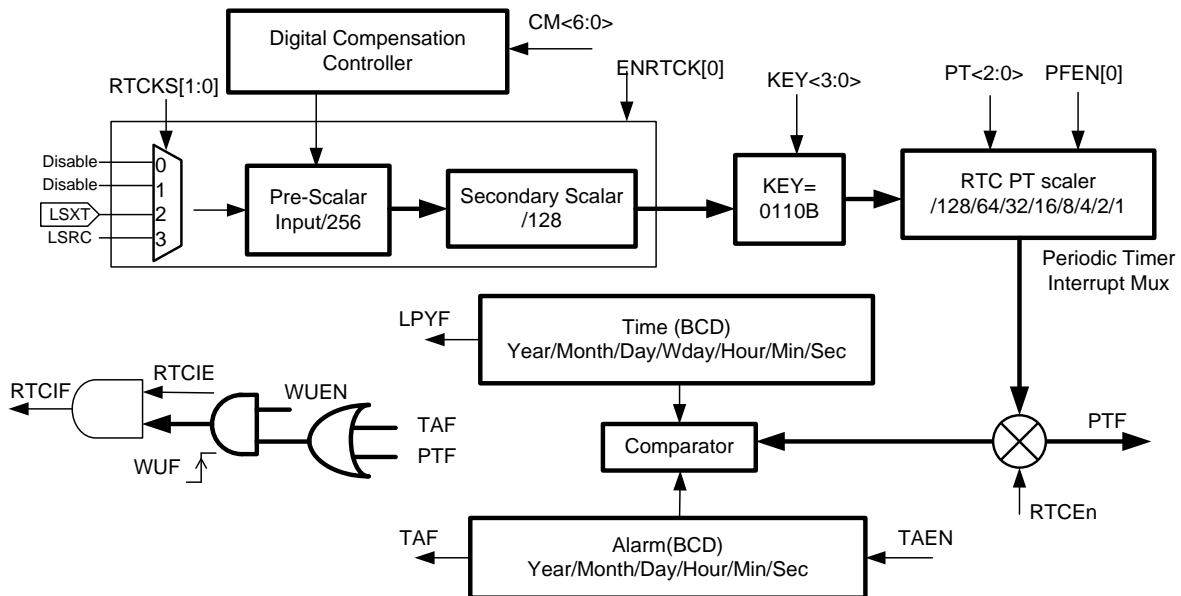


图 28-1 RTC 架构图

对 RTC 寄存器的存取:

因为 RTC 时钟和系统时钟的频率不同, 在使用者写入一个新的数据到寄存器后, 经过两个 RTC 时钟脉冲后, 寄存器会被更新。RTC 数据应该要经常更新。有一个针对写入 RTC 寄存器的保护键。要对 RTC 寄存器作写入的动作时, 这个 KEY 键须被设定为<0110>, 其它的 KEY 键值是无法对 RTC 寄存器作写入的动作。但要注意, 因为 RTC 不会检查对寄存器写入的数据格式, 使用者必须很小心这个写入动作。

启动 RTC:

要写入一个值到 RTC 寄存器, 就必须先将<0110>写入到 KEY 0x41A00[23:20]。要启动 RTC, 使用者首先要检查 LSXT 或 LSRC 是否可使用。然后将 RTCEn 0x41A00[0]设定为<1>。

频率补偿:

RTC 允许对时钟输入的数字补偿。RTC 中央频率为 32768Hz。任何的不完美可导致频率偏差。可用数字补偿来降低频率偏差。补偿方式为每一步骤执行+/-2ppm, 可允许最大+126ppm 及最小-126ppm 的频率变动。最大的输入频率为 32772Hz, 最小的输入频率为 32763Hz。

使用者可以输入最大到 16MHz 的基准频率, 在制造期间来测量 RTC 时钟。测量值被计算以求得补偿值。然后, 这个补偿值被储存在闪存中。一旦系统被开机后, 这个补偿值就会被载入到 CM 0x41A04[22:16]。

时间信息:

时间储存在 0x41A08 与 0x41A0C 这些寄存器中。它们使用 BCD 格式。使用者可将时间设定为 24 小时制或 12(AM/PM)制。时间的默认值为 00:00:00 (时/分/秒), 且为 24 小制。

日历信息:

日历信息储存在 0x41A10 与 0x41A14 这些寄存器中。它们使用 BCD 格式。闰年的算法是由硬件来完成。有效年份期间为 2012~2099。当 LPYF 0x41A00[19]为<1>, 就表示为闰年。年份是使用两个数字来表示, 代表 20xx 年。系统重设后的预设日期为 12/1/1 星期天(2012 年 1 月 2 日)。最大年度为 99, 且在 99/12/31 之后年度会变成 00/1/1, 但此时闰年补偿就会无效。

星期计数器:

RTC 控制器提供星期的信息。这个 WDA 0x41A14[2:0]值被定义为从 0 到 6, 分别代表星期天到星期六。

TAF 闹钟中断:

当 0x41A08/0x41A0C/0x41A10/0x41A14 这些寄存器与 0x41A18/0x41A1C 这些寄存器内的时间设定相符合, 且 TAEEn 0x41A00[03]为<1>时, TAF 0x41A00[16]中断标志位会设定为<1>给 MCU。

PTF 定期定时器中断:

这个定期定时器中断有 8 个定期选项: 1/128、1/64、1/32、1/16、1/8、1/4、1/2 和 1 秒。将 PTF 0x41A00[18]设定为<1>以启动这个定期定时器中断。这些定期的选项系由 PT 0x41A04[2:0]所控制。

WUF 系统唤醒中断:

当 MCU 进入闲置模式后, 它可经由系统唤醒中断程序被唤醒。有两个来源可唤醒 MCU: 定期定时器中断和闹钟中断。将 WUF 0x41A00[17]设定为<1>以启动这个中断程序。

