Fremont Micro Devices

FT62F28

数据手册

主要特性

8-bit 基于 EEPROM 的 RISC MCU

Program: 8k x 14; RAM: 616 x 8; Data: 128 x 8

20 / 24 / 28 / 32 引脚

高精度 12-bit ADC

4 个定时器, 4 路独立 PWM - 1 路带死区控制

高可靠性的 28 路触摸按键

运算放大器，2 路模拟比较器

SPI，I2C，2x USART

低 Standby, WDT 和工作电流

POR, LVR, LVD – 单输入比较器

可配置源电流和灌电流

高 ESD，高 EFT

低 V_{DD} 工作电压

HIRC 可微调

Rev2.02

www.fremontmicro.com

8-bit CPU (EEPROM)

- 37 条 RISC 指令: 2T or 4T
- 16 MHz / 2T (V_{DD} ≥ 2.7)
- 多达 32 个引脚

Memory

- PROGRAM: 8k x 14 bit (读/写保护)
- DATA: 128 x 8 bit (读保护)
- RAM: 616 x 8 bit
- 8 层硬件堆栈
- 扇区加密

工作条件 (5V, 25°C)

- V_{DD} (V_{POR} ≤ 1.9V) V_{POR} – 5.5 V
  (通过 POR 自动调整，0°C 以上 ≤1.7V)
- 工作温度等级 1 ⁻40 – ⁺125 °C
- 工作温度等级 2 ⁻40 – ⁺105 °C
- 工作温度等级 3 ⁻40 – ⁺85 °C
- 低 Standby 0.6 μA
- WDT 1.8 μA
- 正常模式 (16 MHz / 2T) 81 μA/mips

高可靠性

- 100 万次擦写次数 (typical)
- > 20 年 / 85°C 存储 (typical)
- ESD > 8 kV, EFT > 5.5 kV

ADC (12-bit)

- 12-bit 精度 (≤ 1 MHz ADC 时钟)
- 30 + 2 通道
- $V_{ADC-REF}$
  - ✓ 内部: 0.5, 2.0, 3.0, V_{DD}
  - ✓ 外部: +, – 可选
- 自动阈值比较和中断，支持延时触发
- 触发方式: 手动，自动可选

PWM (Total 5)

- 支持在 SLEEP 下运行
- 4 个 PWM 通道 (相同周期)：
  - ✓ 独立: 占空比，极性
- 1 个通道 (多达 2 个 I/O)：互补输出+死区
- 自动故障刹车 (I/O, LVD, ADC)
- 单脉冲模式，蜂鸣器模式

Timers

- WDT (16-bit): 7-bit 后分频
- Timer0 (8-bit): 8-bit 预分频
- Timer1 (12-bit)
- Timer2 (16-bit): 4-bit 预分频和后分频
- 支持在 SLEEP 下运行
- LIRC, 1 or 2x {指令时钟, HIRC，晶振, EC}

TOUCH

- 多达 28 个触摸按键

运算放大器

- 手动校准，输出到 IO / ADC 通道

模拟比较器 (2 路)

- 独立和窗口比较模式，极性，可编程 Vref
- 输入: IO，7bit DAC，运放输出

通信接口

- SPI, I2C, 2 x USART

I/O PORTS (多达 30 个 I/O)

- 上拉/下拉电阻，开漏
- 30 个 I/O 源电流: 3, 6 or 24mA (5V, 25°C)
- 30 个 I/O 漏电流: 53 or 70 mA (5V, 25°C)
- 8 个 I/O: 中断唤醒

电源管理

- SLEEP
- LVR: 2.0, 2.2, 2.5, 2.8, 3.1, 3.6, 4.1 (V)
- LVD: 1.2, 2.0, 2.4, 2.7, 3.0, 3.3, 3.6, 4.0 (V)
  (LVD 可用作极性可选的单输入比较器功能)

系统时钟 (SysClk)

- HIRC 高速内部振荡器 (可微调)
  - ✓ 16MHz <±1% typical (2.5-5.5V, 25°C)
  - ✓ 1, 2, 4, 8, 16, 32, 64 分频
- LIRC 低功耗低速内部振荡器
  - ✓ 32 kHz 或 256 kHz
- EC 外部时钟 (I/O 输入)
- LP / XT 晶振输入
  - ✓ 双速时钟启动 (HIRC 或 LIRC)
  - ✓ 故障保护时钟监控

其他特性 (欢迎垂询)

- 13.56 MHz 载波
- 用于锂电池的 3.2V – 4.7V 监控
- ½ V_{DD} LCD 偏置

集成开发环境 (IDE)

- 片上调试 (OCD)，ISP
- 3 个硬件断点
- 软复位，暂停，单步，运行等

封装

- SOP20 TSSOP20 TSSOP24 SOP28 QFN32

产品信息和选型表

型号	I/Os	封装
FT62F285A- ab	18	SOP20
FT62F285A- Tab		TSSOP20
FT62F286A- Tab	22	TSSOP24
FT62F287A- ab	26	SOP28
FT62F287B- ab		
FT62F287C- ab		
FT62F288A- Nab	30	QFN32

此处 a = R; RHOS b = B; Tube
= G; Green = T; T&R

F T 6 2 F 2 8 8 A _ - N R B

6: 8-bit

Type
0: GPIO
1: ADC
2: Touch
3: LCD
4: Opamp
7: Opamp + 485

F: EEPROM

Code Size
1: 1k
2: 2k
3: 3k
4: 4k
8: 8k
A: 10k
G: 16k

# of Pins
0: 6
1: 8
F: 10
2: 14
3: 16
4: 18
5: 20
6: 24
7: 28
8: 32
9: 36
A: 40
B: 44
C: 48
D: 64
E: 100

Version
(empty)
A - Z

Temperature Grade
(empty)/3: -40 ~ 85℃
2: -40 ~ 105℃
1: -40 ~ 125℃

B: Tube
T: T&R

R: RoHS
G: Green

Packaging
(empty) SOP
D: DIP
E: SSOP
F: TQFP
L: LQFP
M: MSOP
N: QFN
Q: QFP
T: TSSOP
U: SOT23
H: DFN

MCU 产品订购信息

目录

1. 结构框图和引脚 10
1.1. 引脚图 12
1.2. 管脚描述---按功能分类 15
2. GPIO 21
2.1. IO 端口相关寄存器汇总 25
2.2. I/O 配置 31
2.3. PORTx 功能及优先级 33
3. 上电复位 34
3.1. 上电复位流程 34
3.2. 上电校验过程 34
3.3. 上电复位延时 35
4. 系统复位 36
4.1. 系统复位相关寄存器汇总 37
4.2. 低电压复位 37
4.3. 非法指令复位 38
4.4. 上电配置过程(BOOT) 38
4.5. 复位源标志位 39
5. LVD 低电压侦测 40
5.1. LVD 相关寄存器汇总 40
5.2. 检测外部电压 41
5.3. LVD 中断 41
6. 振荡器和系统时钟 42
6.1. 振荡器模块相关寄存器汇总 43
6.2. 时钟源模式 45
6.3. 外部时钟模式 45
6.3.1. 振荡器起振定时器 (OST) 45
6.3.2. EC 模式 45
6.3.3. LP 和 XT 模式 45
6.4. 内部时钟模式 45

6.4.1. 频率选择位 (IRCF) 46
6.4.2. HIRC 和 LIRC 时钟切换时序 46
6.4.3. 频率与最小工作电压的关系 47
6.4.4. HIRC 频率微调 48
6.5. 时钟切换 48
6.5.1. 系统时钟选择 (SCS) 位 48
6.5.2. 振荡器起振超时状态 (OSTS) 位 49
6.5.3. 双速时钟启动模式 49
6.5.4. 双速启动模式配置 49
6.5.5. 双速启动顺序 49
6.5.6. 故障保护时钟监控器 50
6.5.7. 故障保护检测 50
6.5.8. 故障保护操作 50
6.5.9. 故障保护条件清除 50
6.5.10. 复位或从休眠中唤醒 51
6.6. 系统时钟输出 51
7. 慢时钟测量 52
7.1. 慢时钟测量相关寄存器汇总 52
7.2. 测量原理 53
7.3. 操作步骤 53
8. 看门狗定时器 55
8.1. 看门狗时钟源 55
8.2. WDT 相关寄存器汇总 56
9. 定时器 0 57
9.1. Timer0 相关寄存器汇总 58
9.2. Timer0 定时器模式 59
9.2.1. Timer0 的时钟源 59
9.2.2. TMR0 寄存器的读写 59
9.3. Timer0 计数器模式 60
9.3.1. 软件可配置预分频电路 60

9.3.1.1. 在定时器和 watchdog 之间切换预分频电路 60
9.3.2. 定时器 0 中断 60
9.3.3. 用外部时钟驱动定时器 0 61
9.3.4. 睡眠模式下的状态 61
9.4. Timer0 的 PWM 模式 61
9.4.1. 周期和占空比 61
9.4.2. 睡眠时的状态 61
10. 定时器 1 62
10.1. Timer1 相关寄存器汇总 63
10.2. Timer1 工作原理 65
10.3. Timer1 计数值的读写 65
10.4. Timer1 的自动关闭 65
11. 定时器 2 66
11.1. Timer2 相关寄存器汇总 67
11.2. Timer2 工作原理 68
11.3. 关于 PR2 的更新 69
11.4. Timer2 计数值的读写 69
12. PWM 模块 71
12.1. PWM 相关寄存器汇总 72
12.2. 周期 77
12.3. 占空比 77
12.4. 时钟源选择 77
12.5. 睡眠下 PWM 状态 77
12.6. P1A 的死区时间 77
12.7. 故障刹车 78
12.7.1. 刹车状态 78
12.7.2. 故障清除 78
12.7.3. 自动重启 79
12.7.4. 前沿消隐 79
12.8. 周期和占空比寄存器的更新 80

12.9. 蜂鸣器模式 (Buzzer) 81
12.10. 单脉冲输出 82
12.11. PWM1 输出重映射 82
12.12. P1C、P1D 的第 2 功能输出 82
13. 睡眠模式 84
13.1. 进入 SLEEP 84
13.2. 睡眠的唤醒 85
13.3. 看门狗唤醒 85
13.4. 中断唤醒 85
14. 中断 86
14.1. 中断相关寄存器汇总 88
14.2. INT 外部中断 93
14.3. 端口变化中断 93
14.3.1. PAIF 标志位的清除 93
14.4. 中断响应 93
14.5. 中断过程中的现场保存 94
14.6. 关于中断标志位 94
15. 数据 EEPROM 95
15.1. 与数据 EEPROM 相关寄存器汇总 95
15.2. 编程数据 EEPROM 步骤 96
15.3. 读数据 EEPROM 97
15.4. 关于编程周期 98
15.5. EEPROM 的单编程模式 98
15.6. 关于最后 16 个字节 98
16. 12bit ADC 模块 99
16.1. ADC 相关寄存器汇总 99
16.2. ADC 的配置 103
16.2.1. 端口配置 103
16.2.2. 通道选择 104
16.2.3. 触发方式选择 104

16.2.4. 触发源选择 104
16.2.5. 触发类型选择 104
16.2.6. 触发延时配置 104
16.2.7. ADC 参考电压 104
16.2.8. 转换时钟 105
16.2.9. 中断 106
16.2.10. 转换结果的格式 106
16.2.11. 阈值比较 107
16.3. ADC 的工作原理 108
16.3.1. 启动转换 108
16.3.2. 转换完成 108
16.3.3. 终止转换 108
16.3.4. 休眠模式下 ADC 的工作 108
16.3.5. A/D 转换步骤 109
16.3.6. A/D 采集时间要求 110
17. 运放 112
17.1. 运放 0 相关寄存器汇总 113
17.2. 运放 0 典型应用 114
17.3. 运放 0 输出 114
18. 比较器模块 115
18.1. 比较器相关寄存器汇总 116
18.2. 比较器控制 118
18.2.1. 比较器输入 118
18.2.2. 比较器输出 118
18.2.3. 比较器输出去抖 119
18.3. 比较器中断 119
18.4. 睡眠状态下的工作 119
18.5. 比较器的自动使能 119
18.6. 参考电压 120
18.6.1. 输出电压 121

19. SPI 接口 122
19.1. SPI 相关寄存器汇总 123
19.2. SPI 功能描述 127
19.3. SPI 配置 128
19.3.1. 数据处理流程 129
19.3.2. 睡眠模式唤醒 130
19.3.3. CRC 处理流程 130
20. I2C 接口 132
20.1. I2C 接口相关寄存器汇总 133
20.2. I2C 配置 136
20.2.1. 主机时序 136
20.2.2. 主机时序 137
20.2.3. 从机发送 140
20.2.4. 主机接收 142
20.2.5. 从机接收 144
20.2.6. 中断映射 147
21. 通用异步收发器 (USART0/1) 148
21.1. USART 接口相关寄存器汇总 149
21.2. 功能描述 152
21.2.1. 一般描述 152
21.2.2. 异步工作模式 152
21.2.3. 半双工模式 153
21.2.4. 中断映射 154
22. 电容按键模块 155
22.1. 触摸按键功能 155
22.2. 触摸按键结构 155
23. 存储区读/写保护 155
24. 程序存储器 156
将程序存储器当作数据存储器读取 156
25. 特殊功能寄存器 (SPECIAL FUNCTION REGISTERS, SFR) 158

25.1. 初始化配置寄存器 158
25.2. 用户寄存器 161
25.3. STATUS 寄存器 169
25.4. PCL 和 PCLATH 170
25.5. INDF 和 FSR 寄存器 171
26. 指令集汇总 173
27. 芯片的电气特性 175
27.1. 极限参数 175
27.2. 工作特性 175
27.3. POR, LVR, LVD 176
低电压复位 LVR) 176
低电压侦测 LVD) 176
27.4. I/O PAD 电路 177
27.5. 工作电流(I_{DD}) 177
27.6. 内部荡器 178
内置高频振荡器(HIRC) 178
27.7. 7bit DAC 电路 (比较器参考电压设置) 178
27.8. ADC(12bit) 和 ADC VREF 179
27.9. 运算放大器 0 特性 180
27.10. Comparator 比较器 180
27.11. 存储器编程特性 181
27.12. EMC 特性 181
27.13. 直流和交流特性曲线图 182
28. 芯片封装信息 187
附录 1，寄存器类型 192
附录 2，文档更改历史 193
联系信息 ..... **错误!未定义书签。**

1. 结构框图和引脚

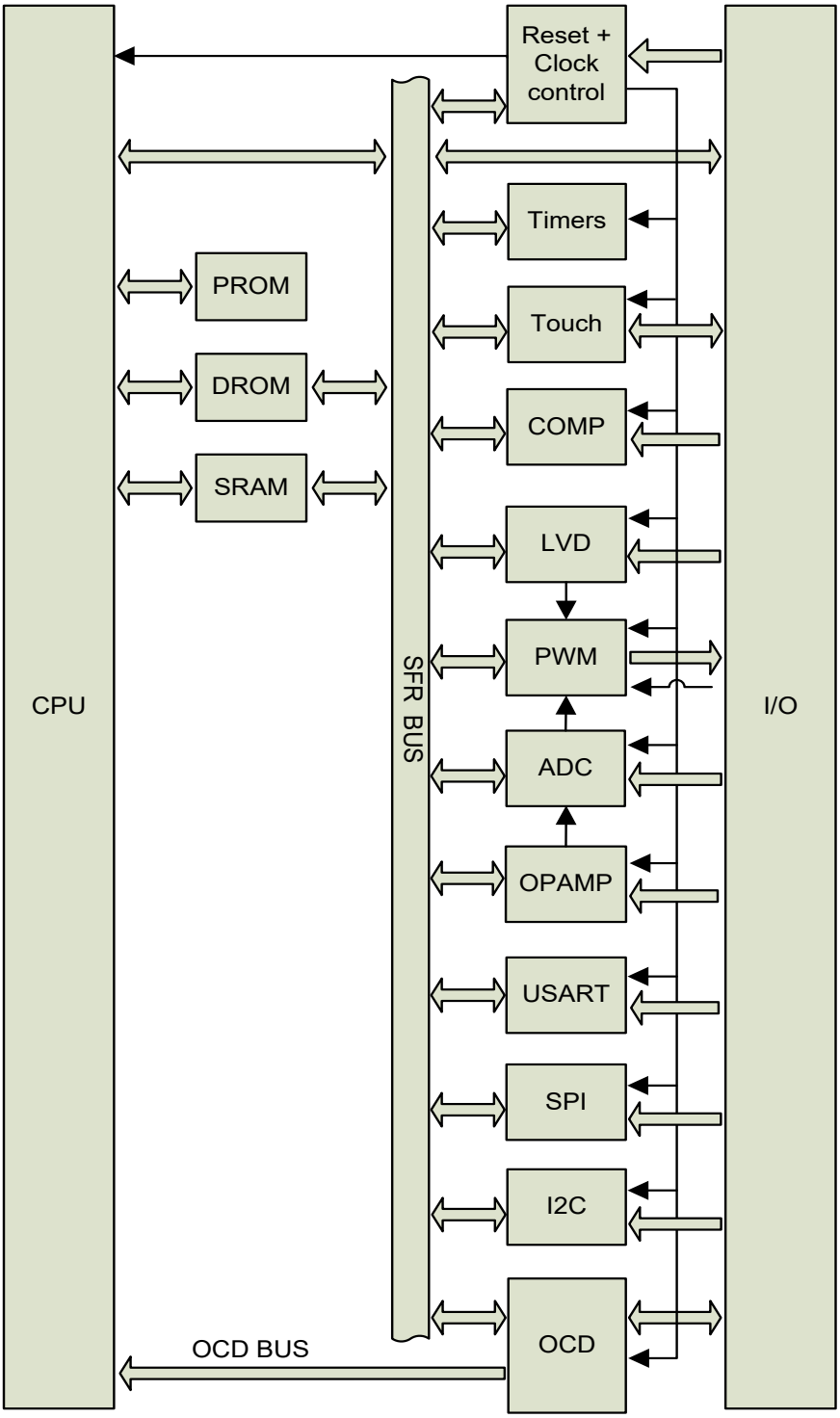


图 1-1 系统结构框图

标准缩写列表如下:

缩写	描述
CPU	Central Processing Unit
SFR	Special Function Registers
SRAM	Static Random Access Memory
DROM	Data EEPROM
PROM	Program EEPROM
Timers	WDT, Timer0, Timer1, Timer2
PWM	Pulse Width Modulator
ADC	Analog to Digital Converter
OPAMP	Operational amplifier
COMP	Comparator
LVD	Low Voltage Detect / comparator
Touch	Touch
SPI	Serial Peripheral Interface
USART	Universal Synchronous Asynchronous Receiver Transmitter
I2C	Inter-Integrated Circuit bus (I2C-bus)
OCD	On Chip Debug
I/O	Input / Output

1.1. 引脚图

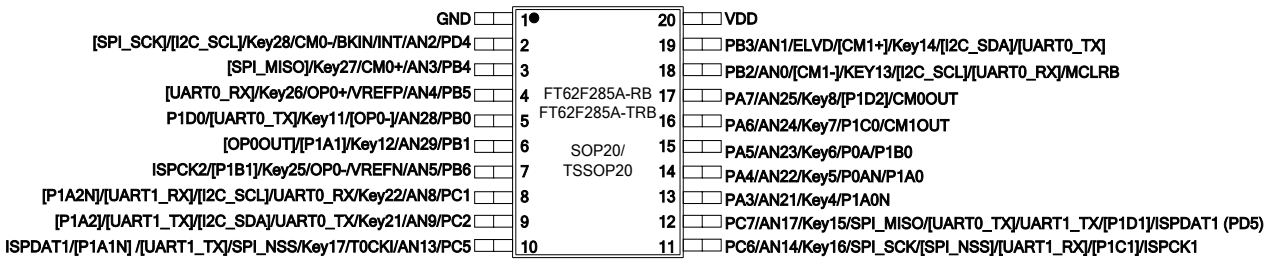


图 1-2 SOP20 / TSSOP20 [1]

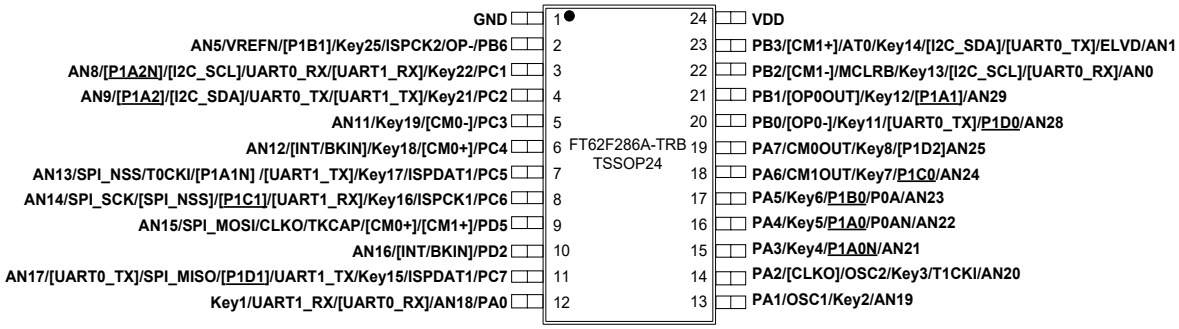


图 1-3 TSSOP24 [2]

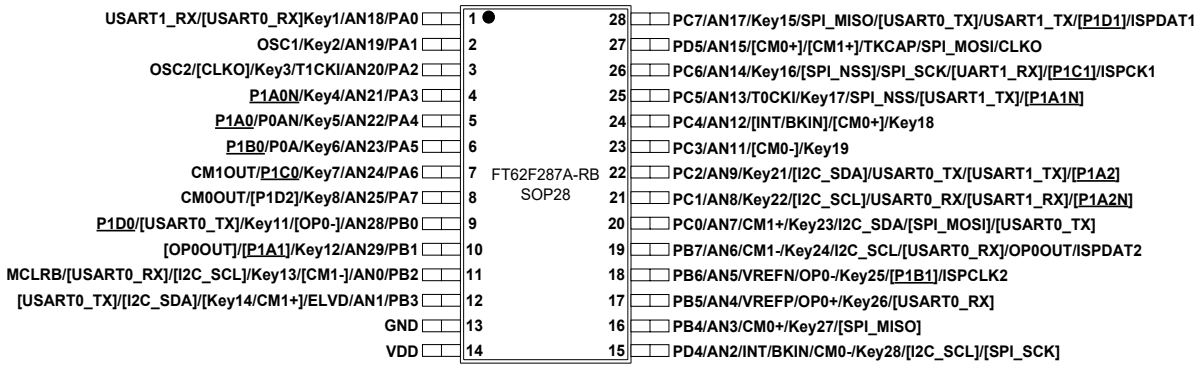


图 1-4 SOP28 (A) [2]

[1] 对于 20-pin 封装，PC7 与 PD5 共同打线到 pin-12。如果需要将 PC7 和 PD5 同时设置为输出，则需满足 PORTC7 = PORTD5，另外上拉/下拉的设置也不能与输出值相反。此外，因 PC7 (ISPDAT1) 为烧录调试时钟脚，而 PC7 和 PD5 打线在一起，因此在芯片复位后的 20ms 内，禁止将 PD5 设置成输出，以免影响烧录调试功能；

[2] 对于 FT62F286A / FT62F287A 封装，PD3 与 GND 打线在一起，因此禁止将其设置成输出高或使能内部上拉；

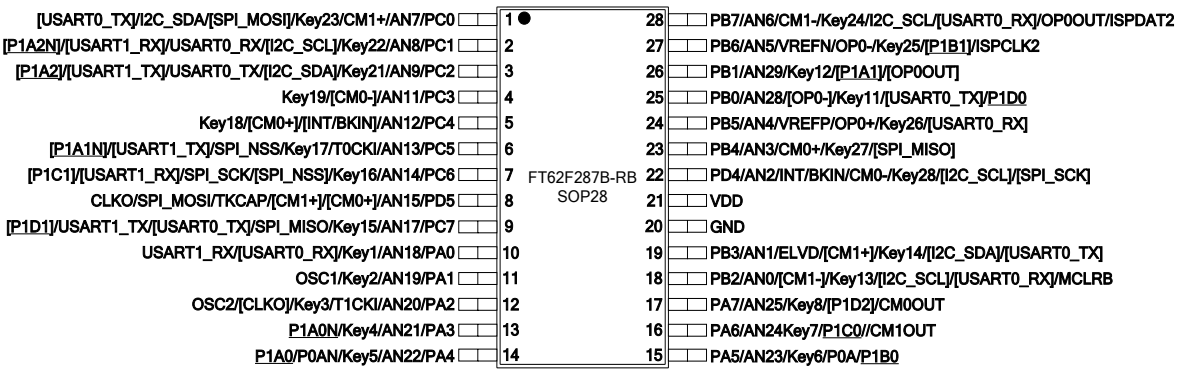


图 1-5 SOP28 (B)

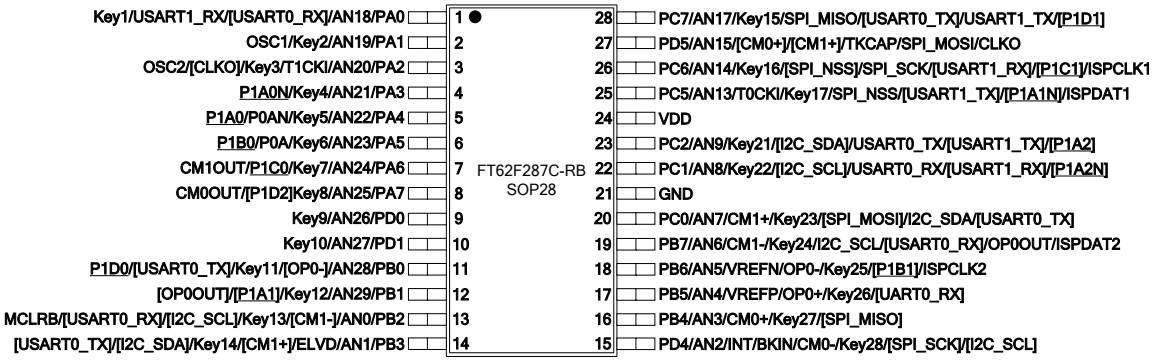


图 1-6 SOP28 (C) [3]

[3] 对于 FT62F287C 封装，PD3 与 VDD 打线在一起，因此禁止将其设置成输出低或使能内部下拉；

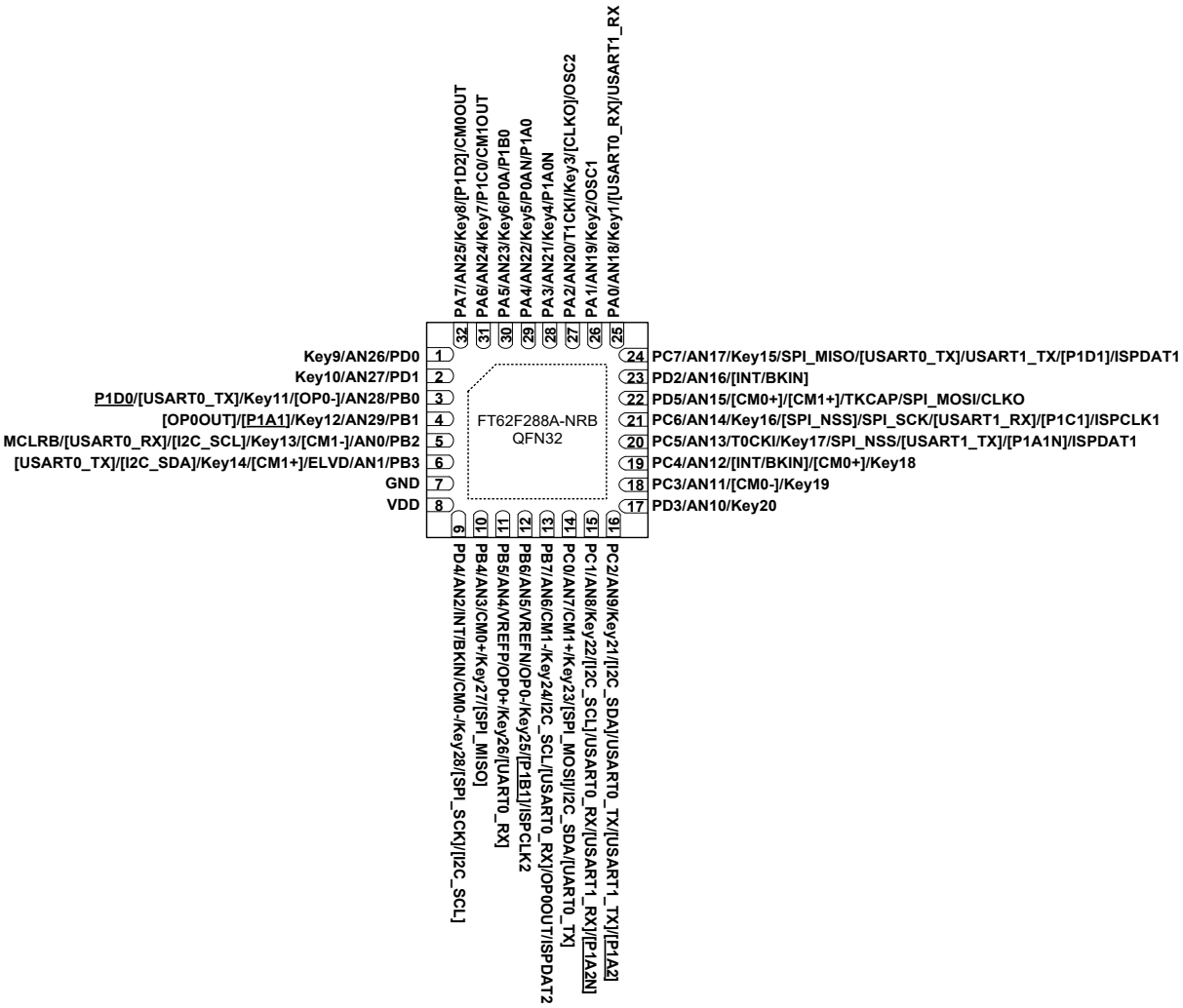


图 1-7 QFN32 [4]

[4] 当 PD3 外接至 GND 时，ISPDAT1 的有效引脚为 PC7；当 PD3 外接至 VDD 时，ISPDAT1 的有效引脚为 PC5；

1.2. 管脚描述---按功能分类

功能	描述	引脚名	对应 GPIO	20 pins	24 pins	28(A) pins	28(B) pins	28(C) pins	QFN32 pins
电源		VDD		20	24	14	21	24	8
		GND		1	1	13	20	21	7
GPIO	上拉/下拉, 数字输入, 数字输出 开漏	PD5		12 [1]	9	27	8	27	22
		PD4		2		15	22	15	9
		PD3							17
		PD2			10				23
		PD1						10	2
		PD0						9	1
		PC7		12 [1]	11	28	9	28	24
		PC6		11	8	26	7	26	21
		PC5		10	7	25	6	25	20
		PC4			6	24	5		19
		PC3			5	23	4		18
		PC2		9	4	22	3	23	16
		PC1		8	3	21	2	22	15
		PC0				20	1	20	14
		PB7				19	28	19	13
		PB6		7	2	18	27	18	12
		PB5		4		17	24	17	11
		PB4		3		16	23	16	10
		PB3		19	23	12	19	14	6
		PB2		18	22	11	18	13	5
		PB1		6	21	10	26	12	4
		PB0		5	20	9	25	11	3
		PA7		17	19	8	17	8	32
		PA6		16	18	7	16	7	31
		PA5		15	17	6	15	6	30
		PA4		14	16	5	14	5	29
		PA3		13	15	4	13	4	28
		PA2			14	3	12	3	27
		PA1			13	2	11	2	26
		PA0			12	1	10	1	25
LVD	输入	ELVD	PB3	19	23	12	19	14	6
外部复位	外部复位管脚	/MCLRB	PB2	18	22	11	18	13	5

功能	描述	引脚名	对应 GPIO	20 pins	24 pins	28(A) pins	28(B) pins	28(C) pins	QFN32 pins
时钟	输出	CLKO	PD5	12 [1]	9	27	8	27	22
		[CLKO]	PA2		14	3	12	3	27
	OSC +	OSC1	PA1		13	2	11	2	26
	OSC −	OSC2	PA2		14	3	12	3	27
ISP 调试	ISP-Data1	ISPDAT1	PC5	10	7		6		20 [4]
	ISP-Data1	ISPDAT1	PC7	12 [1]	11			28	24 [4]
	ISP-CLK1	ISPCLK1	PC6	11	8		7	26	21
	ISP-Data2	ISPDAT2	PB7			19	28	19	13
	ISP-CLK2	ISPCLK2	PB6	7	2	18	27	18	12
外部沿中断		INT	PD4	2		15	22	15	9
		[INT]	PC4		6	24	5		19
		[INT]	PD2		10				23
PORTA 端口变化中断	输入	PA7		17	19	8	17	8	32
		PA6		16	18	7	16	7	31
		PA5		15	17	6	15	6	30
		PA4		14	16	5	14	5	29
		PA3		13	15	4	13	4	28
		PA2			14	3	12	3	27
		PA1			13	2	11	2	26
		PA0			12	1	10	1	25
PWM0		P0A	PA5	15	17	6	15	6	30
	/PWM0	P0AN	PA4	14	16	5	14	5	29
PWM1		P1A0	PA4	14	16	5	14	5	29
		[P1A1]	PB1	6	21	10	26	12	4
		[P1A2]	PC2	9	4	22	3	23	16
	/PWM1	P1A0N	PA3	13	15	4	13	4	28
	/PWM1	[P1A1N]	PC5	10	7	25	6	25	20
	/PWM1	[P1A2N]	PC1	8	3	21	2	22	15
PWM2		P1B0	PA5	15	17	6	15	6	30
		[P1B1]	PB6	7	2	18	27	18	12
PWM3		P1C0	PA6	16	18	7	16	7	31
		[P1C1]	PC6	11	8	26	7	26	21
PWM4		P1D0	PB0	5	20	9	25	11	3
		[P1D1]	PC7	12 [1]	11	28	9	28	24
		[P1D2]	PA7	17	19	8	17	8	32

功能	描述	引脚名	对应 GPIO	20 pins	24 pins	28(A) pins	28(B) pins	28(C) pins	QFN32 pins
PWM 故障刹车输入		BKIN	PD4	2		15	22	15	9
		[BKIN]	PC4		6	24	5		19
		[BKIN]	PD2		10				23
ADC	输入	AN29	PB1	6	21	10	26	12	4
		AN28	PB0	5	20	9	25	11	3
		AN27	PD1					10	2
		AN26	PD0					9	1
		AN25	PA7	17	19	8	17	8	32
		AN24	PA6	16	18	7	16	7	31
		AN23	PA5	15	17	6	15	6	30
		AN22	PA4	14	16	5	14	5	29
		AN21	PA3	13	15	4	13	4	28
		AN20	PA2		14	3	12	3	27
		AN19	PA1		13	2	11	2	26
		AN18	PA0		12	1	10	1	25
		AN17	PC7	12 [1]	11	28	9	28	24
		AN16	PD2		10				23
		AN15	PD5	12 [1]	9	27	8	27	22
		AN14	PC6	11	8	26	7	26	21
		AN13	PC5	10	7	25	6	25	20
		AN12	PC4		6	24	5		19
		AN11	PC3		5	23	4		18
		AN10	PD3						17
		AN9	PC2	9	4	22	3	23	16
		AN8	PC1	8	3	21	2	22	15
		AN7	PC0			20	1	20	14
		AN6	PB7			19	28	19	13
		AN5	PB6	7	2	18	27	18	12
		AN4	PB5	4		17	24	17	11
		AN3	PB4	3		16	23	16	10
		AN2	PD4	2		15	22	15	9
		AN1	PB3	19	23	12	19	14	6
		AN0	PB2	18	22	11	18	13	5
	V_{REF}−	VREFN	PB6	7	2	18	27	18	12
	V_{REF}+	VREFP	PB5	4		17	24	17	11
	ADC_ETR	ADC_ETR	PD4	2		15	22	15	9
		[ADC_ETR]	PC4		6	24	5		19

功能	描述	引脚名	对应 GPIO	20 pins	24 pins	28(A) pins	28(B) pins	28(C) pins	QFN32 pins
ADC	ADC_ETR	[ADC_ETR]	PD2		10				23
运放	正相端输入	OP0+	PB5	4		17	24	17	11
	反相端输入	OP0−	PB6	7	2	18	27	18	12
		[OP0−]	PB0	5	20	9	25	11	3
	输出	OP0OUT	PB7			19	28	19	13
		[OP0OUT]	PB1	6	21	10	26	12	4
比较器 0	正相端输入	CM0+	PB4	3		16	23	16	10
		[CM0+]	PC4		6	24	5		19
		[CM0+]	PD5	12 [1]	9	27	8	27	22
	反相端输入	CM0−	PD4	2		15	22	15	9
		[CM0−]	PC3		5	23	4		18
	输出	CM0OUT	PA7	17	19	8	17	8	32
比较器 1	正相端输入	CM1+	PC0			20	1	20	14
		[CM1+]	PB3	19	23	12	19	14	6
		[CM1+]	PD5	12 [1]	9	27	8	27	22
	反相端输入	CM1−	PB7			19	28	19	13
		[CM1−]	PB2	18	22	11	18	13	5
	输出	CM1OUT	PA6	16	18	7	16	7	31
SPI	SPI_MISO	SPI_MISO	PC7	12	11	28	9	28	24
		[SPI_MISO]	PB4	3		16	23	16	10
	SPI_MOSI	SPI_MOSI	PD5	12 [1]	9	27	8	27	22
		[SPI_MOSI]	PC0			20	1	20	14
	SPI_NSS	SPI_NSS	PC5	10	7	25	6	25	20
		[SPI_NSS]	PC6	11	8	26	7	26	21
	SPI_SCK	SPI_SCK	PC6	11	8	26	7	26	21
		[SPI_SCK]	PD4	2		15	22	15	9
I2C	I2C_SDA	I2C_SDA	PC0			20	1	20	14
		[I2C_SDA]	PC2	9	4	22	3	23	16
		[I2C_SDA]	PB3	19	23	12	19	14	6
	I2C_SCL	I2C_SCL	PB7			19	28	19	13
		[I2C_SCL]	PB2	18	22	11	18	13	5
		[I2C_SCL]	PD4	2		15	22	15	9
		[I2C_SCL]	PC1	8	3	21	2	22	15
USART0	USART0_TX	USART0_TX	PC2	9	4	22	3	23	16
		[USART0_TX]	PB0	5	20	9	25	11	3
		[USART0_TX]	PB3	19	23	12	19	14	6

功能	描述	引脚名	对应 GPIO	20 pins	24 pins	28(A) pins	28(B) pins	28(C) pins	QFN32 pins
USART0	USART0_TX	[USART0_TX]	PC0			20	1	20	14
		[USART0_TX]	PC7	12 [1]	11	28	9	28	24
	USART0_RX	USART0_RX	PC1	8	3	21	2	22	15
		[USART0_RX]	PB2	18	22	11	18	13	5
		[USART0_RX]	PB5	4		17	24	17	11
		[USART0_RX]	PB7			19	28	19	13
		[USART0_RX]	PA0		12	1	10	1	25
USART1	USART1_TX	USART1_TX	PC7	12 [1]	11	28	9	28	24
		[USART1_TX]	PC2	9	4	22	3	23	16
		[USART1_TX]	PC5	10	7	25	6	25	20
	USART1_RX	USART1_RX	PA0		12	1	10	1	25
		[USART1_RX]	PC1	8	3	21	2	22	15
		[USART1_RX]	PC6	11	8	26	7	26	21
TOUCH	输入	KEY1	PA0		12	1	10	1	25
		KEY2	PA1		13	2	11	2	26
		KEY3	PA2		14	3	12	3	27
		KEY4	PA3	13	15	4	13	4	28
		KEY5	PA4	14	16	5	14	5	29
		KEY6	PA5	15	17	6	15	6	30
		KEY7	PA6	16	18	7	16	7	31
		KEY8	PA7	17	19	8	17	8	32
		KEY9	PD0					9	1
		KEY10	PD1					10	2
		KEY11	PB0	5	20	9	25	11	3
		KEY12	PB1	6	21	10	26	12	4
		KEY13	PB2	18	22	11	18	13	5
		KEY14	PB3	19	23	12	19	14	6
		KEY15	PC7	12 [1]	11	28	9	28	24
		KEY16	PC6	11	8	26	7	26	21
		KRY17	PC5	10	7	25	6	25	20
		KEY18	PC4		6	24	5		19
		KEY19	PC3		5	23	4		18
		KEY20	PD3						17
		KEY21	PC2	9	4	22	3	23	16
		KEY22	PC1	8	3	21	2	22	15
		KEY23	PC0			20	1	20	14
		KEY24	PB7			19	28	19	13

功能	描述	引脚名	对应 GPIO	20 pins	24 pins	28(A) pins	28(B) pins	28(C) pins	QFN32 pins
TOUCH	输入	KEY25	PB6	7	2	18	27	18	12
		KEY26	PB5	4		17	24	17	11
		KEY27	PB4	3		16	23	16	10
		KEY28	PD4	2		15	22	15	9
	C_{REF}	TKCAP	PD5	12 [1]	9	27	8	27	22

表 1-1 按功能分类的引脚描述

2. GPIO

FT62F28x 最多支持 30 个 GPIO。这些 I/O 除了作为普通输入/输出端口以外还通常具备一些与内核周边电路通讯的功能，具体见下。

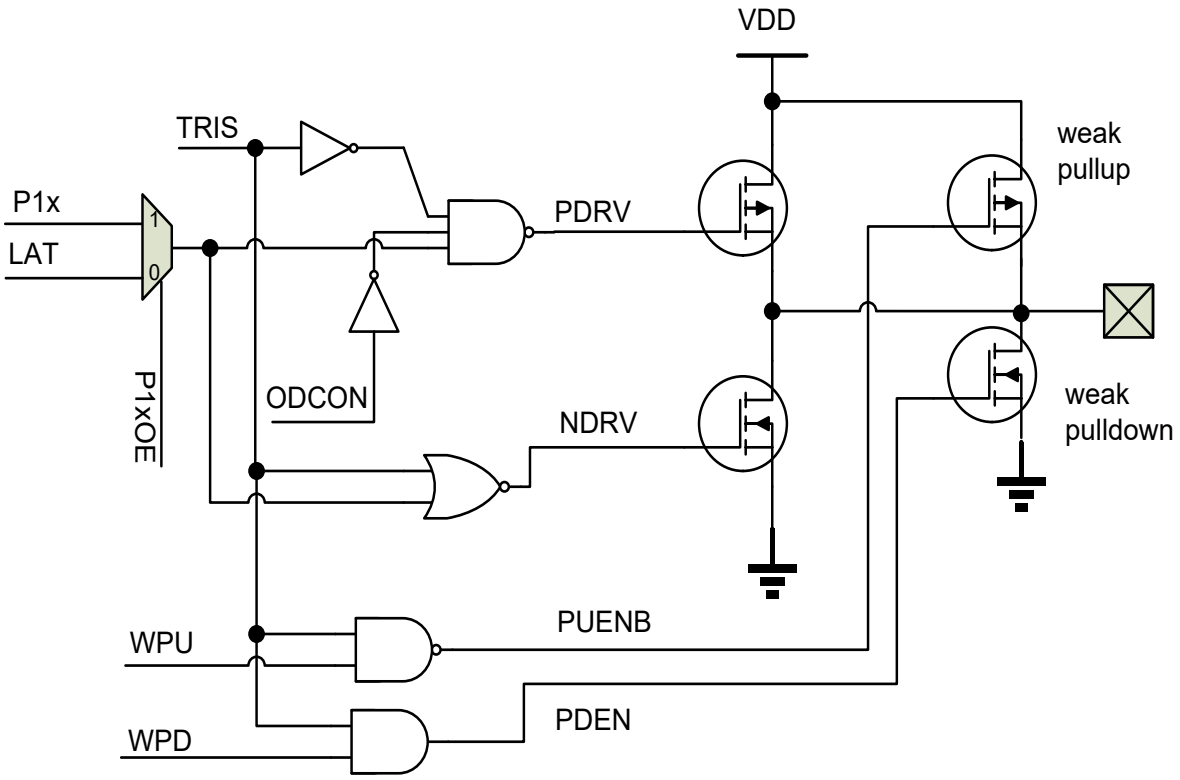


图 2-1 PORT 端口结构原理

所有 I/O 引脚均具有以下功能（**表 2-3**，**表 2-4**）：

- 数字输出
- 数字输入
- 开漏
- 弱上拉
- 弱下拉

此外，部分 I/O 具有以下特殊功能：

1. 烧录调试引脚（ISP-Data，ISP-CLK），硬件内部连接，不需设置。
2. 通过 IDE 界面配置，且在芯片初始化配置时加载的功能（**表 2-6**）：
   - 外部时钟/晶振输入（OSC1, OSC2）
   - 内部时钟输出
   - 系统外部复位（/MCLRB）
3. 通过指令对相应 I/O 引脚进行配置的其他功能，可分为 5 类：
   a. 数字输出
      - PWM
      - 内部时钟输出
      - CM0 结果输出
      - CM1 结果输出
   b. 数字输入
      - PWM 故障刹车
      - Timer1 触发
      - GPIOA 端口变化中断
      - ADC 触发（ADC_ETR）
   c. 模拟输入
      - LVD / BOR
      - ADC
      - V_{REF}+
      - V_{REF}−
      - TOUCH
      - OP0+ / OP0−
      - CM0+ / CM0−
      - CM1+ / CM1−
   d. 模拟输出
      - OP0 输出
   e. 通信接口
      - SPI
      - I2C
      - USART0 / USART1

引脚名	ISP 调试	时钟	中断	LVD	TOUCH	数字 I/O 上拉/下拉	开漏	源电流 (mA)	灌电流 (mA)
PA0			√		KEY1	√	√	3, 6, 24	53, 70
PA1		OSC+	√		KEY2	√	√	3, 6, 24	53, 70
PA2		OSC- / [CLKO]	√		KEY3	√	√	3, 6, 24	53, 70
PA3			√		KEY4	√	√	3, 6, 24	53, 70
PA4			√		KEY5	√	√	3, 6, 24	53, 70
PA5			√		KEY6	√	√	3, 6, 24	53, 70
PA6			√		KEY7	√	√	3, 6, 24	53, 70
PA7			√		KEY8	√	√	3, 6, 24	53, 70
PB0					KEY11	√	√	3, 6, 24	53, 70
PB1					KEY12	√	√	3, 6, 24	53, 70
PB2			/MCLRB		KEY13	√	√	3, 6, 24	53, 70
PB3				ELVD	KEY14	√	√	3, 6, 24	53, 70
PB4					KEY27	√	√	3, 6, 24	53, 70
PB5					KEY26	√	√	3, 6, 24	53, 70
PB6	CLK2				KEY25	√	√	3, 6, 24	53, 70
PB7	DAT2				KEY24	√	√	3, 6, 24	53, 70
PC0					KEY23	√	√	3, 6, 24	53, 70
PC1					KEY22	√	√	3, 6, 24	53, 70
PC2					KEY21	√	√	3, 6, 24	53, 70
PC3					KEY19	√	√	3, 6, 24	53, 70
PC4			[INT]		KEY18	√	√	3, 6, 24	53, 70
PC5	DAT1	T0CK1			KEY17	√	√	3, 6, 24	53, 70
PC6	CLK1				KEY16	√	√	3, 6, 24	53, 70
PC7	DAT1				KEY15	√	√	3, 6, 24	53, 70
PD0					KEY9	√	√	3, 6, 24	53, 70
PD1					KEY10	√	√	3, 6, 24	53, 70
PD2			[INT]			√	√	3, 6, 24	53, 70
PD3					KEY20	√	√	3, 6, 24	53, 70
PD4			INT		KEY28	√	√	3, 6, 24	53, 70
PD5		CLKO			TKCAP	√	√	3, 6, 24	53, 70
注		T1CK1 = PA2						V_{DD}=5, V_{DS}=0.5	

表 2-1 I/O 端口功能 (1)

注： 所有 IO 支持 3 档可配置源电流驱动能力 (参阅 “PSRCx”，**表 2-4**)，和 2 档可配置灌电流驱动能力 (参阅 “PSINKx”，**表 2-4**)。

引脚名	ADC	比较器	运算放大器	SPI	I2C	USART0	USART1	PWM
PA0	AN18					[RX]	RX	
PA1	AN19							
PA2	AN20							
PA3	AN21							P1A0N
PA4	AN22							P0AN / P1AN
PA5	AN23							P0A / P1B0
PA6	AN24	CM1OUT						P1C0
PA7	AN25	CM0OUT						[P1D2]
PB0	AN28		[OP0-]			[TX]		P1D0
PB1	AN29		[OP0OUT]					[P1A1]
PB2	AN0	[CM1-]			[SCL]	[RX]		
PB3	AN1	[CM1+]			[SDA]	[TX]		
PB4	AN3	CM0+		[MISO]				
PB5	AN4 / (V_{REF}+)		OP0+			[RX]		
PB6	AN5 / (V_{REF}-)		OP0-					[P1B1]
PB7	AN6	CM1-	OP0OUT		SCL	[RX]		
PC0	AN7	CM1+		[MOSI]	SDA	[TX]		
PC1	AN8				[SCL]	RX	[RX]	[P1A2N]
PC2	AN9				[SDA]	TX	[TX]	[P1A2]
PC3	AN11	[CM0-]						
PC4	AN12	[CM0+]						[BKIN]
PC5	AN13			NSS [NSS]			[TX]	[P1A1N]
PC6	AN14			SCK			[RX]	[P1C1]
PC7	AN17			MISO		[TX]	TX	[P1D1]
PD0	AN26							
PD1	AN27							
PD2	AN16							[BKIN]
PD3	AN10							
PD4	AN2	CM0-		[SCK]	[SCL]			BKIN
PD5	AN15	[CM0+]/[CM1+]		MOSI				
注：								

表 2-2 I/O 端口功能 (2)

2.1. IO 端口相关寄存器汇总

名称	地址	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	复位值
PORTA	0x05	PORTA 输出寄存器								xxxx xxxx
PORTB	0x06	PORTB 输出寄存器								xxxx xxxx
PORTC	0x07	PORTC 输出寄存器								xxxx xxxx
PORTD	0x08	—	—	PORTD 输出寄存器						--xx xxxx
TRISA	0x85	PORTA 方向控制								1111 1111
TRISB	0x86	PORTB 方向控制								1111 1111
TRISC	0x87	PORTC 方向控制								1111 1111
TRISD	0x88	—	—	PORTD 方向控制						--11 1111
WPUA	0x294	PORTA 弱上拉								1111 1111
WPUB	0x295	PORTB 弱上拉								0000 0000
WPUC	0x296	PORTC 弱上拉								0000 0000
WPUD	0x297	—	—	PORTD 弱上拉						--00 0000
WPDA	0x290	PORTA 弱下拉								0000 0000
WPDB	0x291	PORTB 弱下拉								0000 0000
WPDC	0x292	PORTC 弱下拉								0000 0000
WPDD	0x293	—	—	PORTD 弱下拉						--00 0000
ODCONA	0x205	PORTA 开漏								0000 0000
ODCONB	0x206	PORTB 开漏								0000 0000
ODCONC	0x207	PORTC 开漏								0000 0000
ODCOND	0x208	—	—	PORTD 开漏						--00 0000
PSINKA	0x214	PORTA 灌电流设置								0000 0000
PSINKB	0x215	PORTB 灌电流设置								0000 0000
PSINKC	0x216	PORTC 灌电流设置								0000 0000
PSINKD	0x217	—	—	PORTD 灌电流设置						--00 0000
PSRCAL	0x20C	PORTA 源电流设置（低 8 位）								1111 1111
PSRCAH	0x20D	PORTA 源电流设置（高 8 位）								1111 1111
PSRCBL	0x20E	PORTB 源电流设置（低 8 位）								1111 1111
PSRCBH	0x20F	PORTB 源电流设置（高 8 位）								1111 1111
PSRCCL	0x210	PORTC 源电流设置（低 8 位）								1111 1111
PSRCCH	0x211	PORTC 源电流设置（高 8 位）								1111 1111
PSRCDL	0x212	PORTD 源电流设置（低 8 位）								1111 1111
PSRCDH	0x213	—	—	—	—	PORTD 源电流设置（高 4 位）				---- 1111
ANSEL0	0x11E	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	0000 0000
ANSEL1	0x11F	ANS15	ANS14	ANS13	ANS12	ANS11	ANS10	ANS9	ANS8	0000 0000
ANSEL2	0x28C	ANS23	ANS22	ANS21	ANS20	ANS19	ANS18	ANS17	ANS16	0000 0000
ANSEL3	0x28D	—	—	ANS29	ANS28	ANS27	ANS26	ANS25	ANS24	--00 0000
COMAF0	0x285	SCKPO	NSSPO	MOSIPO	MISOPO	SDAPO[1:0]		SCLPO[1:0]		0000 0000
COMAF1	0x286	INTPO[1:0]		TX0PO[2:0]			RX0PO[2:0]			0000 0000
COMAF2	0x287	UR1SW	UR0SW	—	TX1PO[1:0]		INTFIXB	RX1PO[1:0]		00-0 0100
IOCA	0x98	IOCA[7:0]: PORTA 端口变化中断设置								0000 0000
OPTION	0x81	/PAPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111

表 2-3 I/O 相关用户寄存器的地址和复位值

名称	状态		寄存器	地址	复位值
TRISA	PORTA	PORT 端口数字输出 (方向控制) 1 = 关闭 0 = 使能 (关闭上拉/下拉)	TRISA[7:0]	0x85	RW-1111 1111
TRISB	PORTB		TRISB[7:0]	0x86	RW-1111 1111
TRISC	PORTC		TRISC[7:0]	0x87	RW-1111 1111
TRISD	PORTD		TRISD[5:0]	0x88	RW-11 1111
ANSEL0	AN[7:0]	1 = 关闭上拉/下拉，及数字输入 (适用于 30 个 ADC 通道) 0 = (无动作)	ANSEL0[7:0]	0x11E	RW-0000 0000
ANSEL1	AN[15:8]		ANSEL1[7:0]	0x11F	RW-0000 0000
ANSEL2	AN[23:16]		ANSEL2[7:0]	0x28C	RW-0000 0000
ANSEL3	AN[29:24]		ANSEL3[5:0]	0x28D	RW-00 0000
/PAPU	1 = 关闭所有 PORTA 上拉功能 0 = 上拉由 WPUA 控制		OPTION[7]	0x81	RW-1
WPUA	PORTA	弱上拉 1 = 使能 (PORTA 默认值) 0 = 关闭 (PORTB, C, D 默认值)	WPUA[7:0]	0x294	RW-1111 1111
WPUB	PORTB		WPUB[7:0]	0x295	RW-0000 0000
WPUC	PORTC		WPUC[7:0]	0x296	RW-0000 0000
WPUD	PORTD		WPUD[5:0]	0x297	RW-00 0000
WPDA	PORTA	弱下拉 1 = 使能 0 = 关闭	WPDA[7:0]	0x290	RW-0000 0000
WPDB	PORTB		WPDB[7:0]	0x291	RW-0000 0000
WPDC	PORTC		WPDC[7:0]	0x292	RW-0000 0000
WPDD	PORTD		WPDD[5:0]	0x293	RW-00 0000
ODCONA	PORTA	开漏 1 = 使能 0 = 关闭	ODCONA[7:0]	0x205	RW-0000 0000
ODCONB	PORTB		ODCONB[7:0]	0x206	RW-0000 0000
ODCONC	PORTC		ODCONC[7:0]	0x207	RW-0000 0000
ODCOND	PORTD		ODCOND[5:0]	0x208	RW-00 0000
PORTA	PORTA	数据输出寄存器	PORTA[7:0]	0x05	RW-xxxx xxxx
PORTB	PORTB		PORTB[7:0]	0x06	RW-xxxx xxxx
PORTC	PORTC		PORTC[7:0]	0x07	RW-xxxx xxxx
PORTD	PORTD		PORTD[5:0]	0x08	RW-xx xxxx
PSRCA [1]	PORTA	源电流 (mA) (00) = 3 (01) = 6 / (10) = 6 (11) = 24	PSRCAL[7:0]	0x20C	RW-1111 1111
			PSRCAH[7:0]	0x20D	RW-1111 1111
PSRCB [1]	PORTB		PSRCBL[7:0]	0x20E	RW-1111 1111
			PSRCBH[7:0]	0x20F	RW-1111 1111
PSRCC [1]	PORTC		PSRCCL[7:0]	0x210	RW-1111 1111
			PSRCCH[7:0]	0x211	RW-1111 1111
PSRCD [1]	PORTD		PSRCDL[7:0]	0x212	RW-1111 1111
			PSRCDH[3:0]	0x213	RW-1111

[1] 每个 I/O 由 2 个 bit 控制，即 PSRCx[(2y+1) : 2y] 控制管脚 PORTx.y (其中 x = A/B/C/D, y = 0~7)。如: PSRCA[1:0] 控制 PA0。

名称	状态		寄存器	地址	复位值
PSINKA	PORTA	灌电流（mA) 1 = 70 0 = 53	PSINKA[7:0]	0x214	RW-0000 0000
PSINKB	PORTB		PSINKB[7:0]	0x215	RW-0000 0000
PSINKC	PORTC		PSINKC[7:0]	0x216	RW-0000 0000
PSINKD	PORTD		PSINKD[5:0]	0x217	RW-00 0000
SCKPO	SPI_SCK 管脚映射 1 = PD4 0 = PC6		COMAF0[7]	0x285	RW-0
NSSPO	SPI_NSS 管脚映射 1 = PC6 0 = PC5		COMAF0[6]		RW-0
MOSIPO	SPI_MOSI 管脚映射 1 = PC0 0 = PD5		COMAF0[5]		RW-0
MISOPO	SPI_MISO 管脚映射 1 = PB4 0 = PC7		COMAF0[4]		RW-0
SDAPO	I2C_SDA 管脚映射 00 = PC0 10 = PC2 01 = PB3 11 = PB4		COMAF0[3:2]		RW-00
SCLPO	I2C_SCL 管脚映射 00 = PB7 10 = PC1 01 = PB2 11 = PD4		COMAF0[1:0]		RW-00
INTPO	INT/BKIN/ADC_ETR 管脚映射 0x = PD4 10 = PD2 11 = PC4		COMAF1[7:6]	0x286	RW-00
TX0PO	USART0_TX 管脚映射 当 UR0SW = 0 时 当 UR0SW = 1 时 0xx = PC2 0xx = PC1 100 = PC0 100 = PB7 101 = PB3 101 = PB2 110 = PB0 110 = PB5 111 = PC7 111 = PA0		COMAF1[5:3]		RW-000
RX0PO	USART0_RX 管脚映射 当 UR0SW = 0 时 当 UR0SW = 1 时 0xx = PC1 0xx = PC2 100 = PB7 100 = PC0 101 = PB2 101 = PB3 110 = PB5 110 = PB0 111 = PA0 111 = PC7		COMAF1[2:0]		RW-000

名称	状态		寄存器	地址	复位值
UR1SW	USART1 TX 和 RX 管脚交换 1 = 交换 (详见 TX1PO 和 RX1PO) 0 = 正常		COMAF2[7]	0x287	RW-0
UR0SW	USART0 TX 和 RX 管脚交换 1 = 交换 (详见 TX0PO 和 RX0PO) 0 = 正常		COMAF2[6]		RW-0
TX1PO	USART1_TX 管脚映射 当 UR1SW = 0 时： 0x = PC7 10 = PC5 11 = PC2 当 UR1SW = 1 时： 0x = PA0 10 = PC6 11 = PC1		COMAF2[4:3]		RW-00
INTFIXB	INT 中断管脚映射 (≥ G 版芯片) 1 = 由 INTPO 决定 0 = 映射在 PD4 (BKIN 和 ADC_ETR 映射仍由 INTPO 决定) 注：<G 版芯片，此位为保留位，读为 0		COMAF2[2]		RW-1
RX1PO	USART1_RX 管脚映射 当 UR1SW=0 时： 0x = PA0 10 = PC6 11 = PC1 当 UR1SW=1 时： 0x = PC7 10 = PC5 11 = PC2		COMAF2[1:0]		RW-00
CLKOS	CLKO 信号映射管脚	1 = PD5 0 = PA2	MSCON0[4]	0x10C	RW-1
OP0PSEL	运放 0 正相端输入	1 = GND 0 = PB5	OP0CR0[6]	0x96	RW-1
OP0NSEL	运放 0 反相端输入 00 = GND 01 = PB6 或 PB0 (由 IPINSW 决定) 10 = 串联 4k 电阻连接到 PB6 或 PB0 (由 IPINSW 决定) 11 = 串联 4k 电阻连接到 GND		OP0CR0[5:4]		RW-10
OPINSW	运放 0 输出管脚	1 = PB1 0 = PB7	OP0CR1[5]	0x97	RW-0
IPINSW	运放 0 反相端外部输入管脚	1 = PB0 0 = PB6	OP0CR1[4]		RW-0

名称	状态		寄存器	地址	复位值
OPTOIO	运放 0 输出	1 = 使能 0 = 禁止	OP0CR1[0]		RW-0
CM0OE	比较器 0 输出到 PA7	1 = 使能 0 = 禁止	CM0CON0[5]	0x29D	RW-0
CM0PSEL	比较器 0 正相端输入	(00) / (11) = PB4 (01) = PC4 (10) = PD5	CM0CON0[3:2]		RW-00
CM0NSEL	比较器 0 反相端输入	00 = PD4 01 = PC3 10 = 运放 0 输出 11 = DAC1OUT	CM0CON0[1:0]		RW-00
CM1OE	比较器 1 输出到 PA6	1 = 使能 0 = 禁止	CM1CON0[5]	0x29E	RW-0
CM1PSEL	比较器 1 正相端输入	(00) / (11) = PB3 (01) = PC0 (10) = PD5	CM1CON0[3:2]		RW-00
CM1NSEL	比较器 1 反相端输入	00 = PB2 01 = PB7 10 = 运放 0 输出 11 = DAC2OUT	CM1CON0[1:0]		RW-00
P1C0OE	P1C0 输出到 PA6	1 = 使能 0 = 禁止	P1OE0[7]	0x9E	RW-0
P1B0OE	P1B0 输出到 PA5		P1OE0[6]		RW-0
P1A2NOE	P1A2N 输出到 PC1		P1OE0[5]		RW-0
P1A2OE	P1A2 输出到 PC2		P1OE0[4]		RW-0
P1A1NOE	P1A1N 输出到 PC5		P1OE0[3]		RW-0
P1A1E	P1A1 输出到 PB1		P1OE0[2]		RW-0
P1A0NOE	P1A0N 输出到 PA3		P1OE0[1]		RW-0
P1A0OE	P1A0 输出到 PA4		P1OE0[0]		RW-0
P1D2OE	P1D2 输出到 PA7		P1OE1[7]	0x9F	RW-0
P1D1OE	P1D1 输出到 PC7		P1OE1[6]		RW-0
P1D0OE	P1D0 输出到 PB0		P1OE1[5]		RW-0
P1C1OE	P1C1 输出到 PC6		P1OE1[2]		RW-0
P1B1OE	P1B1 输出到 PB6		P1OE1[1]		RW-0

表 2-4 I/O 相关用户寄存器

名称	状态		寄存器	地址	复位值
GIE	全局中断	1 = 使能 (PEIE, INTE, PAIE 适用) 0 = 全局关闭 (唤醒不受影响)	INTCON[7]	Bank 首地址 +0x0B	RW-0
PEIE	外设总中断	1 = 使能 (PAIE 适用) 0 = 关闭 (无唤醒)	INTCON[6]		RW-0
INTE	外部中断	1 = 使能 0 = 关闭 (无唤醒)	INTCON[4]		RW-0
PAIE	PORTA 端口变化总中断	1 = 使能 0 = 关闭	INTCON[3]		RW-0
INTF	外部中断标志位	1 = Yes (锁存) 0 = No	INTCON[1]		RW0-0
PAIF	PORTA 端口变化中断标志位		INTCON[0]		RW0-0
IOCA	PORTA 各端口变化中断	1 = 使能 0 = 禁止	IOCA[7:0]	0x98	RW-0000 0000
INTEDG	INT 管脚中断沿选择	1 = 上升沿 0 = 下降沿	OPTION[6]	0x14	RW-1

表 2-5 PORTA 和外部中断相关寄存器

名称	功能	默认
MCLRE	外部 I/O 复位	关闭
FOSC	• LP：PA1 (+) 和 PA2 (−) 接外部低速晶振 • XT：PA1 (+) 和 PA2 (−) 接外部高速晶振 • EC：PA1 (+) 接外部时钟输入，PA2 为 I/O • INTOSC：PA2 (或 PD5) 输出系统时钟的 2 分频，PA1 为 IO 管脚 • INTOSCIO：PA1 和 PA2 为 I/O	INTOSCIO

表 2-6 I/O 相关初始化配置寄存器

2.2. I/O 配置

每个 PORT 端口，均需根据其相应功能配置以下 4 个模块（**表 2-4**）：

- 数字输出
- 数字输入
- 开漏
- 弱上拉
- 弱下拉

功能	数字输入	上拉/下拉	数字输出	设置
ISP-DATA	On	Off	On	(硬件内置，忽略指令)
ISP-CLK	On	Off	Off	(硬件内置，忽略指令)
/MCLRB	On	上拉	Off	(初始化配置，忽略指令)
时钟输出	(忽略)	Off	On	(初始化配置，忽略指令)
OSC+ (EC)	On	(可选)	Off	(初始化配置，忽略指令)
OSC+ / OSC− (LP, XT)	Off	Off	Off	(初始化配置，忽略指令)
ADC	Off	Off	Off	TRISx = 1; ANSELx = 1
OP0OUT	Off	Off	Off	OP0ON = 1; OPTOIO = 1
CM0 / CM1 输出	On	(可选)	On	TRISx = 0
TOUCH	Off	Off	Off	TRISx = 1
SPI 输出	On	Off	On	TRISx = 0
I2C 输出	On	Off	On	TRISx = 0
USART 输出	On	Off	On	TRISx = 0
LVD	Off (5)	Off	Off	TRISx = 1; ANSELx = 1
V_{REF}+ / V_{REF}−	Off	Off	Off	TRISx = 1
ADC 触发	On	(可选)	Off	TRISx = 1
OP0+ / OP0−输入	Off	Off	Off	TRISx = 1; ANSELx = 1
CMx+ / CMx−输入	Off	Off	Off	TRISx = 1; ANSELx = 1
SPI 输入	On	(可选)	Off	TRISx = 1
I2C 输入	On	(可选)	Off	TRISx = 1
USART 输入	On	(可选)	Off	TRISx = 1
端口变化中断	On	(可选)	Off	TRISx = 1
BKIN	On	(可选)	Off	TRISx = 1
数字输入	On	(可选)	Off	TRISx = 1
PWM	On	Off	On	TRISx = 0
数字输出	On	Off	On	TRISx = 0

