Fremont Micro Devices

FT61F14

数据手册

主要特性

8−bit 基于 EEPROM 的 RISC MCU

Program: 4k x 14; RAM: 512 x 8; Data: 128 x 8

8 / 10 / 14 / 16 / 20 引脚

12−bit 高精度 ADC ($V_{ADC-REF}$ = 0.5V 时为 11−bit 精度)

4 个定时器, 7 路独立 PWM − 3 路带死区控制

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)

- 49 条 RISC 指令: 2T or 4T
- 16 MHz / 2T (V_{DD} ≥ 2.7)
- 多达 20 个引脚

Memory

- PROGRAM: 4k x 14 bit (读/写保护)
- DATA: 128 x 8 bit
- RAM: 512 x 8 bit
- 16 层硬件堆栈
- 扇区加密，支持 IAP

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 0.4 μA
- WDT 3.1 μA
- 正常模式 (16 MHz) 286 μA/mips

高可靠性

- 100 万次擦写次数 (typical)
- > 20 年 / 85°C 存储 (typical)
- ESD > 7 kV, EFT > 5.5 kV

ADC (12−bit)

- 12−bit 精度 (≤ 2 MHz ADC 时钟)
- 7 + 1 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

PWM (Total 7 路)

- 支持在 SLEEP 下运行
- 7 个捕获/比较/PWM 通道：
  - ✓ 独立：占空比，极性
- 3 个通道 (多达 6 个 I/O)：
  - ✓ 互补输出+死区
- 自动故障刹车 (I/O, LVD, ADC)
- 边沿对齐，中心对齐
- 单脉冲模式

Timers

- WDT (16−bit): 3−bit 预分频
- Timer1 (16−bit): 16−bit 预分频
- Timer2 (16−bit): 4−bit 预分频
- Timer4 (8−bit): 3−bit 预分频
- 自动重载
- 支持在 SLEEP 下运行
- Sysclk, LIRC, 1 or 2x {HIRC，晶振, EC}

通信接口

- USART

I/O PORTS (多达 18 个 I/O)

- 上拉/下拉电阻
- 开漏
- 18 个 I/O 源电流: 4, 8 or 26mA (5V, 25°C)
- 18 个 I/O 灌电流: 53 or 62 mA (5V, 25°C)
- 18 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: 2.0, 2.4, 2.8, 3.0, 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
  - ✓ 16MHz <±0.5% typical (2.5−5.5V, 25°C)
  - ✓ 可微调
  - ✓ 1, 2, 4, 8, 16, 32, 64, 128 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)

- ADC 最小分辨率为 0.12mV，精度为 0.24mV，适用于电流检测
- 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，跳跃等

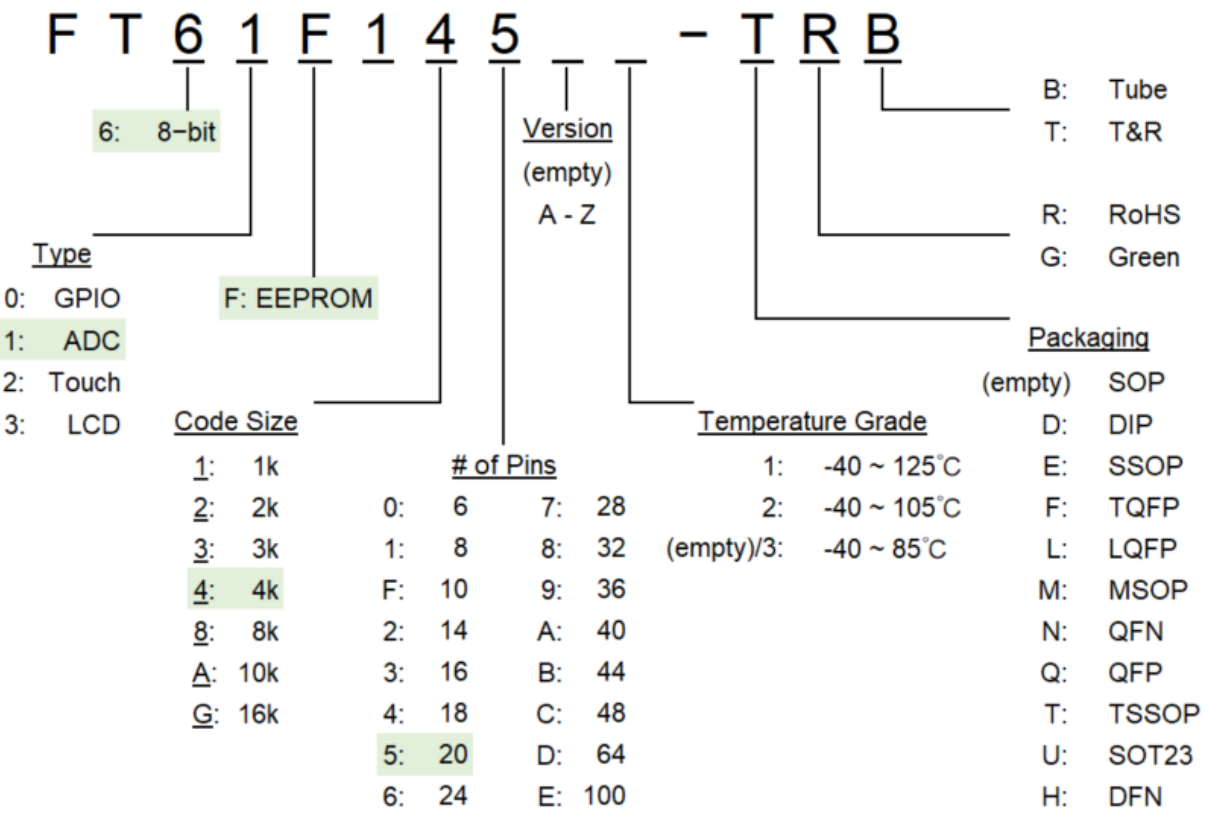
封装

- SOP8 MSOP10 SOP14 SOP16 SOP20 TSSOP20 QFN20

产品信息和选型表

型号	I/O 数	封装
FT61F141- ab	6	SOP8
FT61F14F- Mab	8	MSOP10
FT61F142- ab	12	SOP14
FT61F143- ab	14	SOP16
FT61F143A- ab		
FT61F143B- ab		
FT61F145- ab	18	SOP20
FT61F145A- ab		
FT61F145- Tab		TSSOP20
FT61F145A- Tab		
FT61F145- Nab		QFN20
FT61F145A- Nab		

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



MCU 产品订购信息

文档修改历史

日期	版本	描述
2020-12-07	1.09	初版
2021-10-22	2.00	全面优化寄存器表格，更新 MCU 产品订购信息
2022-08-25	2.01	全面优化版本 (请忽略旧版)
2023-05-05	2.02	更新章节 10 和章节 11 的示例程序；更新 SOP16 封装信息

目录

1. 结构框图和引脚 8
1.1. 引脚图 9
1.2. 引脚描述---按功能分类 12
2. I/O 端口 15
2.1. IO 端口相关寄存器汇总 18
2.2. I/O 配置 21
2.3. 引脚输出优先级 24
3. 上电复位 (POR) 25
3.1. 初始化配置时序 25
4. 系统复位 27
4.1. 系统复位相关寄存器汇总 28
4.2. 欠压复位 (Brown-Out Reset, LVR / BOR) 28
4.3. 非法指令复位 (IIlegal Instruction Reset) 29
4.4. 软件复位 29
4.5. 堆栈溢出复位 29
4.6. EMC 复位 29
4.7. 看门狗定时器 (Watch Dog Timer, WDT) 复位 29
4.8. 外部 I/O 系统复位 /MCLRB 30
4.9. 检测上次复位类型 30
5. 低电压检测/比较器 (LVD) 32
5.1. LVD 相关寄存器汇总 32
6. 振荡器和系统时钟 33
6.1. 振荡器模块相关寄存器汇总 34
6.2. 内部时钟模式 (HIRC 和 LIRC) 36
6.3. 外部时钟模式 (EC / LP / XT) 38
6.3.1. EC 模式 38
6.3.2. LP 和 XT 模式 38
6.4. HIRC, LIRC 和 EC 时钟的内部切换 39
7. 定时器 (TIMERS) 40

7.1. 看门狗定时器 (Watch Dog Timer, WDT) 41
7.1.1. WDT 相关寄存器汇总 41
7.1.2. WDT 的设置和使用 42
7.2. 高级定时器 1 (TIMER1) 43
7.2.1. Timer1 相关寄存器汇总 44
7.2.2. 计数基本单元 58
7.2.3. 时钟/触发控制器 62
7.2.4. 捕获/比较通道 70
7.3. 通用定时器 2 (TIMER2) 82
7.3.1. Timer2 相关寄存器汇总 83
7.3.2. 计数基本单元 90
7.3.3. 捕获/比较通道 92
7.4. 基本定时器 4 (TIMER4) 95
7.4.1. Timer4 相关寄存器汇总 96
8. SLEEP 睡眠模式 (POWER-DOWN) 98
8.1. 进入 SLEEP 98
8.2. 从 SLEEP 中唤醒 99
9. 中断 (INTERRUPTS) 100
9.1. 中断相关寄存器汇总 102
9.2. 外部管脚中断 108
10. 数据 EEPROM 和程序 PROM 109
10.1. DATA EEPROM 和 PROM 相关寄存器汇总 109
10.2. DATA EEPROM 111
10.2.1. 写 DATA EEPROM 111
10.2.2. 读 DATA EEPROM 112
10.3 程序 PROM 113
10.3.1 擦除程序 PROM 113
10.3.2 写程序 PROM 114
10.3.3 读程序 PROM 116
10.4 读初始化配置寄存器 UCFGx 116

11 12-bit 模/数转换器 (ANALOG TO DIGITAL CONVERTER, ADC) 117
11.1 ADC 相关寄存器汇总 118
11.2 ADC 配置 122
11.2.1 ADC 触发和延时配置 123
11.2.2 ADC 中止转换 124
11.2.3 阈值比较 124
11.2.4 中断 125
11.3 ADC 采样保持时间 125
11.4 ADC 最短采样时间 126
11.5 ADC 转换步骤示例 126
12 USART 接口 128
12.1 USART 接口相关寄存器汇总 130
12.2 USART 功能 135
12.2.1 异步工作模式 135
12.2.2 同步工作模式 137
12.2.3 红外工作模式 137
12.2.4 智能卡模式 138
12.2.5 LIN Master 模式 138
12.2.6 多处理器通信模式 139
12.2.7 自动波特率检测 139
13 存储区读/写保护 141
14 特殊功能寄存器 (SPECIAL FUNCTION REGISTERS, SFR) 142
14.1 初始化配置寄存器 142
14.2 用户寄存器 144
14.3 STATUS 寄存器 156
14.4 堆栈 157
14.5 间接寻址 157
14.5.1 传统数据存储器 158
14.5.2 线性数据存储器 158
14.5.3 闪存程序存储器 159

15 指令集 (INSTRUCTION SET) 161
15.1 读-修改-写 (RMW)指令 163
15.2 指令详细描述 164
16 电气特性 173
16.1 极限参数 173
16.2 工作特性 173
16.3 POR, LVR, LVD 174
16.4 I/O 端口电路 175
16.5 工作电流 (I_{DD}) 175
16.6 内部振荡器 176
16.7 ADC (12bit) 和 ADC VREF 176
16.8 Program 和 Data EEPROM 178
16.9 EMC 特性 178
17 特性图 179
18 封装信息 184
19 附录：寄存器类型 191
联系信息 192

1. 结构框图和引脚

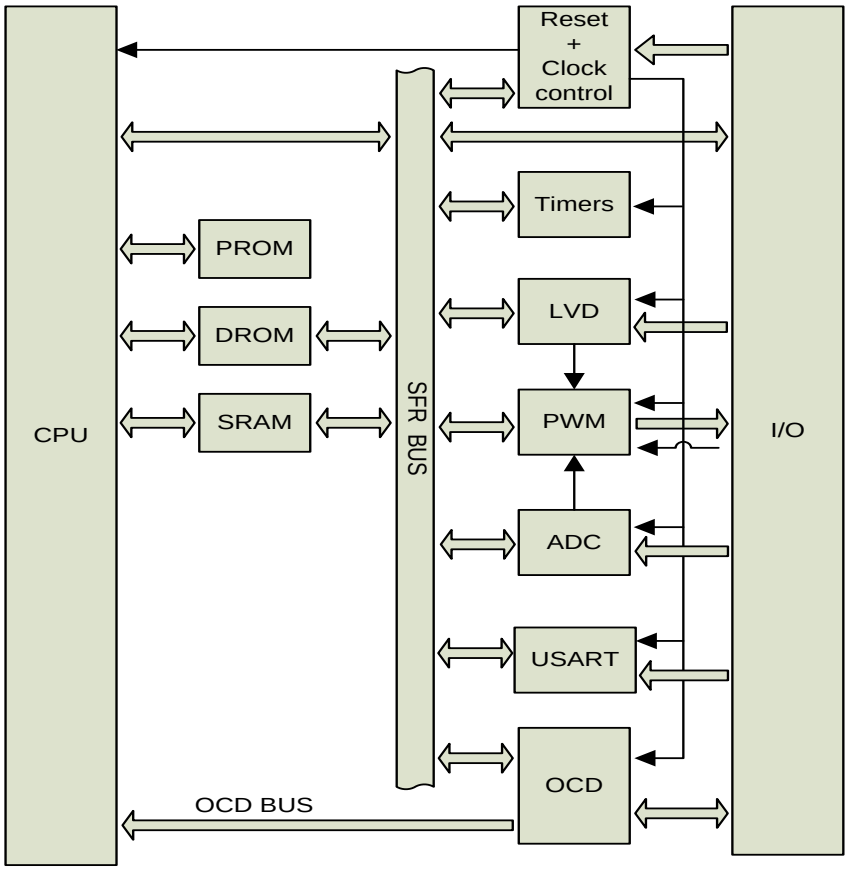


图 1-1 系统结构框图

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, Timer1, Timer2, Timer4
PWM	Pulse Width Modulator
ADC	Analog to Digital Converter
LVD	Low Voltage Detect / comparator
USART	Universal Synchronous Asynchronous Receiver Transmitter
OCD	On Chip Debug
I/O	Input / Output

1.1. 引脚图

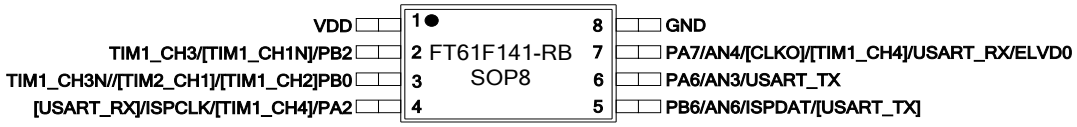


图 1-2 SOP8

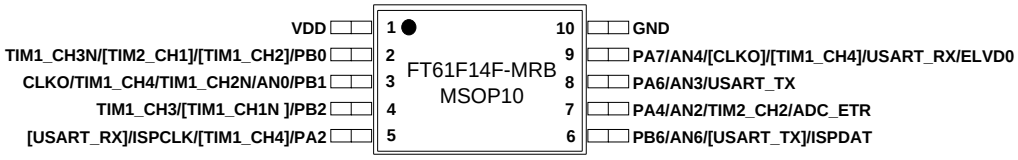


图 1-3 MSOP10

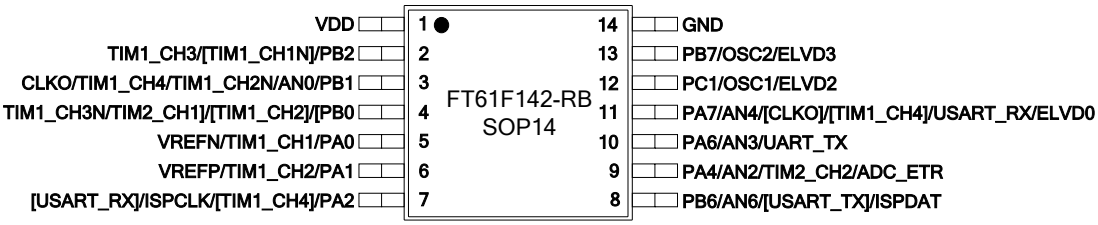


图 1-4 SOP14

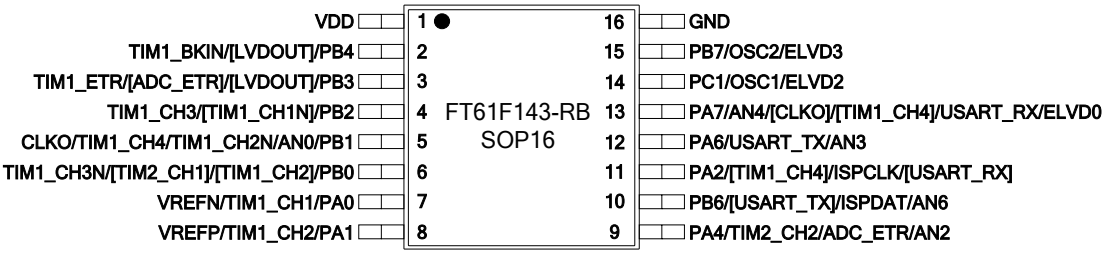


图 1-5 SOP16 (A)

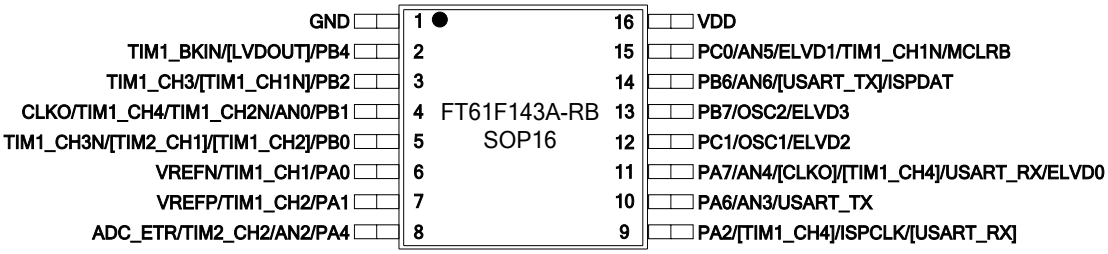


图 1-6 SOP16 (B)

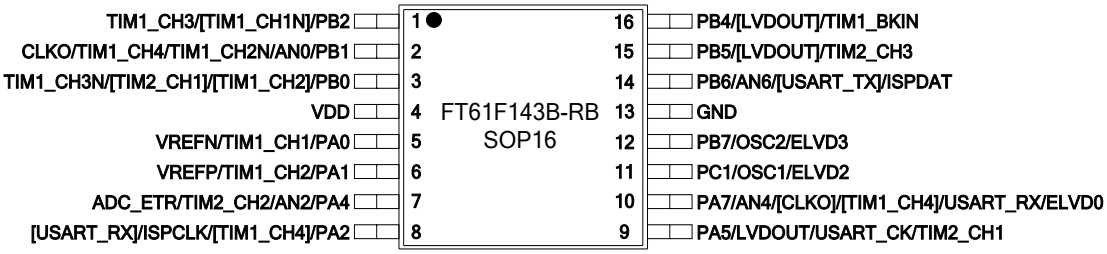


图 1-7 SOP16 (C)

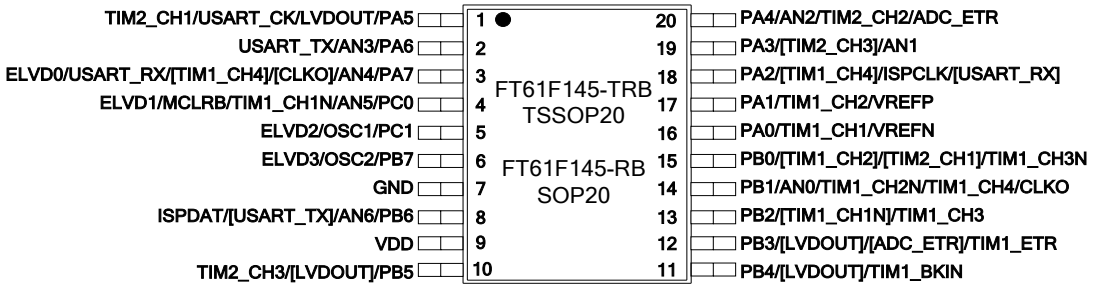


图 1-8 TSSOP20 / SOP20 (A)

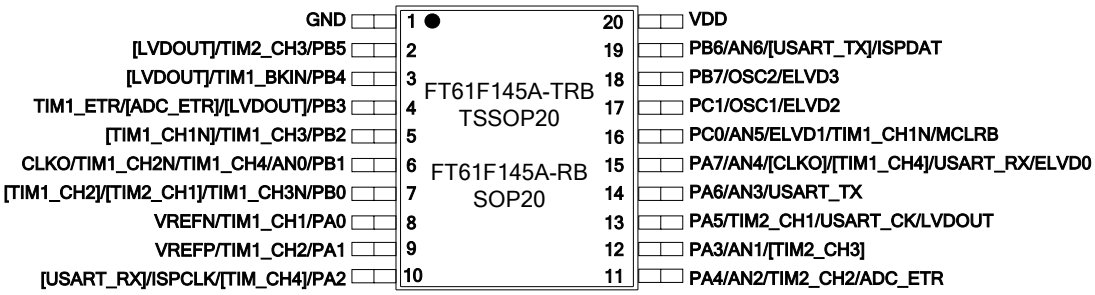


图 1-9 TSSOP20 / SOP20 (B)

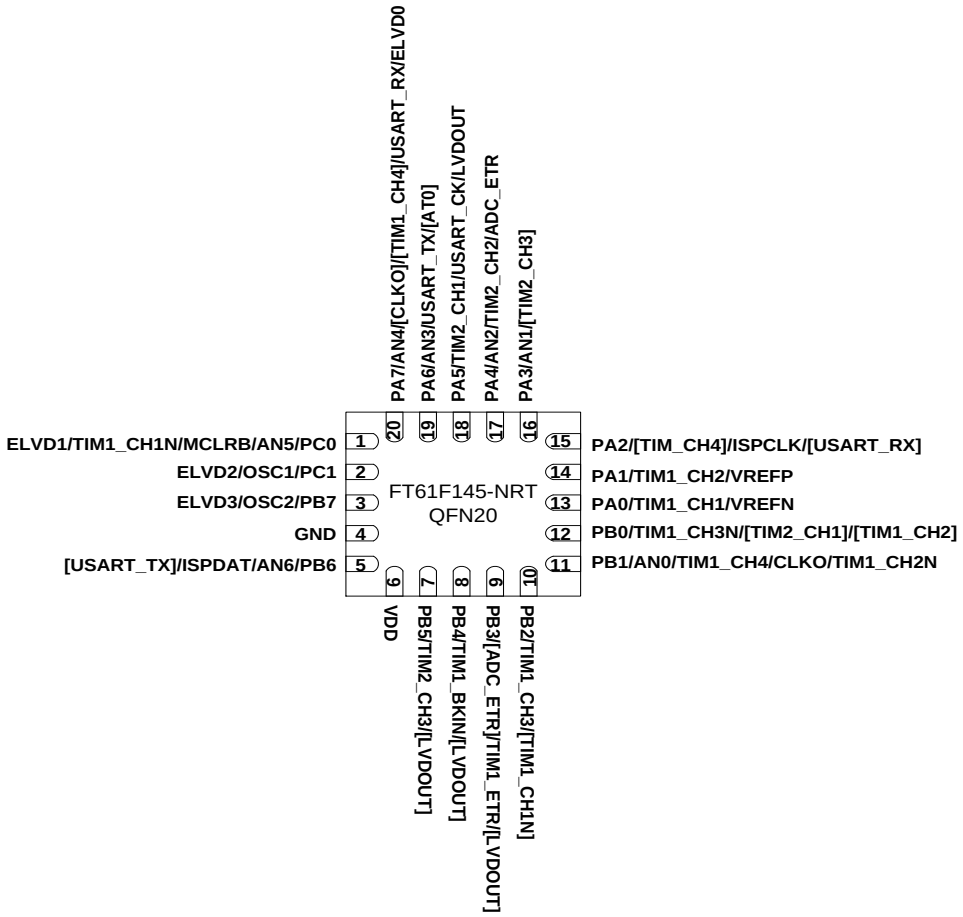


图 1-10 QFN20 (A) [1]

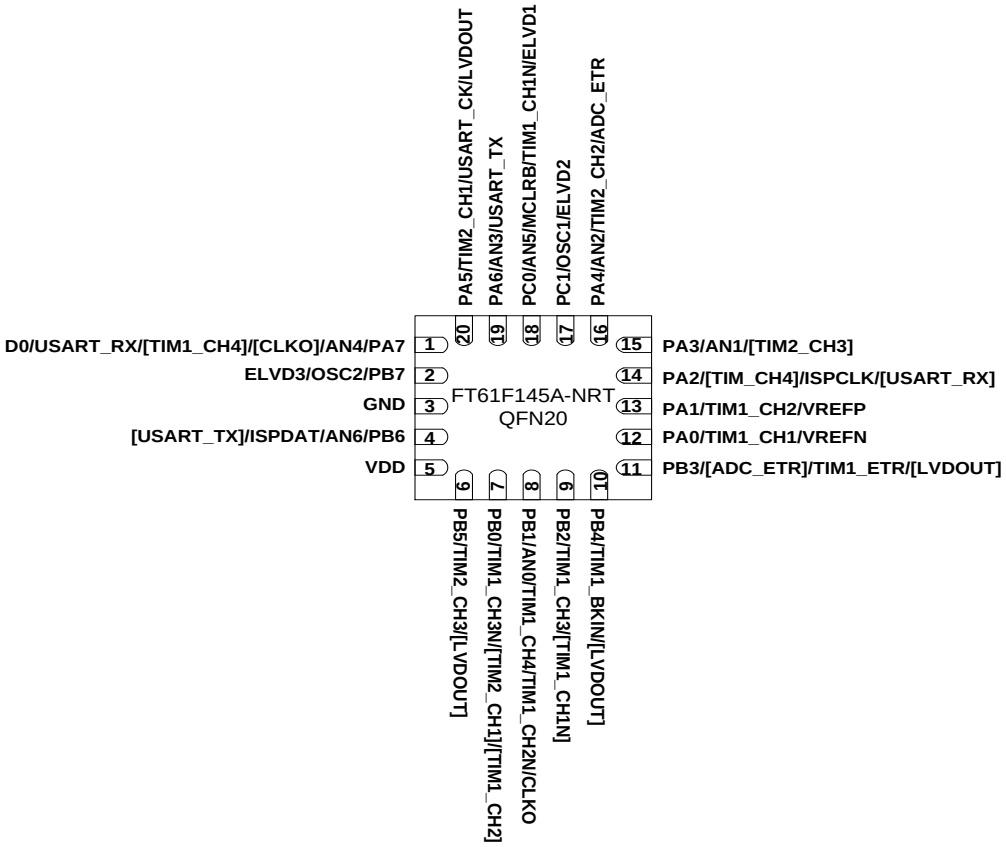


图 1-11 QFN20 (B) [1]

[1] 对于 QFN20 封装，底部裸露的焊垫与 GND 连在一起。

1.2. 引脚描述---按功能分类

功能	描述	引脚名	对应 GPIO	8 pins	10 pins	14 pins	16(A) pins	16(B) pins	16(C) pins	20(A) pins	20(B) pins	QFN20 (A) pins	QFN20 (B) pins
电源		VDD		1	1	1	1	16	4	9	20	6	5
		GND		8	10	14	16	1	13	7	1	4	3
GPIO	上拉/下拉, 数字输入, 数字输出	PC1				12	14	12	11	5	17	2	17
		PC0						15		4	16	1	18
		PB7				13	15	13	12	6	18	3	2
		PB6		5	6	8	10	14	14	8	19	5	4
		PB5							15	10	2	7	6
		PB4					2	2	16	11	3	8	10
		PB3					3			12	4	9	11
		PB2		2	4	2	4	3	1	13	5	10	9
		PB1			3	3	5	4	2	14	6	11	8
		PB0		3	2	4	6	5	3	15	7	12	7
		PA7		7	9	11	13	11	10	3	15	20	1
		PA6		6	8	10	12	10		2	14	19	19
		PA5							9	1	13	18	20
		PA4			7	9	9	8	7	20	11	17	16
		PA3								19	12	16	15
		PA2		4	5	7	11	9	8	18	10	15	14
		PA1				6	8	7	6	17	9	14	13
		PA0				5	7	6	5	16	8	13	12
ISP 调试	ISP-Data	ISPDAT	PB6	5	6	8	10	14	14	8	19	5	4
	ISP-CLK	ISPCLK	PA2	4	5	7	11	9	8	18	10	15	14
外部复位	上拉	/MCLRB	PC0					15		4	16	1	18
LVD	输入	ELVD0	PA7	7	9	11	13	11	10	3	15	20	1
		ELVD1	PC0					15		4	16	1	18
		ELVD2	PC1			12	14	12	11	5	17	2	17
		ELVD3	PB7			13	15	13	12	6	18	3	2
	输出	LVDOUT	PA5						9	1	13	18	20
		[LVDOUT]	PB5						15	10	2	7	6
		[LVDOUT]	PB4				2	2	16	11	3	8	10
		[LVDOUT]	PB3				3			12	4	9	11