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



28.2. RTC 寄存器地址

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 + 0x10 (0x41A10)	RTYEM		RTYE		RTMOM		RTMO	
RTC Base Address + 0x14 (0x41A14)	RTDAM		RTDA		RTWDM		RTWDA	
RTC Base Address + 0x18 (0x41A18)	-		RCHR		RCMI		RCSE	
RTC Base Address + 0x1C (0x41A1C)	-		RCYE		RCMO		RCDA	

-保留

28.3. RTC 寄存器功能

28.3.1. RTC 寄存器 0

RTC Base Address + 0x00 (0x41A00)									
RTCCR0 (RTC Control Register 0)									
Symbol	[31:28]	[27:24]	[23:20]			[19]	[18]	[17]	[16]
名称	-	MASK	KEY			LPYF	PTF	WUF	TAF
RW	-	R0W-0	RW-0			R-0	RW0-0	R-0	RW0-0
Bit	[15:14]	[13:8]	[07:06]	[05]	[04]	[03]	[02]	[01]	[00]
名称	-	MASK	-	PTEn	WUEn	TAEn	HRF	-	RTCEn
RW	-	R0W-0	-	RW-0					

位	名称	描述	
Bit[23:20]	KEY	RTC 寄存器的密钥，上锁后可保护寄存器，防止被写入	
		0110	写入密钥 6，解除寄存器保护，只有解锁后才能对寄存器写入操作
		Others	上锁，保护寄存器，不能进行写入操作
Bit[19]	LPYF	闰年标志位	
		0	当前年份不是闰年
		1	当前年份是闰年
Bit[18]	PTF	定时器唤醒中断标志位	
		0	正常
		1	定时器唤醒已触发
Bit[17]	WUF	唤醒中断标志位	
		0	正常
		1	唤醒中断已触发
Bit[16]	TAF	闹钟状态标志位	

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述	
		0	正常
		1	闹钟中断触发
Bit[05]	PTEn	RTC 定时器计时功能开启控制	
		0	关闭
		1	开启
Bit[04]	WUEn	RTC 唤醒功能开启控制	
		0	关闭
		1	开启
Bit[03]	TAEn	RTC 闹钟功能开启控制	
		0	关闭
		1	开启
Bit[02]	HRF	RTC 小时格式设置 (24/12)	
		0	24 小时制
		1	12 小时制(PM/AM)
Bit[00]	RTCEn	RTC 功能开启控制	
		0	关闭 RTC 功能
		1	开启 RTC 功能

注意事项：

RTC Clcok Source Selection“CKS”具有防呆保护，如果 CKS 选择 LSXT 但是 LSXT 没有 Enable 的情况下，电路会自动切换至 LSRC 作为 Clcok Source。

当 RTC 设定工作于 24 小时制的时候，RTC 的小时(Hour)单位计数范围是 0~23 循环计数，当 RTC 设定工作于 12 小时制的时候，RTC 的小时(Hour)单位计数范围是 0~11 循环计数

当 HRF 控制位设定为<1>的时候，也就是工作在 12 小时制，此时如果要对 RTC 时间做写入动作，在小时(Hour)的单位，如果超过数字 12 以上会造成 RTC 写入无效动作。

RTC 寄存器数据写入时需注意，如果在 HRF 控制位设定为<0>的时候，也就是工作在 24 小时制，此时写入时间如果是大于 12 小时，资料可以正常写入 RTC 寄存器内。而这时如果再把 HRF 控制位设定为<1>的时候，则会造成 RTC 寄存器的小时单位不断的往上计数上去，此时，即使是设定工作在 12 小时制，小时单位计数也不会是 0~11 的循环计数，会有异常状况发生。

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



28.3.2. RTC 寄存器 1

RTC Base Address + 0x04 (0x41A04)						
RTCCR1 (RTC Control Register 1)						
Symbol	[31:24]		[23]		[22:16]	
名称	MASK		Rsv		CM	
RW	R0W-0		R-0		RW-0	
Bit	[15:08]	[07]	[06:05]	[04]	[03]	[02:00]
名称	MASK	CHK	12HM	-	Rsv	PT
RW	R0W-0	R-0	RW-0	R-0	R-0	RW-0

位	名称	描述
Bit[22:16]	CM	RTC 频率频率补偿值设置
		CM[6:0] 补偿值
		0111111 +126 PPM 的振荡器频率补偿(最大值)
		0111110 +124 PPM 的振荡器频率补偿
		... 递增步长: +2 PPM 的振荡器频率补偿
		0000001 +2 PPM 的振荡器频率补偿
		0000000 0 PPM 的振荡器频率补偿
		1000000 0 PPM 的振荡器频率补偿
		1000001 -2 PPM 的振荡器频率补偿
		... 递减步长: -2 PPM 的振荡器频率补偿
		1111110 -124 PPM 的振荡器频率补偿
		1111111 -126 PPM 的振荡器频率补偿(最小值)
Bit[07]	CHK	Check Format
		0 Normal
		1 Fail
Bit[6:5]	12HM	The Hour Format
		12HM[1:0] AM PM
		00 00, 01 ~ 11 00, 01 ~ 11
		01 12, 01 ~ 11 00, 01 ~ 11
		10 00, 01 ~ 11 12, 01 ~ 11
11 12, 01 ~ 11 12, 01 ~ 11		
Bit[2:0]	PT	定时器定时唤醒时间设置
		PT[2:0] 唤醒时间
		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

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



28.3.3. RTC 寄存器 2

RTC Base Address + 0x08 (0x41A08)					
Symbol	RTCHRC (RTC Hour Control Register For calendar)				
Bit	[31:15]				
名称	RSV				
RW	R-0				
Bit	[14:08]	[07]	[06]	[05:04]	[03:00]
名称	MASK	-	PM	10HR	1HR
RW	ROW-0	-		RW-0	