表 2-5 I/O 配置标志和用户寄存器

注：

1. TRISx = 0：“数字输出”使能，“上拉/下拉”自动关闭（忽略 WPDx, WPUx）。
2. TRISx = 1：“数字输出”关闭。
3. ANSELx = 1：“上拉”、“下拉”、“数字输入”自动关闭（忽略 WPDx, WPUx）。
4. 可关闭“数字输入”的唯一指令为“ANSELx = 1”。

5. 将 PORT 端口设置为 LVD 输入时，其“数字输入”、“上拉”和“下拉”功能被自动关闭。
6. “/PAPU = 1”关闭所有 PAx 端口的“弱上拉”功能。PBx、PCx 和 PDx 没有此类控制位。
7. /MCLR 使能：PB2 的弱上拉功能自动使能 (忽略 WPUB[2])；读 PORTB[2] 的值为“0”。
8. 对 PORTx 数据输出寄存器进行写操作，I/O 端口将输出相应的逻辑电平。每组多达 8 个 I/O 的数据寄存器共用相同的地址，写操作实际执行‘读-修改-写’的过程，即先读取该组 PORTx 端口锁存器值 (输出或输入)，然后修改，再写回 PORTx 数据寄存器。
9. 数字输出和数字输入功能可以共存，有些应用需要同时使能数字输出和数字输入。
10. 当 TRISx = 0 时，通过 IDE 界面可选择读取 PORTx 输出或输入锁存器的值。
11. ODCONx = 1：管脚开漏输出。管脚的开漏功能和内部上拉功能可以同时打开
12. 完全复位或系统复位时，PORTx 寄存器不会复位，但 TRISx 将被重置为“1”，从而关闭输出。

INT 和 PORTA 端口变化中断的设置，请参阅 **章节 14** “中断”。

2.3. PORTx 功能及优先级

每个 I/O 管脚均复用了多个功能，当某管脚复用的功能模块都使能输出的情况下，就存在优先级的问题。

因为输入是连到各个功能模块的，故输入不存在优先级问题，例如 PC1 作为 GPIO 输入功能时，同时也可以作为 USART 的数据输入 (USART0_RX)。

注意：调试串口优先级最高，其次是代码选项的配置，再次是模拟输出。例如 PB7 管脚，在其上面复用了调试数据 ISPDAT2，运放输出 (模拟) OP0OUT 以及其它数字功能，其优先级如下表:

管脚名称	功能优先级 0	功能优先级 1	功能优先级 2	功能优先级 3	功能优先级 4
PA0	PA0	–	–	–	–
PA1	PA1	OSC1	–	–	–
PA2	PA2	CLKO	OSC2	–	–
PA3	PA3	P1A0N	–	–	–
PA4	PA4	P0AN	P1A0	–	–
PA5	PA5	P0A	P1B0	–	–
PA6	PA6	P1C0	CM1OUT	–	–
PA7	PA7	[P1D2]	CM0OUT	–	–
PB0	PB0	P1D0	–	–	–
PB1	PB1	[P1A1]	[OP0OUT] [1]	–	–
PB2	PB2	[I2C_SCL]	–	–	–
PB3	PB3	[USART0_TX]	[I2C_SDA]	–	–
PB4	PB4	[SPI_MISO]	–	–	–
PB5	PB5	–	–	–	–
PB6	PB6	[P1B1]	ISPCK2	–	–
PB7	PB7	I2C_SCL	OP0OUT [2]	ISPDAT2	–
PC0	PC0	I2C_SDA	[SPI_MOSI]	[USART0_TX]	–
PC1	PC1	[P1A2]	[I2C_SDA]	USART0_TX	–
PC2	PC2	[P1A2N]	[I2C_SCL]	[USART1_TX]	–
PC3	PC3	–	–	–	–
PC4	PC4	–	–	–	–
PC5	PC5	SPI_NSS	[P1A1N]	[USART1_TX]	ISPDAT1
PC6	PC6	SPI_SCK	[SPI_NSS]	[P1C1]	ISPCK1
PC7	PC7	SPI_MISO	[P1D1]	USART1_TX	ISPDAT1
PD0	PD0	–	–	–	–
PD1	PD1	–	–	–	–
PD2	PD2	–	–	–	–
PD3	PD3	–	–	–	–
PD4	PD4	[SPI_SCK]	–	–	–
PD5	PD5	SPI_MOSI	[CLKO]	–	–

表 2-7 IO 功能优先级

[1] 当 OPTOIO=1 且 OPINSW=1 时
[2] 当 OPTOIO=1 且 OPINSW=0 时

3. 上电复位

片上的 POR 电路会将芯片保持在复位状态直到 VDD 电源电压达到足够高，上电复位释放后，系统复位不会立即释放，还要等一个约 8ms 的延时，期间数字电路保持在复位状态。

3.1. 上电复位流程

在上电过程中，芯片内部的超时动作顺序按以下流程执行：

a) POR 结束后，启动 8ms 计时

b) 启动芯片配置过程 (BOOT)

c) 启动 PWRT 计时 (如果使能)

3.2. 上电校验过程

如果 UCFG1.6(CSUMENB)为 0，则上电配置过程结束后，CPU 不会立即执行程序，而是会启动 PROM 的内部校验过程。硬件会把 PROM 主程序区内容读出来自加，8k 字全部做完累加后和存储在 0x2007 的值相比较，如果相等说明校验成功，CPU 会从 0 地址执行程序，如图 3-4 所示；如果不相等说明校验失败，CPU 将不会执行程序。校验过程大概需要耗时 8ms。

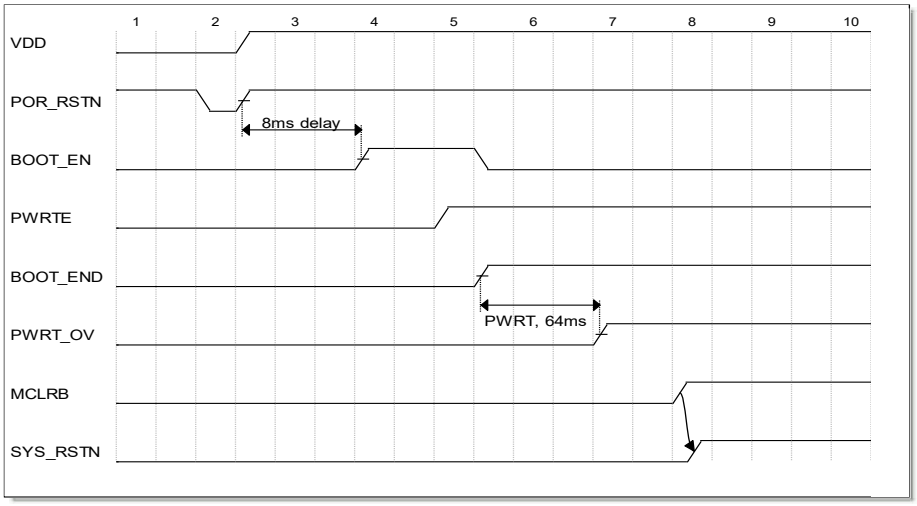


图 3-1 上电复位，使用了 MCLRB

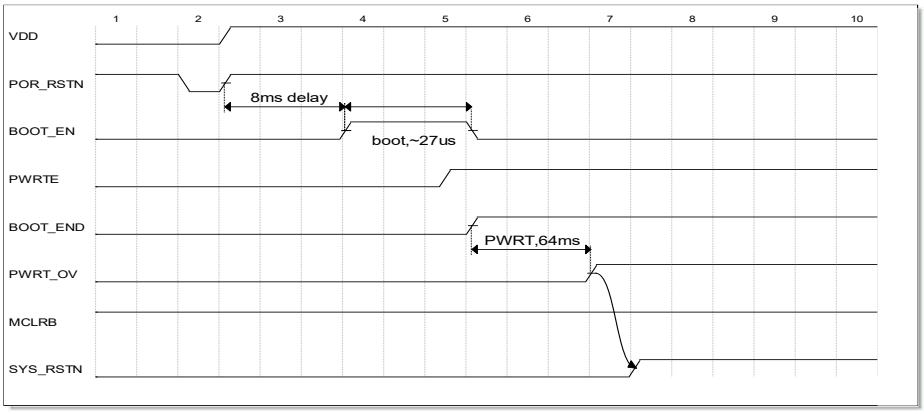


图 3-2 上电复位，没使用 MCLRB (< G 版芯片)

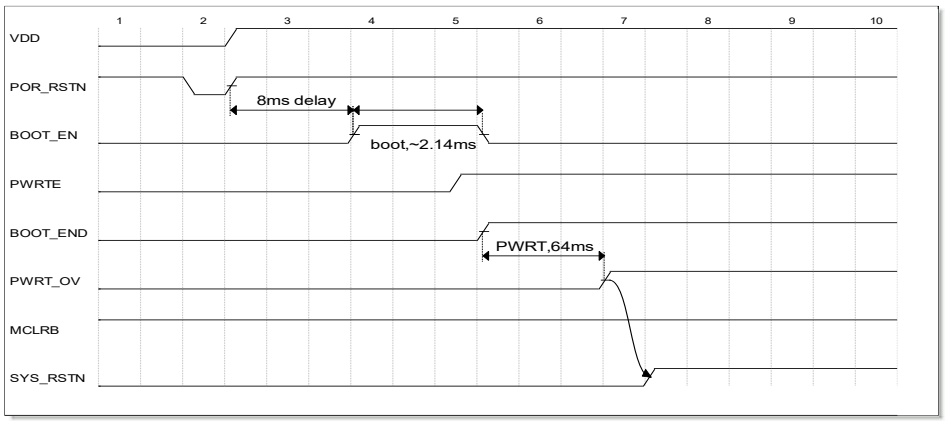


图 3-3 上电复位，没使用 MCLRB (≥ G 版本芯片)

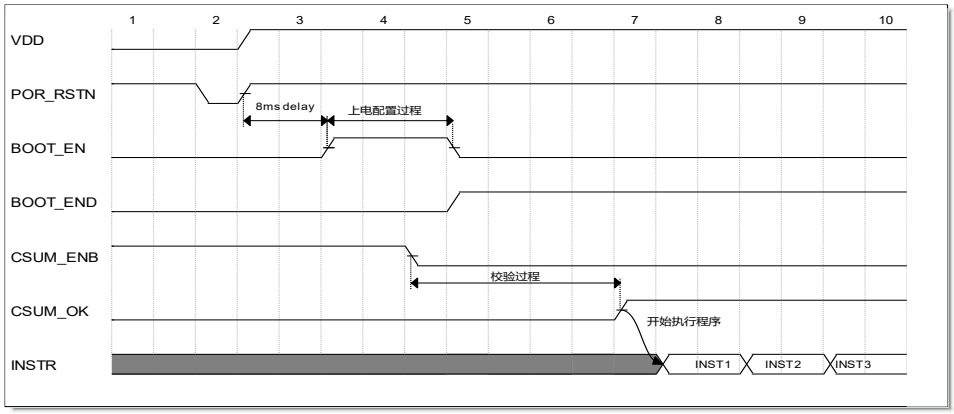


图 3-4 校验过程

3.3. 上电复位延时

PWRT 为上电复位，低电压复位提供一个固定的 64ms (正常情况下) 的定时。这个定时器由内部慢时钟驱动。芯片在定时器超时之前都是被保持在复位状态。这段时间能保证 VDD 上升到足够高的电压使得系统能正常工作。PWRT 也可以通过系统 CONFIG 寄存器 (UCFG0) 来使能。在开启低电压复位功能时，用户应该也打开 PWRT。PWRT 定时是由 VDD 电压超过 VBOR 门限事件启动的。另外需要注意的是，由于由内部慢时钟驱动，定时的实际时间长度是随温度，电压等条件变化而变化的。这个时间不是一个精准参数。

4. 系统复位

有以下复位源：

- 上电复位
- 低电压复位
- 看门狗复位
- 非法指令复位
- 外部管脚复位

有些寄存器是不被任何复位影响的；这些寄存器的状态在上电复位时是未知的，也不受复位事件影响。大多数其它寄存器都会在复位事件时恢复到其“复位状态”，见寄存器 SFR 表格。

WDT (看门狗) 睡眠唤醒不会造成和在常规运行状态下 WDT (看门狗) 超时所造成的复位。因为睡眠唤醒本身就是一种继续的意思，而不是复位。/TF 和/PF 位的置位和清零在不同复位条件下的动作是不同的。具体可参考**表 4-3** 和**表 4-4**。

MCLRB 管脚的电路带有防抖功能，能够滤除一些干扰造成的尖细脉冲信号。

下图为复位电路的总体概述框图。

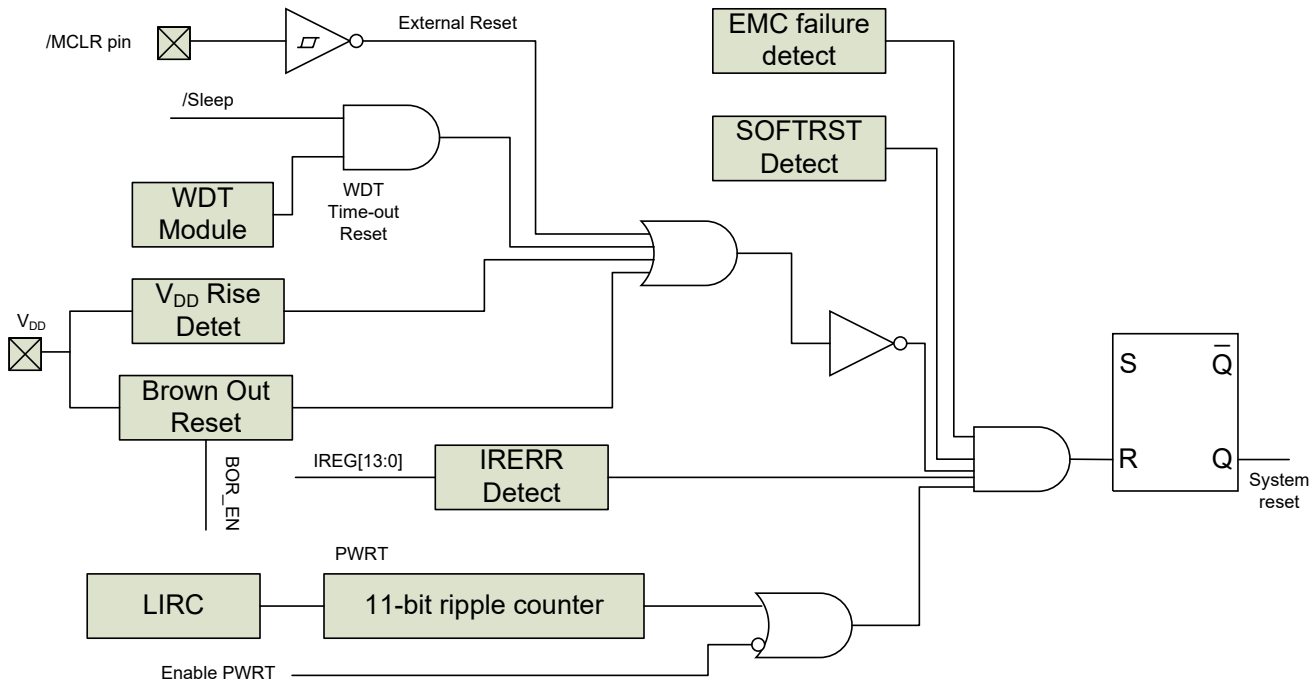


图 4-1 复位框图

4.1. 系统复位相关寄存器汇总

系统复位的大多数设置均由 IDE 界面配置，而不能通过指令修改。

名称	功能	默认
LVRS	7 档 V_{BOR} 电压(V)： 2.0 / 2.2 / 2.5 / 2.8 / 3.1 / 3.6 / 4.1	2.0
LVREN	LVR • 使能 • 关闭 • 非 SLEEP 模式下使能 • 通过指令控制 (SLVREN)	关闭
WDTE	WDT • 使能 (指令不能禁止) • 由指令控制 (SWDTEN)	SWDTEN 控制
MCLRE	外部 I/O 复位	关闭
IRBTE	错误指令 IRERR 复位 BOOT 使能位	关闭
MRBTE	MCLR 复位 BOOT 使能位	关闭
WDTBTE	WDT 复位 BOOT 使能位	关闭

表 4-1 复位相关初始化配置寄存器

4.2. 低电压复位

低电压复位由 UCFG1[1:0]位和 MSCON0 的 SLVREN 位来控制。低电压复位是指当电源电压低于 V_{BOR} 门限电压时所产生的复位。不过当 VDD 电压低于 V_{BOR} 不超过 T_{BOR} 时间时，低电压复位就不会发生。

如果 BOR (低电压复位) 是使能 (UCFG1[1:0]=00) 的，那么最大 VDD 电压上升时间的要求就不存在。BOR 电路会将芯片控制在复位状态，一直到 VDD 电压达到 V_{BOR} 门限电压以上。需要注意的是，当 VDD 低于系统能正常工作的门限时，POR 电路并不会产生复位信号。

当 UCFG1[1:0]=10 时，BOR 电路关闭将由 CPU 的运行状态决定：CPU 正常工作时 BOR 电路工作，CPU 处于睡眠模式时 BOR 电路关闭，这样可以方便的使系统功耗降至更低水平。

BOR 默认情况下是带去抖功能的，以过滤 VDD 可以出现的毛刺。该去抖功能可以通过 MSCON1 的 LVRDEB 位关闭。

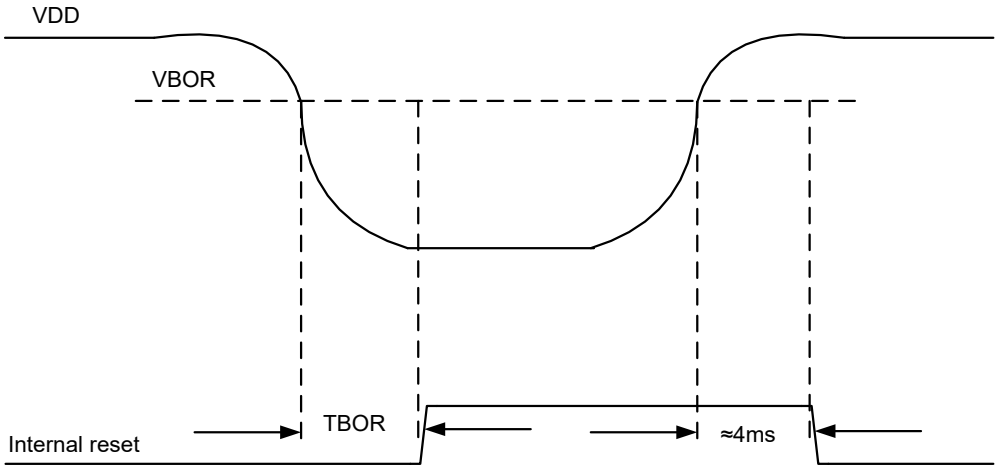


图 4-2 欠压复位

注意：

1. 上电复位或低电压复位后，并且在 PWRTEB (UCFG0.4) 为低时，PWRT 有效。它是 2048 个内部慢时钟周期，约 64ms；
2. T_{BOR} 时间约为 122~152μs；
3. 电压恢复正常之后，内部复位不会立即释放，而是要等约为 8ms 的时间。

名称	状态		寄存器	地址	复位值
SLVREN [1]	仅适用于 LVREN 配置成由指令 SLVREN 控制 LVR 1 = 使能 LVR 0 = 关闭 LVR		MSCON0[3]	0x10C	RW-0
LVRDEB [2]	LVR 消抖	1 = 使能 0 = 禁止	MSCON1[3]	0x101 / 0x301	RW-1
LVRCKS [2]	LVR 消抖时钟	1 = HIRC 0 = LIRC	MSCON1[2]		RW-0
LVRADJ [2]	LVR 修调位，~3% / step		LVDTRIM[2:0]	0x21C	RO-011

表 4-2 LVR 用户寄存器

4.3. 非法指令复位

当 CPU 指令寄存器取指到非法指令 (未定义的操作码) 时，系统将进行复位，可增加系统的抗干扰能力。

4.4. 上电配置过程(BOOT)

发生上电复位或者低电压复位后，除了固有的 8ms 复位延时外，还有一个初始化配置寄存器 UCFGx 的动作 (BOOT)。该动作从 PROM 的保留地址读取内容写到 UCFGx，待所有配置地址读取完成后，才可

[1] 发生欠压复位时，此位不会清 0。其它复位会将此位清 0。
[2] 只受上电复位影响，其它复位不起作用

以释放系统复位，如图 3-2 和图 3-3 的所示，≥ G 版芯片该过程大概需要 2.14ms，< G 版芯片需要 27us。

可触发 BOOT 过程的复位源汇总

复位源	可触发 Boot
上电复位	√
低电压复位	√
看门狗复位	√
非法指令复位	√
外部管脚复位	√

4.5. 复位源标志位

名称	状态		寄存器	地址	复位值
/POR	上电复位标志位	1 = No (软件置 1) 0 = Yes (锁存)	PCON[1]	0x8E	RW1−0
/BOR	低电压复位标志位		PCON[0]		RW1−x
/TF	超时状态位 1 = 上电后，执行了 CLRWDT 指令或 SLEEP 指令 0 = 发生 WDT 超时溢出		STATUS[4]	Bank 首地址 + 0x03	RO-1
/PF	掉电标志位 1 = 上电复位后或执行了 CLRWDT 指令 0 = 执行了 SLEEP 指令		STATUS[3]		RO-1

表 4-3 复位标志寄存器

复位源	/POR	/BOR	/TF	/PF
	PCON[1]	PCON[0]	STATUS[4]	STATUS[3]
	0x8E		Bank 首地址 + 0x03	
POR	0	x	1	1
LVR	–	0	1	1
正常模式下(非 SLEEP) WDT 溢出(复位)	–	–	0	–
SLEEP 模式下 WDT 溢出(唤醒)	–	–	0	0
SLEEP 模式下/MCLR 复位	–	–	1	0
正常模式下(非 SLEEP) /MCLR 复位	–	–	–	–
非法指令复位	–	–	–	–
片上调试 (OCD)	–	–	–	–

表 4-4 复位相关状态标志位（“x”未知；“–“ 无变化)

5. LVD 低电压侦测

除低电压复位功能外，芯片还内置低电压侦测功能。当电源电压低于设置的电压档位（由 PCON 的 LVDL[3:0]选择）超过 T_{LVD} (3 到 4 个慢时钟周期) 以上时，标志位 LVDW 将被置 1，软件可用此位来监控电源电压。若电源电压大于 LVDL 设置的电压档，该标志位会自动清除，即 LVDW 位不具有锁存功能。

5.1. LVD 相关寄存器汇总

名称	状态		寄存器	地址	复位值
LVDL	$V_{LVD-REF}$	0000 = 1.8 0001 = 2.0 0010 = 2.4 0011 = 2.7 0100 = 3.0 0101 = 3.3 0110 = 3.6 0111 = 4.0 1xxx = 1.2	PCON[7:4]	0x8E	RW-0000
LVDEN	LVD	1 = 使能 0 = 关闭	PCON[3]		RW-0
LVDW	LVD 触发?	1 = Yes (不锁存) 0 = No	PCON[2]		RO-x
LVDCKS	LVD 消抖时钟	00 = LIRC 01 = HIRC 10 = Sysclk 11 = 指令时钟	LVDCON[6:5]	0x21A	RW-00
LVDP	LVDW 极性	1 = 检测电压 > $V_{LVD-REF}$ 0 = 检测电压 < $V_{LVD-REF}$	LVDCON[4]		RW-0
LVDDEB	LVD 消抖	1 = 使能 (消抖时间为 3 – 4x LVDCKS) 0 = 禁止	LVDCON[3]		RW-1
LVDM	LVD 检测源	00 = PB3 01 = VDD 1x = PD5	LVDCON[1:0]		RW-00
LFMOD	1: LIRC = 256 kHz 0: LIRC = 32 kHz		OSCCON[7]	0x8F	RW-0
LVDADJ [1]	LVDL 修调位，~3% / step		LVDTRIM[6:3]	0x21C	RW-1000

表 5-1 LVD 用户设置和标志寄存器

[1] 只受上电复位影响，其它复位不起作用

名称	状态		寄存器	地址	复位值
GIE	全局中断	1 = 使能 (PEIE, LVDIE 适用) 0 = 全局关闭 (唤醒不受影响)	INTCON[7]	Bank 首地址 +0x0B	RW-0
PEIE	外设总中断	1 = 使能 (LVDIE 适用) 0 = 关闭 (无唤醒)	INTCON[6]		RW-0
LVDIE	LVD 中断使能	1 = 使能 0 = 关闭 (无唤醒)	PIE1[5]	0x8C	RW-0
LVDIF	LVD 产生中断？	1 = Yes 0 = No，或已被清零	PIR1[5]	0x0C	RW0-0

表 5-2 LVD 用户设置和标志寄存器

5.2. 检测外部电压

除了可以监控片内 VDD 外，LVD 模块还具备检测外部电压的功能。寄存器位 LVDM 决定了 LVD 作用于 VDD 还是外部电压。当选择对 EVLD 进行检测时，对应管脚的斯密特输入被关闭以防漏电。

注：ELVD 管脚的电压不能高于 VDD。

5.3. LVD 中断

除了通过轮询 LVDW 位了解低电压侦测事件外，软件还可以能过中断的方式来获得低电压的情况。当低电压侦测事件发生后，LVDIF 位自动置 1，它是一个电平触发锁存器，只能通过软件清 0，清 0 的前提是电源电压恢复到 LVDL[3:0]设置的水平以上，LVD 事件结束。

当 PEIE 和 LVDIE 被置 1 且 LVDIF 为 1 时，睡眠状态下 LVD 中断标志位还可以作为一个唤醒源，如果之前 GIE = 1，则唤醒后 CPU 进入中断处理。

6. 振荡器和系统时钟

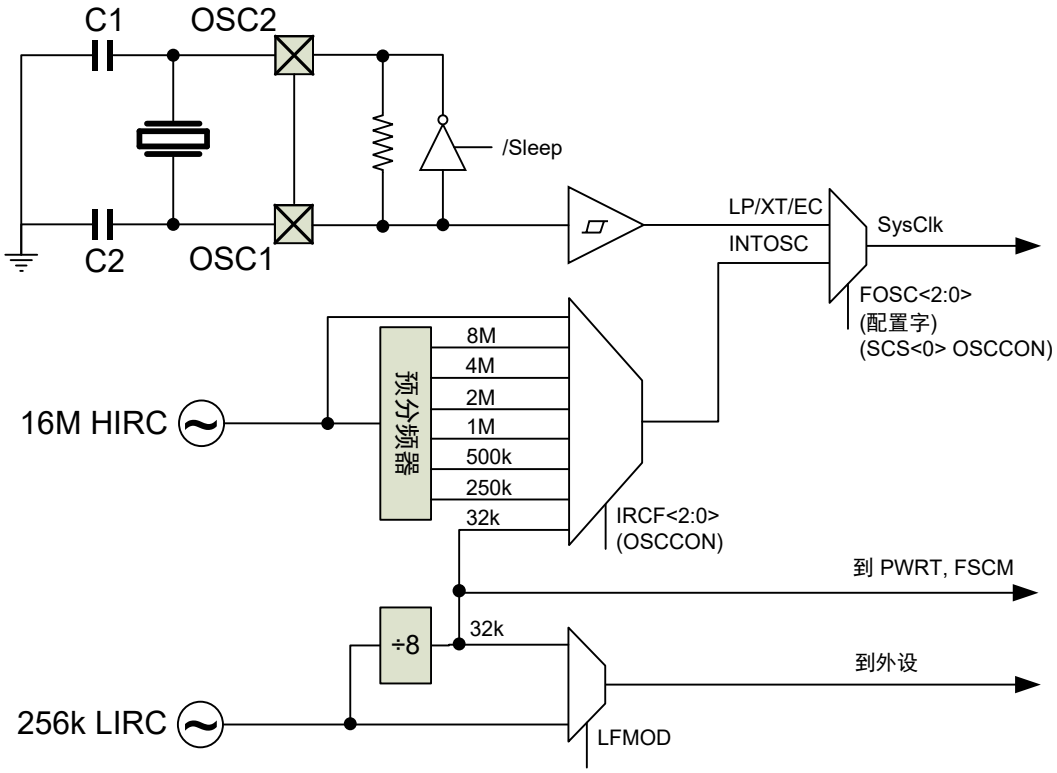


图 6-1 系统时钟源框图

本芯片包含 4 个时钟源：2 个内置 RC 振荡器（高、低速），1 个外部晶体振荡器，1 个外部时钟灌入源。内置振荡器包括 1 个内部 16M 高速精准振荡器（HIRC），1 个内部 32kHz / 256kHz (LIRC) 低速低功耗振荡器。这些时钟或振荡器结合预分频器可以给系统提供各种频率的时钟源。

系统时钟源的预分频器比例由 OPTION 寄存器里的 IRCF[2:0] 位控制。

6.1. 振荡器模块相关寄存器汇总

名称	功能	默认
FOSC	• LP：PA1 (+) 和 PA2 (−) 接外部低速晶振 • XT：PA1 (+) 和 PA2 (−) 接外部高速晶振 • EC：PA1 (+) 接外部时钟输入，PA2 为 I/O • INTOSC：PA2 (或 PD5) 输出系统时钟的 2 分频，PA1 为 IO 管脚 • INTOSCIO：PA1 和 PA2 为 I/O	INTOSCIO
IESO	XT / LP 双速时钟启动 • 使能 • 关闭	使能
FCMEN	故障保护时钟监控器 • 使能 • 关闭	使能
TSEL	指令时钟与系统时钟的对应关系 (2T or 4T) • 2 (指令时钟 = Sysclk/2) • 4 (指令时钟 = Sysclk/4)	2

表 6-1 FOSC 和双速启动初始化配置寄存器

SysClk 系统时钟源			配置			
			SCS	IRCF	LFMOD	OST
			OSCCON[0]	OSCCON[6:4]	OSCCON[7]	(固定值)
			0x8F			
			RW-0	RW-100	RW-0	
外部	EC		0	-	-	
	XT		0	-	-	1024 (默认值)
	LP		0	-	-	1024 (默认值)
内部	HIRC	16 MHz	1	111	-	-
		8 MHz	1	110	-	-
		4 MHz	1	101	-	-
		2 MHz	1	100	-	-
		1 MHz	1	011	-	-
		500 kHz	1	010	-	-
		250 kHz	1	001	-	-
	LIRC	256 kHz [1]	1	000	1	-
		32 kHz [2]	1	000	0	-

表 6-2 SysClk 系统时钟源设置相关用户寄存器

[1] 256 kHz LIRC 只供 LVD (参阅 LVDCKS 和 LFMOD，**表 5-2**) 和 ADC (参阅 ADCS 和 LFMOD，**表 16-2**) 使用。

[2] 系统时钟源 (IRCF=000)、PWRT、FSCM 和 WDT (WCKSRC=00) 统一使用 LIRC 的 8 分频，即 32 kHz，而不管 LFMOD 为何值。

名称	状态	寄存器	地址	复位值
OSTS	振荡器启动超时状态位(锁存) 1 = 运行在外部振荡器下(启动成功) 0 = 运行在内部振荡器下	OSCCON[3]	0x8F	RO-x
HTS	HIRC ready (锁存) 1 = Yes 0 = No	OSCCON[2]		RO-0
LTS	LIRC ready (锁存) 1 = Yes 0 = No	OSCCON[1]		RO-0
SYSON	Sleep 状态下，系统时钟控制 1 = 保持运行 0 = 关闭	LVDCON[7]	0x21A	RW-0
SCKEN	系统时钟输出 1 = 使能 0 = 禁止	SCKCFG[7]	0x209	RW-0
SCKOE	当 SCKEN = 1 时，系统时钟输出到 IO 1 = PA2 或 PD5 (取决于 CLKOS) 0 = 禁止	SCKCFG[1]		RW-0
ROMLPE	PROM 低功耗模式 1 = 使能 0 = 关闭 注：仅当 Sysclk ≤ 2MHz (2T) 或 Sysclk ≤ 4MHz (4T) 时有效；	MSCON0[5]	0x10C	RW-0
CLKOS	CLKO 管脚映射 1 = PD5 0 = PA2	MSCON0[4]		RW-1
FOSCCAL	内部高速时钟 HIRC 频率调节寄存器 高 5 位	FOSCCALH[4:0]	0x309	RW-x xxxx
	内部高速时钟 HIRC 频率调节寄存器 低 4 位 注：bit[4] 及 bit[2:0] 需保持为 1	FOSCCALL[7:0]	0x109	RW-xxx1 x111
HIRCM	HIRC 中心频率 1 = 13.5 MHz 0 = 16 MHz	MSCON1[0]	0x101/ 0x301	RW-0

表 6-3 振荡器控制位/状态位

6.2. 时钟源模式

时钟源模式分为外部和内部模式。

外部时钟模式依靠外部电路提供时钟源，比如外部时钟 EC 模式，晶体谐振器 XT、LP 模式。

内部时钟模式内置于振荡器模块中，振荡器模块有 16MHz 高频振荡器和 32kHz 低频振荡器。可通过 OSCCON 寄存器的系统时钟选择位 (SCS) 来选择内部高速或者外部时钟源。

6.3. 外部时钟模式

6.3.1. 振荡器起振定时器 (OST)

如果振荡器模块配置为 LP、XT 模式，振荡器起振定时器 (OST)将对来自 OSC1 的振荡计数 1024 次。这发生在上电复位 (POR) 之后以及上电延时定时器 (PWRT) 延时结束 (如果被使能) 时，或从休眠中唤醒后。在此期间，程序计数器不递增，程序执行暂停。OST 确保使用石英晶体谐振器或陶瓷谐振器的振荡器电路已经启动并向振荡器模块提供稳定的系统时钟信号。当在时钟源之间切换时，需要一定的延时以使新时钟稳定。

注意：

1. OST 复用了 WDT 定时器，故在 OST 对晶体时钟计数时，WDT 功能被屏蔽，待 OST 发生溢出后，WDT 功能才恢复 (如果此前 WDT 被使能的话)。
2. 由于 OST 和 WDT 共用一个定时器，所以在 OST 未溢出期间，不要写 WDTCON 或 OPTION 寄存器，否则可产生不可预期的行为。

6.3.2. EC 模式

外部时钟模式允许外部产生的逻辑电平作为系统时钟源。工作在此模式下时，外部时钟源连接到 OSC1 输入，OSC2 引脚可用作通用 I/O。

当选取 EC 模式时，振荡器起振定时器 (OST) 被禁止。因此，上电复位 (POR) 后或者从休眠中唤醒后的操作不存在延时。MCU 被唤醒后再次启动外部时钟，器件恢复工作，就好像没有停止过一样。

6.3.3. LP 和 XT 模式

LP 和 XT 模式支持连接到 OSC1 和 OSC2 的石英晶体谐振器或陶瓷谐振器的使用。

模式选择内部反相放大器的低或高增益设定，以支持各种谐振器类型及速度。

LP 振荡器模式选择内部反相放大器的最低增益设定。

LP 模式的电流消耗在两种模式中最小。该模式设计仅用于驱动 32.768 kHz 音叉式晶振 (钟表晶振)。XT 振荡器模式选择内部反相放大器的高增益设定。

6.4. 内部时钟模式

振荡器模块有两个独立的内部振荡器，可配置或选取为系统时钟源。

1. HIRC (高频内部振荡器) 出厂时已校准，工作频率为 16MHz。
2. LIRC (低频内部振荡器) 未经校准，工作频率为 32kHz。软件对 OSCCON 寄存器的系统时钟选择位

IRCF[2:0]进行写操作，可选择系统时钟速度。

可通过 OSCCON 寄存器的系统时钟选择（SCS）位，在外部或内部高速时钟源之间选择系统时钟。

注意：OSCCON 寄存器的 LFMOD 可以选择 LIRC 是 32kHz 或者 256kHz，但看门狗固定使用 32kHz，不管 LFMOD 为何值。

6.4.1. 频率选择位（IRCF）

外部晶体时钟，16MHz HIRC 和 32kHz LIRC 的输出连接到预分频器和多路复用器（见图 6-1）。OSCCON 寄存器的内部振荡器频率选择位 IRCF[3:0] 用于选择不同的分频输出。

可通过软件选择以下 8 个频率之一：

- 16MHz
- 8MHz
- 4MHz
- 2MHz (复位后的缺省值)
- 1MHz
- 500kHz
- 250kHz
- 32 kHz (LIRC)

6.4.2. HIRC 和 LIRC 时钟切换时序

当在 LIRC 和 HIRC 之间切换时，新的振荡器可能为了省电已经关闭（见图 6-2 和图 6-3）。在这种情况下，OSCCON 寄存器的 IRCF 位被修改之后、频率选择生效之前，存在一个延时。OSCCON 寄存器的 LTS 和 HTS 位将反映 LIRC 和 HIRC 振荡器的当前活动状态。频率选择时序如下：

- OSCCON 寄存器的 IRCF[3:0]位被修改
- 如果新时钟是关闭的，会有一个时钟的启动延时
- 时钟切换电路等待当前时钟的 2 个下降沿的到来
- CLKOUT 保持为低，时钟切换电路等待两个新时钟下降沿的到来
- 现在 CLKOUT 连接到新时钟。OSCCON 寄存器的 HTS 和 LTS 位被更新
- 时钟切换完成

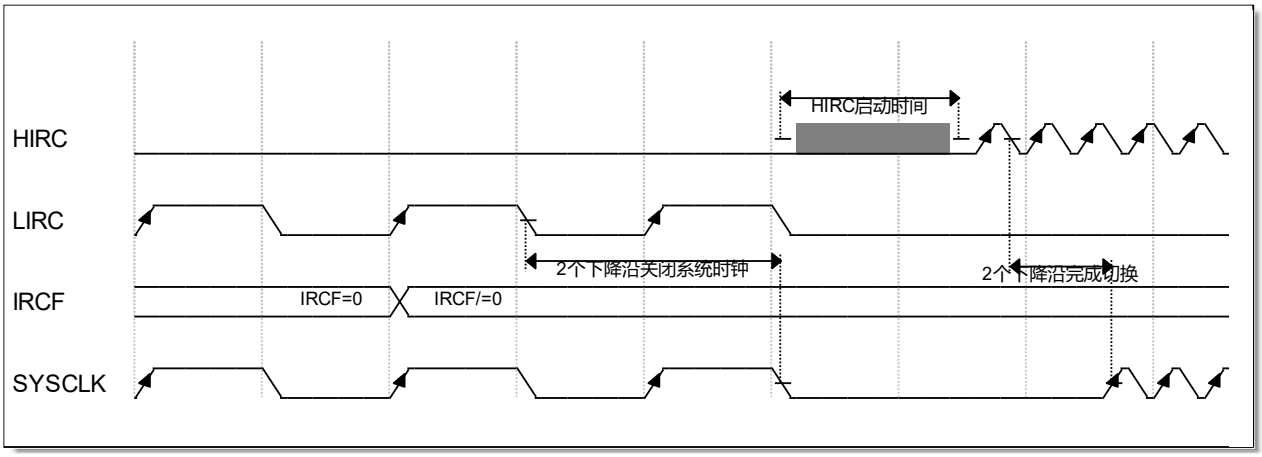


图 6-2 由慢时钟切换到快时钟

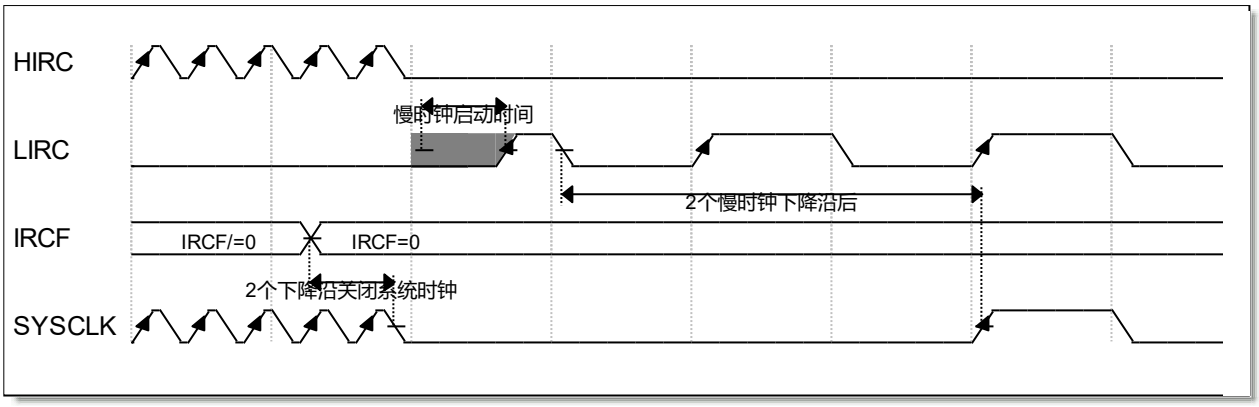


图 6-3 由快时钟切换到慢时钟

6.4.3. 频率与最小工作电压的关系

系统时钟频率越高，MCU 的对最小工作电压（安全工作电压）的要求就越高，例如，2T 工作模式下如果要跑 16M，VDD 应至少在 2.7V 以上。

FT62F28x 系列芯片的上电复位电压 V_{POR} 在 1.6V 左右，即当上电过程 VDD 超过 V_{POR} 之后复位释放，再经过约 8ms 延时之后完成上电配置，然后开始执行程序指令。对于一些慢上电且要跑 16M/2T 的应用，如果 VDD 从 V_{POR} 上升到 2.7V 的 VDDmin 的时间过长，而恰好在这段“死区时间”内程序切换到最高的系统时钟 16M 的话，MCU 将有可能跑飞。

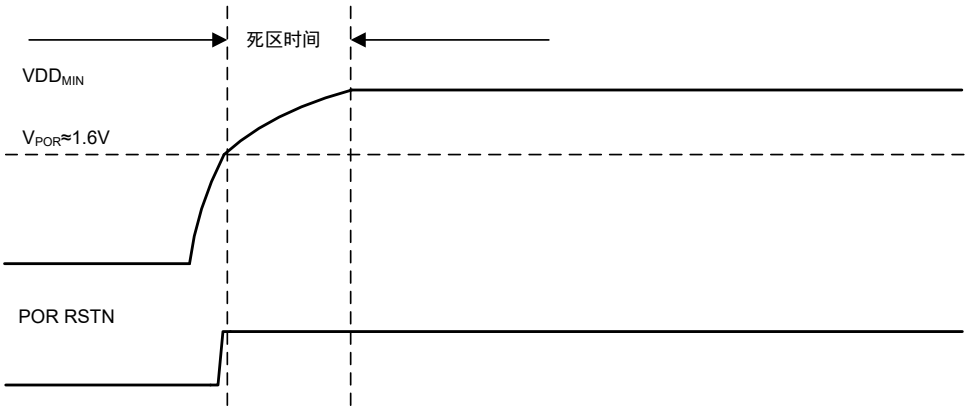


图 6-4 上电过程的“死区时间”

对于慢上电的应用，有以下几种方法可避免这种情况发生：

1. 烧录选项 LVR 必须使能且设置合适的值，比如 16M/2T 应该设置 2.8V 的复位电压；
2. 上电复位后，软件可延时足够长的时间让 VDD 升到安全工作电压后再切换到 16M 的系统时钟，即延时一段时间再做时钟的初始化；
3. 使能 PWRT 选项，PWRT 时间大约为 64ms，这段额外的复位时间有利于让 VDD 爬升到的最小工作电压；

以上 3 点中，强烈建议采用第 1 种方式，因为它不仅能解决上电速度过慢的问题，还能监测在正常工作时的 VDD 意外跌落。

6.4.4. HIRC 频率微调

内建的高精度 HIRC 出厂时被校准至 16MHz 或 13.5MHz@ 2.5V/25℃(取决于"HIRCM")。校准过程是过滤掉制程上的偏差对精度造成的影响，此 HIRC 还会到受工作环境温度和工作电压的影响，其频率会有一定的漂移。

除了出厂校准外，还提供了一种方式供用户对 HIRC 进行微调：通过对 FOSCCAL 寄存器的值进行改写。FOSCCAL 寄存器一共 13 位，{FOSCCALH[4:0]，FOSCCALL[7:0]}，但其中 bit[4] 及 bit[2:0] 需要保持为 1，因此{FOSCCALH[4:0]，FOSCCALL[7:5]，FOSCCALL[3]}等效为一个 9 位的寄存器，设初始值为 FOSCCAL[s]，此时芯片工作在 16MHz 或 13.5MHz，频率变化约为 0.2% / LSb。

6.5. 时钟切换

通过软件对 OSCCON 寄存器的系统时钟选择 (SCS) 位进行操作，可将系统时钟源在外部和内部高速时钟源之间切换。

6.5.1. 系统时钟选择 (SCS) 位

OSCCON 寄存器的系统时钟选择 (SCS) 位选择用于 CPU 和外设的系统时钟源。

OSCCON 寄存器的位 SCS = 0 时，系统时钟源由配置字寄存器 (UCFG0) 中 FOSC[2:0]位的配置决定。

OSCCON 寄存器的位 SCS = 1 时，忽略 FOSC[2:0]位，根据 OSCCON 寄存器的 IRCF[2:0]位决定系统时钟源：HIRC 的分频时钟或者 32k 时钟。

注：

1. 任何由硬件引起的时钟切换 (可能产生自双速启动或故障保护时钟监控器) 都不会更新 OSCCON 寄存器的 SCS 位。用户应该监控 OSCCON 寄存器的 OSTS 位以确定当前的系统时钟源；
2. 当 IRCF[2:0]等于 0 时，无论 SCS 为何值，系统时钟都选择内部慢时钟。

6.5.2. 振荡器起振超时状态（OSTS）位

OSCCON 寄存器的振荡器起振超时状态（OSTS）位用于指示系统时钟是来自外部时钟源，还是来自内部时钟源。外部时钟源由配置字寄存器（UCFG0）的 FOSC[2:0]定义。OSTS 还特别指明在 LP 或 XT 模式下，振荡器起振定时器（OST）是否已超时。

6.5.3. 双速时钟启动模式

双速启动模式通过最大限度地缩短外部振荡器起振与代码执行之间的延时，进一步节省了功耗。对于频繁使用休眠模式的应用，双速启动模式将在器件唤醒后除去外部振荡器的起振时间，从而可降低器件的总体功耗。该模式使得应用能够从休眠中唤醒，将 INTOSC 用作时钟源执行数条指令，然后再返回休眠状态而无需等待主振荡器的稳定。

注：执行 SLEEP 指令将中止振荡器起振时间，并使 OSCCON 寄存器的 OSTS 位保持清零。

当振荡器模块配置为 LP 或 XT 模式时，振荡器起振定时器（OST）使能（见**章节 6.3.1** “振荡器起振定时器”）。OST 将暂停程序执行，直到完成完成 1024 次振荡计数。双速启动模式在 OST 计数时使用内部振荡器进行工作，使代码执行的延时最大限度地缩短。当 OST 计数到 1024 且 OSCCON 寄存器的 OSTS 位置 1 时，程序执行切换至外部振荡器。

6.5.4. 双速启动模式配置

通过以下设定来配置双速启动模式：

- 配置字寄存器（UCFG1）中的位 IESO = 1；（使能双速启动模式）
- OSCCON 寄存器的位 SCS = 0 (内部/外部切换位)
- 配置字寄存器（UCFG0）中的 FOSC[2:0]配置为 LP 或 XT 模式

在下列操作之后，进入双速启动模式：

- 上电复位（POR）且上电延时定时器（PWRT）延时结束（使能时）后，
- 从休眠状态唤醒

如果外部时钟振荡器配置为除 LP 或 XT 模式以外的任一模式，那么双速启动将被禁止。这是因为 POR 后或从休眠中退出时，外部时钟振荡器不需要稳定时间。

6.5.5. 双速启动顺序

- 从上电复位或休眠中唤醒
- 使用内部振荡器以 OSCCON 寄存器的 IRCF[2:0]位设置的频率开始执行指令
- OST 使能，计数 1024 个时钟周期
- OST 超时，等待内部振荡器下降沿的到来
- OSTS 置 1
- 系统时钟保持为低，直到新时钟下一个下降沿的到来（LP 或 XT 模式）
- 系统时钟切换到外部时钟源

6.5.6. 故障保护时钟监控器

故障保护时钟监控器 (FSCM) 使得器件在出现外部振荡器故障时仍能继续工作。FSCM 能在振荡器起振延时定时器 (OST) 到期后的任一时刻检测振荡器故障。FSCM 通过将配置字寄存器 (UCFG1) 中的 FSCMEN 位置 1 来使能。FSCM 可用于所有外部振荡模式 (LP、XT 和 EC)。

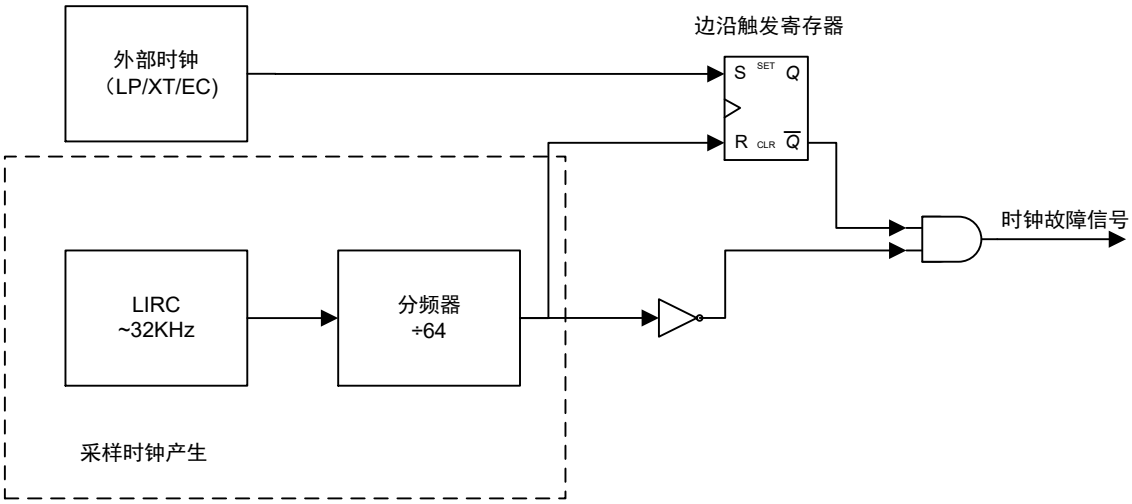


图 6-5 FSCM 原理框图

6.5.7. 故障保护检测

FSCM 模块通过将外部振荡器与 FSCM 采样时钟比较来检测振荡器故障。LIRC 除以 64，就产生了采样时钟。请参见图 6-5。故障检测器内部有一个锁存器。在外部时钟的每个下降沿，锁存器被置 1。在采样时钟的每个上升沿，锁存器被清零。如果采样时钟的整个半周期流逝而主时钟依然未进入低电平，就检测到故障。

6.5.8. 故障保护操作

当外部时钟出现故障时，FSCM 将器件时钟切换到内部时钟源，并将 PIR1 寄存器的 OSFIF 标志位置 1。如果在 PIR1 寄存器的 OSFIE 位置 1 的同时将该标志位置 1，将产生中断。固件随后会采取措施减轻可能由故障时钟所产生的问题。系统时钟将继续来自内部时钟源，直到器件固件成功重启外部振荡器并切换回外部操作。

FSCM 所选的内部时钟源由 OSCCON 寄存器的 IRCF[2:0]位决定。这使内部振荡器可以在故障发生前就得以配置。

6.5.9. 故障保护条件清除

复位、执行 SLEEP 指令或翻转 OSCCON 寄存器的 SCS 位后，故障保护条件被清除。OSCCON 寄存器的 SCS 位被修改后，OST 将重新启动。OST 运行时，器件继续从 OSCCON 中选定的 INTOSC 进行操作。OST 超时后，故障保护条件被清除，器件将从外部时钟源进行操作。必须先清除故障保护条件，才能清零 OSFIF 标志位。

6.5.10. 复位或从休眠中唤醒

FSCM 设计为能在振荡器起振延时定时器 (OST) 到期后的任一时刻检测振荡器故障。OST 可用在从休眠状态唤醒之后和任何类型的复位之后。OST 不能在 EC 时钟模式下使用，所以一旦复位或唤醒完成，FSCM 就处于激活状态。

注：由于振荡器起振时间的范围变化较大，在振荡器起振期间 (从复位或休眠中退出时)，故障保护电路不处于激活状态。经过一段适当的时间后，用户应检查 OSCCON 寄存器的 OSTS 位，以验证振荡器是否已成功起振以及系统时钟是否切换成功。

6.6. 系统时钟输出

通过先置位 SCKCFG 寄存器的 SCKEN 位,然后再将 SCKOE 位置高，可以将系统时钟输出到 PA2 或 PD5。当不再需要输出时钟时，将 SCKCFG 寄存器的 SCKEN 和 SCKOE 位都清零即可。

软件写 SCKCFG 寄存器有一定限制，即写之前必须执行把 EECON1 的 WREN3/2/1 位置 1，写完成后要把 WREN3/2/1 位清零，以免后面的指令误触发写操作，读操作则不需要上述步骤。

7. 慢时钟测量

芯片集成了两个内部 RC 振荡器，一个是经过出厂校准的高速高精度的 16M 快时钟 HIRC，一个是低速低功耗的 32k 时钟 LIRC，利用慢时钟测量功能可以把 LIRC 的周期用系统时钟计算出来。此功能可以比较精准的测量内部慢时钟周期。

7.1. 慢时钟测量相关寄存器汇总

名称	状态	寄存器	地址	复位值
CKMAVG	LIRC 和 HIRC 交叉校准时 4 次平均测量模式 1 = 使能 0 = 关闭	MSCON0[2]	0x10C	RW-0
CKCNTI	启动 LIRC 和 HIRC 的交叉校准功能 1 = 启动 0 = 完成 (自动清零)	MSCON0[1]		RW-0
TMR2ON	Timer2 (PWM 单脉冲模式下自动清 0) 1 = 使能 0 = 关闭	T2CON0[2]	0x1E	RW-0
SOSCPR	校准 LIRC 周期所需的 HIRC 周期数	SOSCPR[11:0]	0x219[3:0] 0x218[7:0]	RW-FFF

表 7-1 慢时钟测量控制位/状态位

名称	状态		寄存器	地址	复位值
GIE	全局中断 1 = 使能 (CKMIE, PEIE 适用) 0 = 全局关闭 (唤醒不受影响)		INTCON[7]	Bank 首地址 +0x0B	RW-0
PEIE	外设总中断	1 = 使能 (CKMIE 适用) 0 = 关闭 (无唤醒)	INTCON[6]		RW-0
CKMIE	LIRC 和 HIRC 交叉校准完成中断	1 = 使能 0 = 关闭 (无唤醒)	PIE1[6]	0x8C	RW-0
CKMIF	LIRC 和 HIRC 交叉校准完成标志位	1 = Yes (锁存) 0 = No	PIR1[6]	0x0C	RW0-0

表 7-2 慢时钟测量中断控制位/标志位

7.2. 测量原理

处于慢时钟测量模式下，Timer2 的预分频、后分频配置自动变为 1:1，Timer2 的计数时钟为系统时钟 F_{SYS}，而不是普通模式下的指令时钟 $F_{SYS}/2$。测量结束后结果自动存到 SOSCPR 寄存器，其单位是系统时钟的个数。

7.3. 操作步骤

1. 为提高计量精度，建议设置 IRCF 为 111，SCS=1，选择 16M 的系统时钟；
2. 把 T2CON0.2 置 1，使能 Timer2；
3. 如果选择 4 次平均，则把 MSCON0.2 置 1，否则把它清 0；
4. 置位 MSCON0.1，开始测量；
5. 测量结束后 MSCKCON0.1 自动清 0，中断标志置 1；
6. 可以用查询或中断的方式等待结束；
7. 当查询到中断标志为 1 时读取得到的 SOSCPR 即为最终结果。

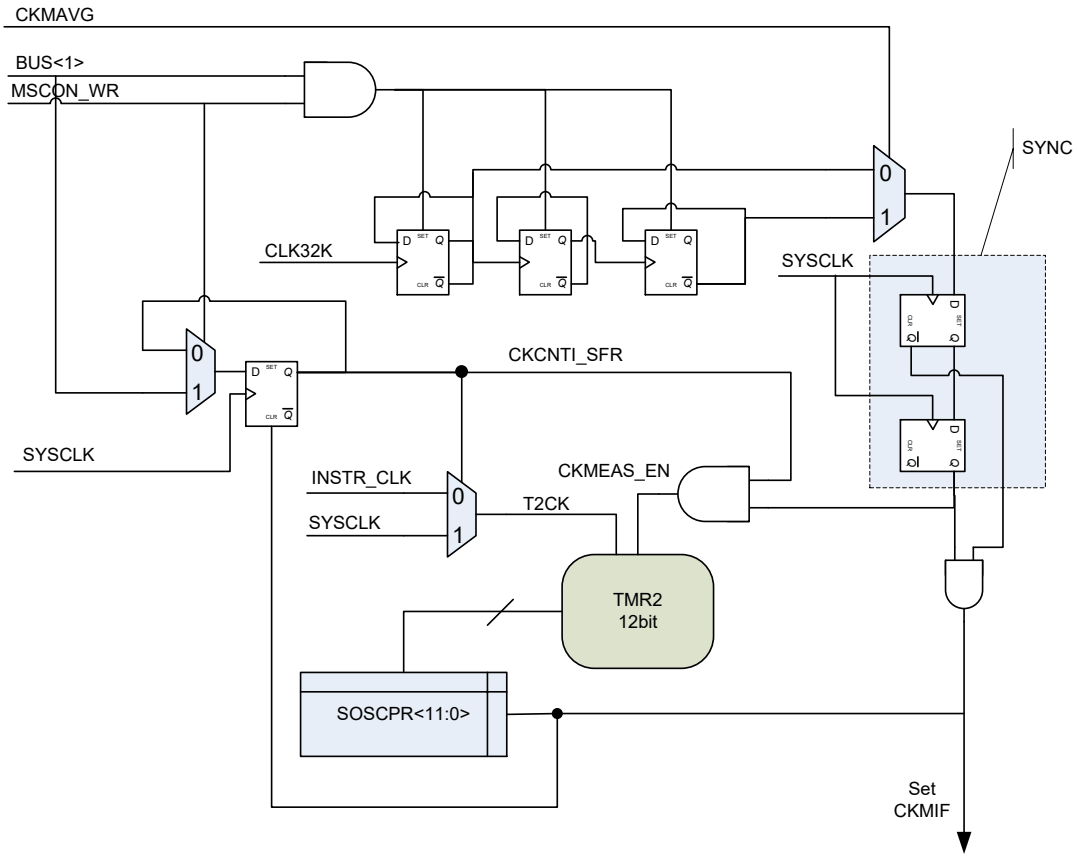


图 7-1 慢时钟测量模式原理框图

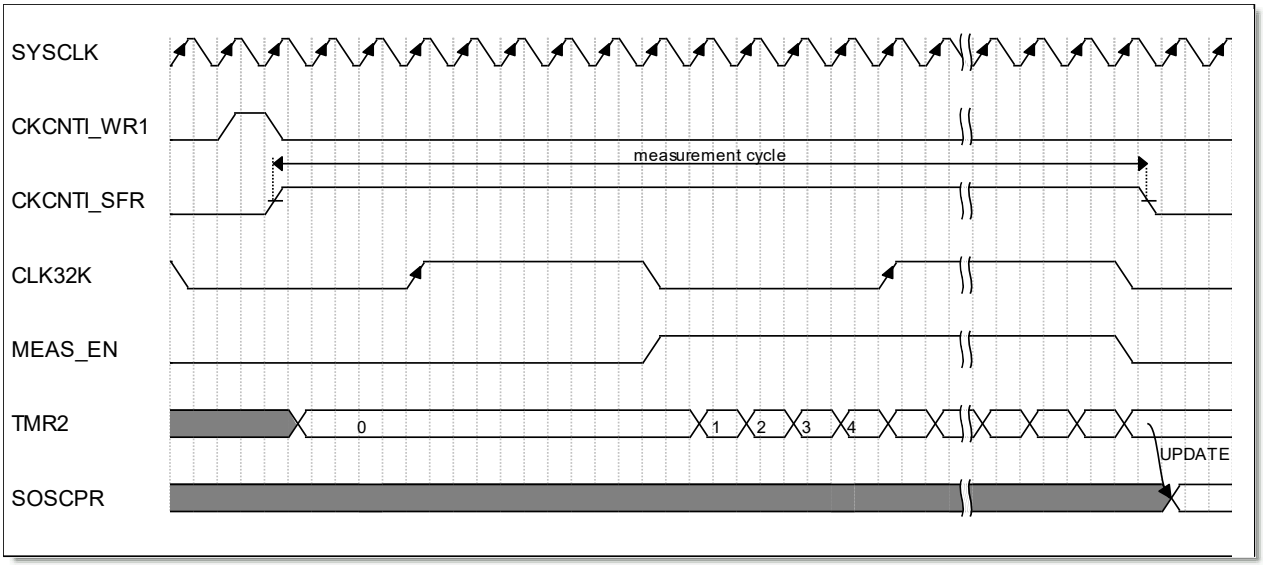


图 7-2 单次测量时序示意图 (CLK32K 和 SYSCLK 未按真实比例画)

注意：

1. 在慢时钟测量过程中软件不能写 SOSCPRH/L；
2. 不能在单步调试下做慢时钟测量，因为暂停模式下 Timer2 被停止，这样会导致测量结果不正确；

8. 看门狗定时器

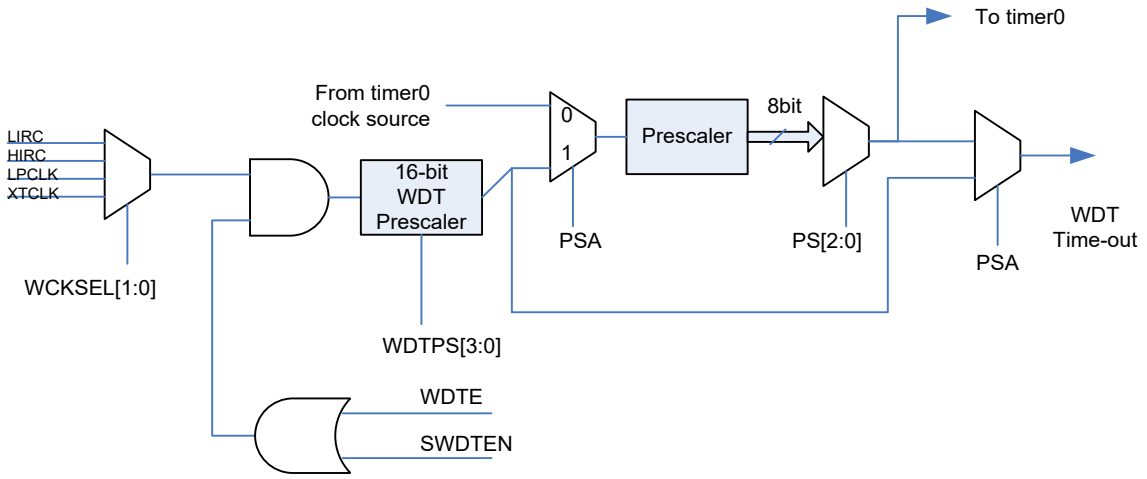


图 8-1 看门狗和定时器 0 框图

看门狗是一个 16 位的计数器，和定时器 0 共用一个 8 位的预分频器，硬件使能位 WDTEN 位于配置寄存器 UCFG0 的第 3 位，软件使能位 SWDTEN 位于 WDTCON 寄存器的第 0 位，为 1 时表示使能看门狗，为 0 时禁止。

清看门狗指令 CLRWDT、SLEEP 会清除看门狗计数器。

在使能了看门狗的情况下，MCU 睡眠时看门狗溢出事件可以作为一个唤醒源，而 MCU 正常工作时作为一个复位源。

条件	看门狗状态
WDTEN 和 SWDTEN 同时为 0	清零
CLRWDT 指令	
进入 SLEEP、退出 SLEEP 时刻	
OST 计数溢出时	
写 WDTCON	