功能	描述	引脚名	对应 GPIO	8 pins	10 pins	14 pins	16(A) pins	16(B) pins	16(C) pins	20(A) pins	20(B) pins	QFN20 (A) pins	QFN20 (B) pins
时钟	输出	CLKO	PB1		3	3	5	4	2	14	6	11	8
		[CLKO]	PA7	7	9	11	13	11	10	3	15	20	1
	OSC +	OSC1	PC1			12	14	12	11	5	17	2	17
	OSC −	OSC2	PB7			13	15	13	12	6	18	3	2
Timer1 (死区)	PWM1	TIM1_CH1	PA0			5	7	6	5	16	8	13	12
	/PWM1	TIM1_CH1N	PC0					15		4	16	1	18
		[TIM1_CH1N]	PB2	2	4	2	4	3	1	13	5	10	9
	PWM2	TIM1_CH2	PA1			6	8	7	6	17	9	14	13
		[TIM1_CH2]	PB0	3	2	4	6	5	3	15	7	12	7
	/PWM2	TIM1_CH2N	PB1		3	3	5	4	2	14	6	11	8
	PWM3	TIM1_CH3	PB2	2	4	2	4	3	1	13	5	10	9
	/PWM3	TIM1_CH3N	PB0	3	2	4	6	5	3	15	7	12	7
	PWM4	TIM1_CH4	PB1		3	3	5	4	2	14	6	11	8
		[TIM1_CH4]	PA7	7	9	11	13	11	10	3	15	20	1
	PWM 故障刹车输入	TIM1_BKIN	PB4				2	2	16	11	3	8	10
	触发	TIM1_ETR	PB3				3			12	4	9	11
Timer2	PWM5	TIM2_CH1	PA5						9	1	13	18	20
		[TIM2_CH1]	PB0	3	2	4	6	5	3	15	7	12	7
	PWM6	TIM2_CH2	PA4		7	9	9	8	7	20	11	17	16
	PWM7	TIM2_CH3	PB5						15	10	2	7	6
		[TIM2_CH3]	PA3							19	12	16	15
ADC	输入	AN6	PB6	5	6	8	10	14	14	8	19	5	4
		AN5	PC0					15		4	16	1	18
		AN4	PA7	7	9	11	13	11	10	3	15	20	1
		AN3	PA6	6	8	10	12	10		2	14	19	19
		AN2	PA4		7	9	9	8	7	20	11	17	16
		AN1	PA3							19	12	16	15
		AN0	PB1		3	3	5	4	2	14	6	11	8
	触发	ADC_ETR	PA4		7	9	9	8	7	20	11	17	16
		[ADC_ETR]	PB3				3			12	4	9	11
	V_{REF}−	VREFN	PA0			5	7	6	5	16	8	13	12
	V_{REF}+	VREFP	PA1			6	8	7	6	17	9	14	13

功能	描述	引脚名	对应 GPIO	8 pins	10 pins	14 pins	16(A) pins	16(B) pins	16(C) pins	20(A) pins	20(B) pins	QFN20 (A) pins	QFN20 (B) pins
外部管脚中断	低电平,上升沿,下降沿,双边沿	PC1				12	14	12	11	5	17	2	17
		PC0						15		4	16	1	18
		PB7				13	15	13	12	6	18	3	2
		PB6		5	6	8	10	14	14	8	19	5	4
		PB5							15	10	2	7	6
		PB4					2	2	16	11	3	8	10
		PB3					3			12	4	9	11
		PB2		2	4	2	4	3	1	13	5	10	9
		PB1			3	3	5	4	2	14	6	11	8
		PB0		3	2	4	6	5	3	15	7	12	7
		PA7		7	9	11	13	11	10	3	15	20	1
		PA6		6	8	10	12	10		2	14	19	19
		PA5							9	1	13	18	20
		PA4			7	9	9	8	7	20	11	17	16
		PA3								19	12	16	15
		PA2		4	5	7	11	9	8	18	10	15	14
		PA1				6	8	7	6	17	9	14	13
		PA0				5	7	6	5	16	8	13	12
USART	USART_CK	USART_CK	PA5						9	1	13	18	20
	USART_TX (开漏)	USART_TX	PA6	6	8	10	12	10		2	14	19	19
		[USART_TX]	PB6	5	6	8	10	14	14	8	19	5	4
	USART_RX	USART_RX	PA7	7	9	11	13	11	10	3	15	20	1
		[USART_RX]	PA2	4	5	7	11	9	8	18	10	15	14

表 1-1 按功能分类的引脚描述

2. I/O 端口

根据不同的封装类型，FT61F14x 系列芯片最多有 18 个 I/O 引脚。共分为 3 组：PORTA(8)、PORTB(8)和 PORTC(2)。**表 2-1** 列出了所有 I/O 引脚的功能。

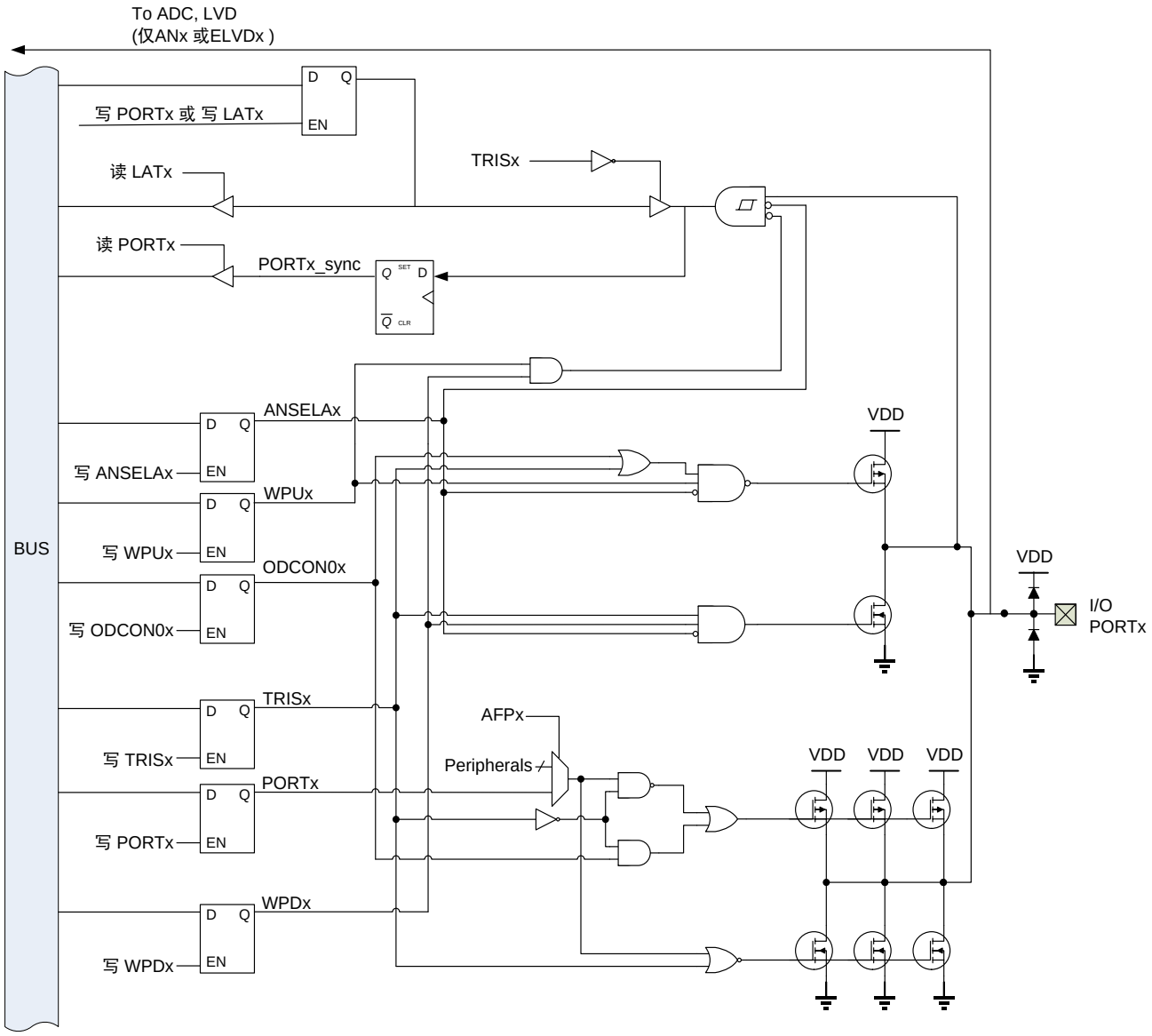


图 2-1 PORT 端口结构框图

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

- 数字输出
- 数字输入
- 开漏（USART 相应端口）
- 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，不需要设置。
2. 通过 IDE 界面选择，且在芯片初始化配置时加载的功能（**表 2-7**）：
   - 外部时钟/晶振输入（OSC1, OSC2）
   - 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 4 类：
   a) 数字输出
      - PWM
      - 内部时钟输出
      - LVD 输出
   b) 数字输入
      - PWM 故障刹车
      - Timer1 触发(TIM1_ETR)
      - GPIO 端口变化中断
      - ADC 触发（ADC_ETR）
   c) 模拟输入
      - LVD / BOR
      - ADC
      - V_{REF}+
      - V_{REF}-
   d) 通信接口
      - USART

引脚名	ISP 调试	时钟	ADC	USART	中断	LVD	PWM	数字 I/O 上拉/下拉	开漏	源电流 (mA)	灌电流 (mA)
PA0			(V_{REF}−)		√		PWM1	√		4, 26	53, 62
PA1			(V_{REF}+)		√		PWM2	√		4, 26	53, 62
PA2	CLK			[RX]	√			√		4, 26	53, 62
PA3			AN1		√		[PWM7]	√		4, 26	53, 62
PA4			AN2/ Trigger		√		PWM6	√		4, 26	53, 62
PA5				CLK	√	LVDOUT	PWM5	√		4, 26	53, 62
PA6			AN3	TX	√			√	√	4, 26	53, 62
PA7		输出	AN4	RX	√	ELVD0	[PWM4]	√		4, 26	53, 62
PB0					√		[PWM2]/ PWM3N/ [PWM5]	√		8,26	53, 62
PB1		输出	AN0		√		PWM4/ PWM2N	√		8,26	53, 62
PB2					√		PWM3/ [PWM1N]	√		8,26	53, 62
PB3			Trigger		√	[LVDOUT]	TIM1_ETR	√		8,26	53, 62
PB4					√	[LVDOUT]	BKIN	√		8,26	53, 62
PB5					√	[LVDOUT]	PWM7	√		8,26	53, 62
PB6	DATA		AN6	[TX]	√			√	√	8,26	53, 62
PB7		OSC−			√	ELVD3		√		8,26	53, 62
PC0			AN5		√	ELVD1	PWM1N	√		4, 8, 26	53, 62
PC1		OSC+			√	ELVD2		√		4, 8, 26	53, 62
注					/MCLRB = PC0					V_{DD}=5, V_{DS}=0.5	

表 2-1 I/O 端口功能

注： PORTA 和 PORTB 支持 2 档可配置源电流驱动能力(参阅 “PSRCAx” 和 “PSRCBx”，**表 2-3**)，PORTC 支持 3 档可配置源电流驱动能力(参阅 “PSRCCx”，**表 2-3**)，所有 IO 支持 2 档可配置灌电流驱动能力 (参阅 “PSINKx”，**表 2-3**)。

2.1. IO 端口相关寄存器汇总

名称	地址	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
ANSELA	0x197	–	ANSELA[6:0]							-000 0000
TRISA	0x8C	TRISA[7:0], PORTA 方向控制								1111 1111
TRISB	0x8D	TRISB[7:0], PORTB 方向控制								1111 1111
TRISC	0x8E	–	–	–	–	–	–	PORTC 方向控制		---- --11
PORTA	0x0C	PORTA 输出寄存器								xxxx xxxx
PORTB	0x0D	PORTB 输出寄存器								xxxx xxxx
PORTC	0x0E	–	–	–	–	–	–	PORTC 输出寄存器		---- --xx
LATA	0x10C	PORTA 数据锁存器								xxxx xxxx
LATB	0x10D	PORTB 数据锁存器								xxxx xxxx
LATC	0x10E	–	–	–	–	–	–	PORTC 数据锁存器		---- --xx
WPUA	0x18C	PORTA 弱上拉								0000 0000
WPUB	0x18D	PORTB 弱上拉								0000 0000
WPUC	0x18E	–	–	–	–	–	–	PORTC 弱上拉		---- --00
WPDA	0x20C	PORTA 弱下拉								0000 0000
WPDB	0x20D	PORTB 弱下拉								0000 0000
WPDC	0x20E	–	–	–	–	–	–	PORTC 弱下拉		---- --00
ODCON0	0x21F	–	–	–	–	–	–	–	UROD	---- ---0
PSRC0	0x11A	PORTA 源电流设置								1111 1111
PSRC1	0x11B	PORTB 源电流设置								1111 1111
PSRC2	0x11C	–	–	–	–	PORTC 源电流设置				---- 1111
PSINK0	0x19A	PORTA 灌电流设置								0000 0000
PSINK1	0x19B	PORTB 灌电流设置								0000 0000
PSINK2	0x19C	–	–	–	–	–	–	PORTC 灌电流设置		---- --00
ITYPE0	0x11E	PORTx[1:0] (x = A, B, C)和 PORTy[3:2] (y = A, B)外部管脚中断类型设置								0000 0000
ITYPE1	0x11F	PORTy[7:4] (y = A, B) 外部管脚中断类型设置								0000 0000
AFP0	0x19E	–	–	管脚重映射寄存器 0					–	--00 000-
AFP1	0x19F	–	–	管脚重映射寄存器 1						--00 0000
EPS0	0x118	外部中断 EINT3~0 管脚选择								0000 0000
EPS1	0x119	外部中断 EINT7~4 管脚选择								0000 0000
EPIE0	0x94	外部管脚中断使能位								0000 0000
EPIF0	0x14	外部管脚中断标志位								0000 0000

表 2-2 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）	TRISA[7:0]	0x8C	RW-1111 1111
TRISB	PORTB	1 = 关闭	TRISB[7:0]	0x8D	RW-1111 1111
TRISC	PORTC	0 = 使能（关闭上拉/下拉）	TRISC[1:0]	0x8E	RW-11
ANSELA	1 = 关闭上拉/下拉及数字输入 (仅适用于 7 个 ADC 通道) 0 = (无动作)		ANSELA[6:0]	0x197	RW-000 0000
WPUA	PORTA	弱上拉	WPUA[7:0]	0x18C	RW-0000 0000
WPUB	PORTB	1 = 使能	WPUB[7:0]	0x18D	RW-0000 0000
WPUC	PORTC	0 = 关闭	WPUC[1:0]	0x18E	RW-00
WPDA	PORTA	弱下拉	WPDA[7:0]	0x20C	RW-0000 0000
WPDB	PORTB	1 = 使能	WPDB[7:0]	0x20D	RW-0000 0000
WPDC	PORTC	0 = 关闭	WPDC[1:0]	0x20E	RW-00
PORTA	PORTA	数据输出寄存器	PORTA[7:0]	0x0C	RW-xxxx xxxx
PORTB	PORTB	读：返回 IO 引脚上的电平	PORTB[7:0]	0x0D	RW-xxxx xxxx
PORTC	PORTC	写：写入相应的 LATx 寄存器	PORTC[1:0]	0x0E	RW-xx
LATA	PORTA	数据输出锁存器	LATA[7:0]	0x10C	RW-xxxx xxxx
LATB	PORTB		LATB[7:0]	0x10D	RW-xxxx xxxx
LATC	PORTC		LATC[1:0]	0x10E	RW-xx
UROD	USART_TX 开漏	1 = 使能 0 = 关闭	ODCON0[0]	0x21F	RW-0
CLKOS [1]	CLKO	1 = PB1 0 = PA7	CKAUX[1]	0xF8D	RW-1
AFP0 [1]	TIM1_CH2	1 = PB0 0 = PA1	AFP0[5]	0x19E	RW-0
	TIM2_CH1	1 = PB0 0 = PA5	AFP0[4]		RW-0
	TIM2_CH3	1 = PA3 0 = PB5	AFP0[3]		RW-0
	TIM1_CH1N	1 = PB2 0 = PC0	AFP0[2]		RW-0
	ADC_ETR	1 = PB3 0 = PA4	AFP0[1]		RW-0
AFP1 [1]	USART_TX	00 = PA6 10 = PA7 01 = PB6 11 = PA2	AFP1[5:4]	0x19F	RW-00
	USART_RX	00 = PA7 10 = PA6 01 = PA2 11 = PB6	AFP1[3:2]		
	TIM1_CH4	00 = PB1 10 = PA7 01 = PB1 11 = PA2	AFP1[1:0]		
PSINK0	PA7-PA0	灌电流（mA)	PSINK0[7:0]	0x19A	RW-0000 0000
PSINK1	PB7-PB0	1 = 62	PSINK1[7:0]	0x19B	RW-0000 0000
PSINK2	PC1-PC0	0 = 53	PSINK2[1:0]	0x19C	RW-00

[1] 管脚重映射选择。

名称	状态		寄存器	地址	复位值
PSRCA[7:0]	PA7−PA0	源电流 (mA) 1 = 26 0 = 4	PSRC0[7:0]	0x11A	RW−1111 1111
PSRCB[7:0]	PB7−PB0	源电流 (mA) 1 = 26 0 = 8	PSRC1[7:0]	0x11B	RW−1111 1111
PSRCC[3:2]	PC1	源电流 (mA) 11 = 26 01/10 = 8 00 = 4	PSRC2[3:0]	0x11C	RW−1111
PSRCC[1:0]	PC0				

表 2-3 I/O 相关用户寄存器

名称	状态		寄存器	地址	复位值
ITYPE0[1:0]	PORTx.0	外部中断管脚 EINTx 触发类型 00 = 低电平 01 = 上升沿 10 = 下降沿 11 = 双边沿	ITYPE0[1:0]	0x11E	RW−00
ITYPE0[3:2]	PORTx.1		ITYPE0[3:2]		RW−00
ITYPE0[5:4]	PORTy.2		ITYPE0[5:4]		RW−00
ITYPE0[7:6]	PORTy.3		ITYPE0[7:6]		RW−00
ITYPE1[1:0]	PORTy.4		ITYPE1[1:0]	0x11F	RW−00
ITYPE1[3:2]	PORTy.5		ITYPE1[3:2]		RW−00
ITYPE1[5:4]	PORTy.6		ITYPE1[5:4]		RW−00
ITYPE1[7:6]	PORTy.7		ITYPE1[7:6]		RW−00

表 2-4 外部管脚中断触发类型寄存器 (x = A, B, C； y = A, B)

名称	状态	寄存器	地址	复位值
EINT0	00 = PA0 01 = PB0 10 = PC0 11 = 保留	EPS0[1:0]	0x118	RW−00
EINT1	00 = PA1 01 = PB1 10 = PC1 11 = 保留	EPS0[3:2]		RW−00
EINT2	00 = PA2 01 = PB2 1x = 保留	EPS0[5:4]		RW−00
EINT3	00 = PA3 01 = PB3 1x = 保留	EPS0[7:6]		RW−00
EINT4	00 = PA4 01 = PB4 1x = 保留	EPS1[1:0]	0x119	RW−00
EINT5	00 = PA5 01 = PB5 1x = 保留	EPS1[3:2]		RW−00
EINT6	00 = PA6 01 = PB6 1x = 保留	EPS1[5:4]		RW−00
EINT7	00 = PA7 01 = PB7 1x = 保留	EPS1[7:6]		RW−00

表 2-5 外部中断管脚选择寄存器

名称	状态		寄存器	地址	复位值
EPIE0x	外部管脚中断	1 = 使能 0 = 禁止	EPIE0[7:0]	0x94	RW−00000000
EPIF0x [2]	外部管脚中断标志位	1 = Yes (锁存) 0 = No	EPIF0[7:0]	0x14	R_W1C−00000000

表 2-6 外部管脚中断使能和标志寄存器

名称	功能	默认
MCLRE	外部 I/O 复位	关闭
FOSC	• LP：PC1 (+) 和 PB7 (−) 接外部低速晶振 • XT：PC1 (+) 和 PB7 (−) 接外部高速晶振 • EC：PC1 (+) 接外部时钟输入，PB7 为 I/O • INTOSCIO：PC1 和 PB7 为 I/O	INTOSCIO

表 2-7 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 5 个模块 (**表 2-3**)：

[2] 写 1 清零，写 0 无效。建议只使用 STR、MOVWI 指令进行写操作，而不要用 BSR 指令。

- 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)
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; ANSELAx = 1
LVD	Off (5)	Off	Off	TRISx = 1; ANSELAx = 1(除 PC1, PB7 外)
V_{REF}+ / V_{REF}−	Off	Off	Off	TRISx = 1
ADC 触发	On	(可选)	Off	TRISx = 1
USART 输入	On	(可选)	Off	TRISx = 1
外部管脚中断	On	(可选)	Off	TRISx = 1
BKIN	On	(可选)	Off	TRISx = 1
数字输入	On	(可选)	Off	TRISx = 1
时钟输出	(忽略)	Off	On	TRISx = 0
PWM	On	Off	On	TRISx = 0
数字输出	On	Off	On	TRISx = 0
USART 输出	On	Off	On	TRISx = 0

表 2-8 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. ANSELAx = 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“ANSELAx = 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。当 LVD 输入需要在不同的通道之间切换使用时，通过设置“ANSELAx = 1”可关闭当前未被选择通道的“数字输入”，但 PB7 和 PC1 无 ANSELAx 控制，无法关闭“数字输入”，因此不应仅在部分时间作为 LVD 输入。
6. /MCLR 使能：PC0 的弱上拉功能自动使能 (忽略 WPUC[0])；读 PORTC[0] 的值为“0”。
7. 对 PORTx 数据输出寄存器或 LATx 数据锁存器进行写操作，I/O 端口都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) 或 LATx 数据锁存器，然后修改，再写回 PORTx/LATx 数据寄存器。
8. 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。

9. ODCON0x = 1：“ USART_TX ” 所选映射管脚(参阅 “AFP1”)开漏输出。开漏和内部上拉功能可以同时打开。
10. 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。

外部管脚中断的设置，请参阅 **章节 9** ”中断”。

2.3. 引脚输出优先级

每个 I/O 引脚均复用多个功能，当相应模块都使能输出时，输出优先级从低到高如**表 2-9** 所示。因为输入连接到各个功能模块，所以输入不存在优先级问题。

名称	优先级 0	优先级 1	优先级 2	优先级 3
PA0	PA0	TIM1_CH1	–	–
PA1	PA1	TIM1_CH2	–	–
PA2	PA2	[TIM1_CH4]	ISPCLK	
PA3	PA3	[TIM1_CH3]	–	–
PA4	PA4	TIM2_CH2	–	–
PA5	PA5	LVDOUT	USART_CK	TIM2_CH1
PA6	PA6	USART_TX	–	–
PA7	PA7	[CLKO]	[TIM1_CH4]	–
PB0	PB0	[TIM1_CH2]	[TIM2_CH1]	TIM1_CH3N
PB1	PB1	TIM1_CH2N	TIM1_CH4	CLKO
PB2	PB2	[TIM1_CH1N]	TIM1_CH3	–
PB3	PB3	[LVDOUT]	–	–
PB4	PB4	[LVDOUT]	–	–
PB5	PB5	[LVDOUT]	TIM2_CH3	–
PB6	PB6	[USART_TX]	ISPDAT	–
PB7	PB7	OSC2	–	–
PC0	PC0	TIM1_CH1N	MCLRB	–
PC1	PC1	OSC1	–	–

表 2-9 管脚输出优先级

对于 VerA 版芯片，注意事项如下：

1. TIM1_CH2 的映射管脚 PB0 只能用作 PWM 输出和比较输出，不能用作捕获输入；
2. 当 TIM1_CH2 输出使能时 (T1CC2E=1)，如果 AFP0[5] 选择 PB0，此时 PA1 仍被 TIM1_CH2 功能占据，不能作为 GPIO 输出，但可作为输入
3. 当 TIM2_CH1 输出使能时 (T2CC1E=1)，如果 AFP0[4] 选择 PA5，此时 PB0 仍被 TIM2_CH1 功能占据，不能作为 TIM1_CH2 输出或 GPIO 输出，但可作为输入

对于 ≥ VerB 版芯片，则无上述限制。

3. 上电复位 (POR)

上电过程，即 V_{DD} 从低于 Power-On-Reset 电压(V_{POR})上升至高于 V_{POR} 的过程。当 CPU 重新上电时，V_{DD} 可能没有完全掉电至 0V。

1. 当 V_{DD} 低于 V_{POR} 时，CPU 处于完全复位状态。
   a. 所有校准配置寄存器不复位。除 INDFx、Z、DC、C、FSRxL/H、BSREG、WREG、PORTx、LATx、OSCTUNE、EEDATL、EEDATH、EECON2 和 SRAM 以外（参阅 **章节 14** ”特殊功能寄存器”）的其他特殊功能寄存器（Special Function Registers, SFR）均处于复位状态。而不复位的寄存器如 SRAM，将保持其数据直至 V_{DD} 降到 0.6V(典型值)，当 V_{DD} 低于 0.6V 时，其值为不确定值。
   b. 程序计数器 PC = 0x00，指令寄存器 = “NOP”，堆栈指针 = “TOS” (栈顶)。
2. 当 V_{DD} 上升至 V_{POR} 以上时，芯片开始初始化配置(BOOT)过程。
3. 初始化配置完成后，指令将从 PC = 0x00 地址开始执行。

常温(25°C)下，V_{POR} 的典型值~1.6V，低温(-40°C)上升至~1.9V。当 $V_{DD} \geq V_{POR}$ 时, CPU 即可在较低的速度 8 MHz / 2T 下正常工作，因此 V_{DD} 的工作范围随温度变化而自动调整。此特性对于电池供电系统来说很重要，在典型的电池工作环境中，当电池电压低至~1.6V 时，CPU 仍可工作，从而提高电池使用寿命。

注：

1. V_{POR} 不可配置。

2. POR 的硬件电路默认为开启状态，当 V_{DD} 电压低于 V_{POR} 时即执行芯片电源复位，而不是仅在上电时执行。

3.1. 初始化配置时序

名称	功能	默认
PWRTEB	上电延时定时器，初始化配置完成后额外延时~64ms	关闭

表 3-1 初始化配置

以上初始化配置，由 IDE 界面设置，不能通过指令修改。初始化配置过程：

1. CPU 空闲等待~4ms；
2. 从非易失性存储器中加载初始化配置寄存器值，该过程~39us。这些寄存器值由 IDE 预先设置，不受指令影响；
3. 如果使能上电延时定时器(Power-On-Timer, PWRT)，CPU 将额外空闲等待~64ms。

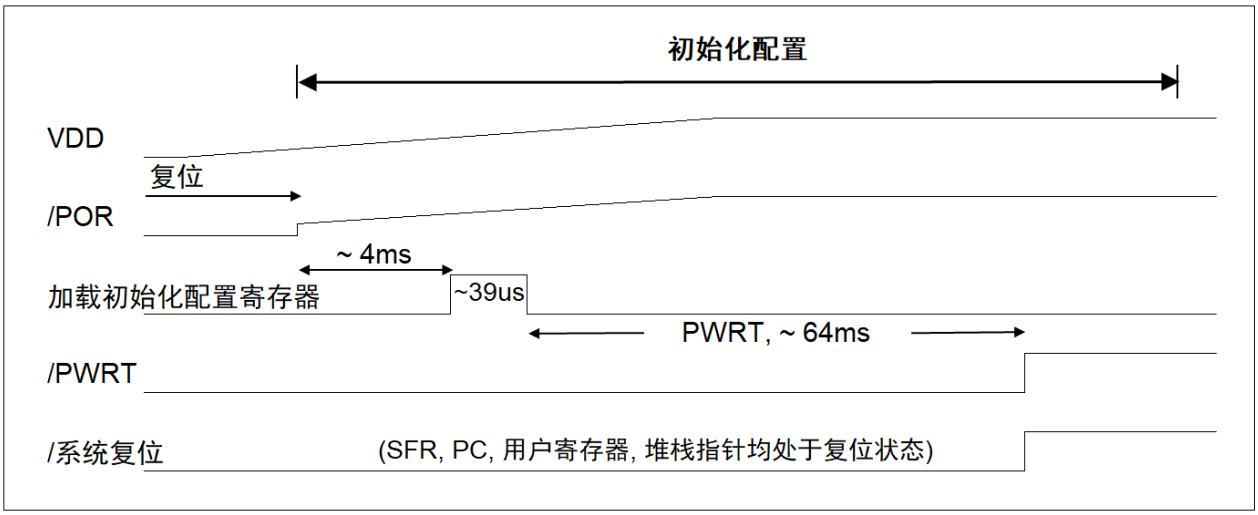


图 3-1 上电时序（PWRT 使能)