位	名称	描述
Bit[6]	PM	万年历小时格式 AM/PM
		0 AM 或 24 小时制
		1 PM
Bit[5:4]	10HR	小时十位数的值
		10HR[1:0] 值(BCD 码格式)
		00 0
		01 1
		10 2; HRF=0 时无效
11 无效		
Bit[3:0]	1HR	小时个位数的值(BCD 码格式)
		1HR[3:0] 值(BCD 码格式)
		0000 0
		0001 1
		0010 2
		0011 3
		0100 4
		0101 5
		0110 6
		0111 7
		1000 8
		1001 9
		1010 1111 无效

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



28.3.4. RTC 寄存器 3

RTC Base Address + 0x0C (0x41A0C)				
Symbol	RTCSCMC (RTC seconds and min Control Register For calendar)			
Bit	[31:24]	[23]	[22:20]	[19:16]
名称	MASK	-	10MIN	1MIN
RW	ROW-0	-	RW-0	RW-0
Bit	[15:08]	[07]	[06:04]	[03:00]
名称	MASK	-	10SEC	1SEC
RW	ROW-0	-	RW-0	RW-0

位	名称	描述
Bit[22:20]	10MIN	分钟十位数的值(BCD 码格式)
		10MIN[2:0] 值(BCD 码格式)
		000 0
		001 1
		010 2
		011 3
		100 4
		101 5
		110 6
		111 无效
Bit[19:16]	1MIN	分钟个位数的值(BCD 码格式)
		1MIN[3:0] 值(BCD 码格式)
		0000 0
		0001 1
		0010 2
		0011 3
		0100 4
		0101 5
		0110 6
		0111 7
		1000 8
		1001 9
		1010 1111 无效
Bit[06:04]	10SEC	秒钟十位数的值(BCD 码格式)
		10SEC[2:0] 值(BCD 码格式)
		000 0
		001 1
		010 2
		011 3
		100 4
		101 5
		110 6
		111 无效
Bit[03:00]	1SEC	秒钟个位数的值(BCD 码格式)
		1SEC[3:0] 值(BCD 码格式)
		0000 0

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述	
		0001	1
		0010	2
		0011	3
		0100	4
		0101	5
		0110	6
		0111	7
		1000	8
		1001	9
		1010 1111	无效

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



28.3.5. RTC 寄存器 4

RTC Base Address + 0x10 (0x41A10)				
Symbol	RTCYMC (RTC Year and Month Control Register For Calendar)			
Bit	[31:24]	[23:20]		[19:16]
名称	MASK	10YEAR		1YEAR
RW	R0W-0	RW-1		RW-2
Bit	[15:08]	[07:05]	[04]	[03:00]
名称	MASK	-	10MO	1MO
RW	R0W-0	-	RW-0	RW-1

位	名称	描述																										
Bit[23:20]	10YEAR	年份十位数的值 (BCD 码格式)																										
		<table border="1"> <thead> <tr> <th>10YEAR[3:0]</th> <th>值(BCD 码格式)</th> </tr> </thead> <tbody> <tr><td>0000</td><td>0</td></tr> <tr><td>0001</td><td>1</td></tr> <tr><td>0010</td><td>2</td></tr> <tr><td>0011</td><td>3</td></tr> <tr><td>0100</td><td>4</td></tr> <tr><td>0101</td><td>5</td></tr> <tr><td>0110</td><td>6</td></tr> <tr><td>0111</td><td>7</td></tr> <tr><td>1000</td><td>8</td></tr> <tr><td>1001</td><td>9</td></tr> <tr><td>1010</td><td rowspan="3">无效</td></tr> <tr><td> </td></tr> <tr><td>1111</td></tr> </tbody> </table>	10YEAR[3:0]	值(BCD 码格式)	0000	0	0001	1	0010	2	0011	3	0100	4	0101	5	0110	6	0111	7	1000	8	1001	9	1010	无效		1111
		10YEAR[3:0]	值(BCD 码格式)																									
		0000	0																									
		0001	1																									
		0010	2																									
		0011	3																									
		0100	4																									
		0101	5																									
		0110	6																									
		0111	7																									
		1000	8																									
		1001	9																									
		1010	无效																									
1111																												
Bit[19:16]	1YEAR	年份个位数的值 (BCD 码格式)																										
		<table border="1"> <thead> <tr> <th>1YEAR[3:0]</th> <th>值(BCD 码格式)</th> </tr> </thead> <tbody> <tr><td>0000</td><td>0</td></tr> <tr><td>0001</td><td>1</td></tr> <tr><td>0010</td><td>2</td></tr> <tr><td>0011</td><td>3</td></tr> <tr><td>0100</td><td>4</td></tr> <tr><td>0101</td><td>5</td></tr> <tr><td>0110</td><td>6</td></tr> <tr><td>0111</td><td>7</td></tr> <tr><td>1000</td><td>8</td></tr> <tr><td>1001</td><td>9</td></tr> <tr><td>1010</td><td rowspan="3">无效</td></tr> <tr><td> </td></tr> <tr><td>1111</td></tr> </tbody> </table>	1YEAR[3:0]	值(BCD 码格式)	0000	0	0001	1	0010	2	0011	3	0100	4	0101	5	0110	6	0111	7	1000	8	1001	9	1010	无效		1111
		1YEAR[3:0]	值(BCD 码格式)																									
		0000	0																									
		0001	1																									
		0010	2																									
		0011	3																									
		0100	4																									
		0101	5																									
		0110	6																									
		0111	7																									
		1000	8																									
		1001	9																									
		1010	无效																									
1111																												
Bit[04]	10MO	月份十位数的值(BCD 码格式)																										
		<table border="1"> <tbody> <tr><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td></tr> </tbody> </table>	0	0	1	1																						
0	0																											
1	1																											
Bit[3:0]	1MO	月份个位数的值(BCD 码格式)																										
		<table border="1"> <thead> <tr> <th>1MO[3:0]</th> <th>值(BCD 码格式)</th> </tr> </thead> <tbody> <tr><td>0000</td><td>0</td></tr> <tr><td>0001</td><td>1</td></tr> </tbody> </table>	1MO[3:0]	值(BCD 码格式)	0000	0	0001	1																				
		1MO[3:0]	值(BCD 码格式)																									
0000	0																											
0001	1																											