8.1. 看门狗时钟源

看门狗时钟源有以下几种：

- LIRC
- HIRC
- LP 晶体时钟
- XT 晶体时钟

在看门狗被使能的情况下，被选择的时钟源将自动打开，在 SLEEP 模式下也将保持运行。

注意：选择 LP/XT 作为 WDT 时钟源时，配置选项 FOSC 必须是 LP/XT 或者 INTOSCIO 模式，否则，LP/XT 晶体电路将不打开；

8.2. WDT 相关寄存器汇总

<table>
<tr><th>名称</th><th>状态</th><th>寄存器</th><th>地址</th><th>复位值</th></tr>
<tr><td>WCKSRC</td><td>WDT 时钟源
00 = LIRC
01 = HIRC
10 = LP (仅当 FOSC 为 LP 或 INTOSCIO 模式*)
11 = XT (仅当 FOSC 为 XT 或 INTOSCIO 模式*)
*否则配置错误，无 WDT 时钟源</td><td>WDTCON[6:5]</td><td rowspan="3">0x99</td><td>RW-00</td></tr>
<tr><td>WDTPS</td><td>WDT 周期
0000 = 32   0111 = 4096
0001 = 64   1000 = 8192
0010 = 128   1001 = 16384
0011 = 256   1010 = 32768
0100 = 512 (默认)   1011 = 65536
0101 = 1024   11xx = 65536
0110 = 2048</td><td>WDTCON[4:1]</td><td>RW-0100</td></tr>
<tr><td>SWDTEN</td><td>1 = WDT 使能
0 = WDT 关闭 (当 WDTE 选择由 SWDTEN 控制时)</td><td>WDTCON[0]</td><td>RW-0</td></tr>
<tr><td>PSA</td><td>1 = 分频电路分配给 WDT 后分频器
0 = 分频电路分配给 Timer0 预分频器</td><td>OPTION[3]</td><td rowspan="2">0x81</td><td>RW-1</td></tr>
<tr><td>PS</td><td>
<table>
<tr><th>Value</th><th>WDT 后分频比</th><th></th><th>Timer0 预分频比</th><th></th></tr>
<tr><td>000</td><td rowspan="8">(PSA=1)</td><td>1</td><td rowspan="8">(PSA=0)</td><td>2</td></tr>
<tr><td>001</td><td>2</td><td>4</td></tr>
<tr><td>010</td><td>4</td><td>8</td></tr>
<tr><td>011</td><td>8</td><td>16</td></tr>
<tr><td>100</td><td>16</td><td>32</td></tr>
<tr><td>101</td><td>32</td><td>64</td></tr>
<tr><td>110</td><td>64</td><td>128</td></tr>
<tr><td>111</td><td>128</td><td>256</td></tr>
<tr><td>xxx</td><td>(PSA=0)</td><td>1</td><td>(PSA=1)</td><td>1</td></tr>
</table>
</td><td>OPTION[2;0</td><td>RW-111</td></tr>
</table>

表 8-1 WDT 相关用户寄存器

名称	功能	默认
WDTE	WDT • 使能 (指令不能禁止) • 通过指令控制 (SWDTEN)	SWDTEN 控制
WDTBTE	WDT 复位启动初始化配置	

表 8-2 WDT 选项初始化配置寄存器

9. 定时器 0

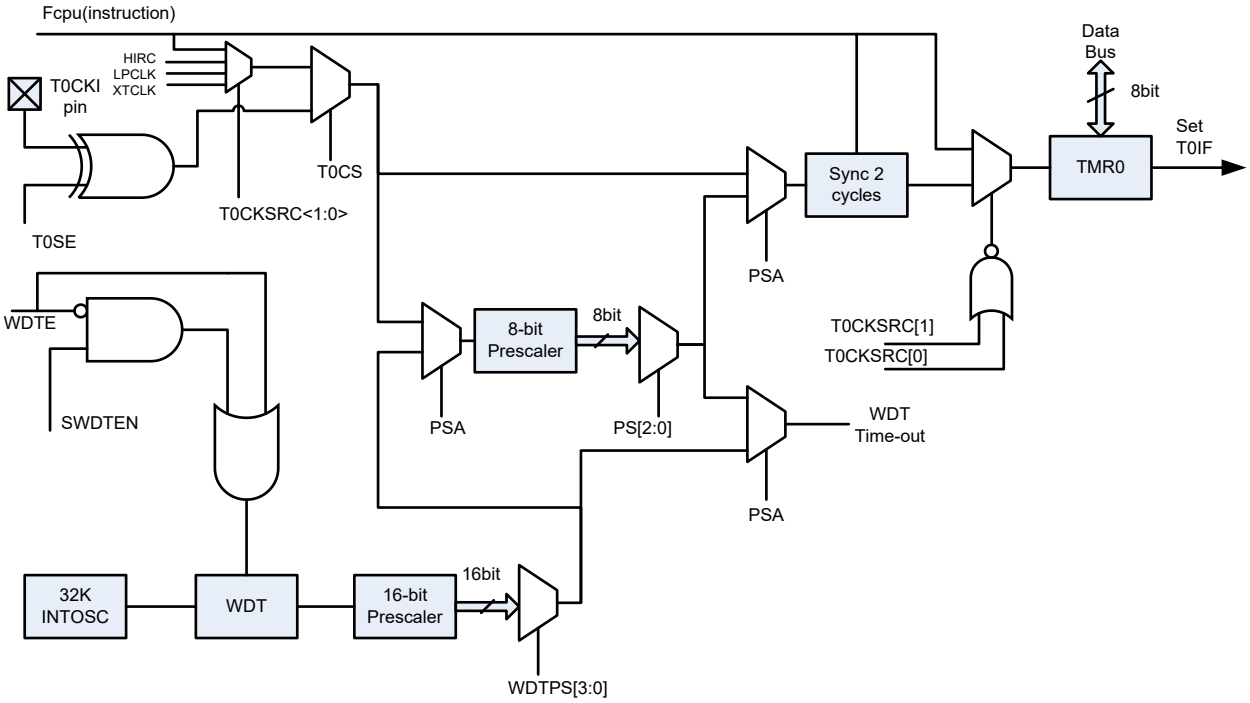


图 9-1 看门狗和定时器 0 框图

定时器 0 为 8 位，可配置为计数器或定时器使用，当作为外部事件 (T0CKI) 计数器时，可以配置为上升沿或者下降沿计数。作为定时器时，其计数时钟源由 T0CKSRC 可选时钟源，即每一时钟周期递增一次。有一个与 WDT 共用的 8 位预分频器，PSA 为 0 时该预分频器分配给定时器 0 使用。

注：当改变 PSA 的值时，硬件会自动把预分频器清 0。

9.1. Timer0 相关寄存器汇总

名称	状态		寄存器	地址	复位值
P0ANP	P0AN 极性	1 = 取反	MSCON2[7]	0x10D	RW-0
P0AP	P0A 极性	0 = 正向	MSCON2[6]		RW-0
P0ANOE	P0AN 通道输出到 PA4	1 = 使能 0 = 关闭	T0CON0[5]	0x105	RW-0
P0AOE	P0A 通道输出到 PA5	1 = 使能 0 = 关闭	T0CON0[4]		RW-0
T0ON	Timer0	1 = 使能 0 = 禁止	T0CON0[3]		RW-1
T0CKRUN	SLEEP 模式下 T0CK 保持运行	1 = Yes (时钟源非指令时钟) 0 = No	T0CON0[2]		RW-0
T0CKSRC	Timer0 时钟源选择 (T0CS = 0) 00 = 指令时钟 10 = LP (*) 01 = HIRC 11 = XT (*) (*) FOSC 应相应配置成 LP/XT 模式或选择 INTOSCIO 模式，否则振荡器将不会运行；		T0CON0[1:0]		RW-00
T0DUTY	PWM0 (P0A / P0AN) 占空比		T0DUTY[7:0]	0x108	RW-0000 0000
PR0	PR0 周期寄存器		PR0[7:0]	0x107	RW-1111 1111
TMR0	Timer0 计数值		TMR0[7:0]	0x01	RW-xxxx xxxx
T0CS	Timer0 输入源	1 = PC5 / T0CKI (计数器) 0 = T0CKSRC (定时器)	OPTION[5]	0x81	RW-1
T0SE	计数器触发沿	1 = 下降沿 0 = 上升沿	OPTION[4]		RW-1
PSA	1 = 分频电路分配给 WDT 后分频器 0 = 分频电路分配给 Timer0 预分频器		OPTION[3]		RW-1

PS	Value	WDT 后分频比		Timer0 预分频比		OPTION[2:0]		RW-111
	000	(PSA=1)	1	(PSA=0)	2			
	001		2		4			
	010		4		8			
	011		8		16			
	100		16		32			
	101		32		64			
	110		64		128			
	111		128		256			
	xxx	(PSA=0)	1	(PSA=1)	1			

表 9-1 Timer0 相关用户控制寄存器

名称	状态	寄存器	地址	复位值
GIE	全局中断 1 = 使能 (T0IE 适用) 0 = 全局关闭 (唤醒不受影响)	INTCON[7]	Bank 首地址 +0x0B	RW-0
T0IE	Timer0 溢出中断控制位 1 = 使能 0 = 关闭 (无唤醒)	INTCON[5]		RW-0
T0IF	Timer0 溢出中断标志位 1 = 已经溢出 (锁存) 0 = 未溢出	INTCON[2]		RW0-0

表 9-2 Timer0 中断使能和状态位

9.2. Timer0 定时器模式

该模式下，定时器 0 在每个时钟周期 (时钟源可选)加 1 (不带预分频)。软件可以清零 OPTION 寄存器里的 T0CS 位以进入定时器模式。当软件对 TMR0 进行写操作时，定时器在写后面 2 个周期内不会递增。

9.2.1. Timer0 的时钟源

Timer0 的时钟源由寄存器位 T0CKSRC 控制，默认选择指令时钟。

- 指令时钟
- HIRC 内部高速时钟
- LP 晶体时钟
- XT 晶体时钟

在改变 Timer0 的时钟配置之前，建议先把 T0ON 清 0，以防止时钟切换过程中产生的毛刺对 Timer0 的影响。

在非 SLEEP 模式下且 Timer0 被使能 (T0ON=1)的情况下，被选择的时钟源将自动打开；

注意：

1. 选择 LP 作为 T0 时钟源时，配置选项 FOSC 必须是 LP 或者 INTOSCIO 模式，否则，LP 晶体电路将不打开；
2. 同理，选择 XT 作为 T0 时钟源时，配置选项 FOSC 必须是 XT 或者 INTOSCIO 模式，否则，XT 晶体电路将不打开。

在 SLEEP 模式下，如果要 Timer0 继续保持计数，则需要把 T0CKRUN 置 1，且时钟源不能选择指令时钟，因为在 SLEEP 模式下，指令时钟是关闭的。

9.2.2. TMR0 寄存器的读写

当 Timer0 时钟源跟 CPU 时钟不一样时 (如配置为 LP 或者 XT 模式)，建议软件对 TMR0 的启动读写操作时先把 T0ON 清 0，以避免读错或写错。

9.3. Timer0 计数器模式

该模式下，定时器 0 由每个 T0CKI 管脚的上升沿或下降沿触发加 1(不带预分频)。具体哪一钟沿触发由 OPTION 寄存器里的 T0SE 位决定。软件可以将 OPTION 寄存器里的 T0CS 位置 1 以进入计数器模式。

9.3.1. 软件可配置预分频电路

芯片在定时器 0 和 watchdog 定时器前面有一个分频电路，可以分配给 Timer0 或者 watchdog 定时器用，但二者不能同时使用这个预分频电路。具体分配给 Timer0 还是 watchdog 由 OPTION 寄存器里的 PSA 位决定，PSA 为 0 时，预分频分配给 Timer0 使用。在 Timer0 预分频模式下，总共有 8 个预分频比（1:2 到 1:256）可以通过 OPTION 寄存器里的 PS[2:0]位设置。

预分频电路既不可读也不可写。任何对 TMR0 寄存器的写动作会清零预分频电路。

当预分频电路分配给 watchdog 时，1 条 CLRWDT 指令会清零预分频电路。

9.3.1.1. 在定时器和 watchdog 之间切换预分频电路

由于分频电路可以分配给 Timer0 或者 watchdog 定时器用，在二者之间切换预分频器是有可能导致误复位。

在将预分频电路从分配给 TMR0 切换到分配给 watchdog 时，请务必执行以下指令顺序：

```
BANKSEL   TMR0
CLRWDT                          ;Clear WDT
CLRR      TMR0                  ;Clear TMR0 and prescaler
BANKSEL   OPTION_REG
BSR       OPTION_REG,PSA        ;Select WDT
CLRWDT

LDWI      b'11111000'           ;Mask prescaler bits
ANDWR     OPTION_REG,W
IORWI     b'00000101'           ;Set WDT prescaler bits to 1:32
STR       OPTION_REG
```

在将预分频电路从分配给 watchdog 切换到分配给 TMR0 时，请务必执行以下指令顺序：

```
CLRWDT                          ;Clear WDT andprescaler
BANKSEL   OPTION_REG
LDWI      b'11110000'           ;Mask TMR0 select and prescaler bits
ANDWR     OPTION_REG,W
IORWI     b'00000011'           ;Set prescale to 1:16
STR       OPTION_REG
```

9.3.2. 定时器 0 中断

当寄存器 TMR0 (定时器 0 计数值) 与 PR0 匹配时将置起 T0IF 标志，并产生中断 (如果使能了的话)。注意，Timer0 中断无法唤醒 CPU，因为在睡眠状态下，定时器是被冻结的，除非 T0CKRUN 为 1 且其时钟源不是指令时钟时。

9.3.3. 用外部时钟驱动定时器 0

在计数其模式下，T0CKI 管脚输入和 Timer0 寄存器之间的同步，是由 Timer0 的时钟源（时钟源可选）产生的非交叠时钟来采样实现的，所以外部时钟源周期的高电平时间和低电平时间必须满足相关时序要求。

9.3.4. 睡眠模式下的状态

当 T0CKRUN=1 而且 Timer0 的时钟源不是选择指令时钟时，MCU 进入睡眠后，Timer0 以保持在运行状态，由 T0CKSRC 所选择的时钟源不会关闭。否则，Timer0 将停止计数，维持睡眠前的计数值。

9.4. Timer0 的 PWM 模式

Timer0 具有 1 路极性可配置、带死区互补功能的 PWM (PWM0)，它既可作为电容按键的充放电扫描驱动信号，也可以作为单独 PWM 输出到 I/O 上，由 P0AOE 和 P0ANOE 位控制。

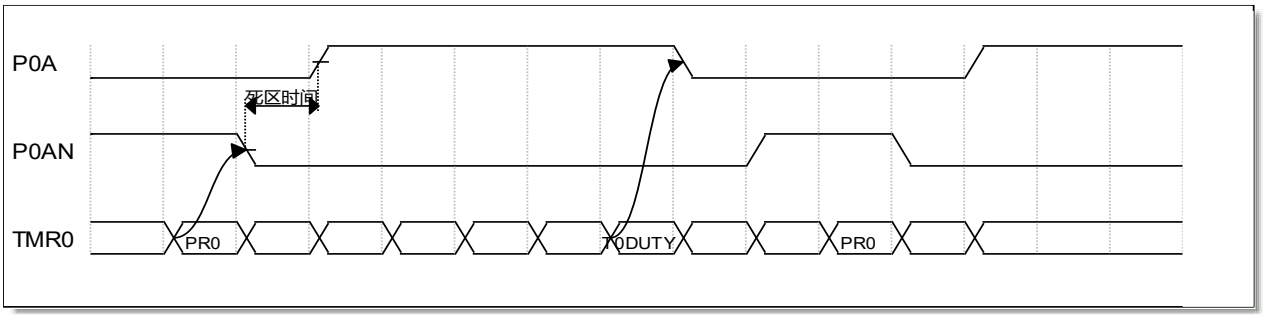


图 9-2 带死区时间的 PWM0 输出

9.4.1. 周期和占空比

PWM0 的周期由 PR0 决定：

PWM0 周期 = (PR0 + 1)*TT0CK*(TMR0 预分频值)……公式 9.1

占空比 TODUTY 决定：

脉冲宽度 = T0DUTY*TT0CK*(TMR0 预分频值)…… ……公式 9.2

占空比 = T0DUTY÷(PR0+1)……公式 9.3

9.4.2. 睡眠时的状态

Timer0 可以在睡眠下保持工作，前提是其时钟源为非指令时钟（睡眠时指令时钟停止），且 T0CKRUN 被设置为 1。在这种条件下，PWM0 可保持正常输出。

10. 定时器 1

定时器 1 为 12 位定时器，包含以下功能：

- 12 位计数寄存器
- 12 位周期寄存器
- TMR1 值等同 PR1 时产生中断
- 1:1，1:4，1:16 预分频比（与 Timer2 共用同一分频器）

Timer1 的时钟源由寄存器位 T1CKSRC 控制，默认选择指令时钟。

- 指令时钟
- HIRC 内部高速时钟
- LP 晶体时钟
- XT 晶体时钟

注意：当 TKMOD 为 1 时，Timer1 时钟源固定选择为 Timer0 占空比匹配脉冲。

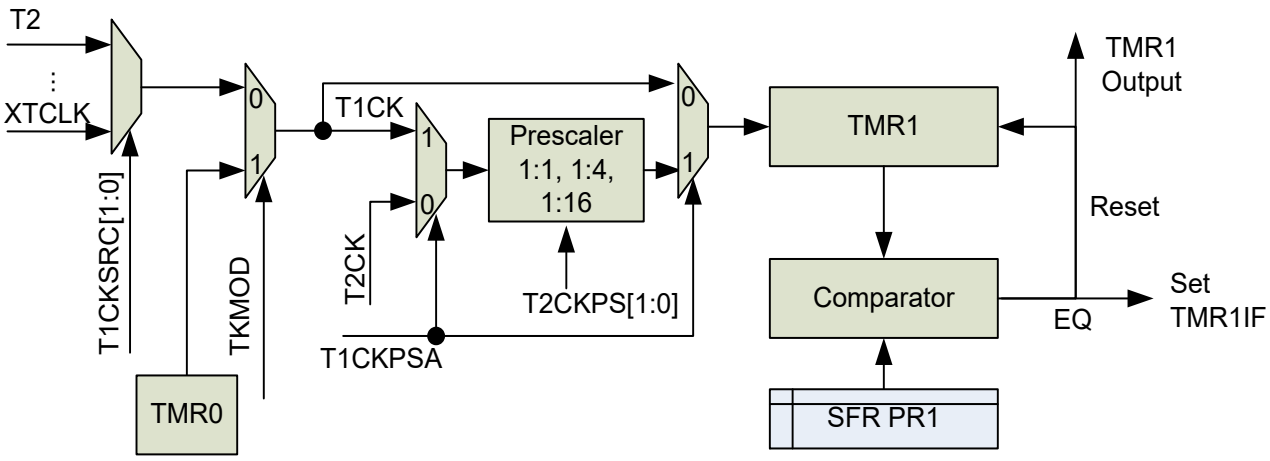


图 10-1 Timer1 结构框图

10.1. Timer1 相关寄存器汇总

名称	状态		寄存器	地址	复位值
T1AUSTP	Timer1 自动关闭	1 = 使能 0 = 禁止	T1CON0[6]	0x106	RW-0
T1ECKE	Timer1 输入源	1 = PA2/T1CKI (计数器) 0 = T1CKSRC (定时器)	T1CON0[5]		RW-0
T1CKPSA	预分频器分配位	1 = Timer1 0 = Timer2	T1CON0[4]		RW-0
T1CKRUN	睡眠时 T1CK 保持运行	1 = Yes (时钟源非指令时钟) 0 = No	T1CON0[3]		RW-0
T1ON	Timer1	1 = 使能 0 = 关闭	T1CON0[2]		RW-0
T1CKSRC	Timer1 时钟源选择 (T1ECKE = 0) 00 = 指令时钟 10 = LP (*) 01 = HIRC 11 = XT (*) (*) FOSC 应相应配置成 LP/XT 或选择 INTOSCIO 模式，否则振荡器将不会运行		T1CON0[1:0]		RW-00
T2CKPS (T1CKPS)	Timer2 / Timer1 预分频比 00 = 1 01 = 4 1x= 16		T2CON0[1:0]	0x1E	RW-00
PR1L	PR1 周期寄存器低 8 位		PR1L[7:0]	0x117	RW-1111 1111
PR1H	PR1 周期寄存器高 4 位		PR1H[3:0]	0x118	RW-1111
TMR1L	TMR1 计数结果寄存器低 8 位		TMR1L[7:0]	0x119	RW-0000 0000
TMR1H	TMR1 计数结果寄存器高 4 位		TMR1H[3:0]	0x11A	RW-0000
T1EVTS	Timer1 自动关闭事件源 (当 T1AUSTP = 1 时) 0xx = LVD 中断事件 100 = CM0 中断事件 101 = CM1 中断事件 110 = LVD 或 CM0 中断事件 111 = LVD 或 CM1 中断事件		MSCON2[2:0]	0x10D	RW-000

表 10-1 Timer1 相关用户控制寄存器

名称	状态	寄存器	地址	复位值
GIE	全局中断 1 = 使能 (PEIE, TMR1IE 适用) 0 = 全局关闭 (唤醒不受影响)	INTCON[7]	Bank 首地址 +0x0B	RW−0
PEIE	外设总中断 1 = 使能 (TMR1IE 适用) 0 = 关闭 (无唤醒)	INTCON[6]		RW−0
TMR1IE	Timer1 与 PR1 匹配中断 1 = 使能 0 = 关闭 (无唤醒)	PIE[3]	0x8C	RW-0
TMR1IF	Timer1 与 PR1 匹配中断标志位 1 = 匹配 (锁存) 0 = 不匹配	PIR[3]	0x0C	RW0-0

表 10-2 Timer1 中断使能和状态位

名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	复位值
PR1L	0x117	PR1 周期寄存器低 8 位 PR1L[7:0]								1111 1111
PR1H	0x118	—	—	—	—	PR1 周期寄存器高 4 位 PR1H[3:0]				---- 1111
TMR1L	0x119	Timer 1 计数结果寄存器低 8 位 TMR1L[7:0]								0000 0000
TMR1H	0x11A	—	—	—	—	Timer 1 计数结果寄存器高 4 位 TMR1H[3:0]				---- 0000
T1CON0	0x106	—	T1AUSTP	T1ECKE	T1CKPSA	T1CKRUN	T1ON	T1CKSRC[1:0]		-000 0000
MSCON2	0x10D	P0ANP	P0AP	CMAUSTR	STRTYP[1:0]		T1EVTS[2:0]			0000 0000
INTCON	0x0B	GIE	PEIE	T0IE	INTE	PAIE	T0IF	INTF	PAIF	0000 0000
PIE1	0x8C	EEIE	CKMIE	LVDIE	ACMPIE	TMR1IE	OSFIE	TMR2IE	ADCIE	0000 0000
PIR1	0x0C	EEIF	CKMIF	LVDIF	ACMPIF	TMR1IF	OSFIF	TMR2IF	ADCIF	0000 0000

表 10-3 Timer1 相关用户寄存器汇总

10.2. Timer1 工作原理

Timer1 模块的时钟源可选，默认输入是系统指令时钟 (FOSC/2)。该时钟被用于递增 TMR1 寄存器。

{TMR1H, TMR1L} 和{PR1H, PR1L} 的值被不断比较以确定何时匹配。{TMR1H,TMR1L} 将从 00h 开始递增直到与{PR1H, PR1L} 的值相同。匹配时将发生以下两种情况：

1. {TMR1H, TMR1L} 在下一递增周期复位为 00h
2. Timer1/PR1 比较器的匹配输出用于将 PIR1 寄存器的 TMR1IF 中断标志置 1。

{TMR1H, TMR1L} 和{PR1H , PR1L} 都是可读写寄存器。在复位时，他们的值分别是 0 和 0xFFF。将 T1CON 寄存器中的 T1ON 位置 1 可打开 Timer1，反之将 T1ON 位清零关闭 Timer1。

10.3. Timer1 计数值的读写

具体读写操作步骤请参照 **11.4 小节**的 Timer2 计数值的读写。

10.4. Timer1 的自动关闭

当 T1AUSTP 设置为 1 时，T1ON 位将受硬件事件的控制 (这时软件对 T1ON 仍然可读写)，这些硬件事件包括 LVD 中断，CM0 中断，CM1 中断，或者它们之间的组合，由 T1EVTS 位控制，当发生所选择的事件时，T1ON 被清 0。需要提出的是，清 0 动作是边沿而不是电平，换言之即使相关事件中断标志一直存在，T1ON 也可以重新被软件置 1。

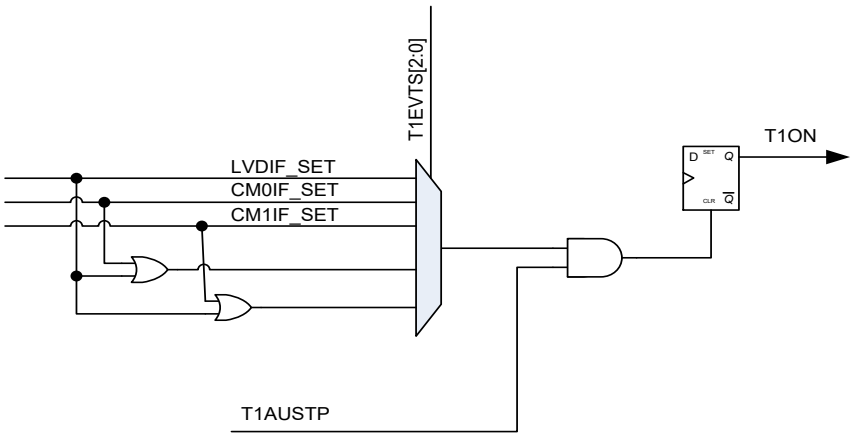


图 10-2 Timer1 自动关闭框图

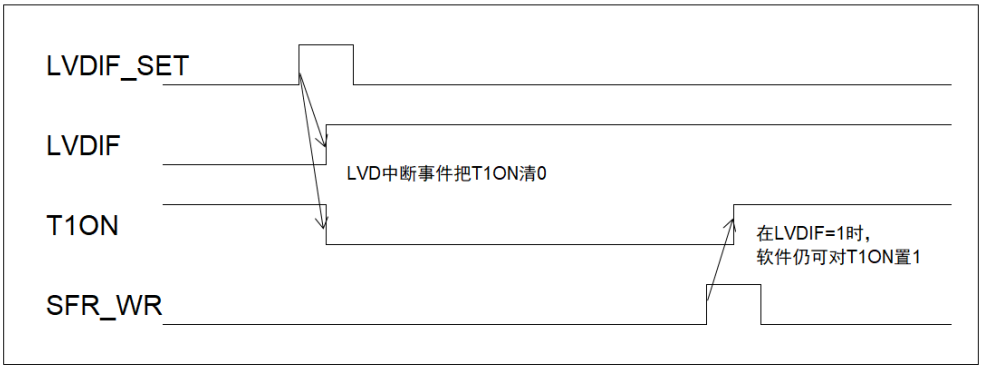


图 10-3 Timer1 的自动清 0 时序

11. 定时器 2

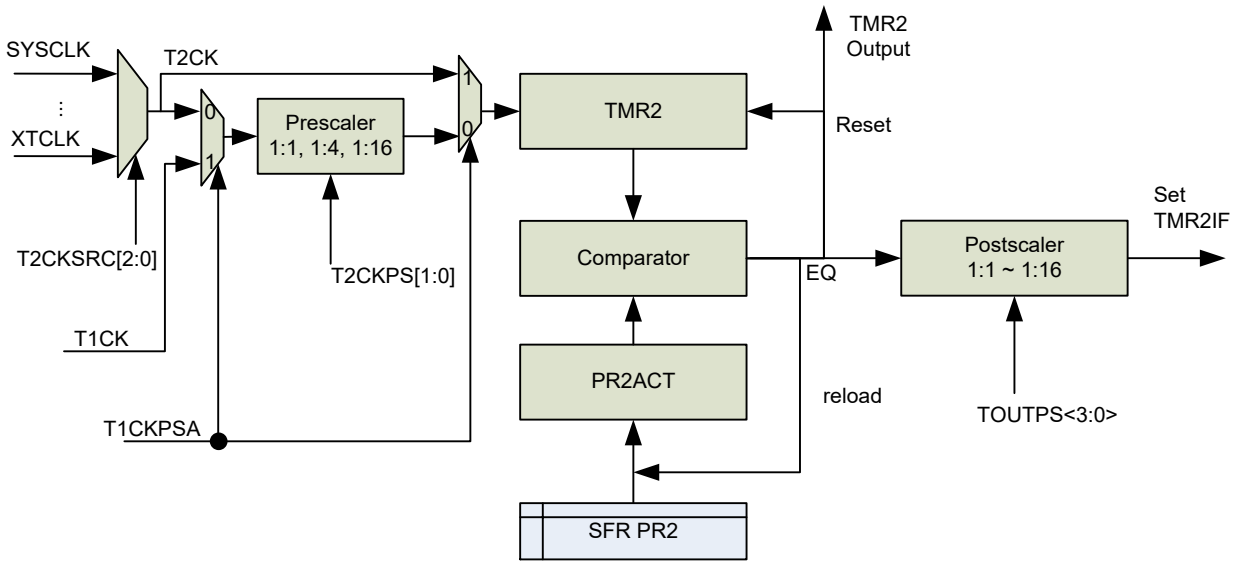


图 11-1 定时器 2 框图

定时器 2 为 16 位定时器，包含以下功能：

- 16 位计数寄存器
- 16 位周期寄存器，双缓冲
- TMR2 值等同 PR2 时产生中断
- 1:1，1:4，1:16 预分频比（Timer1 复用同一预分频器）
- 1:1–1:16 后分频比
- 时钟源可选：系统时钟或者内部 32MHz 时钟（由晶体的 2 倍频得到）或 LIRC

11.1. Timer2 相关寄存器汇总

名称	状态		寄存器	地址	复位值
T2CKRUN	睡眠时 T2CK 保持运行	1 = Yes (时钟源非指令时钟) 0 = No	MSCON0[0]	0x10C	RW-0
T1CKPSA	预分频器分配位	1 = Timer1 0 = Timer2	T1CON0[4]	0x106	RW-0
PR2U	更新周期和占空比的即时生效控制位 1 = PR2/P1xDTy 缓冲值立即分别更新到 PR2ACT 和 P1xDTyACT 0 = 周期结束后正常更新		T2CON0[7]	0x1E	RW1-0
TOUTPS	Timer2 输出后分频比 0000 = 1:1 0100 = 1:5 1000 = 1:9 1100 = 1:13 0001 = 1:2 0101 = 1:6 1001 = 1:10 1101 = 1:14 0010 = 1:3 0110 = 1:7 1010 = 1:11 1110 = 1:15 0011 = 1:4 0111 = 1:8 1011 = 1:12 1111 = 1:16		T2CON0[6:3]		RW-0000
TMR2ON	Timer2 (PWM 单脉冲模式下自动清 0)	1 = 使能 0 = 关闭	T2CON0[2]		RW-0
T2CKPS (T1CKPS)	Timer2/Timer1 预分频比 00 = 1 01 = 4 1x = 16		T2CON0[1:0]		RW-00
T2CKSRC	Timer2 时钟源选择 000 = 指令时钟 100 = HIRC 001 = 2x 指令时钟 101 = LIRC 010 = 2x HIRC 110 = LP (*) 011 = 2x LP, XT or EC (*) 111 = XT (*) (*) FOSC 应相应配置成 LP/XT/EC 模式或选择 INTOSCIO 模式，否则振荡器将不会运行		T2CON1[2:0]	0x1F	RW-000
TMR2L	Timer 2 计数结果寄存器低 8 位		TMR2L[7:0]	0x15	RW-0000 0000
TMR2H	Timer 2 计数结果寄存器高 8 位		TMR2H[7:0]	0x16	RW-0000 0000
PR2L	PR2 周期寄存器低 8 位		PR2L[7:0]	0x17	RW-1111 1111
PR2H	PR2 周期寄存器高 8 位		PR2H[7:0]	0x18	RW-1111 1111

表 11-1 Timer2 相关用户控制寄存器

名称	状态	寄存器	地址	复位值
GIE	全局中断 1 = 使能 (PEIE, TMR2IE 适用) 0 = 全局关闭 (唤醒不受影响)	INTCON[7]	Bank 首地址 +0x0B	RW−0
PEIE	外设总中断 1 = 使能 (TMR2IE 适用) 0 = 关闭 (无唤醒)	INTCON[6]		RW−0
TMR2IE	Timer2 与 PR2 匹配中断 1 = 使能 0 = 关闭	PIE1[1]	0x8C	RW-0
TMR2IF	Timer2 与 PR2 匹配中断标志位 1 = 匹配 (锁存) 0 = 不匹配	PIR1[1]	0x0C	RW0-0

表 11-2 Timer2 中断使能和状态位

名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	复位值
TMR2L	0x15	Timer 2 计数结果寄存器低 8 位 TMR2L[7:0]								0000 0000
TMR2H	0x16	Timer 2 计数结果寄存器高 8 位 TMR2H[7:0]								0000 0000
PR2L	0x17	PR2 周期寄存器低 8 位 PR2L[7:0]								1111 1111
PR2H	0x18	PR2 周期寄存器高 8 位 PR2H[7:0]								1111 1111
INTCON	0x0B	GIE	PEIE	T0IE	INTE	PAIE	T0IF	INTF	PAIF	0000 0000
PIE1	0x8C	EEIE	CKMIE	LVDIE	ACMPIE	TMR1IE	OSFIE	TMR2IE	ADCIE	0000 0000
PIR1	0x0C	EEIF	CKMIF	LVDIF	ACMPIF	TMR1IF	OSFIF	TMR2IF	ADCIF	0000 0000
MSCON0	0x10C	—	—	ROMLPE	CLKOS	SLVREN	CKMAVG	CKCNTI	T2CKRUN	--01 0000
T2CON0	0x1E	PR2U	TOUTPS[3:0]				TMR2ON	T2CKPS[1:0]		0000 0000
T2CON1	0x1F	—			P1OS	P1BZM	T2CKSRC [2:0]			---0 0000

表 11-3 Timer2 相关用户寄存器汇总

11.2. Timer2 工作原理

在非 PWM 模式下，Timer2 模块的时钟输入是指令时钟。该时钟送入 Timer2 预分频器，其预分频比有 1:1、1:4 或 1:16 三种选择。随后预分频器的输出被用于递增 TMR2 寄存器。

TMR2 和 PR2 的值被不断比较以确定何时匹配。TMR2 将从 00h 开始递增直到与 PR2 的值相同。匹配时将发生以下两种事件：

- TMR2 在下一递增周期复位为 00h
- Timer2 后分频比递增

Timer2/PR2 比较器的匹配输出送入 Timer2 后分频器。后分频器的选项范围为 1:1 至 1:16。Timer2 后分频器的输出用于将 PIR1 寄存器的 TMR2IF 中断标志置 1。

TMR2 和 PR2 都是可读写寄存器。在复位时，他们的值分别是 0 和 0xFFFF。

将 T2CON0 寄存器中的 TMR2ON 位置 1 可打开 Timer2，反之将 TMR2ON 位清零关闭 Timer2。

Tmer2 预分频器由 T2CON0 寄存器的 T2CKPS 位控制。

Timer2 后分频器由 T2CON0 寄存器的 TOUTPS 位控制。

预分频和后分频计数器会在写以下寄存器时清零：

- 写 TMR2
- 写 T2CON0
- 任何 reset 动作

注：

1. 写 T2CON0 并不会清零 TMR2 寄存器；
2. Timer2 的时钟源由 T2CKSRC 控制，当 T2CKSRC≠000 时，如果 T2CKRUN 设置为 1，则 Timer2 的时钟在睡眠状态下继续运行。

11.3. 关于 PR2 的更新

Timer2 的周期寄存器 PR2 具有双缓冲结构，在模块内部分别是 PR2ACT，PR2。PR2ACT 是活动寄存器，也就是 TMR2 要比较的寄存器。正常情况下，只有在 Timer2 发生匹配事件时 PR2ACT 才会被更新为 PR2 的内容。

软件也可以在写完 PR2 寄存器对 PR2ACT 进行更新，而无需等待匹配事件，方法是向 PR2U 位写 1。

注意：PR2ACT 对软件不可见。

11.4. Timer2 计数值的读写

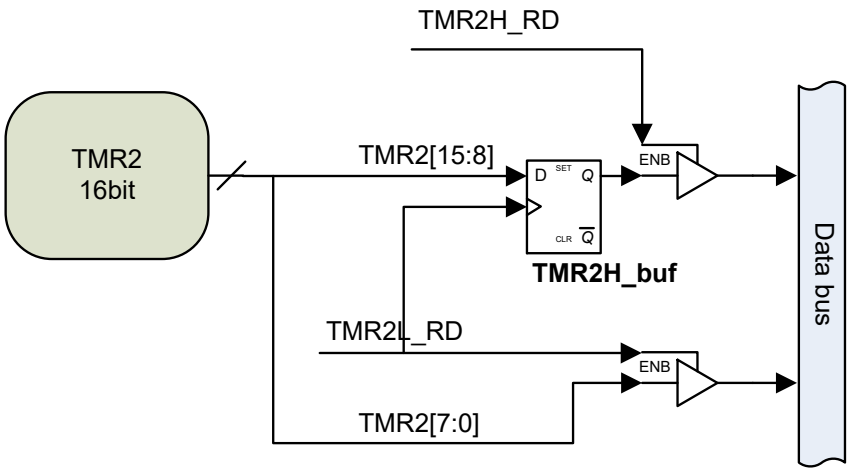


图 11-2 Timer2 计数值的读操作原理框图

Timer2 是 16bit 的定时器，由于内部数据总线是 8bit 的限制，软件读 Timer2 的计数值需要分开两次读，计数值的低 8 位 TMR2L 可直接访问，高 8 位有一个内部缓存 TMR2H_buf，该缓存在软件读 TMR2L 的时刻被更新。这种机制保证了即使 Timer2 在计数期间，软件总是可以读到一个完整的 16bit 计数值，避免了诸如在两次读间隔内，Timer2 发生了溢出等情况。

综上所述，读操作应按照下面的顺序进行：

- 读 TMR2L；
- 读 TMR2H；

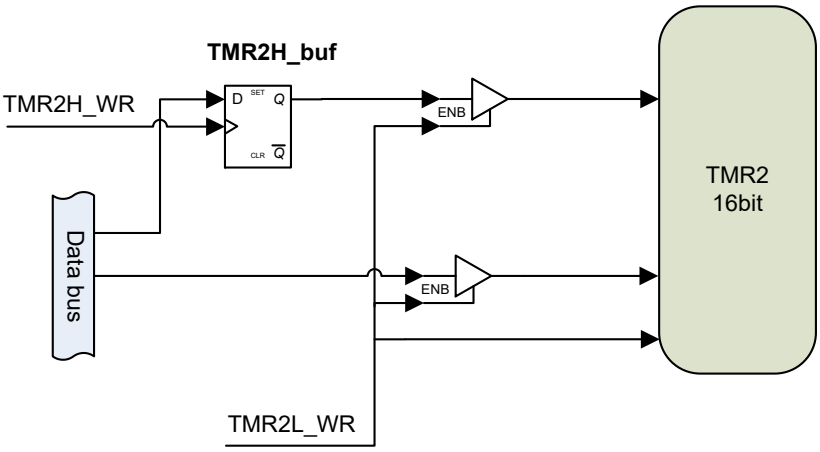


图 11-3 Timer2 计数值的写操作原理框图

和读操作相似，软件对 TMR2H 寄存器的写并不会立即更新内部计数值，而是先写到缓存 TMR2H_buf 中，当软件写 TMR2L 时，硬件自动把缓存的高 8 位一起更新到计数值当中。

写顺序：

- 写 TMR2H；
- 写 TMR2L；

注意：当 Timer2 工作在异步时钟时，建议先清除 TMR2ON 位以停止计数，然后至少等 1 个计数时钟后再发起对 TMR2 的读。

另外，对于写操作，建议用户直接停止计数器，然后写入所期望的值。如果寄存器正进行递增计数，对定时器寄存器进行写操作，可能会导致写入竞争，从而可能在 TMR2H:TMR2L 这对寄存器中产生不可预测的值。

11.5. Timer2 时钟源

Timer2 支持以下不同的时钟源：

- 指令时钟
- 系统时钟
- HIRC 的 2 倍频
- 晶体/外部时钟 2 倍频 (只有当 FOSC 配置为 LP/XT 或 EC 模式时才有效)
- HIRC
- LIRC
- LP 晶体时钟 (只有当 FOSC 配置为 INTOSCIO，或 LP 模式时才有效)
- XT 晶体时钟 (只有当 FOSC 配置为 INTOSCIO，或 XT 模式时才有效)

12. PWM 模块

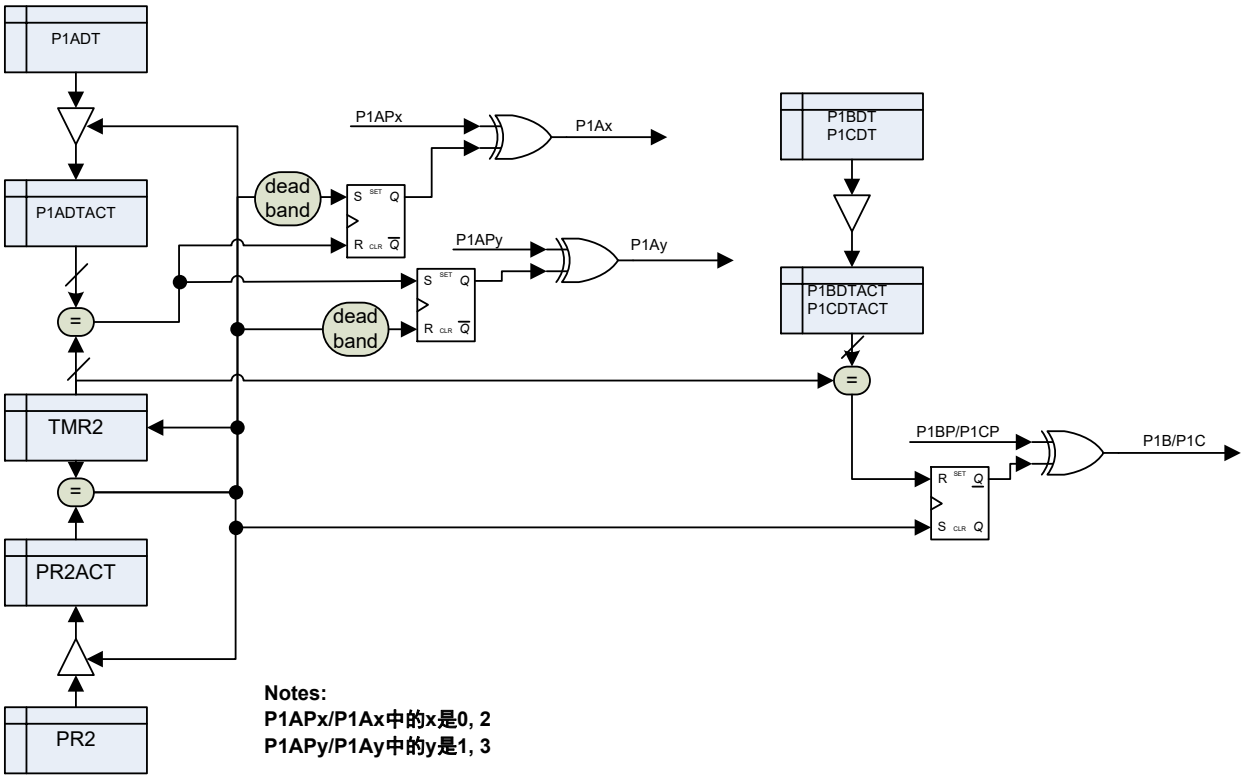


图 12-1 PWM 结构框图

PWM 支持以下特性：

- 16bit 的分辨率
- 周期和占空比匹配双缓冲设计
- 1 路带死区控制的 PWM 输出：P1A
- 4 路独立占空比的 PWM 输出：P1A，P1B，P1C，P1D
- 每路 PWM 输出极性可独立设置
- 故障刹车以及自动重启

12.1. PWM 相关寄存器汇总

名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	复位值
P1ADTL	0x0D	P1A 占空比低 8 位								0000 0000
P1ADTH	0x0E	P1A 占空比高 8 位								0000 0000
P1BDTL	0x0F	P1B 占空比低 8 位								0000 0000
P1BDTH	0x10	P1B 占空比高 8 位								0000 0000
P1CDTL	0x11	P1C 占空比低 8 位								0000 0000
P1CDTH	0x12	P1C 占空比高 8 位								0000 0000
P1DDTL	0x13	P1D 占空比低 8 位								0000 0000
P1DDTH	0x14	P1D 占空比高 8 位								0000 0000
TMR2L	0x15	Timer2 计数结果低 8 位								0000 0000
TMR2H	0x16	Timer2 计数结果高 8 位								0000 0000
PR2L	0x17	PR2 周期低 8 位								1111 1111
PR2H	0x18	PR2 周期低 8 位								1111 1111
P1BR0	0x19	P1BEVT	P1BKS			P1BSS	P1ASS			0000 0000
P1BR1	0x1A	P1D2SS		P1DSS		P1C2SS		P1CSS		0000 0000
P1POL0	0x1B	P1C0P	P1B0P	P1A2NP	P1A2P	P1A1NP	P1A1P	P1A0NP	P1A0P	0000 0000
P1POL1	0x1C	P1D2P	P1D1P	P1D0P	—	—	P1C1P	P1B1P	—	000- -00-
P1CON	0x1D	P1AUE	P1DC							0000 0000
T2CON0	0x1E	PR2U	TOUTPS				TMR2ON	T2CKPS(T1CKPS)		0000 0000
T2CON1	0x1F	—			P1OS	P1BZM	T2CKSRC			---0 0000
P1AUX	0x8D	P1BKM	P1CDM	P1B2SS[1:0]		P1CF2E	P1CF2	P1DF2E	P1DF2	0000 0000
P1OE1	0x9F	P1D2OE	P1D1OE	P1D0OE	—	—	P1C1OE	P1B1OE	—	000- -00-
P1OE0	0x9E	P1C0OE	P1B0OE	P1A2NOE	P1A2OE	P1A1NOE	P1A1OE	P1A0NOE	P1A0OE	0000 0000

表 12-1 PWM 相关用户寄存器汇总

	Timer2 周期		Timer2 计数器		占空比		死区时间
	MSB	LSB	MSB	LSB	MSB	LSB	
PWM1	PR2H	PR2L	TMR2H	TMR2L	P1ADTH	P1ADTL	P1DC
/PWM1							
PWM2					P1BDTH	P1BDTL	–
PWM3					P1CDTH	P1CDTL	–
PWM4					P1DDTH	P1DDTL	–

名称	状态	寄存器	地址	复位值
TMR2L	Timer 2 计数结果低 8 位	TMR2L[7:0]	0x15	RW-0000 0000
TMR2H	Timer 2 计数结果高 8 位	TMR2H[7:0]	0x16	RW-0000 0000
PR2L	PR2 周期低 8 位	PR2L[7:0]	0x17	RW-1111 1111
PR2H	PR2 周期高 8 位	PR2H[7:0]	0x18	RW-1111 1111
P1ADTL	P1A 占空比低 8 位	P1ADTL[7:0]	0x0D	RW-0000 0000
P1ADTH	P1A 占空比高 8 位	P1ADTH[7:0]	0x0E	RW-0000 0000
P1BDTL	P1B 占空比低 8 位	P1BDTL[7:0]	0x0F	RW-0000 0000
P1BDTH	P1B 占空比高 8 位	P1BDTH[7:0]	0x10	RW-0000 0000
P1CDTL	P1C 占空比低 8 位	P1CDTL[7:0]	0x11	RW-0000 0000
P1CDTH	P1C 占空比高 8 位	P1CDTH[7:0]	0x12	RW-0000 0000
P1CDTL	P1D 占空比低 8 位	P1CDTL[7:0]	0x11	RW-0000 0000
P1CDTH	P1D 占空比高 8 位	P1CDTH[7:0]	0x12	RW-0000 0000
P1DC	PWM1 死区时间	P1CON[6:0]	0x1D	RW-000 0000

表 12-2 PWM 时序设置

	通道分配			输出使能			极性		
	Ch0	Ch1	Ch2	Ch0	Ch1	Ch2	Ch0	Ch1	Ch2
PWM1	PA4	PB1	PC2	P1A0OE	P1A1OE	P1A2OE	P1A0P	P1A1P	P1A2P
/PWM1	PA3	PC5	PC1	P1A0NOE	P1A1NOE	P1A2NOE	P1A0NP	P1A1NP	P1A2NP
PWM2	PA5	PB6	–	P1B0OE	P1B1OE	–	P1B0P	P1B1P	–
PWM3	PA6	PC6	–	P1C0OE	P1C1OE	–	P1C0P	P1C1P	–
PWM4	PB0	PC7	PA7	P1D0OE	P1D1OE	P1D2OE	P1D0P	P1D1P	P1D2P

名称	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	地址	复位值 (RW)
P1OE0	P1C0OE	P1B0OE	P1A2NOE	P1A2OE	P1A1NOE	P1A1OE	PA10NOE	P1A0OE	0x9E	0000 0000
P1OE1	P1D2OE	P1D1OE	P1D0OE	–	–	P1C1OE	P1B1OE	–	0x9F	000- -00-
P1POL0	P1C0P	P1B0P	P1A2NP	P1A2P	P1A1NP	P1A1P	P1A0NP	P1A0P	0x1B	0000 0000
P1POL1	P1D2P	P1D1P	P1D0P	–	–	P1C1P	P1B1P	–	0x1C	000- -00-

表 12-3 PWM 输出极性 (1 = 反向，0 = 正常) 和输出使能位 (1 = 使能， 0 = 关闭)

	关闭 = 0，使能 = 1			异或 (XOR) = 0，同或 (XNOR) = 1			第 2 功能
	Ch0	Ch1	Ch2	Ch0	Ch1	Ch2	
PWM1	–	–	–	–	–	–	N/A
/PWM1	–	–	–	–	–	–	N/A
PWM2	–	–	–	–	–	–	N/A
PWM3	–	P1CF2E	–	–	P1CF2	–	P1C xor/xnor P1D
PWM4	–	P1DF2E		–	P1DF2		P1B xor/xnor P1C

名称	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	地址	复位值 (RW)
P1AUX	P1BKM	P1CDM	P1B2SS[1:0]		P1CF2E	P1CF2	P1DF2E	P1DF2	0x8D	0000 0000

表 12-4 PWM 第 2 功能

名称	状态	寄存器	地址	复位值
PR2U	更新周期和占空比的即时生效控制位 1 = PR2/P1xDTy 缓冲值立即分别更新到 PR2ACT 和 P1xDTyACT 0 = 周期结束后正常更新	T2CON0[7]	0x1E	RW1-0
P1OS	1 = 单脉冲(One pulse) 模式 0 = 正常连续模式	T2CON1[4]	0x1F	RW-0
P1BZM	1 = 蜂鸣器(Buzzer) 模式，50%占空比 0 = 正常 PWM 模式	T2CON1[3]		RW-0

表 12-5 4 路 PWM 通道的通用功能控制

名称	状态	寄存器	地址	复位值
P1BKS	PWM1 故障源 000: 禁止故障刹车功能 001: BKIN= 0 010: BKIN=1 011: C0OUT=1 100: C1OUT=1 101: C0OUT=1 or C1OUT=1 110: ADC 阈值比较标志位为'1' 111: BKIN= 0 or C0OUT=1 or C1OUT=1 or ADC 阈值比较标志位为'1'	P1BR0[6:4]	0x19	RW-000
P1BEVT	PWM1 发生故障事件标志位 1 = Yes (锁存，直至被清零) 0 = No	P1BR0[7]		RW-0
P1AUE	PWM1 自动重启 1 = 当故障条件被清除时，P1BEVT 自动清零，PWM1 自动重启 0 = 当故障条件被清除时，P1BEVT 由指令清零，PWM1 重启	P1CON[7]	0x1D	RW-0
P1CDM	故障刹车时，P1C/P1D 输出控制 1 = 保持输出 0 = 由 P1BR1 寄存器控制	P1AUX[6]	0x8D	RW-0
P1BKM	故障刹车时，Timer2 及其预分频 1 = 不受影响 0 = 停止	P1AUX[7]		RW-0

表 12-6 PWM 故障刹车相关寄存器

名称	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	地址	复位值 (RW)
P1BR0	P1BEVT	P1BKS[2:0]			P1BSS[1:0]		P1ASS[1:0]		0x19	0000 0000
P1BR1	P1D2SS[1:0]		P1DSS[1:0]		P1C2SS[1:0]		P1CSS[1:0]		0x1A	0000 0000
P1AUX	P1BKM	P1CDM	P1B2SS[1:0]		P1CF2E	P1CF2	P1DF2E	P1DF2	0x8D	0000 0000

	故障下，引脚输出状态			备注	
	Ch0	Ch1	Ch2	< G 版芯片	≥ G 版芯片 or (1)
PWM1	P1ASS			00 = 高阻	00 = 高阻
/PWM1				01 = 逻辑“0”	01 = 0
PWM2	P1BSS	P1B2SS (1)	–	1x = 逻辑“1”	1x = 1
PWM3	P1CSS	P1C2SS (1)	–	如 *p1xxp* = 0，逻辑“0” = 0	
PWM4	P1DSS	P1D2SS (1)		如 *p1xxp* = 1，逻辑“0” = 1	

表 12-7 PWM 发生故障时的输出状态

名称	状态		寄存器	地址	复位值
P1C0OE	P1C0 输出到 PA6	1 = 使能 0 = 禁止	P1OE0[7]	0x9E	RW-0
P1B0OE	P1B0 输出到 PA5		P1OE0[6]		RW-0
P1A2NOE	P1A2N 输出到 PC1		P1OE0[5]		RW-0
P1A2OE	P1A2 输出到 PC2		P1OE0[4]		RW-0
P1A1NOE	P1A1N 输出到 PC5		P1OE0[3]		RW-0
P1A1OE	P1A1 输出到 PB1		P1OE0[2]		RW-0
P1A0NOE	P1A0N 输出到 PA3		P1OE0[1]		RW-0
P1A0OE	P1A0 输出到 PA4		P1OE0[0]		RW-0
P1D2OE	P1D2 输出到 PA7		P1OE1[7]	0x9F	RW-0
P1D1OE	P1D1 输出到 PC7		P1OE1[6]		RW-0
P1D0OE	P1D0 输出到 PB0		P1OE1[5]		RW-0
P1C1OE	P1C1 输出到 PC6		P1OE1[2]		RW-0
P1B1OE	P1B1 输出到 PB6		P1OE1[1]		RW-0

表 12-8 PWM 用户相关 IO 配置寄存器

12.2. 周期

PWM 周期由 Timer2 的 PR2 寄存器指定。用 **PWM 周期 = (PR2 + 1) * TT2CK * (TMR2 预分频值)**……公式 12.1 可计算 PWM1 周期。

PWM 周期 = (PR2 + 1) * TT2CK * (TMR2 预分频值)……公式 12.1

当 TMR2 等于 PR2 时，下一次递增周期将发生以下三个事件：

- TMR2 被清零
- P1A0, [P1A0], P1B, P1C, P1D 置 1 (4 路 PWM 都是高有效的情况下)
- 内部的周期寄存器 PR2ACT 和占空比寄存器 P1xDTACT 被更新

12.3. 占空比

通过对以下几个寄存器写入 16 位值可指定 PWM1 占空比：

P1xDTL(x= A, B, C, D)
P1xDTH(x= A, B, C, D)

其中，P1xDTH 保存的是 4 路 PWM 占空比寄存器的 8 位 MSb，P1xDTL 则是低 8 位。由于内部的双缓冲设计，占空比寄存器在任何时候被写入，它在软件更新占空比时 PWM 的不会产生毛刺起重要作用。

脉冲宽度 = P1xDT * T_{T2CK} * (TMR2 预分频值) ……公式 12.2

占空比 = P1xDT ÷ (PR2+1) …… ……公式 12.3

12.4. 时钟源选择

PWM 使用的时基定时器为 Timer2，Timer2 的时钟源有以下选择：

- 系统时钟
- 指令时钟 (即系统时钟的 2 分频或 4 分频)
- HIRC 的 2 倍频
- 外部时钟 2 倍频 (只有当 FOSC 配置为 EC 模式时才有效)
- HIRC
- LIRC

12.5. 睡眠下 PWM 状态

当 T2CKRUN=1 而且 Timer2 的时钟源不是选择指令时钟时，MCU 进入睡眠后，PWM 可以保持在运行状态，由 T2CKSRC 所选择的时钟源不会关闭。否则，Timer2 将停止计数，而 PWM 各管脚电平保留在执行完 SLEEP 指令后的状态。

12.6. P1A 的死区时间

有 1 路 PWM 带互补输出，P1Ax。其中 P1A0/[P1A1]/[P1A2]定义为正输出，[P1A0N]/[P1A1N]/[P1A2N]为互补输出。P1A 的 PWM 带有死区插入功能，其死区时间由 P1DC[6:0]控制。死区定时器以 Timer2 时钟作为作为计数时钟源。
注意：{P1A0,[P1A0N]}，{[P1A1], [P1A1N]}, {[P1A2], [P1A2N]}这 3 对互补输出共享同一死区设置。

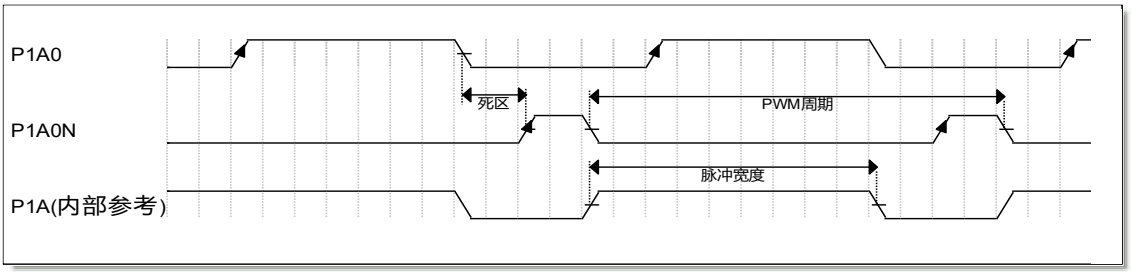


图 12-2 死区 PWM 示意图

12.7. 故障刹车

PWM 模块支持故障刹车模式，它会在发生外部刹车事件时禁止 PWM 输出，同时 Timer2 及预分频器处于复位状态。刹车模式会将 PWM1 输出引脚置于预定状态，该模块用于防止发生故障条件时 PWM 损坏应用。

使用 P1BR0 寄存器的 P1BKS 位可选择故障源，故障事件可以是以下几种：

- BKIN 管脚为低电平
- BKIN 管脚为高电平
- C0OUT=1
- C1OUT=1
- C0OUT=1 或者 C1OUT=1
- ADC 阈值比较为 1

刹车状态由 P1BR0 寄存器的 P1BEVT 位指示。如果该位为 0，PWM1 引脚正常工作。如果该位为 1，PWM 输出处于预设状态，由 P1xSS 决定。

12.7.1. 刹车状态

发生故障时，故障下的 PWM1 管脚电平状态由寄存器 P1xSS 位选择，有以下几种：
< G 版芯片：

- PWM 置于有效电平
- PWM 置于无效电平
- PWM 关闭，处于悬空高阻状态
- TMR2 定时器和预分频停止计数
- TMR2ON 位不受影响

有效电平由 P1POL 各寄存器位决定。

≥ G 版芯片：

- PWM 输出 0
- PWM 输出 1
- PWM 关闭，处于悬空高阻状态
- TMR2 定时器和预分频停止计数
- TMR2ON 位不受影响

12.7.2. 故障清除

故障刹车条件是基于电平的信号，而非基于边沿的信号。只要故障条件有效，就故障状态一直保持，软件

不能清除；只有当相关故障输入或 LVD 事件消除了，P1BEVT 才可能被清 0。

12.7.3. 自动重启

PWM 可配置为在故障条件被清除时自动重启 PWM1 信号。通过将 P1CON 寄存器中的 P1AUE 位置 1 使能自动重启。

自动重启使能时，只要故障条件有效，P1BEVT 标志位（下图中的 P1BEVT_FLAG 信号）就保持置 1。当故障条件被清除时，P1BEVT 标志位（下图中的 P1BEVT_FLAG 信号）将被硬件清零，TMR2 恢复计数，在下一次计数溢出时，实际控制信号 P1BEVT 被清除，PWM1 恢复正常输出。

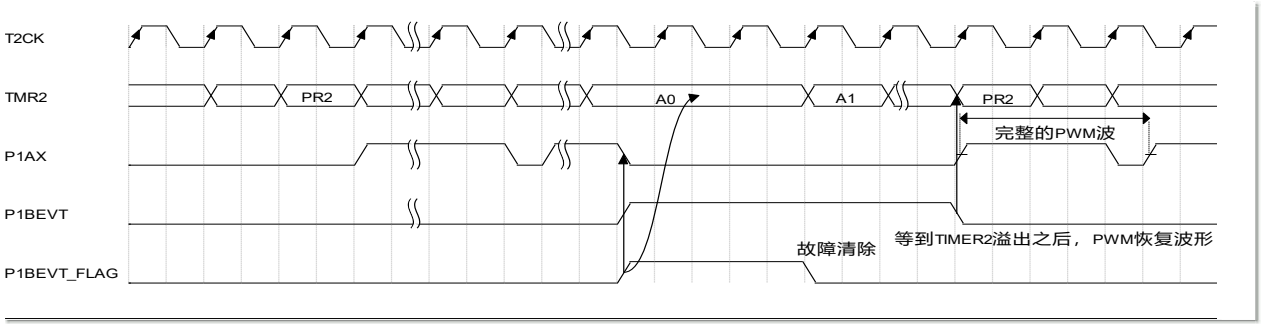


图 12-3 PWM 的自动重启时序图

12.7.4. 前沿消隐

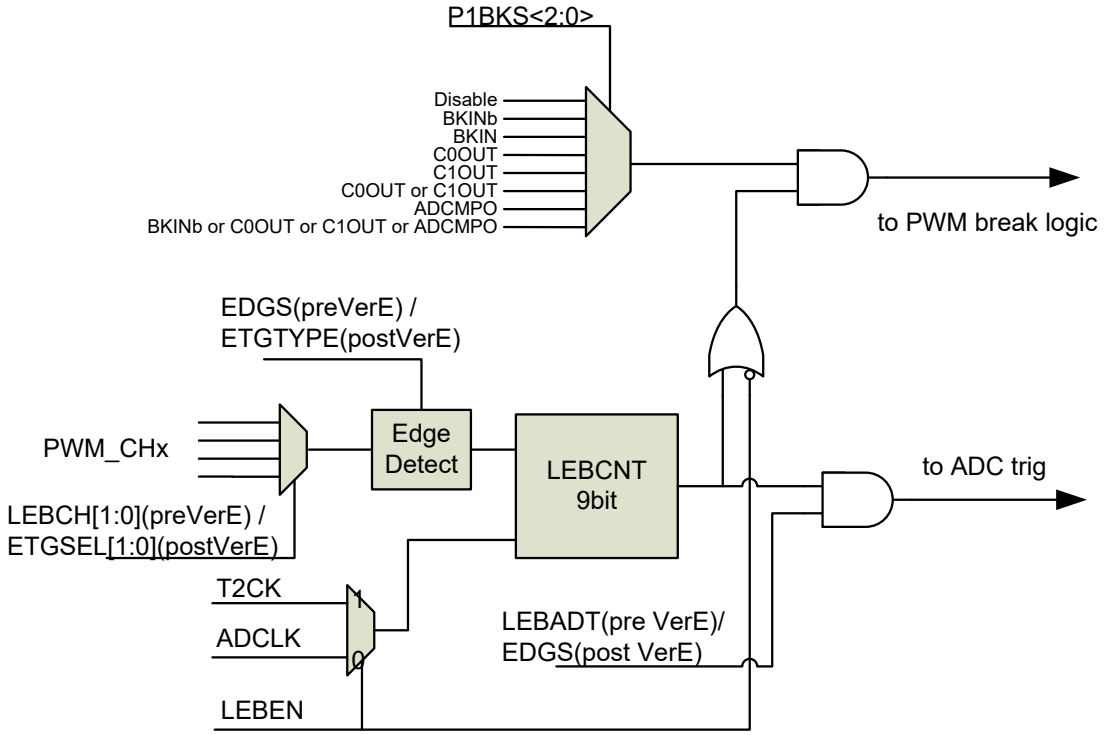


图 12-4 前沿消隐原理框图

在高速开关应用中，开关管（如 MOSFET/IGBT）通常会产生非常大的电压瞬变，这些电压瞬变会导致有问题的测量误差。利用前沿消隐（LEB）功能，应用程序可以忽略 PWM 输出信号边沿附近的

MOSFET/IGBT 开关导致的预期电压瞬变。

LEBCH (≥G 版改为 ETGSEL) 用于选择要被消隐的 PWM 通道，EDGS (≥G 版改为 ETGTYP) 选择 PWM 信号的边沿类型。当 LEBEN 为 1，选择的 PWM 信号边沿将触发 LEB 定时器计数，计数时钟源为 Timer2 时钟。直到计数值等于 LEBPR，LEB 定时器停止计数，这段时间为消隐周期，期间所发生的刹车事件将被忽略。在消隐周期内如果再次发生有效的 PWM 边沿，则 LEB 定时器将清 0，重新开始计数。

注意：LEB 定时器和 ADC 延时定时器复用了同一个 9bit 计数器，当 LEBEN 为 1 时，原 ADC 的延时触发功能被禁止，但如果 LEBADT 为 1 (≥G 版为 EDGS=1)，LEB 定时器溢出将触发一次 AD 转换。

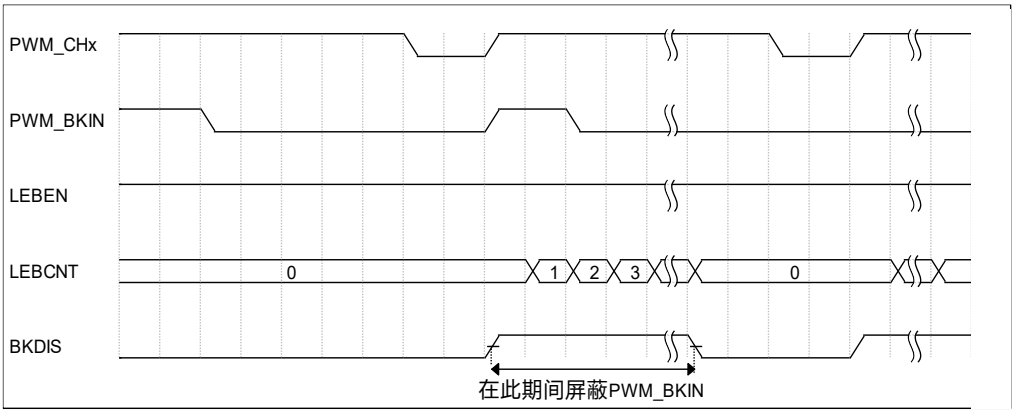


图 12-5 前沿消隐时序示意图

12.8. 周期和占空比寄存器的更新

在 Timer2 已经开启的情况下，周期和各占空比寄存器的更新需要 TMR2 和 PR2 的匹配事件，如果用户不想等待，可以通过写 PR2U 位来立即更新。

当 TMR2ON 为 0 时，软件对 PR2，{P1xDTH, P1xDTL}寄存器的写会马上更新到对应的工作寄存器，此时 PWM 输出保持旧值，不会因为 PR2 或 P1xDT 寄存器变化而变化。