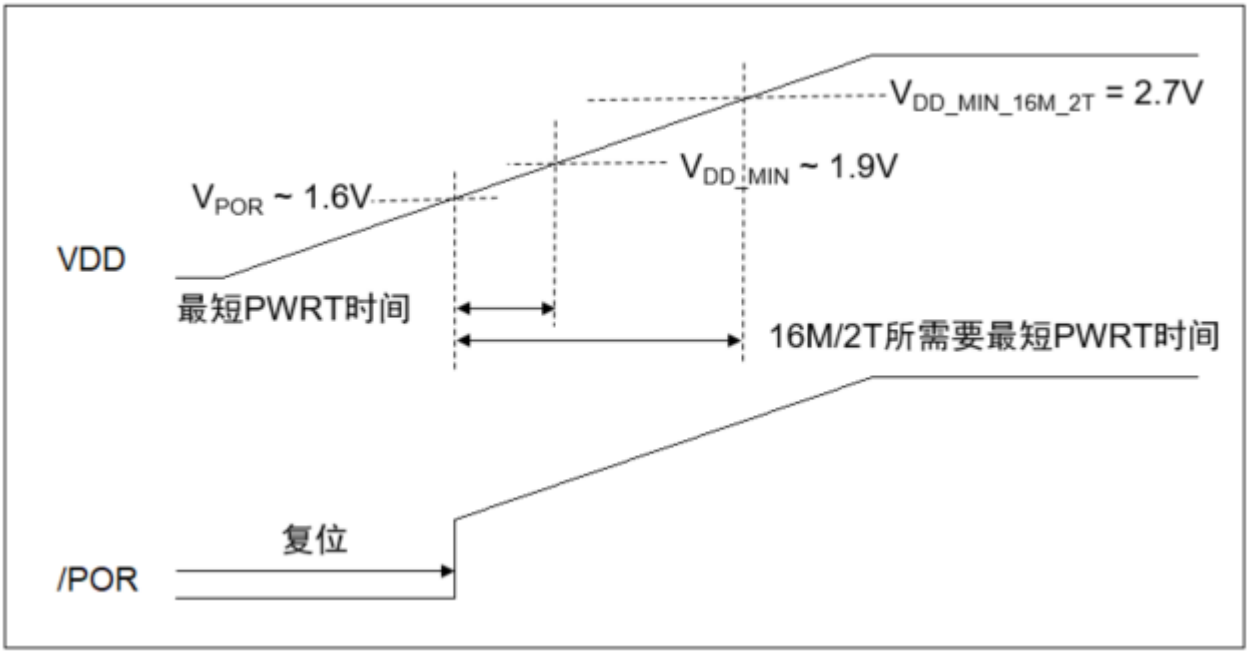


图 3-2 上电过程所需最小 PWRT 时间

如果 CPU 需要在 16MHz / 2T 的速度下运行，那么在初始化配置(BOOT)结束时 V_{DD} 必须高于 2.7V。通过使能 PWRT，可使初始化配置时间从~4ms 增加至~68ms，从而提高电源系统的稳定时间。

在以 16MHz / 2T 的速度运行时，应使能 LVR 且设置 $V_{BOR} \geq 2.7V$。另外，可通过指令控制 LVR 使能的频率以不时地监测 V_{DD}，而无需一直使能(参阅“LVREN”, “SLVREN”)以降低功耗。

注:

1. V_{DD} 上电过程不可以太慢，另外不建议 V_{DD} 的电容 $C_{VDD} \geq 22\ \mu F$；
2. V_{DD} 电容值以 1 到 10μF 为佳。出于 EFT 性能考虑，$C_{VDD} < 1\mu F$ 可能太小；
3. 如果可以接受启动延时，那么建议使能 PWRT 以提高 CPU 的稳定性；

4. 系统复位

与 POR 不同，系统复位(system reset) 并不会完全复位。系统复位时，CPU 是否启动初始化配置过程则取决于复位触发类型。若启动初始化配置则空闲等待~4ms，然后重新加载初始化配置寄存器值，如果使能 PWRT 将额外延时~64ms，随后系统正常启动。在系统复位中：

- 除初始化配置寄存器外，POR 过程中被重置的寄存器在系统复位时也会被同样重置；
- 程序计数器 PC = 0x00，指令寄存器 = “NOP”，堆栈指针 = “TOS” (栈顶)；

除仿真调试的 OCD(On-Chip Debugger) 模块外，以下 7 种事件可触发系统复位：

1. 欠压复位 (BOR / LVR) – 总会启动初始化配置；
2. 非法指令复位；
3. 看门狗复位 (WDT，CPU 处于非 SLEEP 状态)；
4. EMC 复位– 总会启动初始化配置；
5. 软件复位 (执行指令 “RESET”)；
6. 堆栈溢出复位 (上溢或下溢)；
7. 外部 I/O 复位 (/MCLRB) – 总会启动初始化配置 (≥ VerB 芯片)。

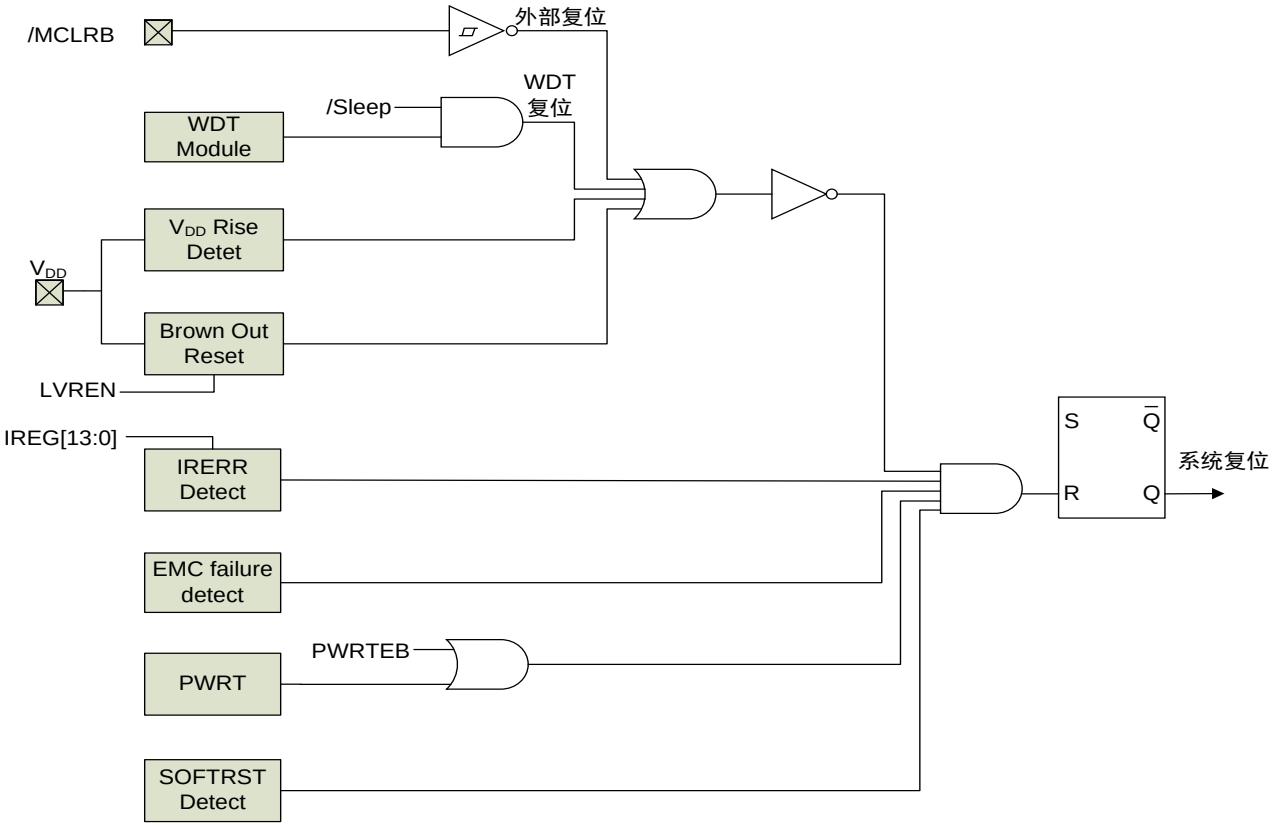


图 4-1 复位电路结构框图

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
STVREN	堆栈上/下溢复位 • 使能 • 关闭	使能
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

表 4-1 复位相关初始化配置寄存器

4.2. 欠压复位 (Brown-Out Reset, LVR / BOR)

当 V_{DD} 值降低并低于预设的欠压值 (V_{BOR}) 超过 T_{BOR} 时间时，就会出现欠压状态。T_{BOR} 大概为 3 到 4 个 LIRC 时钟周期 (~94 – 125us，如果未预先启动，LIRC 将自动开启)。当 $V_{DD} \le V_{BOR}$ 时， CPU 保持系统复位状态，直至 $V_{DD} > V_{BOR}$ 时 CPU 开始初始化配置过程(BOOT)，欠压复位标志位 /BORF 将置 0。

V_{POR} 值不能配置，而 V_{BOR} 值可以设置为 2.0、2.2、2.5、2.8、3.1、3.6、4.1V (参阅 “LVRS”，**表 4-1**)。

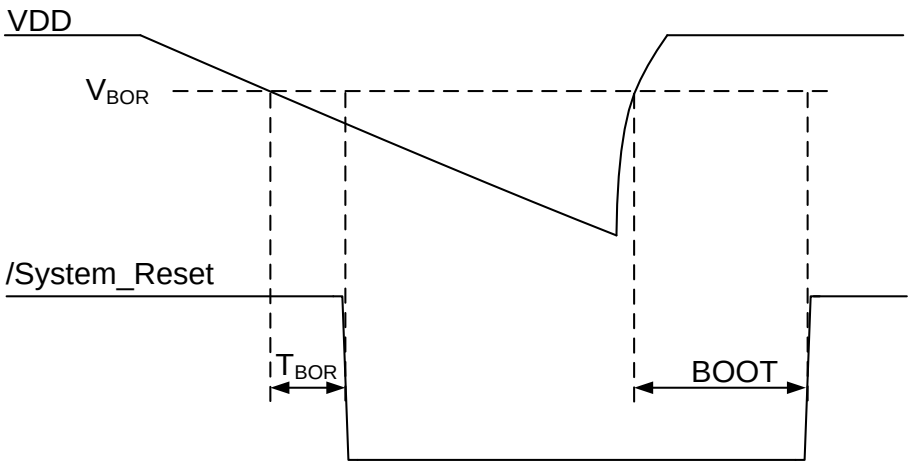


图 4-2 LVR 初始化配置时序图

LVR 可配置成 4 种不同功能(参阅 “LVREN”, **表 4-1**)。

1. LVR 使能；
2. LVR 关闭；
3. 非 SLEEP 模式下 LVR 使能；
4. 由指令控制使能或关闭 LVR (SLVREN, **表 4-2**)。

注： SLEEP 模式下，可通过指令关闭 LVR 以降低功耗。但如果系统 V_{DD} 不稳定，CPU 应定时唤醒并使能 LVR 来监测 V_{DD}。

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	LVDCON0[7]	0x199	RW-0

表 4-2 LVR 用户寄存器

4.3. 非法指令复位 (Illegal Instruction Reset)

CPU 获取指令错误的原因有很多，最常见的为干扰和 V_{DD} 不稳定。

虽然没有专用的复位指令，但任何蓄意的非法指令都等同于复位指令。出现非法指令时，产生系统复位，标志位 IERRR 将被置 1。

4.4. 软件复位

程序执行软件复位指令 “RESET” 时，产生系统复位，标志位 /SRSTF 将被置 0。

4.5. 堆栈溢出复位

堆栈上溢或下溢时，产生系统复位(通过配置 ”STVREN”)，上溢标志位 STKOVF 或下溢标志位 STKUNF 将被置 1。

4.6. EMC 复位

EMC 检测模块始终开启，当发生某种 EMC 干扰时，产生系统复位，并启动初始化配置过程，标志位 EMCF 将被置 1。

4.7. 看门狗定时器 (Watch Dog Timer, WDT) 复位

SLEEP 模式下，WDT 溢出将导致唤醒。

正常模式 (非 SLEEP 模式)下，WDT 溢出将触发系统复位。WDT 复位可用于复位挂起的 CPU。应在程序中不时地清除 WDT 以避免错误复位。

关于 WDT 的操作和设置等细节，请参阅 **章节 7.1** 看门狗定时器 (Watch Dog Timer, WDT)。

[1] 发生欠压复位时，此位不会清 0。其它复位会将此位清 0。

4.8. 外部 I/O 系统复位 /MCLRB

如果已相应设置初始化配置寄存器，那么可通过在/MCLRB (PC0)脚上施加低电压来使 CPU 复位。/MCLRB 脚通常经过一个电阻弱上拉到 V_{DD}，而不是直接连到 V_{DD}，如 **图 4-3** 所示，建议采用外部 RC 电路以提供故障滤波和过流保护。

对于 ≥ VerB 芯片，/MCLR 系统复位后，将产生初始化配置过程。

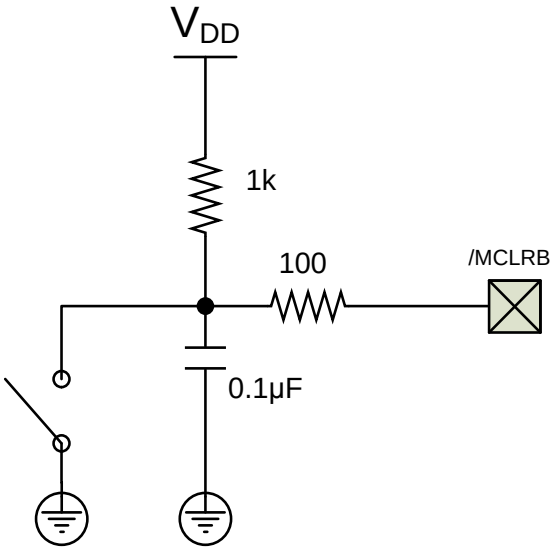


图 4-3 /MCLRB 复位电路

4.9. 检测上次复位类型

通过 PCON 寄存器中的 8 个状态标志位，以及 Time Out (/TO)和 Power Down (/PD)的不同组合可以追溯上一次系统复位的类型。/BORF 需由指令置 1，复位后将被锁定为“0”。

名称	状态		寄存器	地址	复位值
STKOVF [2]	堆栈上溢标志位	1 = Yes 0 = No，或已清 0	PCON[7]	0x96	RW0−0
STKUNF [2]	堆栈下溢标志位		PCON[6]		RW0−0
EMCF [2]	EMC 复位标志位		PCON[5]		RW0−0
IERRR [2]	非法指令复位标志位		PCON[4]		RW0−0
/MCLRR [3]	外部复位标志位	1 = No (软件置 1) 0 = Yes (锁存)	PCON[3]		RW1−1
/SRSTF [3]	软件复位标志位		PCON[2]		RW1−1
/PORF [3]	上电复位标志位		PCON[1]		RW1−0
/BORF [3]	低电压复位标志位		PCON[0]		RW1−x

表 4-3 复位标志寄存器

[2] 写 0 清零，写 1 无效。
[3] 只能写 1，写 0 无效。

复位源	STKOVF	STKUNF	EMCF	IERRR	/MCLRR	/SRSTF	/PORF	/BORF	/TO	/PD
	PCON[7]	PCON[6]	PCON[5]	PCON[4]	PCON[3]	PCON[2]	PCON[1]	PCON[0]	STATUS[4]	STATUS[3]
	0x96								Bank 首地址 + 0x03	
POR	0	0	0	0	1	1	0	1	1	1
LVR	0	0	0	0	1	1	1	0	–	–
CLRWDT 指令	–	–	–	–	–	–	–	–	1	1
SLEEP 指令	–	–	–	–	–	–	–	–	1	0
正常模式下(非 SLEEP) WDT 溢出(复位)	–	–	–	–	–	–	–	–	0	–
SLEEP 模式下 WDT 溢出(唤醒)	–	–	–	–	–	–	–	–	0	0
软件复位	–	–	–	–	–	0	–	–	–	–
MCLR 复位 (≥ verB)	0	0	0	0	1	1	0	1	1	1
MCLR 复位 (< verB)	–	–	–	–	0	–	–	–	–	–
非法指令复位	–	–	–	1	–	–	–	–	–	–
EMC 复位	–	–	1	–	–	–	–	–	–	–
堆栈下溢复位	–	1	–	–	–	–	–	–	–	–
堆栈上溢复位	1	–	–	–	–	–	–	–	–	–
片上调试 (OCD)	–	–	–	–	–	–	–	–	–	–

表 4-4 复位相关状态标志位（“–“ 无变化)

5. 低电压检测/比较器 (LVD)

LVD 的工作原理与 LVR 类似，但以下几点除外：

- 所有控制位和参数选择位均由用户指令设置，而不是初始化配置时；
- 必须正确设置 I/O：TRISx = 1; ANSELAx = 1 (不适用于 PB7 和 PC1，当 PB7 和 PC1 配置为输入但没有被选择为 LVD 输入，且端口电压不是 V_{DD} 或 GND 时，可能会有数字输入模块的漏电)；
- LVD 事件将置位 LVDW 而不是/BOR；
- 可通过指令配置为中断，且不会触发系统复位；
- 可通过 LVDDEB 使能消抖功能。消抖时间 (T_{LVD})为 3 – 4 LIRC 周期 (如果未预先启动，LIRC 将自动开启)；
- LVD 输入可以配置成 V_{DD} 或其他 4 个 I/O，后者允许将 LVD 当作单输入比较器功能使用，与 6 档 LVDL 电压值($V_{LVD-REF}$) 之一进行比较；
- 可设置 LVD 的极性，因此 LVD 可作为 $V_{LVD-REF}$ 的“高”或“低”比较器；
- 6 档可选 LVD 电压(LVDL)，可通过指令实现~2%/step 的微调精度；
- LVD 电平可输出到管脚 (参阅” LVDOE” 和 ” LVDOS”)；
- PC0 的外部复位功能 MCLRB 优先级高于 ELVD 功能。当配置为外部复位管脚时，ELVD 检测无效。

5.1. LVD 相关寄存器汇总

名称	状态			寄存器	地址	复位值
LVDM	LVD 输入	1 = 外部管脚 (ELVDx) 0 = VDD		LVDCON0[6]	0x199	RW−0
LVDEN	LVD	1 = 使能	0 = 关闭	LVDCON0[4]		RW−0
LVDW	LVD 触发?	1 = Yes (不锁存)	0 = No	LVDCON0[3]		RO−0
LVDL	$V_{LVD-REF}$	000 = 保留 001 = 保留 010 = 2.0 011 = 2.4	100 = 2.8 101 = 3.0 110 = 3.6 111 = 4.0	LVDCON0[2:0]		RW−000
LVDP	LVDW 极性	1 = 检测电压 > $V_{LVD-REF}$ 0 = 检测电压 < $V_{LVD-REF}$		LVDCON1[4]	0xF8E	RW−0
LVDDEB	LVD 消抖	1 = 使能	0 = 关闭	LVDCON1[3]		RW−1
LVDOE	LVD 输出	1 = 使能	0 = 关闭	LVDCON1[2]		RW−0
LVDOS	LVD 输出管脚	00 = PA5 01 = PB5	10 = PB4 11 = PB3	LVDCON1[1:0]		RW−00
LVDCAL	LVDL 修调位, ~2% / step			LVDTUNE[7:4]	0xF92	RW−1000
ELVDS	LVD 外部输入管脚选择	00 = ELVD0 01 = ELVD1	10 = ELVD2 11 = ELVD3	ADCON3[1:0]	0x41A	RW−00
LVDIE	LVD 中断	1 = 使能	0 = 关闭	INTCON[4]	0x0B	RW−0
LVDIF [1]	LVD 产生中断?	1 = Yes	0 = No，或已被清零	INTCON[1]		R_W1C−0

表 5-1 LVD 用户设置和标志寄存器

[1] 写 1 清零，写 0 无效。建议只使用 STR、MOVWI 指令进行写操作，而不要用 BSR 或 IOR 指令。

6. 振荡器和系统时钟

系统时钟(SysClk) 可通过指令选择为内部高速振荡器 HIRC，内部低速振荡器 LIRC，或外部振荡器 (EC, LP, XT, 参阅 “SCS”, **表 6-2**)。如果选择外部振荡器，那么由初始化配置寄存器 “FOSC” (**表 6-1**) 选定 3 种外部振荡器之一。系统时钟还可通过指令进一步选择为内部或外部振荡器的分频 (参阅 MCKCF, **表 6-2**)。系统时钟用于产生指令时钟(Instruction Clock)：

指令时钟 = SysClk / N ; N = 2 for 2T, 4 for 4T.

外部时钟输入引脚由初始化配置寄存器设置 (参阅 FOSC)。可通过指令选择 8 种时钟源输出(参阅 "CCOSEL"和"CCOEN")，当时钟输出使能时，其所选时钟源自动开启，标志位 CCORDY 置 1，输出引脚可选择为 PA7 或 PB1 (参阅 "CLKOS", **表 6-3**) 。

外设 Timers、ADC 和 USART 均有独立的模块系统时钟控制位(参阅"PCKEN")。需先使能模块时钟，再使能相应模块。关闭时，需先关闭模块功能，再关闭模块时钟。此外，Timers 的计数时钟源和 ADC 的转换时钟源有独立的振荡器，当模块系统时钟和模块功能均使能时，其所选振荡器将自动开启，且在模块运行期间一直保持有效，因此可有多个振荡器同时运行。

SLEEP 模式下指令停止运行，指令时钟默认停止，时钟输出暂停。当 SYSON = 1 时，指令时钟将保持运行，因此使能模块系统时钟的相应外设也将在 SLEEP 模式下保持工作，此外时钟输出也将继续。

注：

1. SLEEP 模式下，当 SYSON = 1 且 TIMxEN = 1 时，Timers 所选计数时钟源将保持运行；
2. 当 ADC 转换时钟源选择 LIRC 时，进入 SLEEP 模式后，LIRC 将保持运行，与 SYSON 无关；
3. 建议关闭未使用外设的模块时钟，以降低功耗。

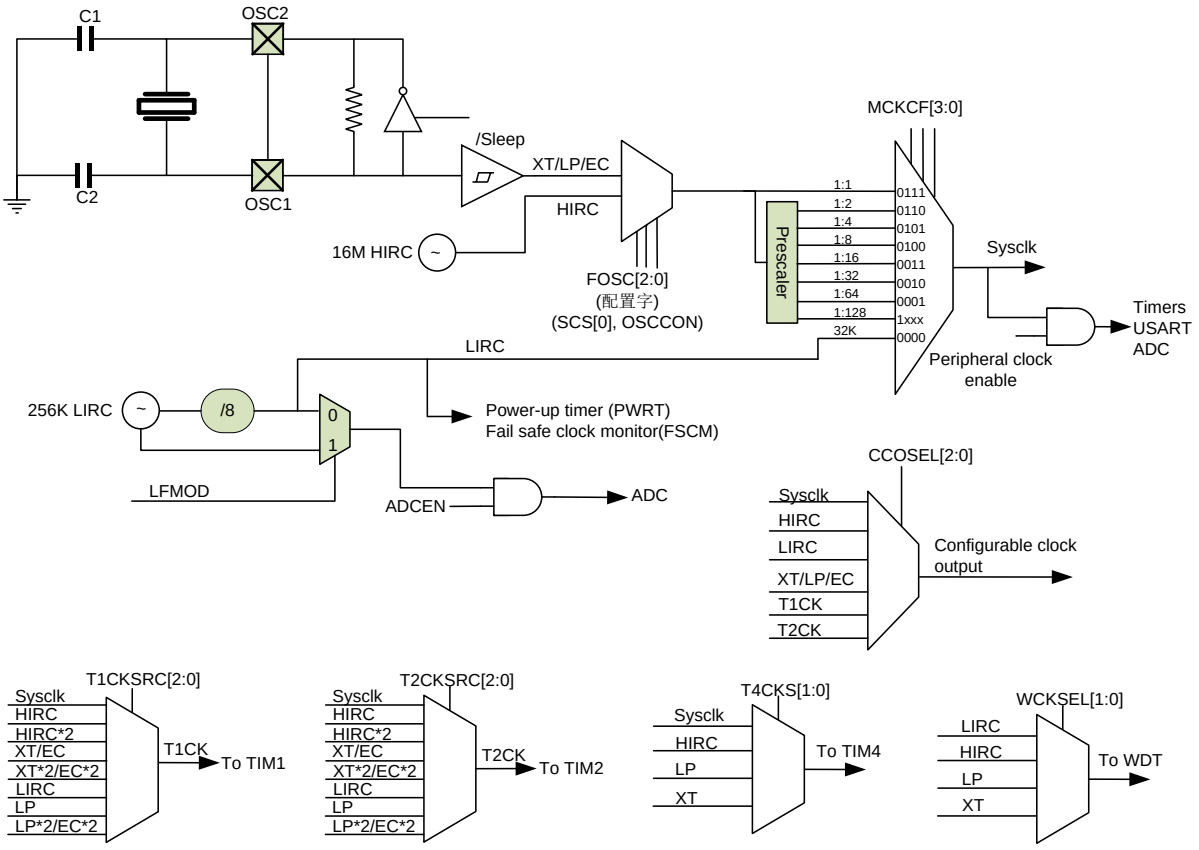


图 6-1 系统时钟 SysClk 的时钟源框图

6.1. 振荡器模块相关寄存器汇总

名称	功能		默认
FOSC	• LP：PC1 (+) 和 PB7 (−) 接外部低速晶振 • XT：PC1 (+) 和 PB7 (−) 接外部高速晶振 • EC：PC1 (+) 接外部时钟输入，PB7 为 I/O • INTOSCIO：PC1 和 PB7 为 I/O		INTOSCIO
IESO	XT / LP 双速时钟启动	• 使能 • 关闭	使能
FSCMEN	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
OSTPER	OST 定时器周期选择 (XT / LP 适用)	• 512 • 1024 • 2048 • 4096 (LP 模式时为 32768)	1024

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	LFMOD	OST
			OSCCON[0]	TCKSRC[7]	(可选，参阅 OSTPER)
			0x99	0x31F	
			RW-0	RW-0	
外部	EC		0	-	-
	XT		0	-	1,024 (默认值)
	LP		0	-	1,024 (默认值)
内部	HIRC	16 MHz	1	-	-
	LIRC	256 kHz [2]	1	1	-
		32 kHz [3]	1	0	-

名称	状态	寄存器	地址	复位值
MCKCF	SysClk 系统时钟分频 (EC/XT/LP/HIRC) 0111 = 1　0100 = 8　0001 = 64 0110 = 2　0011 = 16　1xxx = 128 0101 = 4　0010 = 32　0000 = LIRC	OSCCON[7:4]	0x99	RW−0011

表 6-2 SysClk 系统时钟源设置相关用户寄存器

[2] 256 kHz LIRC 只供 ADC (参阅 ADCS 和 LFMOD，**表 11-3**) 使用。

[3] 系统时钟源 (MCKCF=0000)、PWRT、LIRC 和 HIRC 交叉校准、FSCM、WDT (WCKSRC=00)、Timer1 (T1CKSRC = 101)、Timer2 (T2CKSRC = 101) 和时钟输出 (CCOSEL = 010) 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]	0x99	RO-0
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 模式下，系统时钟控制 1 = 保持运行 0 = 关闭		CKOCON[7]	0x95	RW-0
CCORDY	时钟输出标志位	1 = Yes 0 = No	CKOCON[6]		RO-0
DTYSEL	TIM1/TIM2 倍频时钟占空比调节位 00 = 2ns 延迟 10 = 4ns 延迟 01 = 3ns 延迟 11 = 7ns 延迟		CKOCON[5:4]		RW-10
CCOSEL	输出时钟选择位 000 = Sysclk 100 = T1CK 001 = HIRC 101 = T2CK 010 = LIRC 110 = LP (*) 011 = XT (*) 111 = EC (*) (*) FOSC 应相应配置成 LP/XT/EC 模式，否则时钟输出可能不正确或无输出；		CKOCON[3:1]		RW-000
CCOEN	时钟输出	1 = 使能 0 = 禁止	CKOCON[0]		RW-0
CLKOS [4]	时钟输出引脚 1 = CLKO 映射到 PB1 0 = CLKO 映射到 PA7		CKAUX[1]	0xF8D	RW-1
CKMAVG	LIRC 和 HIRC 交叉校准时 4 次平均测量模式 1 = 使能 0 = 关闭		MSCKCON[1]	0x41D	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)		MSCKCON [0]		RW-0

[4] 在修改 CLKOS 之前需将 EECON1.CFGS 置 1，写之后需清零 EECON1.CFGS。

名称	状态		寄存器	地址	复位值
SOSCPR	校准 LIRC 周期所需的 HIRC 周期数		SOSCPR[11:0]	0x41F[3:0] 0x41E[7:0]	RW−FFF
TUN	内部高速时钟 HIRC 频率调节寄存器		OSCTUNE[7:0]	0x98	RW−xxxx xxxx
UARTEN	USART 模块时钟	1 = 使能 0 = 禁止	PCKEN[5]	0x9A	RW−0
TIM4EN	Timer4 模块时钟		PCKEN[3]		RW−0
TIM2EN	Timer2 模块时钟		PCKEN[2]		RW−0
TIM1EN	Timer1 模块时钟		PCKEN[1]		RW−0
ADCEN	ADC 模块时钟		PCKEN[0]		RW−0