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述	
		0010	2
		0011	3
		0100	4
		0101	5
		0110	6
		0111	7
		1000	8
		1001	9
		1010 1111	无效

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



28.3.6. RTC 寄存器 5

RTC Base Address + 0x14 (0x41A14)				
Symbol	RTCDWC (RTC Date and week Control Register For calendar)			
Bit	[31:24]	[23:22]	[21:20]	[19:16]
名称	MASK	-	10DAT	1DAT
RW	R0W-0	-	RW-0	RW-1
Bit	[15:08]	[07:03]		[02:00]
名称	MASK	-		WDA
RW	R0W-0	-		RW-0

位	名称	描述																											
Bit[21:20]	10DAT	日期十位数的值 (BCD 码格式)																											
		00 0																											
		01 1																											
		10 2																											
		11 3																											
Bit[19:16]	1DAT	日期个位数的值 (BCD 码格式)																											
		<table border="1"> <thead> <tr> <th>1DAT[3:0]</th> <th>值(BCD 码格式)</th> </tr> </thead> <tbody> <tr><td>0000</td><td>0</td></tr> <tr><td>0001</td><td>1</td></tr> <tr><td>0010</td><td>2</td></tr> <tr><td>0011</td><td>3</td></tr> <tr><td>0100</td><td>4</td></tr> <tr><td>0101</td><td>5</td></tr> <tr><td>0110</td><td>6</td></tr> <tr><td>0111</td><td>7</td></tr> <tr><td>1000</td><td>8</td></tr> <tr><td>1001</td><td>9</td></tr> <tr><td>1010</td><td rowspan="3">无效</td></tr> <tr><td> </td></tr> <tr><td>1111</td></tr> </tbody> </table>	1DAT[3:0]	值(BCD 码格式)	0000	0	0001	1	0010	2	0011	3	0100	4	0101	5	0110	6	0111	7	1000	8	1001	9	1010	无效		1111	
		1DAT[3:0]	值(BCD 码格式)																										
		0000	0																										
		0001	1																										
		0010	2																										
		0011	3																										
		0100	4																										
		0101	5																										
		0110	6																										
		0111	7																										
		1000	8																										
		1001	9																										
		1010	无效																										
1111																													
1010	无效																												
1111																													
Bit[2:0]	WDA	星期的值 (BCD 码格式)																											
		<table border="1"> <thead> <tr> <th>WDA[2:0]</th> <th>值 (BCD 码格式)</th> <th></th> </tr> </thead> <tbody> <tr><td>000</td><td>0</td><td>Sunday 星期日</td></tr> <tr><td>001</td><td>1</td><td>Monday 星期一</td></tr> <tr><td>010</td><td>2</td><td>Tuesday 星期二</td></tr> <tr><td>011</td><td>3</td><td>Wednesday 星期三</td></tr> <tr><td>100</td><td>4</td><td>Thursday 星期四</td></tr> <tr><td>101</td><td>5</td><td>Friday 星期五</td></tr> <tr><td>110</td><td>6</td><td>Saturday 星期六</td></tr> <tr><td>111</td><td>7</td><td>无效</td></tr> </tbody> </table>	WDA[2:0]	值 (BCD 码格式)		000	0	Sunday 星期日	001	1	Monday 星期一	010	2	Tuesday 星期二	011	3	Wednesday 星期三	100	4	Thursday 星期四	101	5	Friday 星期五	110	6	Saturday 星期六	111	7	无效
		WDA[2:0]	值 (BCD 码格式)																										
		000	0	Sunday 星期日																									
		001	1	Monday 星期一																									
		010	2	Tuesday 星期二																									
		011	3	Wednesday 星期三																									
		100	4	Thursday 星期四																									
		101	5	Friday 星期五																									
		110	6	Saturday 星期六																									
111	7	无效																											
000	0	Sunday 星期日																											
001	1	Monday 星期一																											
010	2	Tuesday 星期二																											
011	3	Wednesday 星期三																											
100	4	Thursday 星期四																											
101	5	Friday 星期五																											
110	6	Saturday 星期六																											
111	7	无效																											

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



28.3.7. RTC 寄存器 6

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]	
名称	RSV	-	CPM	10CHR	1CHR	
RW	R-0	-		RW-0		
Bit	[15]	[14:12]	[11:08]	[07]	[06:04]	[03:00]
名称	-	10CMI	1CMI	-	10CSE	1CSE
RW	-	RW-0	-		RW-0	