注意：工作寄存器 xxxACT 对软件不可见，软件只能读 PR2 和 P1xDTL, P1xDTH。

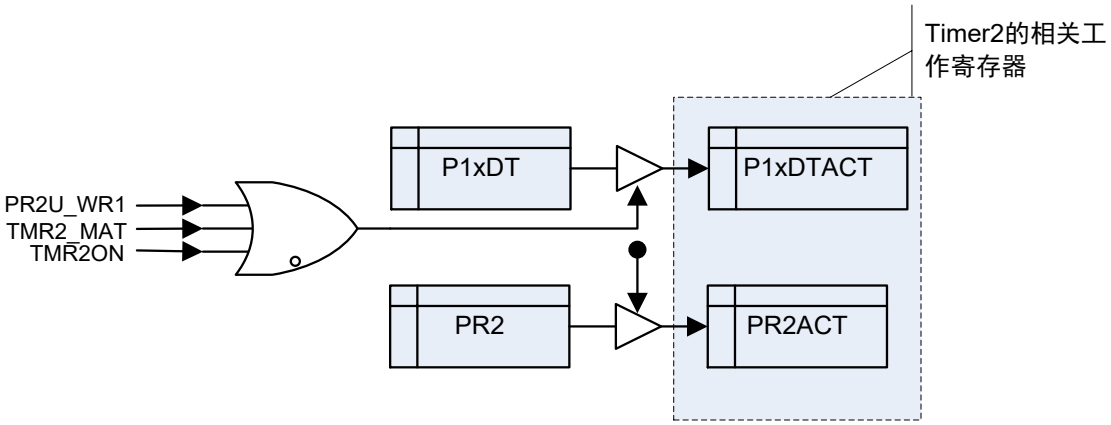


图 12-6 T2 工作寄存器的更新

虽然周期和占空比的双缓冲在很大程度保证 PWM1 输出不会产生毛刺，但如果软件非常靠近 TMR2 匹配时刻去写这此寄存器，特别是在 T2 时钟频率比系统时钟频率快的情况下，则有可能出现不可预料的情况，导致工作寄存器组的值不是期望值，见下**图 12-7**。

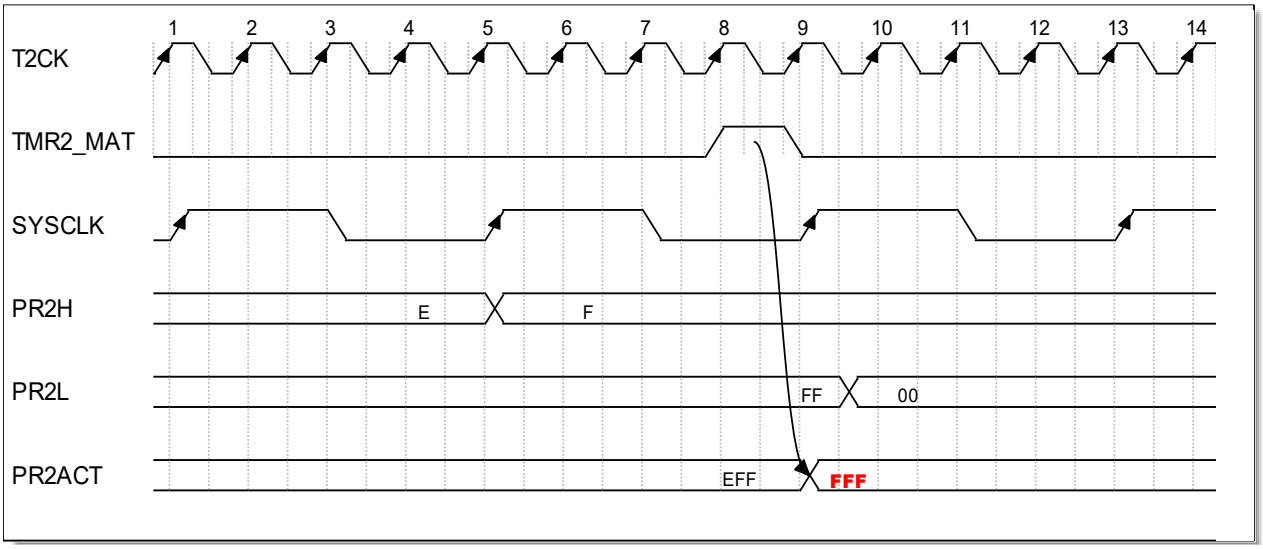


图 12-7 PR2ACT 被更新为意外值 FFF (期望值是 F00)

所以强烈建议更新 PR2 和 xxxDT 只在 TMR2 匹配中断里面做。

12.9. 蜂鸣器模式 (Buzzer)

当 T2CON1.P1BZM 设置为 1 时，PWM1 模式将作为蜂鸣器输出，在这种模式下，占空比设置寄存器不起作用，P1Ax，P1B，P1C 将输出周期为 (2 * (PR2+1) * TT2CK * TMR2 预分频) 的方波。

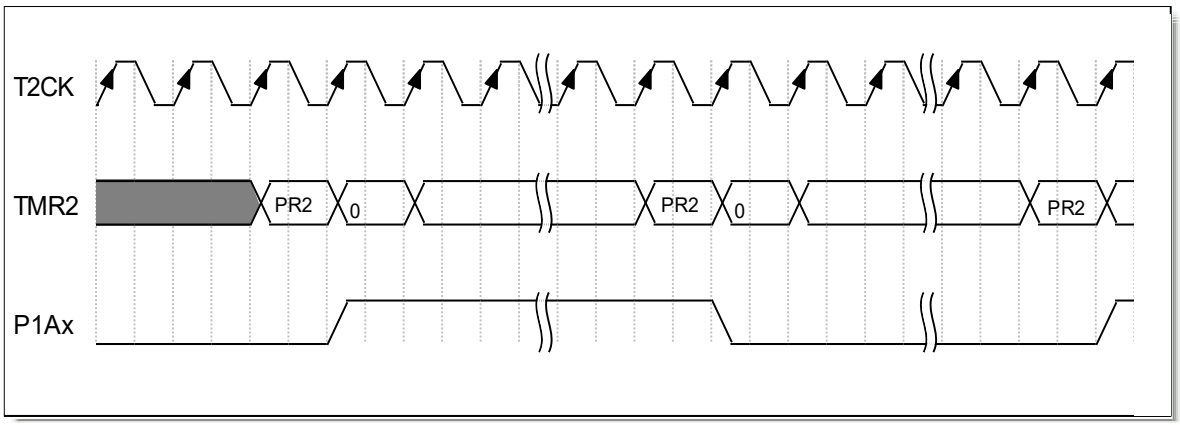


图 12-8 蜂鸣器模式输出

12.10. 单脉冲输出

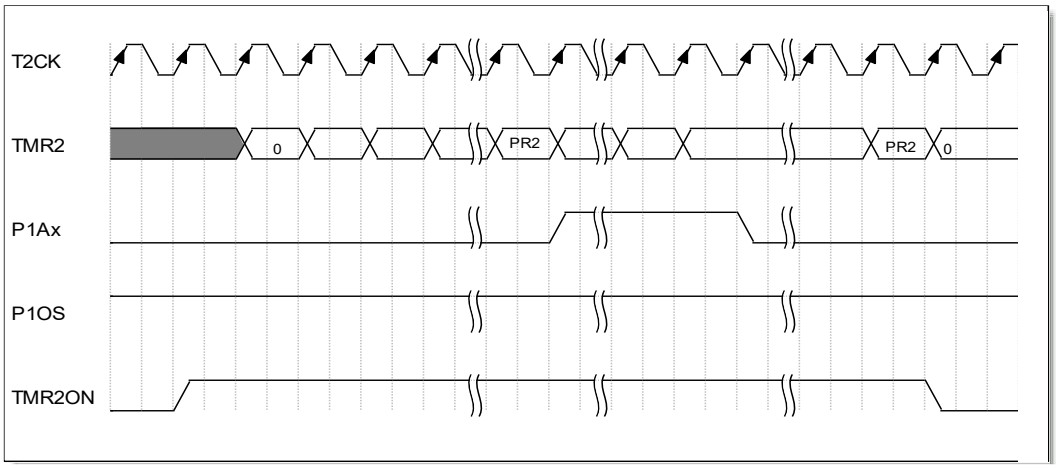


图 12-9 单脉冲模式输出

当 P1OS 设置为 1 时，PWM1 就处于单脉冲输出模式。在该模式下，首次 TMR2 和 (PR2+1) 的匹配将使 P1Ax，P1B，P1C，P1D 输出 PWM 脉冲，在下一次匹配时，TMR2ON 被硬件关闭,且相应管脚的 P1XOE 使能也将被关闭。

12.11. PWM1 输出重映射

P1A、P1B、P1C 和 P1D 这 4 路 PWM 可以分别映射到不同的 I/O，由寄存器 P1OE0 和 P1OE1 控制，它们的复位值为 0，具体映射的 I/O 可以在本文档第 1 章的各脚位图及脚位描述找到。

这个特性可以同时允许在两个 I/O 输出同一路 PWM。

12.12. P1C、P1D 的第 2 功能输出

除了正常的 PWM 波形输出，P1C 的重映射管脚 (PC6/[P1C1])，输出 P1C、P1D 之间的同或，异或输出，由寄存器 P1CF2E 控制。P1D 的重映射管脚 (PC7/[P1D1]、PA7[P1D2]) 可输出 P1B、P1C 之间的同或，异或输出，由寄存器 P1DF2E 控制。

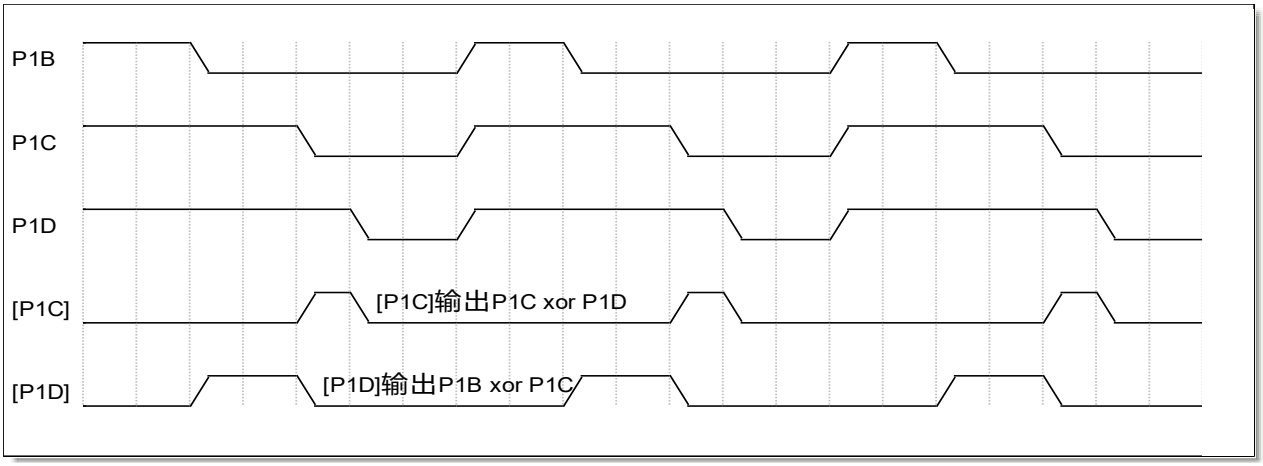


图 12-10 P1B 和 P1C 的第 2 功能时序示例

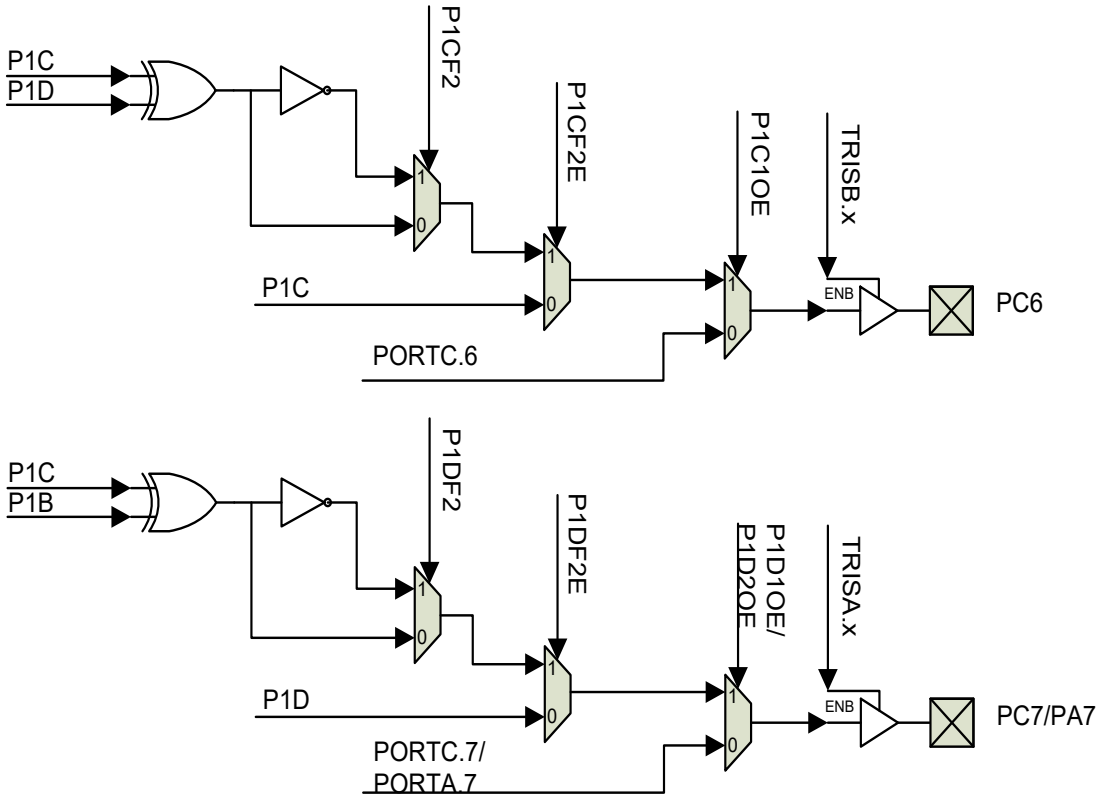


图 12-11 P1C 和 P1D 的第 2 功能输出

13. 睡眠模式

睡眠模式下，指令时钟关闭，指令执行停止，大多数模块掉电以降低功耗。如表 13-1 所示，FT62F28x 可根据实际需求在睡眠时有选择地开启各个模块，而无须指令介入，以使其相应功能如 LVR、LVD、WDT、Timers、PWM、运放、比较器和 ADC 能在 SLEEP 模式下保持运行。一些模块也可配置成进入 SLEEP 后自动关闭，而无须由指令关闭。

模块	SLEEP 模式下的各模块配置条件	
	运行	自动关闭?
指令时钟	(始终关闭)	Yes
LVR	LVREN = 00 or (LVREN = 01 & SLVREN=1)	LVREN = 10
LVD	LVDEN = 1	No
WDT	WDTE or SWDTEN	No
TIMER0	SYSON = 1 & T0CKRUN = 1 (时钟源非指令时钟)	SYSON = 0
TIMER1	SYSON = 1 & T1CKRUN = 1 (时钟源非指令时钟)	SYSON = 0
TIMER2	SYSON = 1 & T2CKRUN = 1 (时钟源非指令时钟)	SYSON = 0
PWM	(跟随 Timer2)	
HIRC / LIRC / EC / LP / XT	(跟随使用它们的外设状态)	
ADC	(当 ADON = 1 且 ADC 所选时钟源保持运行时，ADC 即可运行)	
运放	(当 OP0ON = 1 时，运放即可运行)	
比较器	(当 CM0EN = 1 或 CM1EN = 1 时，比较器即可运行)	
SPI	(当 SPI 所选时钟源保持运行时，SPI 即可运行)	
I2C	(当 I2C 所选时钟源保持运行时，I2C 即可运行)	
USART0 / USART1	(当 USART 所选时钟源保持运行时，USART 即可运行)	
TOUCH	(当在硬件扫描模式下，TOUCH 即可运行)	
I/O	(除非 SLEEP 时使能 PWM，否则 I/O 将保持其进入 SLEEP 前的状态)	

表 13-1 除指令时钟外，其他模块可根据需求在 SLEEP 模式下保持运行

13.1. 进入 SLEEP

CPU 通过执行 SLEEP 指令进入睡眠模式。进入睡眠时：

1. WDT 将清零但是保持运行 (如果使能了在休眠期间工作)
2. STATUS 寄存器的 PD 位清零
3. STATUS 寄存器的 TO 位置 1
4. CPU 时钟停止
5. 32kHz LIRC 不受影响，并且由其提供时钟的外设可以在休眠模式下继续工作
6. LP 晶体振荡器不受影响 (当 TIMx 使用它作为工作时钟时)
7. ADC 不受影响 (如果选择了专用 FRC 时钟)
8. I/O 端口保持执行 SLEEP 指令之前的状态 (驱动为高电平、低电平或高阻态)
9. WDT 之外的复位不受休眠模式影响

关于外设在休眠期间工作的更多详细信息，请参见各个章节。

要最大程度地降低电流消耗，应考虑以下条件：

1. I/O 引脚不应悬空，I/O 作为输入时可打开内部的上拉或下拉
2. 外部电路从 I/O 引脚灌电流
3. 内部电路从 I/O 引脚拉电流
4. 内部弱上拉的引脚
5. 模块使用 32kHz LIRC
6. 模块使用 LP 振荡器

13.2. 睡眠的唤醒

可以通过下列任一事件将器件从休眠状态唤醒：

1. MCLR 引脚上的外部复位输入 (如果使能)
2. BOR 复位 (如果使能)
3. POR 复位
4. 看门狗定时器 (如果使能)溢出
5. 任何外部中断
6. 能够在休眠期间运行的外设产生的中断 (更多信息请参见各个外设)

前 3 个事件会使器件复位，后 3 个事件认为是程序执行的延续。

清看门狗指令 CLRWDT、SLEEP (进入睡眠模式)或者从睡眠模式唤醒，都将清除看门狗计数器。

13.3. 看门狗唤醒

看门狗工作在内部慢时钟 (32kHz)，它是一个 16 位的计数器，和定时器 0 共用一个 8 位的预分频器，使能位位于配置寄存器 UCFG0 的第 3 位，WDTEN，为 1 时表示使能看门狗，为 0 时将由 SWDTEN 位决定使能与否，SWDTEN 位于 WDTCON 寄存器。

清看门狗指令 CLRWDT、SLEEP 会清除看门狗计数器。

在使能了看门狗的情况下，MCU 睡眠时看门狗溢出事件可以作为一个唤醒源，而 MCU 正常工作时作为一个复位源。

13.4. 中断唤醒

FT62F28x 系列芯片在中断唤醒时，CPU 先执行 SLEEP 的下一条指令，再先进入中断处理程序。

14. 中断

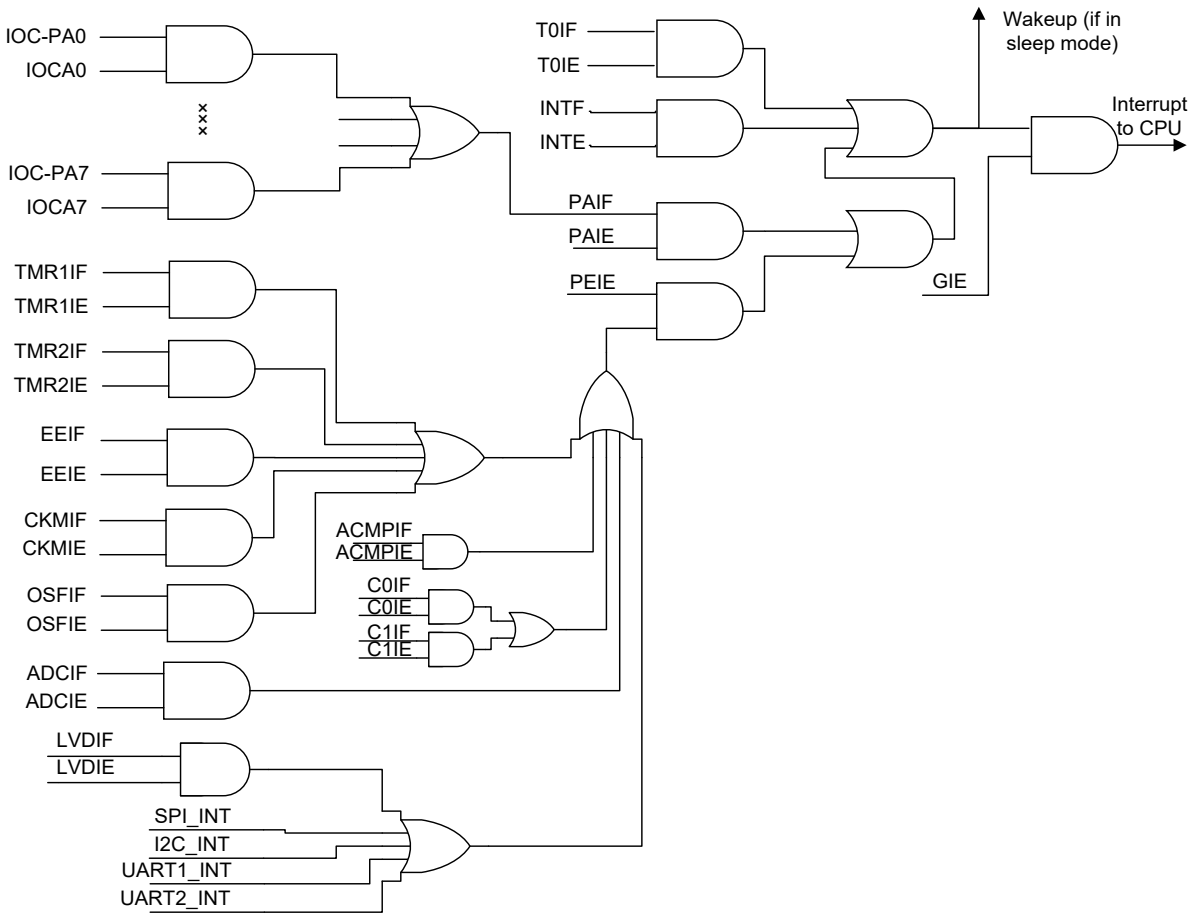


图 14-1 中断逻辑框图

FT62F28x 有以下中断源，部分中断可以把 CPU 从睡眠状态唤醒：

- ADC 转换结束中断
- ADC 阈值比较中断
- INT 管脚中断
- Timer0 匹配中断
- PORTA 电平变化中断
- Timer2 匹配相等中断
- Timer1 匹配相等中断
- EEPROM 数据写中断
- 时钟缺失中断
- LVD 中断
- 比较器 0/1 中断
- USART0/1 中断
- I2C 中断

- SPI 中断

中断控制寄存器 (INTCON) 和外设中断请求寄存器 (PIR1/PIR2) 记录了中断标志位。INTCON 同时也包含全局中断使能位 GIE。

当中断被服务后，以下动作自动发生：

- GIE 被清零，从而关闭中断
- 返回地址被推上堆栈
- 程序指针加载 0004h 地址

中断返回指令，RETI 将退出中断函数时同时设置 GIE 位，重新使能未屏蔽的中断。需要注意的是，执行中断返回 RETI 之前应该把相关的中断标志位清 0，以免重复进入中断处理程序。

INTCON 寄存器包含以下中断标志位：

- INT 管脚中断
- PORTA 变化中断
- Timer0 溢出中断

PIR1 和 PIR2 中包含着外设中断标志位，PIE1/PIE2 中包含着其对应的中断使能位，具体请参照寄存器各位的描述。

注意：

从**图 14-1** 中断逻辑框图可以看出，除了 Timer0，外部管脚，以及 PORTA 电平变化中断之外，其它外设中断皆受外设中断使能位 PEIE 的控制。

14.1. 中断相关寄存器汇总

名称	地址	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位(RW)
INTCON	0x0B	GIE	PEIE	T0IE	INTE	PAIE	T0IF	INTF	PAIF	0000 0000
PIE1	0x8C	EEIE	CKMIE	LVDIE	ACMPIE	TMR1IE	OSFIE	TMR2IE	ADCIE	0000 0000
PIR1	0x0C	EEIF	CKMIF	LVDIF	ACMPIF	TMR1IF	OSFIF	TMR2IF	ADCIF	0000 0000
PIE2	0x89	–	–	–	–	–	–	C1IE	C0IE	---- --00
PIR2	0x09	–	–	–	–	–	–	C1IF	C0IF	---- --00
TRISA	0x85	TRISA[7:0]，PORTA 方向控制								1111 1111
IOCA	0x98	IOCA[7:0]，端口变化中断允许位								0000 0000
OPTION	0x81	/PAPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111
COMAF1	0x286	INTPO[1:0]		TX0PO[2:0]			RX0PO[2:0]			0000 0000
COMAF2	0x287	UR1SW	UR0SW	–	TX1PO[1:0]		INTFIXB	RX1PO[1:0]		00-0 0100
UR0CR2	0x188	UR0BRRH[3:0]				–	UR0ERRIE	UR0RXNEIE	UR0TXEIE	000- 0011
UR1CR2	0x93	UR1BRRH[3:0]				–	UR1ERRIE	UR1RXNEIE	UR1TXEIE	000- 0011
UR0STAT	0x19F	–	UR0TXBSY	UR0OVF	UR0PEF	UR0RXFULL	UR0FEF	UR0RXNEF	UR0TXEF	-000 0001
UR1STAT	0x95	–	UR1TXBSY	UR1OVF	UR1PEF	UR1RXFULL	UR1FEF	UR1RXNEF	UR1TXEF	-000 0001
I2CCR2	0x197	I2CADDRH[1:0]		NACK	DUTY	–	–	–	I2CIE	0000 0--0
I2CISR	0x19E	HOLDF	STOPF	ADDRF	NACKF	I2CBUSY	DIRF / BUSERR	IICRXNEF	IICTXEF	0000 0001
SPIIER	0x193	–	–	–	–	WAKUP	RXERR	RXNE	TXE	---- 0000
SPISTAT	0x195	SPIF	WCOL	MODF	RXOVRN	–	–	WKF	CRCERR	0000 --00
SPICFG	0x18E	SPIBUSY	MSTEN	CPHA	CPOL	SLAS	NSSVAL	SRMT	RXBMT	0000 0111
SPICR0	0x18D	–	–	–	–	NSSM		TXBMT	SPIEN	---- 1110

表 14-1 中断相关寄存器地址和默认值

名称	状态		寄存器	地址	复位值
GIE	全局中断	1 = 使能 (PEIE，各中断独立使能位适用) 0 = 全局关闭 (唤醒不受影响)	INTCON[7]	Bank 首地址 +0x0B	RW-0
PEIE	外设总中断	1 = 使能 (各中断独立使能位适用) 0 = 关闭 (无唤醒)	INTCON[6]		RW-0
T0IE	Timer0 溢出中断	1 = 使能 0 = 关闭 (无唤醒)	INTCON[5]		RW-0
INTE	外部中断		INTCON[4]		RW-0
PAIE	PORTA 端口变化总中断		INTCON[3]		RW-0
T0IF	Timer0 中断标志位	1 = Yes (锁存) 0 = No	INTCON[2]		RW0-0
INTF	外部中断标志位		INTCON[1]		RW0-0
PAIF	PORTA 端口变化中断标志位		INTCON[0]		RW0-0

表 14-2 INTCON 寄存器

名称	状态		寄存器	地址	复位值
EEIE	EE 写完成中断	1 = 使能 0 = 关闭 (无唤醒)	PIE1[7]	0x8C	RW-0
CKMIE	LIRC 和 HIRC 交叉校准完成中断		PIE1[6]		RW-0
LVDIE	低电压检测中断		PIE1[5]		RW-0
ACMPIE	ADC 阈值比较结果中断		PIE1[4]		RW-0
TMR1IE	Timer1 与 PR1 匹配中断		PIE1[3]		RW-0
OSFIE	振荡器故障中断		PIE1[2]		RW-0
TMR2IE	Timer2 与 PR2 匹配中断		PIE1[1]		RW-0
ADCIE	ADC 转换完成中断		PIE1[0]		RW-0
C1IE	比较器 1 中断		PIE2[1]	0x89	RW-0
C0IE	比较器 0 中断		PIE2[0]		RW-0
EEIF	EE 写完成中断标志位	1 = Yes (锁存) 0 = No	PIR1[7]	0x0C	RW0-0
CKMIF	LIRC 和 HIRC 交叉校准完成中断标志位		PIR1[6]		RW0-0
LVDIF	低电压检测中断标志位		PIR1[5]		RW0-0
ACMPIF	ADC 阈值比较结果中断标志位		PIR1[4]		RW0-0
TMR1IF	Timer1 与 PR1 匹配中断标志位		PIR1[3]		RW0-0
OSFIF	振荡器故障中断标志位		PIR1[2]		RW0-0
TMR2IF	Timer2 与 PR2 匹配中断标志位		PIR1[1]		RW0-0
ADCIF	ADC 转换完成中断标志位		PIR1[0]		RW0-0
C1IF	比较器 1 中断标志位		PIR2[1]	0x09	RW0-0
C0IF	比较器 0 中断标志位		PIR2[0]		RW0-0

表 14-3 PIE1/PIR1 和 PIE2/PIR2 寄存器

名称	状态		寄存器	地址	复位值
IOCA	PORTA 各端口变化中断	1 = 使能 0 = 禁止	IOCA[7:0]	0x98	RW-0000 0000
INTEDG	INT 中断沿	1 = 上升沿 0 = 下降沿	OPTION[6]	0x81	RW-1
INTPO	INT / BKIN / ADC_ETR 管脚映射	0x = PD4 11 = PC4 10 = PD2	COMAF1[7:6]	0x286	RW-00
INTFIXB	INT 中断管脚映射 (≥ G 版芯片) 1 = 由 INTPO 决定 0 = 映射在 PD4 (BKIN 和 ADC_ETR 复用映射仍由 INTOP 决定) 注：<G 版芯片，此位为保留位，读为 0		COMAF2[2]	0x287	RW-1

表 14-4 IO 相关中断配置

名称	状态		寄存器	地址	复位值
GIE	全局中断	1 = 使能 (PEIE, URxERRIE, URxRXNEIE, URxTXEIE 适用) 0 = 全局关闭 (唤醒不受影响)	INTCON[7]	Bank 首地址 +0x0B	RW-0
PEIE	外设总中断	1 = 使能 (URxERRIE, URxRXNEIE, URxTXEIE 适用) 0 = 关闭 (无唤醒)	INTCON[6]		RW-0
UR0ERRIE	USART0/1 接收错误中断	1 = 使能 0 = 关闭	UR0CR2[2]	0x188	RW-0
UR1ERRIE			UR1CR2[2]	0x93	RW-0
UR0OVF [1]	USART0 接收 FIFO 溢出标志	1 = 发生 0 = 未发生，或已清零	UR0STAT[5]	0x18F	RW0-0
UR0PEF [1]	USART0 接收奇偶校验错标志		UR0STAT[4]		RW0-0
UR0FEF [1]	USART0 接收帧错误标志		UR0STAT[2]		RW0-0
UR1OVF [1]	USART1 接收 FIFO 溢出标志		UR1STAT[5]	0x95	RW0-0
UR1PEF [1]	USART1 接收奇偶校验错标志		UR1STAT[4]		RW0-0
UR1FEF [1]	USART1 接收帧错误标志		UR1STAT[2]		RW0-0
UR0RXNEIE	USART0/1 接收 FIFO 非空中断	1 = 使能 0 = 关闭	UR0CR2[1]	0x188	RW-1
UR1RXNEIE			UR1CR2[1]	0x93	RW-1
UR0RXNEF	USART0/1 接收 FIFO 状态	1 = 非空 0 = 空，或已被清零	UR0STAT[1]	0x18F	RO-0
UR1RXNEF			UR1STAT[1]	0x95	RO-0
UR0TXEIE	USART0/1 发送 FIFO 为空中断	1 = 使能 0 = 关闭	UR0CR2[0]	0x188	RW-1
UR1TXEIE			UR1CR2[0]	0x93	RW-1
UR0TXEF	USART0/1 发送 FIFO 状态	1 = 空 0 = 非空	UR0STAT[0]	0x18F	RO-1
UR1TXEF			UR1STAT[0]	0x95	RO-1
UR0TXBSY	USART0/1 发送器状态 (半双工模式)	1 = 忙碌 0 = 空闲	UR0STAT[6]	0x18F	RO-0
UR1TXBSY			UR1STAT[6]	0x95	RO-0
UR0RXFULL	USART0/1 接收 FIFO 状态 (FIFO 深度为 2)	1 = 已满 0 = 未满	UR0STAT[3]	0x18F	RO-0
UR1RXFULL			UR1STAT[3]	0x95	RO-0

表 14-5 USART 中断使能和标志位

[1] 写 0 清 0，写 1 无效；；

名称	状态		寄存器	地址	复位值
WAKUP	SPI 唤醒中断	1 = 使能 0 = 关闭	SPIIER[3]	0x193	RW-0
RXERR	SPI 接收错误中断		SPIIER[2]		RW-0
RXNE	SPI 接收 BUF 不为空中断		SPIIER[1]		RW-0
TXE	SPI 发送 BUF 为空中断		SPIIER[0]		RW-0
WKF [2]	SPI WAKEUP 唤醒标志位	1 = 发生唤醒 0 = 未发生唤醒	SPISTAT[1]	0x195	RW0-0
RXOVRN [2]	SPI 接收溢出标志位	1 = 溢出 0 = 正常	SPISTAT[4]		RW0-0
CRCERR [2]	SPI CRC 错误标志位	1 = 发生 CRC 错误 0 = 没有 CRC 错误	SPISTAT[0]		RW0-0
MODF [2]	SPI 工作模式错误标志位	1 = 出错 0 = 正常	SPISTAT[5]		RW0-0
WCOL	BUF 写入失败标志位	1 = 失败（当进行写入时 BUF 为非空) 0 = 正常	SPISTAT[6]		RW0-0
SPIF [2]	传输完成标志	1 = 传输完成 0 = 未传输完成，或已清零	SPISTAT[7]		RW0-0
TXBMT	SPI 发送 BUFFER 为空状态标志位	1 = 为空 0 = 非空	SPICR0[1]	0x18D	RO-1
RXBMT	SPI 接收 BUFFER 为空状态标志位		SPICFG[0]	0x18E	RO-1
SRMT	FIFO 为空状态		SPICFG[1]		RO-1
SLAS	SLAVE 选择标志 1 = 该模块被选中 0 = 该模块未被选中 注：当 NSS 用作输入时，该值可以被 SSM 软件管理，当 SSM 为 1 时，这里的值表示的是 SSI 的值取反		SPICFG[3]		RO-0
SPIBUSY	BUSY 状态	1 = 忙碌 0 = 空闲	SPICFG[71]		RO-0

表 14-6 SPI 中断使能和标志位

[2] 写 0 清 0，写 1 无效

名称	状态		寄存器	地址	复位值
I2CIE	I2C 接口中断	1 = 使能 0 = 关闭	I2CCR2 [0]	0x197	RW-0
HOLDF	主机模式，SCL 拉伸状态	1 = SCL 正被主机拉伸 0 = SCL 未被主机拉伸	I2CISR[7]	0x19E	RO-0
	从机模式，SCL 拉伸状态 [3]	1 = SCL 正被从机拉伸 0 = SCL 未被从机拉伸，或已被清零			RW0-0
STOPF [3]	主/从机模式，Stop 标志	1 = Yes 0 = No	I2CISR[6]		RW0-0
ADDRF [3]	主机模式，目标从机地址匹配标志	1 = 匹配 (ACK 后置位) 0 = 不匹配，或未发送地址	I2CISR[5]	0x19E	RW0-0
	从机模式，本机地址匹配标志	1 = 匹配 [4] 0 = 不匹配			
NACKF [3]	主/从机模式，接收应答状态	1 = NACK 0 = ACK	I2CISR[4]		RW0-0
I2CBUSY	主/从机状态	1 = Busy (繁忙) 0 = IDLE (空闲) 注：从机模式，地址匹配成功后即置位，接收到 Start / Restart / Stop 后清零；	I2CISR[3]		RO-0
BUSERR / DIRF	主机模式，总线仲裁失败 (BUSERR) 标志	1 = 仲裁失败 0 = 未发生仲裁失败	I2CISR[2]		RO-0
	从机模式，数据传输方向 (DIRF) 标志 [3]	1 = 发送 (从机接收地址字节的读写位为 1 时置位) 0 = 接收			RW0-0
IICRXNEF [3]	RX-FIFO 状态	1 = 非空 (读 I2CDAT 或写 0 清零) 0 = 空	I2CISR[1]		RW0-0
IICTXEF [5]	TX-FIFO 状态	1 = 空 (写 I2CDAT 清零) 0 = 非空 (写 1 清空)	I2CISR[0]		RW1-1

表 14-7 I2C 中断使能和标志位

[3] 写 0 清零，写 1 无效；
[4] 10 位地址格式，低 8 位地址和高 2 位地址匹配时均会置位；
[5] 只能写 1，写 0 无效；

14.2. INT 外部中断

INT 管脚上的外部中断是边沿触发的：当 OPTION 寄存器的 INTEDG 位被置 1 时在上升沿触发，而当 INTEDG 位被清零时在下降沿触发。当 INT 管脚上出现有效边沿时，INTCON 寄存器的 INTF 位置 1。可以通过将 INTCON 寄存器的 INTE 控制位清零来禁止该中断。在重新允许该中断前，必须在中断服务程序中先用软件将 INTF 位清零。如果 INTE 位在进入休眠状态前被置 1，则 INT 管脚中断能将处理器从休眠状态唤醒。

INT 中断管脚一共有 3 种映射，分别是 PD4，PD2 以及 PC4，由 COMAF1 的 INTPO[1:0]位控制。

14.3. 端口变化中断

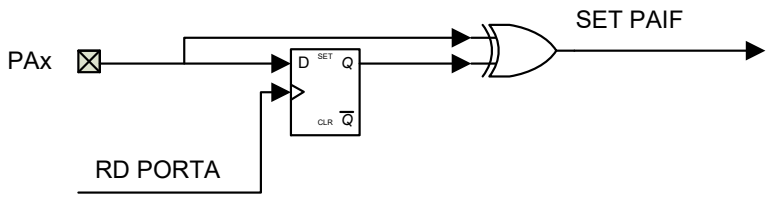


图 14-2 端口变化中断原理框图

PORTA 输入电平的变化会使 INTCON 寄存器的 PAIF 位置 1。可以通过置 1/清零 INTCON 寄存器的 PAIE 位来使能/禁止该中断。此外，可通过 IOCA 寄存器对该端口的各个管脚进行配置。

注意：
1. 初始化电平变化中断时，应先配置为数字输入 IO，把相应的 IOCA 置 1，然后读取一下该 PORTA；
2. 当 I/O 电平发生变化时，PAIF 被置 1；
3. 清中断标志位之前应该读取一下 PORTA，然后再对 PAIF 清 0；

14.3.1. PAIF 标志位的清除

PAIF 寄存位是异步置位的，即，如果端口不匹配事件一直存在，软件是无法完成对其清除操作的。要想把它清 0，有以下两种方法：
方法 1
a) 等待外部管脚恢复原来的电平
b) 软件清 PAIF

方法 2
a) 读取 PORTA，主动清除不匹配事件
b) 软件清除 PAIF

14.4. 中断响应

外部中断包括 INT 管脚进来的或者 PORTA 变化中断的延时一般为 1 到 2 个指令周期。它跟中断发生时刻与正在执行的指令有关系。

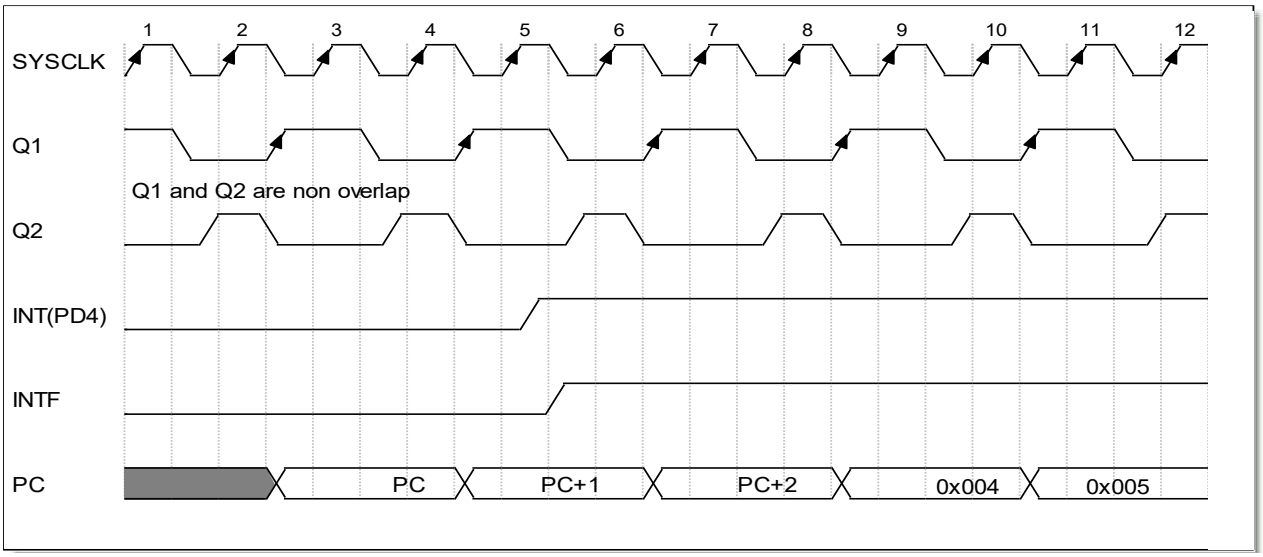


图 14-3 中断响应时序图

14.5. 中断过程中的现场保存

在中断过程中，只有返回 PC 被自动保存在堆栈上。一般来说，用户可能需要保存重要的寄存器值在堆栈上，例如 W，STATUS 寄存器等。这些必须由软件来完成。临时寄存器 W_TEMP 和 STATUS_TEMP 应该被放置在 GPR 的最后 16byte 里。这 16 个 GPR 落在各个页区间，因此可以稍微节省代码。

如果中断服务程序涉及到 AUXPGE 修改操作的，则要对 AUXPGE 进行现场保护，退出中断前再恢复现场，以下是一段示例代码。

```
void interrupt ISR() {
   AUXPGE_SHADOW= AUXPGE;      // 处理中断前，保存 AUXPGE
   ...                         // 其它代码
   AUXPGE= AUXPGE_SHADOW;      // 退出中断前，恢复 AUXPGE
}
```

14.6. 关于中断标志位

所有外设的中断标志位均独立于其中断使能（允许）位，即使各中断使能位为 0，发生了相关中断事件，其标志位仍然会被置 1。

15. 数据 EEPROM

片内集成有 128 个字节的 EEPROM，通过 EEADR 进行寻址访问。软件可通过 EECON1 和 EECON2 对 EEPROM 进行编程操作，硬件实现了擦除和编程的自定时功能，无需软件查询，节省有限的代码空间，同时利用此特性，启动编程周期之后可以进入睡眠模式，以降低功耗。

编程 EEPROM 需要遵循一定的步骤，这种机制可以防止程序跑飞或者程序丢失引起的误写操作。

15.1. 与数据 EEPROM 相关寄存器汇总

名称	状态	寄存器	地址	复位值
EEDAT	DATA EEPROM 数据	EEDAT[7:0]	0x9A	RW-0000 0000
EEADR	DATA EEPROM 地址	EEADR[7:0]	0x9B	RW-0000 0000
WREN3	DATA EEPROM 写使能 (bit 3) 111 = 使能，完成后重置为 000 (其他) = 关闭	EECON1[5]	0x9C	RW-0
WREN2	DATA EEPROM 写使能 (bit 2)	EECON1[4]		RW-0
WRERR	DATA EEPROM 写错误标志位 1 = 中止 (发生 MCLR 或 WDT 复位) 0 = 正常完成	EECON1[3]		RW-x
WREN1	DATA EEPROM 写使能(bit 1)	EECON1[2]		RW-0
PONLY	DATA EEPROM 自动擦除 1 = No (不擦除，只写) 0 = Yes (先擦除，再写)	EECON1[1]		RW-0
RD	DATA EEPROM 读控制位 1 = Yes (保持 4 个 Sysclk 周期，然后 = 0) 0 = No	EECON1[0]		RW-0
WR	DATA EEPROM 写控制位 1 = 启动一次写或写正在进行中 (完成后重置为 0) 0 = 完成	EECON2[0]	0x9D	RW-0

表 15-1 EEPROM 相关用户控制寄存器

名称	状态		寄存器	地址	复位值
GIE	全局中断	1 = 使能 (PEIE, EEIE 适用) 0 = 全局关闭 (唤醒不受影响)	INTCON[7]	Bank 首地址 +0x0B	RW-0
PEIE	外设总中断	1 = 使能 (EEIE 适用) 0 = 关闭 (无唤醒)	INTCON[6]		RW-0
EEIE	EEPROM 写完成中断	1 = 使能 0 = 关闭 (无唤醒)	PIE1[7]	0x8C	RW-0
EEIF	EEPROM 写完成中断标志位	1 = Yes (锁存) 0 = No	PIR1[7]	0x0C	RW0-0

表 15-2 数据 EEPROM 中断使能和状态位

15.2. 编程数据 EEPROM 步骤

A. 把 INTCON 的 GIE 位清 0；

B. 判断 GIE 是否为 1，是则重复 A 步骤，否则可以进行下一步；

C. 往 EEADR 写入目标地址；

D. 往 EEDAT 写入目标数据；

E. 把位 WREN3/WREN2/WREN1 全部置 1；

F. 把位 WR 置 1 (EECON2.0，此后 WR 会维持高)；

G. 写过程不能改变 WREN3/2/1 的值，否则编程终止；

H. 等大概 2~4 ms 之后编程自动完成，WR 自动清 0，WREN3、WREN2、WREN1 清 0；

I. 如果想再次编程，重复步骤 C–H 即可；

例子 1:

```
BCR      INTCON, GIE
BTSC     INTCON, GIE
LJUMP    $-2
BANKSEL  EEADR
LDWI     55H
STR      EEADR          ;地址为 0x55
STR      EEDAT          ;数据为 0x55
LDWI     34H
STR      EECON1         ;WREN3/2/1 同时置 1
BSR      EECON2, 0      ;启动写
BSR      INTCON, GIE    ;把 GIE 置 1
```

例子 2:

```
BCR      INTCON, GIE
BTSC     INTCON, GIE
```

```
LJUMP   $-2
BANKSEL     EEADR
LDWI    55H
STR     EEADR               ;地址为 0x55
STR     EEDAT               ;数据为 0x55
LDWI    34H
STR     EECON1              ;WREN3/2/1 同时置 1
NOP                         ;这里 NOP 可以换成其他指令
BSR     EECON2, 0           ;启动写，实际上硬件不会启动编程 EEPROM 操作
BCR     EECON1, WREN1       ;先清 WREN1，使得 WREN3/2/1 不同时为 1
BSR     EECON1, WREN1       ;重新置位 WREN1，令 WREN3/2/1 同时为 1
BSR     EECON2, 0           ;启动写，这次硬件将对 EEPROM 编程
BSR     INTCON, GIE
```

注意：

1. 以上步骤的 E、F 两步必须是连续的两条指令周期完成，不能错开 (如例子 2)，否则编程操作不会启动，其中 WREN3、WREN2 和 WREN1 可以不是同一条指令置 1，比如可以用 BSR 指令分开对各位置 1；
2. 如果 E、F 两步被错开执行，要想启动下一次编程操作，必须在 E、F 之前加入一步，把 WREN3、WREN2 或者 WREN1 任意一位清 0，如例子 2；
3. 编程过程中读操作无效；
4. 如编程完成前 WREN3、WREN2 或者 WREN1 任意一位清 0，在下次编程前需清除 EEIF 标志位。

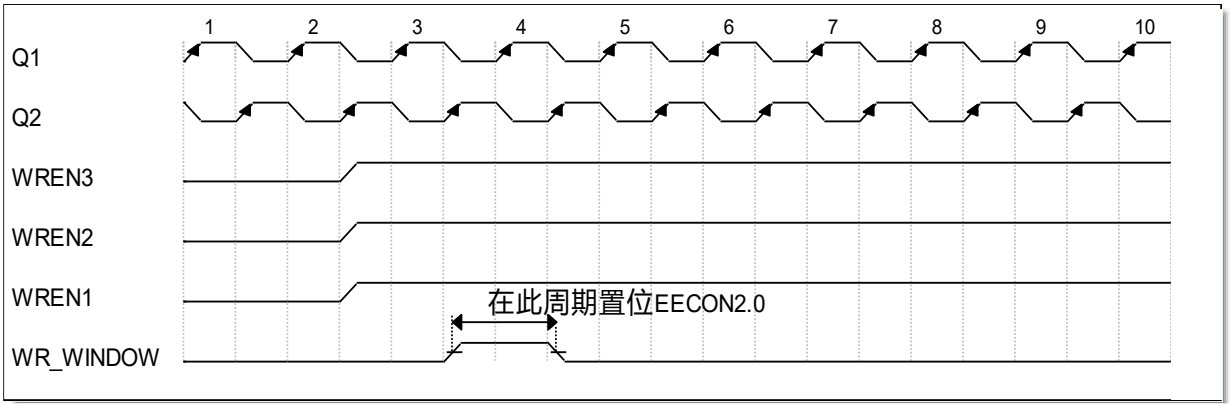


图 15-1 软件编程数据 EEPROM 时序

15.3. 读数据 EEPROM

要读取数据存储单元，用户必须将地址写入 EEADR 寄存器，然后将 EECON1 寄存器的控制位 RD 置 1。EEPROM 读周期为 4 个系统时钟，所以在发起读操作之后的第 2 个指令周期，EEDAT 寄存器才被 EEPROM 数据写入，即该数据可由下下一条指令读取。EEDAT 将保持这个值直到用户下一次从该单元读取或向该单元写入数据时 (在写操作过程中)。

下面是读取 EEPROM 的一段示例程序：

```
BANKSEL     EEADR
LDWI    dest_addr
```

```
STR     EEADR
BSR     EECON1, RD
NOP                     ；读等待
LDR     EEDAT, W        ；这时，数据可以被软件读取
```

15.4. 关于编程周期

启动数据 EEPROM 的编程操作后，编程计时开始，在这段时间内，CPU 并不会暂停，而是继续执行程序。

15.5. EEPROM 的单编程模式

EEPROM 除了正常的自擦写模式 (编程操作包含了自动擦除)，还支持单编程模式，即按 15.2 小节所述步骤启动编程后，所选中地址单元并不做自动擦除，而是仅仅包含编程，通过 PONLY 寄存器位控制。注意：EEPROM 的单编程模式只能把数据由 1 写成 0，而不能由 0 写为 1，这有点像“按位与”运算。

举例说明，假如 0 地址存储了数据 0xAA，对其启动正常的自擦写模式 (PONLY=0) 写 0x55，待操作结束后，0 地址存储的数据变成 0x55；

但如果设置的是单编程模式 (PONLY=1)，写同样的数据 0x55，最终 0 地址存放的数据将会是 0x00。

15.6. 关于最后 16 个字节

EEPROM 的最后两页 (0x70~07F，共 16byte) 被用来存放 ADC 内部 VREF 值，以及 OP0 V_{OS} 的 AD 转换值，如下表：

地址	名称	说明
0x70	VREF0P5_CAL[7:0]	ADC 内部参考 0.5V 的实测值
0x71	VREF0P5_CAL[11:8]	(VDD = 5V)
0x72	VREF2P0_CAL[7:0]	ADC 内部参考 2V 的实测值
0x73	VREF2P0_CAL[11:8]	(VDD = 5V)
0x74	VREF3P0_CAL[7:0]	ADC 内部参考 3V 的实测值
0x75	VREF3P0_CAL[11:8]	(VDD = 5V)
0x76	N/A	
0x77	N/A	
0x78	VOS_G10_CAL[7:0]	配置为 10 (反馈电阻 40k/4k) 时的 AD 转换值
0x79	VOS_G10_CAL[11:8]	(VREF = 内部 3V，VDD = 5V)
0x7A	VOS_G20_CAL[7:0]	配置为 20 (反馈电阻 80k/4k) 时的 AD 转换值
0x7B	VOS_G20_CAL[11:8]	(VREF = 内部 3V，VDD = 5V)
0x7C	VOS_G40_CAL[7:0]	配置为 40 (反馈电阻 160k/4k) 时的 AD 转换值
0x7D	VOS_G40_CAL[11:8]	(VREF = 内部 3V，VDD = 5V)
0x7E	VOS_G80_CAL[7:0]	配置为 80 (反馈电阻 320k/4k) 时的 AD 转换值
0x7F	VOS_G80_CAL[11:8]	(VREF = 内部 3V，VDD = 5V)

16. 12bit ADC 模块

模数转换器 (Analog-to-digital Converter，ADC) 可将模拟输入信号转换为相应的 12 位二进制表征值。该系列器件采用多个模拟输入复用到一个采样保持电路。采样保持电路的输出与转换器的输入相连接。转换器通过逐次逼近法产生 12 位二进制值，并将转换结果保存在 ADC 结果寄存器 (ADRESL:ADRESH) 中。ADC 参考电压可用软件选择为 VDD、外部参考电压或内部产生的参考电压。ADC 可在转换完成时产生中断。该中断可用于将器件从休眠唤醒。

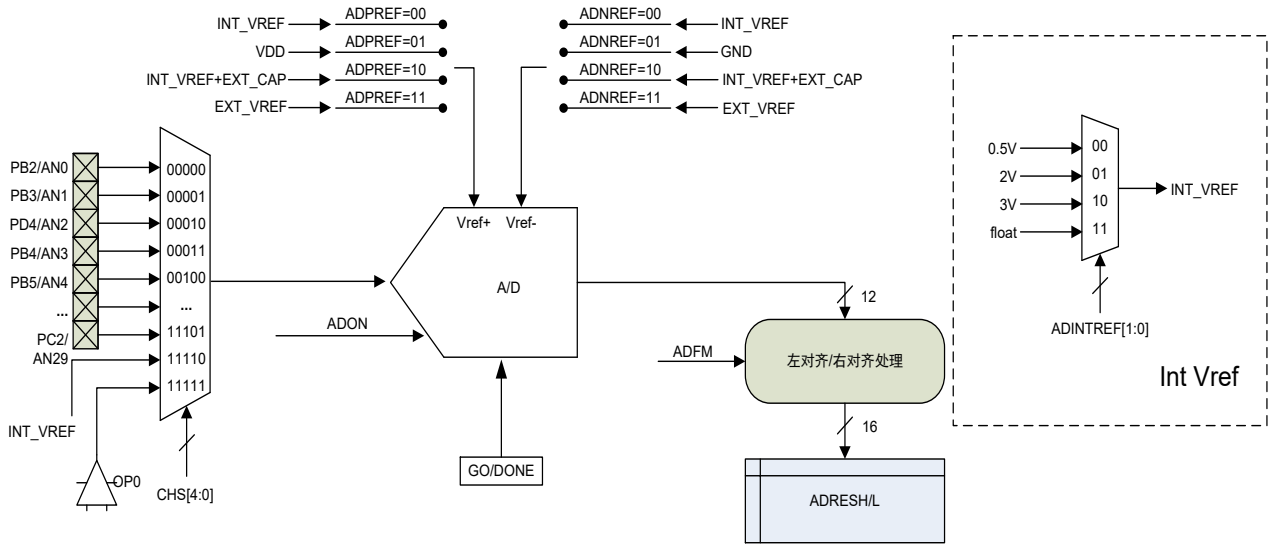


图 16-1 ADC 原理框图

16.1. ADC 相关寄存器汇总

名称	地址	bit 7	bit 6	bit 5	bit 4	bit 3	Bit 2	bit 1	bit 0	复位值
ADRESL	0x111	A/D 转换结果低有效位								xxxx xxxx
ADRESH	0x112	A/D 转换结果高有效位								xxxx xxxx
ADCON0	0x113	CHS[4:0]					ADEX	GO/DONE	ADON	0000 0000
ADCON1	0x114	ADFM	ADCS[2:0]			ADNREF[1:0]		ADPREF[1:0]		0000 0000 0010 0100 (≥VerE)
ADCON2	0x115	ADINTREF[1:0]		ETGTYP[1:0]		ADDLY.8	ETGSEL[2:0]			0000 0000 0100 0000 (≥VerE)
ADDLY	0x110	ADDLY[7:0] / LEBPRL[7:0]								0000 0000
ADCON3	0x116	ADFBEN	ADCMPP	ADCMPE	–	LEBADT	–			000– 0–––
ADCMPH	0x10F	ADCMPH[7:0]								0000 0000
LEBCON	0x10E	LEBEN	LEBCH		–	EDGS	–			000– 0–––

表 16-1 ADC 相关用户寄存器地址

名称	状态	寄存器	地址	复位值
ADRESL	ADC 转换结果低有效位 (LSB) ADFM=0: ADRESL[7:4] = 低 4 位 (其余为“0”) ADFM=1; ADRESL[7:0] = 低 8 位	ADRESL[7:0]	0x111	RW-xxxx xxxx
ADRESH	ADC 转换结果高有效位 (MSB) ADFM=0: ADRESH[7:0] = 高 8 位 ADFM=1: ADRESH[3:0] = 高 4 位 (其余为“0”)	ADRESH[7:0]	0x112	RW-xxxx xxxx
LEBEN	ADC 触发和 BKIN 的 LEB 使能位 1 = 使能 (当 GO/DONE=1 时进行切换将产生不可预知的结果) 0 = 关闭	LEBCON[7]	0x10E	RW-0
LEBCH	LEB 信号源 (<G 版芯片) 00 = P1A0 10 = P1C 01 = P1B 11 = P1D (≥ G 版芯片为保留位)	LEBCON[6:5]		RW-00
EDGS	LEB 触发沿 (<G 版芯片) 1 = PWM 下降沿 0 = PWM 上升沿 LEB 周期结束后，触发 ADC 转换 (≥ G 版芯片) 1 = 触发 0 = 不触发	LEBCON[3]		RW-0
ADCMPH	ADC 比较阈值 (仅高 8 位，0.4% steps)	ADCMPH[7:0]	0x10F	RW-0000 0000
ADDLY / LEBPRL	ADC 延迟/ LEB (非软件触发，ADEX = 1) (此为低 8 位, ADDLY.8 为高有效位) 外部延迟时间 = (ADDLY+6) / F_{ADC} 外部延迟时间 = (ADDLY+3) / F_{ADC} (≥ G 版芯片) 注：如果启用 PWM 输出触发 ADC，在 PWM 运行过程中不得更改 ADDLY；	ADDLY[7:0]	0x110	RW-0000 0000
CHS	ADC 模拟输入通道 00000 = AN0 10000 = AN16 00001 = AN1 10001 = AN17 00010 = AN2 10010 = AN18 00011 = AN3 10011 = AN19 00100 = AN4 10100 = AN20 00101 = AN5 10101 = AN21 00110 = AN6 10110 = AN22 00111 = AN7 10111 = AN23 01000 = AN8 11000 = AN24 01001 = AN9 11001 = AN25	ADCON0[7:3]	0x113	RW-00000

名称	状态	寄存器	地址	复位值
	01010 = AN10 11010 = AN26 01011 = AN11 11011 = AN27 01100 = AN12 11100 = AN28 01101 = AN13 11101 = AN29 01110 = AN14 11110 = (内部 $V_{ADC-REF}$) 01111 = AN15 11111 = 运放 0 输出			
ADEX	ADC 触发条件 (GO/DONE) 1 = 由 ADC_ETR 或 PWM 外部触发事件置位 GO/DONE (硬件触发) 0 = 由指令置位 GO/DONE (软件触发)	ADCON0[2]		RW-0
GO/DONE	ADC 转换启动和状态位 1 = 由软件，或 ADC_ETR 或 PWM 启动 A/D 转换 (转换完成后自动清零) 0 = 转换完成 / 未进行转换	ADCON0[1]		RW-0
ADON	1 = ADC 使能 0 = ADC 关闭 (无电流消耗)	ADCON0[0]		RW-0
LFMOD	1: LIRC = 256 kHz 0: LIRC = 32 kHz	OSCCON[7]	0x8F	RW-0
ADFM	A/D 转换结果格式 (参阅 “ADRESH”) 1 = 右对齐 0 = 左对齐	ADCON1[7]		RW-0
ADCS	ADC 转换时钟源 TSEL = 2T TSEL = 4T 000 = SysClk/2 000 = SysClk/4 001 = SysClk/8 001 = SysClk/16 010 = SysClk/32 010 = SysClk/64 011 = SysClk 011 = SysClk/2 100 = SysClk/4 100 = SysClk/8 101 = SysClk/16 101 = SysClk/32 110 = SysClk/64 110 = SysClk/128 111 = LIRC 111 = LIRC	ADCON1[6:4]	0x114	RW-000 (≥G 版)RW-010
ADNREF	$V_{ADC-REF}$ – (负参考电压) 00 = 内部 $V_{ADC-REF}$ 01 = GND 10 = 内部 $V_{ADC-REF}$ + 外部电容 Cap 11 = 外部参考电压 (I/O)	ADCON1[3:2]		RW-00 (≥G 版)RW-01

名称	状态	寄存器	地址	复位值
ADPREF	$V_{ADC-REF}$ + (正参考电压) 00 = 内部 $V_{ADC-REF}$ 01 = V_{DD} 10 = 内部 $V_{ADC-REF}$ + 外部电容 Cap 11 = 外部参考电压 (I/O)	ADCON1[1:0]		RW-00
ADINTREF	内部 $V_{ADC-REF}$ 00 = 0.5 01 = 2.0 10 = 3.0 11 = (未连接)	ADCON2[7:6]	0x115	RW-00 (≥G 版)RW-01
ETGTYP	外部触发沿 (当 ADEX=1 时适用) 00 = (PWM 或 ADC_ETR) 下降沿 01 = (PWM 或 ADC_ETR) 上升沿 1x = 保留值 注：≥G 版芯片，也作 LEB 触发沿选择	ADCON2[5:4]		RW-00
ADDLY.8 /LEBPR9	ADC 延迟计数器或 LEB 计数器的第 8 位 (参阅 “ADDLY”)	ADCON2[3]		RW-0
ETGSEL	外部触发源 (当 ADEX=1 时适用) 000 = P1A0 100 = P1D 001 = P1A0N 101 = ADC_ETR 010 = P1B 110 = (无) 011 = P1C 111 = (无) 注：≥G 版芯片，也作 LEB 触发源选择	ADCON2[2:0]	0x115	RW-000
ADFBEN	ADC 比较结果触发故障刹车 1 = 使能 0 = 关闭	ADCON3[7]	0x116	RW-0
ADCMPP	ADC 阈值比较的极性 1 = ADC 结果的高 8 位 < ADCMPH[7:0]，ADCMPO 为 1 0 = ADC 结果的高 8 位 ≥ ADCMPH[7:0]，ADCMPO 为 1	ADCON3[6]		RW-0
ADCMPEN	ADC 阈值比较 1 = 使能 0 = 关闭	ADCON3[5]		RW-0
LEBADT	前沿消隐周期结束后，触发 ADC 转换 1 = 触发 0 = 不触发 注：≥G 版芯片为保留位	ADCON3[3]		RW-0

表 16-2 ADC 相关用户寄存器

名称	状态		寄存器	地址	复位值
GIE	全局中断 1 = 使能 (PEIE, ADCIE, ACMPIE 适用) 0 = 全局关闭 (唤醒不受影响)		INTCON[7]	Bank 首地址 + 0x0B	RW-0
PEIE	外设总中断	1 = 使能 (ADCIE, ACMPIE 适用) 0 = 关闭 (无唤醒)	INTCON[6]		RW-0
ADCIE	ADC 转换完成中断	1 = 使能 0 = 关闭 (无唤醒)	PIE1[0]	0x8C	RW-0
ADCIF	ADC 转换完成中断标志位	1 = Yes (锁存) 0 = No	PIR1[0]	0x0C	RW0-0
ACMPIE	ADC 阈值比较匹配中断	1 = 使能 0 = 关闭 (无唤醒)	PIE1[4]	0x8C	RW-0
ACMPIF	ADC 阈值比较匹配中断标志位	1 = Yes (锁存) 0 = No	PIR1[4]	0x0C	RW0-0

表 16-3 ADC 中断使能和状态位

16.2. ADC 的配置

配置和使用 ADC 时，必须考虑以下功能：

- 校准 ADC
- 端口配置
- 通道选择
- 触发方式选择
- 触发源选择
- 触发类型选择
- 触发延时配置
- ADC 参考电压的选择
- ADC 转换时钟源
- 中断控制
- 转换结果的格式
- 阈值比较

注意：在进行各项配置更改的时候，需要确保 AD 转换并未正在进行或外部触发功能未开启。建议在 ADON 关闭时进行更改。

16.2.1. 端口配置

ADC 可用于转换模拟和数字信号。转换模拟信号时，应将相关的 TRIS 和 ANSEL 位置 1 将 I/O 引脚应配置为模拟功能。更多信息请参见相应的端口章节。

注意：如果定义为数字输入的引脚上存在模拟电压，会导致输入缓冲器传导过大的电流。

16.2.2. 通道选择

ADCON0 寄存器的 CHS 位决定将哪个通道连接到采样保持电路。改变通道时，根据采样稳定的需要可在启动转换前加入一定延时。更多信息请参见第 16.3 节“ADC 的工作原理”。