表 6-3 振荡器控制/状态位

名称	状态		寄存器	地址	复位值
GIE	全局中断 1 = 使能 (PEIE, OSFIE, CKMIE 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 (OSFIE, CKMIE 适用) 0 = 关闭 (无唤醒)		INTCON[6]		RW−0
OSFIE	外部振荡器故障中断	1 = 使能 0 = 关闭 (无唤醒)	INTCON[3]		RW−0
OSFIF [5]	外部振荡器故障标志位	1 = Yes (锁存) 0 = No	INTCON[0]		R_W1C−0
CKMIE	LIRC和HIRC交叉校准完成中断	1 = 使能 0 = 关闭 (无唤醒)	PIE1[1]	0x91	RW−0
CKMIF [5]	LIRC和HIRC交叉校准完成标志位	1 = Yes (锁存) 0 = No	PIR1[1]	0x11	R_W1C−0

表 6-4 振荡器中断使能/状态位

6.2. 内部时钟模式 (HIRC 和 LIRC)

内部高频时钟 (Internal high frequency clock, HIRC) 出厂时已校准到 16 MHz @ 2.5V/25°C。芯片之间的频率变化典型值 < ±1.5% @2.5 − 5.5V/25°C，温度变化典型值为 ±2.0% @$^{-}$40 − $^{+}$85 °C。

HIRC 精度在晶圆测试时已进行校准。封装过程可能会导致 HIRC 频率漂移。烧录器软件可选择对 HIRC 进行重新校准。HIRC 出厂校准值已存储到 “OSCTUNE” 寄存器中，用户可通过指令修改 “OSCTUNE” 来微调 HIRC 频率 (默认 16 MHz)，微调 steps 是非线性的(~40 kHz)。粗略估计如下：

$$OSCTUNE[7:0] \pm N \approx 16000 \pm N * 40 \text{ (kHz)}$$

内部低频时钟 (Internal low frequency clock, LIRC) 出厂时已校准至 32kHz。芯片之间的频率变化典型值为 < ±9.5% @2.5 − 5.5V/25°C，温度变化典型值 < ±3.0% @$^{-}$40 − $^{+}$85 °C。

[5] 写 1 清零，写 0 无效。建议只使用 STR、MOVWI 指令进行写操作，而不要用 BSR 或 IOR 指令。

LIRC 和 HIRC 可相互交叉校准 – 在一个 LIRC 周期 (32 kHz) 内使用 Timer2 来测量指令时钟数(SysClk 选择 16MHz HIRC)，此为内置硬件功能。

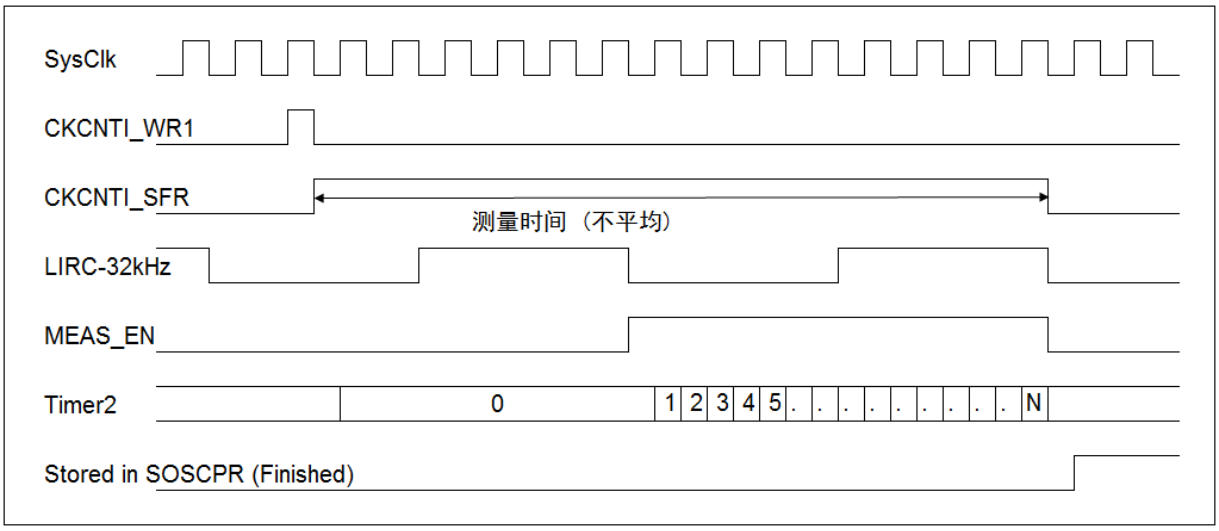


图 6-2 单次测量时序图

LIRC 和 HIRC 交叉校准步骤：

1. 设置 MCKCF = 111, SCS = 1 ; SysClk 选择 16MHz HIRC (其他频率设置的精准度会降低)
2. 设置 CKMAVG = 1 ; 4 次测量平均，选择 0 表示不做平均
3. 设置 TIM2EN= 1, T2CEN = 1 ; 使能 Timer2
4. 设置 CKCNTI = 1 ; 开始校准，默认 Timer2 预分频比 = 1，后分频比 = 1,
   ; T2CKSRC = HIRC
5. 校准完成时，CKCNTI 自动清零(“CKCNTI =0”), CKMIF 自动置位(“CKMIF = 1”)。
6. 测量值存储在 SOSCPR 寄存器中。
7. LIRC 默认为 32kHz，且 CPU 运行在 16MHz / 2T 下，则理想的匹配值为 500。

注：

- LIRC 和 HIRC 交叉校准时，不要对 SOSCPRH/L 寄存器进行写操作；
- LIRC 和 HIRC 交叉校准时，Timer2 不能被其他外设使用；
- LIRC 和 HIRC 交叉校准功能与 IDE 的单步调试模式不兼容；
- 当 CKCNTI = 1 时，LIRC 自动开启，且进入 SLEEP 模式后保持运行，但仅当 SYSON = 1 时，校准才可在 SLEEP 模式下运行。
- 上电后，LIRC 和 HIRC 交叉校准将会自动启动，此时 CKCNTI = 1，CKMAVG = 0，无需置位 T2CEN。自动交叉校准完成后，CKMIF 标志位不会置位，CKCNTI 自动清零，此时 TIM2 才可配置或被其他外设使用。

6.3. 外部时钟模式 (EC / LP / XT)

6.3.1. EC 模式

外部数字信号作为时钟源连接到 OSC1 脚(OSC2 用作 I/O)。当 SysClk 选择 EC 模式时，当 POR 复位或从睡眠中唤醒时，EC 模式不需要设置或转换时间延迟。

6.3.2. LP 和 XT 模式

LP 或 XT 模式下，石英晶体谐振器或陶瓷谐振器作为时钟源连接到 OSC1 和 OSC2 脚。

LP 振荡器模式是 3 种模式 (EC，LP，XT) 中增益设定和电流消耗最低的。该模式仅用于驱动 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最高增益设定。

如果时钟源选择 XT 或 LP 模式，当初始化配置结束或从睡眠中唤醒时，CPU 在振荡器起振定时器 (OST) 计数期间将暂停执行程序，这有利于 XT 或 LP 时钟的稳定。对于 XT 和 LP 模式，OST 对 OSC1 (晶体输入+ve 端) 进行计数，计数个数由初始化配置寄存器 OSTPER 决定。对于 32.768 kHz 音叉式晶振，当 OSTPER=32,768 时，OST 计时至少需要 1 秒。

注：

- WDT 将保持清零状态直至 OST 完成计数；
- OST 计数期间，不要对 WDTCON 或 OPTION 寄存器进行写操作，否则将产生不可预期的行为；

双速时钟启动 (参阅“IESO”，**表 6-1**) 允许 CPU 在 OST 计数期间选择内部振荡器 INTOSC 模式作为 SysClk 进而执行指令。在需要频繁进出睡眠模式的情况下，双速时钟启动功能可使芯片在唤醒后立即执行指令，从而除去外部振荡器所需的起振时间，以降低整机功耗。即 CPU 从睡眠中唤醒，将 INTOSC 作为 SysClk 执行几条指令后，再返回睡眠状态，而无需等待外部振荡器的稳定。

注：EC 模式下双速启动功能关闭，因其振荡器不需要稳定时间。

双速启动时序：

1. 初始化配置结束或从睡眠中唤醒；
2. 选择 INTOSC 作为 SysClk 执行指令直至 OST 超时；
3. SysClk 从 INTOSC 的下降沿一直保持为低，直至新时钟的下降沿到来 (LP 或 XT 模式)；
4. SysClk 切换到外部时钟源；

振荡器起振超时状态位(OSTS)用于提示 SysClk 运行在外部时钟源下或内部时钟源下。当开启双速时钟启动功能时，通过 OSTS 可间接查询 LP 或 XT 模式下的振荡器起振定时器(OST)是否已经超时。

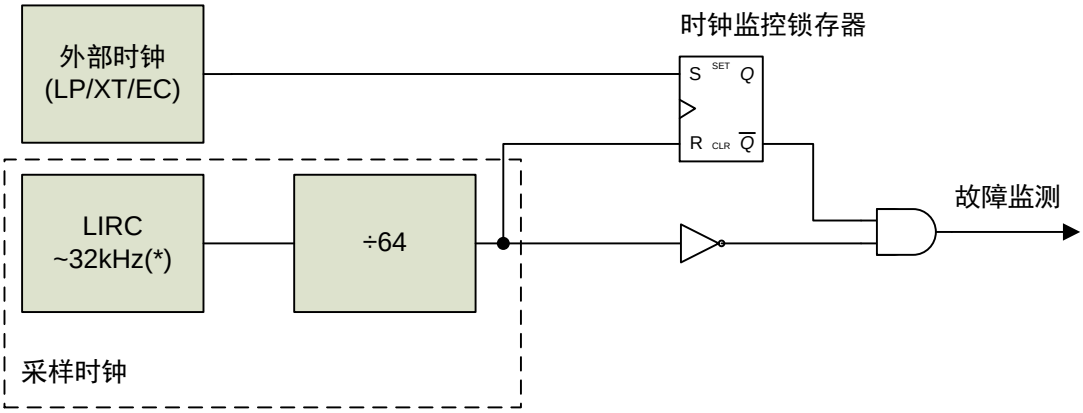
执行 SLEEP 指令将中止 OST 计时，而 OSTS 将保持为“0”。

故障保护时钟监控器 (Fail-Safe Clock Monitor, FSCM，由“FSCMEN”使能，参阅**表 6-1**) 可使芯片在外部振荡器出现故障时仍能继续工作。在振荡器起振定时器 (OST) 超时后，FSCM 随时检测振荡器故障。FSCM 适用于所有外部振荡器模式(EC，LP 和 XT)。当选择外部振荡器时，建议使能 FSCM 功能。

如果外部振荡器的振荡频率在~1 kHz 及以下时，则判定其出现故障。由 LIRC 除以 64 产生采样时钟，故障监控器内部有一个锁存器，外部时钟在其每个下降沿将锁存器置 1，采样时钟在其每个上升沿将锁存器清 0。当采样时钟的整个半周期结束而主时钟依然未进入低电平时，即检测到故障。

当外部时钟出现故障时，FSCM 自动将 SysClk 切换成内部时钟源并置位 OSFIF。如果 OSFIE 使能，OSFIF 置 1 将产生中断。芯片固件随后应采取措施来减轻可能由故障时钟所导致的问题。SysClk 将继续运行在内部时钟源下，直到芯片固件成功重启外部振荡器。

由“MCKCF”决定“FSCM”所用的内部时钟源，这使得内部振荡器在外部时钟发生故障前就得以配置。



注：LFMOD不影响采样时钟。

图 6-3 FSCM 结构框图

复位、执行 SLEEP 指令或翻转 SCS 位后，故障保护条件将被清除。当 SCS 位被修改后，OST 将重新启动计时。OST 运行期间，CPU 选择 INTOSC 模式作为 SysClk 继续执行指令。OST 超时后，故障保护条件被清除，芯片将切换回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。

注：任何可能由双速时钟启动或故障保护时钟监控器所产生的时钟自动切换，都不会更新 SCS 位。程序应监控 OSTS 位以确定当前的 SysClk 系统时钟源。

6.4. HIRC, LIRC 和 EC 时钟的内部切换

图 6-4 为时钟内部切换时序图。若 HIRC 或 LIRC 在切换前已经关闭(为了省电)，则将会有额外的振荡器设置延迟时间，可通过 HTS 和 LTS 标志位查询相应振荡器的状态。

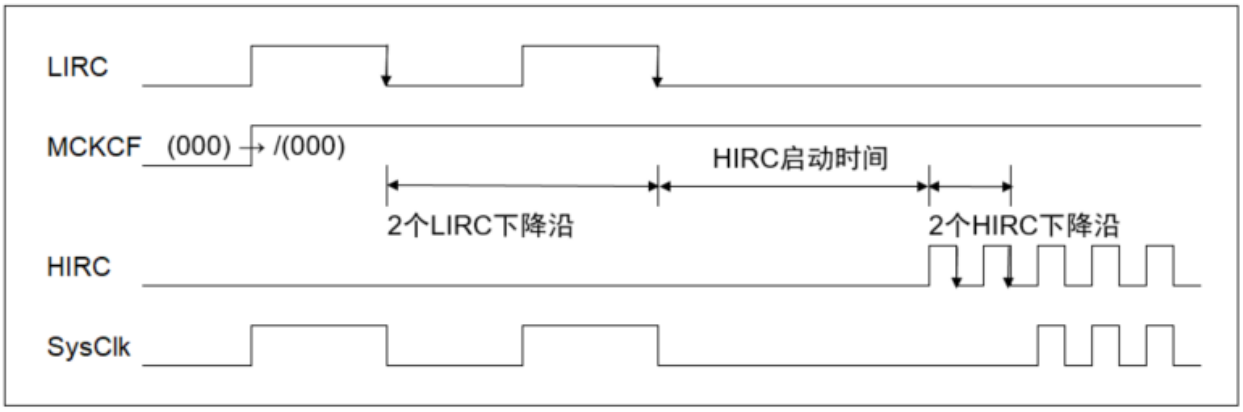


图 6-4 由 LIRC 切换到 HIRC 时序图 (同样的原则页适用于 EC, LIRC, HIRC 之间的切换)

7. 定时器 (TIMERS)

共有 4 个定时器，包括看门狗定时器(WDT)在内。

	WDT	Timer1	Timer2	Timer4
预分频器（位)	3	16	4 (1x, 4x, 16x)	3
计数器（位)	16	16	16	8
后分频器（位)	–	–	–	–
时钟源	• LP • XT • HIRC • LIRC	• EC, LP or XT • HIRC • Sysclk • LIRC • 2x HIRC • 2x (EC, LP or XT)	• EC, LP or XT • HIRC • Sysclk • LIRC • 2x HIRC • 2x (EC, LP or XT)	• LP • XT • HIRC • Sysclk

表 7-1 定时器资源

注： 如果定时器的时钟源不是系统时钟，在更改 TMRx 之前需先设置“TxCEN = 0”。

当定时器使能时，其所选的时钟源会自动开启。系统时钟在 SLEEP 模式下被默认关闭，因此不能用于 WDT。当定时器选择 LP / XT/ EC 振荡器作为时钟源时，FOSC 必须相应配置成 LP / XT / EC 模式，否则振荡器将处于关闭状态，不会产生计数。

当 POR 或系统复位时，所有定时器的计数器和预分频器都将复位。以下事件也将复位相应定时器的计数器和预分频器：

	WDT	Timer1	Timer2	Timer4
预分频器	• WDT 关闭	• 复位模式	• T2CEN = 0	• T4CEN = 0
计数器	• WDT, OST 溢出 • 进入/退出 SLEEP • CLRWDT • 写 WDTCON • 写 WCKSEL	• T1CNT = T1ARR	• T1CNT=T2ARR	• T4CNT = T4ARR

表 7-2 定时器的计数器和分频器的重置事件

7.1. 看门狗定时器 (Watch Dog Timer, WDT)

WDT 用于 “从 SLEEP 中唤醒” 或 “CPU 挂起时产生系统复位”。当 WDT 计数到预设数量的时钟周期数时则产生溢出。

- SLEEP 模式下，WDT 溢出将触发唤醒。CPU 将从其进入 SLEEP 前的位置恢复操作。唤醒不是中断，也不是系统复位事件。
- 正常模式(非 SLEEP 模式)下，WDT 溢出将触发系统复位，并启动初始化配置过程(参阅 **章节 4** 系统复位)。

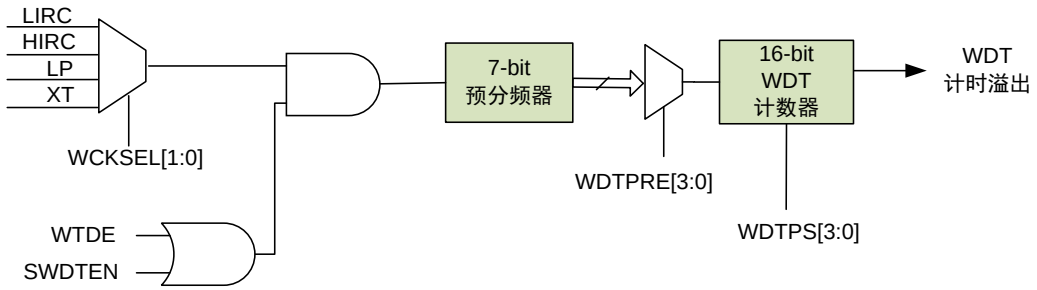


图 7-1 WDT 结构框图

计时超过看门狗定时时间：WDT-周期 x WDT-预分频比 / WDT 时钟频率，WDT 将溢出。

对于选定的时钟源，由于 WDT 预分频器的二进制特性，看门狗定时时间步长呈连续的倍数关系。选用 LIRC 作为时钟源时，WDT 溢出前可设置的最长定时时间为：

$$2^{16} \times 2^{7} / 32\text{kHz} = \sim 262 \text{ seconds.}$$

7.1.1. WDT 相关寄存器汇总

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

表 7-3 WDT 选项初始化配置寄存器

名称	状态	寄存器	地址	复位值
WCKSEL	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	MISC0[1:0]	0x11C	RW-00
WDTPRE	WDT 预分频器 000 = 1　100 = 16 001 = 2　101 = 32 010 = 4　110 = 64 011 = 8　111 = 128 (默认)	WDTCON[7:5]	0x97	RW-111
WDTPS	WDT 周期 0000 = 32 0001 = 64 0010 = 128 0011 = 256 0100 = 512 (默认) 0101 = 1,024 0110 = 2,048 0111 = 4,096 1000 = 8,192 1001 = 16,384 1010 = 32,768 1011 = 65,536 11xx = 65,536	WDTCON[4:1]		RW-0100
SWDTEN	1 = WDT 使能 0 = WDT 关闭 (当 WDTE 选择由 SWDTEN 控制时)	WDTCON[0]		RW-0

表 7-4 WDT 相关用户寄存器

7.1.2. WDT 的设置和使用

由 WDTE (初始化配置寄存器) 以及 SWDTEN (用户寄存器) 使能 WDT，WDT 触发复位后将产生初始化配置过程。

WDT 预分频器由 WDTPRE 设置，时钟源由 WCKSEL 选择 (如果选 LIRC 则默认为 32kHz，而不管 LFMOD 为何值)。当 WDT 使能时，其所选时钟源自动开启，且在 SLEEP 模式下保持运行。

如需阻止 WDT 溢出，必须在设定的定时时间之前清除 WDT，具体可参阅**表 7-2** 中的清除 WDT 事件。WDT 被清除后将重新开始计时。

7.2. 高级定时器 1 (TIMER1)

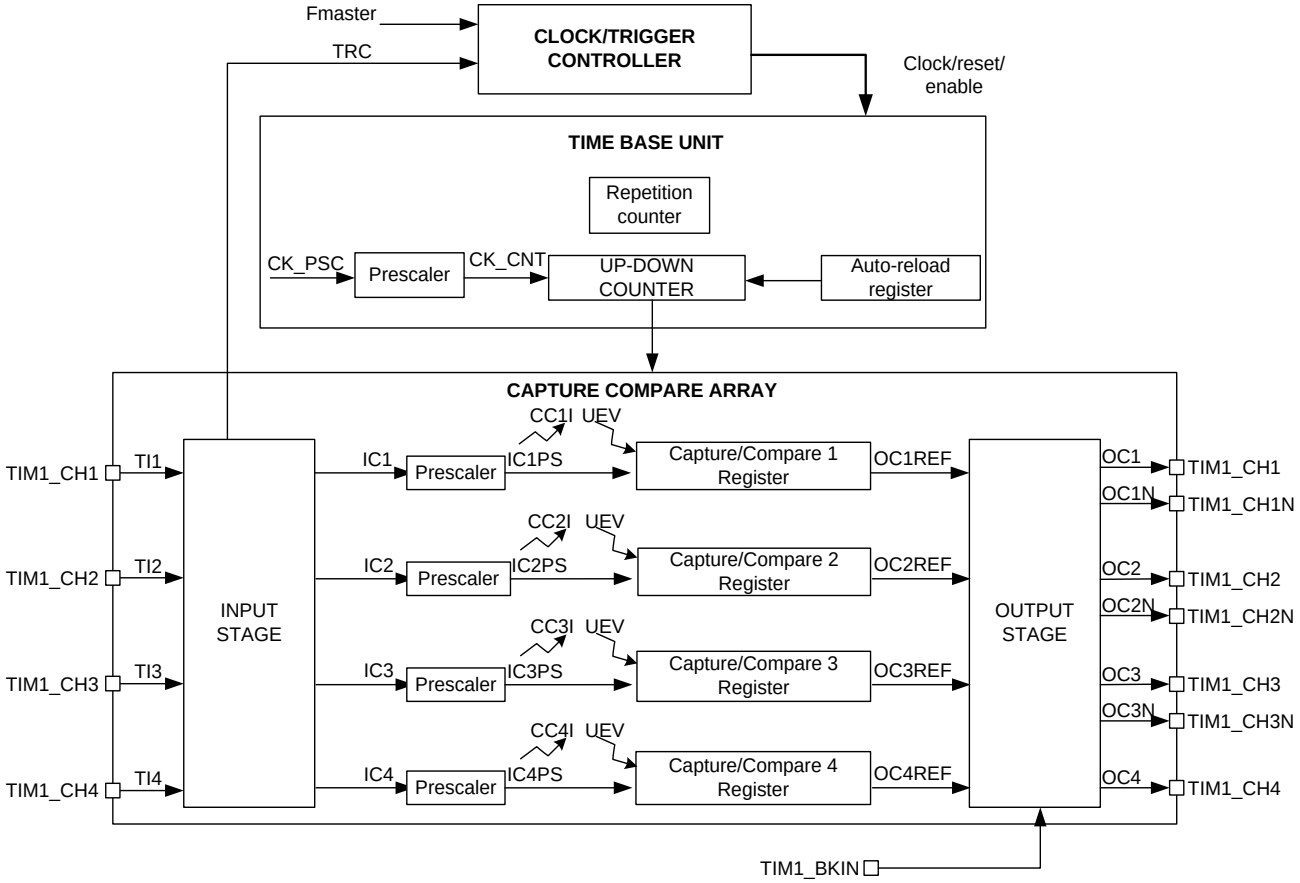


图 7-2 TIM1 结构框图

TIM1 特性：

- 16 位向上、向下、上/下计数器，支持自动重载
- 重复计数器
- 16 位可编程预分频器
- 计数控制模式：
  - ✓ 内部时钟模式、编码器模式 1/2/3、复位模式、门控模式、触发模式、外部时钟模式 1/2
- 4 路极性可选的通道支持：
  - ✓ 输入捕获
  - ✓ 输出比较
  - ✓ 周期相同、独立占空比的 PWM 通道(边沿或中心对齐)，3 路支持互补输出及可编程死区时间
  - ✓ 单脉冲输出
  - ✓ 6 步 PWM、同步模式、群组模式
  - ✓ 故障刹车功能(可选自动重启功能)
- TIM1 / TIM2 同步触发模式
- 中断事件：更新事件、输入触发、换相事件、输入捕获、输出比较、故障刹车输入
- 支持 3 级寄存器写保护锁定设置(T1LOCK)

7.2.1. Timer1 相关寄存器汇总

名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	复位值
PCKEN	0x9A	—	—	UARTEN	—	TIM4EN	TIM2EN	TIM1EN	ADCEN	0000 0000
CKOCON	0x95	SYSON	CCORDY	DTYSEL[1:0]		CCOSEL[2:0]			CCOEN	0010 0000
TIM1CR1	0x211	T1ARPE	T1CMS[1:0]		T1DIR	T1OPM	T1URS	T1UDIS	T1CEN	0000 0000
TIM1CR2	0x212	T1ENCTRL	—	—	—	—	T1COMS	—	T1CCPC	0--- -0-0
TIM1SMCR	0x213	—	T1TS[2:0]			—	T1SMS[2:0]			-000 -000
TIM1ETR	0x214	T1ETP	T1ECE	T1ETPS[1:0]		T1EFT[3:0]				0000 0000
TIM1IER	0x215	T1BIE	T1TIE	T1COMIE	T1CC4IE	T1CC3IE	T1CC2IE	T1CC1IE	T1UIE	0000 0000
TIM1SR1	0x216	T1BIF	T1TIF	T1COMIF	T1CC4IF	T1CC3IF	T1CC2IF	T1CC1IF	T1UIF	0000 0000
TIM1SR2	0x217	—	—	—	T1CC4OF	T1CC3OF	T1CC2OF	T1CC1OF	—	---0 000-
TIM1EGR	0x218	T1BG	T1TG	T1COMG	T1CC4G	T1CC3G	T1CC2G	T1CC1G	T1UG	0000 0000
TIM1CCMR1 (output mode)	0x219	T1OC1CE	T1OC1M[2:0]			T1OC1PE	T1OC1FE	T1CC1S[1:0]		0000 0000
TIM1CCMR1 (input mode)		T1IC1F[3:0]				T1IC1PSC[1:0]		T1CC1S[1:0]		0000 0000
TIM1 CCMR2 (output mode)	0x21A	T1OC2CE	T1OC2M[2:0]			T1OC2PE	T1OC2FE	T1CC2S[1:0]		0000 0000
TIM1CCMR2 (input mode)		T1IC2F[3:0]				T1IC2PSC[1:0]		T1CC2S[1:0]		0000 0000
TIM1CCMR3 (output mode)	0x21B	OC3CE	T1OC3M[2:0]			OC3PE	OC3FE	T1CC3S[1:0]		0000 0000
TIM1CCMR3 (input mode)		T1IC3F[3:0]				T1IC3PSC[1:0]		T1CC3S[1:0]		0000 0000
TIM1CCMR4 (output mode)	0x21C	OC4CE	T1OC4M[2:0]			OC4PE	OC4FE	T1CC4S[1:0]		0000 0000
TIM1CCMR4 (input mode)		T1IC4F[3:0]				T1IC4PSC[1:0]		T1CC4S[1:0]		0000 0000
TIM1CCER1	0x21D	T1CC2NP	T1CC2NE	T1CC2P	T1CC2E	T1CC1NP	T1CC1NE	T1CC1P	T1CC1E	0000 0000
TIM1CCER2	0x21E	T1GP	T1SMOD	T1CC4P	T1CC4E	T1CC3NP	T1CC3NE	T1CC3P	T1CC3E	0000 0000
TIM1CNTRH	0x28C	T1CNT[15:8]								0000 0000
TIM1CNTRL	0x28D	T1CNT[7:0]								0000 0000
TIM1PSCRH	0x28E	T1PSC[15:8]								0000 0000
TIM1PSCRL	0x28F	T1PSC[7:0]								0000 0000
TIM1ARRH	0x290	T1ARR[15:8]								1111 1111
TIM1ARRL	0x291	T1ARR[7:0]								1111 1111
TIM1RCR	0x292	T1REP[7:0]								0000 0000
TIM1CCR1H	0x293	T1CCR1[15:8]								0000 0000
TIM1CCR1L	0x294	T1CCR1[7:0]								0000 0000
TIM1CCR2H	0x295	T1CCR2[15:8]								0000 0000
TIM1CCR2L	0x296	T1CCR2[7:0]								0000 0000
TIM1CCR3H	0x297	T1CCR3[15:8]								0000 0000
TIM1CCR3L	0x298	T1CCR3[7:0]								0000 0000
TIM1CCR4H	0x299	T1CCR4[15:8]								0000 0000
TIM1CCR4L	0x29A	T1CCR4[7:0]								0000 0000
TIM1BKR	0x29B	T1MOE	T1AOE	T1BKP	T1BKE	T1OSSR	T1OSSI	T1LOCK[1:0]		0000 0000
TIM1DTR	0x29C	T1DTG[7:0]								0000 0000
TIM1OISR	0x29D	—	T1OIS4	T1OIS3N	T1OIS3	T1OIS2N	T1OIS2	T1OIS1N	T1OIS1	-000 0000
LEBCON	0x41C	LEBEN	LEBCH[1:0]		—	EDGS	BKS[2:0]			000- 0000

表 7-5 Timer1 相关用户寄存器汇总(–保留位必须保持为复位值，不可更改)