位	名称	描述																								
Bit[22]	CPM	闹钟显示的格式为 am/pm																								
		0 AM 或者 24 小时制																								
		1 PM (当 HRF=1 时, 该位需要被置 1)																								
Bit[21:20]	10CHR	闹钟模式下小时十位数的值 (BCD 码格式)																								
		<table border="1"> <thead> <tr> <th>10CHR[3:0]</th> <th>值(BCD 码格式)</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>0</td> </tr> <tr> <td>01</td> <td>1</td> </tr> <tr> <td>10</td> <td>2 (HRF=1) / HRF=0 时无效</td> </tr> <tr> <td>11</td> <td>无效</td> </tr> </tbody> </table>	10CHR[3:0]	值(BCD 码格式)	00	0	01	1	10	2 (HRF=1) / HRF=0 时无效	11	无效														
		10CHR[3:0]	值(BCD 码格式)																							
		00	0																							
		01	1																							
10	2 (HRF=1) / HRF=0 时无效																									
11	无效																									
Bit[19:16]	1CHR	闹钟模式下小时个位数的值 (BCD 码格式)																								
		<table border="1"> <thead> <tr> <th>1CHR[3:0]</th> <th>值(BCD 码格式)</th> </tr> </thead> <tbody> <tr> <td>0000</td> <td>0</td> </tr> <tr> <td>0001</td> <td>1</td> </tr> <tr> <td>0010</td> <td>2</td> </tr> <tr> <td>0011</td> <td>3</td> </tr> <tr> <td>0100</td> <td>4</td> </tr> <tr> <td>0101</td> <td>5</td> </tr> <tr> <td>0110</td> <td>6</td> </tr> <tr> <td>0111</td> <td>7</td> </tr> <tr> <td>1000</td> <td>8</td> </tr> <tr> <td>1001</td> <td>9</td> </tr> <tr> <td>1010 1111</td> <td>无效</td> </tr> </tbody> </table>	1CHR[3:0]	值(BCD 码格式)	0000	0	0001	1	0010	2	0011	3	0100	4	0101	5	0110	6	0111	7	1000	8	1001	9	1010 1111	无效
		1CHR[3:0]	值(BCD 码格式)																							
		0000	0																							
		0001	1																							
		0010	2																							
		0011	3																							
		0100	4																							
		0101	5																							
		0110	6																							
		0111	7																							
		1000	8																							
		1001	9																							
1010 1111	无效																									
Bit[14:12]	10CMI	闹钟模式下分钟十位数的值 (BCD 码格式)																								
		<table border="1"> <thead> <tr> <th>10CMI[2:0]</th> <th>值(BCD 码格式)</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>0</td> </tr> <tr> <td>001</td> <td>1</td> </tr> <tr> <td>010</td> <td>2</td> </tr> <tr> <td>011</td> <td>3</td> </tr> <tr> <td>100</td> <td>4</td> </tr> <tr> <td>101</td> <td>5</td> </tr> <tr> <td>110</td> <td>6</td> </tr> <tr> <td>111</td> <td>无效</td> </tr> </tbody> </table>	10CMI[2:0]	值(BCD 码格式)	000	0	001	1	010	2	011	3	100	4	101	5	110	6	111	无效						
		10CMI[2:0]	值(BCD 码格式)																							
		000	0																							
		001	1																							
		010	2																							
		011	3																							
		100	4																							
101	5																									
110	6																									
111	无效																									

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述	
Bit[11:8]	1CMI	闹钟模式下分钟个位数的值 (BCD 码格式)	
		1CMI[3:0]	值(BCD 码格式)
		0000	0
		0001	1
		0010	2
		0011	3
		0100	4
		0101	5
		0110	6
		0111	7
		1000	8
		1001	9
		1010 1111	无效
Bit[6:4]	10CSE	闹钟模式下秒钟十位数的值(BCD 码格式)	
		10CSE[2:0]	值(BCD 码格式)
		000	0
		001	1
		010	2
		011	3
		100	4
		101	5
		110	6
111	无效		
Bit[3:0]	1CSE	闹钟模式下秒钟个位数的值 (BCD 码格式)	
		1CSE[3:0]	值(BCD 码格式)
		0000	0
		0001	1
		0010	2
		0011	3
		0100	4
		0101	5
		0110	6
		0111	7
		1000	8
		1001	9
		1010 1111	无效

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



28.3.8. RTC 寄存器 7

RTC Base Address + 0x1C(0x41A1C)						
Symbol	RTCYMDA (RTC Year /month/date Control Register For alarm)					
Bit	[31:24]			[23:20]		[19:16]
名称	RSV			10CYE		1CYE
RW	R-0			RW-1		RW-2
Bit	[15:13]	[12]	[11:8]	[07:06]	[05:04]	[03:00]
名称	-	10CMO	1CMO	-	10CDAT	1CDAT
RW	-	RW-0	RW-1	-	RW-0	RW-1

位	名称	描述																										
Bit[23:20]	10CYE	闹钟模式下年份十位数的值 (BCD 码格式)																										
		<table border="1"> <thead> <tr> <th>10CYE[3:0]</th> <th>值(BCD 码格式)</th> </tr> </thead> <tbody> <tr><td>0000</td><td>0</td></tr> <tr><td>0001</td><td>1</td></tr> <tr><td>0010</td><td>2</td></tr> <tr><td>0011</td><td>3</td></tr> <tr><td>0100</td><td>4</td></tr> <tr><td>0101</td><td>5</td></tr> <tr><td>0110</td><td>6</td></tr> <tr><td>0111</td><td>7</td></tr> <tr><td>1000</td><td>8</td></tr> <tr><td>1001</td><td>9</td></tr> <tr><td>1010</td><td rowspan="3">无效</td></tr> <tr><td> </td></tr> <tr><td>1111</td></tr> </tbody> </table>	10CYE[3:0]	值(BCD 码格式)	0000	0	0001	1	0010	2	0011	3	0100	4	0101	5	0110	6	0111	7	1000	8	1001	9	1010	无效		1111
		10CYE[3:0]	值(BCD 码格式)																									
		0000	0																									
		0001	1																									
		0010	2																									
		0011	3																									
		0100	4																									
		0101	5																									
		0110	6																									
		0111	7																									
		1000	8																									
		1001	9																									
		1010	无效																									
1111																												
Bit[19:16]	1CYE	闹钟模式下年份个位数的值 (BCD 码格式)																										
		<table border="1"> <thead> <tr> <th>1CYE[3:0]</th> <th>值(BCD 码格式)</th> </tr> </thead> <tbody> <tr><td>0000</td><td>0</td></tr> <tr><td>0001</td><td>1</td></tr> <tr><td>0010</td><td>2</td></tr> <tr><td>0011</td><td>3</td></tr> <tr><td>0100</td><td>4</td></tr> <tr><td>0101</td><td>5</td></tr> <tr><td>0110</td><td>6</td></tr> <tr><td>0111</td><td>7</td></tr> <tr><td>1000</td><td>8</td></tr> <tr><td>1001</td><td>9</td></tr> <tr><td>1010</td><td rowspan="3">无效</td></tr> <tr><td> </td></tr> <tr><td>1111</td></tr> </tbody> </table>	1CYE[3:0]	值(BCD 码格式)	0000	0	0001	1	0010	2	0011	3	0100	4	0101	5	0110	6	0111	7	1000	8	1001	9	1010	无效		1111
		1CYE[3:0]	值(BCD 码格式)																									
		0000	0																									
		0001	1																									
		0010	2																									
		0011	3																									
		0100	4																									
		0101	5																									
		0110	6																									
		0111	7																									
		1000	8																									
		1001	9																									
		1010	无效																									
1111																												
Bit[12]	10CMO	闹钟模式下月份十位数的值 (BCD 码格式)																										
		<table border="1"> <tbody> <tr><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td></tr> </tbody> </table>	0	0	1	1																						
0	0																											
1	1																											
Bit[11:8]	1CMO	闹钟模式下月份个位数的值 (BCD 码格式)																										
		<table border="1"> <thead> <tr> <th>1CMO[3:0]</th> <th>值(BCD 码格式)</th> </tr> </thead> <tbody> <tr><td>0000</td><td>0</td></tr> </tbody> </table>	1CMO[3:0]	值(BCD 码格式)	0000	0																						
		1CMO[3:0]	值(BCD 码格式)																									
0000	0																											