16.2.3. 触发方式选择

ADCON0 寄存器的 ADEX 位决定是否使用外部触发信号。

若 ADEX=0 时，ADGO 位可由程序置位，AD 转换完成自动清零。

若 ADEX=1 时，ADGO 位将由外部硬件触发置位，AD 转换完成清零。

注意：
G 版之前：若选择了前沿消隐触发 ADC，即 LEBADT 设为 1 时，需要先置位 ADEX 和 ADON。
≥G 版：若选择了前沿消隐触发 ADC，即 EDGS 设为 1 时，需要先置位 ADEX 和 ADON。

16.2.4. 触发源选择

在设定 ADEX 后，ADCON2 寄存器的 ETGSEL 位决定使用哪个外部触发信号。其中可选 I/O 引脚触发，需要配置相关寄存器。具体请参见相应的端口章节。

16.2.5. 触发类型选择

ADCON2 寄存器的 ETGTYP 位决定外部触发信号的触发类型。

16.2.6. 触发延时配置

ADCON2 寄存器的 ADDLY.8 位和 ADDLY 寄存器组成 9 位延时计数器，共同决定外部触发信号的触发延时时间。由于需要同步异步信号，实际延迟时间为：$(ADDLY+6)/F_{ADC}$，≥G 版为$(ADDLY+3)/F_{ADC}$。

注意：若选择了前沿消隐触发功能时，则实际延迟时间为：$(ADDLY+4)/F_{T2CK}+2*F_{ADC}$，≥G 版为$(ADDLY+3)/F_{T2CK}$。T2CK 为 Timer2 的时钟源，由 T2CKSRC 寄存器位选择。

16.2.7. ADC 参考电压

ADCON1 寄存器的 ADPREF 位提供对正参考电压的控制，ADNREF 位提供对负参考电压的控制。正/负参考电压可以是内部参考电压、VDD/GND、内部参考电压加外部电容、外部参考电压。正/负参考电压可以有各种组合，但不可以同时选择内部参考电压。若发生则强制负参考电压选择 GND。

ADCON2 寄存器的 ADINTREF 位提供对内部参考电压的控制。内部参考电压可以选择 0.5V、2V、3V 或者悬空。

16.2.8. 转换时钟

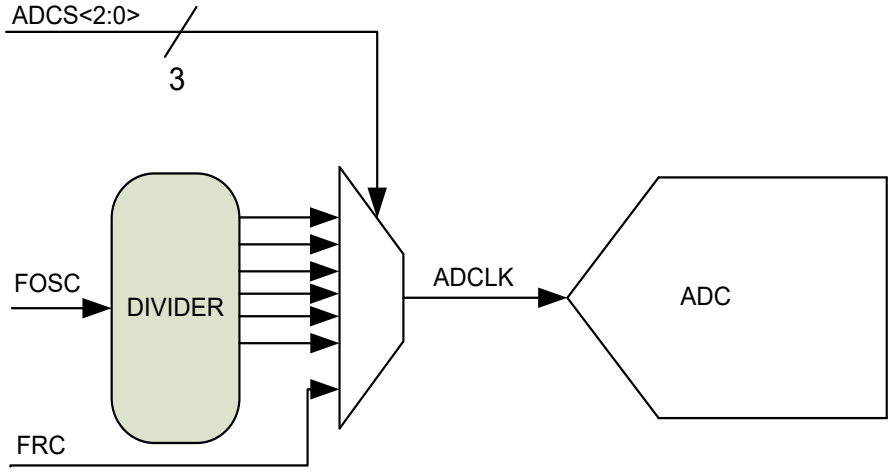


图 16-2 ADC 的时钟配置原理

转换时钟源可通过 ADCON1 寄存器的 ADCS 位用软件选择。有以下 8 种时钟选项：

- F_{SysClk}
- $F_{SysClk}/2$
- F_{SysClk} /4
- F_{SysClk} /8
- F_{SysClk} /16
- F_{SysClk} /32
- F_{SysClk} /64
- F_{LIRC} (内部慢时钟振荡器)

完成一位 (bit) 的转换时间定义为 T_{AD}。完成 12 位转换需要 12*T_{AD} 周期，还需要 2~3*T_{AD} 的转换启动时间和 1*T_{AD} 的数据传输处理时间，如

图 16-3 所示。

进行正确的转换必须满足相应的 T_{AD} 规范。更多信息请参见第 **27.10** 节“芯片的电气特性”中的 12bit ADC 要求。**表 16-4** 所示为正确选择 ADC 时钟的示例。

ADC 时钟周期 (T_{AD})		系统时钟频率 (Sysclk, 2T)			
ADC 时钟源	ADCS[2:0]	16MHz	8MHz	4MHz	1MHz
F_{SysClk}	011	62.5ns	125ns	250ns	1.0μs
F_{SysClk} /2	000	125ns	250ns	500ns	2.0μs
F_{SysClk} /4	100	250ns	500ns	1.0μs	4.0μs
F_{SysClk} /8	001	0.5μs	1.0μs	2.0μs	8.0μs
F_{SysClk} /16	101	1.0μs	2.0μs	4.0μs	16.0μs
F_{SysClk} /32	010	2.0μs	4.0μs	8.0μs	32.0μs
F_{SysClk} /64	110	4.0μs	8.0μs	16.0μs	64.0μs
F_{LIRC}	x11	4.0μs	4.0μs	4.0μs	4.0μs

表 16-4 ADC 时钟周期和器件工作频率

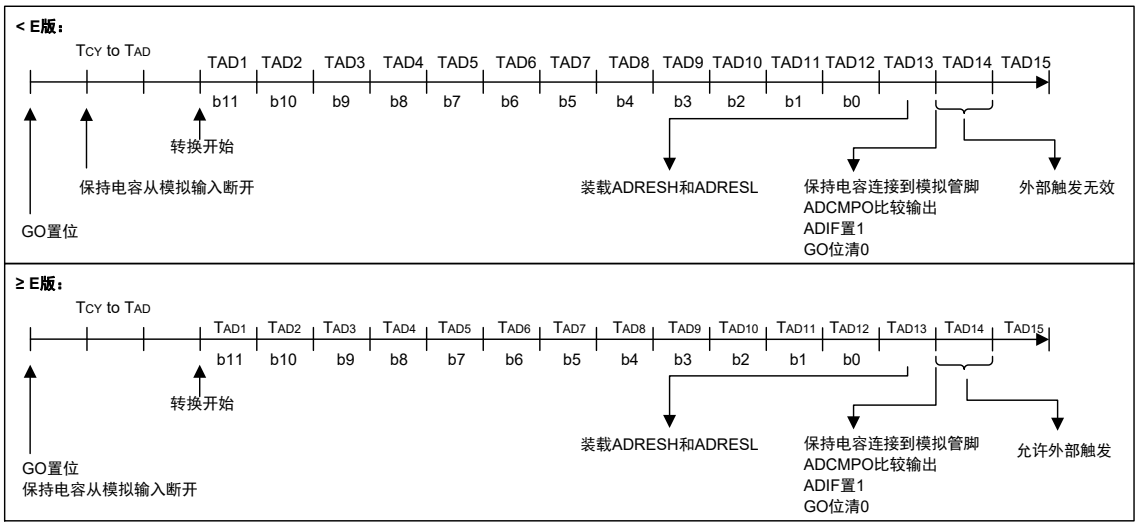


图 16-3 模数转换 T_{AD} 周期

注意：

1. 除非使用的是 F_{LIRC}，否则任何系统时钟频率的变化均会改变 ADC 时钟频率，这将对 ADC 结果产生负面影响；
2. F_{LIRC} 可以是 256kHz 或者是 32kHz，取决于 LFMOD 为何值；

16.2.9. 中断

ADC 模块可使中断在模数转换完成时产生，或者通过转换完成的阈值比较产生。ADC 转换中断标志为 PIR1 寄存器中的 ADIF 位。ADC 中断使能为 PIE1 寄存器中的 ADIE 位。ADIF 位必须用软件置 1 清零。

ADC 的阈值比较中断标志为 PIR1 寄存器中的 ACMPIF。ADC 阈值比较中断使能为 PIE1 寄存器中的 ACMPIE 位。

注意： 1、无论 ADC 中断是否被打开，ADIF 位在每次正常转换完成时均置 1。

2、软件停止 AD 转换都不会置位 ADIF。

3、仅当在选择了 F_{LIRC} 振荡器，ADC 才能在休眠期间工作。

器件工作或处于休眠状态时均可产生中断。如果器件处于休眠状态，中断可唤醒器件。从休眠唤醒时，始终执行 SLEEP 指令后的那条指令。如果用户试图唤醒器件并恢复顺序执行代码，必须禁止全局中断。如果允许全局中断，代码执行将转至中断服务程序。

16.2.10. 转换结果的格式

12 位 A/D 转换结果有两种格式，即左对齐和右对齐。ADCON1 寄存器的 ADFM 位控制输出格式。
AD 自动校准值也受输出格式影响。

两种输出格式：

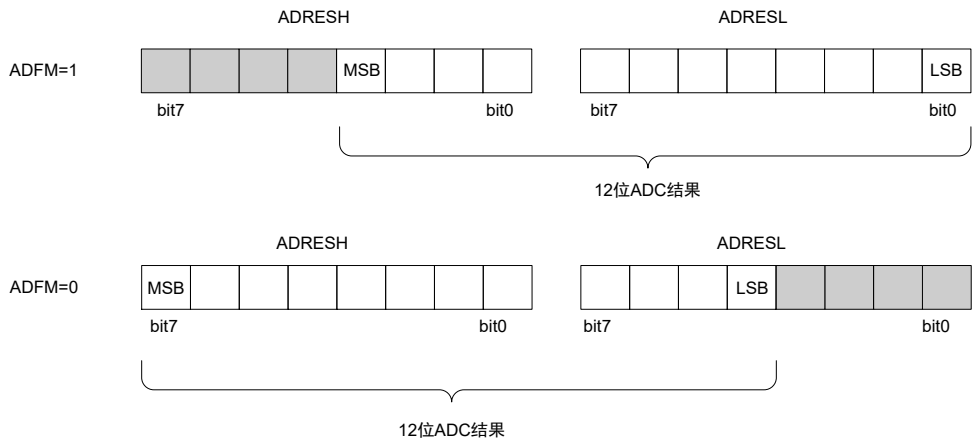


图 16-4 ADC 转换结果格式示意图

16.2.11. 阈值比较

ADCMPH 寄存器为 ADC 结果比较阈值，ADCON3 寄存器的 ADCMPE 位控制比较功能使能，ADCMPP 位控制比较极性，ADCMPO 指示比较结果。

AD 可以在每次转换完成时进行比较。比较结果会一直保持，直到下次转换完成被更新。ADCMPE 或 ADON 的清零可以关闭比较功能或 AD 模块，同时可以清零 ADCMPO。进入睡眠不会清零 ADCMPO。

在每次比较完成时可以产生故障刹车事件，由 ADCON3 寄存器的 ADFBEN 控制。

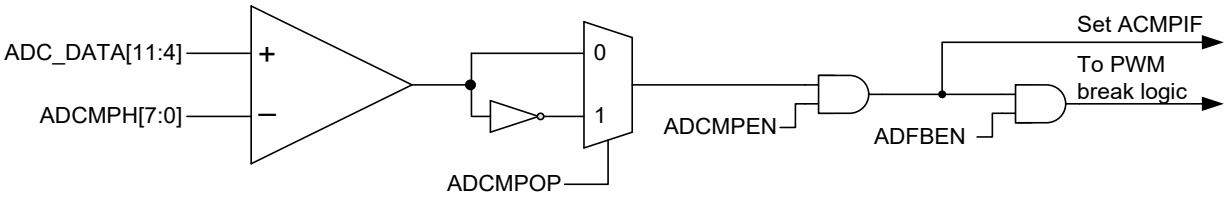


图 16-5 ADC 阈值比较功能框图

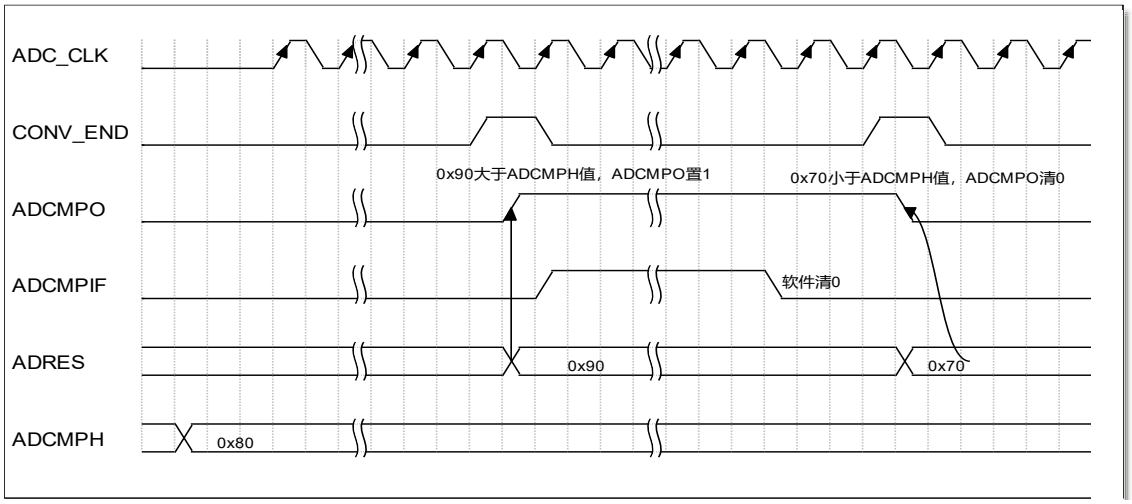


图 16-6 ADC 阈值比较结果产生时序

注意：

1. ADCMPO 是内部信号，对软件不可见，但软件可以通过 ADCMPIF 位间接判断 ADCMPO 的值。ADC 控制模块在转换结束时刻判断 ADCMPO 的值，当 ADCMPO 为 1 时，ADCMPIF 被置 1。

16.3. ADC 的工作原理

16.3.1. 启动转换

要使能 ADC 模块，必须将 ADCON0 寄存器的 ADON 位置 1。

若 ADEX=0 时，将 ADCON0 寄存器的 GO/DONE 位置 1 将启动 AD 转换,。

若 ADEX=1 时，需要外部触发信号才能启动，并且硬件置位 GO/DONE 位，程序置位 ADGO 无效。

注意：

1. 不应在打开 ADC 的那条指令中将 GO/DONE 位置 1。请参见第 16.3.7 节“A/D 转换步骤”
2. 不应在启动 ADC 转换后或等待外部触发时更改 AD 配置。
3. 置位 ADGO 后需要等待一个系统周期才可读回 ADGO 标志。

16.3.2. 转换完成

转换完成时，ADC 模块将：

- 将 GO/DONE 位清零
- 将 ADIF 标志位置 1
- 用新的转换结果更新 ADRESH:ADRESL 寄存器

16.3.3. 终止转换

当 ADEX=0，ADC 处于软件触发状态，如果转换必须在完成前被终止，可用软件将 GO/DONE 清零。那么 ADC 将不会更新这次转换数据。当 ADEX=1，ADC 处于硬件触发状态，如果需要终止转换，则需要将 ADON 置 0，关闭 ADC 的使能开关。

注意：器件复位将强制所有寄存器回到其复位状态。这样，ADC 模块就被关闭，并且任何待处理的转换均被终止。

16.3.4. 休眠模式下 ADC 的工作

ADC 模块可在休眠期间工作，这要求将 ADC 时钟源置于 F_{LIRC} 选项。

ADC 需要等待 $3*T_{AD}$ 后才开始转换。这允许软件在设置 ADGO 后，执行一个 SLEEP 指令置 MCU 于 SLEEP 模式，从而降低 ADC 转换期间的系统噪声。通过配置 ADC 时钟为 F_{LIRC}，可进一步降低系统噪声。

如果允许 ADC 中断，转换完成后器件将从休眠唤醒。如果禁止 ADC 中断，ADC 模块在转换完成后关闭，尽管 ADON 位保持置 1 状态。

如果 ADC 时钟源不是 F_{LIRC}，执行一条 SLEEP 指令将使当前转换强制中止，ADC 模块被直接关闭，尽管 ADON 位保持置 1 状态。

16.3.5. A/D 转换步骤

以下是使用 ADC 进行模数转换的步骤示例：

1. 配置端口：
   - 禁止引脚输出驱动器（见 TRIS 寄存器）
   - 将引脚配置为模拟
2. 配置 ADC 模块：
   - 选择 ADC 转换时钟
   - 配置参考电压
   - 选择 ADC 输入通道
   - 选择转换结果的格式
   - 打开 ADC 模块
3. 配置 ADC 中断（可选）：
   - 将 ADC 中断标志清零
   - 允许 ADC 中断
   - 允许外设中断
   - 允许全局中断
4. 等待所需稳定时间 T_{ST}(1)；
5. 等待所需的采集时间 T_{ACQ}(2)；
6. 将 GO/DONE 置 1 启动转换或等待硬件触发；
7. 等待一个系统周期才可回读 GO/DONE；
8. 通过以下情况之一等待 ADC 转换完成：
   - 查询 GO/DONE 位
   - 等待 ADC 中断（允许中断时）
9. 读取 ADC 结果；
10. 将 ADC 中断标志清零（在允许了中断的情况下这一步是必需的）。

以下是一段示例代码：

```
BANKSEL ADCON1          ;
LDWI B'01110000'        ;ADC Frc clock
STR ADCON1              ;

BANKSEL TRISA           ;
BSR TRISA,0             ;Set RA0 to input

BANKSEL ANSEL           ;
BSR ANSEL,0             ;Set RA0 to analog

BANKSEL ADCON0          ;
LDWI B'10000001'        ;Right justify,
STR ADCON0              ; Vdd Vref, AN0, On

LCALL StableTime        ; ADC stable time
```

```
LCALL SampleTime          ;Acquisiton delay

BSR ADCON0,GO             ;Start conversion
BTSC ADCON0,GO            ;Is conversion done?
LJUMP $-1                 ;No, test again

BANKSEL ADRESH ;
LDR ADRESH,W              ;Read upper 2 bits
STR RESULTHI              ;store in GPR space
BANKSEL ADRESL ;
LDR ADRESL,W              ;Read lower 8 bits

STR RESULTLO              ;Store in GPR space
```

注意：

1. T_{ST} 时间是 ADC 的稳定时间，当使用内部参考时，ADC 首次启动还需要考虑参考电压的稳定时间 T_{VRINT}，等待时间应取两者的较大者，即 max(T_{VRINT}, T_{ST})；
2. 见**图 16-6**，ADC 转换时序；
3. 切换通道后必须等待足够长的 T_{ACQ} 时间，即上面示例中的 SampleTime 必须满足 27.10 节的 T_{ACQ} 时间要求，否则 ADC 精度、线性度将不能保证；

16.3.6. A/D 采集时间要求

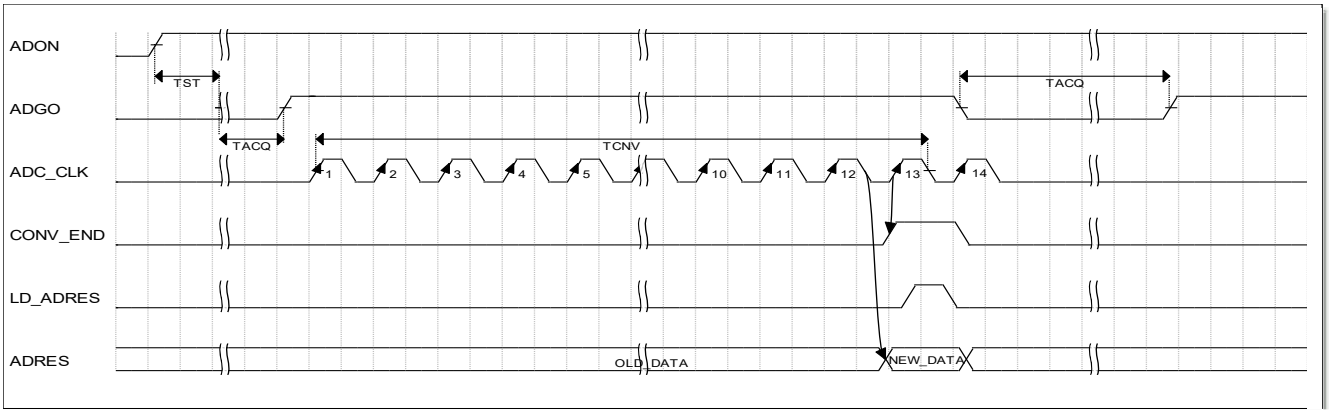
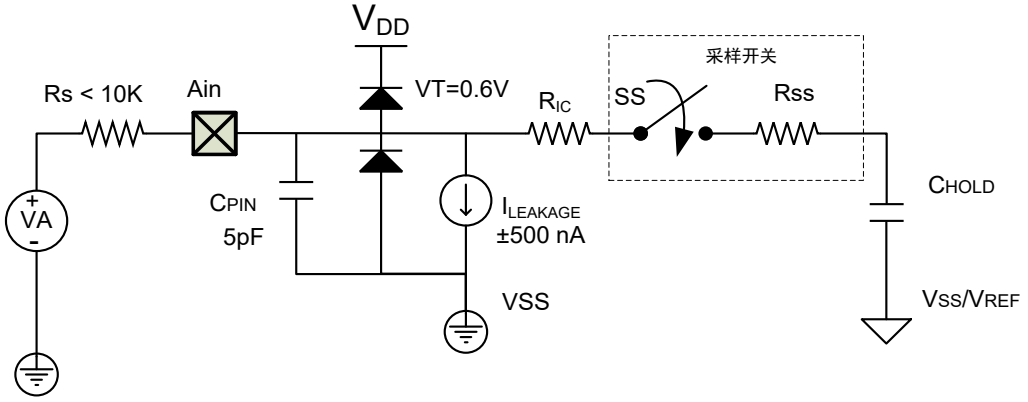


图 16-7 ADC 软件触发转换时序图

为了使 ADC 达到规定的精度，必须使充电保持电容 (CHOLD) 充满至输入通道的电平。模拟输入模型请参见**图 16-8**。源阻抗 (RS) 和内部采样开关 (RSS) 阻抗直接影响电容 CHOLD 的充电时间。采样开关(RSS)阻抗随器件电压 (VDD) 的变化而变化。建议模拟信号源的最大阻抗为 10kΩ。采集时间随着源阻抗的降低而缩短。在选择 (或改变) 模拟输入通道后，必须在开始转换前完成采集。



图注：

CPIN	= 输入电容
VT	=门限电压
ILEAKAGE	=结点漏电流
RIC	=互联电阻
SS	=采样开关
CHOLD	=采样保持电容

图 16-8 模拟输入模型

17. 运放

FT62F28x 内部集成了一个运放，用于对模拟信号进行调理。

运放 0 具有以下特点：

1. 3.64MHz 的单位增益带宽
2. 外部直接或者串 4kΩ 电阻到反相端
3. 输出可选 40kΩ~320kΩ 电阻反馈到反相端
4. 输出可连接外部管脚

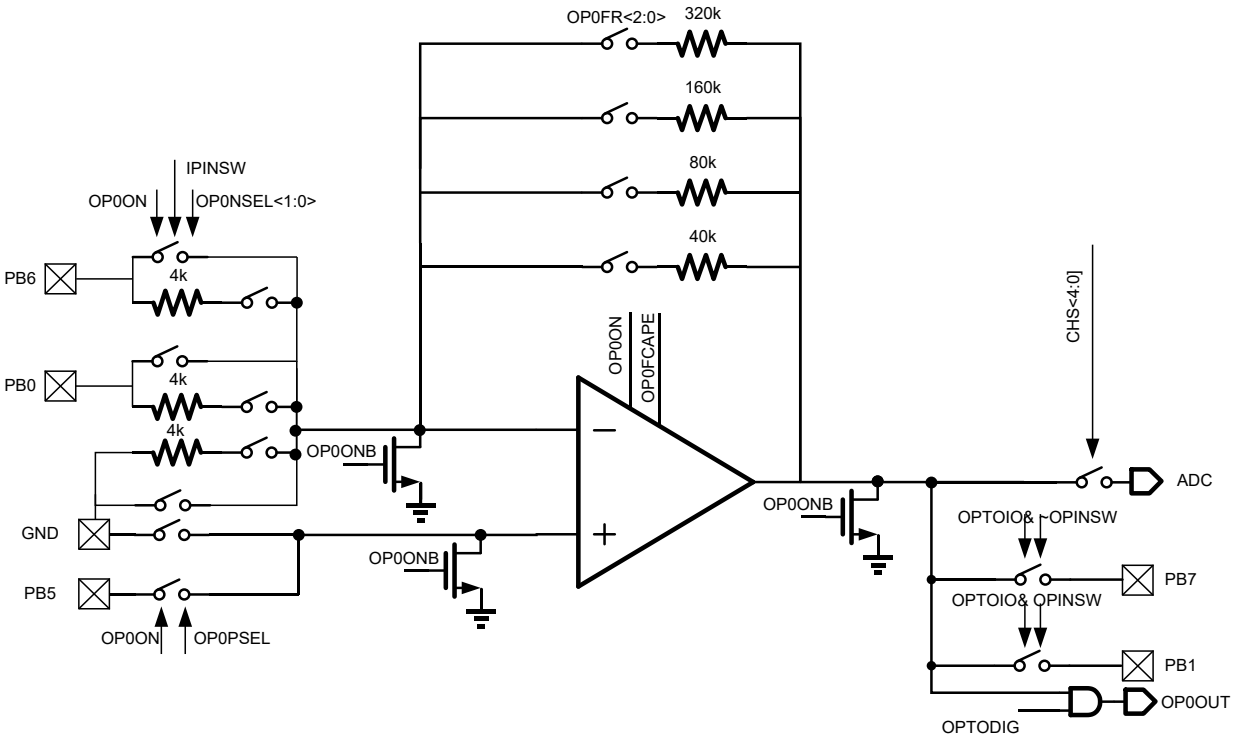


图 17-1 运放结构框图

17.1. 运放 0 相关寄存器汇总

名称	地址	bit 7	bit 6	bit 5	bit 4	bit 3	Bit 2	bit 1	bit 0	复位值
OP0CR0	0x96	OP0OUT	OP0PSEL	OP0NSEL[1:0]		OP0FR[2:0]			OP0ON	x110 1100
OP0CR1	0x97	–	–	OPINSW	IPINSW	OP0FCAPE	–	OPTODIG	OPTOIO	––00 0–00

表 17-1 运放 0 相关寄存器汇总

名称	状态		寄存器	地址	复位值
OP0OUT	运放 0 输出		OP0CR0[7]	0x96	RO–x
OP0PSEL	运放 0 正相端输入选择 1 = GND 0 = PB5		OP0CR0[6]		RW–1
OP0NSEL	运放 0 反相端输入选择 00 = GND 01 = 外部输入管脚 (由 IPINSW 决定) 10 = 串联 4k 电阻连接到外部输入管脚 (由 IPINSW 决定) 11 = 串联 4k 电阻连接到 GND		OP0CR0[5:4]		RW–10
OP0FR	运放 0 反馈电阻选择 0xx = 无反馈网络 100 = 40k 101 = 80k 110 = 160k 111 = 320k		OP0CR0[3:1]		RW–110
OP0ON	运放 0	1 = 使能 0 = 禁止	OP0CR0[0]		RW–0
OPINSW	运放 0 输出到管脚	1 = PB1 0 = PB7	OP0CR1[5]	0x97	RW–0
IPINSW	运放 0 反相端外部输入管脚	1 = PB0 0 = PB6	OP0CR1[4]		RW–0
OP0FCAPE	运放 0 补偿电容	1 = 禁止 (运放用做比较器) 0 = 使能	OP0CR1[3]		RW–0
OPTODIG	运放 0 输出到寄存器	1 = 使能 0 = 禁止	OP0CR1[1]		RW–0
OPTOIO	运放 0 输出	1 = 使能 (输出到管脚) 0 = 禁止	OP0CR1[0]		RW–0

表 17-2 运放 0 相关用户控制寄存器

17.2. 运放 0 典型应用

电流检测：
此时配置为 OP0PSEL 置为 1，OP0NSEL 置为 1，反馈电阻为 160kΩ，运放输出到 ADC。

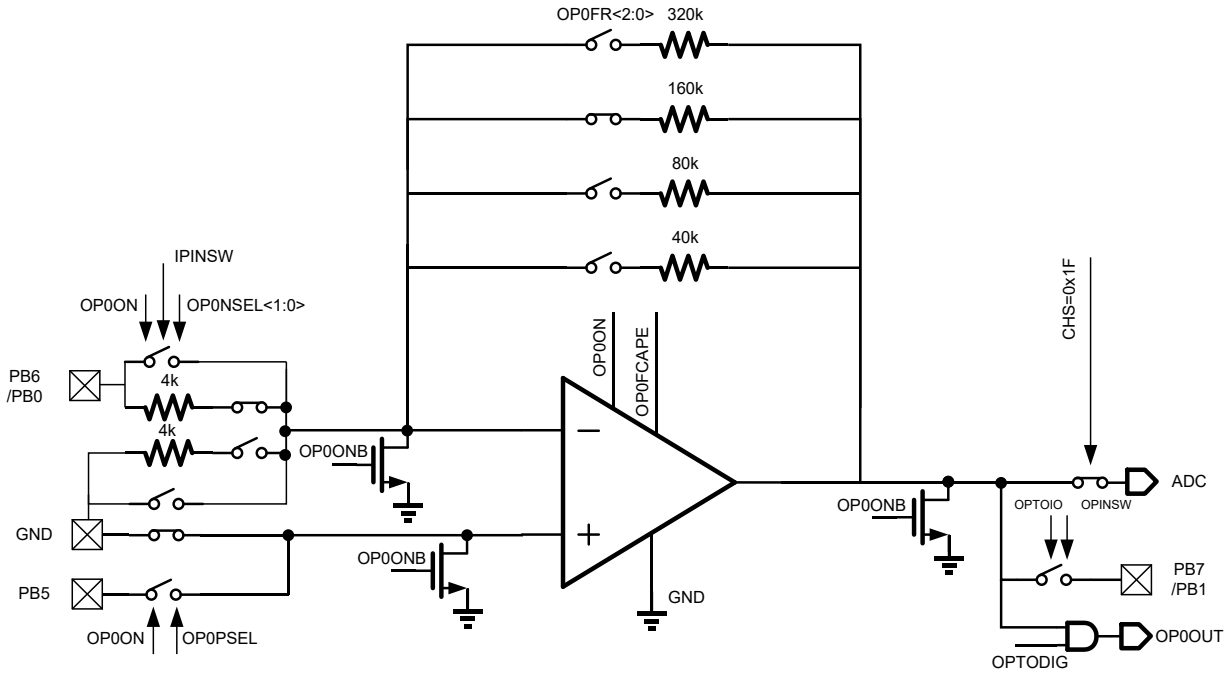


图 17-2 电流检测连接方式 1

17.3. 运放 0 输出

运放 0 的输出 OP0OUT 可以映射到 SFR 给软件访问，也可以输出到外部管脚上，由寄存器位 OPTOIO 和 OPINSW 决定。需要注意的是，运放 0 输出优先级高于其它数字功能，当相关 I/O 配置成 OP0OUT 时，不管 TRISx 为何值，该 I/O 都变为模拟管脚，其斯密特输入自动关闭。

18. 比较器模块

FT62F28x 集成了 2 个模拟比较器，支持以下特性：

- 比较结果可输出到片外
- 输出极性可编程
- 输出可作为中断源 (电平变化中断)
- 输出可作为唤醒源
- 输出可作为 PWM1 刹车源
- 反相输入可连 7bit DAC，或者接运放 0
- 窗口比较模式
- 可编程参考电压

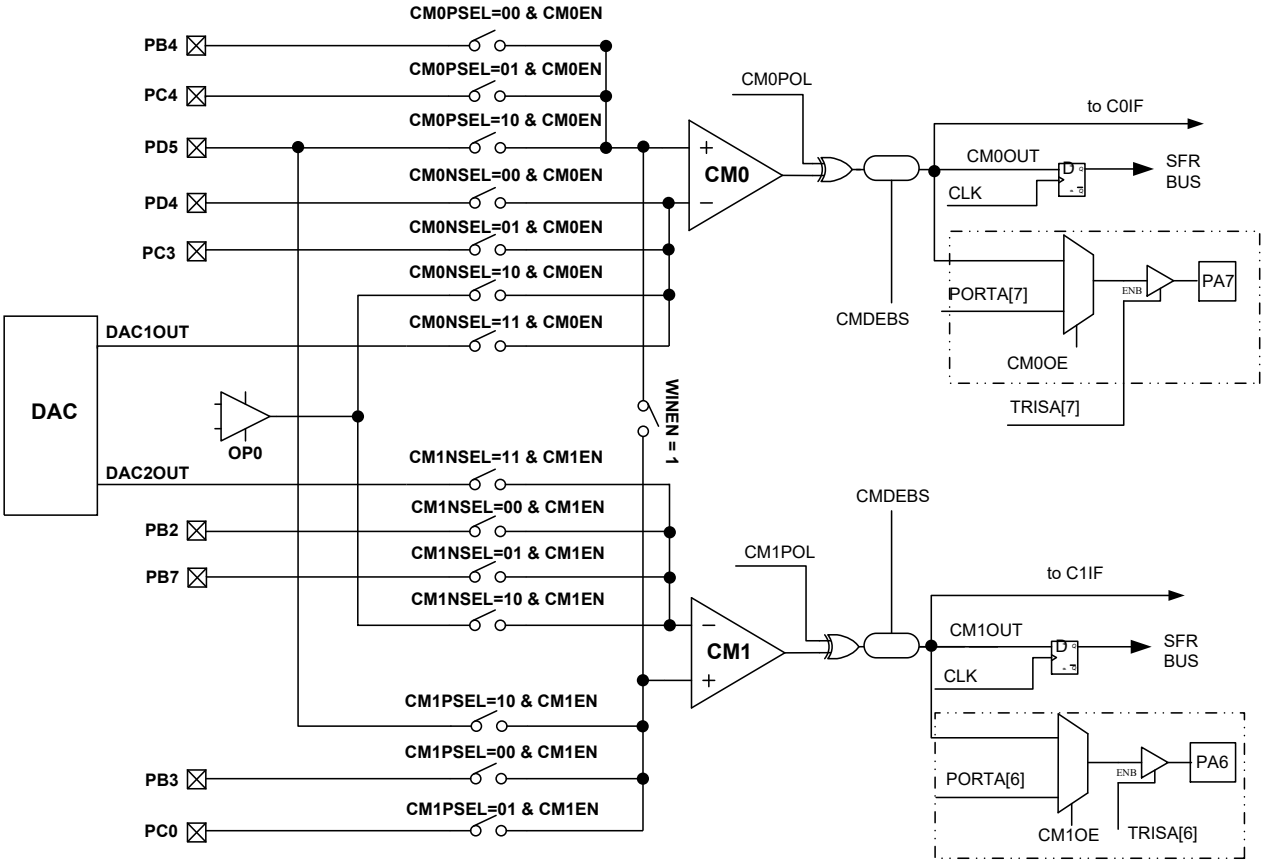


图 18-1 比较器结构框图

18.1. 比较器相关寄存器汇总

名称	地址	bit 7	bit 6	bit 5	bit 4	bit 3	Bit 2	bit 1	bit 0	复位值
DAC1DAT	0x29A	–	DAC1DAT[6:0]							–000 0000
DAC2DAT	0x29B	–	DAC2DAT[6:0]							–000 0000
DACCON0	0x29C	CMDEBS	DEBPR[1:0]		–	WNDEN	DACEN	DACVREF[1:0]		000– 0000
CM0CON0	0x29D	CM0EN	CM0POL	CM0OE	C0OUT	CM0PSEL[1:0]		CM0NSEL[1:0]		000x 0000
CM1CON0	0x29E	CM1EN	CM1POL	CM1OE	C1OUT	CM1PSEL[1:0]		CM1NSEL[1:0]		000x 0000
MSCON2	0x10D	P0ANP	P0AP	CMAUSTR	STRTYP[1:0]		T1EVTS[2:0]			0000 0000

表 18-1 比较器相关寄存器汇总

名称	状态		寄存器	地址	复位值
CM0EN	比较器 0	1 = 使能 0 = 禁止	CM0CON0[7]	0x29D	RW–0
CM0POL	比较器 0 输出 C0OUT 的极性	1 = 取反 0 = 正常	CM0CON0[6]		RW–0
CM0OE	比较器 0 输出到 PA7	1 = 使能 0 = 禁止	CM0CON0[5]		RW–0
C0OUT	比较器 0 比较结果，其值与 CM0POL 有关		CM0CON0[4]		RO–x
CM0PSEL	比较器 0 正相端输入 (00) / (11) = PB4 (01) = PC4 (10) = PD5		CM0CON0[3:2]		RW–00
CM0NSEL	比较器 0 反相端输入 00 = PD4 01 = PC3	10 = 运放 0 输出 11 = DAC1OUT	CM0CON0[1:0]		RW–00
CM1EN	比较器 1 使能位	1 = 使能 0 = 禁止	CM1CON0[7]	0x29E	RW–0
CM1POL	比较器 1 输出 C1OUT 的极性	1 = 取反 0 = 正常	CM1CON0[6]		RW–0
CM1OE	比较器 1 输出到 PA6	1 = 使能 0 = 禁止	CM1CON0[5]		RW–0
C1OUT	比较器 1 比较结果，其值与 CM1POL 有关		CM1CON0[4]		RO–x
CM1PSEL	比较器 1 正相端输入 (00) /(11) = PB3 (01) = PC0 (10) = PD5		CM1CON0[3:2]		RW–00

名称	状态		寄存器	地址	复位值
CM1NSEL	比较器 1 反相端输入 00 = PB2 10 = 运放 0 输出 01 = PB7 11 = DAC2OUT		CM1CON0[1:0]	0x29E	RW-00
CMDEBS	比较器 0/1 的去抖选择	1 = 比较器 1 去抖 0 = 比较器 0 去抖	DACCON0[7]	0x29C	RW-0
DEBPR	比较器 0/1 的去抖时间 00 = 不去抖 01 = 1~2 个去抖时钟周期 10 = 2~3 个去抖时钟周期 11 = 3~4 个去抖时钟周期		DACCON0[6:5]		RW-00
WNDEN	窗口比较模式	1 = 使能 0 = 关闭	DACCON0[3]		RW-0
DACEN	DAC	1 = 使能 0 = 禁止	DACCON0[2]		RW-0
DACVREF	DAC 参考电压 00 = 2V 10 = 4V 01 = 3V 11 = VDD		DACCON0[1:0]		RW-00
DAC1DAT	DAC1 通道输出电压 $V_{DAC1OUT} = V_{DACVREF}$*(DAC1DAT[6:0]+1)/128 (0x20 ≤ DAC1DAT[6:0] ≤ 0x4F)		DAC1DAT[6:0]	0x29A	RW-000 0000
DAC2DAT	DAC2 通道输出电压 $V_{DAC2OUT} = V_{DACVREF}$*(DAC2DAT[6:0])/128 (0x00 ≤ DAC2DAT [6:0] ≤ 0x58)		DAC2DAT[6:0]	0x29B	RW-000 0000
CMAUSTR	比较器 CMx 的自动开启 1 = 使能（当 LVDIF 置 1 时，CMxEN 自动置 1） 0 = 禁止		MSCON2[5]	0x10D	RW-0
STRTYP	比较器 CMx 自动开启类型(仅 CMAUSTR = 1 时有效) 00 = CM0EN 被 LVD 中断事件置 1 01 = CM1EN 被 LVD 中断事件置 1 10 = SLEEP 模式，LVD 中断唤醒 CPU 后置位 CM0EN 11 = SLEEP 模式，LVD 中断唤醒 CPU 后置位 CM1EN		MSCON2[4:3]		RW-00

表 18-2 比较器相关用户控制寄存器

名称	状态		寄存器	地址	复位值
GIE	全局中断 1 = 使能 (PEIE,C0IE, C1IE 适用) 0 = 全局关闭 (唤醒不受影响)		INTCON[7]	Bank 首地址 +0x0B	RW-0
PEIE	外设总中断 1 = 使能 (C0IE,C1IE 适用) 0 = 关闭 (无唤醒)		INTCON[6]		RW-0
C1IE	比较器 1 中断	1 = 使能	PIE2[1]	0x89	RW-0
C0IE	比较器 0 中断	0 = 关闭(无唤醒)	PIE2[0]		RW-0
C1IF	比较器 1 中断标志位	1 = Yes (锁存)	PIR2[1]	0x09	RW0-0
C0IF	比较器 0 中断标志位	0 = No	PIR2[0]		RW0-0

表 18-3 比较器中断使能和状态位

18.2. 比较器控制

18.2.1. 比较器输入

CM0 和 CM1 的输入可以来自多个外部管脚，或者可以连接到内部的 DAC、运放输出，取决于 CMxNSEL 和 CMxPSEL。

除了可以被配置为单比较器使用外，两个比较器还可以配置成窗口比较器，这时需要把 WNDEN 置 1，CM0OE 和 CM1OE 置 1。

18.2.2. 比较器输出

软件可以通过读 CMxCON 的 CxOUT 位得到当前的比较器输出，也可以通过把 CMxOE 置 1，把比较器输出到外部的管脚上。

比较器输出极性可配置，当 CMxPOL 为 1 时，效果等同于把 CMx+和 CMx-端交换位置。

注意：

1. 必须把相关管脚的 TRISx 位清 0，CxOUT 才能输出到管脚上；
2. CxOUT 属于数字信号，配置为输出时，相关 I/O 的斯密特输入不会关闭。

比较器输出状态与输入和极性的关系：

输入条件	CMxPOL	CxOUT
CMx+ > CMx-	0	1
CMx+ < CMx-	0	0
CMx+ > CMx-	1	0
CMx+ < CMx-	1	1

18.2.3. 比较器输出去抖

为防止比较电压的不稳定而导致的毛刺，比较器可以配置为去抖输出，两个比较器共用一个去抖模块，由 CMDEBS 位决定。去抖时钟源与 LVD 的去抖时钟一致，且有 4 种去抖周期可选。

注意：硬件只实现了单边去抖功能，即当 CxPOL 为 0 时，只对高脉冲毛刺过滤，而当 CxPOL 为 1 时，只对低脉冲毛刺过滤。

注意：睡眠时若要使用去抖功能，需要把 SYSON 置 1。

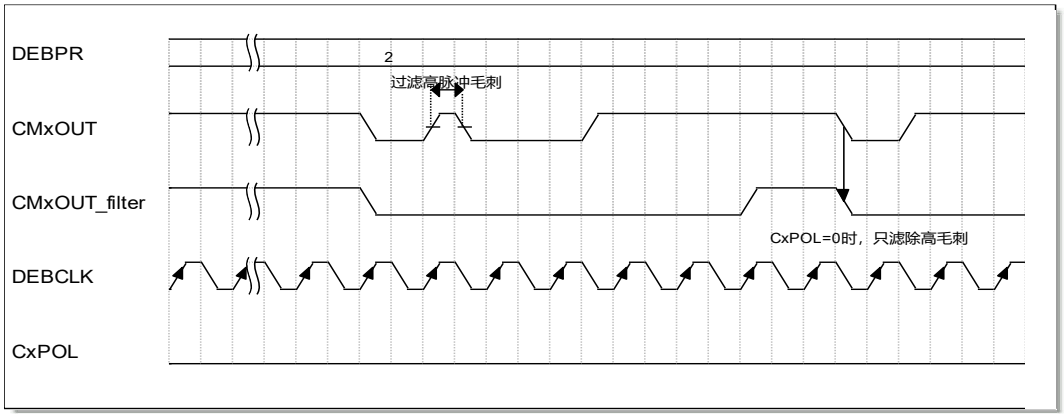


图 18-2 比较器输出去抖时序图

18.3. 比较器中断

当比较器输出由低变化到高时，相应置的中断标志位就被 1。比较器中断为上升沿中断，电路原理如下图所示，它受极性配置位 CMxPOL 影响。

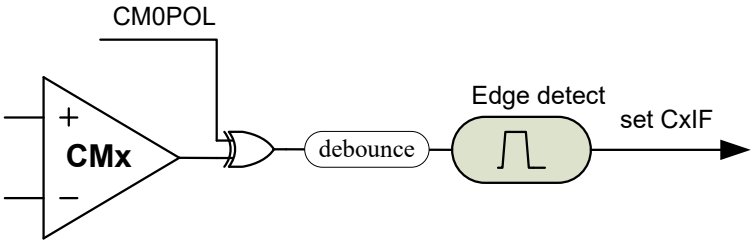


图 18-3 比较器中断检测原理图

18.4. 睡眠状态下的工作

MCU 处于睡眠下，比较器将维持之前的状态，并可以产生相关中断唤醒。

18.5. 比较器的自动使能

比较器 CM0、CM1 可以和 LVD 配合使用，达到级联启动，节省功耗的目的。在一些应用场合可以先让低速低功耗的 LVD 工作，待 LVD 中断事件发生后，自动使能高速高功耗的比较器。其启动关系如下面的时序所示。

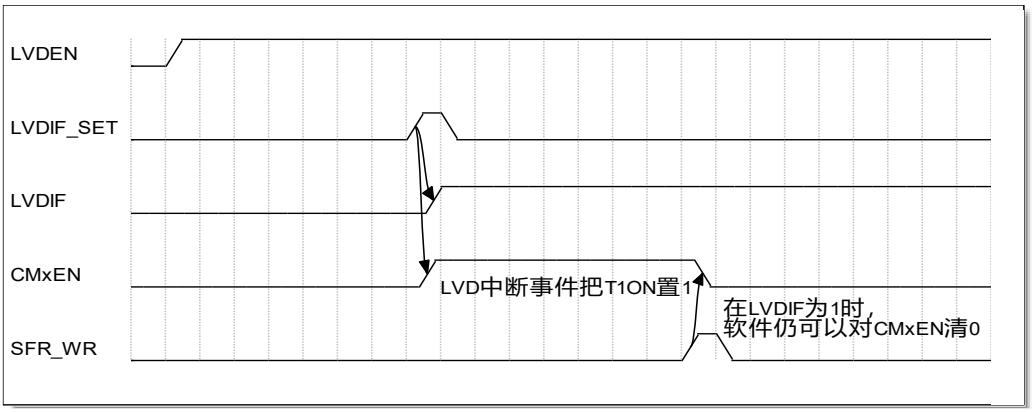


图 18-4 比较器 CMx 自动开启

比较器的自动使能功能由 MSCON2 寄存器的 CMAUSTR 和 STRTYP 位控制，CMxEN 除了在发生 LVD 中断事件后立即置位外，还可以配置为等 CPU 唤醒后再置位，以适应不同的需要。

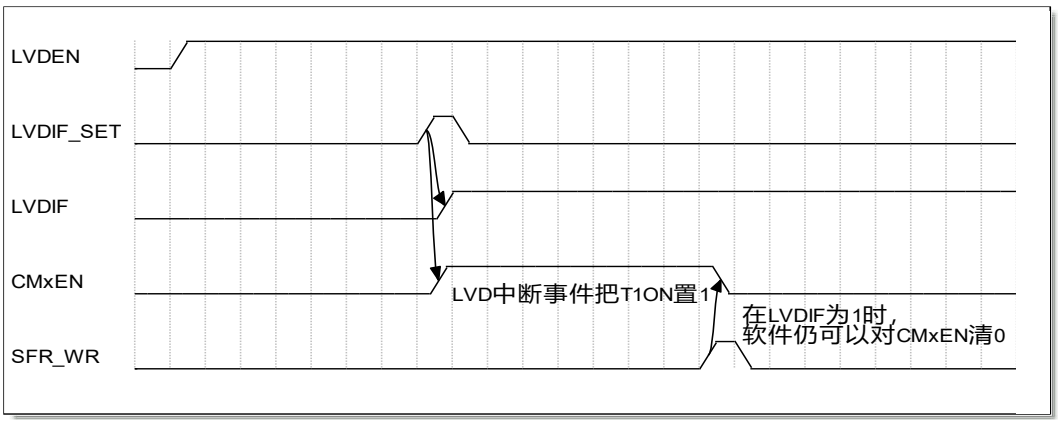


图 18-5 比较器 CMx 在睡眠下的自动开启

18.6. 参考电压

比较器 0/1 的反相输入可以连接到一个内部参考电压，该电压由一个 DAC 模块提供，DAC 有以下特性：

- 可选的参考电压，独立于 VDD：2V/3V/4V/VDD
- 两个 128 级电压范围

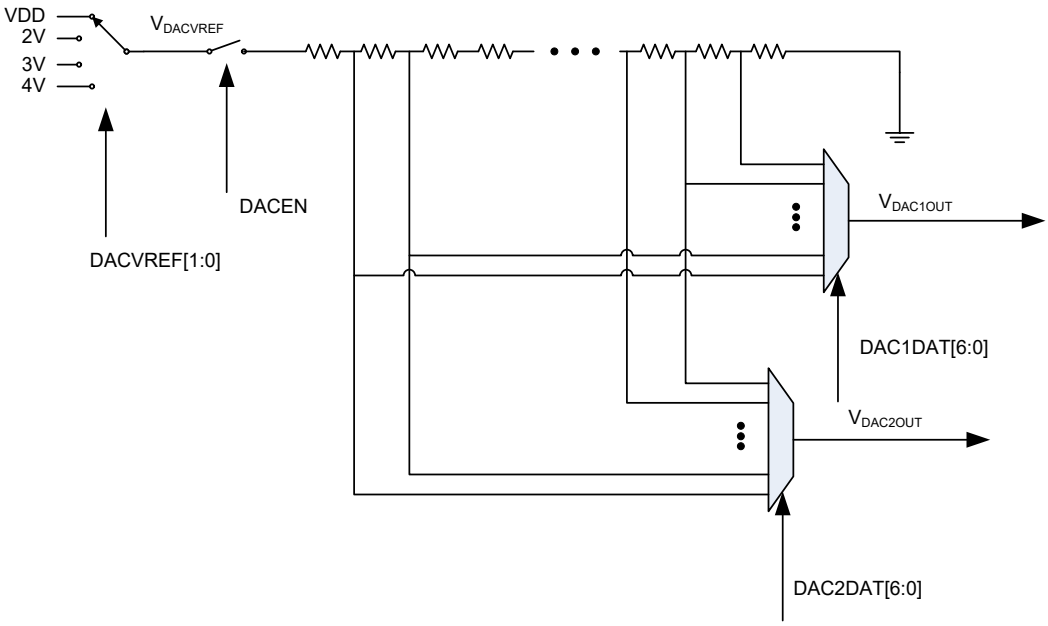


图 18-6 参考电压原理框图

18.6.1. 输出电压

DAC 两个通道的输出电压由以下公式决定：

$$V_{DAC1OUT} = V_{DACVREF} * (DAC1DAT[6:0] + 1) / 128 \quad (0x20 \le DAC1DAT[6:0] \le 0x4F)$$

$$V_{DAC2OUT} = V_{DACVREF} * (DAC2DAT[6:0]) / 128 \quad (0x00 \le DAC2DAT[6:0] \le 0x58)$$

19. SPI 接口

SPI 接口可通过 SPI 协议与外部设备进行通信，特性如下：

- 3 线全双工同步传输
- 2 线半双工同步传输，或单向传输
- 主机模式或从机模式操作
- nss pin 软件或硬件管理
- 可编程的同步时钟极性和相位控制
- 可编程的 LSB first 或 MSB first
- 配置模式错误和 overrun 标志
- 硬件 CRC 校验支持
- Wakeup 唤醒支持

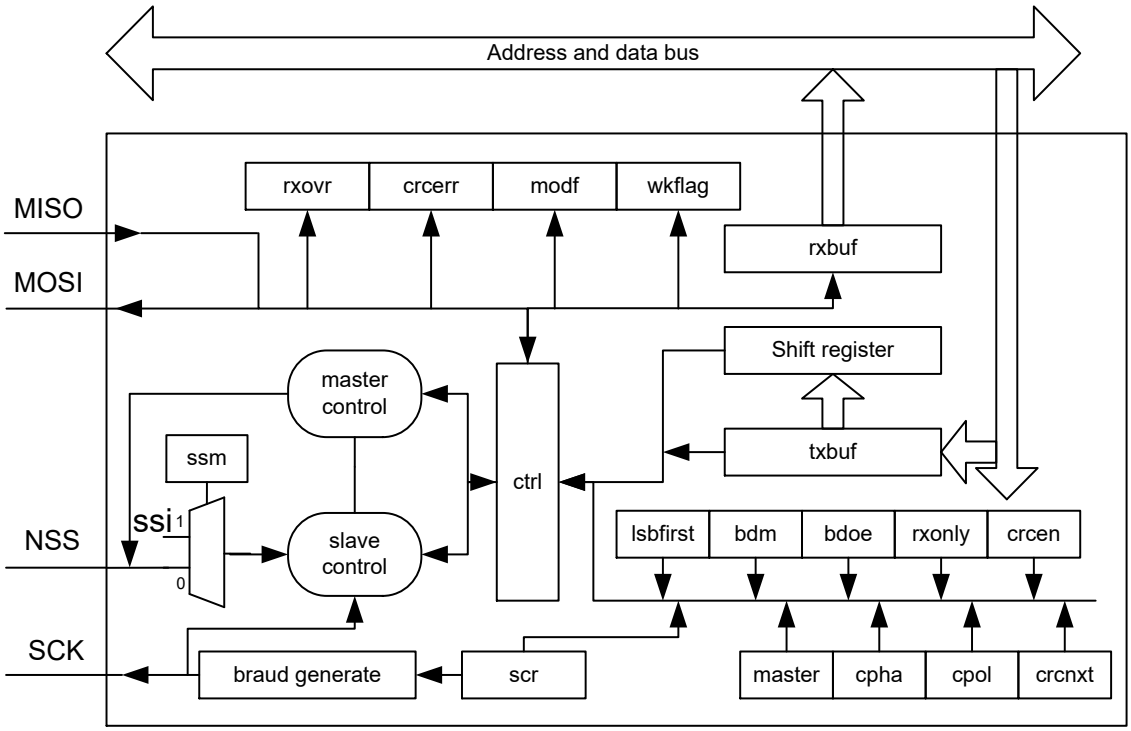


图 19-1 SPI 结构框图

19.1. SPI 相关寄存器汇总

名称	状态	寄存器	地址	复位值
SPIDAT	数据发送/接收 BUF (TXBUF/RXBUF) 写时：将新数据写入到 TXBUF 中 读时：返回 RXBUF 中未读的数据	SPIDAT[7:0]	0x18C	RW-0000 0000
SPIF [1]	数据传输完成标志 1 = 完成 (锁存) 0 = 未完成，或已被清零	SPISTAT[7]	0x195	RW0-0
WCOL [1]	BUF 写入失败(非空时写入)标志 1 = 失败 (锁存) 0 = 正常	SPISTAT[6]		RW0-0
NSSM	NSS 引脚模式选择 00 = 禁用 01 = 输入 (输入值 NSSVAL 与 SSM，端口电平及 SSI 有关) 1x = 输出 (输出值 = NSSM[0])	SPICR0[3:2]	0x18D	RW-01
SPIEN	SPI 接口 1 = 使能 0 = 关闭	SPICR0[0]		RW-0
SPIBUSY	SPI 状态 1 = 忙碌中 0 = 空闲	SPICFG[7]	0x18E	RO-0
MSTEN	工作模式 1 = 主机模式 (MASTER) 0 = 从机模式 (SLAVE)	SPICFG[6]		RW-0
CPHA	SCK 相位选择 (数据采样点) 1 = 第 2 个时钟转换沿 0 = 第 1 个时钟转换沿	SPICFG[5]		RW-0
CPOL	SCK 极性选择 (SPI 空闲时，SCK 时钟状态) 1 = 高电平 0 = 低电平	SPICFG[4]		RW-0
SLAS	从机 (SLAVE) 选中标志位 1 = 被选中 0 = 未被选中	SPICFG[3]		RO-0
NSSVAL	NSS 引脚输入值 当 SSM=0 时，NSSVAL= NSS 引脚端口电平值 当 SSM=1 时，NSSVAL= SSI	SPICFG[2]		RO-1

[1] 写 0 清零，写 1 无效；

名称	状态	寄存器	地址	复位值
SRMT	内部串行移位寄存器状态 1 = 空 0 = 非空	SPICFG[1]		RO-1
SYSON	睡眠模式，系统时钟控制 1 = 保持运行 0 = 关闭	LVDCON[7]	0x21A	RW-0
SPISCR	SCK 波特率设置（仅主机模式有效） 波特率 = Fmaster/(2*(SPISCR+1)) (SPI 外设时钟 Fmaster = Sysclk)	SPISCR[7:0]	0x18F	RW-0000 0000
BDM	半双工 1 = 使能 0 = 关闭	SPICR1[7]	0x194	RW-0
BDOE	半双工工作模式 1 = 发送 0 = 接收	SPICR1[6]		RW-0
RXONLY	全双工工作模式 1 = 只允许接收 0 = 允许发送和接收	SPICR1[5]		RW-0
SSI	NSS 引脚软件输入值（仅当 SSM = 1 时有效） 1 = 输入值为 1 0 = 输入值为 0	SPICR1[4]		RW-0
SSM	从机模式下，NSS 引脚输入值管理 1 = 软件 0 = 硬件	SPICR1[3]		RW-0
CRCNXT	发送 TXCRC 到 TXBUF 1 = 发送（完成后自动清零） 0 = 不发送	SPICR1[2]		RW-0
CRCEN	硬件 CRC 校验模块 1 = 使能 0 = 关闭	SPICR1[1]		RW-0
LSBFIRST	数据传输格式 1 = 优先发送低比特位（LSB） 0 = 优先发送高比特位（MSB）	SPICR1[0]		RW-0
SPICRCP	CRC 计算多项式 (默认值: 0x07)	SPICRCP[7:0]	0x190	RW-0000 0111
SPIRXCRC	接收数据的 CRC 计算结果 (CRCEN 由 0 变 1，此寄存器自动清零)	SPIRXCRC[7:0]	0x191	RO-0000 0000
SPITXCRC	发送数据的 CRC 计算结果 (CRCEN 由 0 变 1，此寄存器自动清零)	SPITXCRC[7:0]	0x192	RO-0000 0000

表 19-1 SPI 相关寄存器

名称	状态		寄存器	地址	复位值
GIE	全局中断 1 = 使能 (PEIE, TXE, RXNE, RXERR, WAKUP 适用) 0 = 全局关闭 (唤醒不受影响)		INTCON[7]	Bank 首地址 +0x0B	RW-0
PEIE	外设总中断 1 = 使能 (TXE, RXNE, RXERR, WAKUP 适用) 0 = 关闭 (无唤醒)		INTCON[6]		RW-0
TXE	发送 BUF 为空中断	1 = 使能 0 = 关闭 (无唤醒)	SPIIER[0]	0x193	RW-0
TXBMT	发送 BUF 为空状态位	1 = 为空 0 = 非空	SPICR0[1]	0x18D	RO-1
RXNE	接收 BUF 不为空中断	1 = 使能 0 = 关闭 (无唤醒)	SPIIER[1]	0x193	RW-0
RXBMT	接收 BUFFER 为空状态位	1 = 为空 0 = 非空	SPICFG[0]	0x18E	RO-1
RXERR	接收错误中断 (工作模式错误，接收溢出，CRC 错误)	1 = 使能 0 = 关闭 (无唤醒)	SPIIER[2]	0x193	RW-0
MODF [2]	工作模式错误标志位	1 = 出错 0 = 正常	SPISTAT[5]	0x195	RW0-0
RXOVRN [2]	接收溢出标志位	1 = 溢出 0 = 正常	SPISTAT[4]		RW0-0
CRCERR [2]	CRC 错误标志位	1 = 发生 CRC 错误 0 = 没有 CRC 错误	SPISTAT[0]		RW0-0
WAKUP	唤醒中断	1 = 使能 0 = 关闭	SPIIER[3]	0x193	RW-0
WKF [2]	WAKEUP 唤醒标志位	1 = 发生唤醒 0 = 未发生唤醒	SPISTAT[1]	0x195	RW0-0

表 19-2 SPI 中断使能和状态位

[2] 写 0 清 0，写 1 无效；

名称	状态		寄存器	地址	复位值
SCKPO	SPI_SCK	1 = PD4 0 = PC6	COMAF0[7]	0x285	RW-0
NSSPO	SPI_NNS	1 = PC6 0 = PC5	COMAF0[6]		RW-0
MOSIPO	SPI_MOSI	1 = PC0 0 = PD5	COMAF0[5]		RW-0
MISOPO	SPI_MISO	1 = PB4 0 = PC7	COMAF0[4]		RW-0
ODCONA	PORTA 开漏	1 = 使能 0 = 关闭	ODCONA[7:0]	0x205	RW-0000 0000
ODCONB	PORTB 开漏		ODCONB[7:0]	0x206	RW-0000 0000
ODCONC	PORTC 开漏		ODCONC[7:0]	0x207	RW-0000 0000
ODCOND	PORTD 开漏		ODCOND[5:0]	0x208	RW-00 0000

表 19-3 SPI 接口引脚控制

名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	复位值
SPIDAT	0x18C	数据发送/接收寄存器 (BUF)SPIDAT [7:0]								0000 0000
SPISTAT	0x195	SPIF	WCOL	MODF	RXOVRN	–	–	WKF	CRCERR	0000 --00
SPICR0	0x18D	–	–	–	–	NSSM		TXBMT	SPIEN	---- 0110
SPICFG	0x18E	SPIBUSY	MSTEN	CPHA	CPOL	SLAS	NSSVAL	SRMT	RXBMT	0000 0111
SPISCR	0x18F	SPISCR [7:0]								0000 0000
SPICR1	0x194	BDM	BDOE	RXONLY	SSI	SSM	CRCNXT	CRCEN	LSBFIRST	0000 0000
SPICRCP	0x190	SPICRCP [7:0]								0000 0111
SPIRXCRC	0x191	SPIRXCRC [7:0]								0000 0000
SPITXCRC	0x192	SPITXCRC [7:0]								0000 0000
SPIIER	0x193	–	–	–	–	WAKUP	RXERR	RXNE	TXE	---- 0000
COMAF0	0x285	SCKPO	NSSPO	MOSIPO	MISOPO	SDAPO[1:0]		SCLPO[1:0]		0000 0000

表 19-4 SPI 相关寄存器地址

19.2. SPI 功能描述

SPI 接口总共有四个引脚：

- NSS:从机选择引脚低电平有效，在 NSS 引脚采用硬件管理用作输入时，如果 NSS 引脚为零则表示选择该从机，如果该接口配置为主机模式，则会产生 MODF 标志位，表示配置错误，这时该接口自动进入从机模式，这种特性用于兼容多主机通信；
- MOSI:主机数据输出/从机数据输入；
- MISO:主机数据输出/从机数据输入；
- SCK:主机串行时钟输出/从机串行时钟输入；

SPI 接口引脚通常的接法是采用三线全双工和四线全双工，如下图所示的引脚接法，NSS 引脚可以根据 NSSM 的值选择作为输入，或者是输出；当 NSS 引脚用作输入时，输入的值取决于 SSM 的值，当 SSM 置 1 时，NSS 引脚的值无效，SSI 的值会被传输到通信控制单元。SPI 模块支持双线半双工模式，主机模式时采用的数据通信引脚是 MOSI，从机模式时采用的数据通信引脚是 MISO。

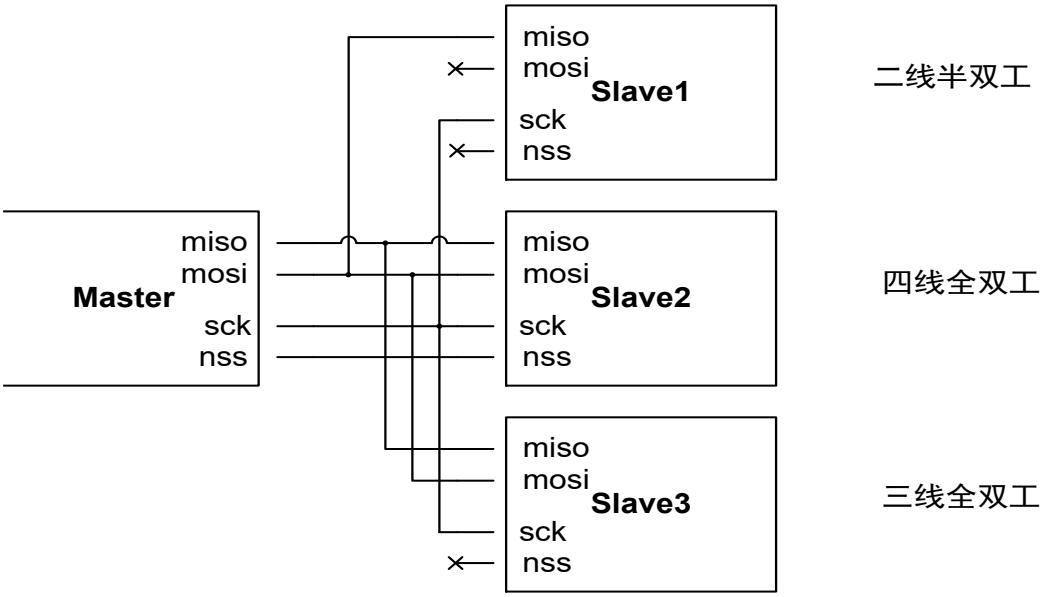


图 19-2 SPI 接口引脚连接示意图

SPI 通信的时钟相位和极性设置如下图所示，时钟和相位的控制总共有四种情况，可以通过 CPHA 和 CPOL 设定相应配置，其中 CPOL 是控制模块空闲时的 SCK 的电平，当 CPOL 为 1 时，SCK 空闲时的电平为高电平，相反为低电平，图中所示箭头的位置是接收数据采样点；当使用了 NSS 引脚时，NSS 为低电平时才会接收数据；发送方串行数据的发送格式可以根据 LSBFIRST 来控制，默认情况下是先发送的八位数据的高位，最后发送的是八位数据的低位。