名称	状态		寄存器	地址	复位值
T1CNT	TIM1 计数值	高 8 位	TIM1CNTRH[7:0]	0x28C	RW-0000 0000
		低 8 位	TIM1CNTRL[7:0]	0x28D	RW-0000 0000
T1PSC	TIM1 预分频器	高 8 位	TIM1PSCRH[7:0]	0x28E	RW-0000 0000
		低 8 位	TIM1PSCRL[7:0]	0x28F	RW-0000 0000
T1ARR	计数周期的自动重载寄存器(预装载值) 注：此值为 0 时，计数器不工作；	高 8 位	TIM1ARRH[7:0]	0x290	RW-1111 1111
		低 8 位	TIM1ARRL[7:0]	0x291	RW-1111 1111
T1REP	重复向下计数值		TIM1RCR[7:0]	0x292	RW-0000 0000
T1CCR1	输入捕获模式：上一次捕获事件 (IC1) 捕获的计数值	高 8 位	TIM1CCR1H[7:0]	0x293	RO-0000 0000
		低 8 位	TIM1CCR1L[7:0]	0x294	RO-0000 0000
	输出比较模式：TIM1_CH1 的输出比较值 (预装载值)	高 8 位	TIM1CCR1H[7:0]	0x293	RW-0000 0000
		低 8 位	TIM1CCR1L[7:0]	0x294	RW-0000 0000
T1CCR2	输入捕获模式：上一次捕获事件 (IC2) 捕获的计数值	高 8 位	TIM1CCR2H[7:0]	0x295	RO-0000 0000
		低 8 位	TIM1CCR2L[7:0]	0x296	RO-0000 0000
	输出比较模式：TIM1_CH2 的输出比较值 (预装载值)	高 8 位	TIM1CCR2H[7:0]	0x295	RW-0000 0000
		低 8 位	TIM1CCR2L[7:0]	0x296	RW-0000 0000
T1CCR3	输入捕获模式：上一次捕获事件 (IC3) 捕获的计数值	高 8 位	TIM1CCR3H[7:0]	0x297	RO-0000 0000
		低 8 位	TIM1CCR3L[7:0]	0x298	RO-0000 0000
	输出比较模式：TIM1_CH3 的输出比较值 (预装载值)	高 8 位	TIM1CCR3H[7:0]	0x297	RW-0000 0000
		低 8 位	TIM1CCR3L[7:0]	0x298	RW-0000 0000
T1CCR4	输入捕获模式：上一次捕获事件 (IC4) 捕获的计数值	高 8 位	TIM1CCR4H[7:0]	0x299	RO-0000 0000
		低 8 位	TIM1CCR4L[7:0]	0x29A	RO-0000 0000
	输出比较模式：TIM1_CH4 的输出比较值 (预装载值)	高 8 位	TIM1CCR4H[7:0]	0x299	RW-0000 0000
		低 8 位	TIM1CCR4L[7:0]	0x29A	RW-0000 0000

表 7-6 Timer1 周期相关寄存器

名称	状态		寄存器	地址	复位值
TIM1EN	TIM1 模块时钟	1 = 使能 0 = 关闭	PCKEN[1]	0x9A	RW-0
SYSON	睡眠模式下，系统时钟控制	1 = 使能 0 = 关闭	CKOCON[7]	0x95	RW-0
T1CKSRC	TIM1 时钟源 (Fmaster) 000 = Sysclk　100 = 2 x (XT or EC) (*) 001 = HIRC　101 = LIRC 010 = XT or EC (*)　110 = LP or EC (*) 011 = 2 x HIRC　111 = 2 x (LP or EC) (*) (*) FOSC 需相应配置成 LP/XT/EC 模式，否则振荡器将不会运行。		TCKSRC[2:0]	0x31F	RW-000
DTYSEL	TIM1/TIM2 倍频时钟占空比调节位 00 = 2ns 延迟　10 = 4ns 延迟 01 = 3ns 延迟　11 = 7ns 延迟		CKOCON[5:4]	0x95	RW-10
T1ARPE	计数周期的自动预装载 1 = 使能 (T1ARR 预装载值在更新事件到来时被加载) 0 = 禁止 (T1ARR 立即被加载)		TIM1CR1[7]	0x211	RW-0
T1CMS	计数器对齐模式 00 = 边沿对齐模式 (计数方向由 T1DIR 决定) 01 = 中央对齐模式 1 (向下计数时 T1CCxIF 置 1) 10 = 中央对齐模式 2 (向上计数时 T1CCxIF 置 1) 11 = 中央对齐模式 3 (向上和向下计数时 T1CCxIF 均置 1) 注： 1. 中央对齐模式为计数器交替地向上和向下计数。 2. 仅允许在计数器关闭(T1CEN=0)时切换模式。		TIM1CR1[6:5]		RW-00
T1DIR	计数器计数方向 (中央对齐/编码器模式下，该位只读) 1 = 向下 0 = 向上		TIM1CR1[4]		RW-0
T1OPM	单脉冲模式 1 = 使能 (下一次更新事件到来时，T1CEN 自动清零，计数器停止) 0 = 关闭 (发生更新事件时，计数器不停止)		TIM1CR1[3]		RW-0

名称	状态	寄存器	地址	复位值
T1URS	当 T1UDIS = 0 时，更新事件源 1 = 计数器上溢/下溢 0 = 计数器上溢/下溢、软件设置 T1UG 位、或复位触发事件	TIM1CR1[2]		RW-0
T1UDIS	产生更新事件控制 1 = 禁止 0 = 允许	TIM1CR1[1]		RW-0
T1CEN	TIM1 计数器 1 = 使能 0 = 关闭	TIM1CR1[0]		RW-0
T1ENCTRL	TIM1/TIM2 同步触发控制 1 = TIM1/TIM2 同时由 T1CEN 使能 0 = TIM2 由 T2CEN 使能	TIM1CR2[7]		RW-0
T1COMS	当 T1CCPC = 1 时，换相事件源 1 = T1COMG 由软件置 1，或产生触发事件 0 = T1COMG 由软件置 1 注：仅对带互补输出的通道有效；	TIM1CR2[2]	0x212	RW-0
T1CCPC	捕获/比较控制位的自动预装载 (控制位：T1SMOD, T1GP, T1CCxE, T1CCxNE, T1CCxP, T1CCxNP, T1OCxM) 1 = 使能 (控制位的预装载值在换相事件到来时被加载) 0 = 禁止 (控制位立即被加载) 注：仅对带互补输出的通道有效；	TIM1CR2[0]		RW-0
T1TS	同步计数器的触发输入源 (TRGI) 0xx = 保留 100 = 通道 1 输入 TI1 的边沿检测器 (TI1F_ED) 101 = 滤波后的通道 1 输入 (TI1FP1) 110 = 滤波后的通道 2 输入 (TI2FP2) 111 = 外部触发输入 (ETRF) 注： 1. 仅允许在 T1SMS = 000 时更改触发输入源； 2. 触发输入有效沿/有效电平的极性参阅 T1CC1P/T1CC2P 或 T1ETP；	TIM1SMCR[6:4]	0x213	RW-000

名称	状态	寄存器	地址	复位值
T1SMS	触发模式 000 = 内部时钟 001 = 编码器模式 1 (根据 TI1FP1 电平，计数器在 TI2FP2 的有效沿向上/下计数) 010 = 编码器模式 2 (根据 TI2FP2 电平，计数器在 TI1FP1 的有效沿向上/下计数) 011 = 编码器模式 3 (根据 TI1FP1/TI2FP2 的电平，计数器在 TI2FP2/TI1FP1 的有效沿向上/下计数) 100 = 复位模式 (在触发输入有效沿时，计数器清零并从 0 开始重新计数) 101 = 门控模式 (计数器在触发输入有效电平期间计数，无效电平则停止计数，但不复位) 110 = 触发模式 (计数器在触发输入有效沿时计数且不复位) 111 = 外部时钟模式 1 (计数器在触发输入有效沿时计数) 注：门控模式的触发输入不能选择 TI1F_ED；	TIM1SMCR[2:0]		RW-000
T1ETP	外部触发极性 1 = ETR 反相，即低电平或下降沿有效 0 = ETR 不反相，即高电平或上升沿有效	TIM1ETR[7]	0x214	RW-0
T1ECE	外部时钟模式 2 1 = 使能 (计数器在外部触发输入 ETR 有效沿时计数) 0 = 关闭 注： 1. 当外部时钟模式 1 选择外部触发输入 ETR 时，与外部时钟模式 2 的功能相同； 2. 外部时钟模式 2，与外部时钟模式 1 和编码器模式不能同时使用；外部时钟模式 2 比外部时钟模式 1 的优先级更高； 3. 外部时钟模式 2 可与触发/复位/门控模式同时使用，但此时其他模式的触发输入源不能选择 ETRF；	TIM1ETR[6]		RW-0

<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>T1ETPS[1:0]</td><td>外部触发预分频比
00 = 1 10 = 4
01 = 2 11 = 8
注：外部触发信号的频率需满足 ≤ Fmaster/4</td><td>TIM1ETR[5:4]</td><td rowspan="3">0x21E</td><td>RW−00</td></tr>
<tr><td rowspan="2">T1ETF[3:0]</td><td>外部触发滤波器采样频率及数字滤波器长度</td><td rowspan="2">TIM1ETR[3:0]</td><td rowspan="2">RW-0000</td></tr>
<tr><td>
<table>
<tr><th>Value</th><th>采样频率 ($f_{SAMPLING}$)</th><th>数字滤波器长度 N</th></tr>
<tr><td>0000</td><td>Fmaster</td><td>0</td></tr>
<tr><td>0001</td><td>Fmaster</td><td>2</td></tr>
<tr><td>0010</td><td>Fmaster</td><td>4</td></tr>
<tr><td>0011</td><td>Fmaster</td><td>8</td></tr>
<tr><td>0100</td><td>Fmaster / 2</td><td>6</td></tr>
<tr><td>0101</td><td>Fmaster / 2</td><td>8</td></tr>
<tr><td>0110</td><td>Fmaster / 4</td><td>6</td></tr>
<tr><td>0111</td><td>Fmaster / 4</td><td>8</td></tr>
<tr><td>1000</td><td>Fmaster / 8</td><td>6</td></tr>
<tr><td>1001</td><td>Fmaster / 8</td><td>8</td></tr>
<tr><td>1010</td><td>Fmaster / 16</td><td>5</td></tr>
<tr><td>1011</td><td>Fmaster / 16</td><td>6</td></tr>
<tr><td>1100</td><td>Fmaster / 16</td><td>8</td></tr>
<tr><td>1101</td><td>Fmaster / 32</td><td>5</td></tr>
<tr><td>1110</td><td>Fmaster / 32</td><td>6</td></tr>
<tr><td>1111</td><td>Fmaster / 32</td><td>8</td></tr>
</table>
</td></tr>
<tr><td>T1GP</td><td>PWM 群组模式
1 = 使能 (通道 CH1/2/3 及 CH1N/2N/3N 的占空比由 T1CCR1 决定，原配置 T1CCR2 / T1CCR3 失效)
0 = 禁止</td><td>TIM1CCER2[7]</td><td>RW−0</td></tr>
<tr><td>T1SMOD</td><td>PWM 同步模式
1 = 使能 (通道 CH1N/2N/3N 与 CH1/2/3 的信号完全相同，且无死区功能)
0 = 禁止</td><td>TIM1CCER2[6]</td><td>RW−0</td></tr>
</table>

表 7-7 Timer1 相关用户控制寄存器

名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	Bit0	复位值
TIM1CCMR1	0x219	T1IC1F[3:0]				T1IC1PSC[1:0]		T1CC1S[1:0]		RW-0000 0000
TIM1CCMR2	0x21A	T1IC2F[3:0]				T1IC2PSC[1:0]		T1CC2S[1:0]		RW-0000 0000
TIM1CCMR3	0x21B	T1IC3F[3:0]				T1IC3PSC[1:0]		T1CC3S[1:0]		RW-0000 0000
TIM1CCMR4	0x21C	T1IC4F[3:0]				T1IC4PSC[1:0]		T1CC4S[1:0]		RW-0000 0000

名称	状态			寄存器	地址	复位值
T1ICxF	通道 x 输入捕获的采样频率和数字滤波器长度			TIM1CCMRx[7:4] x = 1, 2, 3, 4	0x219/ 0x21A/ 0x21B/ 0x21C	RW−0000
	Value	采样频率 ($f_{SAMPLING}$)	数字滤波器长度 N			
	0000	Fmaster	0			
	0001	Fmaster	2			
	0010	Fmaster	4			
	0011	Fmaster	8			
	0100	Fmaster / 2	6			
	0101	Fmaster / 2	8			
	0110	Fmaster / 4	6			
	0111	Fmaster / 4	8			
	1000	Fmaster / 8	6			
	1001	Fmaster / 8	8			
	1010	Fmaster / 16	5			
	1011	Fmaster / 16	6			
	1100	Fmaster / 16	8			
	1101	Fmaster / 32	5			
	1110	Fmaster / 32	6			
	1111	Fmaster / 32	8			
T1ICxPSC	通道 x 输入捕获预分频器（几个事件触发一次捕获） 00 = 1 个　　10 = 4 个 01 = 2 个　　11 = 8 个 注：当 T1CCxE = 0 时，该预分频器复位为 00			TIM1CCMRx[3:2]		RW−00
T1CC1S [1]	通道 1 模式选择	00 = 输出 01 = 输入，输入脚映射在 TI1FP1 10 = 输入，输入脚映射在 TI2FP1 11 = 输入，输入脚映射在 TRC		TIM1CCMR1[1:0]	0x219	RW−00

[1] 仅在通道 x 关闭（T1CCxE = 0 和 T1CCxNE = 0）时可写。

T1CC2S [2]	通道 2 模式选择	00 = 输出 01 = 输入，输入脚映射在 TI2FP2 10 = 输入，输入脚映射在 TI1FP2 11 = 输入，输入脚映射在 TRC	TIM1CCMR2[1:0]	0x21A	RW-00
T1CC3S [2]	通道 3 模式选择	00 = 输出 01 = 输入，输入脚映射在 TI3FP3 10 = 输入，输入脚映射在 TI4FP3 11 = 保留	TIM1CCMR3[1:0]	0x21B	RW-00
T1CC4S [2]	通道 4 模式选择	00 = 输出 01 = 输入，输入脚映射在 TI3FP4 10 = 输入，输入脚映射在 TI4FP4 11 = 保留	TIM1CCMR4[1:0]	0x21C	RW-00

表 7-8 TIM1CCMRx 作为输入配置寄存器

名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	Bit0	复位值
TIM1CCMR1	0x219	T1OC1CE	T1OC1M[2:0]			T1OC1PE	T1OC1FE	T1CC1S[1:0]		RW-0000 0000
TIM1CCMR2	0x21A	T1OC2CE	T1OC2M[2:0]			T1OC2PE	T1OC2FE	T1CC2S[1:0]		RW-0000 0000
TIM1CCMR3	0x21B	T1OC3CE	T1OC3M[2:0]			T1OC3PE	T1OC3FE	T1CC3S[1:0]		RW-0000 0000
TIM1CCMR4	0x21C	T1OC4CE	T1OC4M[2:0]			T1OC4PE	T1OC4FE	T1CC4S[1:0]		RW-0000 0000

T1OCxM	通道 x 输出比较模式		参考信号 OCxREF 的电平值
000	冻结（不比较）		保持不变
001	当 T1CNT = CCRx_SHAD 时		1
010	当 T1CNT = CCRx_SHAD 时		0
011	当 T1CNT = CCRx_SHAD 时		电平翻转
100	强制无效		0
101	强制有效		1
110	PWM1 模式	T1CNT < CCRx_SHAD	1
		T1CNT > CCRx_SHAD	0
111	PWM2 模式	T1CNT < CCRx_SHAD	0
		T1CNT > CCRx_SHAD	1
注：输出参考信号 OCxREF 为高电平有效，其与极性选择 T1CCxP 共同决定引脚 OCx 的实际输出值；			

表 7-9 T1OCxM 配置为输出比较模式

[2] 仅在通道 x 关闭（T1CCxE = 0 和 T1CCxNE = 0）时可写。

名称	状态		寄存器	地址	复位值
T1OCxCE	通道 x 输出参考信号 OCxREF 由 ETR 清零控制 1 = 使能 (当 ETR 为有效电平时，OCxREF 清零) 0 = 禁止 (OCxREF 不受 ETR 输入的影响)		TIM1CCMRx[7] x = 1, 2, 3, 4	0x219/ 0x21A/ 0x21B/ 0x21C	RW−0
T1OCxPE	通道 x 输出比较值的自动预装载 1 = 使能 (T1CCRx 预装载值在更新事件到来时被加载) 0 = 禁止 (T1CCRx 立即被加载) 注：PWM 模式下必须使能，单脉冲模式可选		TIM1CCMRx[3] x = 1, 2, 3, 4		RW−0
T1OCxFE	通道 x 输出信号快速使能 1 = 使能 (输入触发有效沿到通道 x 输出的延时 ≥3*$T_{Fmaster}$，且通道 x 输出为固定值，输出值 =T1CEN (PWM1 模式) / !T1CEN(PWM2 模式)) 0 = 禁止 (输入触发有效沿到通道 x 输出的延时 ≥5*$T_{Fmaster}$) 注：该位只在通道 1 被配置成 PWM1/PWM2 模式时起作用		TIM1CCMRx[2] x = 1, 2, 3, 4		RW−0
T1CC1S [3]	通道 1 模式选择	00 = 输出 01 = 输入，输入脚映射在 TI1FP1 10 = 输入，输入脚映射在 TI2FP1 11 = 输入，输入脚映射在 TRC	TIM1CCMR1[1:0]	0x219	RW−00
T1CC2S [3]	通道 2 模式选择	00 = 输出 01 = 输入，输入脚映射在 TI2FP2 10 = 输入，输入脚映射在 TI1FP2 11 = 输入，输入脚映射在 TRC	TIM1CCMR2[1:0]	0x21A	RW−00
T1CC3S [3]	通道 3 模式选择	00 = 输出 01 = 输入，输入脚映射在 TI3FP3 10 = 输入，输入脚映射在 TI4FP3 11 = 保留	TIM1CCMR3[1:0]	0x21B	RW−00
T1CC4S [3]	通道 4 模式选择	00 = 输出 01 = 输入，输入脚映射在 TI3FP4 10 = 输入，输入脚映射在 TI4FP4 11 = 保留	TIM1CCMR4[1:0]	0x21C	RW−00

表 7-10 TIM1CCMRx 作为输出配置寄存器

[3] 仅在通道 x 关闭 (T1CCxE = 0 和 T1CCxNE = 0) 时可写。

名称	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	地址	复位值
TIM1CCER1	T1CC2NP	T1CC2NE	T1CC2P	T1CC2E	T1CC1NP	T1CC1NE	T1CC1P	T1CC1E	0x21D	RW-0000 0000
TIM1CCER2	T1GP	T1SMOD	T1CC4P	T1CC4E	T1CC3NP	T1CC3NE	T1CC3P	T1CC3E	0x21E	RW-0000 0000

名称	功能	输入捕获/触发模式 (T1CCxS ≠ 00)	输出比较模式 (T1CCxS = 00)
T1CCxP	通道x输入/输出极性选择	1 = 捕获/触发发生在 TIxF 的下降沿或低电平 0 = 捕获/触发发生在 TIxF 的上升沿或高电平 注：输入触发源仅可选通道 1 和 2	1 = OCx 低电平有效 0 = OCx 高电平有效
T1CCxE	通道x输入/输出引脚功能	1 = 使能引脚的输入捕获/触发功能 0 = 禁止	1 = 使能引脚的 OCx 输出功能 0 = 禁止
T1CCxNP	通道 x 互补输出极性选择	–	1 = OCxN 低电平有效 0 = OCxN 高电平有效
T1CCxNE	通道 x 互补输出引脚功能	–	1 = 使能引脚的 OCxN 输出功能 0 = 关闭
注：通道输出电平由 T1MOE, T1OSSI, T1OSSR, T1OISx, T1OISxN, T1CCxE 和 T1CCxNE 位的值共同决定，参阅**表 7-14**。			

表 7-11 Timer1 通道输出和极性选择

名称	状态	寄存器	地址	复位值
TIM1_CH1N	通道 1 互补引脚重映射 1 = PB2 0 = PC0	AFP0[2]	0x19E	RW-0
TIM1_CH2	通道 2 引脚重映射 1 = PB0 0 = PA1	AFP0[5]		RW-0
TIM1_CH4	通道 4 引脚重映射 0x = PB1 10 = PA7 11 = PA2	AFP1[1:0]	0x19F	RW-00

表 7-12 Timer1 管脚功能重映射寄存器

<table>
<tr><th>名称</th><th colspan="4">控制</th><th>寄存器</th><th>地址</th><th>复位值</th></tr>
<tr><td>BKS</td><td colspan="4">TIM1 的故障刹车源
000 = 关闭
001 = BKIN 管脚
010 = LVD 检测
100 = ADC 阈值比较</td><td>LEBCON[2:0]</td><td>0x41C</td><td>RW-000</td></tr>
<tr><td>T1MOE [4]</td><td colspan="4">主输出控制 (仅对配置为输出的通道有效)
1 = 使能 (若 T1CCxE/T1CCxNE = 1，则使能 OCx 和 OCxN 输出)
0 = 禁止 (禁止 OCx 和 OCxN 输出或强制为空闲状态)</td><td>TIM1BKR[7]</td><td rowspan="8">0x29B</td><td>RW-0</td></tr>
<tr><td>T1AOE</td><td colspan="4">主输出自动控制
1 = T1MOE 在下一个更新事件到来时被自动置 1(当刹车输入无效时) 或由软件置 1
0 = T1MOE 只能由软件置 1</td><td>TIM1BKR[6]</td><td>RW-0</td></tr>
<tr><td>T1BKP</td><td colspan="4">故障源 TIM1_BKIN 刹车输入极性
1 = 高电平有效
0 = 低电平有效</td><td>TIM1BKR[5]</td><td>RW-0</td></tr>
<tr><td>T1BKE</td><td colspan="4">刹车输入(BRK)功能
1 = 使能
0 = 禁止</td><td>TIM1BKR[4]</td><td>RW-0</td></tr>
<tr><td>T1OSSR</td><td colspan="4">运行模式下(当 T1MOE = 1 时)输出“关闭状态”选择
详情请见表 7-14 Timer1 输出控制和状态”</td><td>TIM1BKR[3]</td><td>RW-0</td></tr>
<tr><td>T1OSSI</td><td colspan="4">空闲模式下(当 T1MOE=0 时)输出“关闭状态”选择
详情请见表 7-14 Timer1 输出控制和状态”</td><td>TIM1BKR[2]</td><td>RW-0</td></tr>
<tr><td rowspan="4">T1LOCK [5]</td><td colspan="4">锁定设置 (写保护，防止软件错误)</td><td rowspan="4">TIM1BKR[1:0]</td><td rowspan="4">RW-00</td></tr>
<tr><td>00</td><td>01</td><td>10</td><td>11</td></tr>
<tr><td>关闭</td><td>锁定级别 1</td><td>锁定级别 2</td><td>锁定级别 3</td></tr>
<tr><td>寄存器无写保护</td><td>T1BKE, T1BKP, T1AOE, T1OISx, T1OISxN, T1DTG</td><td>包含级别 1, T1CCxP, T1CCxNP, T1OSSR, T1OSSI</td><td>包含级别 2, T1OCxM, T1OCxPE</td></tr>
</table>

表 7-13 Timer1 主输出使能、刹车和锁定级别寄存器

[4] 当刹车输入有效时，该位将被硬件异步清 0。

[5] 系统复位后只能写一次 LOCK 位，一旦写入则其内容将保持不变直至复位。

<table>
<tr><th colspan="5">控制位</th><th colspan="2">输出状态</th></tr>
<tr><th>T1MOE</th><th>T1OSSI</th><th>T1OSSR</th><th>T1CCxE</th><th>T1CCxNE</th><th>OCx 输出状态</th><th>OCxN 输出状态</th></tr>
<tr><td rowspan="8">1</td><td rowspan="8">x</td><td>0</td><td>0</td><td>0</td><td>OCx = 0 (输出关闭)</td><td>OCxN = 0 (输出关闭)</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>OCx = 0 (输出关闭)</td><td>OCxN = OCxREF ^ T1CCxNP</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>OCx = OCxREF ^ T1CCxP</td><td>OCxN = 0 (输出关闭)</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>OCx = OCxREF ^ T1CCxP + 死区</td><td>OCxN = OCxREF 的互补信号 ^ T1CCxNP + 死区</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>OCx = T1CCxP (输出关闭)</td><td>OCxN =T1CCxNP (输出关闭)</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>OCx = T1CCxP</td><td>OCxN = OCxREF ^ T1CCxNP</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>OCx = OCxREF ^ T1CCxP</td><td>OCxN =T1CCxNP</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>OCx = OCxREF ^ T1CCxP + 死区</td><td>OCxN = OCxREF 的互补信号 ^ T1CCxNP + 死区</td></tr>
<tr><td rowspan="8">0</td><td>0</td><td rowspan="8">x</td><td>0</td><td>0</td><td>OCx = T1CCxP (输出关闭)</td><td>OCxN = T1CCxNP (输出关闭)</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td colspan="2" rowspan="3">死区时间内：OCx = T1CCxP, OCxN = T1CCxNP (输出关闭)
死区时间后：OCx = T1OISx, OCxN = T1OISxN (输出关闭)</td></tr>
<tr><td>0</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>OCx = T1CCxP (输出关闭)</td><td>OCxN = T1CCxNP (输出关闭)</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td colspan="2" rowspan="3">死区时间内：OCx = T1CCxP, OCxN = T1CCxNP (输出无效值)
死区时间后：OCx = T1OISx, OCxN = T1OISxN</td></tr>
<tr><td>1</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td>1</td></tr>
</table>

表 7-14 Timer1 输出控制和状态

名称	控制			寄存器	地址	复位值
T1DTG	死区发生器设置			TIM1DTR[7:0]	0x29C	RW−0000 0000
	T1DTG[7:0]	DT(死区持续时间)	t_{DTG}			
	0xxxxxxx	T1DTG[7:0] x t_{DTG}	$T_{Fmaster}$ (f1)			
	10xxxxxx	(64+T1DTG[5:0]) x t_{DTG}	2 x $T_{Fmaster}$ (f2)			
	110xxxxx	(32+T1DTG[4:0]) x t_{DTG}	8 x $T_{Fmaster}$ (f3)			
	111xxxxx	(32+T1DTG[4:0]) x t_{DTG}	16 x $T_{Fmaster}$ (f4)			
	* Fmaster 为 TIM1 时钟源 例如当 $T_{Fmaster}$ = 125 ns (8 MHz) 时，死区时间如下：					
	T1DTG[7:0]	死区时间 (μs)	步长时间			
	0 ~ 7Fh	0 ~ 15.875	125 ns (f1)			
	80h ~ BFh	16 ~ 31.75	250 ns (f2)			
	C0h ~ DFh	32 ~ 63	1 μs (f3)			
	E0h ~ FFh	64 ~ 126	2 μs (f4)			

表 7-15 Timer1 互补输出死区时间配置

名称	状态	寄存器	地址	复位值
T1OIS4	当 T1MOE=0 时，通道 4(OC4) 空闲状态输出 1 = OC4 输出 1 0 = OC4 输出 0	TIM1OISR[6]	0x29D	RW−0
T1OIS3	当 T1MOE=0 时，通道 3/2/1(OCx) 空闲状态输出	TIM1OISR[4]		RW−0
T1OIS2	1 = 死区时间后，OCx 输出 1	TIM1OISR[2]		RW−0
T1OIS1	0 = 死区时间后，OCx 输出 0	TIM1OISR[0]		RW−0
T1OIS3N	当 T1MOE=0 时，互补通道 3/2/1(OCxN) 空闲状态输出	TIM1OISR[5]		RW−0
T1OIS2N	1 = 死区时间后，OCxN 输出 1	TIM1OISR[3]		RW−0
T1OIS1N	0 = 死区时间后，OCxN 输出 0	TIM1OISR[1]		RW−0

表 7-16 Timer1 通道输出空闲状态寄存器

名称	状态		寄存器	地址	复位值
GIE	全局中断 1 = 使能 (PEIE, T1BIE, T1BG, T1TIE, T1TG, T1COMIE, T1COMG, T1CCxIE, T1CCxG, T1UIE 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 (T1BIE, T1BG, T1TIE, T1TG, T1COMIE, T1COMG, T1CCxIE, T1CCxG, T1UIE 适用) 0 = 关闭 (无唤醒)		INTCON[6]		RW-0
T1BIE	刹车中断	1 = 使能 0 = 关闭	TIM1IER[7]	0x215	RW-0
T1BG [6]	刹车软件中断		TIM1EGR[7]	0x218	WO-0
T1BIF [7]	刹车中断标志位 1 = 刹车输入上检测到有效电平 0 = 无刹车事件产生		TIM1SR1[7]	0x216	R_W1C-0
T1TIE	触发中断	1 = 使能 0 = 关闭	TIM1IER[6]	0x215	RW-0
T1TG [6]	触发软件中断		TIM1EGR[6]	0x218	WO-0
T1TIF [7]	触发中断标志位	1 = 已触发 0 = 无触发事件产生	TIM1SR1[6]	0x216	R_W1C-0
T1COMIE	换相(COM)中断	1 = 使能 0 = 关闭	TIM1IER[5]	0x215	RW-0
T1COMG [6]	换相(COM)软件中断		TIM1EGR[5]	0x218	WO-0
T1COMIF [7]	换相(COM)中断标志位	1 = 已换相 0 = 无换相事件产生	TIM1SR1[5]	0x216	R_W1C-0
T1CC4IE	通道 4 捕获/比较中断	1 = 使能 0 = 关闭	TIM1IER[4]	0x215	RW-0
T1CC3IE	通道 3 捕获/比较中断		TIM1IER[3]		RW-0
T1CC2IE	通道 2 捕获/比较中断		TIM1IER[2]		RW-0
T1CC1IE	通道 1 捕获/比较中断		TIM1IER[1]		RW-0
T1CC4G [6]	通道 4 捕获/比较软件中断		TIM1EGR[4]	0x218	WO-0
T1CC3G [6]	通道 3 捕获/比较软件中断		TIM1EGR[3]		WO-0
T1CC2G [6]	通道 2 捕获/比较软件中断		TIM1EGR[2]		WO-0
T1CC1G [6]	通道 1 捕获/比较软件中断		TIM1EGR[1]		WO-0