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述	
		0001	1
		0010	2
		0011	3
		0100	4
		0101	5
		0110	6
		0111	7
		1000	8
		1001	9
		1010 1111	无效
Bit[5:4]	10CDAT	闹钟模式下日期十位数的值 (BCD 码格式)	
		10CDAT[3:0]	值(BCD 码格式)
		00	0
		01	1
		10	2
		11	3
Bit[3:0]	1CDAT	闹钟模式下日期个位数的值 (BCD 码格式)	
		1CDAT[3:0]	值(BCD 码格式)
		0000	0
		0001	1
		0010	2
		0011	3
		0100	4
		0101	5
		0110	6
		0111	7
		1000	8
		1001	9
		1010 1111	无效

29. 省电模式介绍

29.1. 整体总说明

本节说明不同的电源模式，能够开启哪些功能模块。活动模式是所有的的外围电路，都能被打开，且 MCU 是 HS_CK 或 LS_CK 频率；在活动模式下，系统能自由地跳到其它模式，且响应时间最短。低电力模式是 MCU 是频率选择为 LS_CK；在此模式下，MCU 以最低的频率运作，系统可经由指令的执行跳到其它的模式。

有 3 种省电模式，分别为 Sleep Mode, Idle Mode, Wait mode, 可让 MCU 停止执行指令。这些模式可由中断(interrupt)来跳脱。一旦 interrupt 被触发，MCU 就会离开省电模式。芯片进入省电模式前，必须要开启任何一个可唤醒的对应中断向量，否则无法达到省电的功效。例如：在 Sleep Mode，定时器中断是无效的，且只能通过通信中断、IO 外部中断及复位来唤醒芯片。详细可参考下表有列出各省电模式下所对应的唤醒中断向量表。尤其需要注意在不同的省电模式下，只能开启一些功能模块，只有一些功能才能将芯片从省电模式唤醒。

中断进入点设置:

CPU 在不同工作模式下，可支持中断触发项目亦不相同，下表说明在不同模式下，各功能支持中断与唤醒程度。当然不同模式下消耗电流亦不相同，消耗电流大小：Active Mode(工作模式) > Wait Mode(等待模式) > Idle Mode(待机模式) > Sleep Mode(睡眠模式)。需要注意的是当进入 Idle Mode 或 Sleep Mode 时，如要达到更省电设置需求，则应该在进入省电模式之前，把 CPU 的工作频率先切换到 LPO 低频之后，再把 HAO 高频关掉。如有开启相关模拟电源输出，则也需要做相对应的关闭动作，这样进入省电模式之后才可以达到正确的耗电流等级。唤醒时间：Sleep Mode(睡眠模式)> Idle Mode(待机模式)> Wait Mode(等待模式)。Sleep Mode 与 Idle Mode 虽然比 Wait Mode 都还省电许多，但是透过中断唤醒时间相对来说则比较长。中断唤醒程度：如 I²C TX 引脚的中断功能仅可以支持 Idle Mode、Wait Mode、Active Mode，也就是说当芯片进入 Sleep mode 时，是无法透过 I²C TX 引脚信号唤醒芯片，使芯片可以进入断点。如芯片进入 Sleep mode 之后，也仅有以下动作与中断才能使芯片离开睡眠模式，Power On Reset(BOR1/BOR2)、Reset PIN、I²C RX IRQ、UART1/2 RX IRQ、SPI RX IRQ、PT1 IRQ、PT2 IRQ、PT3 IRQ 等

Interrupt/Reset Mode	Sleep Mode		Idle Mode		Wait Mode		Active Mode		Note
	进入	离开	进入	离开	进入	离开	进入	离开	
Power On Reset		V		V		V	V	V	Chip Reset
Reset PIN		V		V		V	V	V	Chip Reset
WDT Reset				V		V	V	V	WDT Reset Type
I ² C TX IRQ			V	V	V	V	V	V	I2CIE
I ² C RX IRQ	V	V	V	V	V	V	V	V	I2CIE
I ² C Error IRQ						V	V	V	I2CEIE
UART1/2 TX IRQ			V	V	V	V	V	V	UTXIE
UART1/2 RX IRQ	V	V	V	V	V	V	V	V	URXIE
SPI TX IRQ			V	V	V	V	V	V	STXIE
SPI RX IRQ	V	V	V	V	V	V	V	V	SRXIE
RTC IRQ			V	V	V	V	V	V	RTCIE
WDog IRQ			V	V	V	V	V	V	WDTIE
TMA IRQ			V	V	V	V	V	V	TMAIE
TMB/TMB2 IRQ			V	V	V	V	V	V	TMBIE/TMB2IE
TMC IRQ			V	V	V	V	V	V	TMCIE
ADC IRQ			V	V	V	V	V	V	ADCIE
PT3 IRQ	V	V	V	V	V	V	V	V	PT3IE