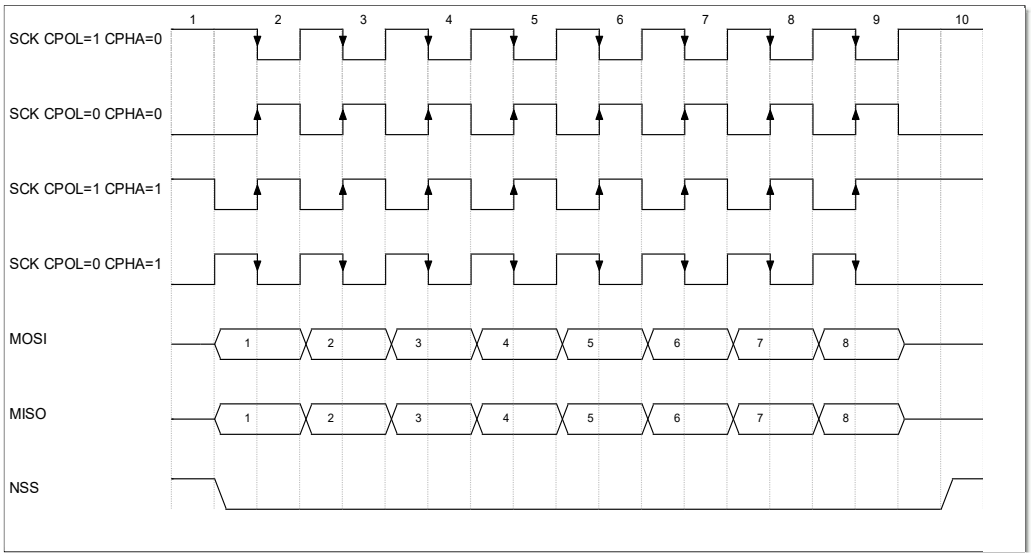


图 19-3 时钟极性和相位时序图

CRC 校验模块用来增强数据传输的可靠性，计算模块初始化的值采用的是零，默认多项式 CRCP 是 0x07，每次 CRCEN 从零到置位的时候都会对 CRC 模块进行初始化（该初始化不会影响 SPICRCP 的值），模块内部中间运算的值会丢失（SPIRXCRC 和 SPITXCRC 的值会被置为零）。当 CRCEN 使能的时候，每次正常写入到 TXBUF 的值都会被送到 CRC 模块用来生成 TXCRC 的值，同样的在接收数据时每次正常写入到 RXBUF 的值也会被送到 CRC 模块，用来生成 SPIRXCRC 的值；当需要传输 CRC 字节的时候，可以置位 CRCNXT，在正常数据传输完成后，下一次传输就会自动把 SPITXCRC 的值写入到 TXBUF (本次写入到 TXBUF 中的值不会送到 CRC 模块进行计算)，同时 CRCNXT 的值自动清零，在传输最后的 CRC 检验码的同时，也会接收对方的 CRC 检验码（本次接收到的数据不会写入 RXBUF)，在接收完成时会比较 SPIRXCRC 与接收到的校验码值，如果不匹配就会产生 CRCERR 标志位。

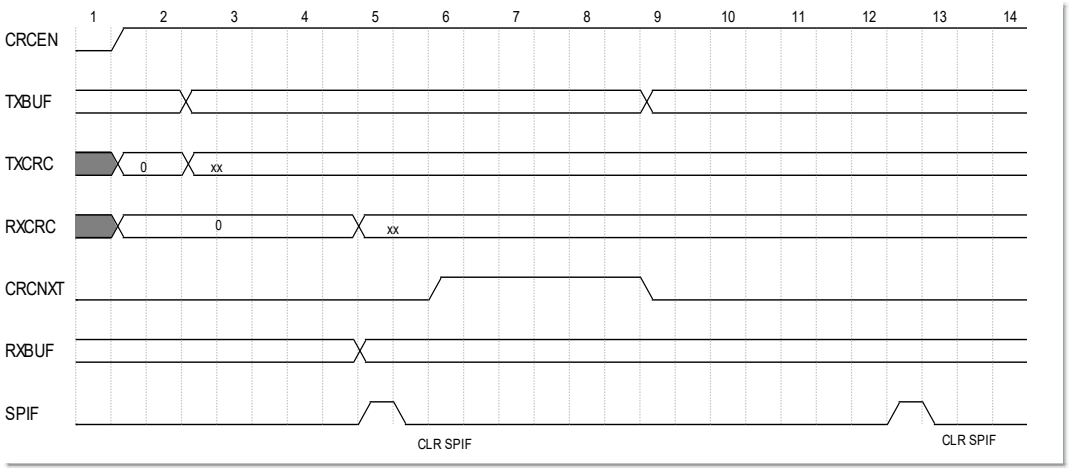


图 19-4 CRC 模块的工作时序图

19.3. SPI 配置

主机模式的 SCK 是由主机产生的，从机的 SCK 是从机输入的，下面是 SPI 配置流程:

1. 根据主从机的需求配置 SPICFG 寄存器中的 MSTEN 位，置 0 是用作从机，置 1 是用作主机
2. 作主机时需要通过 SPISCR 寄存器配置通信的速率，

通信的比特率 = Fmaster / (2 * (SPISCR+1))

该配置对从机无效，只在要求从机接收速率较快时，提高 Fmaster 的频率，以便从机模块可以采样到 SCK 的上升沿或下降沿

3. 配置 NSSM 来设定如何使用 NSS 引脚
   - 四线主机模式：NSS 引脚应配置为输出模式
   - 四线从机模式：NSS 引脚应配置成输入模式
   - 三线通信模式：可以设置禁用 NSS 引脚；
   - 另外 NSS 引脚在配置为输入模式时，可置位 SSM 位启用软件管理 NSS 引脚的输入值，屏蔽掉实际的 NSS 引脚的值
4. 配置 SPICFG 中的 CPOL 和 CPHA 来配置 SCK 的相位和极性
5. 配置 SPICR1 中的 LSBFIRST 来设置数据的传输格式
6. 配置 SPICRCP 寄存器和 CRCEN，使能 CRC 校验
7. 置位 SPICR1 中的 RXONLY 只允许接收或者置位 BDM 来启用半双工通信
8. 置位 SPICFG 中的 SPIEN 来启用 SPI 通信接口，这时相应的 GPIO 接口会用作 SPI 通信接口，同时 SPIEN 从低电平到高电平的变化会导致清零 RXOVER，CRCERR，MODF，SPIF，WCF 标志位，置位 TXBMT，RXBMT 标志位
9. 如果使用中断模式进行通信则需配置 SPIIER 寄存器来使能相应的中断

19.3.1. 数据处理流程

数据通信的流程大致分为阻塞模式通信和非阻塞式模式通信，大致的处理方式是一样的，只是非阻模式是在中断中进行的：

1. 阻塞通信时向 DAT 寄存器写入数据后需要查询 TXBMT，在查询到 TXBMT 为 1 时可以写入下一个数据；在 TXE 中断使能允许的时候，TXBMT 置 1 会直接进入中断
2. 阻塞模式接收数据时，需要一直查询 RXBMT，在查询到该位为 0 时，则可以读取 DATA 寄存器的值；在 RXNE 中断允许时，RXBMT 置 0 时则会进入中断
3. 阻塞模式接收的过程中需要查询 RXOVRN 和 CRCERR 位，在查询到相应的位置 1 后则需要写零清理相应的错误标志位；在 RXERR 中断使能时，发生相应的错误标志位会直接进入中断
4. 非阻塞模式中，进入中断后查询一次状态信息，然后根据相应的状态信息处理发送接收流程，处理完成以后退出中断子程序，继续处理其他的事情

无论是阻塞模式还是非阻塞模式，通信模块中的相关标识位变化如下图所示，在数据写入到发送数据寄存器后，TXBMT 从 1 变为 0，然后数据寄存器中的数据会传送到内部的移位寄存器，移位寄存器标志位从 1 变为 0，直到数据从移位寄存器中完全移出后变为 1；在发送的过程中，BUSY 状态位一直为 1；发送完成标志位在当前字节传输完成后会变为 1，同时 RXBMT 的值会从 1 变为 0。

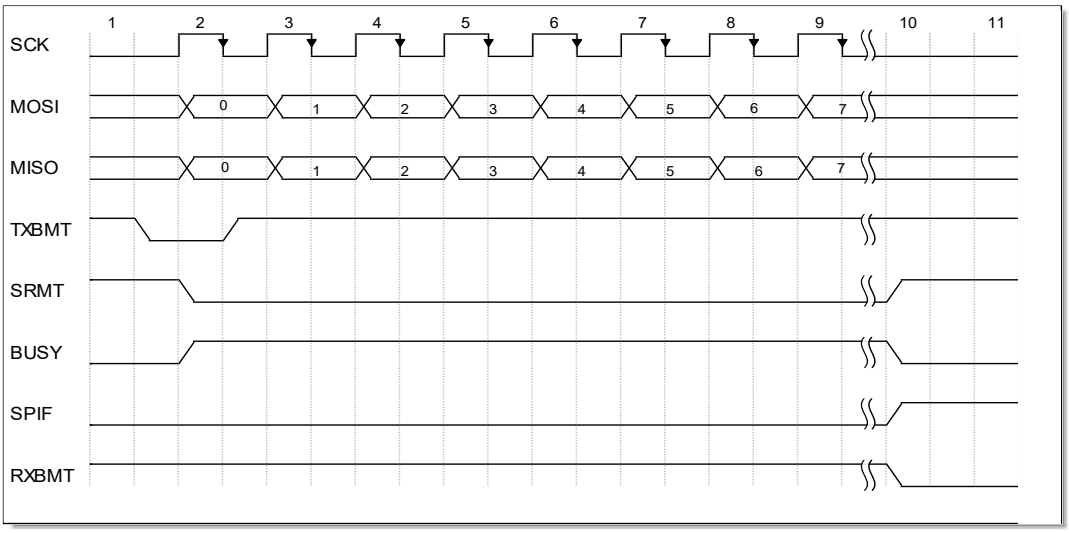


图 19-5 数据处理时序图

19.3.2. 睡眠模式唤醒

当系统进入睡眠模式，外设时钟存在时，SPI 模块有能力唤醒 MCU；如图所示，SPI 模块作为从机模块使用，开启了 WAKUP 中断使能，从机在接收到第一比特的数据时就会产生 WAKEUP 中断信号，在中断控制模块中如果使能了外设接口中断，就会直接唤醒 MCU；

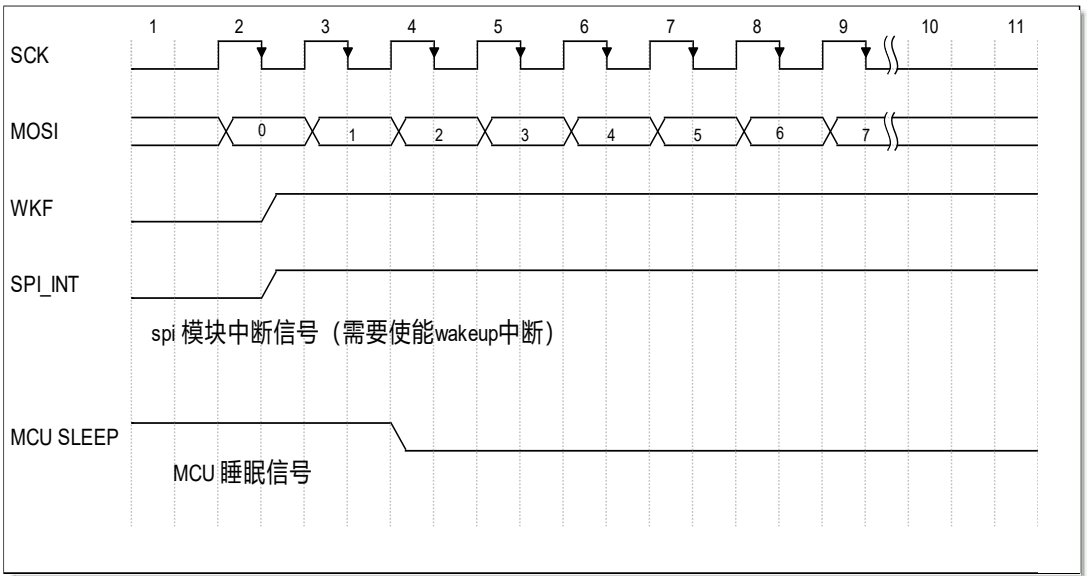


图 19-6 睡眠唤醒时序图

19.3.3. CRC 处理流程

阻塞模式传输数据时，在传输完成最后一个数据时，查询 TXBMT 的状态，在 TXBMT 位 1 时，则

置位 CRCNXT，这时 TXCRC 的值就会自动传输到 DATA 寄存器，然后 CRCNXT 就会自动清零， 查询 CRCNXT，查询到为 0 则清零 SPIF 状态位，然后再查询 SPIF 位，如果状态位置 1，则表示 CRC 校验码发送完成，接着查询 CRCERR 状态位，如果该状态位为 1，表示 CRC 校验码不匹配，写零清零相应的状态位；采用非阻塞模式通信时，TXBMT 为 1 进入中断时，若数据已经发送完整，则置位 CRCNXT，CRC 发送完成后如果产生了 CRCERR，则会直接进入中断，查询相关的标志位，然后写零清零。

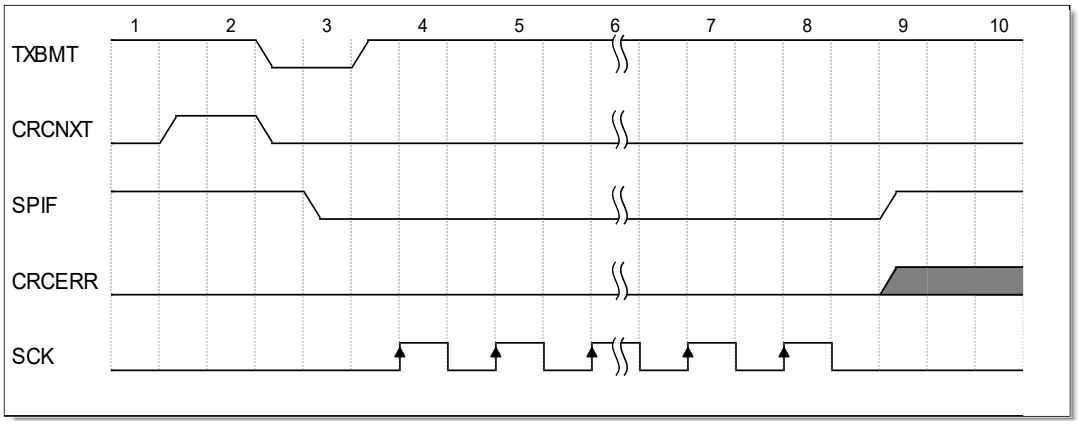


图 19-7 CRC 模块标志位时序图

20. I2C 接口

I2C 模块通过 SDA 和 SCL 管脚与外部 I2C 器件进行通信，数据以字节格式按照先传高位的原则进行传输。主要特性如下：

- 支持主机模式和从机模式
- 支持 7 比特地址和 10 比特地址模式
- 支持标准模式（100kHz)和快速模式（400kHz)
- 支持快速+模式（1MHz)
- 主机和从机支持时钟拉伸控制
- 发送 NACK (从机模式)
- 单个数据输入输出缓存

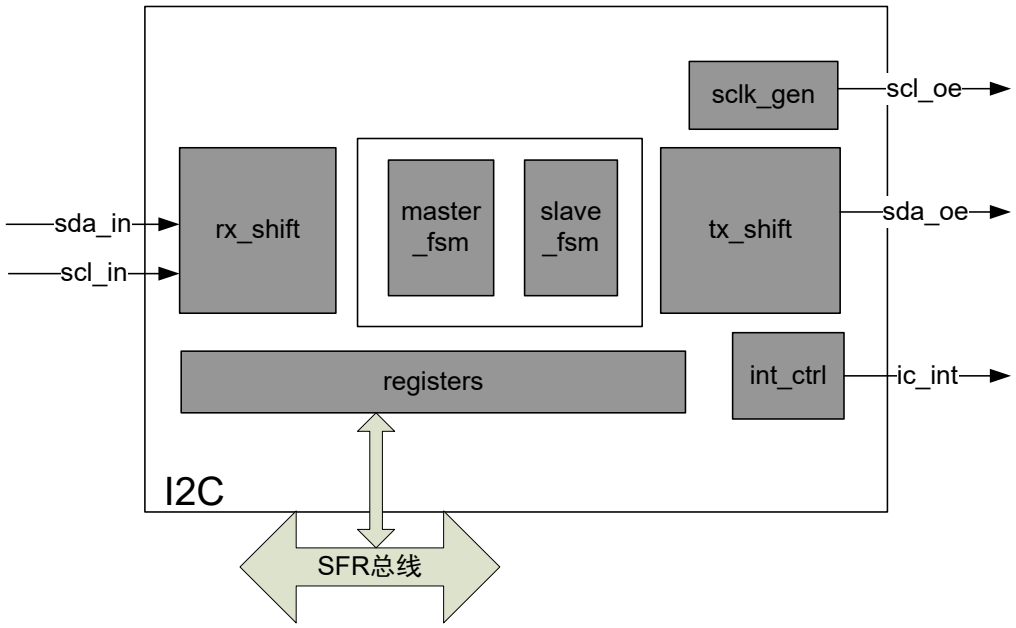


图 20-1 I2C 原理框图

I2C 接口支持主机模式和从机模式，并支持 7 比特和 10 比特地址的寻址；在使能了 I2C 接口的时候，请配置相应的 IO 为开漏输出。

20.1. I2C 接口相关寄存器汇总

名称	状态	寄存器	地址	复位值
SYSON	睡眠模式，系统时钟控制 1 = 保持运行 0 = 关闭	LVDCON[7]	0x21A	RW-0
STRET	主机模式时钟拉伸 (当发送数据为空或接收数据非空时) 1 = 使能 (拉低 SCL) 0 = 禁止 (继续产生时钟) 从机模式时钟拉伸 (当地址匹配、发送数据为空或接收数据非空时) 1 = 使能 (拉低 SCL) 0 = 禁止 (继续接收时钟) 注：如果主/从机禁止拉伸且为发送模式，当发送缓存 TX-FIFO 为空时，则重复发送最后一次写入到 I2CDAT 的数据；	I2CCR1[7]	0x196	RW-1
ADDRM [1]	地址格式 1 = 10 位 0 = 7 位	I2CCR1[5]		RW-0
STOP [2]	主机发送 Stop (字节传输后，或主机拉伸 SCL 时) 1 = 发送 (发送成功后自动清零) 0 = 不发送	I2CCR1[4]		RW-0
WRN	主机模式，数据传输方向 (读写位 R/W) 1 = 读取 0 = 发送	I2CCR1[3]		RW-0
START [3]	主机发送 Start / Restart 1 = 发送 0 = 不发送	I2CCR1[2]		RW-0
MASTER [1]	工作模式 1 = 主机模式 0 = 从机模式	I2CCR1[1]		RW-0
I2CEN	I2C 接口 1 = 使能 0 = 禁止	I2CCR1[0]		RW-0

[1] 当 I2CEN = 0 时可写；
[2] 此位在从机模式下不可置 1；
[3] 此位发送成功后，或检测到 BUSERR 后自动清零；

名称	状态	寄存器	地址	复位值
I2CADDRL	从机地址低有效位 (LSB) 7 位地址：ADD[6:0]有效，ADD[7]忽略； 10 位地址：ADD[7:0] = 低 8 位； 注：主机模式下为目标从机地址，从机模式下为本机地址；	I2CADDR[7:0]	0x198	RW-0000 0000
I2CADDRH	从机地址高有效位 (MSB) 7 位地址：ADD[9:8]忽略； 10 位地址：ADD[9:8] = 高 2 位； 注：主机模式下为目标从机地址，从机模式下为本机地址；	I2CCR2[7:6]	0x197	RW-00
NACK	接收应答 1 = 发送 NACK 0 = 发送 ACK (地址匹配或接收到数据)	I2CCR2[5]		RW-0
DUTY	SCL 占空比 (主机模式) 1：SCLL / SCLH = 2:1 0：SCLL / SCLH = 1:1	I2CCR2[4]		RW-0
I2CCCR	I2C 外设时钟频率 Fmaster Fmaster = Sysclk / (2x (I2CCCR + 3)) 注： 1. 最小值为 1 MHz； 2. 因存在延时，实际频率略有误差；	I2CCCR[7:0]	0x199	RW-1111 1111
I2CDAT	数据寄存器 写时：将新数据写入到 TX-FIFO 中 读时：返回 RX-FIFO 中未读的数据 注： TX-FIFO 和 RX-FIFO 的深度均为 1；	I2CDAT[7:0]	0x19A	RW-0000 0000

表 20-1 I2C 相关寄存器

名称	状态		寄存器	地址	复位值
GIE	全局中断	1 = 使能 (PEIE, I2CIE 适用) 0 = 全局关闭 (唤醒不受影响)	INTCON[7]	Bank 首地址 +0x0B	RW-0
PEIE	外设总中断	1 = 使能 (I2CIE 适用) 0 = 关闭 (无唤醒)	INTCON[6]		RW-0
I2CIE	I2C 接口中断	1 = 使能 0 = 关闭	I2CCR2 [0]	0x197	RW-0

名称	状态		寄存器	地址	复位值
HOLDF	主机模式，SCL 拉伸状态	1 = SCL 正被主机拉伸 0 = SCL 未被主机拉伸	I2CISR[7]	0x19E	RO-0
	从机模式，SCL 拉伸状态 [4]	1 = SCL 正被从机拉伸 0 = SCL 未被从机拉伸，或已被清零			RW0-0
STOPF [4]	主/从机模式，Stop 标志	1 = Yes 0 = No	I2CISR[6]		RW0-0
ADDRF [4]	主机模式，目标从机地址匹配标志	1 = 匹配 (ACK 后置位) 0 = 不匹配，或未发送地址	I2CISR[5]	0x19E	RW0-0
	从机模式，本机地址匹配标志	1 = 匹配 [5] 0 = 不匹配			
NACKF [4]	主/从机模式，接收应答状态	1 = NACK 0 = ACK	I2CISR[4]		RW0-0
I2CBUSY	主/从机状态	1 = Busy (繁忙) 0 = IDLE (空闲) 注：从机模式，地址匹配成功后即置位，接收到 Start / Restart / Stop 后清零；	I2CISR[3]		RO-0
BUSERR / DIRF	主机模式，总线仲裁失败 (BUSERR) 标志	1 = 仲裁失败 0 = 未发生仲裁失败	I2CISR[2]		RO-0
	从机模式，数据传输方向 (DIRF) 标志 [4]	1 = 发送 (从机接收地址字节的读写位为 1 时置位) 0 = 接收			RW0-0
IICRXNEF [4]	RX-FIFO 状态	1 = 非空 (读 I2CDAT 或写 0 清零) 0 = 空	I2CISR[1]		RW0-0
IICTXEF [6]	TX-FIFO 状态	1 = 空 (写 I2CDAT 清零) 0 = 非空 (写 1 清空)	I2CISR[0]		RW1-1

表 20-2 I2C 中断使能和状态位

[4] 写 0 清零，写 1 无效；
[5] 10 位地址格式，低 8 位地址和高 2 位地址匹配时均会置位；
[6] 只能写 1，写 0 无效；

名称	状态		寄存器	地址	复位值
SDAPO	I2C_SDA 管脚映射选择位 00 = PC0 10 = PC2 01 = PB3 11 = PB4		COMAF0[3:2]	0x285	RW-00
SCLPO	I2C_SCL 管脚映射选择位 00 = PB7 10 = PC1 01 = PB2 11 = PD4		COMAF0[1:0]		RW-00
ODCONA	PORTA 开漏	1 = 使能 0 = 关闭	ODCONA[7:0]	0x205	RW-0000 0000
ODCONB	PORTB 开漏		ODCONB[7:0]	0x206	RW-0000 0000
ODCONC	PORTC 开漏		ODCONC[7:0]	0x207	RW-0000 0000
ODCOND	PORTD 开漏		ODCOND[5:0]	0x208	RW-00 0000

表 20-3 I2C 接口引脚控制

名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	复位值
I2CCR1	0x196	STRET	—	ADDRM	STOP	WRN	START	MASTER	I2CEN	1-00 0000
I2CCR2	0x197	I2CADDRH[1:0]		NACK	DUTY	—	—	—	I2CIE	0000 0--0
I2CADDR	0x198	I2CADDRL[7:0]								0000 0000
I2CCCR	0x199	I2CCCR[7:0]								1111 1111
I2CDAT	0x19A	I2CDAT[7:0]								0000 0000
I2CISR	0x19E	HOLDF	STOPF	ADDRF	NACKF	I2CBUSY	DIRF /BUSERR	IICRXNEF	IICTXEF	0000 0001

表 20-4 I2C 相关寄存器

20.2. I2C 配置

20.2.1. 主机时序

主机模式的时序是通过配置 I2CCCR (寄存器最小值为 1)寄存器实现的，接口的时钟输出理论值等于 F_{Sysclk} / 2 (I2CCCR+3)，实际中有延时，因此有误差。

如下图所示主机所产生的时序：

1. Thold 指的是数据保持时间：Thold ≈ (I2CCCR / 4 + 1) * T_{Sysclk}
2. Tsyn 是信号同步时间：Tsyn = 2 * T_{Sysclk}
3. Tdly 是信号传输的延时时间：Tdly ≤ 1* T_{Sysclk}

因此主机产生的时钟高电平时间为

- Thigh= 1* $\mathbf{T_{Sysclk}}$ + 2* T_{Sysclk} + (I2CCCR+1)* T_{Sysclk} <= (I2CCCR+4)* T_{Sysclk}

在 DUTY 配置位为 0 时：Tlow = Thigh (高低电平占空比为 1:1)

在 DUTY 配置位为 1 时：Tlow = 1 * T_{Sysclk} + 2* T_{Sysclk} + (I2CCCR*2+1)* T_{Sysclk} ≤ (I2CCCR*2+4)* T_{Sysclk} ≈ 2*Thigh (高低电平约为 1:2)

START/RESTAT 的建立时间和保持时间，STOP 的建立时间均等于 Thold。

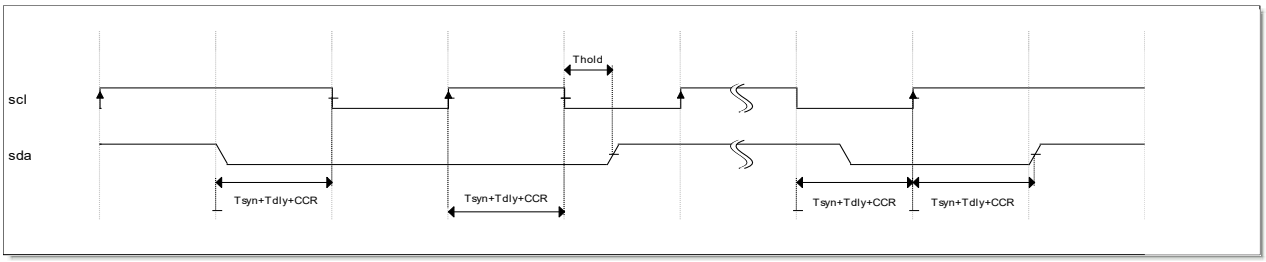


图 20-2 主机时序图

20.2.2. 主机时序

在 STRET 使能时，主机发送主要依赖的标志位是 HOLDF，当 HOLDF 为 1 是，表示需要写入下一个要发送的数据，写入完成后 HOLDF 由硬件自动清零；在 START 信号开始前，如果 I2CTXEF 标志位为零，则 HOLDF 在发送完成一个数据后置 1，否则在发送完地址后置位 HOLDF，如下图所示：

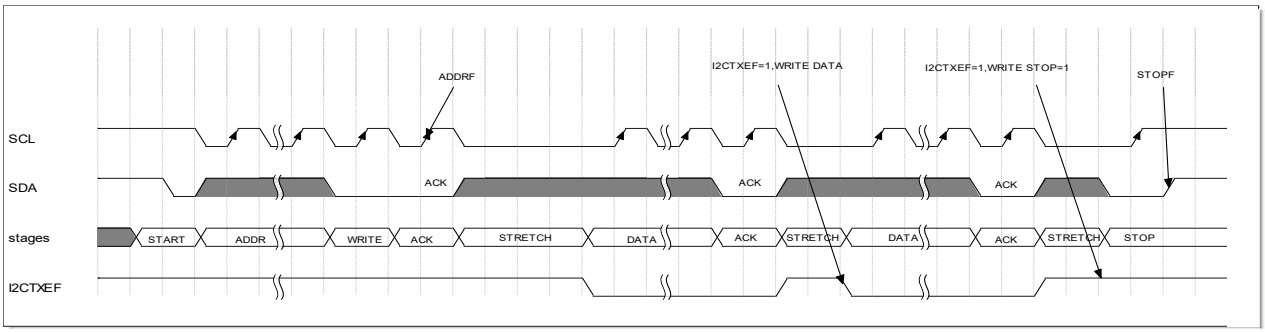


图 20-3 主机发送时序图 (STRET=1)

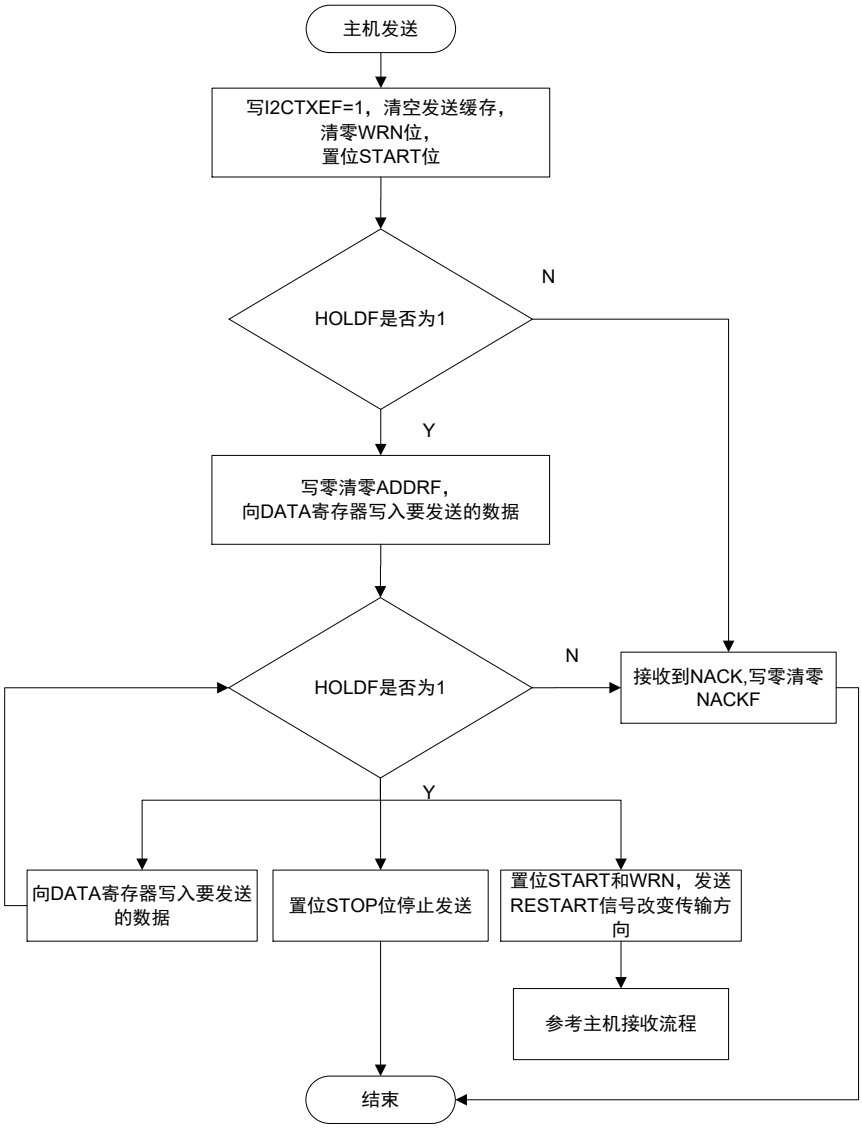


图 20-4 主机发送流程图 (STRET=1)

在 STRET 未使能时，主机发送主要依赖的标志位是 I2CTXEF 标志位，在发起起始信号以后，软件只要判断到 I2CTXEF 为空，则就可以向发送缓存区写入数据；如果想要停止发送数据，需置位 STOP，在当前数据传输完成后发送停止信号，停止信号发送完成后，STOP 位由硬件自动清零。需要注意的是，如果 CPU 的处理速度比较慢，I2C 接口速度比较快，则有可能在查询到 I2CTXEF 为空时不能立即写入数据，导致上一次写入的数据传输两次。

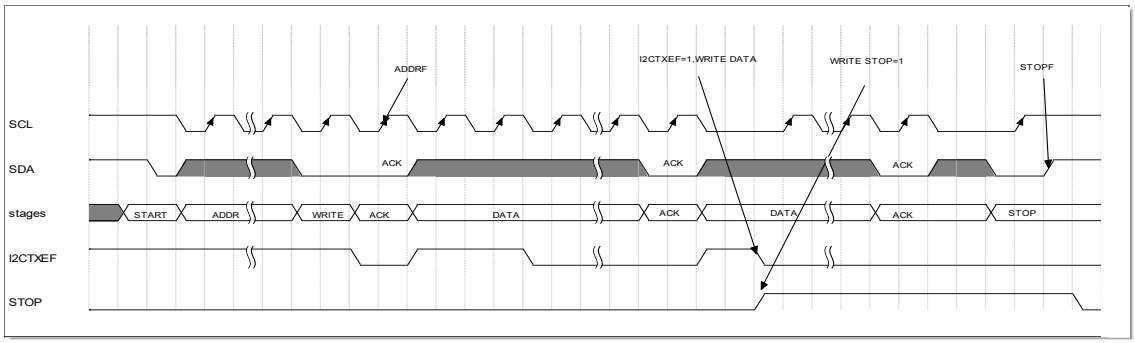


图 20-5 主机发送时序图 (STRET=0)

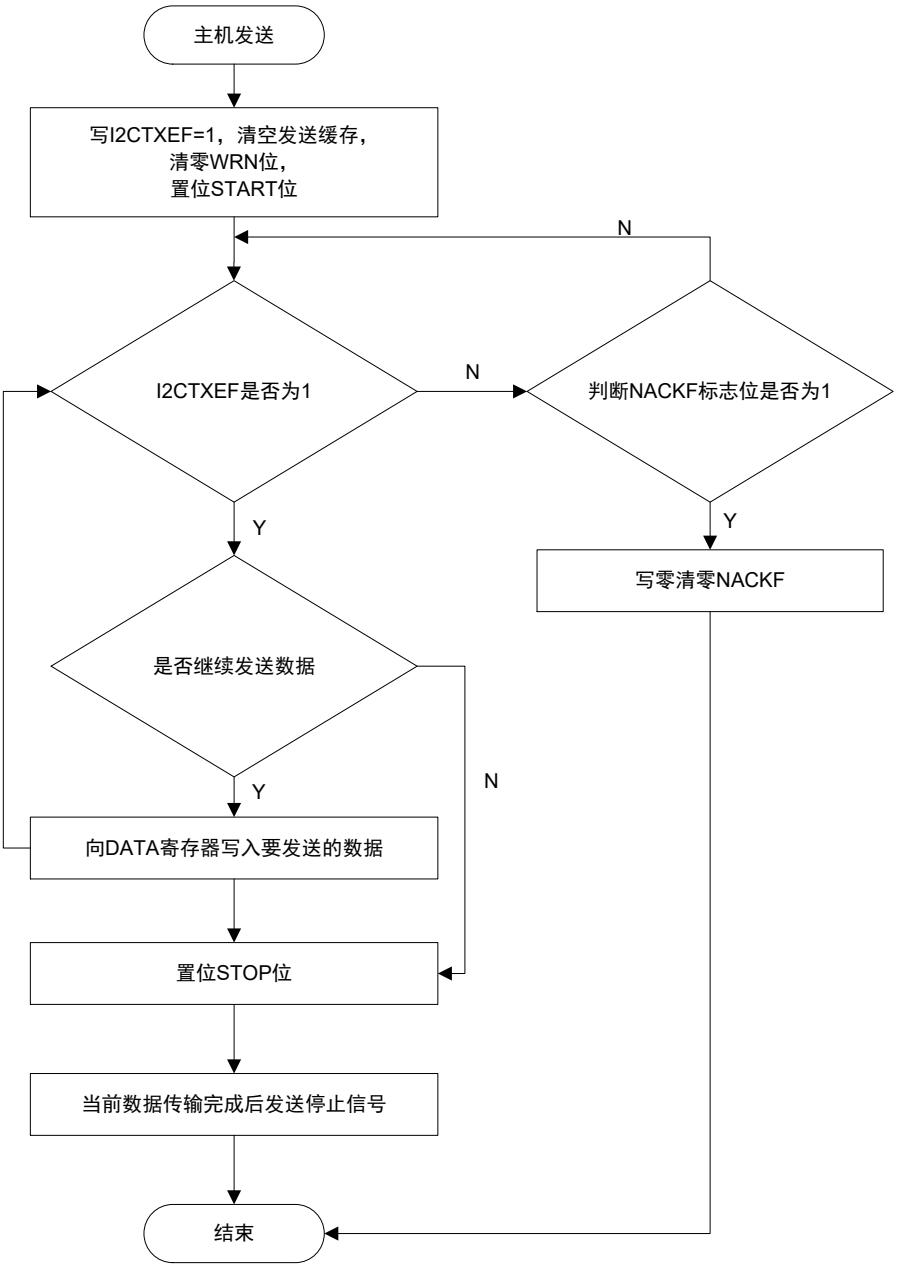


图 20-6 主机发送流程图 (STRET=0)

20.2.3. 从机发送

在 STRET 使能时，从机发送主要依赖的标志位是 HOLDF，当 HOLDF 为 1 是，表示接收到第九个时钟脉冲后拉低时钟，这时可以判断 ADDRF 标志位是否被寻址到，然后判断 DIRF 标志位，判断主机发起的是不是读取操作；如果地址匹配到并且主机是读取数据，则需要写入要发送的数据；从机模式下 HOLDF 的清零是由软件完成的，在清零该位之前需要根据实际进行相应的操作，例如写入数据；发送流程参考从机发送流程图。

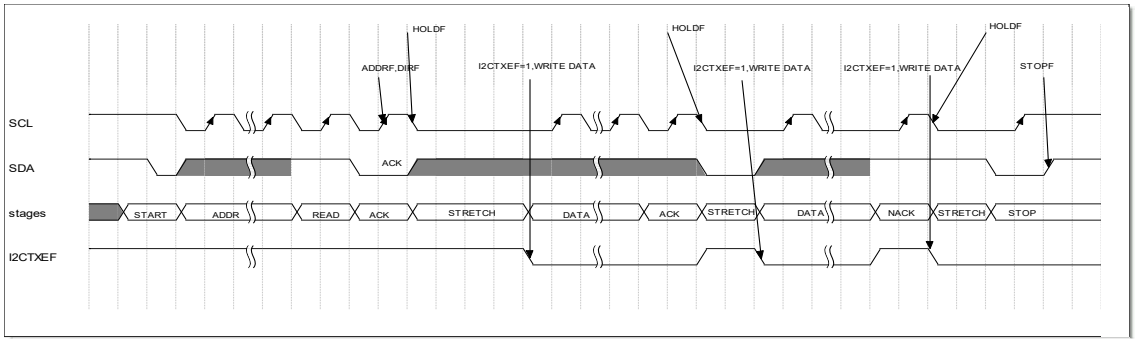


图 20-7 从机发送时序图 (STRET=1)

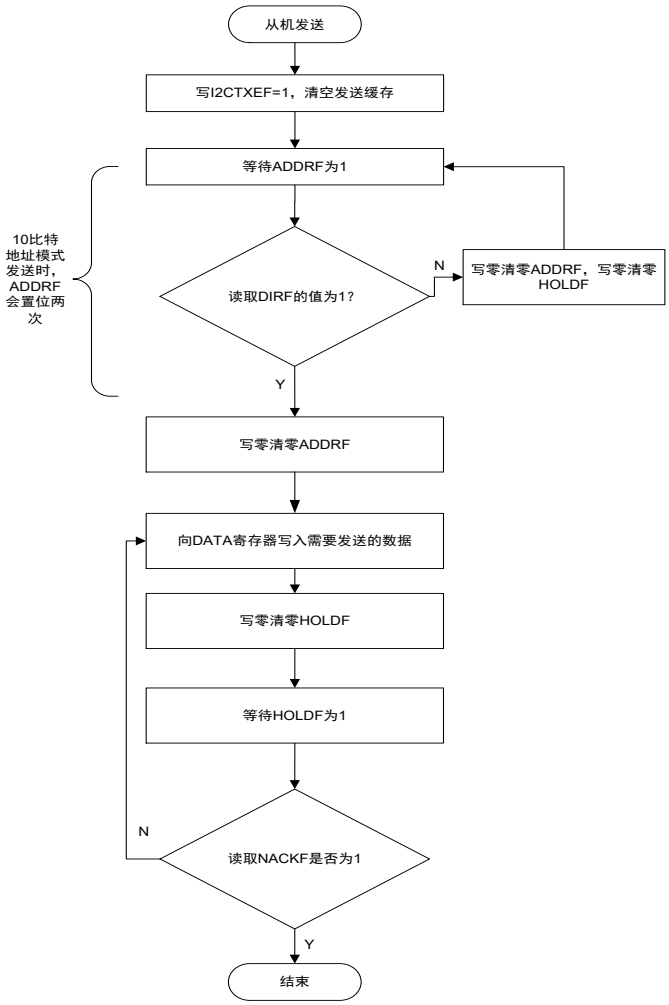


图 20-8 从机发送流程图 (STRET=1)

在 STRET 未使能时，从机发送主要依赖的标志位是 I2CTXEF，在 ADDRF 标志位置 1 以后，判断 DIRF 是否为零，然后只要 I2CTXEF 为 1，就可以向发送缓存器写入数据，直到接收到停止位后，不再写入数据；注意在收到停止位后，缓存区可能还会有一个数据，这时可以写 I2CTXEF 为'1'，来清空发送缓存区。需要注意的是在这种模式下，建议主机的速度相对不能太快，因为可能从机 (CPU) 无法在 I2CTXEF 值为 1 时，快速的写入数据，会导致上一次写入的数据重复发送。

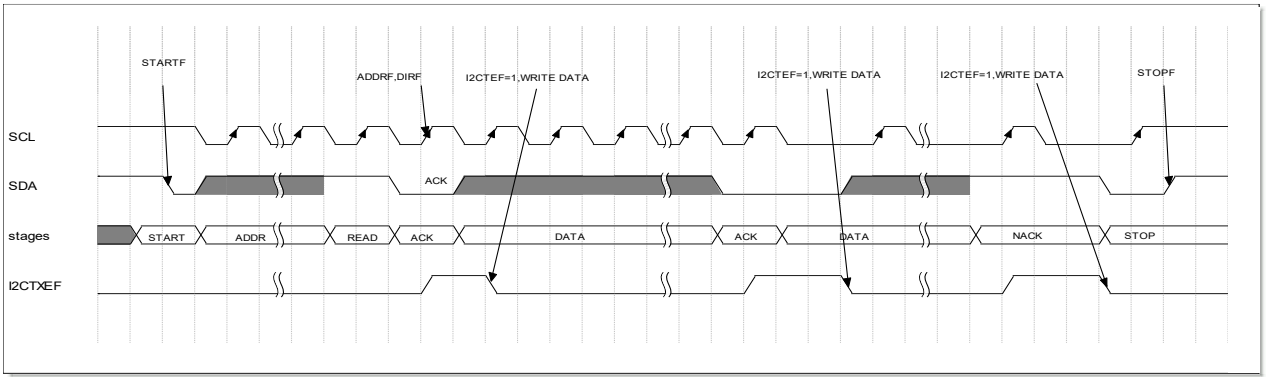


图 20-9 从机发送时序图 (STRET=0)

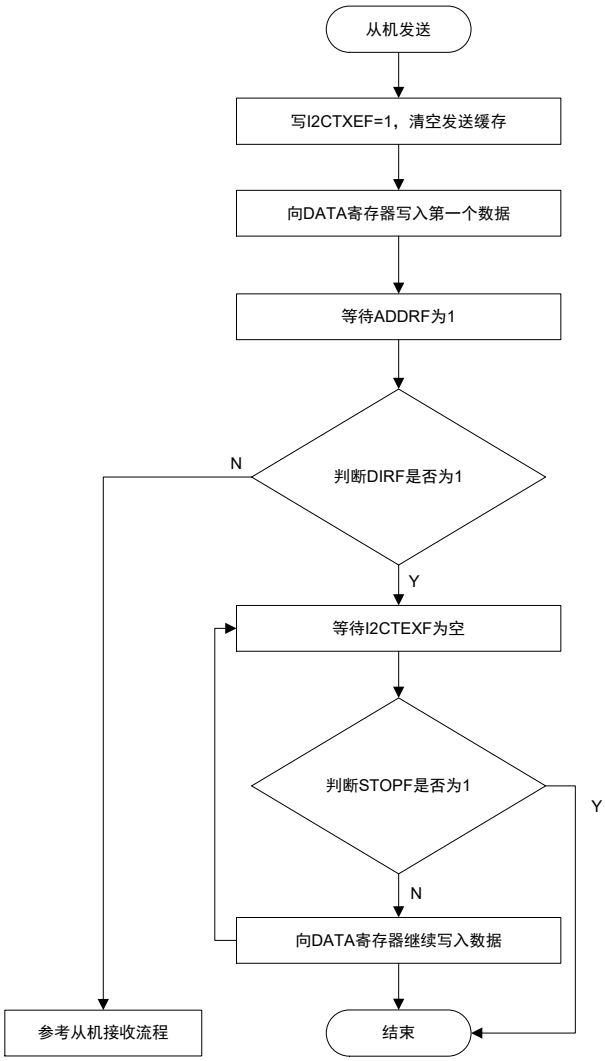


图 20-10 从机发送流程图 (STRET=0)

20.2.4. 主机接收

在 STRET 使能时，主机接收主要依赖的标志位是 HOLDF，当 HOLDF 为 1 是，表示需要读取已接收到的数据，读取完成后 HOLDF 由硬件自动清零；在 START 信号开始前，如果 I2CRXNEF 标志位为 1，则 HOLDF 在发送完成地址数据后置 1，否则在接收到一个数据后置 1，如下图所示：

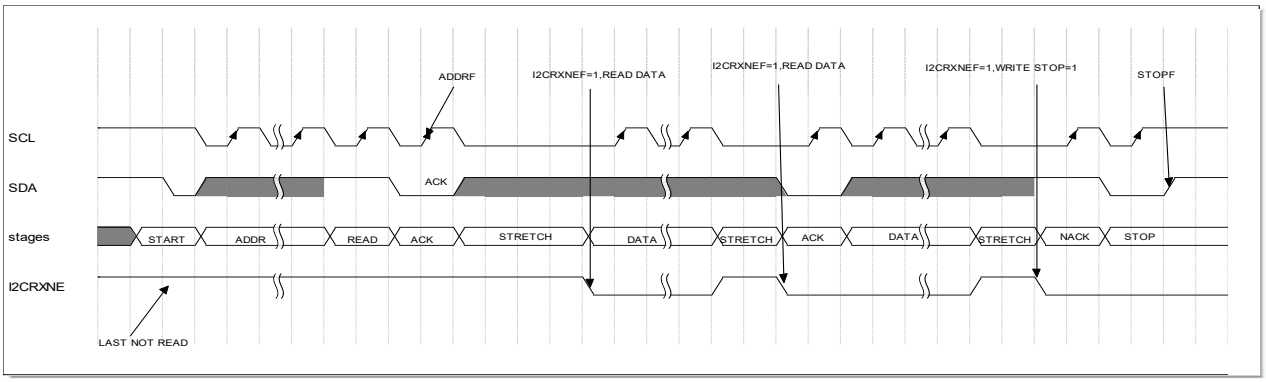


图 20-11 主机接收时序图 (STRET=1)

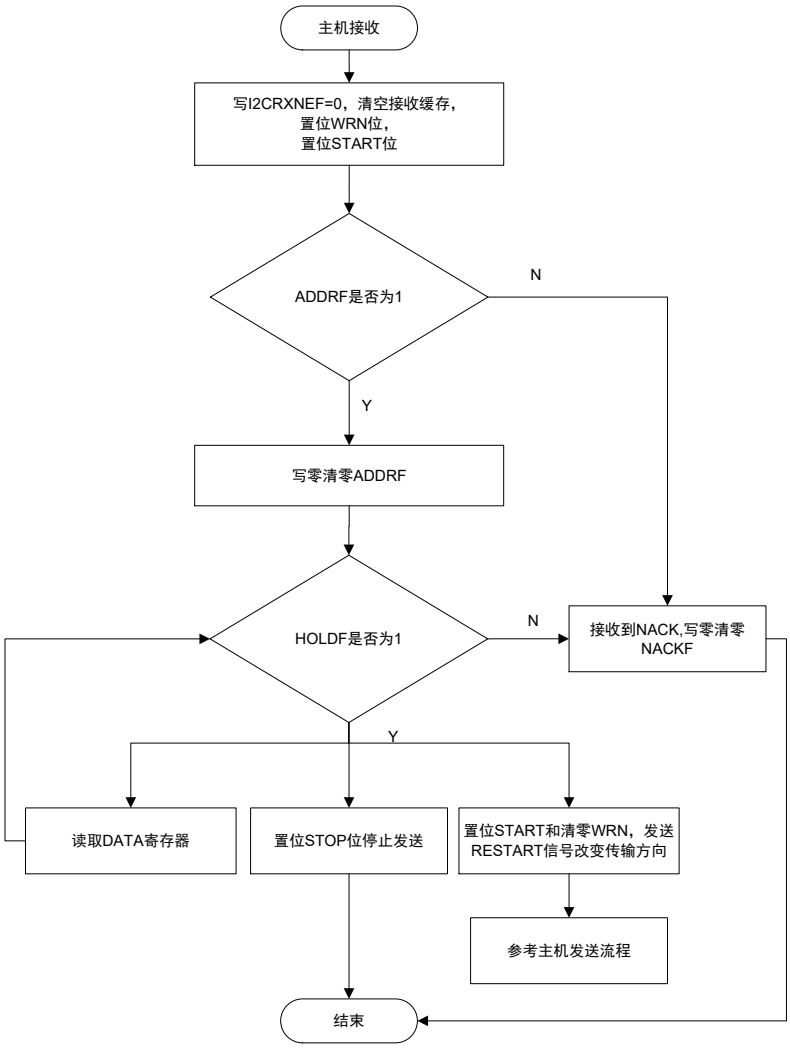


图 20-12 主机接收流程图 (STRET=1)

在 STRET 未使能时，主机接收主要依赖的标志位是 I2CRXNEF，主机发起起始信号，地址匹配以后，开始接收数据，每个数据接收完成后，置位 I2CRXNEF 标志位，读取接收的数据可以清零该标志位；如果不想继续接收数据，可以置位 STOP 位，这样在当前数据接收完成后，硬件自动回复 NACK，接着发送停止位，停止位发送完成以后，STOP 位由硬件自动清零。需要注意的是如果 I2C 接口速度过快，CPU 软件处理速度较慢，则在查询到 I2CRXNEF 为 1 时，不能立即读走数据，这种情况下可能会产生接收溢出。

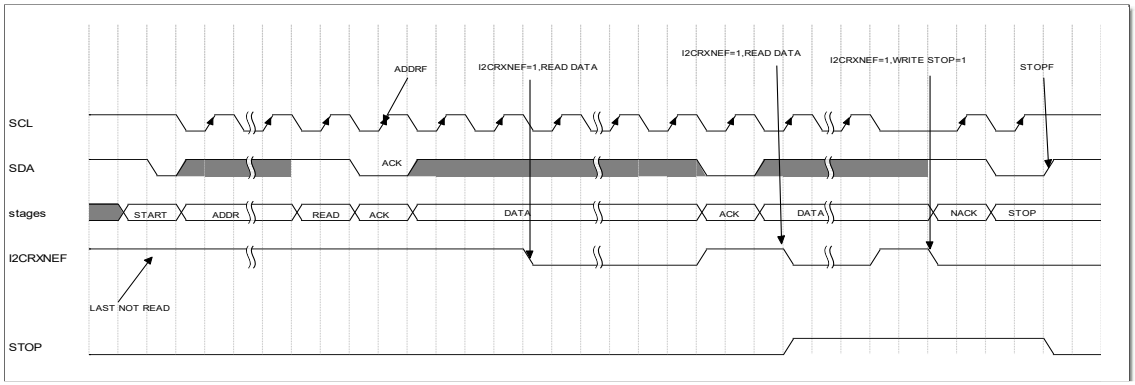


图 20-13 主机接收时序图 (STRET=0)

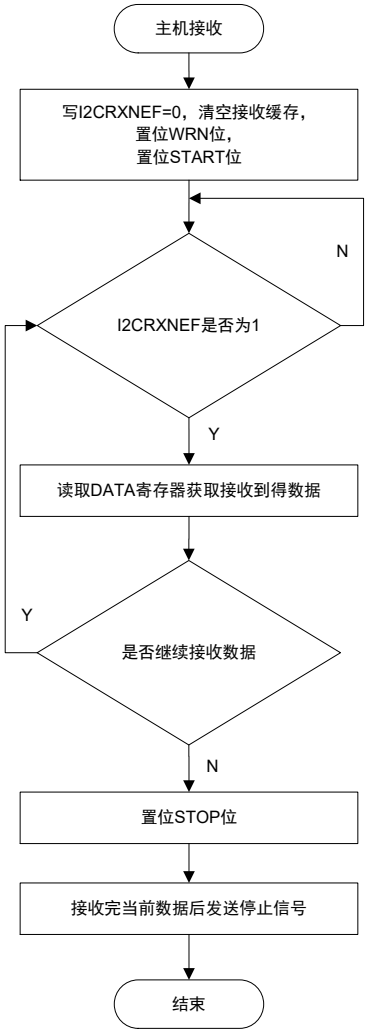


图 20-14 主机接收流程图 (STRET=0)

20.2.5. 从机接收

在 STRET 使能时，从机接收主要依赖的标志位是 HOLDF，当 HOLDF 为 1 是，表示接收到第九个时钟脉冲后拉低时钟，这时可以判断 ADDRF 标志位是否置 1，然后判断 DIRF 标志位，判断主机发起的是不是写操作；如果地址匹配到并且主机是写数据，则等待 RXNEF 标志位置 1，随后读取接收到的数据；从机模式下 HOLDF 的清零是由软件完成的，在清零该位之前需要根据实际进行相应的操作，例如读取数据、置位 NACK；接收流程参考从机接收流程图。

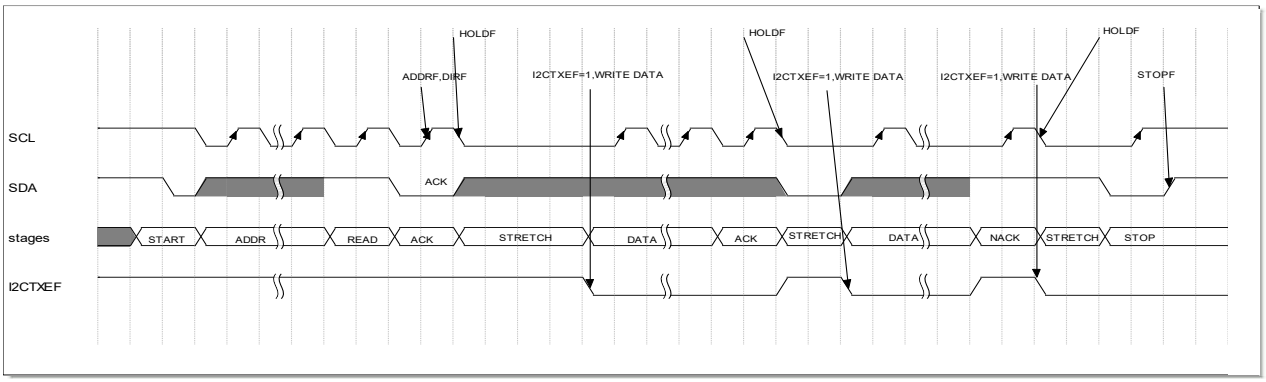


图 20-15 从机接收时序图(STRET=1)

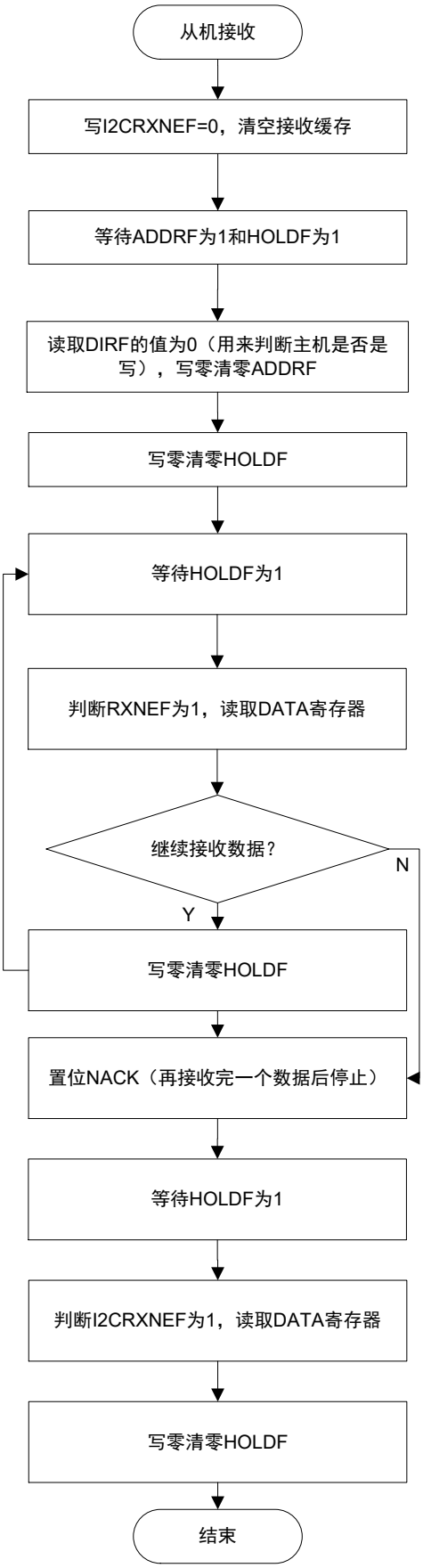


图 20-16 从机接收流程图 (STRET=1)

在 STRET 未使能时，从机的接收主要依赖于 I2CRXNEF 标志位，只要 I2CRXNEF 标志位置 1，就可以读取接收的数据，在接收到停止位以后，STOPF 标志位会置 1。需要注意的是如果 I2C 接口的速度过快，CPU 数据处理较慢时，数据的接收存在不能立即读取的情况，这时新接收的数据会覆盖掉上次接收的数据，产生接收溢出。

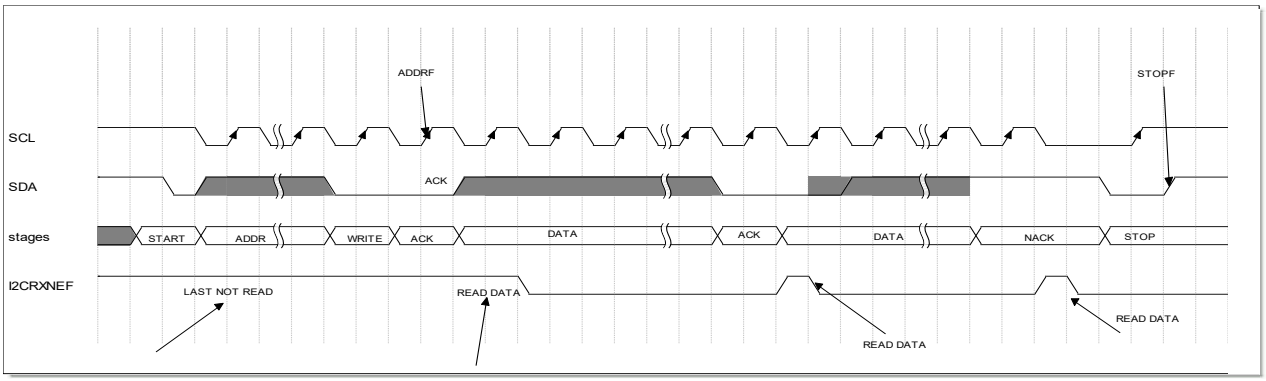


图 20-17 从机接收时序图(STRET=0)

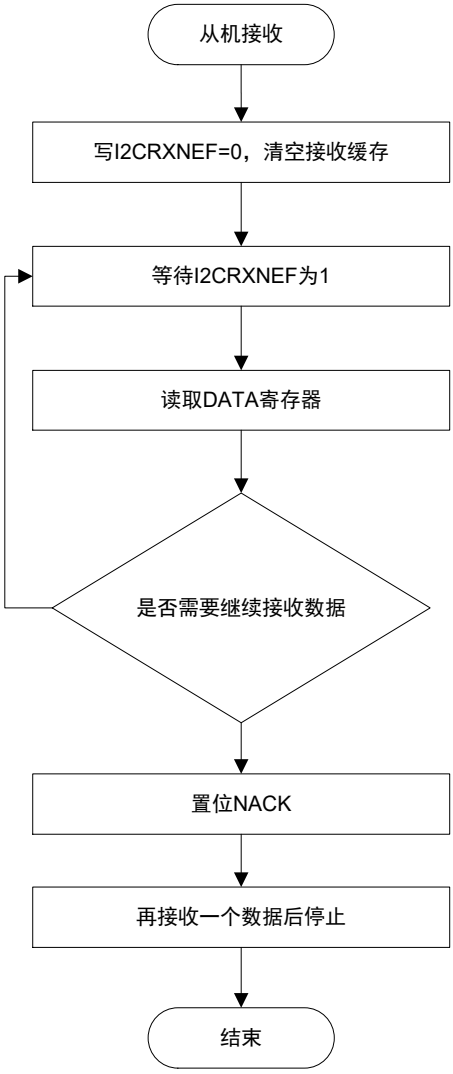


图 20-18 从机接收流程图 (STRET=0)

20.2.6. 中断映射

中断事件	中断标志	中断使能控制位
地址匹配标志、时钟拉伸标志、总线错误标志、停止位标志、接收非空标志和发送为空标志	BUSERR、HOLDF、ADDRF、STOPF、I2CRXNEF 和 I2CTXEF	I2CIE

表 20-5 I2C 中断请求

I2C 中断事件全部连接到同一个中断向量，如下图所示：

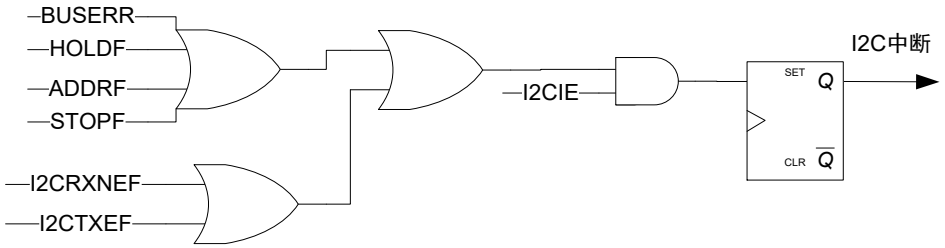


图 20-19 I2C 中断映射图

从机模式下支持中断唤醒功能，睡眠时无论有没有系统时钟，都可以唤醒 MCU。中断唤醒源标志是 ADDRF、I2CRXNEF 和 I2CTXEF。

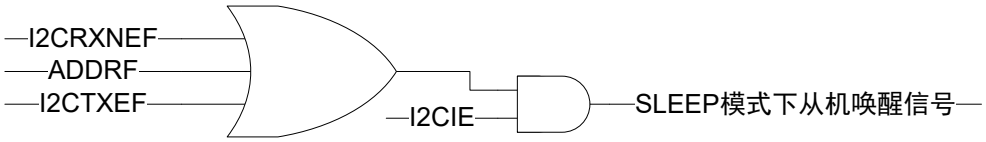


图 20-20 从机模式下 SLEEP 唤醒信号

21. 通用异步收发器 (USART0/1)

FT62F28x 集成了两个 USART，分别是 USART0 和 USART1，它们在功能上完全一样，故以下章节适用于任意一个 USART，URx 中的 x 可以是 0 或 1。

USART 模块主要特性：

- 全双工异步模式
- 8 比特或者 9 比特数据格式
- 停止位可配置 (1 个/2 个)
- 单输出数据缓冲器
- 双输入数据缓存器
- 支持奇偶校验功能
- 支持半双工模式
- 接收错误标志位 (接收溢出、帧错误和奇偶校验错误)