[6] 软件置 1，硬件自动清 0。
[7] 写 1 清零，写 0 无效。建议只使用 STR、MOVWI 指令进行写操作，而不要用 BSR 或 IOR 指令。

名称	状态		寄存器	地址	复位值
T1CC4IF [7]	通道 x 捕获/比较中断标志位 • 输出模式： 1 = T1CNT 与 T1CCRx 值匹配 0 = 不匹配 注：若 T1CCRx>T1ARR，则当 CNT 计数到 T1ARR 值时，T1CCxIF 置 1。 • 输入模式： 1 = 计数器值已被捕获至 T1CCRx (读 T1CCRx 时自动清零) 0 = 无捕获产生		TIM1SR1[4]	0x216	R_W1C-0
T1CC3IF [7]			TIM1SR1[3]		R_W1C-0
T1CC2IF [7]			TIM1SR1[2]		R_W1C-0
T1CC1IF [7]			TIM1SR1[1]		R_W1C-0
T1CC4OF [7]	通道 x 重复捕获中断标志位 1 = 发生重复捕获（计数器值被捕获到 T1CCRx 寄存器时，T1CCxIF 已经置 1) 0 = 无重复捕获		TIM1SR2[4]	0x217	R_W1C-0
T1CC3OF [7]			TIM1SR2[3]		R_W1C-0
T1CC2OF [7]			TIM1SR2[2]		R_W1C-0
T1CC1OF [7]			TIM1SR2[1]		R_W1C-0
T1UIE	允许更新中断	1 = 使能	TIM1IER[0]	0x215	RW-0
T1UG [6]	允许更新软件中断	0 = 关闭	TIM1EGR[0]	0x218	WO-0
T1UIF [7]	更新事件中断标志位	1 = 发生更新事件 0 = 无更新事件	TIM1SR1[0]	0x216	R_W1C-0

表 7-17 Timer1 中断使能和状态位

7.2.2. 计数基本单元

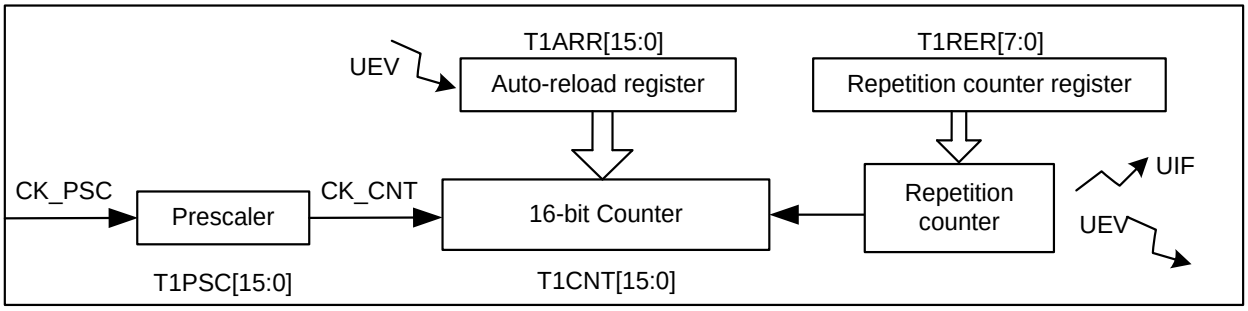


图 7-3 计数基本单元

TIM1 基本单元：

- 16 位向上、向下或上/下计数器
- 16 位预分频器
- 重复计数器
- 16 位自动重载寄存器

预分频器、重复计数器、输出比较值和自动重载寄存器分别由预装载寄存器和影子寄存器组成。

	预分频器	重复计数器	输出比较值	自动重载寄存器
预装载使能位	当 T1CEN = 1 时默认使能		T1OCxPE	T1ARPE
预装载寄存器	T1PSC[15:0]	T1REP[7:0]	T1CCRx[15:0]	T1ARR[15:0]

表 7-18 更新事件相关的预装载寄存器

预分频时钟(CK_PSC)源可选如下（参阅**章节 7.2.3** 时钟/触发控制器)：

- 内部时钟源(Fmaster)
- 外部时钟源(ETR)
- 滤波后的外部通道触发输入(TI1FP1, TI2FP2)

注：由于时钟同步的需求，外部异步时钟源(ETR)的时钟周期必须大于内部时钟源(Fmaster)时钟的 2 倍。

16 位预分频器可对预分频时钟(CK_PSC)进行 1 ~ 65536 分频，从而产生计数时钟(CK_CNT)。

分频公式：$f_{CK_CNT} = f_{CK_PSC} / (PSCR[15:0] + 1)$;　　(PSCR 为预分频器影子寄存器值)

当 T1UDIS = 0 时，允许产生更新事件，更新事件源（参阅"T1URS"）如下：

- 计数器上溢或下溢（当 T1REP = 0 时，请查阅**章节 7.2.2.2** 重复计数器)
- 软件设置 T1UG 位
- 复位模式下，产生触发事件

当产生更新事件时，更新事件标志位 T1UIF 置位，是否触发中断和/或从睡眠中唤醒则取决于相应的使能控制位(GIE, PEIE 和 T1UIE)。

此外，根据不同配置，更新事件可触发以下情况：

1. 与预分频器、重复计数器、输出比较值和自动重载寄存器相关：
   1.1 当计数器使能(T1CEN = 1)，且其相应的预装载使能时(T1OCxPE / T1ARPE =1，如**表 7-18**)，其影子寄存器将在产生更新事件时更新为预装载值，如**图 7-4**；
   1.2 当计数器关闭(T1CEN = 0)，或其相应的预装载关闭时(T1OCxPE / T1ARPE =0)，其影子寄存器将直接更新为预装载值；
2. 单次脉冲模式下，产生更新事件时，将自动关闭计数器(T1CEN = 0)，计数器停止计数；
3. 当故障刹车使能，且自动输出使能(T1AOE = 1)时，故障事件撤销后，PWM 将在产生更新事件后恢复正常输出；

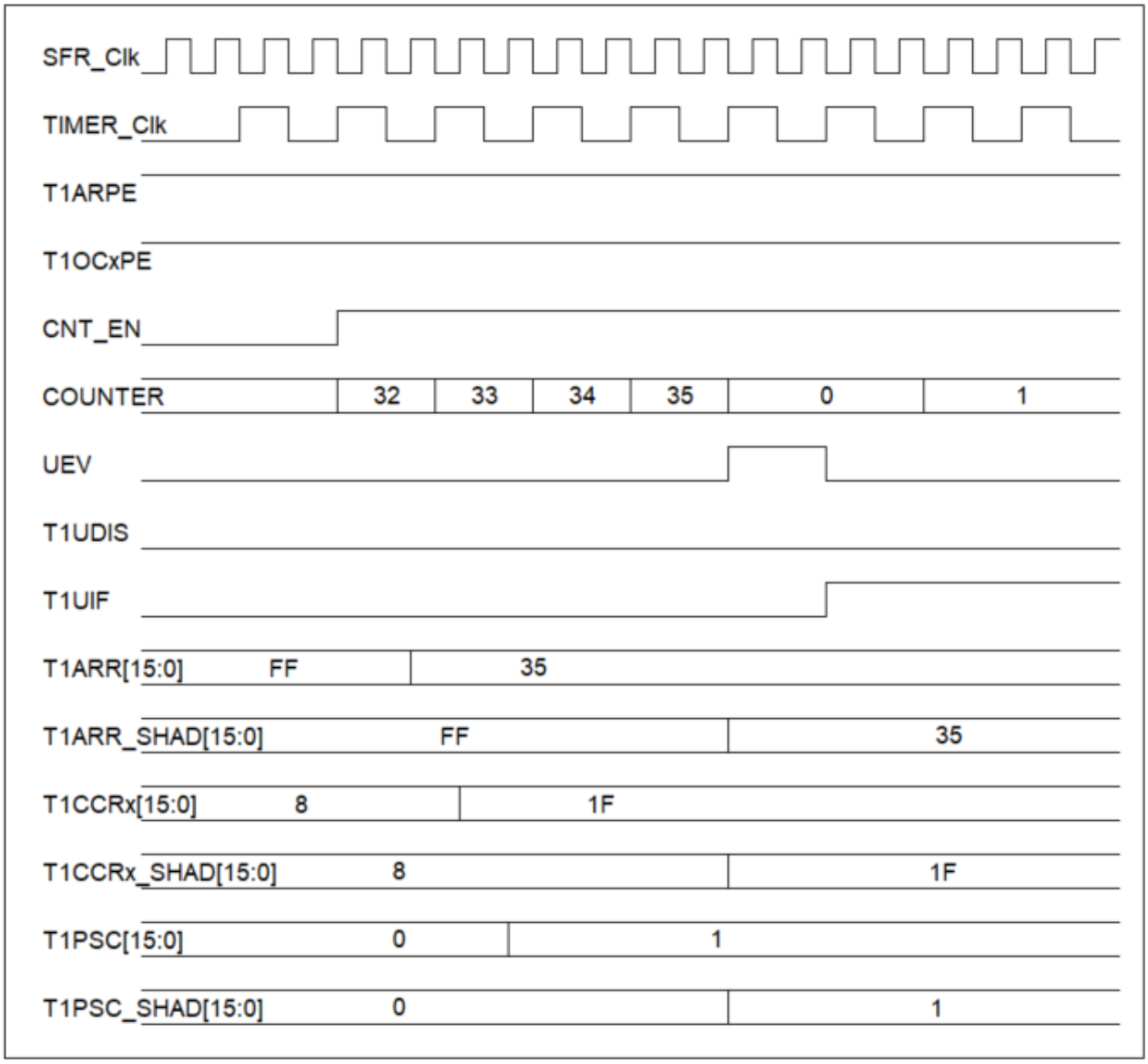


图 7-4 更新事件下，预装载寄存器的更新时序图

7.2.2.1. 计数模式

- **向上计数模式**(T1CMS = 00 且 T1DIR = 0)：计数器从 0 开始向上计数，当 T1CNT = T1ARR 时，产生上溢事件，然后重新从 0 开始计数。

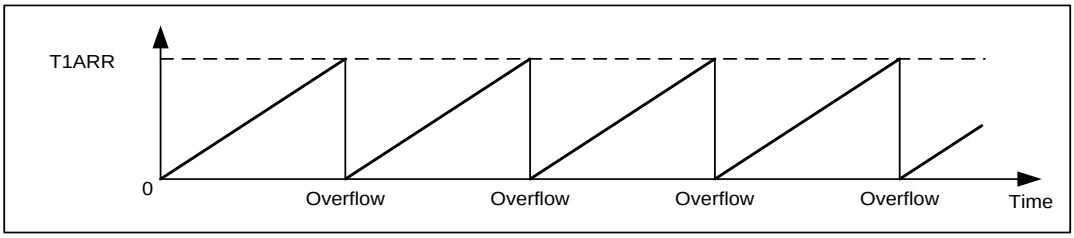


图 7-5 向上计数模式

- **向下计数模式**(T1CMS = 00 且 T1DIR = 1)：计数器从 T1ARR 值开始向下计数，当 T1CNT = 0 时，

产生下溢事件，然后重新从 T1ARR 值开始计数。

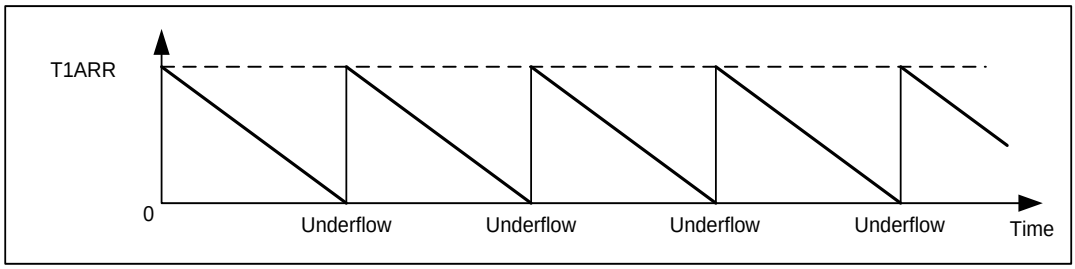


图 7-6 向下计数模式

- **中央对齐模式**(向上/向下计数，T1CMS ≠ 00)：计数初始方向取决于 T1DIR 寄存器值(如图 7-7 和图 7-8 所示)，计数初始值为 T1CNT。如果 T1DIR 初始化为 0，计数器从 T1CNT 开始向上计数，当 T1CNT = T1ARR 时，产生上溢事件；然后计数器从 T1ARR 值开始向下计数，当 T1CNT = 0 时，产生下溢事件；而后从 0 开始向上计数，计数器不断重复上述过程。

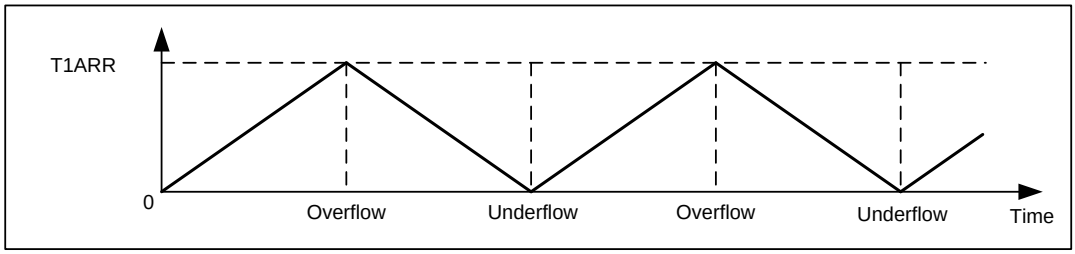


图 7-7 中央对齐模式 (T1DIR 初始化为 0)

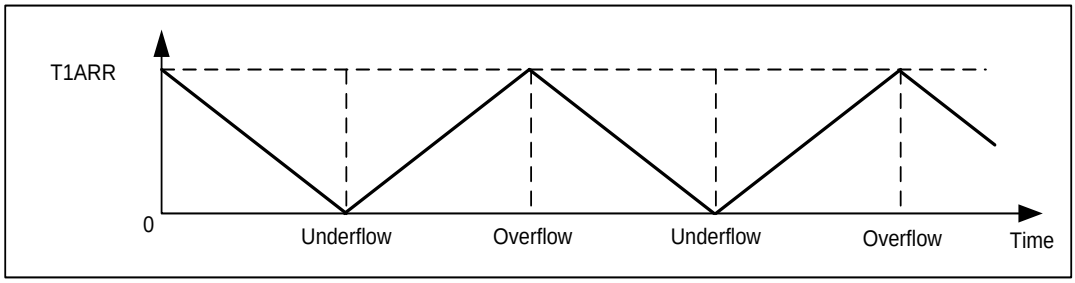


图 7-8 中央对齐模式 (T1DIR 初始化为 1)

配置步骤示例：

1. 使能 TIM1 模块时钟(TIM1EN = 1)，并选择 TIM1 时钟源(T1CKSRC)；
2. 如需要，可使能计数周期预加载功能(T1ARPE = 1)；
3. 配置计数周期值(T1ARR)；
4. 配置计数方向为向上或向下计数(T1DIR)；
5. 配置计数模式为边沿对齐模式或中央对齐模式(T1CMS)；
6. 配置预分频(T1PSC)；
7. 使能计数器(T1CEN = 1)；

注：

1. 建议在计数器停止(T1CEN = 0)时，再对计数器值 T1CNT[15:0] 进行读写操作，避免出错。
2. 软件不可同时改写 T1CMS 和 T1DIR 位；当 T1CMS = 00 时，T1DIR 为可读写寄存器；当 T1CMS ≠ 00 时，T1DIR 为只读寄存器，启动计数后由硬件自动设置计数方向。
3. 中央对齐模式下，需设置初始计数值 T1CNT ≤ T1ARR。
4. 需先配置周期、输出比较值、计数模式等寄存器，并在使能计数器(T1CEN = 1)之前配置预分频寄存器。

7.2.2.2. 重复计数器

当 8 位重复计数器的影子寄存器(RCR)不为 0 时，其将在以下事件发生时自动减 1：

- 向上计数模式下，每个计数上溢事件
- 向下计数模式下，每个计数下溢事件
- 中央对齐模式下，每个计数上溢或下溢事件

当重复计数器减到 0 时，计数器上溢或下溢事件才会产生更新事件(UEV)，即使用重复计数器能够设定更新事件的频率，另外在更新事件中断处理程序中可更改计数周期 T1ARR、占空比 T1CCRx 等配置，这在产生特定数量 PWM 信号时非常有用(具体请参阅 **章节 7.2.4.2** PWM 模式)。

当发生更新事件时，其影子寄存器(RCR)将自动更新为预装载 T1REP 值。

注：当配置 T1REP 且其值不为 0 时，建议在第一次更新事件之后再打开更新事件中断，在下次更新事件(UEV)发生时，其影子寄存器(RCR)才会重载为 T1REP 值。

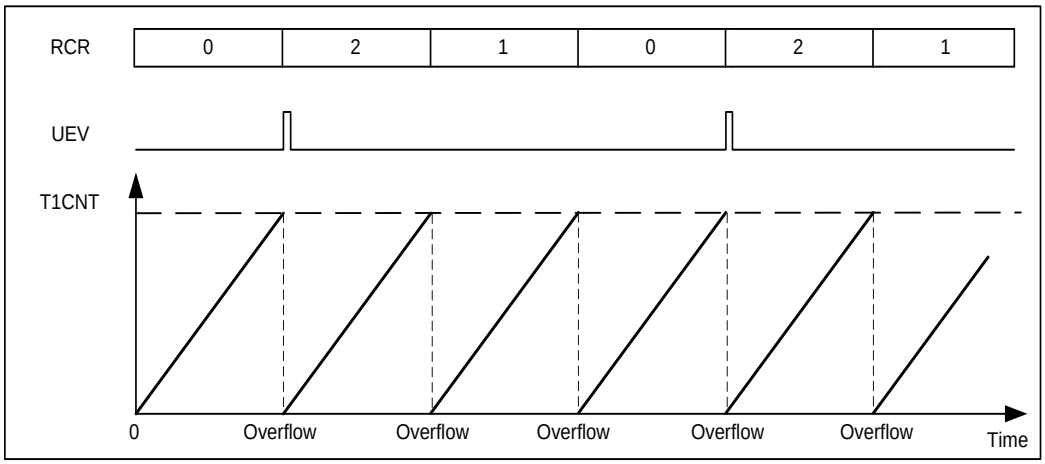


图 7-9 重复计数器计数时序图 (当 T1REP = 2 时)

7.2.3. 时钟/触发控制器

时钟/触发控制器包括计数器的时钟源、触发源和模式控制。

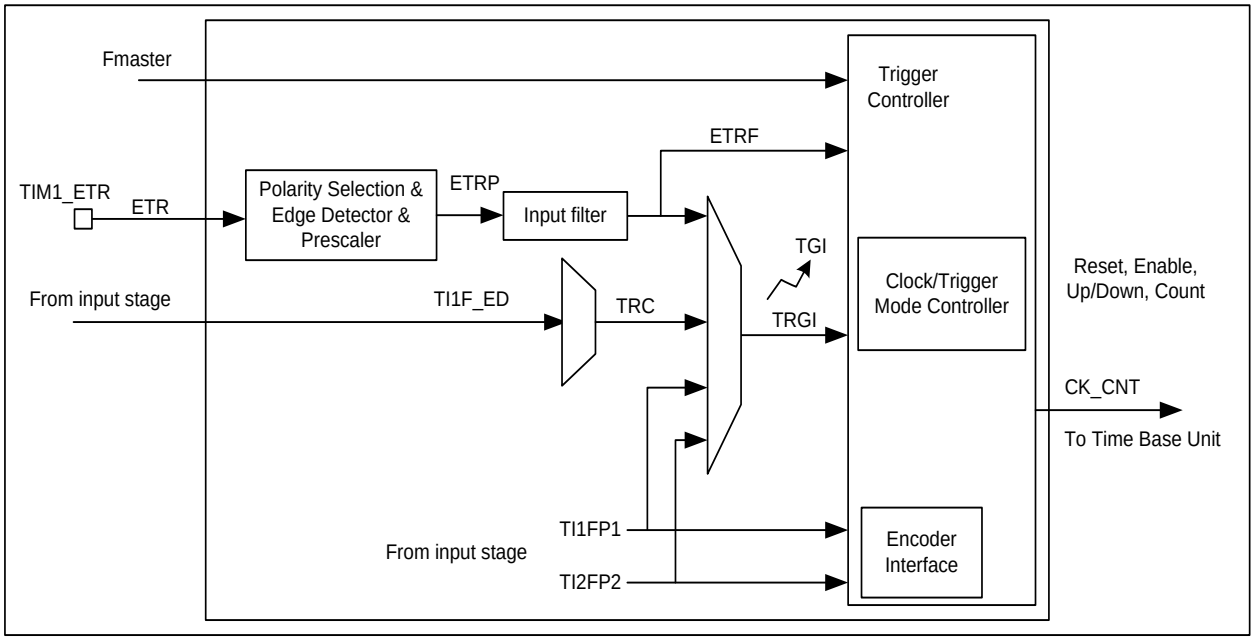


图 7-10 时钟/触发控制器框图

7.2.3.1. 计数器时钟源 (Fmaster)

当 T1SMS = 000 时，计数由内部时钟驱动，其可选的 6 个时钟源(参阅 T1CKSRC)如下：

- Sysclk
- 1x or 2x HIRC
- LIRC
- 1x or 2x 外部时钟 (只有当 FOSC 相应地配置成 LP, XT 或 EC 模式时才有效)

7.2.3.2. 计数器触发源

当 T1SMS ≠ 000 (Slave 模式) 时，计数由触发源(TRGI)驱动，其可选的 4 个触发事件源(参阅 T1TS) 如下：

- 通道 1 输入 TI1 的边沿检测器 (TI1F_ED)
- 滤波后的通道 1 输入 (TI1FP1)
- 滤波后的通道 2 输入 (TI2FP2)
- 外部触发输入 (ETRF)

注：

1. 触发输入有效沿/有效电平的极性参阅 T1CC1P/T1CC2P 或 T1ETP；
2. 当发生触发事件时，触发中断标志位 T1TIF 置位，是否触发中断和/或从睡眠中唤醒则取决于相应的使能控制位(GIE, PEIE 和 T1TIE)。

7.2.3.3. 计数控制模式

TIM1 计数控制模式(参阅 T1SMS 和 T1ECE)除内部时钟模式外，还可选择编码器模式 1/2/3、复位模式、

门控模式、触发模式和外部时钟模式 1/2。

- **内部时钟模式** (T1SMS = 000)：

  计数由内部时钟(Fmaster)驱动；

- **编码器模式**：
  - ✓ 编码器模式 1 (T1SMS = 001)：根据 TI1FP1 电平，计数器在 TI2FP2 的有效沿向上/下计数；
  - ✓ 编码器模式 2 (T1SMS = 010)：根据 TI2FP2 电平，计数器在 TI1FP1 的有效沿向上/下计数；
  - ✓ 编码器模式 3 (T1SMS = 011)：根据 TI1FP1/TI2FP2 的电平，计数器在 TI2FP2/TI1FP1 的有效沿向上/下计数；

编码器模式	有效计数沿	TI2FP2 电平值	TI1FP1 有效沿		TI1FP1 电平值	TI2FP2 有效沿	
			上升	下降		上升	下降
1	TI2FP2	—	—	—	高	向上	向下
		—	—	—	低	向下	向上
2	TI1FP1	高	向下	向上	—	—	—
		低	向上	向下	—	—	—
3	TI1FP1 或 TI2FP2	高	向下	向上	高	向上	向下
		低	向上	向下	低	向下	向上

表 7-19 编码器模式 (计数方向 vs. 译码信号，T1CCxP = 0)

编码器模式主要应用于电机应用。计数方向随着 TI1FP1 和 TI2FP2 信号的组合不断变化。

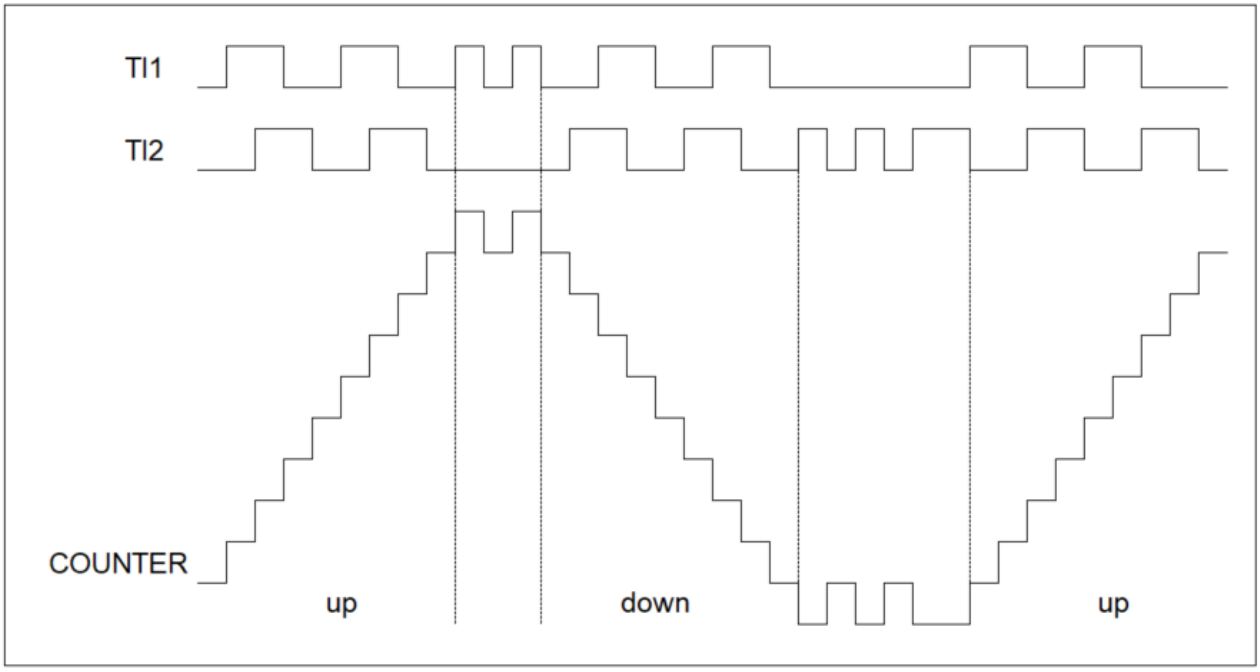


图 7-11 编码器模式 3，计数器时序图

编码器模式下的注意事项如下：

1. Timer1 的输入捕获和输出比较功能均能正常工作；
2. 预分频(T1PSC)功能无效；
3. 与外部时钟模式 2 不能同时使用；

- **复位模式** (T1SMS = 100)：

  计数器开始由内部时钟驱动正常计数，直到触发输入(TRGI)出现一个有效沿，此时计数器清零并从 0 开始重新计数。同时触发标志位 T1TIF 置位。另外当 T1UDIS = 0 且 T1URS= 0 时，则会产生一个更新事件。

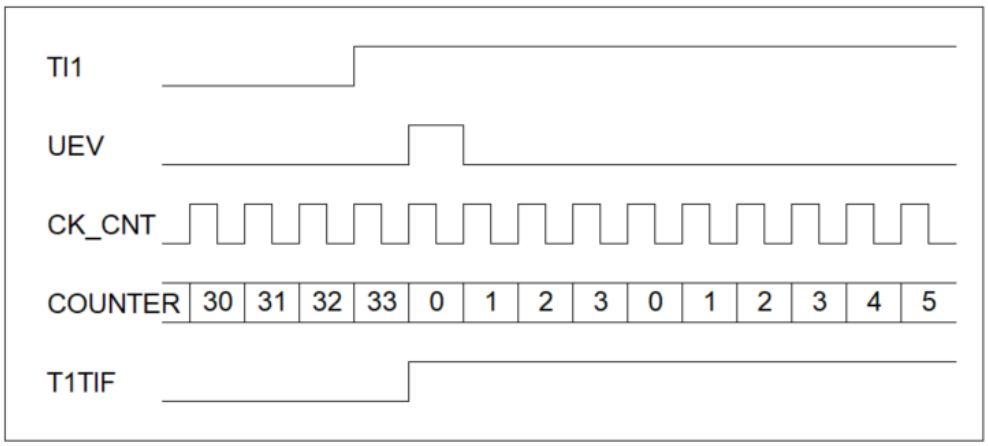


图 7-12 复位模式，计数器时序图 (选择通道 TI1 且触发输入有效沿为上升沿)