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



Interrupt/Reset Mode	Sleep Mode		Idle Mode		Wait Mode		Active Mode		Note
	进入	离开	进入	离开	进入	离开	进 入	离 开	
PT2 IRQ	V	V	V	V	V	V	V	V	PT2IE
PT1 IRQ	V	V	V	V	V	V	V	V	PT1IE
BOR2 IRQ	V	V	V	V	V	V	V	V	BOR2IE
LVD IRQ	V	V	V	V	V	V	V	V	LVDIE
Debug Exception						V	V	V	EDM

30. 液晶驱动器 LCD

30.1. 整体总说明

液晶驱动电路是用于 TN-LCD 与 STN-LCD 等制成的液晶显示，其具有以下特点：

内置倍压稳压电路(Regulated charge pump)

四段可调式驱动电压准位

支持 1/3 Bias 或 1/4 Bias 偏压选择

支持多种 LCD 波形的操作方式

1/3 Duty , 1/3 偏压。(3-mux,1/3bias)

1/4 Duty , 1/3 偏压。(4-mux,1/3bias)

1/5 Duty , 1/3 偏压。(5-mux,1/3bias)

1/6 Duty , 1/3 偏压。(6-mux,1/3bias)

1/7 Duty , 1/3 偏压。(7-mux,1/3bias)

1/8 Duty , 1/3 偏压。(8-mux,1/3bias)

可选择输入频率源与可规划输出频率

具闪烁控制功能(Blinking capability)

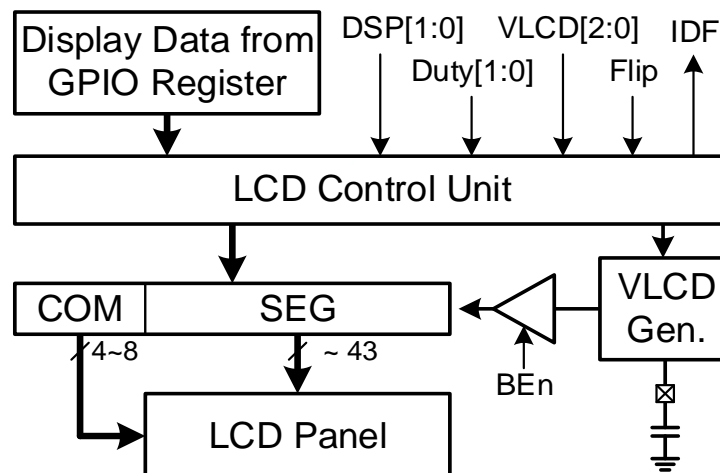
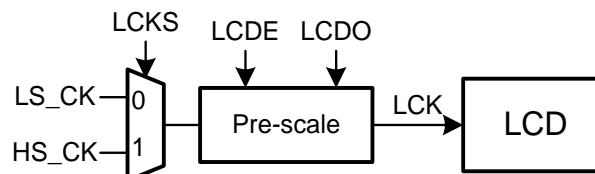


图 30-1 LCD 架构图

LCD 初始化设置:

- (1) 工作频率与输出振幅频率设置，LCD 工作频率可由寄存器的控制位 LCKS 0x40310[0]来选择 LS_CK 或 HS_CK 提供，经过寄存器的控制位 LCDE 0x40310[3:1]与控制位 LCDO 0x40310[6:4]来做频率源 1 阶与 2 阶除频设置，提供适当的工作频率给 LCD 输出振幅频率。



- (2) 倍压电源与 LCD 工作电压设置电压源为 VLCD，其有两种产生方法：由外部输入 VLCD 电压源，必须将寄存器的控制位 VLCD 0x41B00[2:0]设置为<001>VLCD R-Type，然后由外部 VLCD 引脚输入电压以决定 LCD 工作电压。当推动尺寸或负载较大的 LCD 显示器时，可将 LCD 输出缓冲 BEn 0x41B00[3]

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



设置<1>，启用缓冲器以增加 LCD 的驱动能力。将被压电路控制器 0x41B00[2:0]可产生不同 VLCD 电压源供给 LCD。VLCD 0x41B00[2:0]可设置多种不同工作电压且必须在倍压电路启用时才有效。

- (3) 寄存器的控制位 Duty 0x41B00[6:4]可设置 LCD 操作波形，振幅频率与操作波形必须依外接 LCD 显示器的规格做正确的设置，否则 LCD 显示器会出现鬼影或字节显示异常等现象。
- (4) 设置 LCD 的复用 IO 引脚 PT6~PT13 的工作模式，即设置寄存器 0x41B04~0x41B08。
- (5) 写入数据到 LCD 数据寄存器 LCD0~LCD17，做 LCD 数据显示。

30.2. LCD 寄存器地址

LCD Register Address	31	24	23	16	15	8	7	0
LCD Base Address + 0x00 (0x41B00)	Mask0		REG0		Mask1		REG1	
LCD Base Address + 0x04 (0x41B04)	PT9LEn		PT8LEn		PT7LEn		PT6LEn	
LCD Base Address + 0x08 (0x41B08)	PT13LEn		-		-		PT10LEn	

-保留

30.3. 寄存器功能

30.3.1. LCD 寄存器 0

LCD Base Address + 0x00 (0x41B00)							
Symbol	LCDCR0 (LCD Control Register 0)						
Bit	[31:24]	[23:22]	[21]	[20]	[19]	[18]	[17:16]
名称	MASK	LCDCPD	LCDBS	IDF	VLCDEN	-	DSP
RW	R0W-0	R-0		R-1	-	RW-0	
Bit	[15:08]	[07]	[06:04]	[03]	[02:00]		
名称	MASK	LCDEN	Duty	BEn	VLCD		
RW	R0W-0	RW-0				RW-0	

位	名称	描述
Bit[23:22]	LCDCPD	LCD 电荷泵调器频率源选择 设置 LCD internal charge pump 扫描频率，可依实际显示效果及功耗需求作配置
		0 LS_CK/1 or HS_CK/8(LS_CK 或 HS_CK 由 LCKS 决定)
		1 LS_CK/2 or HS_CK/16(LS_CK 或 HS_CK 由 LCKS 决定)
		2 LS_CK/4 or HS_CK/32(LS_CK 或 HS_CK 由 LCKS 决定)
		3 LS_CK/4 or HS_CK/32(LS_CK 或 HS_CK 由 LCKS 决定)
Bit[21]	LCDBS	LCD 偏压控制位
		0 1/3 Bias
		1 1/4 Bias
Bit[20]	IDF	LCD Idle 控制标志位
		0 Active
		1 Idle
Bit[19]	VLCDEN	VLCD 升压稳压启动控制器
		0 VLCD Pump OFF. 此时 VLCD 可由外部输入电压, R- Type.