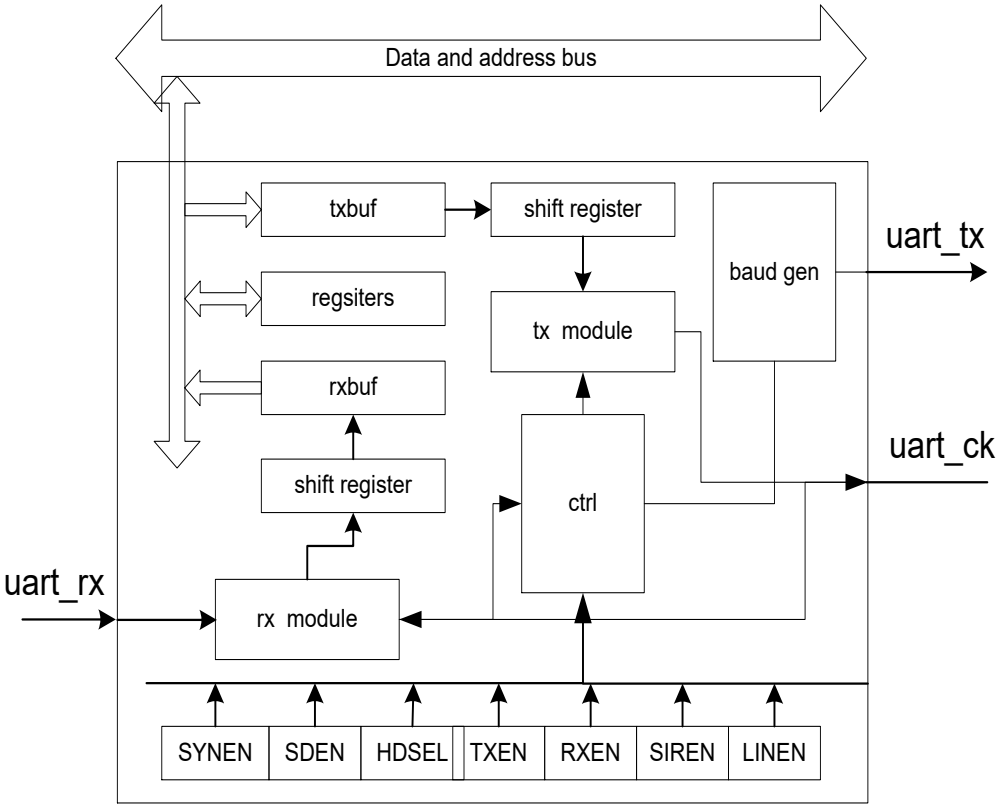


图 21-1 USART 结构框图

21.1. USART 接口相关寄存器汇总

名称	状态		寄存器	地址	复位值
SYSON	睡眠模式，系统时钟控制 1 = 保持运行 0 = 关闭		LVDCON[7]	0x21A	RW-0
UR0DATL	USART0/1 数据发送/接收 FIFO 低 8 位		UR0DATL[7:0]	0x185	RW-0000 0000
UR1DATL	(不适宜位操作)		UR1DATL[7:0]	0x90	RW-0000 0000
UR0DATH	当 URxMODE = 1：数据发送/接收 FIFO 高 1 位 注：要先写 URxDATAL，再写 URxDATAH 当 URxMODE = 0： 1 = URxDATAL 为地址 0 = URxDATAL 为数据		UR0DATH[0]	0x186	RW-0
UR1DATH			UR1DATH[0]	0x91	RW-0
UR0HDSEL	USART0/1 半双工	1 = 使能 0 = 禁止	UR0CR1[7]	0x187	RW-0
UR1HDSEL			UR1CR1[7]	0x92	RW-0
UR0STOP	USART0/1 停止位长度	1 = 2 bit 0 = 1 bit	UR0CR1[6]	0x187	RW-0
UR1STOP			UR1CR1[6]	0x92	RW-0
UR0MODE	USART0/1 通信数据长度 (不包括校验位)	1 = 9 bit 0 = 8 bit	UR0CR1[5]	0x187	RW-0
UR1MODE			UR1CR1[5]	0x92	RW-0
UR0EVEN	USART0/1 奇/偶校验 (仅在 URxPCEN=1 时有效)	1 = 偶校验 0 = 奇校验	UR0CR1[4]	0x187	RW-0
UR1EVEN			UR1CR1[4]	0x92	RW-0
UR0PCEN	USART0/1 校验位	1 = 使能 0 = 禁止	UR0CR1[3]	0x187	RW-0
UR1PCEN			UR1CR1[3]	0x92	RW-0
UR0RXEN	USART0/1 接收模式	1 = 使能 0 = 禁止	UR0CR1[2]	0x187	RW-0
UR1RXEN			UR1CR1[2]	0x92	RW-0
UR0TXEN	USART0/1 发送模式	1 = 使能 0 = 禁止	UR0CR1[1]	0x187	RW-0
UR1TXEN			UR1CR1[1]	0x92	RW-0
UR0EN	USART0/1 模块	1 = 使能 0 = 禁止	UR0CR1[0]	0x187	RW-0
UR1EN			UR1CR1[0]	0x92	RW-0
UR0BRRH	USART0/1 波特率分频计数器 波特率= Sysclk /(16*URxBRR) 注：默认值 URxBRR 为 0x0000 时，串口不工作；URxBRR 最小值为 0x0001	高四位	UR0CR2[7:4]	0x188	RW-0000
UR1BRRH			UR1CR2[7:4]	0x93	RW-0000
UR0BRRL		低八位	UR0BRRL[7:0]	0x189	RW-0000 0000
UR1BRRL			UR1BRRL[7:0]	0x94	RW-0000 0000

表 21-1 USART 相关用户配置寄存器

名称	状态		寄存器	地址	复位值
GIE	全局中断	1 = 使能 (PEIE, URxERRIE, URxRXNEIE, URxTXEIE 适用) 0 = 全局关闭 (唤醒不受影响)	INTCON[7]	Bank 首地址 +0x0B	RW-0
PEIE	外设总中断	1 = 使能 (URxERRIE, URxRXNEIE, URxTXEIE 适用) 0 = 关闭 (无唤醒)	INTCON[6]		RW-0
UR0ERRIE	USART0/1 接收错误中断	1 = 使能 0 = 关闭	UR0CR2[2]	0x188	RW-0
UR1ERRIE			UR1CR2[2]	0x93	RW-0
UR0OVF [7]	USART0 接收 FIFO 溢出标志	1 = 发生 0 = 未发生，或已清零	UR0STAT[5]	0x18F	RW0-0
UR0PEF [7]	USART0 接收奇偶校验错标志		UR0STAT[4]		RW0-0
UR0FEF [7]	USART0 接收帧错误标志		UR0STAT[2]		RW0-0
UR1OVF [7]	USART1 接收 FIFO 溢出标志		UR1STAT[5]	0x95	RW0-0
UR1PEF [7]	USART1 接收奇偶校验错标志		UR1STAT[4]		RW0-0
UR1FEF [7]	USART1 接收帧错误标志		UR1STAT[2]		RW0-0
UR0RXNEIE	USART0/1 接收 FIFO 非空中断	1 = 使能 0 = 关闭	UR0CR2[1]	0x188	RW-1
UR1RXNEIE			UR1CR2[1]	0x93	RW-1
UR0RXNEF	USART0/1 接收 FIFO 状态	1 = 非空 0 = 空，或已被清零	UR0STAT[1]	0x18F	RO-0
UR1RXNEF			UR1STAT[1]	0x95	RO-0
UR0TXEIE	USART0/1 发送 FIFO 为空中断	1 = 使能 0 = 关闭	UR0CR2[0]	0x188	RW-1
UR1TXEIE			UR1CR2[0]	0x93	RW-1
UR0TXEF	USART0/1 发送 FIFO 状态	1 = 空 0 = 非空	UR0STAT[0]	0x18F	RO-1
UR1TXEF			UR1STAT[0]	0x95	RO-1
UR0TXBSY	USART0/1 发送器状态 (半双工模式)	1 = 忙碌 0 = 空闲	UR0STAT[6]	0x18F	RO-0
UR1TXBSY			UR1STAT[6]	0x95	RO-0
UR0RXFULL	USART0/1 接收 FIFO 状态 (FIFO 深度为 2)	1 = 已满 0 = 未满	UR0STAT[3]	0x18F	RO-0
UR1RXFULL			UR1STAT[3]	0x95	RO-0

表 21-2 USARTx 中断使能和状态位

[7] 写 0 清 0，写 1 无效

名称	状态	寄存器	地址	复位值
TX0PO	USART0_TX 管脚映射 当 UR0SW=0 时： 0xx = PC2 100 = PC0 101 = PB3 110 = PB0 111 = PC7 当 UR0SW=1 时： 0xx = PC1 100 = PB7 101 = PB2 110 = PB5 111 = PA0	COMAF1[5:3]	0x286	RW-000
RX0PO	USART0_RX 管脚映射 当 UR0SW=0 时： 0xx = PC1 100 = PB7 101 = PB2 110 = PB5 111 = PA0 当 UR0SW=1 时： 0xx = PC2 100 = PC0 101 = PB3 110 = PB0 111 = PC7	COMAF1[2:0]		RW-000
UR1SW	USART1 TX 和 RX 管脚交换使能位	COMAF2[7]	0x287	RW-0
UR0SW	USART0 TX 和 RX 管脚交换使能位	COMAF2[6]		RW-0
TX1PO	USART1_TX 管脚映射 当 UR1SW=0 时： 0x = PC7 10 = PC5 11 = PC2 当 UR1SW=1 时： 0x = PA0 10 = PC6 11 = PC1	COMAF2[4:3]		RW-00
RX1PO	USART1_RX 管脚映射 当 UR1SW=0 时： 0x = PA0 10 = PC6 11 = PC1 当 UR1SW=1 时： 0x = PC7 10 = PC5 11 = PC2	COMAF2[1:0]		RW-00
ODCONA	PORTA 开漏	ODCONA[7:0]	0x205	RW-0000 0000
ODCONB	PORTB 开漏	ODCONB[7:0]	0x206	RW-0000 0000
ODCONC	PORTC 开漏	ODCONC[7:0]	0x207	RW-0000 0000
ODCOND	PORTD 开漏	ODCOND[5:0]	0x208	RW-00 0000

1 = 使能
0 = 关闭

表 21-3 USART 接口引脚控制

名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	复位值
UR0DATL	0x185	UR0DATL[7:0]								0000 0000
UR0DATH	0x186	—							UR0DATH	---- ---0
UR0CR1	0x187	UR0HDSEL	UR0STOP	UR0MODE	UR0EVEN	UR0PCEN	UR0RXEN	UR0TXEN	UR0EN	0000 0000
UR0CR2	0x188	UR0BRRH[3:0]				—	UR0ERRIE	UR0RXNEIE	UR0TXEIE	0000 -000
UR0BRRL	0x189	UR0BRRL[7:0]								0000 0000
UR0STAT	0x19F	—	UR0TXBSY	UR0OVF	UR0PEF	UR0RXFULL	UR0FEF	UR0RXNEF	UR0TXEF	-000 0001
UR1DATL	0x90	UR1DATL[7:0]								0000 0000
UR1DATH	0x91	—							UR1DATH	---- ---0
UR1CR1	0x92	UR1HDSEL	UR1STOP	UR1MODE	UR1EVEN	UR1PCEN	UR1RXEN	UR1TXEN	UR1EN	0000 0000
UR1CR2	0x93	UR1BRRH[3:0]				—	UR1ERRIE	UR1RXNEIE	UR1TXEIE	0000 -000
UR1BRRL	0x94	UR1BRRL[7:0]								0000 0000
UR1STAT	0x95	—	UR1TXBSY	UR1OVF	UR1PEF	UR1RXFULL	UR1FEF	UR1RXNEF	UR1TXEF	-000 0001

表 21-4 USART 相关寄存器汇总 (USART0 和 USART1 分别由前缀为 UR0 和 UR1 的寄存器控制)

21.2. 功能描述

21.2.1. 一般描述

串口模块总共有两只引脚：

1. sin：用作串口数据的输入引脚
2. sout：用作串口数据的输出引脚

接口支持全双工模式和半双工模式。

21.2.2. 异步工作模式

USART 发送和接收模式可以同时工作，互不干扰，但是他们采用相同的配置模式；硬件自带奇偶校验功能，可以节省软件的开销；内部的接收缓冲器有两个，硬件实现的方式是一个深度为 2 的 FIFO，可以防止因为软件未来得及读取接收的数据，而发生数据丢失。

USART 接口的配置流程：

1. 配置 URxMODE 寄存器的值选择 8 位或者 9 位帧格式
2. 配置 URxPCEN 的值，来选择是否启用奇偶校验
3. URxPCEN 的值为 1 时，配置 URxEVEN 的值，来选择采用偶校验或者奇校验
4. 配置 URxSTOP 位来选择停止位的长度
5. 配置 URxBRRL 和 URxBRRH 的值，来选择发送和接收的波特率，波特率=Fcpu/(16*BRR)
6. 配置 URxTXEN 和 URxRXEN 来使能发送模式和接收模式
7. 置位 URxEN，使能 USART 接口

注：Fcpu 指的是系统时钟的频率；数据传输过程中，相关配置位不能修改，否则会发生数据传输错误。

如下图所示的 URxMODE=0 时的传输时序图，接收时，如果发生了帧错误、奇偶校验错误和溢出错误，则在数据接收完成后，与 URxRXNEF 标志位一起置位。发送时，URxTXEF 为 1 时表示发送数据寄存器为空，在为 0 时，如果发送器空闲，则数据会转移到内部的移位寄存器，同时 URxTXEF 会置 1，这时可以写入第二个要发送的数据。

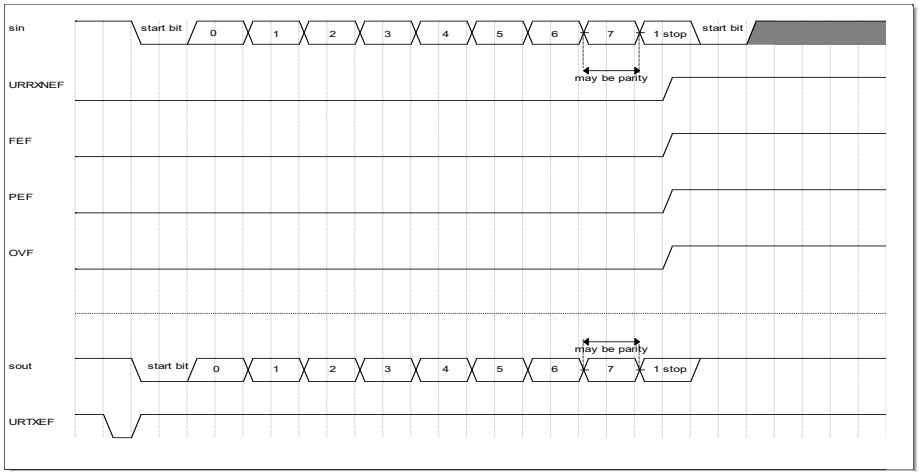


图 21-2 异步模式标志位时序图

奇偶校验位的配置：

设置 URxCR1 寄存器上的 URxPCEN 位，可以使能校验控制(发送时生成一个校验位，接收时进行校验检查)。根据 URxMODE 位定义的帧长度，可能的 USART 帧格式列在下表中。

URxMODE 位	URxPCEN	USART 帧格式
0	0	起始位+8 位数据+停止位
0	1	起始位+7 位数据+奇偶位+停止位
1	0	起始位+9 位数据+停止位
1	1	起始位+8 位数据+奇偶位+停止位

表 21-5 USART 帧格式

21.2.3. 半双工模式

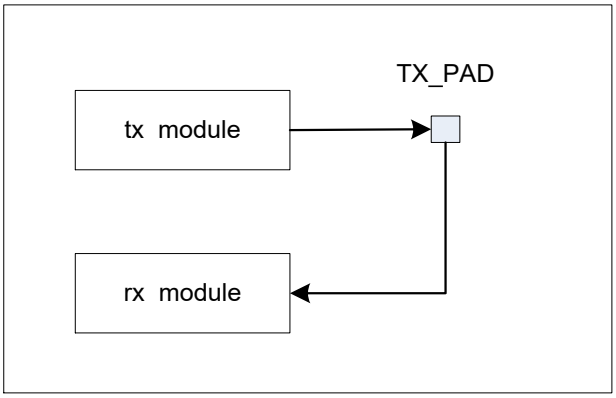


图 21-3 半双工模式内部连接示意图

要启用半双工模式，需要置位 URxHDSEL 位，启用半双工模式后，发送和接收来自同一个引脚，在发送的同时，如果使能了 URxRXEN 则发送的数据同样也会被接收到；在这种模式下，一般发送和接收是交替进行的，需要软件控制 URxTXEN 和 URxRXEN 来实现。

21.2.4. 中断映射

中断事件	中断标志	中断使能控制位
数据寄存器为空	URxTXEF	URxTXEIE
接收数据寄存器非空	URxRXNEF	URxRXNEIE
奇偶校验错误、溢出标志位和帧错误标志位	URxPEF、URxOVF 和 URxFEF	URxERRIE

表 21-6 USART 中断请求

USART 中断事件全部连接到同一个中断向量，如下图所示：

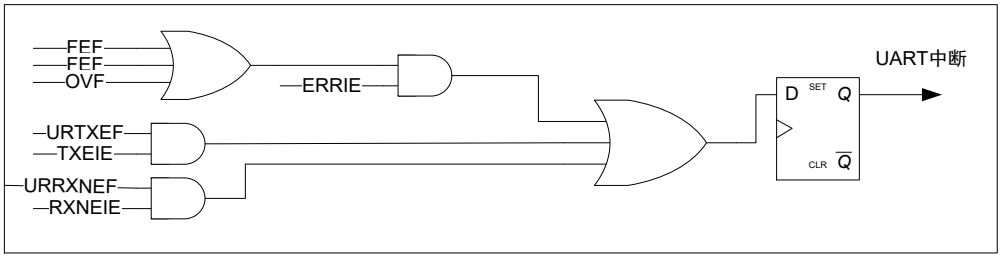


图 21-4 USART 中断映射图

22. 电容按键模块

22.1. 触摸按键功能

该系列单片机均提供多触控按键功能，触控按键功能完全内部集成只需外接少量元件，可通过对内部寄存器的简单操作来实现此功能。

22.2. 触摸按键结构

触控按键与 I/O 引脚共用，通过相应选择寄存器的位来选择此功能。按键编号为 KEY1 ~ KEY28。

23. 存储区读/写保护

程序区 (PROM) 可配置为全区读保护，或扇区读/写保护 (每个扇区空间为 1 k x 14)。数据 EEPROM 区(DROM)可配置为全区读保护。这些保护功能均由 IDE 界面进行选择配置。

名称	功能	默认
CPDB	DROM 全区读保护	关闭
CPB	PROM 全区读保护	关闭
SECPB0	PROM 扇区 0(1k x 14)读/写保护	关闭
SECPB1	PROM 扇区 1(1k x 14)读/写保护	关闭
SECPB2	PROM 扇区 2(1k x 14)读/写保护	关闭
SECPB3	PROM 扇区 3(1k x 14)读/写保护	关闭
SECPB4	PROM 扇区 4(1k x 14)读/写保护	关闭
SECPB5	PROM 扇区 5(1k x 14)读/写保护	关闭
SECPB6	PROM 扇区 6(1k x 14)读/写保护	关闭
SECPB7	PROM 扇区 7(1k x 14)读/写保护	关闭

图 23-1 存储区读/写保护初始化配置寄存器

24. 程序存储器

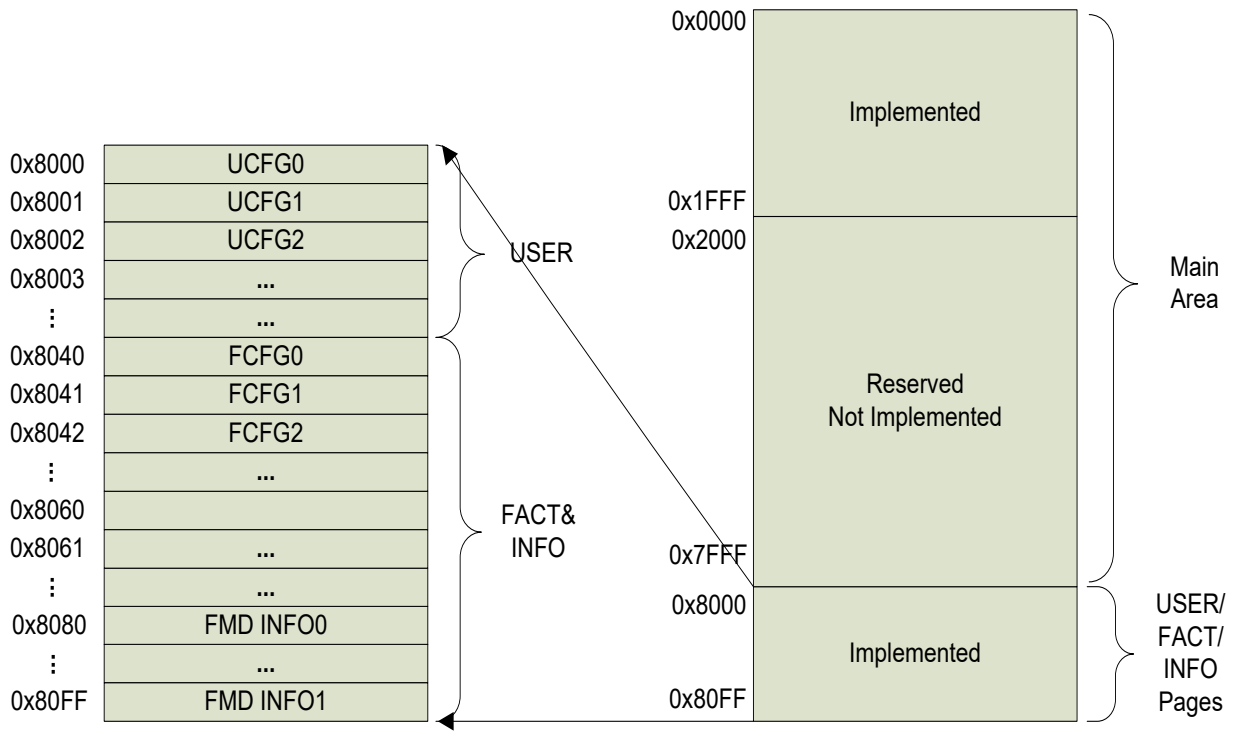


图 24-1 程序存储区地址空间

程序地址计数器 PC 为 15 位（0x0000 ~ 0x7FFF)，最多支持 32k 地址空间。芯片实现了 8K 的程序存储器（0x0000 ~ 0x1FFF)，外加上 2 个额外的用户配置区，工厂配置区，以及 2 个信息区 INFO0/INFO1。

8k 程序存储器由 128 页组成，每页 64 个 word (1word= 14bits)，地址范围为 0x0000~0x1FFF，当程序地址超过 0x1FFF 将导致回卷到 0x0000。

另外的 4 个 NVM 区分别占一个单独的页，每页 64 word，其编址从 0x8000 开始，到 0x80FF 结束。

将程序存储器当作数据存储器读取

有两种方法可访问程序存储器中的常数。第一种方法是使用 RETW 指令表。第二种方法是设置 FSR 指向程序存储器。

RETW 指令

RETW 指令用于提供对常数表的访问。

例 2.1 给出了创建这种表的推荐方法。

BRW 指令使得这种类型的表实现起来非常简单。如果代码必须保持与前几代单片机之间的可移植性，则 BRW 指令不适用，因此必须使用较老的表读取方法。

```
Constants ; example 2.1
BRW                 ;Add Index in W to program counter to ;select data
RETW    DATA0       ;Index0 data
RETW    DATA1       ;Index1 data
RETW    DATA2
RETW    DATA3
my_function
;… LOTS OF CODE…
LDWI    DATA_INDEX
call  constants     ; THE CONSTANT IS IN W
```

适用 FSR 间接读取

可将程序存储器当作数据存储器进行访问，方法是将 FSRxH 寄存器的 bit7 置 1 并读取与之配对的 INDFx 寄存器。MOVIW 指令会将已寻址到的字的低 8 位保存在 W 寄存器中。无法通过 INDF 寄存器执行写程序存储器操作。通过 FSR 访问程序存储器的指令需要一个额外的指令周期才能完成。例 2.2 演示了通过 FSR 访问程序存储器的过程。

如果标号指向程序存储器中的单元，HIGH 伪指令将 bit[7]置 1。

```
constants     ; example 2.2
RETW    DATA0     ; Index0 data
RETW    DATA1     ; Index1 data
RETW    DATA2
RETW    DATA3
my_function
;… LOTS OF CODE…
LDWI    LOW constants
STR     FSR1L
LDWI    HIGH constants
STR     FSR1H
MOVIW   0[FSR1]   ; THE PROGRAM MEMORY IS IN W
```

25. 特殊功能寄存器 (SPECIAL FUNCTION REGISTERS, SFR)

有 2 种特殊功能寄存器 (SFR)：

- 初始化配置寄存器：由仿真器界面设置 (Integrated Development Environment, IDE)；
- 用户寄存器；

25.1. 初始化配置寄存器

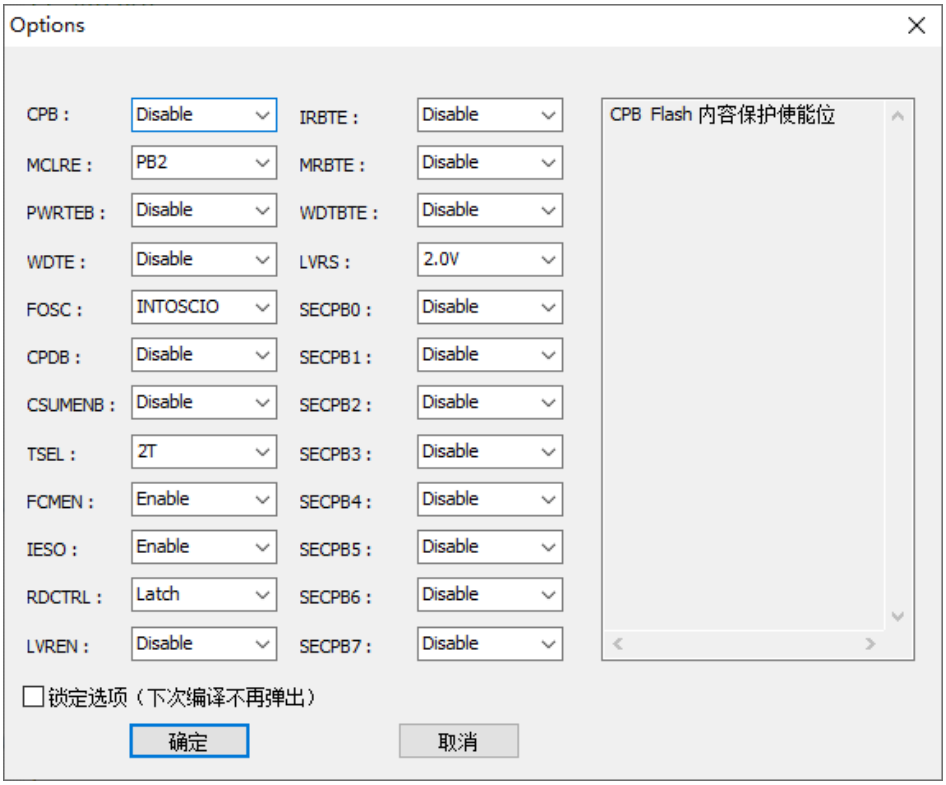


图 25-1 由 IDE 设置的初始化配置寄存器

名称	功能	默认
CPB	PROM 全区读保护	关闭
MCLRE	外部 I/O 复位	关闭
PWRTEB	上电延时定时器(PWRT)，初始化配置完成后额外延时~64ms	关闭
WDTE	WDT • 使能（指令不能禁止） • 由指令控制（SWDTEN）	SWDTEN 控制
FOSC	• LP：PA1 (+) 和 PA2(−) 接外部低速晶振 • XT：PA1 (+) 和 PA7 (−) 接外部高速晶振 • EC：PA1 (+) 接外部时钟输入，PA7 为 I/O • INTOSC：PA2 (或 PD5)输出系统时钟的 2 分频，PA1 为 IO • INTOSCIO：PA1 和 PA2 为 I/O	INTOSCIO
CPDB	DROM 全区读保护	关闭
CSUMEB	Checksum 使能 • 使能 • 关闭	关闭
TSEL	指令时钟与系统时钟 SysClk 的对应关系（2T or 4T）： • 2 (指令时钟= SysClk/2) • 4 (指令时钟= SysClk/4)	2
FCMEN	故障保护时钟监控器 • 使能 • 关闭	使能
IESO	XT / LP 双速时钟启动 • 使能 • 关闭	使能
RDCTRL	输出模式时读端口控制 • 返回的 PAD 上的值 • 返回的 Latch 上的值	Latch
LVREN	LVR • 使能 • 关闭 • 非 SLEEP 模式下使能 • 由 MSCON0 的 SLVREN 决定	关闭
IRBTE	IRERR 复位 BOOT 使能位 • 关闭 • 使能	关闭

名称	功能	默认
MRBTE	MCLR 复位 BOOT 使能位 • 关闭 • 使能	关闭
WDTBTE	WDT 复位 BOOT 使能位 • 关闭 • 使能	关闭
LVRS	7 档 V_{BOR} 电压(V)： 2.0 / 2.2 / 2.5 / 2.8 / 3.1 / 3.6 / 4.1	2.0
FSECPB0	PROM 扇区 0(1k x 14)读/写保护	关闭
FSECPB1	PROM 扇区 1(1k x 14)读/写保护	关闭
FSECPB2	PROM 扇区 2(1k x 14)读/写保护	关闭
FSECPB3	PROM 扇区 3(1k x 14)读/写保护	关闭
FSECPB4	PROM 扇区 4(1k x 14)读/写保护	关闭
FSECPB5	PROM 扇区 5(1k x 14)读/写保护	关闭
FSECPB6	PROM 扇区 6(1k x 14)读/写保护	关闭
FSECPB7	PROM 扇区 7(1k x 14)读/写保护	关闭

表 25-1 初始化配置寄存器 (由 IDE 设置)

25.2. 用户寄存器

用户寄存器，即特殊功能寄存器 (SFR) 分布在 8 个 bank 中，每个 bank 大小为 128 个字节。在访问寄存器前，必须先切换到相应的 bank。

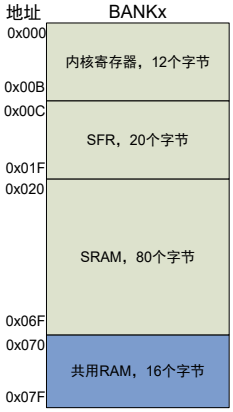


图 25-2 存储区构成

因为在切换 bank 时需要额外的指令，因此一些常用的 SFR 同时存储在 8 个 bank 中，以减少切换操作， 这些 8 个 bank 所共有的寄存器值是同步的。

Bank	首地址
Bank0	000H
Bank1	080H
Bank2	100H
Bank3	180H
Bank4	200H
Bank5	280H
Bank6	300H
Bank7	380H

表 25-2 BANK 首地址

地址	名称	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
首地址 + 0H	INDF	使用 FSR 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
首地址 + 2H	PCL	程序计数器 (PC) 低 8 位								0000 0000
首地址 + 3H	STATUS	FSRB8	PAGE[1:0]		/TF	/PF	Z	HC	C	0001 1xxx
首地址 + 4H	FSR	间接寻址指针寄存器								xxxx xxxx
首地址 + 0AH	PCLATH	–	–	–	程序计数器 (PC) 高 5 位锁存器					---0 0000
首地址 + 0BH	INTCON	GIE	PEIE	T0IE	INTE	PAIE	T0IF	INTF	PAIF	0000 0000
首地址 + (70 – 7F)	公共 BANK SRAM 区									xxxx xxxx

表 25-3 BANK 共有寄存器

地址	名称	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
00	INDF	使用 FSR 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
01	TMR0	Timer0 计数器								xxxx xxxx
02	PCL	程序计数器低 8 位								0000 0000
03	STATUS	FSRB8	PAGE[1:0]		/TF	/PF	Z	HC	C	0001 1xxx
04	FSR	间接寻址指针寄存器								xxxx xxxx
05	PORTA	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0	xxxx xxxx
06	PORTB	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0	xxxx xxxx
07	PORTC	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0	xxxx xxxx
08	PORTD	–	–	PD5	PD4	PD3	PD2	PD1	PD0	––xx xxxx
09	PIR2	–	–	–	–	–	–	C1IF	C0IF	–––– ––00
0A	PCLATH	–	–	–	程序计数器高 5 位锁存器					–––0 0000
0B	INTCON	GIE	PEIE	T0IE	INTE	PAIE	T0IF	INTF	PAIF	0000 0000
0C	PIR1	EEIF	CKMIF	LVDIF	ACMPIF	TMR1IF	OSFIF	TMR2IF	ADCIF	0000 0000
0D	P1ADTL	P1A 占空比寄存器低 8 位								0000 0000
0E	P1ADTH	P1A 占空比寄存器高 8 位								0000 0000
0F	P1BDTL	P1B 占空比寄存器低 8 位								0000 0000
10	P1BDTH	P1B 占空比寄存器高 8 位								0000 0000
11	P1CDTL	P1C 占空比寄存器低 8 位								0000 0000
12	P1CDTH	P1C 占空比寄存器高 8 位								0000 0000
13	P1DDTL	P1D 占空比寄存器低 8 位								0000 0000
14	P1DDTH	P1D 占空比寄存器高 8 位								0000 0000
15	TMR2L	Timer2 计数器[7:0]								0000 0000
16	TMR2H	Timer2 计数器[15:8]								0000 0000
17	PR2L	PR2[7:0], Timer2 周期寄存器低 8 位								1111 1111
18	PR2H	PR2[15:8], Timer2 周期寄存器高 8 位								1111 1111
19	P1BR0	P1BEVT	P1BKS[2:0]			P1BSS[1:0]		P1ASS[1:0]		0000 0000
1A	P1BR1	P1D2SS[1:0]		P1DSS[1:0]		P1C2SS[1:0]		P1CSS[1:0]		0000 0000
1B	P1POL0	P1C0P	P1B0P	P1A2NP	P1A2P	P1A1NP	P1A1P	P1A0NP	P1A0P	0000 0000
1C	P1POL1	P1D2P	P1D1P	P1D0P	–	–	P1C1P	P1B1P	–	000– –00–
1D	P1CON	P1AUE	P1DC[6:0]							0000 0000
1E	T2CON0	PR2U	TOUTPS[3:0]				TMR2ON	T2CKPS[1:0]		0000 0000
1F	T2CON1	–	–	–	P1OS	P1BZM	T2CKSRC[2:0]			–––0 0000
20–6F		SRAM BANK0 (80Bytes)								xxxx xxxx
70–7F		SRAM BANK0 (16Bytes)								xxxx xxxx

表 25-4 SFR, BANK 0

地址	名称	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
80	INDF	使用 FSR 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
81	OPTION	/PAPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111
82	PCL	程序计数器低 8 位								0000 0000
83	STATUS	FSRB8	PAGE[1:0]		/TF	/PF	Z	HC	C	0001 1xxx
84	FSR	间接寻址指针寄存器								xxxx xxxx
85	TRISA	PORTA 方向控制								1111 1111
86	TRISB	PORTB 方向控制								1111 1111
87	TRISC	PORTC 方向控制								1111 1111
88	TRISD	–	–	PORTD 方向控制						––11 1111
89	PIE	–	–	–	–	–	–	C1IE	C0IE	––––––00
8A	PCLATH	–	–	–	程序计数器高 5 位锁存器					–––0 0000
8B	INTCON	GIE	PEIE	T0IE	INTE	PAIE	T0IF	INTF	PAIF	0000 0000
8C	PIR1	EEIE	CKMIE	LVDIE	ACMPIE	TMR1IE	OSFIE	TMR2IE	ADCIE	0000 0000
8D	P1AUX	P1BKM	P1CDM	P1B2SS[1:0]		P1CF2E	P1CF2	P1DF2E	P1DF2	0000 0000
8E	PCON	LVDL[3:0]				LVDEN	LVDW	/POR	/BOR	0000 0xqq
8F	OSCCON	LFMOD	IRCF[2:0]			OSTS	HTS	LTS	SCS	0101 x000
90	UR1DATL	UR1DATL[7:0]								0000 0000
91	UR1DATH	–	–	–	–	–	–	–	UR1DATH	––––––––0
92	UR1CR1	UR1HDSEL	UR1STOP	UR1MODE	UR1EVEN	UR1PCEN	UR1RXEN	UR1TXEN	UR1EN	0000 0000
93	UR1CR2	UR1BRRH[3:0]				–	UR1ERRIE	UR1RXNEIE	UR1TXEIE	0000 –000
94	UR1BRRL	UR1BRRL[7:0]								0000 0000
95	UR1STAT	–	URxTXBSY	URxOVF	URxPEF	URxRXFULL	URxFEF	URxRXNEF	URxTXEF	–000 0001
96	OP0CR0	OP0OUT	OP0PSEL	OP0NSEL[1:0]		OP0FR[2:0]			OP0ON	x110 1100
97	OP0CR1	–	–	OPINSW	IPINSW	OP0FCAPE	–	OPTODIG	OPTOIO	––00 0–00
98	IOCA	IOCA[7:0]								0000 0000
99	WDTCON	–	WCKSRC[1:0]		WDTPS[3:0]				SWDTEN	–000 1000
9A	EEDAT	EEDAT[7:0]								0000 0000
9B	EEADR	EEADR[7:0]								0000 0000
9C	EECON1	–	Reserved	WREN3	WREN2	WRERR	WREN1	PONLY	RD	–000 x000
9D	EECON2	–	–	–	–	–	–	–	WR	––––––––0
9E	P1OE0	P1C0OE	P1B0OE	P1A2NOE	P1A2OE	P1A1NOE	P1A1OE	P1A0NOE	P1A0OE	0000 0000
9F	P1OE1	P1D2OE	P1D1OE	P1D0OE	–	–	P1C1OE	P1B1OE	–	000– –00–
A0–EF		SRAM BANK1 (80Bytes)								xxxx xxxx
F0–FF		SRAM 访问 BANK0 的 0x70–0x7F								xxxx xxxx

表 25-5 SFR, BANK 1

地址	名称	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
100	INDF	使用 FSR 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
101	MSCON1	–	–	–	–	LVRDEB	LVRCKS	AUXPGE	HIRCM	---- 1000
102	PCL	程序计数器低 8 位								0000 0000
103	STATUS	FSRB8	PAGE[1:0]		/TF	/PF	Z	HC	C	0001 1xxx
104	FSR	间接寻址指针寄存器								xxxx xxxx
105	T0CON0	T0DZ[1:0]		P0ANOE	P0AOE	T0ON	T0CKRUN	T0CKSRC[1:0]		0000 1000
106	T1CON0	–	T1AUSTP	T1ECKE	T1CKPSA	T1CKRUN	T1ON	T1CKSRC[1:0]		-000 0000
107	PR0	Timer0 产生的 PWM 波周期设置								1111 1111
108	T0DUTY	Timer0 产生的 PWM 波占空比设置								0000 0000
109	FOSCCAL	FOSCCAL[7:0]								xxxx xxxx
10A	PCLATH	–	–	–	程序计数器高 5 位锁存器					---0 0000
10B	INTCON	GIE	PEIE	T0IE	INTE	PAIE	T0IF	INTF	PAIF	0000 0000
10C	MSCON0	–	–	ROMLPE	CLKOS	SLVREN	CKMAVG	CKCNTI	T2CKRUN	0001 0000
10D	MSCON2	P0ANP	P0AP	CMAUSTR	STRTYP[1:0]		T1EVTS[2:0]			0000 0000
10E	LEBCON	LEBEN	LEBCH[1:0]		–	EDGS	–	–	–	000- 0---
10F	ADCMPH	ADCMPH[7:0]								0000 0000
110	ADDLY	ADDLY[7:0] / LEBPRL[7:0]								0000 0000
111	ADRESL	左对齐格式下 AD 结果的低 4 位或者右对齐下结果的低 8 位								xxxx xxxx
112	ADRESH	左对齐格式下 AD 结果的高 8 位或者右对齐格式下的高 4 位								xxxx xxxx
113	ADCON0	CHS[4:0]					ADEX	GO/DONE	ADON	0000 0000
114	ADCON1	ADFM	ADCS[2:0]			ADNREF[1:0]		ADPREF[1:0]		0010 0100
115	ADCON2	ADINTREF[1:0]		ETGTYP[1:0]		ADDLY.8	ETGSEL[2:0]			0100 0000
116	ADCON3	ADFBEN	ADCMPP	ADCMPE	–	LEBADT	–			000- 0---
117	PR1L	PR1 周期寄存器低 8 位 PR1[7:0]								1111 1111
118	PR1H	–	–	–	–	PR1 周期寄存器高 4 位 PR1[11:8]				---- 1111
119	TMR1L	Timer1[7:0]								0000 0000
11A	TMR1H	–	–	–	–	Timer1[11:8]				---- 0000
11B	TKCON0	–	–	TKRSEL[1:0]		TKMOD	TKCHGS[1:0]		TKON	--00 0000
11C	TKEN0	TKEN0[7:0]								0000 0000
11D	TKEN1	TKREN	–	TKEN1[5:0]						0-00 0000
11E	ANSEL0	ANSEL[7:0]								0000 0000
11F	ANSEL1	ANSEL[15:8]								0000 0000
120–16F		SRAM BANK2 (80Bytes)								xxxx xxxx
170–17F		SRAM 访问 BANK0 的 0x70–0x7F								xxxx xxxx

表 25-6 SFR, BANK 2

地址	名称	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
180	INDF	使用 FSR 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
182	PCL	程序计数器低 8 位								0000 0000
183	STATUS	FSRB8	PAGE[1:0]		/TF	/PF	Z	HC	C	0001 1xxx
184	FSR	间接寻址指针寄存器								xxxx xxxx
185	UR1DATL	UR1DATL[7:0]								0000 0000
186	T1CON0	–	–	–	–	–	–	–	UR1DATH	---- ---0
187	UR0CR1	UR0HDSEL	UR0STOP	UR0MODE	UR0EVEN	UR0PCEN	UR0RXEN	UR0TXEN	UR0EN	0000 0000
188	UR0CR2	UR0BRRH[3:0]				–	UR0ERRIE	UR0RXNEIE	UR0TXEIE	0000 -000
189	UR1BRRL	UR1BRRL[7:0]								0000 0000
18A	PCLATH	–	–	–	程序计数器高 5 位锁存器					---0 0000
18B	INTCON	GIE	PEIE	T0IE	INTE	PAIE	T0IF	INTF	PAIF	0000 0000
18C	SPIDAT	SPIDAT[7:0]								0000 0000
18D	SPICR0	–	–	–	–	NSSM		TXBMT	SPIEN	---- 0010
18E	SPICFG	SPIBUSY	MSTEN	CPHA	CPOL	SLAS	NSSVAL	SRMT	RXBMT	0000 0011
18F	SPISCR	SPISCR[7:0]								0000 0000
190	SPICRCP	SPICRCP[7:0]								0000 0111
191	SPIRXCRC	SPIRXCRC[7:0]								0000 0000
192	SPITXCRC	SPITXCRC[7:0]								0000 0000
193	SPIIER	–	–	–	–	WAKUP	RXERR	RXNE	TXE	---- 0000
194	SPICR1	BDM	BDOE	RXONLY	SSI	SSM	CRCNXT	CRCEN	LSBFIRST	0000 0000
195	SPISTAT	SPIF	WCOL	MODF	RXOVRN	–	–	WKF	CRCERR	0000 --00
196	I2CCR1	STRET	–	ADDRM	STOP	WRN	START	MASTER	I2CEN	1-00 0000
197	I2CCR2	I2CADDRH[1:0]			NACK	DUTY	–	–	–	0000 ---0
198	2CADDR	2CADDRL[7:0]								0000 0000
199	I2CCCR	I2CCCR[7:0]								1111 1111
19A	I2CDAT	I2CDAT[7:0]								0000 0000
19B	–	未实现，读 0								---- ----
19C	TKEN2	TKEN2[7:0]								0000 0000
19D	TKEN3	VREFM	TKCAPM	TKEN3[5:0]						0000 0000
19E	I2CISR	HOLDF	STOPF	ADDRF	NACKF	I2CBUSY	DIRF/ BUSERR	IICRXNEF	IICTXEF	0000 0000
19F	UR0STAT	–	UR0TXBSY	UR0OVF	UR0PEF	UR0RXFULL	UR0FEF	UR0RXNEF	UR0TXEF	-000 0001
1A0–1EF		SRAM BANK3 (80Bytes)								xxxx xxxx
1F0–1FF		SRAM 访问 BANK0 的 0x70–0x7F								xxxx xxxx

表 25-7 SFR, BANK 3

地址	名称	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
200	INDF	使用 FSR 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
201	–	未实现，读 0								---- ----
202	PCL	程序计数器低 8 位								0000 0000
203	STATUS	FSRB8	PAGE[1:0]		/TF	/PF	Z	HC	C	0001 1xxx
204	FSR	间接寻址指针寄存器								xxxx xxxx
205	ODCONA	ODCONA[7:0]								0000 0000
206	ODCONB	ODCONB[7:0]								0000 0000
207	ODCONC	ODCONC[7:0]								0000 0000
208	ODCOND	–	–	ODCOND[5:0]						--00 0000
209	TESTCFG0	TESTEN	BGROE	MGINM[1:0]		IREFOE	MGINEN	SCKOE	LDOOE	0000 0000
20A	PCLATH	–	–	–	程序计数器高 5 位锁存器					---0 0000
20B	INTCON	GIE	PEIE	T0IE	INTE	PAIE	T0IF	INTF	PAIF	0000 0000
20C	PSRCAL	PSRCAL[7:0],PORTA 源电流设置，低字节								1111 1111
20D	PSRCAH	PSRCAH[7:0] ,PORTA 源电流设置，高字节								1111 1111
20E	PSRCBL	PSRCBL[7:0] ,PORTB 源电流设置，低字节								1111 1111
20F	PSRCBH	PSRCBH[7:0] ,PORTB 源电流设置，高字节								1111 1111
210	PSRCCL	PSRCCL[7:0] ,PORTC 源电流设置，低字节								1111 1111
211	PSRCCH	PSRCCH[7:0] ,PORTC 源电流设置，高字节								1111 1111
212	PSRCDL	PSRCDL[7:0] ,PORTD 源电流设置，低字节								1111 1111
213	PSRCDH	–	–	–	–	PSRCDH[3:0] ,PORTD 源电流设置，高字节				---- 1111
214	PSINKA	PSINKA[7:0]，PORTA 灌电流设置								0000 0000
215	PSINKB	PSINKB[7:0]，PORTB 灌电流设置								0000 0000
216	PSINKC	PSINKC[7:0]，PORTC 灌电流设置								0000 0000
217	PSINKD	–	–	PSINKD[5:0]，PORTD 灌电流设置						--00 0000
218	SOSCPRL	SOSCPR[7:0]，慢时钟测量结果低 8 位								1111 1111
219	SOSCPRH	–	–	–	–	SOSCPR[11:8]，慢时钟测量结果高 4 位				---- 1111
21A	LVDCON	SYSON	LVDCKS[1:0]		LVDP	LVDDEB	—	LVDM[1:0]		-000 1100
21B	SECCODE	加密寄存器								xxxx xxxx
21C	LVDTRIM	–	LVDADJ[3:0]				LVRADJ[2:0]			-xxx xxxx
21D	–	–								xxxx xxxx
21E	–	–								xxxx xxxx
21F	–	–								xxxx xxxx
220–26F		SRAM BANK4 (80Bytes)								xxxx xxxx
270–27F		SRAM 访问 BANK0 的 0x70–0x7F								xxxx xxxx

表 25-8 SFR, BANK 4

地址	名称	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
280	INDF	使用 FSR 的内容对数据存储器进行访问（非物理寄存器）								xxxx xxxx
281	–	未实现，读 0								---- ----
282	PCL	程序计数器低 8 位								0000 0000
283	STATUS	FSRB8	PAGE[1:0]		/TF	/PF	Z	HC	C	0001 1xxx
284	FSR	间接寻址指针寄存器								xxxx xxxx
285	COMAF0	SCKPO	NSSPO	MOSIPO	MISOPO	SDAPO[1:0]		SCLPO[1:0]		0000 0000
286	COMAF1	INTPO[1:0]		TX0PO[2:0]			RX0PO[2:0]			0000 0000
287	COMAF2	UR1SW	UR0SW	–	TX1PO[1:0]		INTFIXB	RX1PO[1:0]		00-0 0100
28A	PCLATH	–	–	–	程序计数器高 5 位锁存器					---0 0000
28B	INTCON	GIE	PEIE	T0IE	INTE	PAIE	T0IF	INTF	PAIF	0000 0000
28C	ANSEL2	ANSEL2[7:0]								0000 0000
28D	ANSEL3	–	–	ANSEL3[5:0]						--00 0000
28E	TKCTRIM	–	–	Reserved		TKCTRIM[3:0]				--00 0000
28F	–	未实现，读 0								---- ----
290	WPDA	WPDA[7:0]，PORTA 弱下拉使能								0000 0000
291	WPDB	WPDB[7:0]，PORTB 弱下拉使能								0000 0000
292	WPDC	WPDC[7:0]，PORTC 弱下拉使能								0000 0000
293	WPDD	–	–	WPDD[5:0]，PORTD 弱下拉使能						--00 0000
294	WPUA	WPUA[7:0]，PORTA 弱上拉使能								1111 1111
295	WPUB	WPUB[7:0]，PORTB 弱上拉使能								0000 0000
296	WPUC	WPUC[7:0]，PORTC 弱上拉使能								0000 0000
297	WPUD	–	–	WPUD[5:0]，PORTD 弱上拉使能						--00 0000
298	–	未实现，读 0								---- ----
299	–	未实现，读 0								---- ----
29A	DAC1DAT	–	DAC1DAT[6:0]							-000 0000
29B	DAC2DAT	–	DAC2DAT[6:0]							-000 0000
29C	DACCON0	CMDEBS	DEBPR[1:0]		–	WNDEN	DACEN	DACVREF[1:0]		000- 0000
29D	CM0CON0	CM0EN	CM0POL	CM0OE	C0OUT	CM0PSEL[1:0]		CM0NSEL[1:0]		000x 0000
29E	CM1CON0	CM1EN	CM1POL	CM1OE	C1OUT	CM1PSEL[1:0]		CM1NSEL[1:0]		000x 0000
29F	–	未实现，读 0								---- ----
2A0–2EF		SRAM BANK5 (80Bytes)								xxxx xxxx
2F0–2FF		SRAM 访问 BANK0 的 0x70–0x7F								xxxx xxxx

表 25-9 SFR, BANK 5

地址	名称	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
300	INDF	使用 FSR 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
301	MSCON1	–	–	–	–	LVRDEB	LVRCKS	AUXPGE	HIRCM	---- 1000
302	PCL	程序计数器低 8 位								0000 0000
303	STATUS	FSRB8	PAGE[1:0]		/TF	/PF	Z	HC	C	0001 1xxx
304	FSR	间接寻址指针寄存器								xxxx xxxx
309	FOSCCALH	–	–	–	–	–	–	FOSCCAL[9:8]		---- --xx
30A	PCLATH	–	–	–	程序计数器高 5 位锁存器					---0 0000
30B	INTCON	GIE	PEIE	T0IE	INTE	PAIE	T0IF	INTF	PAIF	0000 0000
320–36F		SRAM BANK6 (80Bytes)								xxxx xxxx
370–37F		SRAM 访问 BANK0 的 0x70–0x7F								xxxx xxxx

表 25-10 SFR, BANK 6

地址	名称	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
380	INDF	使用 FSR 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
381	–	未实现，读 0								---- ----
382	PCL	程序计数器低 8 位								0000 0000
383	STATUS	FSRB8	PAGE[1:0]		/TF	/PF	Z	HC	C	0001 1xxx
384	FSR	间接寻址指针寄存器								xxxx xxxx
38A	PCLATH	–	–	–	程序计数器高 5 位锁存器					---0 0000
38B	INTCON	GIE	PEIE	T0IE	INTE	PAIE	T0IF	INTF	PAIF	0000 0000
3A0–3C7		SRAM BANK7 (40Bytes)								xxxx xxxx
3C8–3EF		未实现，读 0								---- ----
3F0–3FF		SRAM 访问 BANK0 的 0x70–0x7F								xxxx xxxx

表 25-11 SFR, BANK 7

注：

1. INDF 不是物理寄存器；
2. 灰色部分表示没有实现；
3. 不要对未实现的寄存器位进行写操作；

25.3. STATUS 寄存器

名称	状态	寄存器	地址	复位值
FSRB8	FSR 寄存器第 8 位，与 FSR 组成一个 9 位的寄存器，在间接寻址时使用 详见 INDF 和 FSR 寄存器一节	STATUS[7]	Bank 首地址 +0x03	RW-0
PAGE[1:0]	寄存器存储区(bank)选择位 (用于直接寻址) 00 = Bank0 (0x00h – 0x7Fh) 01 = Bank1 (0x80h – 0xFFh) 10 = Bank2 (0x100 – 0x17F) 11 = Bank3 (0x180 – 0x1FF) 注：访问 Bank4~Bank7 需把 AUXPGE 置 1	STATUS[6:5]		RW-00
/TF	/TF:超时状态位 1 = 上电后，执行了 CLRWDT 指令或 SLEEP 指令 0 = 发生 WDT 超时溢出	STATUS[4]		RO-1
/PF	/PF:掉电标志位 1 = 上电复位后或执行了 CLRWDT 指令 0 = 执行了 SLEEP 指令	STATUS[3]		RO-1
Z	0 标志位：算术或逻辑运算的结果为零？ 1 = Yes 0 = No	STATUS[2]		RW-x
HC	半进位 / 半借位 (ADDWF, ADDLI, SUBL, SUBWF)：结果的第 4 位向高位发生了进位或借位？ 1 = 进位，或未借位 0 = 未进位，或借位	STATUS[1]		RW-x
C	进位 / 借位 (ADDWF, ADDWI, SUBWI, SUBWF)：结果的最高位发生了进位或借位？ 1 = 进位，或未借位 0 = 未进位，或借位	STATUS[0]		RW-x

表 25-12 STATUS 寄存器

/TF	/PF	条件
1	1	上电或者低电压复位
0	–	WDT 复位
0	0	WDT 唤醒
U	U	正常运行下发生 MCLR 复位
–	0	睡眠状态下发生 MCLR 复位

注意：

1. “–“ 表示无变化；
2. 和其它寄存器一样，状态寄存器也可以作为任何指令的目标寄存器。如果一条指令影响 Z、HC 或 C 位的指令以状态寄存器作为目标寄存器，将禁止对这三位的写操作，它们只受逻辑结果影响，被置 1 或清 0。因此，当执行一条把状态寄存器作为目标寄存器的指令后，STATUS 内容可能和预想的不一致；
3. 建议只使用 BCR、BSR、SWAPR 和 STR 指令来改变状态寄存器。

25.4. PCL 和 PCLATH

程序计数器 (PC) 为 13 位宽。其低 8 位来自可读写的 PCL 寄存器，高 5 位 (PC[12:8])来自 PCLATH，不能直接读写。只要发生复位，PC 就将被清 0。下图显示了装载 PC 值的两种情形。

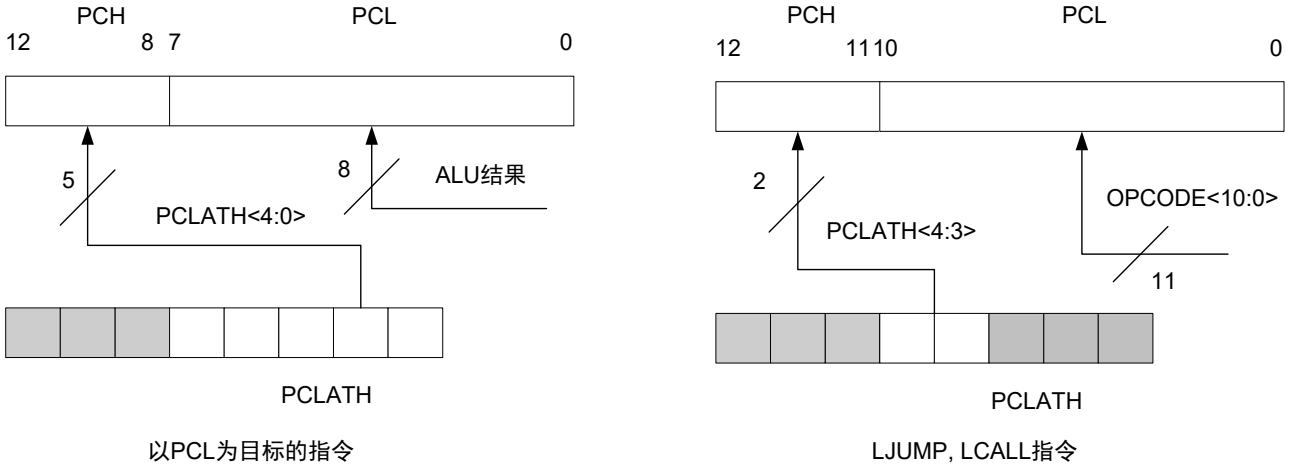


图 25-3 不同情况下 PC 的加载

25.4.1. 修改 PCL

执行任何以 PCL 寄存器为目标寄存器的指令将同时使程序计数器 PC[12:8]位被 PCLATH 内容取代。这样可通过将所需的高 5 位写入 PCLATH 寄存器来改变程序计数器的所有内容。

计算 LJUMP 指令是通过向程序计数器加入偏移量 (ADDWR PCL) 来实现的。通过修改 PCL 寄存器跳转到查找表或程序分支表 (计算 LJUMP) 时应特别谨慎。假定 PCLATH 设置为表的起始地址，如果表长度大于 255 条指令，或如果存储器地址的低 8 位在表的中间从 0xFF 计满返回到 0x00，那么在每次表起始地址与表内的目标地址之间发生计满返回时，PCLATH 必须递增。

25.5. INDF 和 FSR 寄存器

INDF 不是物理存在的寄存器，对 INDF 进行寻址将产生间接寻址。任何使用 INDF 寄存器的指令，实际上是对文件选择寄存器{FSRB8, FSR}所指向的单元进行存取，其中{FSRB8, FSR}组成 9 位地址，可寻址范围为 0 – 511。

在 FT62F28x 系列芯片中，SFR 空间扩展为 1kB，当 AUXPGE 为 0 时，间接寻址访问地址为 0x000~0x1FF，当 AUXPGE 为 1 时，访问 0x200~0x3FF。

名称	状态	寄存器	地址	复位值
AUXPGE	SFR BANK 间接或直接寻址访问 1 = BANK4 ~ 7 0 = BANK0 ~ 3	MSCON1[1]	0x101 /0x301	RW-0

间接对 INDF 进行读操作将返回 0。间接对 INDF 进行写将导致空操作 (可能会影响状态标志位)。

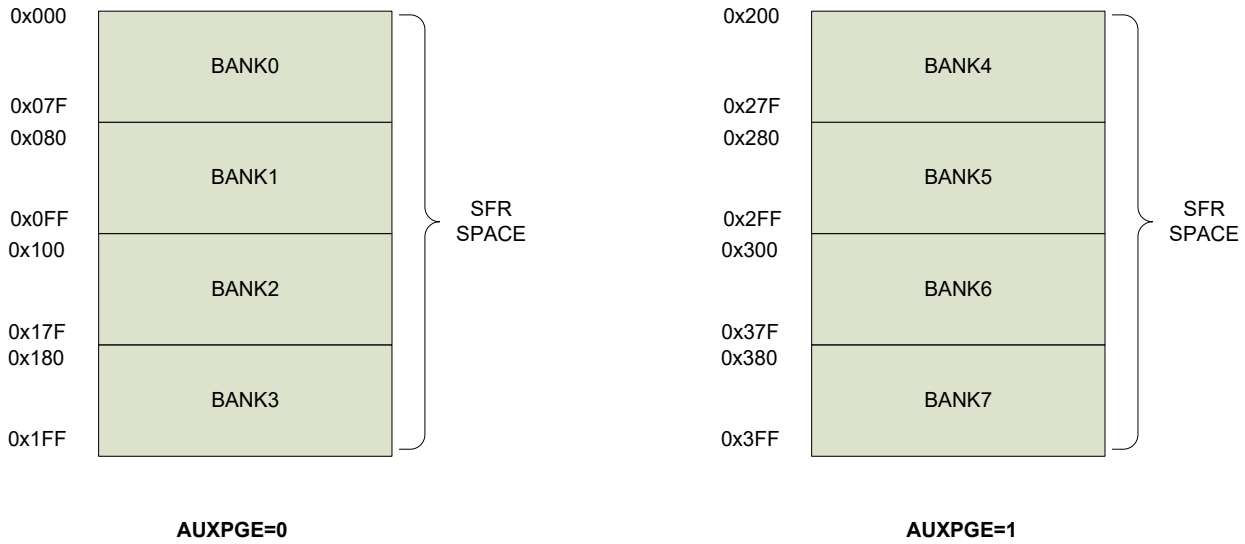


图 25-4 间接寻址

25.6. 关于寄存器保留位

如 25.2 章节的表格所示，SFR 空间有部分寄存器或者寄存器位没有实现，未实现的寄存器位是保留位，软件读返回的是 0，写无效。

不建议程序对这些保留位写 1，这有可能会给以后程序的移植带来问题，因为后续芯片产品可能会用到这此位。

25.7. 关于 BANK4~BANK7 的访问

不同于 Bank0~Bank3 的访问 (汇编时使用伪指令 BANKSEL，在 C 语言使用直接对 SFR 赋值)，访问 Bank4~Bank7 时需要手动把 MSCON1 寄存器位 AUXPGE 置 1，在开发应用时尤其要注意。

例如要对 ODCONA 写 0xF5，应使用以下形式：

```
MSCON1 = 0x02;      // 把 AUXPGE 置 1
ODCONA = 0xF5;
…
MSCON1 = 0x00;      // 把 AUXPGE 清 0
```

另外，如果中断服务程序涉及到 AUXPGE 修改操作的，则要对 AUXPGE 进行现场保护，退出中断前再恢复现场，以下是一段示例代码。

```
void interrupt ISR()
 {
   AUXPGE_SHADOW = AUXPGE; //处理中断前，保存 AUXPGE，AUXPGE_SHADOW 为全局位变量
   ...                                  // 其它代码
   AUXPGE = AUXPGE_SHADOW; // 退出中断前，恢复 AUXPGE
}
```

26. 指令集汇总

汇编语法	功能	运算	状态位
NOP	空操作	None	NONE
SLEEP	进入 SLEEP 模式	0 → WDT; Stop OSC	/PF, /TF
CLRWDT	清看门狗 (喂狗)	0 → WDT	/PF, /TF
LJUMP N	无条件跳转	N → PC	NONE
LCALL N	调用子程序	N → PC; PC + 1 → Stack	NONE
RETI	从中断返回	Stack → PC; 1 → GIE	NONE
RET	从子程序返回	Stack → PC	NONE
BCR R, b	将寄存器 R 的 b 位清 0	0 → R(b)	NONE
BSR R, b	将寄存器 R 的 b 位置 1	1 → R(b)	NONE
CLRR R	将寄存器 R 清 0	0 → R	Z
LDR R, d (MOVF)	将 R 存到 d	R → d	Z
COMR R, d	R 的反码	/R → d	Z
INCR R, d	R + 1	R + 1 → d	Z
INCRSZ R, d	R + 1，结果为 0 则跳过	R + 1 → d	NONE
DECR R, d	R − 1	R − 1 → d	Z
DECRSZ R, d	R − 1，结果为 0 则跳过	R − 1 → d	NONE
SWAPR R, d	将寄存器 R 的半字节交换	R(0-3)R(4−7) → d	NONE
RRR R, d	R 带进位循环右移	R(0) → C; R(n) → R(n−1); C → R(7);	C
RLR R, d	R 带进位循环左移	R(7) → C; R(n) → R(n+1); C → R(0);	C
BTSC R, b	位测试，结果为 0 则跳过	Skip if R(b)=0	NONE
BTSS R, b	位测试，结果为 1 则跳过	Skip if R(b)=1	NONE
CLRW	将工作寄存器 W 清 0	0 → W	Z
STTMD	将 W 内容存到 OPTION	W → OPTION	NONE
CTLIO R	设置 I/O 方向控制寄存器 TRISr	W → TRISr	NONE
STR R (MOVWF)	将 W 存到 R	W → R	NONE
ADDWR R, d	W 与 R 相加	W + R → d	C, HC, Z
SUBWR R, d	R 减 W	R − W → d	C, HC, Z
ANDWR R, d	W 与 R 相与	R & W → d	Z
IORWR R, d	W 与 R 相或	W | R → d	Z
XORWR R, d	W 与 R 异或	W ^ R → d	Z
LDWI I (MOVLW)	将立即数存到 W	I → W	NONE
ANDWI I	W 与立即数 I 相与	I & W → W	Z
IORWI I	W 与立即数 I 相或	I | W → W	Z

汇编语法	功能	运算	状态位
XORWI I	W 与立即数 I 异或	I ^ W → W	Z
ADDWI I	W 与立即数 I 相加	I + W → W	C, HC, Z
SUBWI I	立即数 I 减 W	I − W → W	C, HC, Z
RETW I	返回，将立即数 I 存到 W	Stack → PC; I → W	NONE

表 26-1 37 条 RISC 指令

字段	描述
R(F)	SFR/GPR 地址
W	工作寄存器
b	8-bit 寄存器 R / RAM 中的位地址
I / Imm (k)	立即数
X	不关心，值可以是 0 或 1
d	目标寄存器选择 1 = 结果存放到寄存器 R / RAM 0 = 结果存放到 W
N	程序绝对地址
PC	程序计数器
/PF	掉电标志位
/TF	超时标志位
TRISr	TRISr 寄存器, r 可以是 A, B, C D
C	进位 / 借位
HC	半进位 / 半借位
Z	0 标志位

表 26-2 操作码字段

27. 芯片的电气特性

27.1. 极限参数

工作温度等级 3……-40~+85°C
工作温度等级 2……-40~+105°C
工作温度等级 1……-40~+125°C
存储温度……-40~+125°C
电源电压……V_{SS}-0.3V~V_{SS}+6.0V
端口输入电压……V_{SS}-0.3V~V_{DD}+0.3V

注：

1. 超过上述“极限参数”所规定的范围，可能会对芯片造成永久性损坏。
2. 除非另作说明，所有特性值的测试条件为 25°C, V_{DD} =1.9 – 5.5V。
3. 本节所示的值和范围基于特性值，并非最终出货的标准值。除非另作说明，生产测试温度为 25°C。

27.2. 工作特性

参数		最小值	典型值	最大值	单位	条件/备注
系统时钟频率(F_{sys})	2T/4T	—	—	8	MHz	-40°C~85°C,V_{DD}=1.9~5.5V
		—	—	16	MHz	-40°C~85°C,V_{DD} =2.7~5.5V
指令周期(Tins)	2T	—	125	—	ns	系统时钟 HIRC
	4T	—	250	—	ns	
	2T	—	61	—	μs	系统时钟 LIRC
	4T	—	122	—	μs	
上电复位保持时间(T_{DRH})		—	4.2	—	ms	25°C, PWRT disable
外部复位脉冲宽度(T_{MCLRB})		2000	—	—	ns	25°C
WDT 周期(T_{WDT})		—	1	—	ms	无预分频, WDTPS[3:0]=0000