- **门控模式** (T1SMS = 101)：

  计数器在触发输入(TRGI)有效电平期间由内部时钟驱动计数，无效电平则停止计数，但不复位。触发标志位 T1TIF 在计数器启动或停止时均会置位。

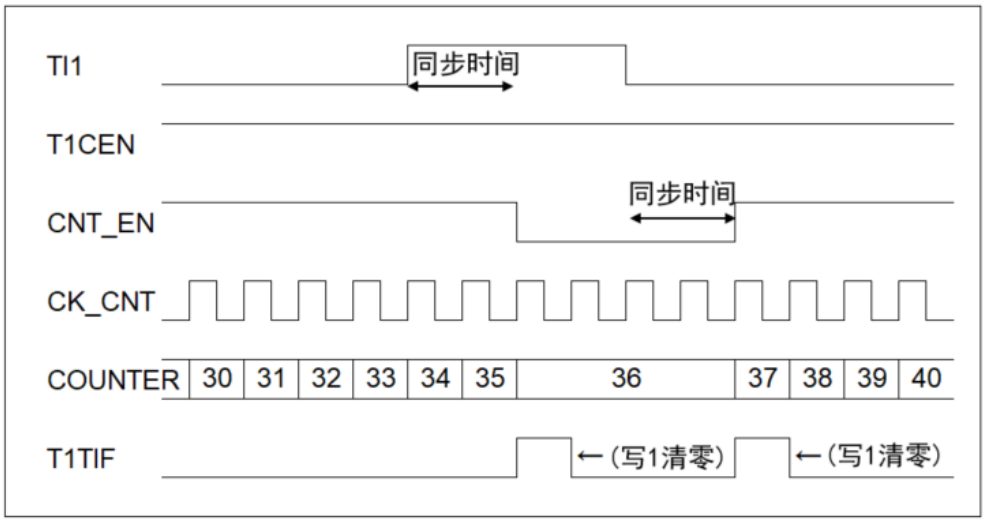


图 7-13 门控模式，计数器时序图 (选择通道 TI1 且触发输入有效电平为低电平)

- **触发模式** (T1SMS = 110)：

  计数器在触发输入(TRGI)有效沿时由内部时钟驱动计数，且不复位。同时触发标志位 T1TIF 置位。

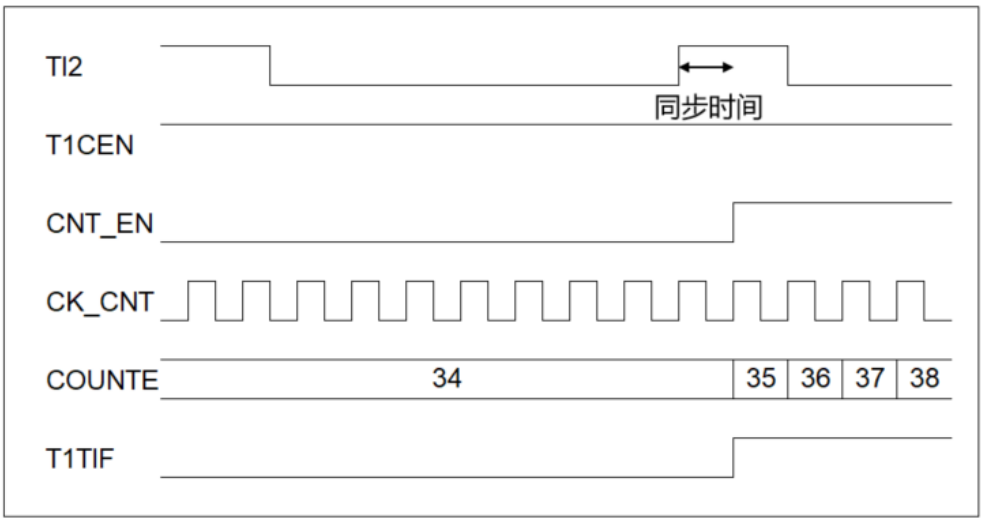


图 7-14 触发模式，计数器时序图 (选择通道 TI2 且触发输入有效沿为上升沿)

- **外部时钟模式 1** (T1SMS = 111)：

  计数器在触发输入(TRGI)的有效沿进行计数。同时触发标志位 T1TIF 置位。

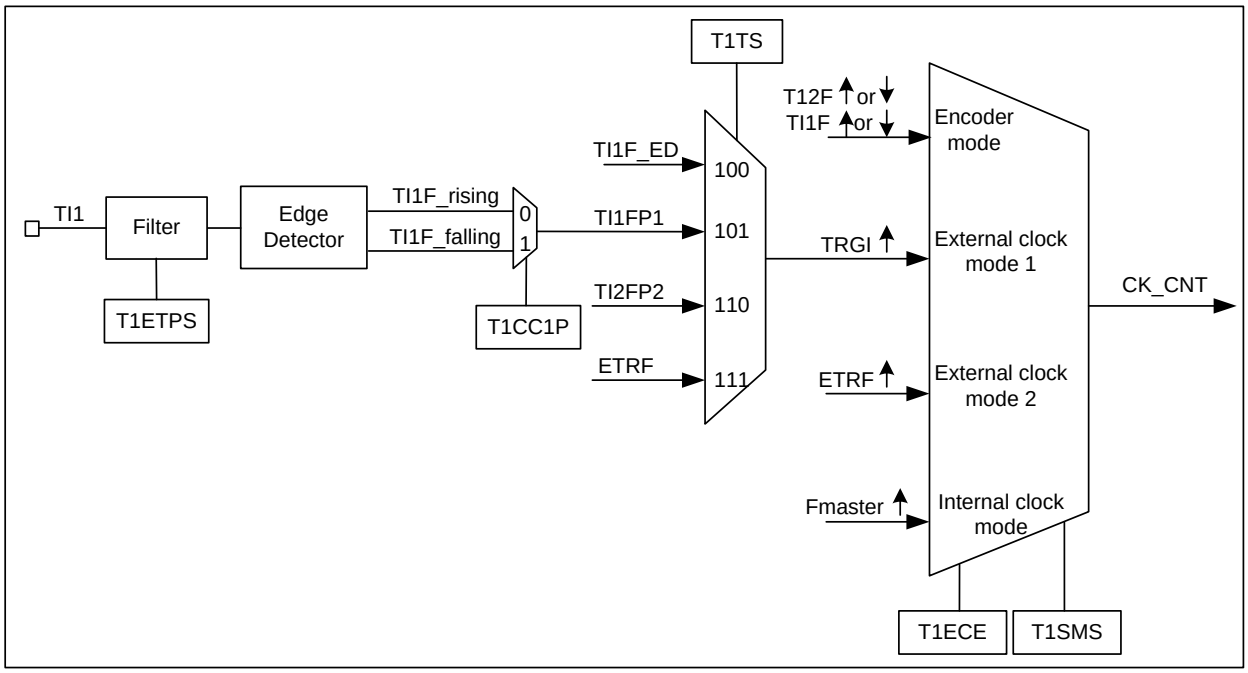


图 7-15 外部时钟模式 1 结构框图 (以 TI1 作为计数时钟源)

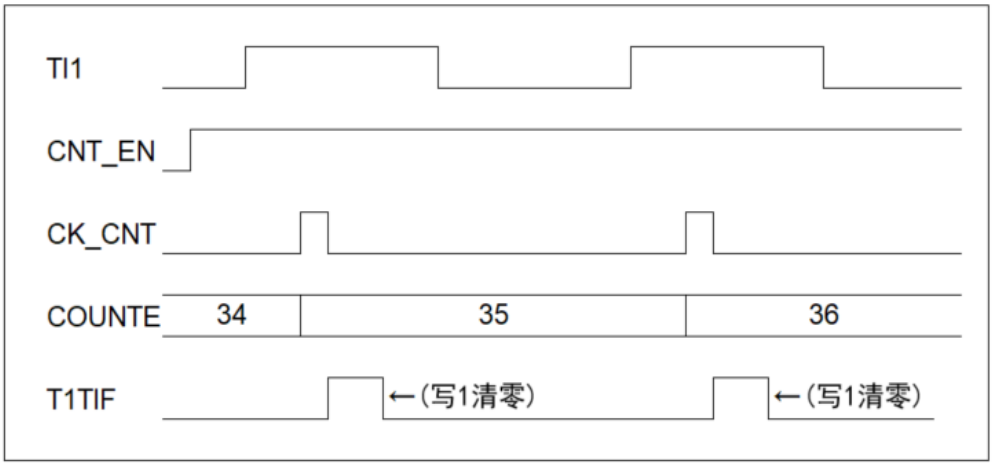


图 7-16 外部时钟模式 1，计数器时序图 (选择通道 TI1 且触发输入有效沿为上升沿)

- **外部时钟模式 2** (T1ECE = 1)：

  计数器在触发输入(ETRF)的有效沿进行计数。

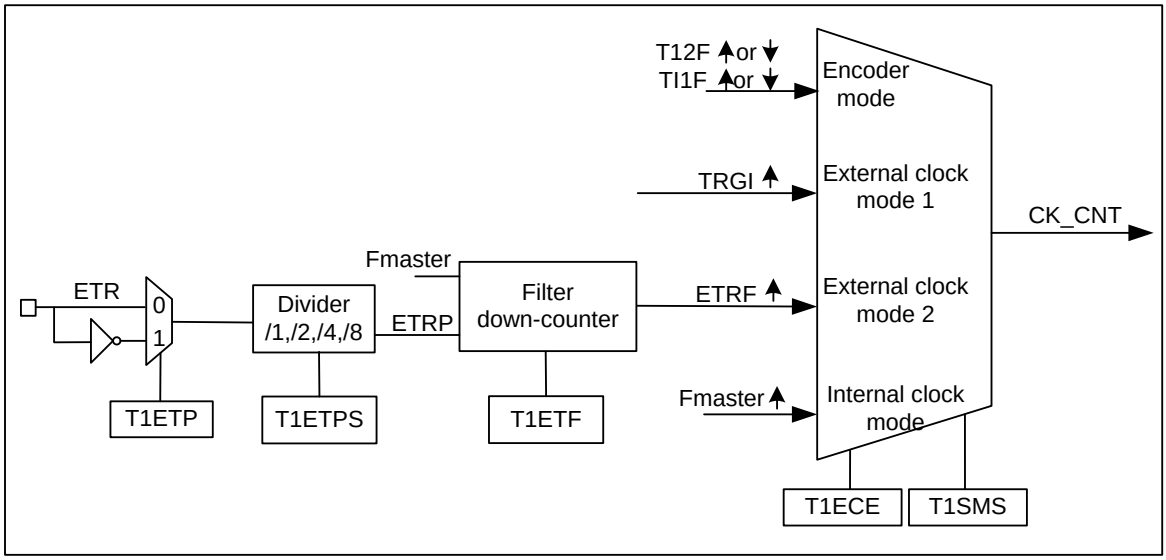


图 7-17 外部时钟模式 2 结构框图

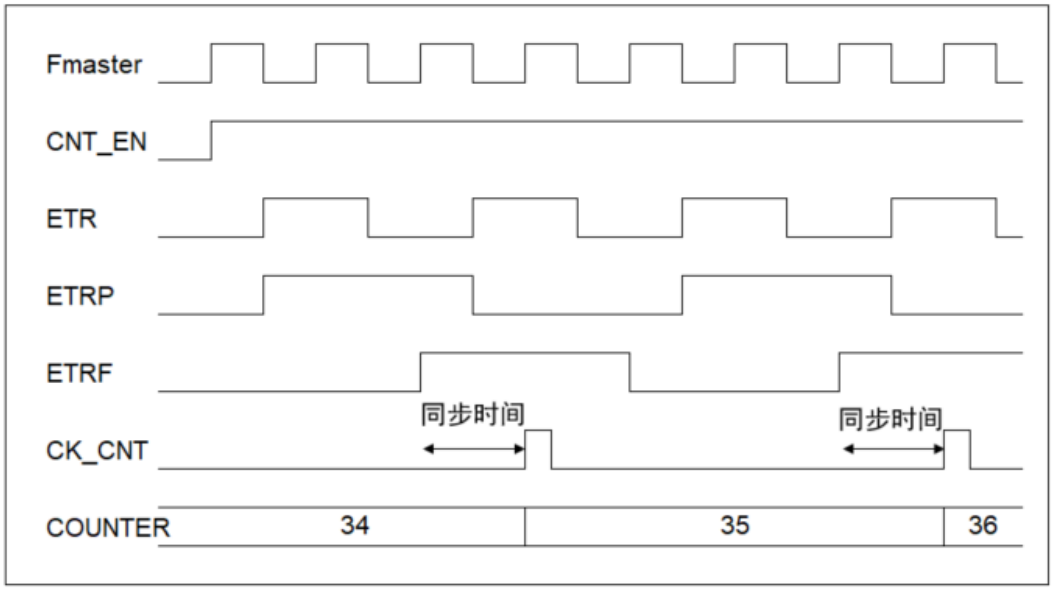


图 7-18 外部时钟模式 2，计数器时序图

控制模式(除外部时钟模式 2 外)的配置步骤示例：

1. 使能 TIM1 模块时钟(TIM1EN = 1)，并选择 TIM1 时钟源(T1CKSRC)；
2. 将所选通道端口配置为输入(TRISx = 1)；
3. 配置通道 x 的输入捕获采样频率和滤波器长度(T1ICxF)；
4. 配置通道 x 的输入捕获预分频(T1ICxPSC)；
5. 根据需要，选择输入捕获通道(T1CC1S/T1CC2S)；
6. 配置触发输入的有效沿或有效电平(T1CC1P/T1CC2P)；
7. 选择计数控制模式为复位模式、门控模式或触发模式(T1SMS)，并选择触发输入源(T1TS)；
8. 使能计数器(T1CEN = 1)；

示例程序 1 (以编码器模式 3 为例，时序图参阅 **图 7-11**)：

```
BANKSEL PCKEN
BSR PCKEN,0              ; 使能 TIM1 模块时钟
BANKSEL TCKSRC
LDWI 01H
STR TCKSRC               ; 选择 TIM1 时钟源为 HIRC
BANKSEL TRISA
LDWI 03H
STR TRISA                ; 配置通道 1 端口 PA0 和通道 2 端口 PA1 为输入
BANKSEL TIM1CCMR1
LDWI 01H
STR TIM1CCMR1            ; 配置通道 1 输入捕获滤波器长度、预分频，且输入脚映射在 TI1FP1
LDWI 01H
STR TIM1CCMR2            ; 配置通道 2 输入捕获滤波器长度、预分频，且输入脚映射在 TI2FP2
```

```
LDWI 53H
STR TIM1SMCR          ; 配置 TIM1 为编码器模式 3
LDWI 11H
STR TIM1CCER1         ; 使能通道 1 和 2 为输入，且触发输入有效电平为低电平
BANKSEL TIM1CR1
BSR TIM1CR1,0         ; 使能计数器
```

示例程序 2 (以门控模式为例，时序图参阅 **图 7-13**)：

```
BANKSEL PCKEN
BSR PCKEN,0           ; 使能 TIM1 模块时钟
BANKSEL TCKSRC
LDWI 01H
STR TCKSRC            ; 选择 TIM1 时钟源为 HIRC
BANKSEL TRISA
LDWI 01H
STR TRISA             ; 配置通道 1 端口 PA0 为输入
BANKSEL TIM1CCMR1
LDWI 01H
STR TIM1CCMR1         ; 配置通道 1 输入捕获滤波器长度、预分频，且输入脚映射在 TI1FP1
LDWI 55H
STR TIM1SMCR          ; 配置 TIM1 为门控模式，触发源为 TI1FP1
LDWI 03H
STR TIM1CCER1         ; 使能通道 1 为输入，且触发输入有效电平为低电平
BANKSEL TIM1CR1
BSR TIM1CR1,0         ; 使能计数器
BTSS TIM1SR1,6        ; 判断触发中断标志位是否为高
LJUMP $-1
BCR TIM1SR1,6         ; 将触发中断标志位清零
```

外部时钟模式 2 的配置步骤示例：

1. 使能 TIM1 模块时钟(TIM1EN = 1)，并选择 TIM1 时钟源(T1CKSRC)；
2. 将所选 ETR 外部通道端口配置为输入(TRISx = 1)；
3. 配置外部触发预分频(T1ETPS)，以及外部触发采样频率和滤波器长度(T1ETF)；
4. 配置外部触发输入的有效沿或有效电平(T1ETP)；
5. 选择计数控制模式为外部时钟模式 2(T1ECE)；
6. 使能计数器(T1CEN = 1)；

示例程序 (时序图参阅 **图 7-18**)：

```
BANKSEL PCKEN
BSR PCKEN,0           ; 使能 TIM1 模块时钟
BANKSEL TCKSRC
LDWI 01H
```

```
STR TCKSRC          ; 选择 TIM1 时钟源为 HIRC
BANKSEL TRISB
LDWI 08H
STR TRISB           ; 配置 ETR 外部通道 PB3 为输入
BANKSEL TIM1ETR
LDWI 50H
STR TIM1ETR         ; 配置外部触发预分频、滤波器、极性，并使能外部时钟模式 2
BANKSEL TIM1CR1
BSR TIM1CR1,0       ; 使能计数器
```

7.2.4. 捕获/比较通道

TIM1 的 CH1~4 端口可配置为输入捕获或输出比较功能(参阅复用寄存器 TIM1CCMRx 的 T1CCxS 位)。

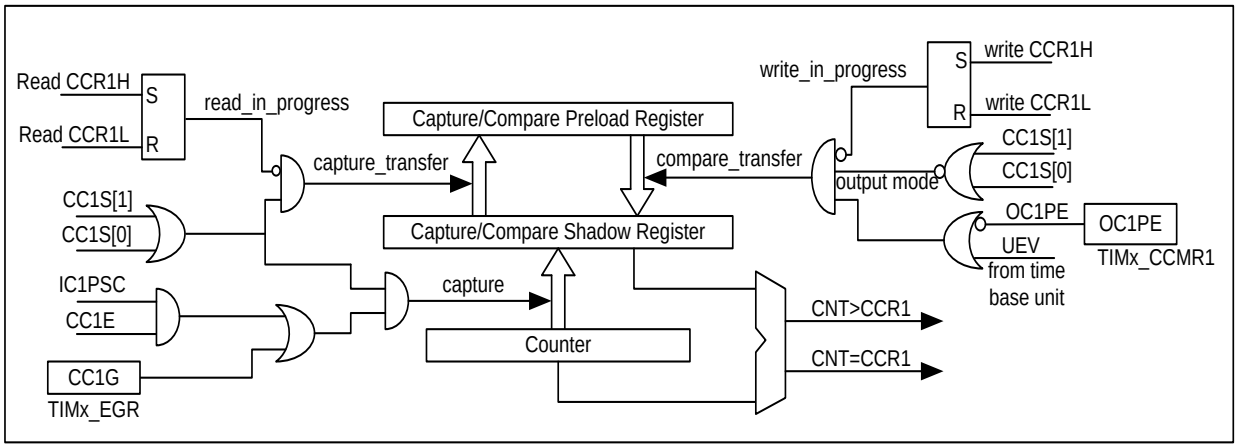


图 7-19 捕获/比较通道 1 框图

T1CCRx 寄存器由一个预装载寄存器和一个影子寄存器组成。读写过程仅操作预装载寄存器。

- 输入捕获模式下：

  T1CCRx[15:0]为只读寄存器。当发生捕获事件时，捕获到的计数器值被写入到影子寄存器中，然后再复制到 T1CCRx 预装载寄存器中。

  读 T1CCRx[15:0]寄存器时，必须先读高 8 位，再读低 8 位。读高 8 位时，预装载寄存器被冻结，然后可读到正确的低 8 位。读完低 8 位后，预装载寄存器才能更新为最新一次的捕获值。

- 输出比较模式下：

  T1CCRx[15:0]为可读可写寄存器。写操作时 T1CCRx 预装载寄存器值被复制到影子寄存器中(参阅**章节 7.2.2**)，然后影子寄存器的内容和计数器进行比较。读操作时读出的值来自预装载寄存器。

输入捕获/输出比较控制位可通过换相事件进行更新。换相事件预加载功能可使计数器在运行过程中改变控制位，而不影响上一个 PWM 波形的完整输出。

换相事件源（参阅” T1COMS”）如下：

- T1COMG 由软件置 1

- 产生触发事件 (参阅 **章节 7.2.3.2 计数器触发源**)

当产生换相事件时，换相事件标志位 T1COMIF 置位，是否触发中断和/或从睡眠中唤醒则取决于相应的使能控制位(GIE, PEIE 和 T1COMIE)。

此外，根据不同配置，换相事件可触发以下情况：

1. 当预装载使能时(T1CCPC = 1)，输入捕获/输出比较控制位(如 **表 7-20**)将在产生换相事件时更新为预装载值；
2. 当预装载关闭时(T1CCPC = 0)，输入捕获/输出比较控制位将直接更新为预装载值；

	PWM 同步模式	PWM 群组模式	通道 x 引脚功能	互补通道 x 引脚功能	通道 x 极性	互补通道 x 极性	通道 x 输出比较模式
预装载寄存器	T1SMOD	T1GP	T1CCxE	T1CCxNE	T1CCxP	T1CCxNP	T1OCxM
预装载使能位	T1CCPC						

表 7-20 换相事件相关的预装载寄存器

7.2.4.1. 输入捕获模式

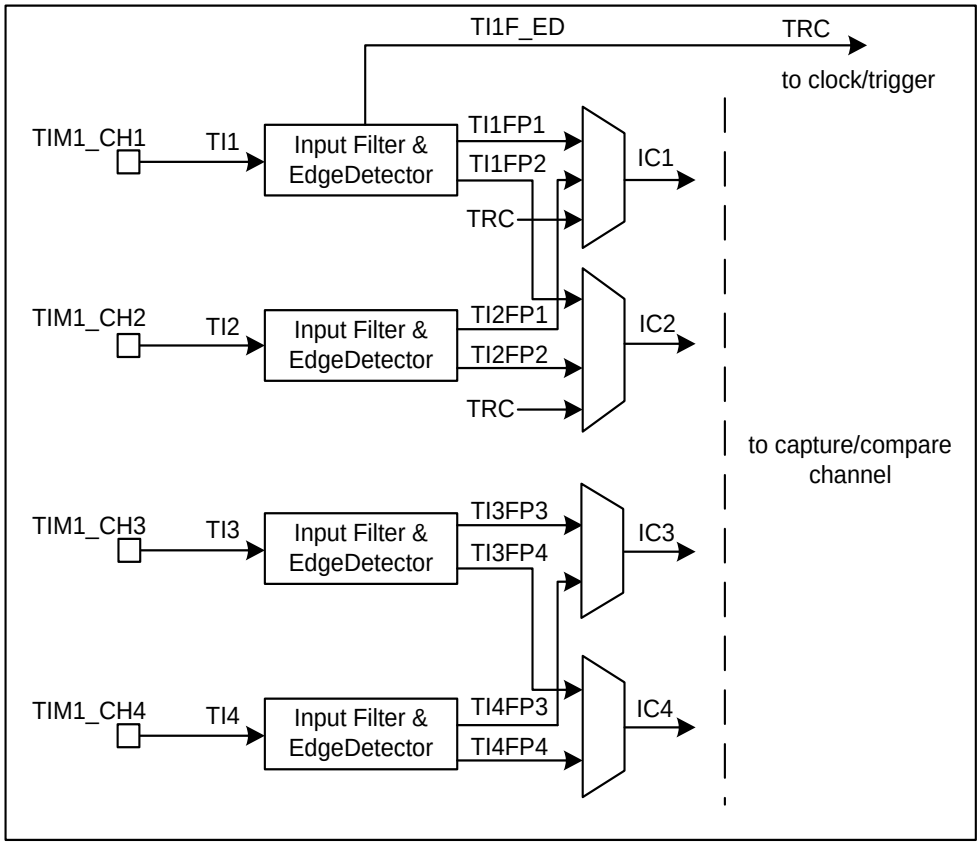


图 7-20 输入捕获通道框图

输入捕获模式下，当通道 x 发生输入捕获事件时，当前的计数值将被捕获到 T1CCRx[15:0]寄存器中，同时输入捕获标志位 T1CCxIF 置位。如果当 T1CCxIF 保持为 1 时，再次发生输入捕获事件，则重复捕获标志位 T1CCxOF 将置位。是否触发中断和/或从睡眠中唤醒则取决于相应的使能控制位(GIE, PEIE 和

T1CCxIE)。此外，可使能输入捕获软件中断位 T1CCxG 来触发中断。

TIM1 各通道的输入捕获源(参阅 T1CCxS)如下：

T1CCxS	通道 1	通道 2	通道 3	通道 4
01	TI1FP1	TI2FP2	TI3FP3	TI4FP4
10	TI2FP1	TI1FP2	TI4FP3	TI3FP4
11	TRC	TRC	—	—

表 7-21 各通道输入捕获源

注： 当通道 x (x = 1/2/3/4) 的捕获源，选择为其他通道对应 I/O 的输入捕获信号时，则需要同时将其他通道设置成输入。例如通道 1 选择 TI2FP1 (T1CC1S = 10)，则通道 2 必须设置成输入 (T1CC2S = 01 或 10)；通道 3 和通道 4 同上。

信号名称	详细说明
TIM1_CH1/2/3/4	通道 1/2/3/4 对应的 I/O 输入
IC1/2/3/4	通过选择后的通道捕获源
TI1FP1	通道 1 对应 I/O 的输入捕获信号，作为通道 1 的捕获源之一
TI1FP2	通道 1 对应 I/O 的输入捕获信号，作为通道 2 的捕获源之一
TI2FP2	通道 2 对应 I/O 的输入捕获信号，作为通道 2 的捕获源之一
TI2FP1	通道 2 对应 I/O 的输入捕获信号，作为通道 1 的捕获源之一
TI3FP3	通道 3 对应 I/O 的输入捕获信号，作为通道 3 的捕获源之一
TI3FP4	通道 3 对应 I/O 的输入捕获信号，作为通道 4 的捕获源之一
TI4FP4	通道 4 对应 I/O 的输入捕获信号，作为通道 4 的捕获源之一
TI4FP3	通道 4 对应 I/O 的输入捕获信号，作为通道 3 的捕获源之一
TRC	通道 1 对应 I/O 的输入双沿捕获信号，作为通道 1 和 2 的捕获源之一

表 7-22 输入捕获信号说明

输入捕获通道的配置步骤示例：

1. 使能 TIM1 模块时钟(TIM1EN = 1)，并选择 TIM1 时钟源(T1CKSRC)；
2. 将所选通道端口配置为输入(TRISx = 1)；
3. 选择输入捕获源(T1CCxS)；
4. 配置捕获源的极性(T1CCxP)；
5. 配置通道 x 的捕获采样频率和滤波器长度(T1ICxF[3:0])，捕获预分频(T1IC1PSC[1:0])；
6. 根据需要，可使能输入捕获中断(GIE, PEIE, T1CCxIE)；
7. 使能捕获通道(T1CCxE = 1)；
8. 使能计数器(T1CEN = 1)；

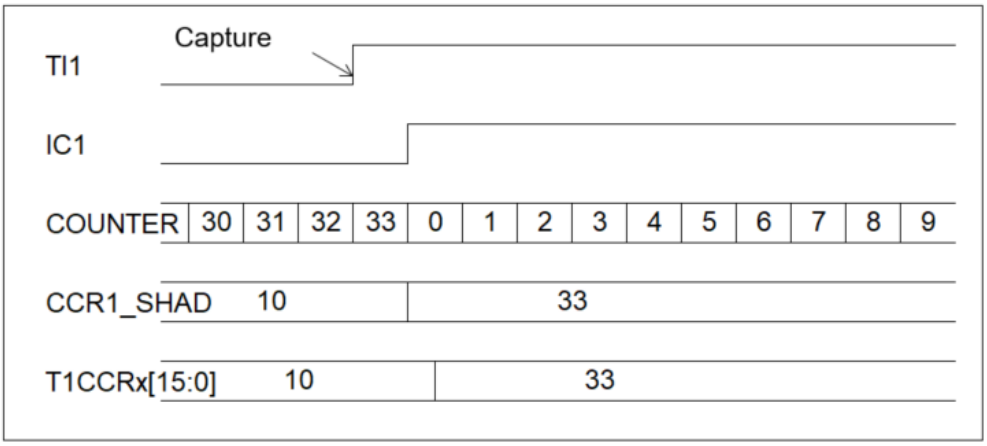


图 7-21 输入捕获时序图

PWM 信号测量应用：运用输入捕获模式和复位模式，且将 2 个通道的输入捕获源选择为同 1 个通道的 PWM 信号输入，即可测量 PWM 信号的周期及占空比。

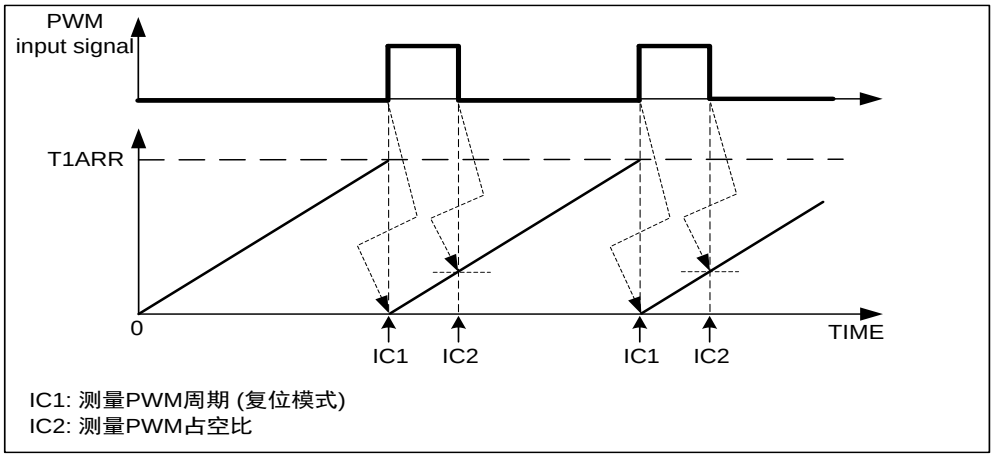


图 7-22 测量 PWM 信号示意图

测量 PWM 的配置步骤示例：

1. 使能 TIM1 模块时钟(TIM1EN = 1)，并选择 TIM1 时钟源(T1CKSRC)；
2. 将通道 1/2 对应的端口配置为输入(TRISx = 1)；
3. 选择输入捕获源(T1CCxS)，将通道 1 的 IC1 映射在 TI1FP1 上，通道 2 的 IC2 映射在 TI2FP1 上；
4. 配置捕获源的极性，将通道 1 配置为上升沿(T1CC1P = 0)，通道 2 配置为下降沿(T1CC2P = 1)；
5. 配置捕获采样频率和滤波器长度(T1ICxF[3:0] = 0000)，捕获预分频(T1IC1PSC[1:0] = 00)；
6. 配置计数控制模式为复位模式(T1SMS = 101)，计数触发源为 TI1FP1(T1TS = 101)；
7. 使能通道 1 和通道 2 的输入捕获功能(T1CC1E=1 且 T1CC2E=1)；
8. 使能计数器(T1CEN = 1)；