HY16F3910 User's Guide

21-bit ENOB ΣADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver



位	名称	描述	
		1	VLCD Pump ON.
Bit[17:16]	DSP	LCD 显示模式	
		00	正常模式
		01	不论输入何值, LCD 全亮
		10	不论输入何值, LCD 全灭
		11	正常模式
Bit[7]	LCDEN	LCD 启动控制器(LCD Clock 输出至 SEG/COM Port)	
		0	LCD OFF
		1	LCD ON
Bit[6:4]	Duty	LCD 工作周期选择	
		000	1/3 Duty
		001	1/4 Duty
		010	1/5 Duty
		011	1/6 Duty
		100	1/7 Duty
		101	1/8 Duty
Bit[03]	BEn	VLCD 缓冲器控制	
		0	关闭
		1	开启(必须启动才能正常使用 LCD 功能)
Bit[2:0]	VLCD	VLCD 升压稳压电压文件位选择	
		000	保留
		001	保留
		010	2.8V
		011	3.0V
		100	3.3V
		101	3.94V
		110	4.5V
111	5.0V		

30.3.2. LCD 寄存器 1

LCD Base Address + 0x04 (0x41B04)		
Symbol	LCDCR1 (LCD Control Register 1)	
Bit	[31:24]	[23:16]
名称	PT9LEn	PT8LEn
RW	RW-0	
Bit	[15:08]	[07:00]
名称	PT7LEn	PT6LEn
RW	RW-0	

位	名称	描述	
Bit[31:24]	PT9LEn	PT9.# 模式选择, #代表的是 7~0	
		0	GPIO 模式
		1	LCD 模式
Bit[23:16]	PT8LEn	PT8.# 模式选择, #代表的是 7~0	

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

位	名称	描述	
		0	GPIO 模式
		1	LCD 模式
Bit[15:08]	PT7LEn	PT7.# 模式选择, #代表的是 7~0	
		0	GPIO 模式
		1	LCD 模式
Bit[07:00]	PT6LEn	PT6.# 模式选择, #代表的是 7~0	
		0	GPIO 模式
		1	LCD 模式

30.3.3. LCD 寄存器 2

LCD Base Address + 0x08 (0x41B08)		
Symbol	LCDCR2 (LCD Control Register 2)	
Bit	[31:24]	[23:16]
名称	PT13LEn	Rsv
RW	RW – FF	
Bit	[15:08]	[07:00]
名称	-	PT10LEn
RW	R-0	RW-0

位	名称	描述	
Bit[31:24]	PT13LEn	COM0 ~ COM7 IO 模式设定	
		0	GPIO 模式
		1	LCD 模式
Bit[7:0]	PT10LEn	PT10.# 模式选择, #代表的是 7~0	
		0	GPIO 模式
		1	LCD 模式

HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

30.4. LCD RAM 功能

LCD Register Address 0x41B04 与 0x41B08 可决定 PT6~PT13 设定为 GPIO Mode 或是 LCD Mode。当设定为 LCD Mode 时，可将 PT6~PT13 IO 寄存器当成 LCD RAM 使用控制 LCD 显示器。

LCD Mode Address	Bit[31:24]	Bit[23:16]	Bit[15:08]	Bit[07:00]
0x40850	MASK	SEG3	MASK	SEG2
0x40854	MASK	SEG5	MASK	SEG4
0x40858	MASK	SEG7	MASK	SEG6
0x4085C	MASK	SEG9	MASK	SEG8
0x40860	MASK	SEG11	MASK	SEG10
0x40864	MASK	SEG13	MASK	SEG12
0x40868	MASK	SEG15	MASK	SEG14
0x4086C	MASK	SEG17	MASK	SEG16
0x40870	MASK	SEG19	MASK	SEG18
0x40874	MASK	SEG21	MASK	SEG20
0x40878	MASK	SEG23	MASK	SEG22
0x4087C	MASK	SEG25	MASK	SEG24
0x40880	MASK	SEG27	MASK	SEG26
0x40884	MASK	SEG29	MASK	SEG28
0x40888	MASK	SEG31	MASK	SEG30
0x4088C	MASK	SEG33	MASK	SEG32
0x40890	MASK	SEG35	MASK	SEG34
0x40894	MASK	SEG37	MASK	SEG36
0x40898	MASK	SEG39	MASK	SEG38
0x4089C	MASK	SEG41	MASK	SEG40
0x408C8	MASK	SEG1	MASK	SEG0
0x408CC	MASK	SEG43	MASK	SEG42

30.5. LCD 省电功能

当 HY16F3910 要进入省电模式时候，LCD 的设置也需注意。在进入省电前，若没先放电，LCD 可能就会有鬼影现象。在进入省电模式的时候，可参照以下设置，可确保 LCD 有先放电，再进入省电模式。

```
DrvLCD_DisplayMode (2); //2 : 不论输入任何值，LCD 都是全灭模式  
DrvLCD_VLCDMode(E_VLCD_DISABLE);  
while((inw(0x41B00)&(1<<IDF))==0); //Wait LCD Idle, IDF=20
```


HY16F3910 User's Guide

21-bit ENOB ΣΔADC, 32-bit MCU & 128k Byte Flash

4 X 44~8 X 40 LCD Driver

31. 修订记录

以下描述本文件差异较大的地方，而标点符号与字形的改变不在此描述范围。

文件版次	页次	日期	摘要
V02	ALL	2022/9/08	初版发行