27.3. POR, LVR, LVD

上电复位 (POR)

电气参数	最小值	典型值	最大值	单位	条件/备注
I_{POR} 工作电流	—	140	—	nA	25°C , V_{DD} = 3.3V
V_{POR}	—	1.65	—	V	25°C

低电压复位 LVR)

电气参数	最小值	典型值	最大值	单位	条件/备注
I_{LVR} 工作电流	—	14.7	—	μA	25°C, V_{DD} = 3.3V
V_{LVR}，LVR 阈值	1.94	2.0	2.06	V	25°C
	2.13	2.2	2.27		
	2.42	2.5	2.58		
	2.72	2.8	2.88		
	3.01	3.1	3.19		
	3.49	3.6	3.71		
	3.98	4.1	4.22		
LVR delay	—	125	157	μs	25°C, V_{DD} = 1.9~5.5V

低电压侦测 LVD)

电气参数	最小值	典型值	最大值	单位	条件/备注
I_{LVD} 工作电流	—	16.1	—	μA	25°C, V_{DD} = 3.3V
V_{LVD}，LVD 阈值	1.16	1.2	1.24	V	25°C
	1.94	2.0	2.06		
	2.33	2.4	2.47		
	2.62	2.7	2.78		
	2.91	3.0	3.09		
	3.20	3.3	3.40		
	3.49	3.6	3.71		
	3.88	4.0	4.12		
LVD delay	—	125	157	μs	25°C, V_{DD} = 1.9~5.5V

27.4. I/O PAD 电路

电气参数		最小值	典型值	最大值	单位	条件/备注
V_{IL}		0	—	0.3* V_{DD}	V	
V_{IH}		0.7* V_{DD}	—	V_{DD}	V	
漏电流		-1	—	1	μA	V_{DD} = 5V
源电流(source)	L0	—	-3	—	mA	25°C, V_{DD} = 5V, V_{OH}= 4.5V
	L1	—	-6	—		
	L2	—	-18	—		
灌电流(sink)	L0	—	45	—	mA	25°C, V_{DD} = 5V, V_{OL}= 0.5V
	L1	—	60	—		
上拉电阻		—	20	—	kΩ	
下拉电阻		—	20	—	kΩ	

27.5. 工作电流(I_{DD})

电气参数	Sysclk	典型值@V_{DD}			单位
		2.0V	3.0V	5.5V	
正常模式(2T), I_{DD}	16MHz	—	1.250	1.310	mA
	8MHz	0.640	0.892	0.926	
	4MHz	0.514	0.708	0.734	
	2MHz	0.433	0.478	0.483	
	1MHz	0.262	0.287	0.289	
	32kHz	0.025	0.031	0.032	
休眠模式(Sleep, ~~WDT OFF, LVR OFF~~), I_{SB}	—	0.20	0.48	0.55	μA
休眠模式(Sleep, WDT ON, ~~LVR OFF~~)	—	1.11	1.71	1.81	
休眠模式(Sleep, ~~WDT OFF~~, LVR ON)	—	11.48	15.40	21.01	
休眠模式(Sleep, WDT ON, LVR ON)	—	17.31	20.41	24.63	
休眠模式(Sleep, WDT OFF, ~~LVR OFF~~, LVD ON)	—	18.57	20.49	24.80	

注：

1. 睡眠电流的测试条件为 I/O 处于输入模式并外部下拉到 0；

27.6. 内部荡器

内部低频振荡器 (LIRC)

测试条件为 LIRC 选择 32 kHz (LFMOD = 0)。

电气参数	最小值	典型值	最大值	单位	条件/备注
振荡频率	30.4	32	33.6	kHz	25°C, V_{DD} = 2.5V
随温度变化范围	-3.0%	—	3.0%	—	-40 ~ 85°C, V_{DD} = 2.5V
随电源电压变化范围	-1.0%	—	1.0%	—	25°C, V_{DD} = 1.9~5.5V
I_{LIRC} 工作电流	—	1.3	—	μA	25°C, V_{DD} = 3.0V
启动时间	—	4.6	—	μs	25°C, V_{DD} = 3.0V

内置高频振荡器(HIRC)

高频振荡器分为两个模块：13.56MHz、16MHz。

电气参数	最小值	典型值	最大值	单位	条件/备注
13.56M 校准范围	13.42	13.56	13.69	MHz	25°C, V_{DD} = 2.5V
13.56M 随温度变化范围	-3.5%	—	3.5%	—	-40~85°C, V_{DD} = 2.5V
16M 校准范围	15.84	16	16.16	MHz	25°C, V_{DD} = 2.5V
16M 随温度变化范围	-3.0%	—	3.0%	—	-40~85°C, V_{DD} = 2.5V
随电源电压变化范围	-1.0%	—	1.0%	—	25°C, V_{DD} = 1.9~5.5V
I_{HIRC} 工作电流	—	136	—	μA	25°C, V_{DD} = 3.0V
启动时间	—	2.5	—	μs	25°C, V_{DD} = 3.0V

27.7. 7bit DAC 电路 (比较器参考电压设置)

电气参数	最小值	典型值	最大值	单位	条件/备注
DAC VREF	1.99	2	2.01	V	-40°C ~85°C,V_{DD} =2.5~5.5V
	2.99	3	3.02		-40°C ~85°C,V_{DD} =3.5~5.5V
	3.98	4	4.02		-40°C ~85°C,V_{DD} =4.5~5.5V
相对精度	—	VREF/128	—	V	-40°C ~85°C,V_{DD} =2.0~5.5V
绝对精度	—	—	1/2*	LSB	同上
单位电阻(unit resistor)	—	5000*	—		同上

27.8. ADC(12bit) 和 ADC VREF

ADC(12bit)

电气参数	最小值	典型值	最大值	单位	条件/备注
ADC 工作电压 V_{DD}	2.7	—	5.5	V	
ADC 工作电流 I_{VDD}	—	120	—	μA	25°C, V_{REFP} = V_{DD} = 2.7V, ADC 转换时钟频率为 250kHz
	—	125	—	μA	25°C, V_{REFP} = V_{DD} = 3.0V, ADC 转换时钟频率为 250kHz
	—	155	—	μA	25°C, V_{REFP} = V_{DD} = 5.5V, ADC 转换时钟频率为 250kHz
模拟输入电压 V_{AIN}	VREFN	—	VREFP	V	
外部参考电压 V_{REF}	—	—	V_{DD}	V	
分辨率	—	—	12	位	
积分误差 E_{IL}	—	±3	—	LSB	25°C, V_{REFP} = V_{DD} = 5.0V, V_{REFN}= GND，ADC 转换时钟频率为 250kHz
微分误差 E_{DL}	—	±2	—	LSB	
偏移误差 E_{OFF}	—	±3	—	LSB	
增益误差 E_{GN}	—	±5	—	LSB	
转换时钟周期 TAD	—	2	—	μs	V_{REFP} > 3.0V, V_{DD} > 3.0V
转换时钟数	—	15	—	TAD	
稳定时间(T_{ST})	—	15	—	μs	
采样时间(T_{ACQ})	—	1.5	—	TAD	
建议模拟电压源阻抗(ZAI)	—	—	10	kΩ	

ADC V_{REF}

电气参数	最小值	典型值	最大值	单位	条件/备注
内置参考电压 ADC Vref	0.495	0.5	0.505	V	25°C, V_{DD} =5V
	1.990	2	2.010	V	25°C, V_{DD} =5V
	2.985	3	3.015	V	25°C, V_{DD} =5V
内置参考电压 0.5V 稳定时间 T_{VRINT}	—	400	—	μs	25°C, V_{DD} =5V
	—	600	—	μs	25°C, V_{DD} =5V, 1μF
内置参考电压 2.0V 稳定时间 T_{VRINT}	—	450	—	μs	25°C, V_{DD} =5V
	—	800	—	μs	25°C, V_{DD} =5V, 1μF
内置参考电压 3.0V 稳定时间 T_{VRINT}	—	450	—	μs	25°C, V_{DD} =5V
	—	1200	—	μs	25°C, V_{DD} =5V, 1μF

27.9. 运算放大器 0 特性

电气参数	最小值	典型值	最大值	单位	条件/备注
I_{VDD} 工作电流	—	70	—	μA	V_{DD} =5V, 正常模式
工作电压	2.5	—	5.5	V	—
输入失调电压 V_{OS}	—	12	—	mV	未校准
共模输入范围	V_{SS}	—	V_{DD} -1.4	V	V_{DD} =5V
增益带宽	—	3.64	—	MHz	RL=1M, CL=100pF
电源电压抑制比	—	100	—	dB	V_{DD} =5V
共模抑制比	—	100	—	dB	VCM=V_{DD}/2, V_{DD} =5V
SR 转换速率	—	1.4	—	V/μs	空载, 5V

27.10. Comparator 比较器

电气参数		最小值	典型值	最大值	单位	条件/备注
I_{VDD} 工作电流	CM0	—	135	—	μA	25°C, V_{DD} = 3V
	CM1	—	105	—	μA	25°C, V_{DD} = 3V
工作电压		1.9	—	5.5	V	
共模输入电压	CM0	1.0	—	V_{DD}	V	-40°C~85°C, V_{DD} = 1.9~5.5V
	CM1	V_{SS}	—	V_{DD}-1.0		
输入偏移电压 (V_{OS})	CM0	—	±3	—	mV	-40°C~85°C, Vin = V_{DD}/2, V_{DD} = 2.0V~5.5V
	CM1	—	±3	—		
共模抑制比(CMRR)		55	—	—	dB	-40°C~85°C,V_{DD} = 1.9~5.5V
迟滞(Hysteresis)		—	0	—	mV	-40°C~85°C,V_{DD} = 1.9~5.5V
响应时间 (Response Time)	CM0	—	55	—	ns	正常模式：输出低->高
		—	50	—	ns	正常模式：输出高->低
	CM1	—	55	—	ns	正常模式：输出低->高
		—	40	—	ns	正常模式：输出高->低
稳定时间(Settle Time)		—	—	10	μs	00000->11111

27.11. 存储器编程特性

参数		最小值	典型值	最大值	单位	条件
$V_{DD\text{-}READ}$	Program / Data EE 读电压	V_{POR}	–	5.5	V	-40 – 85 / 105°C
$V_{DD\text{-}WRITE}$	Program EE 写电压	2.5	–	5.5	V	-40 – 85 / 105°C
	Data EE 写电压	1.9	–	5.5		
N_{END}	Program EE 擦/写次数	100 k	–	–	cycle	25 °C
		40 k	–	–		85 °C
		10 k	–	–		105 °C
	Data EE 擦/写次数	1,000 k	–	–		25 °C
		400 k	–	–		85 °C
		100 k	–	–		105 °C
T_{RET}	Program EE 数据保持	20	–	–	年	1k 次擦写后 @ 85 °C
		10	–	–		1k 次擦写后 @ 105 °C
	Data EE 数据保持	20	–	–		10k 次擦写后 @ 85 °C
		10	–	–		10k 次擦写后 @ 105 °C
T_{WRITE}	Data EE 写时间	0.7	—	1.3	ms	关闭自动擦除
		2.0	—	4.0		使能自动擦除
I_{PROG}	Data EE 编程电流	–	–	300	μA	25 °C, V_{DD} = 3 V

27.12. EMC 特性

ESD

电气参数		最小值	典型值	最大值	单位	条件/备注
V_{ESD}	HBM	8000	—	—	V	MIL-STD-883H Method 3015.8
	MM	200	—	—	V	JESD22-A115

Latch-up

电气参数	最小值	典型值	最大值	单位	条件/备注
LU, static latch-up	200	—	—	mA	EIA/JESD 78

EFT

电气参数	最小值	典型值	最大值	单位	条件/备注
V_{EFT}	5.5	—	—	kV	V_{DD} (5V) 对地接 1μF 电容

27.13. 直流和交流特性曲线图

注意：本节提供的图表基于特性值，仅用作设计参考，未经生产测试。

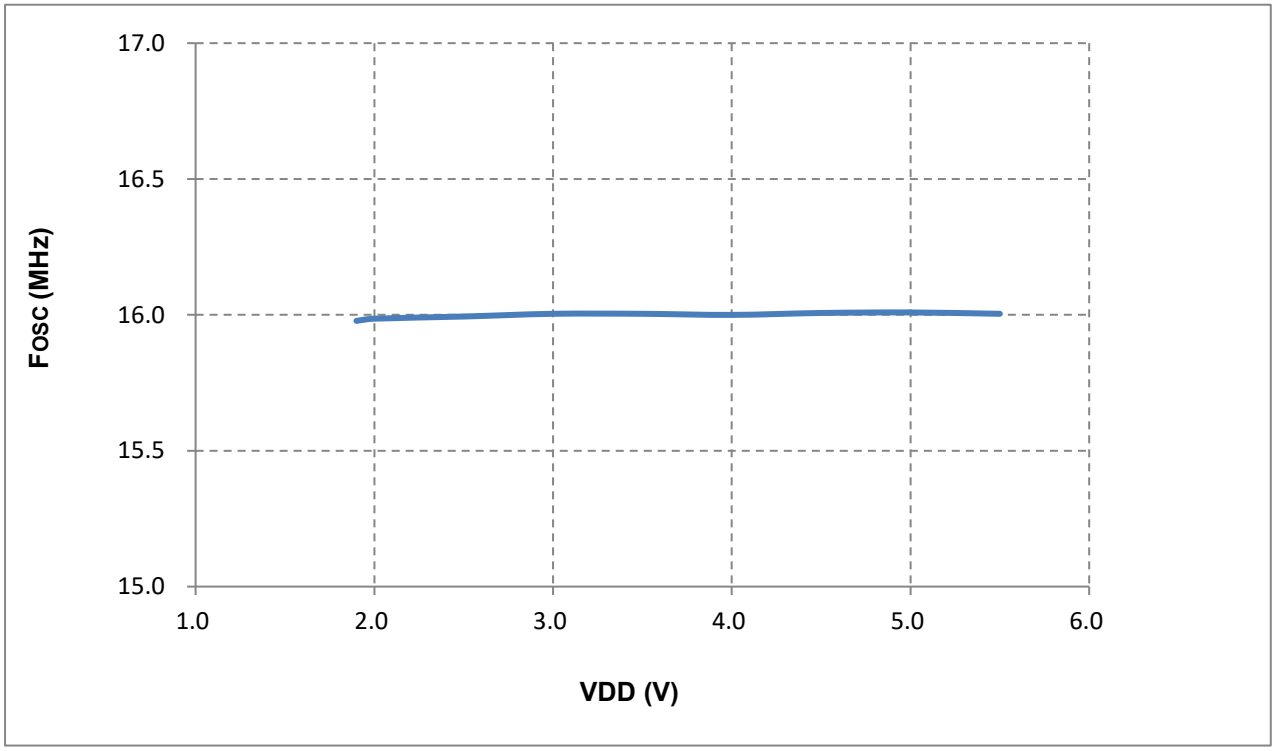


图 27-1 HIRC (16M) vs.V_{DD} (T_A = 25°C)

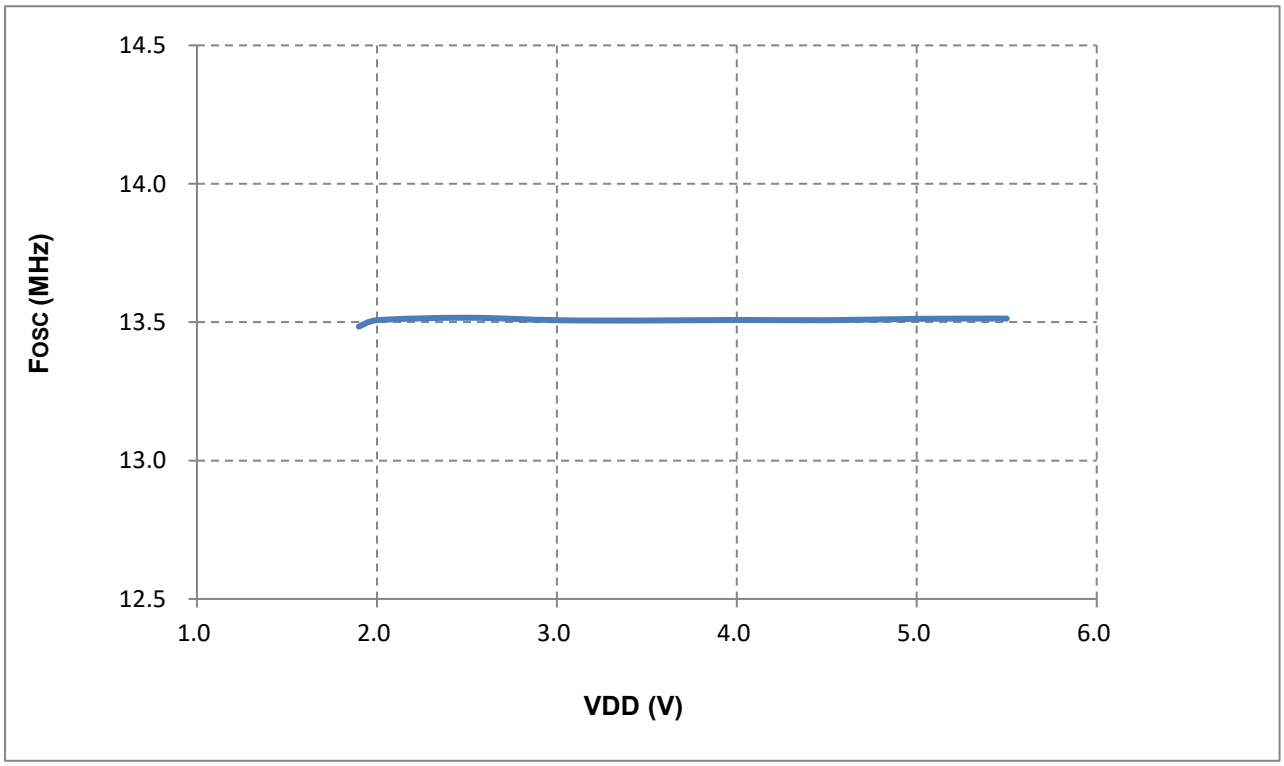


图 27-2 HIRC (13.56M) vs.VDD (TA = 25°C)

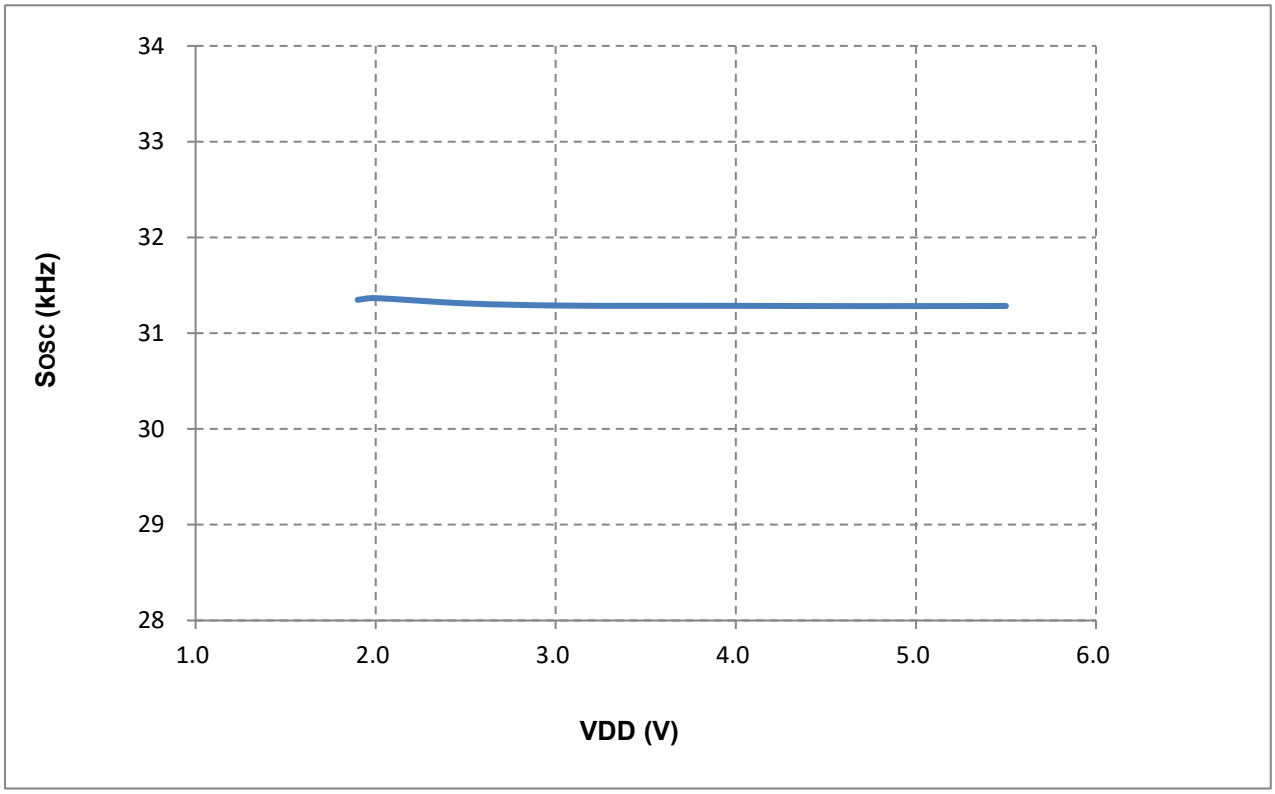


图 27-3 LIRC vs. VDD (TA=25°C)

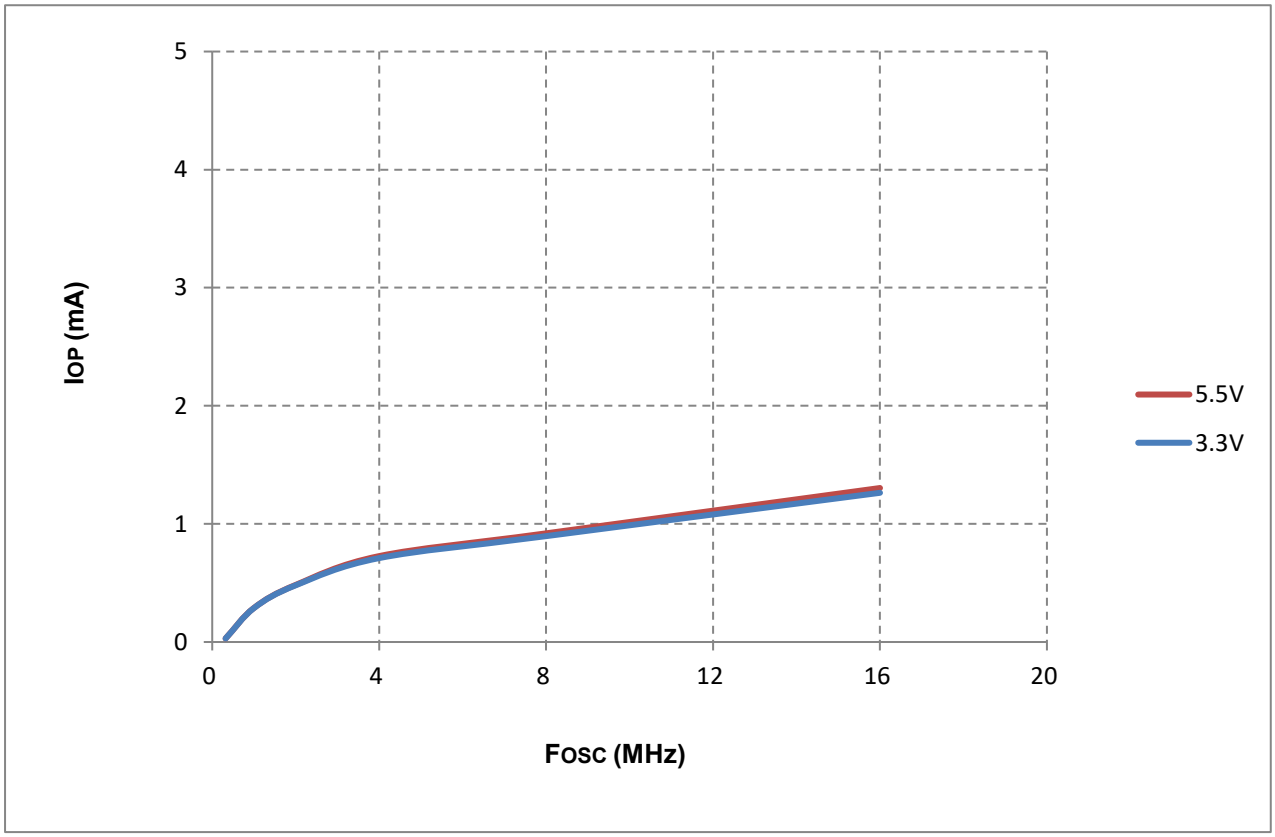


图 27-4 不同 VDD 下，IDD vs. Freq (2T, TA=25°C)

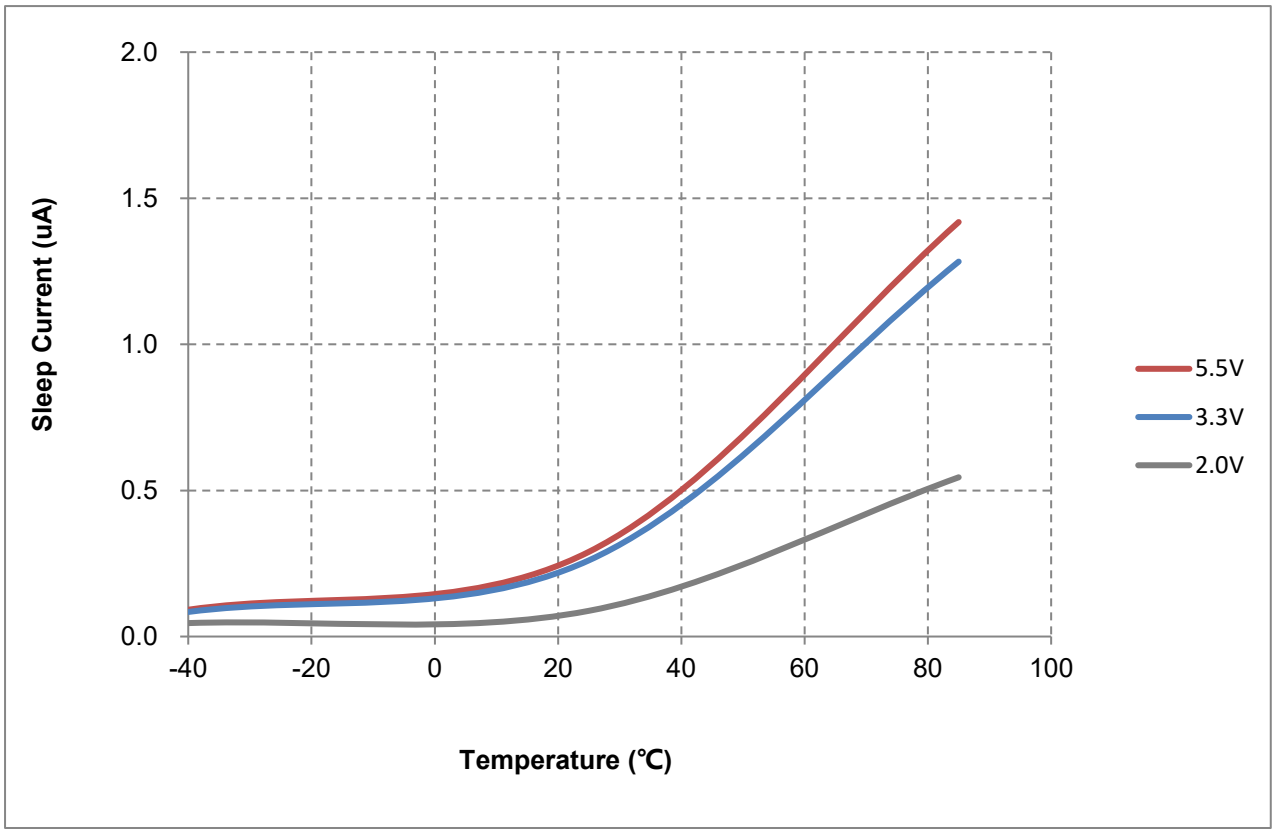


图 27-5 不同 VDD 下，ISB (睡眠电流)随温度变化曲线

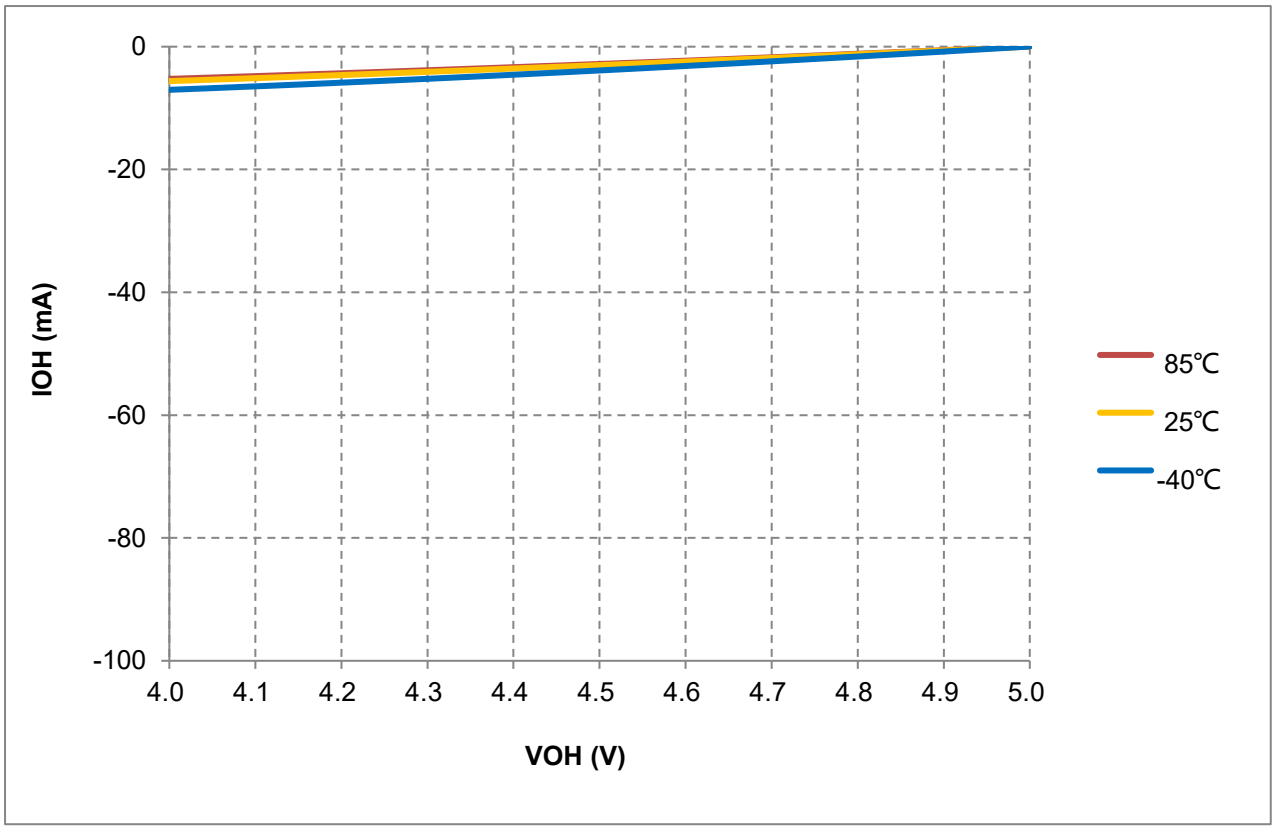


图 27-6 不同温度下，IOH (L0 = -3mA) vs. VOH @VDD=5V

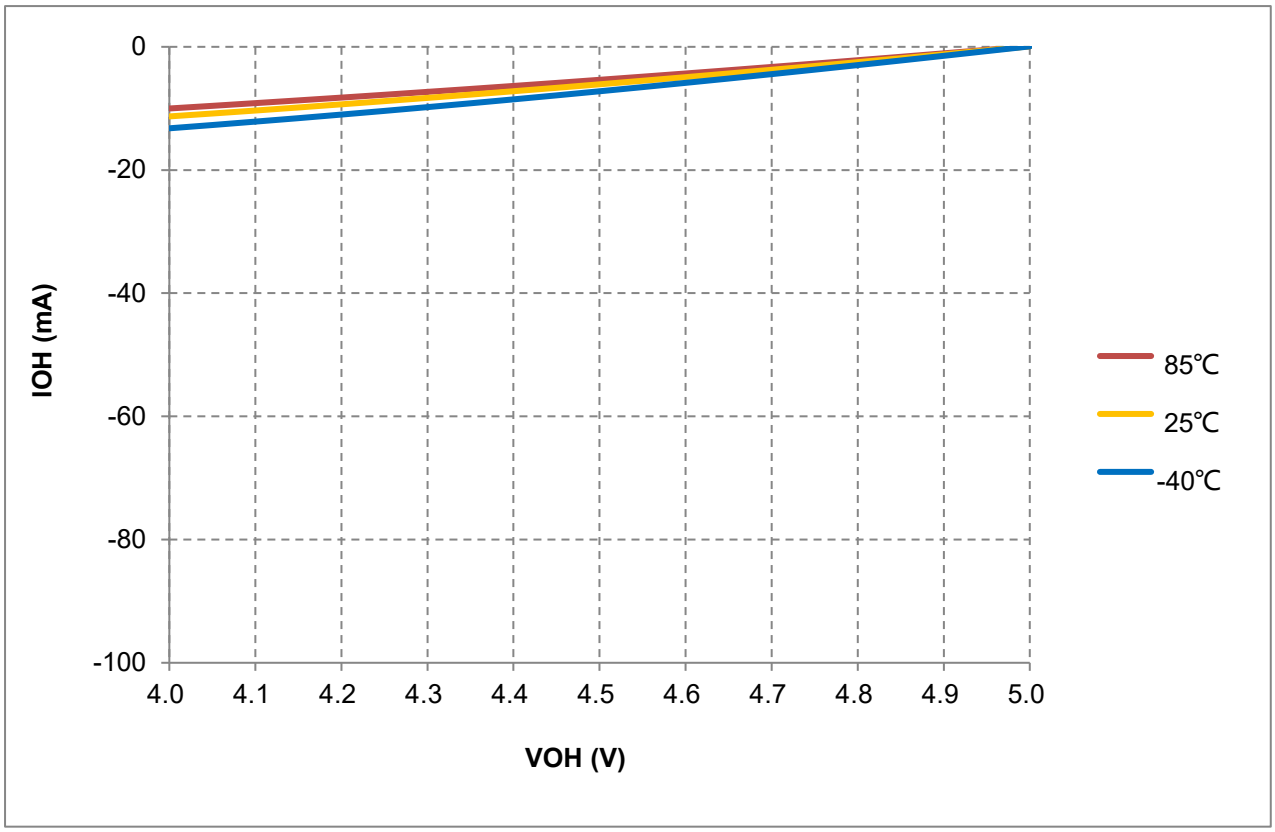


图 27-7 不同温度下，IOH (L1 = -6mA) vs. VOH @VDD=5V

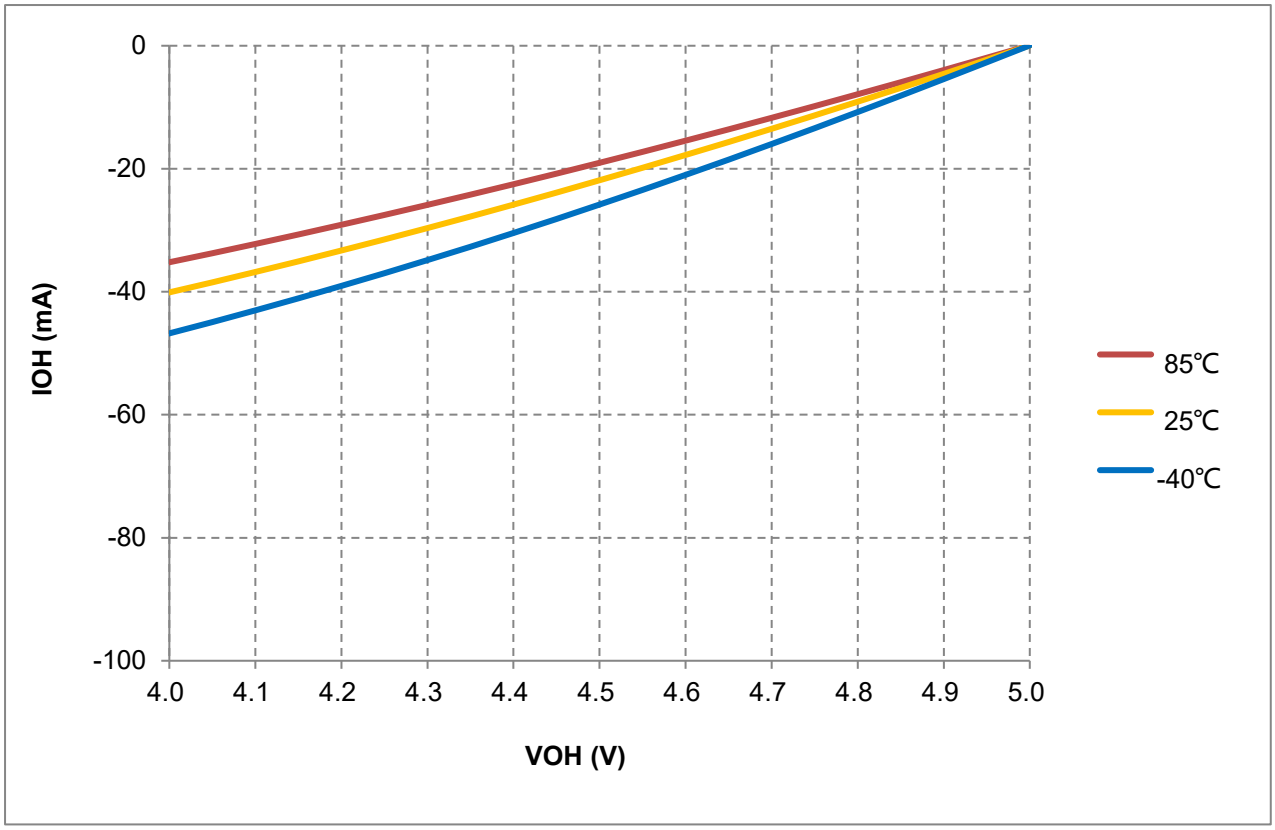


图 27-8 不同温度下，IOH (L2 = -18mA) vs. VOH @VDD=5V

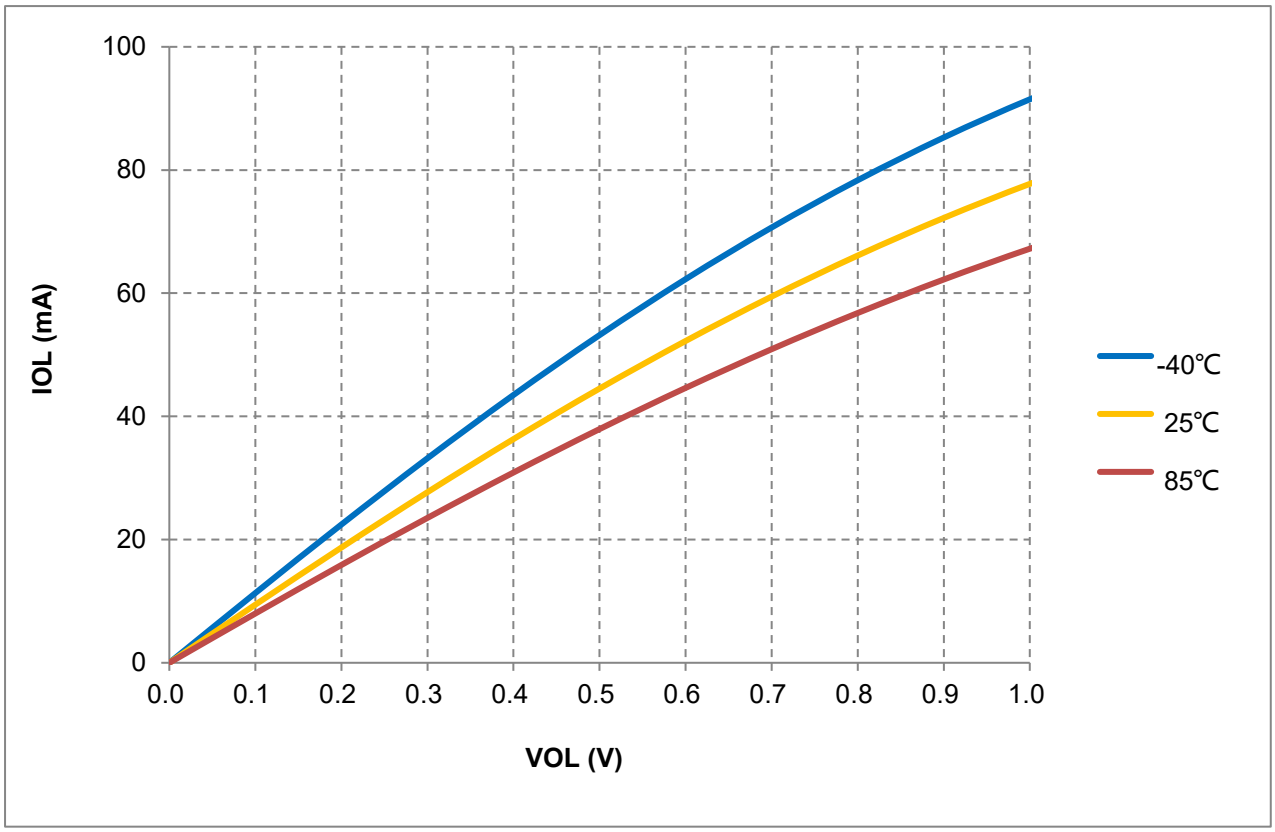


图 27-9 不同温度下，IOL (L0 = 46mA) vs. VOL @VDD=5V

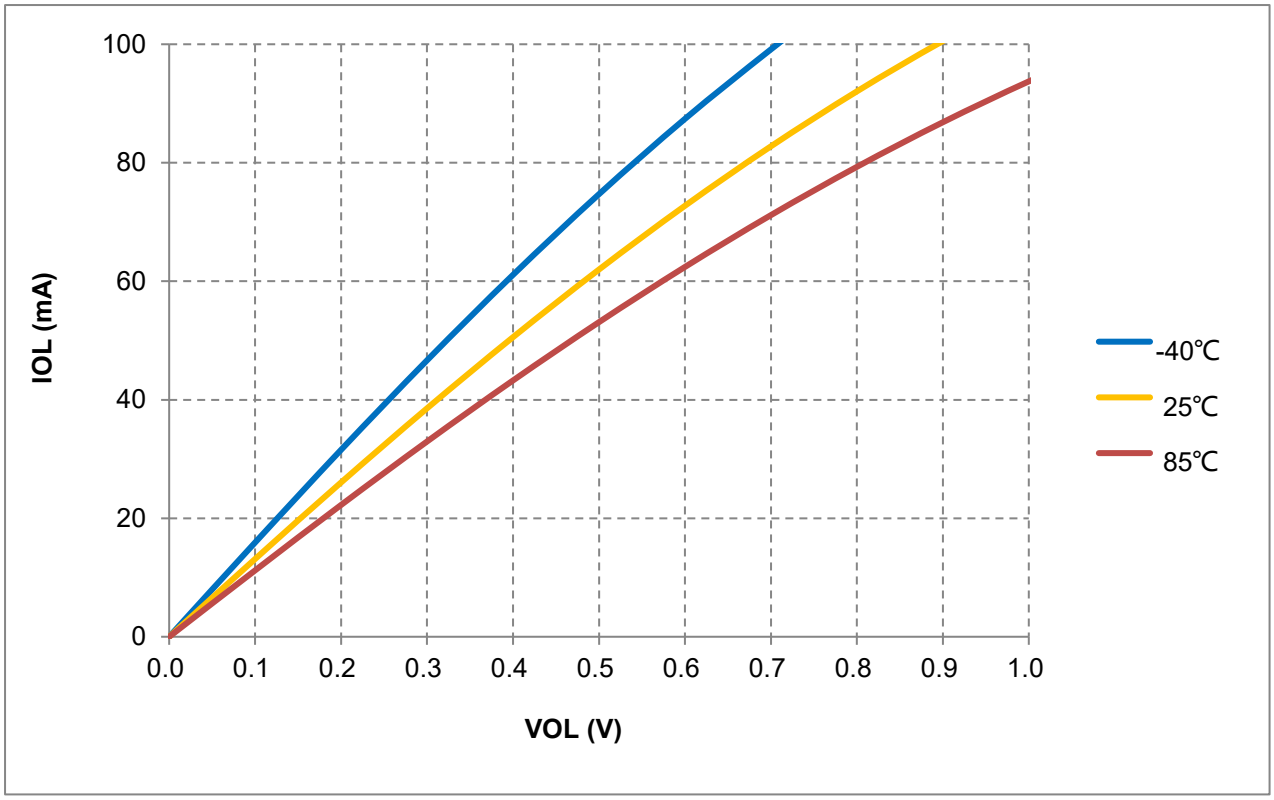
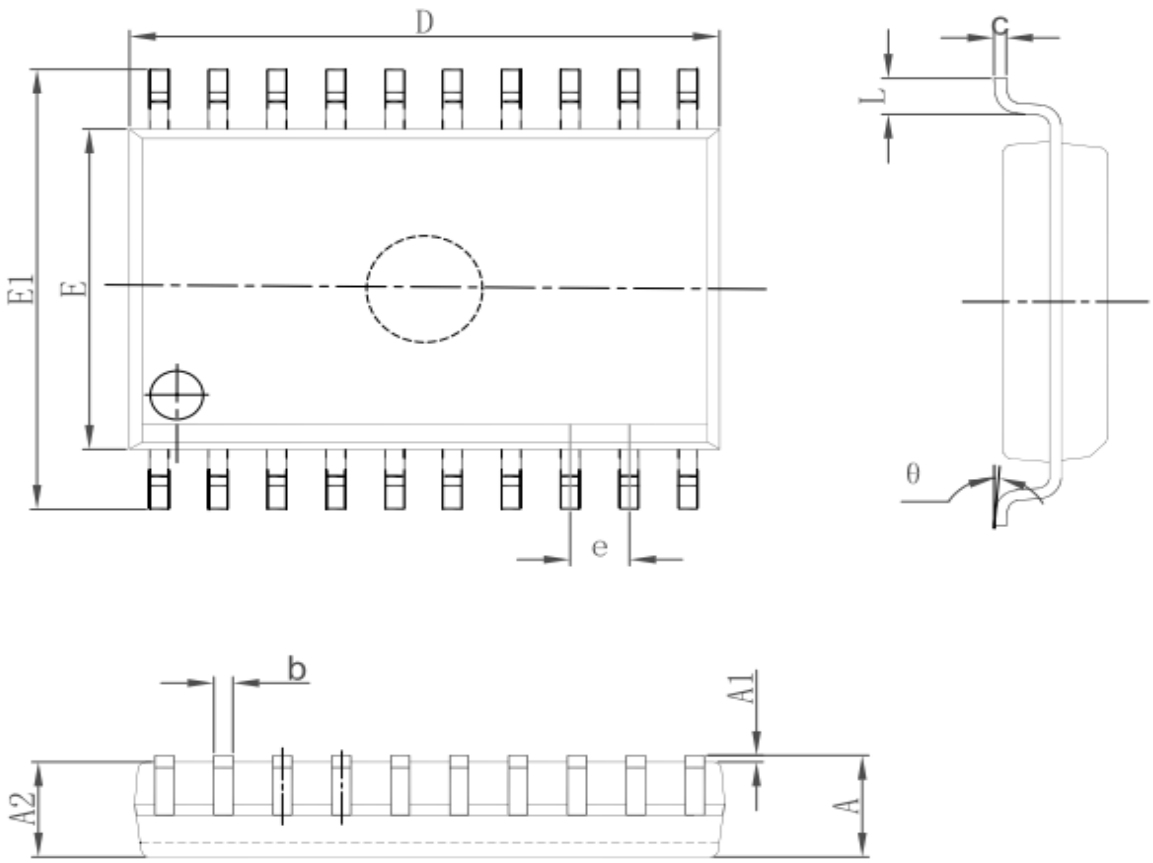


图 27-10 不同温度下，IOL (L1 = 60mA) vs. VOL @VDD=5V

28. 芯片封装信息

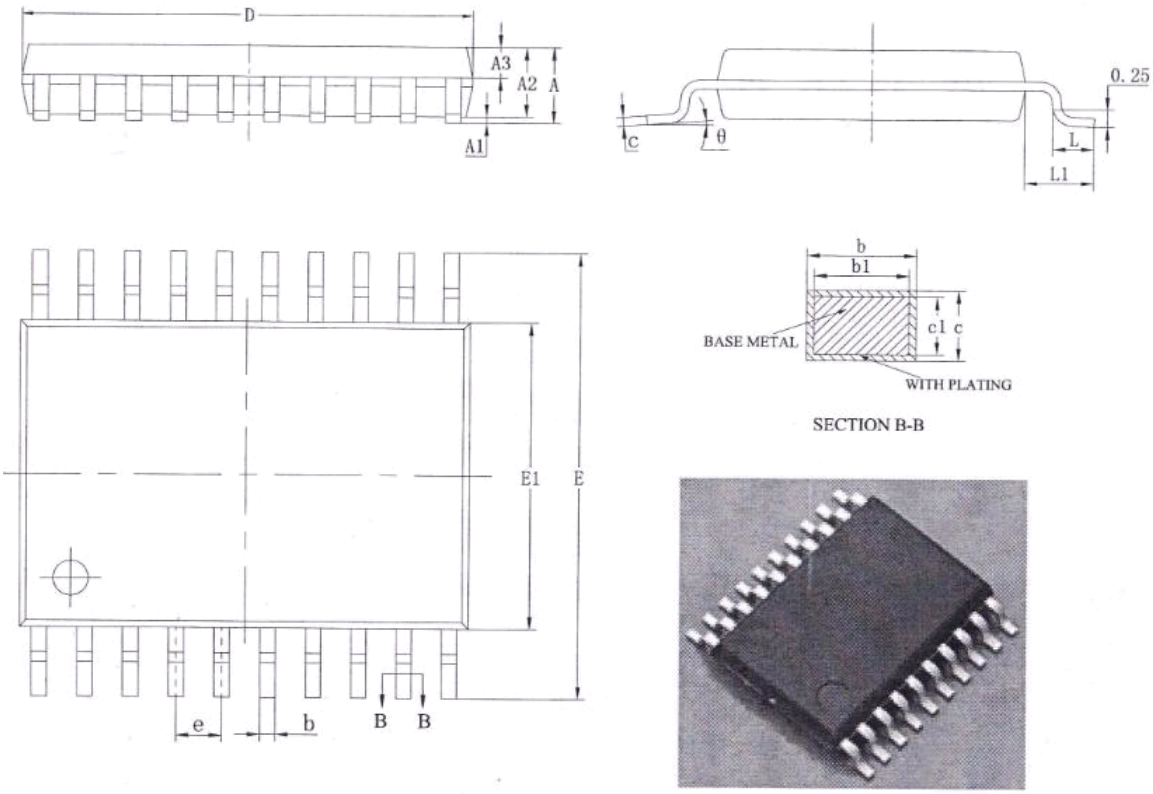
本芯片的封装形式有 SOP20，TSSOP20，TSSOP24，SOP28 和 QFN32 封装。具体封装尺寸信息如下：

SOP20



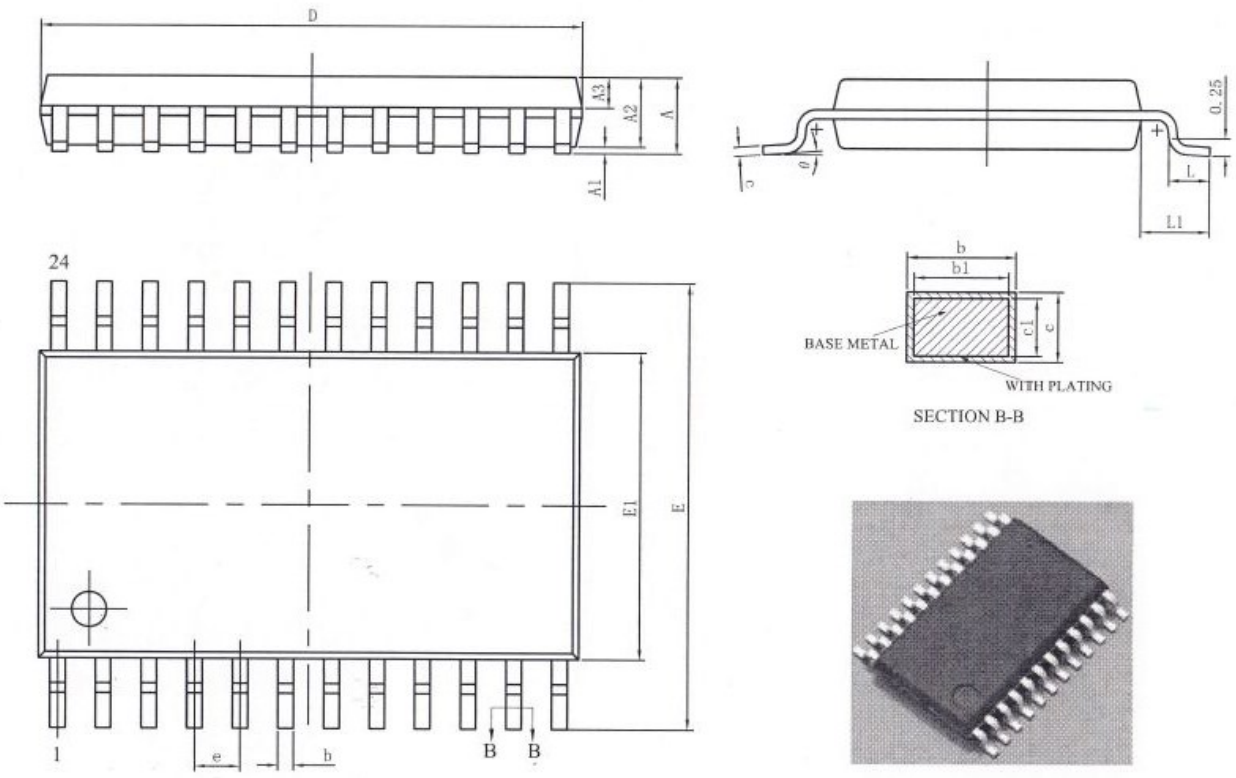
Symbol	Dimensions In Millimeters		Dimensions In Inches	
	Min	Max	Min	Max
A	2.350	2.650	0.093	0.104
A1	0.100	0.300	0.004	0.012
A2	2.100	2.500	0.083	0.098
b	0.330	0.510	0.013	0.020
c	0.204	0.330	0.008	0.013
D	12.520	13.000	0.493	0.512
E	7.400	7.600	0.291	0.299
E1	10.210	10.610	0.402	0.418
e	1.270 (BSC)		0.050 (BSC)	
L	0.400	1.270	0.016	0.050
θ	0°	8°	0°	8°

TSSOP20



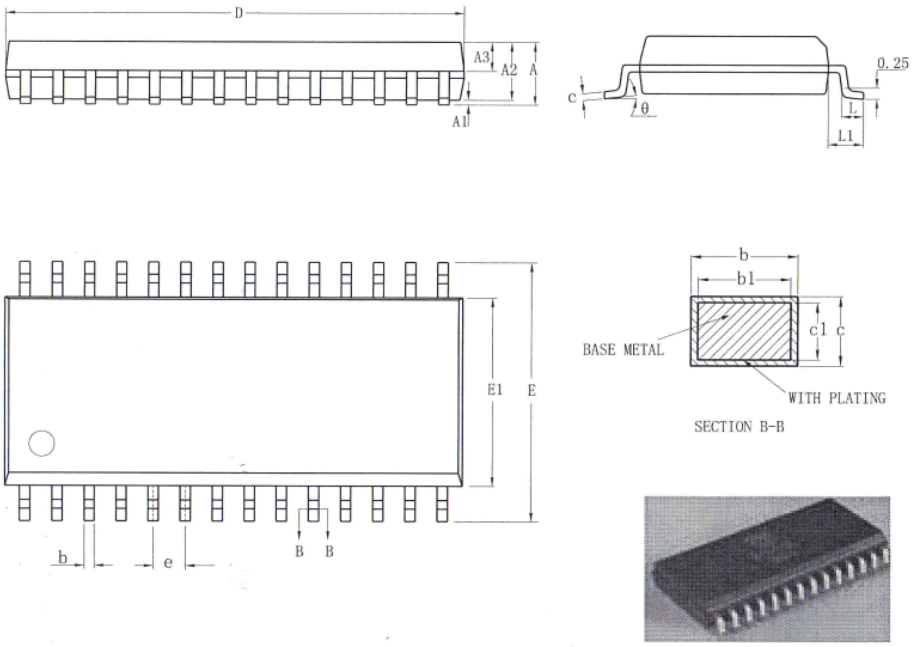
Symbol	Dimensions In Millimeters		Dimensions In Inches	
	Min	Max	Min	Max
A	-	1.20	-	0.472
A1	0.05	0.15	0.020	0.059
A2	0.80	1.05	0.315	0.413
A3	0.39	0.49	0.154	0.193
b	0.20	0.28	0.079	0.110
b1	0.19	0.25	0.075	0.098
c	0.13	0.17	0.051	0.067
c1	0.12	0.14	0.047	0.055
D	6.40	6.60	2.520	2.598
E1	4.30	4.50	1.693	1.771
E	6.20	6.60	2.441	2.598
e	0.65(BSC)		0.256(BSC)	
L	0.45	0.75	0.177	0.295
L1	1.00REF		0.394REF	
θ	0	8°	0	8°

TSSOP24



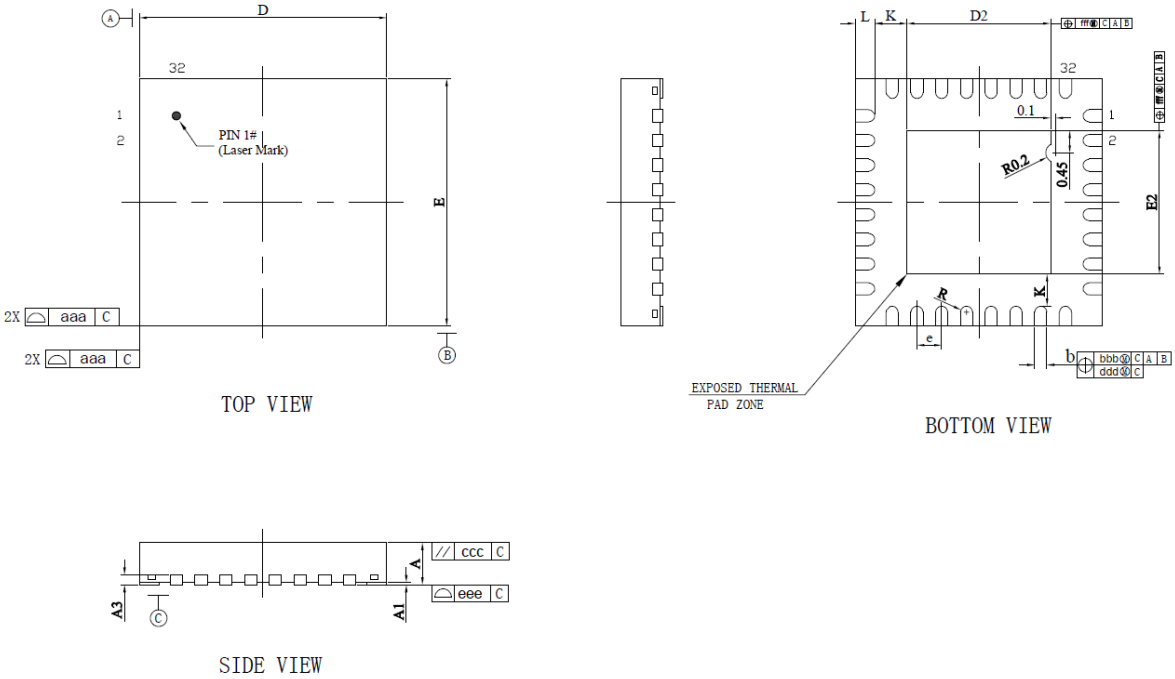
Symbol	Dimensions In Millimeters		
	Min	Nom	Max
A	—	—	1.20
A1	0.05	—	0.15
A2	0.80	1.00	1.05
A3	0.39	0.44	0.49
b	0.20	—	0.29
b1	0.19	0.22	0.25
c	0.13	—	0.18
c1	0.12	0.13	0.14
D	7.70	7.80	7.90
E	6.20	6.40	6.60
E1	4.30	4.40	4.50
e	0.65BSC		
L	0.45	0.60	0.75
L1	1.00BSC		
θ	0	—	8°

SOP28



Symbol	Dimensions In Millimeters		Dimensions In Inches	
	Min	Max	Min	Max
A	-	2.65	-	0.104
A1	0.10	0.30	0.004	0.012
A2	2.25	2.35	0.089	0.093
A3	0.97	1.07	0.038	0.042
b	0.39	0.47	0.015	0.019
b1	0.38	0.44	0.015	0.017
c	0.25	0.29	0.010	0.011
c1	0.24	0.26	0.009	0.010
D	17.90	18.10	0.704	0.712
E	10.10	10.50	0.397	0.413
E1	7.40	7.60	0.290	0.299
e	1.27(BSC)		0.05(BSC)	
L	0.70	1.00	0.027	0.039
L1	1.40REF		0.055REF	
θ	0	8°	0	8°

QFN32



Symbol	Dimensions In Millimeters		Dimensions In Inches	
	Min	Max	Min	Max
A	0.80	0.90	0.031	0.035
A1	0	0.05	0.000	0.002
b	0.20	0.30	0.008	0.012
A3	0.203REF		0.008REF	
D	4.90	5.10	0.193	0.201
D2	2.80	3.00	0.110	0.118
e	0.50BSC		0.020BSC	
E	4.90	5.10	0.193	0.201
E2	2.80	3.00	0.110	0.118
L	0.35	0.45	0.014	0.018
R	0.075	0.175	0.003	0.007
K	0.65REF		0.026REF	
aaa	0.10		0.004	
bbb	0.07		0.003	
ccc	0.10		0.004	
ddd	0.05		0.002	
eee	0.08		0.003	
fff	0.10		0.004	

附录 1，寄存器类型

缩写	描述	说明
WO	Write Only, read “0”	只写，读为 0
RO	Read Only	只读
RW	Read, Write	可读，可写
RW0	Read, Write “0” only	可读，只能写 0，写 1 无效
RW1	Read, Write “1” only	可读，只能写 1，写 0 无效
R_W1C	Read, Cleared by Writing “1”	可读，写 1 清零，写 0 无效
Res	Reserved, read “0”	保留位，只读，读为 0

附录 2，文档更改历史

日期	版本	内容
2020-12-8	1.01	初版 删除多余章节
2020-12-21	1.02	增加选型表 更新 2.12 小节描述 删除 SOP28 脚位多余的 ISPDAT1, ISPCK2/ISPDAT2 添加 P0ANP/P0AP 到表格 21.6 的 MSCON2 更新 4.8 上电配置过程章节的描述 更新 4.11 小节中 LVDCON 寄存器的描述 更新 14.8.21 COMAF2 寄存器的描述，增加 INTFIXB 位 修改 P1Ax、P1B、P1C 和 P1D 在刹车状态下输出状态的定义 更新 2.3 小节 SFR，BANK2 的寄存器表格 更新 2.5 小节 SFR，BANK4 的寄存器表格 更新 2.6 小节 SFR，BANK5 的寄存器表格 文档中所有 HS 改为 XT 更新 ADC 模块寄存器说明 更正 ADC 时钟源描述，F_{OSC} 改为 F_{SYSCLK}，F_{RC} 改为 F_{LIRC} 删除 9.1.2 小节硬件固有采样延时说明 9.1.6 小节添加 E 版触发延时说明 更新 9.1.8 小节中 ADC 转换周期描述和 E 版的模数转换周期图 更新 9.1.11 小节中 ADC 阈值比较功能框图 更新 9.2.4 小节中休眠模式下的 ADC 启动转换延时说明 更新 13.6.1 刹车状态章节 更新 13.6.4 前沿消隐章节
2021-4-19	1.03	更新选型表 更新脚位图，将 LQFP32 改为 QFN32 更新指令集章节 增加芯片封装信息章节 更新电气特性章节 文档中的“E 版”改为“G 版”
2021-4-30	1.04	更新图 6.1 更新 6.4 小节中断过程中的现场保存
2021-10-22	2.00	全面优化寄存器表格，更新 MCU 产品订购信息 添加 SOP20，TSSOP20 脚位
2021-10-29	2.01	删除型号：FT62F287-RB

日期	版本	内容
2021-11-26	2.02	增加型号：FT62F286A-TRB

联系信息

Fremont Micro Devices Corporation

#5−8, 10/F, Changhong Building
Ke-Ji Nan 12 Road, Nanshan District,
Shenzhen, Guangdong, PRC 518057

Tel: (+86 755) 8611 7811
Fax: (+86 755) 8611 7810

Fremont Micro Devices (HK) Limited

#16, 16/F, Block B, Veristrong Industrial Centre,
34−36 Au Pui Wan Street, Fotan, Shatin, Hong Kong SAR

Tel: (+852) 2781 1186
Fax: (+852) 2781 1144

http://www.fremontmicro.com

* Information furnished is believed to be accurate and reliable. However, Fremont Micro Devices Corporation assumes no responsibility for the consequences of use of such information or for any infringement of patents of other rights of third parties, which may result from its use. No license is granted by implication or otherwise under any patent rights of Fremont Micro Devices Corporation. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. Fremont Micro Devices Corporation products are not authorized for use as critical components in life support devices or systems without express written approval of Fremont Micro Devices Corporation. The FMD logo is a registered trademark of Fremont Micro Devices Corporation. All other names are the property of their respective owners.