注：因捕获沿先于复位触发源两个计数时钟周期，为了得到准确测量值，软件需要做以下处理：

- 当预分频 = 0 时，PWM 周期 = T1CCR1H/L+2，占空比 = T1CCR2H/L+2；

- 当预分频 = 1 时，PWM 周期 = T1CCR1H/L+1，占空比 = T1CCR2H/L+1；
- 当预分频 ≥ 1 时，PWM 周期 = T1CCR1H/L，占空比 = T1CCR2H/L；

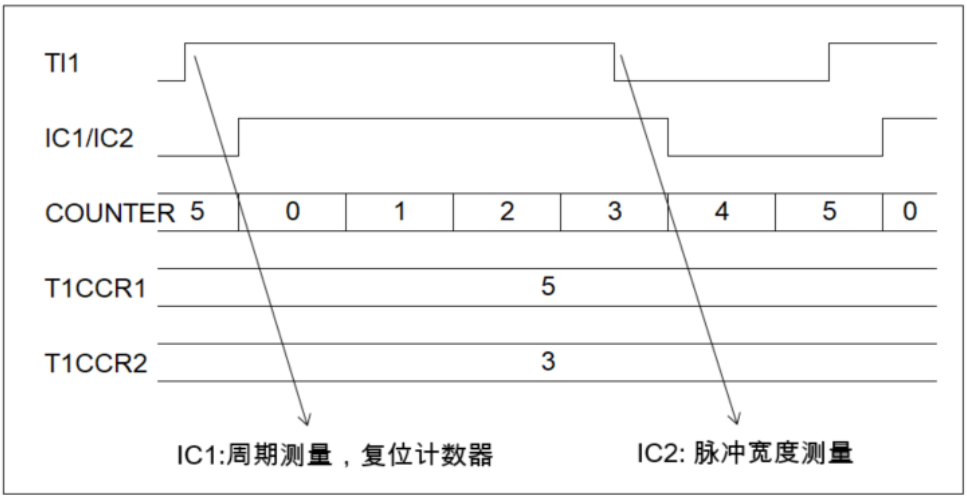


图 7-23 测量 PWM 信号的时序图

7.2.4.2. 输出比较模式

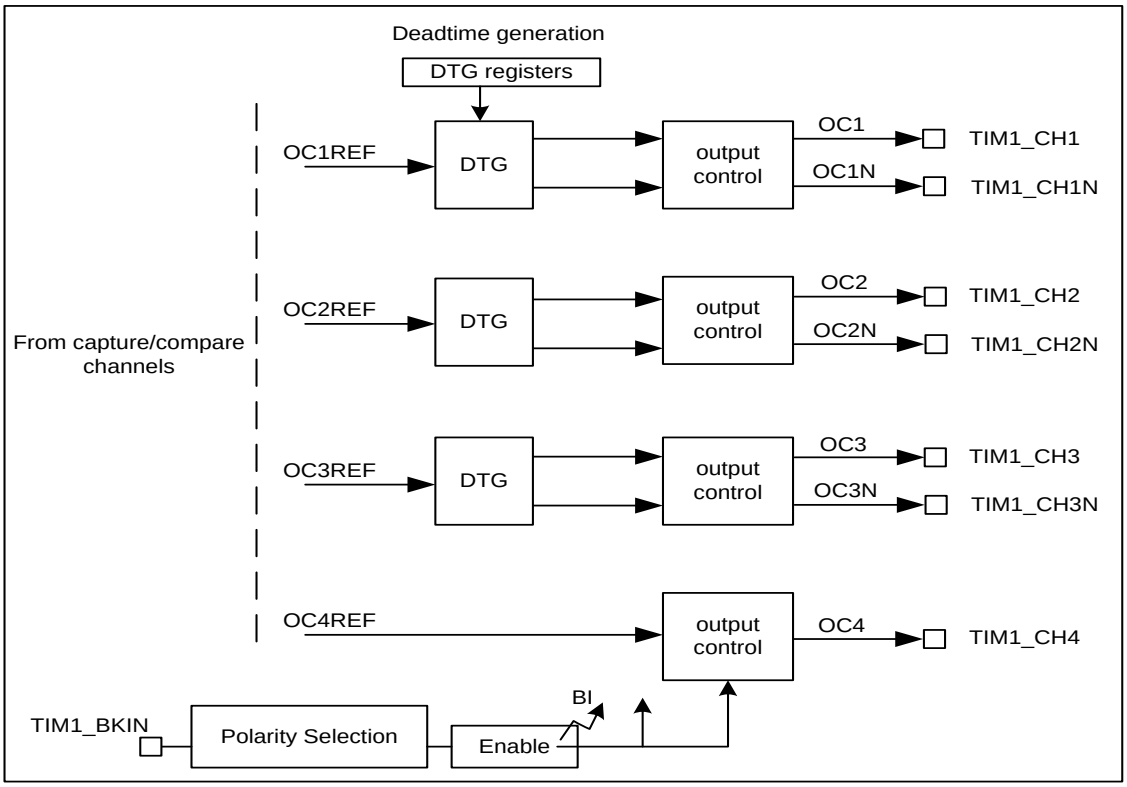


图 7-24 输出比较通道框图

输出比较模块通过对比计数值(T1CNT)与比较值(影子寄存器 CCRx_SHAD)，先产生输出参考信号 OCxREF(高电平有效)，然后送到死区产生模块或刹车模块，再经过极性选择和其他输出控制位 (参阅 T1MOE, T1OSSI, T1OSSR, T1CCxE 和 T1CCxNE) 将波形输出到端口 (参阅 **表 7-14**)。

参考信号 OCxREF 可通过 T1OCxM[2:0] 配置为 8 种输出模式 (参阅 **表 7-9**)：

1. 冻结模式(T1OCxM = 000)：OCxREF 值保持不变；
2. 匹配有效(T1OCxM = 001)：当 T1CNT = CCRx_SHAD 时，OCxREF = 1；
3. 匹配无效(T1OCxM = 010)：当 T1CNT = CCRx_SHAD 时，OCxREF = 0；
4. 翻转模式(T1OCxM = 011)：当 T1CNT = CCRx_SHAD 时，OCxREF 值翻转；
5. 强制无效(T1OCxM = 100)：OCxREF 一直为 0；
6. 强制有效(T1OCxM = 101)：OCxREF 一直为 1；
7. PWM1 模式(T1OCxM = 110)：

   当 T1CNT < CCRx_SHAD 时，OCxREF = 1；当 T1CNT > CCRx_SHAD 时，OCxREF = 0；
8. PWM2 模式(T1OCxM = 111)：

   当 T1CNT < CCRx_SHAD 时，OCxREF = 0；当 T1CNT > CCRx_SHAD 时，OCxREF = 1；

当 T1CNT 与 CCRx_SHAD 匹配时，输出比较标志位 T1CCxIF 置位。是否触发中断和/或从睡眠中唤醒则取决于相应的使能控制位(GIE, PEIE, T1CCxIE)。此外，可使能输出比较软件中断位 T1CCxG 来触发中断。

输出比较通道的配置步骤：

1. 使能 TIM1 模块时钟(TIM1EN = 1)，并选择 TIM1 时钟源(T1CKSRC)；
2. 将通道对应的端口配置为输出(TRISx = 0)；
3. 配置输出波形的周期(T1ARR) 和比较值(T1CCRx)；
4. 配置输出比较模式(T1OCxM)和输出极性(T1CCxP)；
5. 根据需要，可使能输出比较中断(GIE, PEIE, T1CCxIE)；
6. 使能输出比较通道(T1CCxE = 1)；
7. 使能主输出自动控制位(T1AOE = 1)，即硬件会在产生更新事件时自动使能主输出(T1MOE)；
8. 使能计数器(T1CEN = 1)；

注：建议配置输出比较值 T1CCRx ≤ 计数周期值 T1ARR；

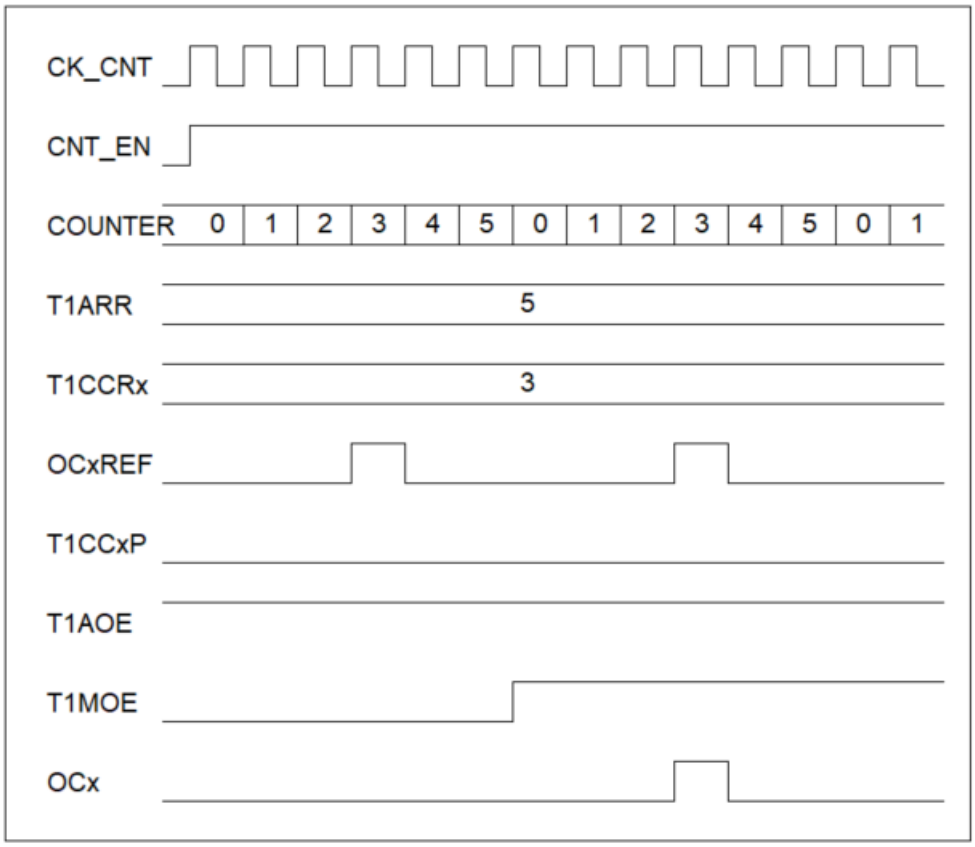


图 7-25 匹配有效模式下的输出时序图

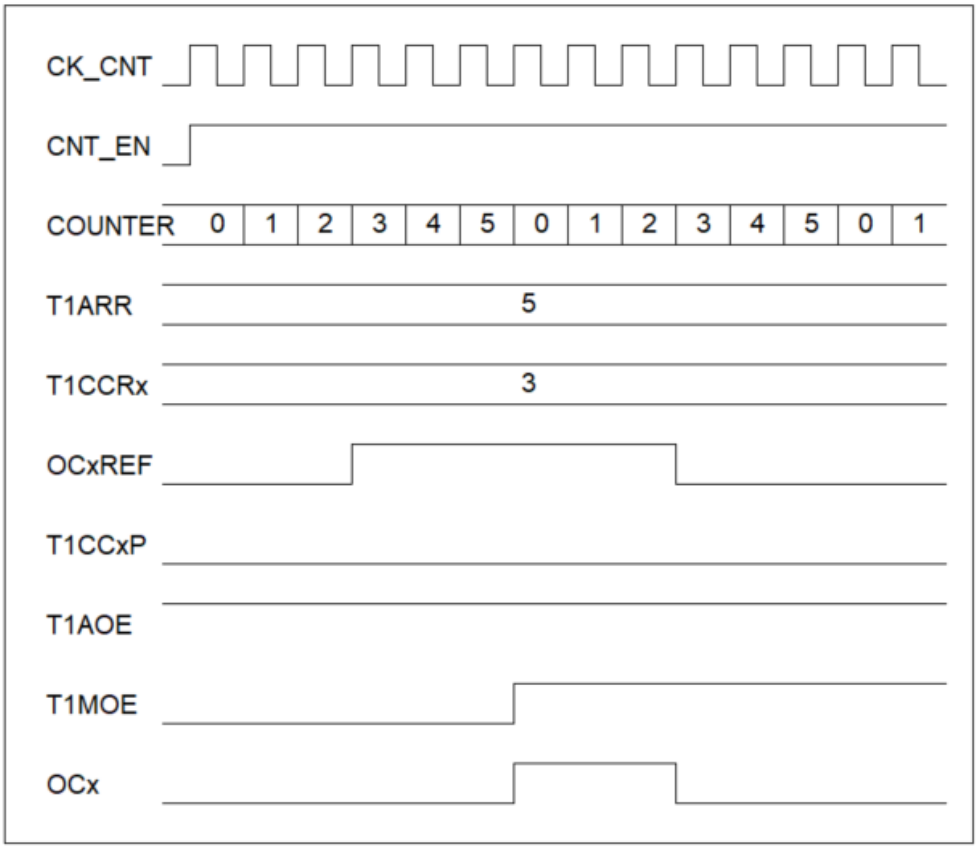


图 7-26 翻转模式下的输出时序图

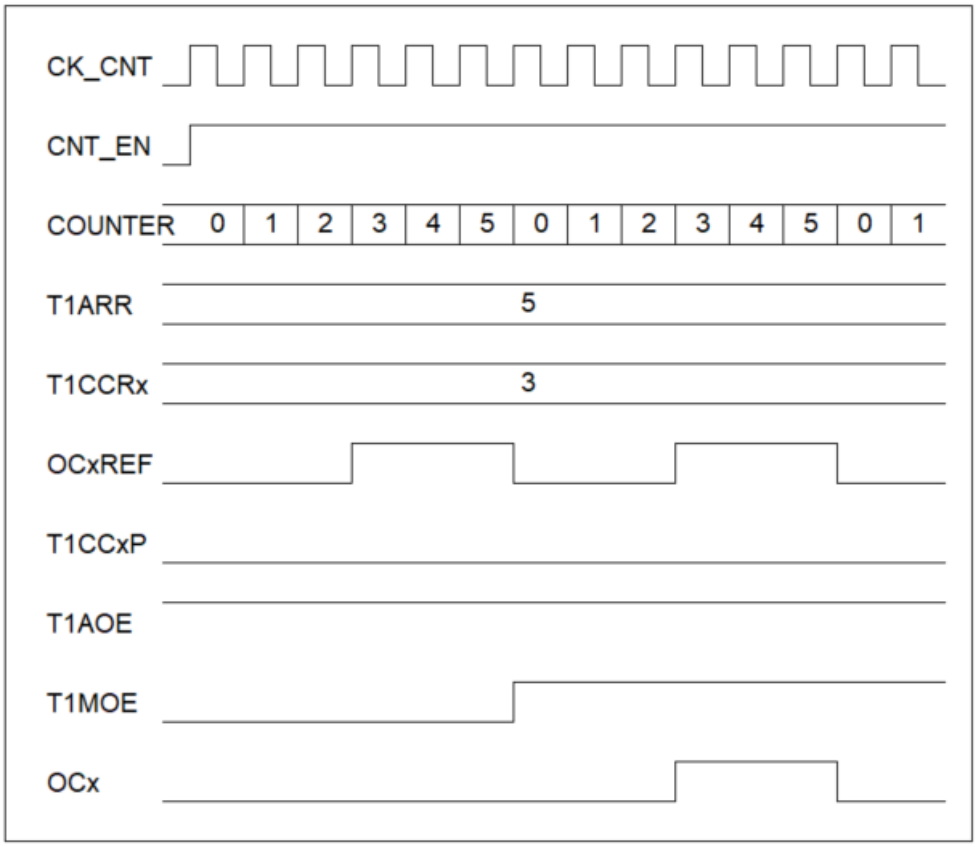


图 7-27 PWM2 模式下的输出时序图

PWM 模式 – PWM1/PWM2 周期由 T1ARR 决定、占空比由 T1CCRx 决定。

公式 7-1 *PWM1/2 周期* = *(T1ARR+1)* * T_{CK_CNT}

公式 7-2 *PWM1/2 占空比* = *T1CCRx ÷ (T1ARR+1)*

TIM1_CH1/2/3/4 通道可独立使能输出 PWM 信号，其中 CHI/2/3 带互补输出功能。输出信号以及互补输出信号的极性可选(参阅 T1CCxP/ T1CCxNP)。当输出通道以及互补输出通道同时使能时(T1CCxE = 1, T1CCxNE = 1)，将自动使能死区功能，并可设置死区时间(参阅 T1DTG)，即每当一个输出信号(OCx 或互补输出 OCxN) 出现下降沿时，另一个信号的上升沿将延后一个死区时间长度。

注：PWM 模式下，必须使能通道 x 占空比的自动预装载功能(T1OCxPE = 1)；

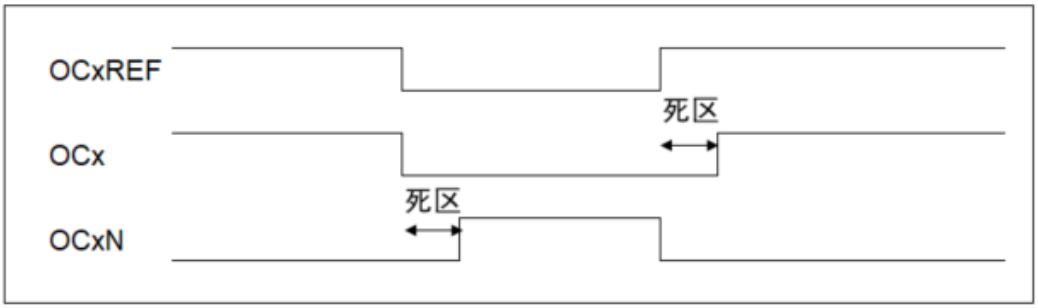


图 7-28 互补输出插入死区时间时序图

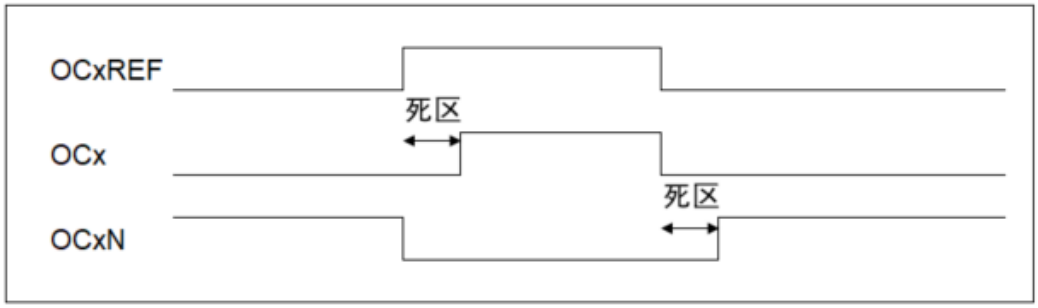


图 7-29 正向输出插入死区时间时序图

当 OCxREF 输出的脉冲时间比死区时间短时，某一个脉冲信号可能会被死区覆盖，导致输出不变。

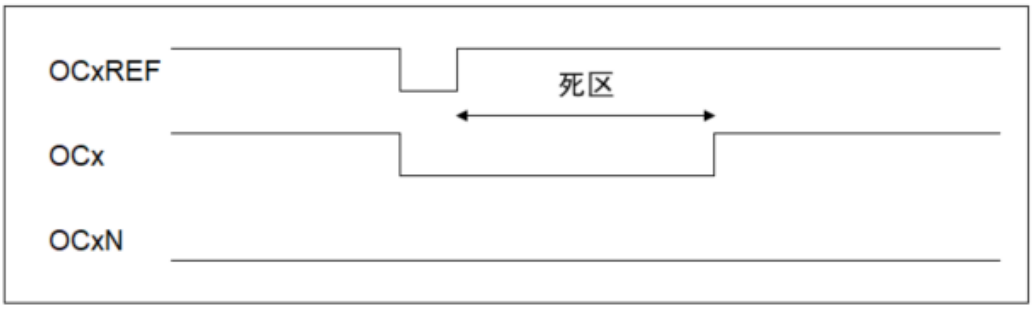


图 7-30 互补输出被死区覆盖时序图

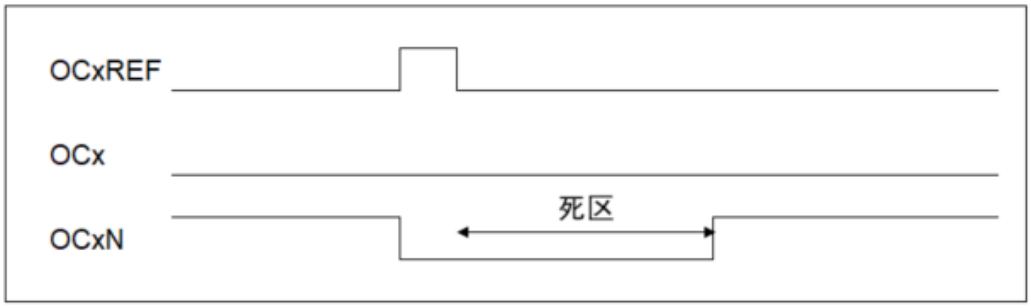


图 7-31 正向输出被死区覆盖时序图

PWM 模式与重复计数器、更新事件、周期预装载和占空比预装载等功能结合使用，可产生特定数量 PWM 信号。

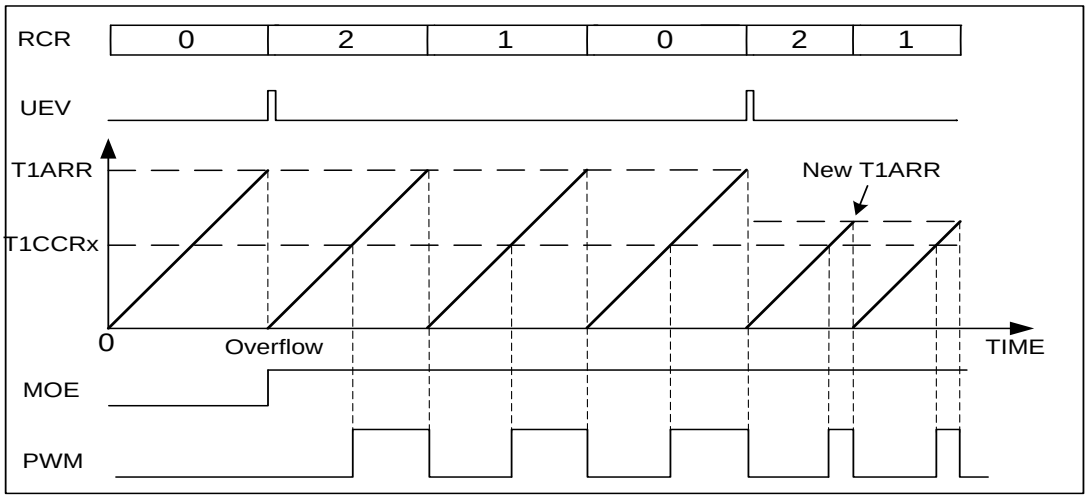


图 7-32 运用重复计数器输出 3 个特定 PWM 的时序图

示例程序：

```
BANKSEL PCKEN
BSR PCKEN,0          ; 使能 TIM1 模块时钟
BANKSEL INTCON
LDWI C0H
STR INTCON           ; 使能全局中断和外设中断
BANKSEL TCKSRC
LDWI 01H
STR TCKSRC           ; 选择 TIM1 时钟源为 HIRC
BANKSEL TRISA
LDWI 00H
STR TRISA            ; 配置通道 1 端口 PA0 为输出
BANKSEL TIM1ARRL
LDWI 1FH
STR TIM1ARRL         ; 配置输出波形周期为 32
LDWI 10H
STR TIM1CCR1L        ; 配置输出波形占空比为 16
LDWI 02H
STR TIM1RCR          ; 配置重复计数器为 2
BANKSEL TIM1CCMR1
LDWI 70H
STR TIM1CCMR1        ; 配置通道 1 为 PWM2 模式输出
BSR TIM1IER,0        ; 使能更新事件中断
LDWI 01H
STR TIM1CCER1        ; 使能通道 1，并选择极性
BANKSEL TIM1BKR
BSR TIM1BKR,6        ; 打开主输出自动使能位
BANKSEL TIM1CR1
LDWI 81H
STR TIM1CR1          ; 使能周期预装载功能，并使能计数器
```

INT:
BANKSEL TIM1ARRL
LDWI 14H
STR TIM1ARRL ; 重新配置输出波形周期为为 20

单脉冲模式 – 单脉冲模式下(T1OPM = 1)，当产生下一次更新事件时，硬件会自动关闭计数器使能位(T1CEN = 0)，计数器停止计数。

为产生一个正确的脉冲，计数器初始值(T1CNT)必须与比较值(T1CCRx)不同。即启动计数前，需满足以下配置：

- 向上计数模式：T1CNT < T1CCRx ≤ T1ARR
- 向下计数模式：T1CNT > T1CCRx

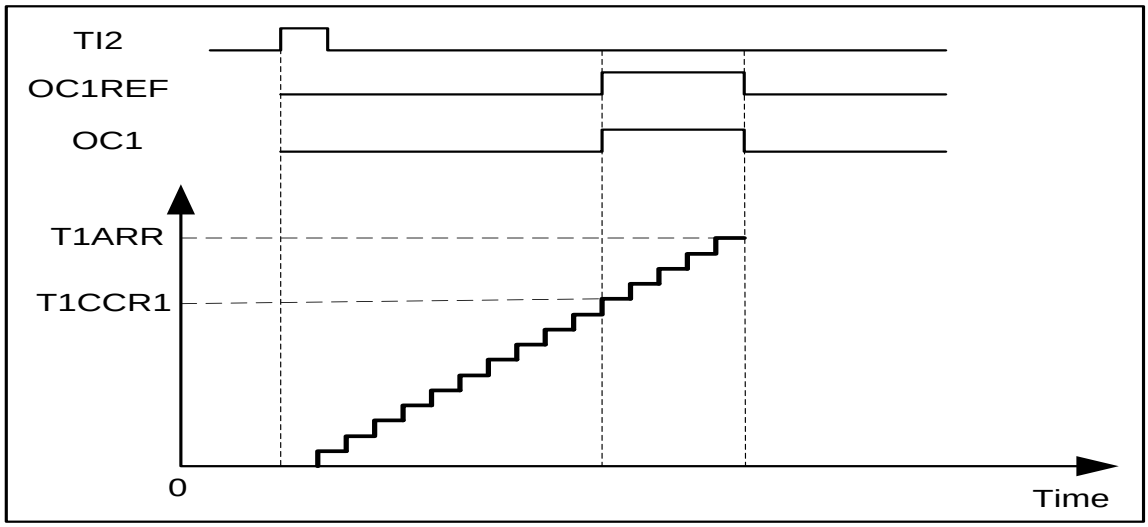


图 7-33 单脉冲应用示意图

单脉冲模式可配合触发模式在特定的时间点产生单个脉冲输出，如上图所示。配置步骤如下：

1. 使能 TIM1 模块时钟(TIM1EN = 1)，并选择 TIM1 时钟源(T1CKSRC)；
2. 将通道 2 对应的端口配置为输入，通道 1 对应的端口配置为输出；
3. 将通道 2 的输入映射在 TI2FP2 上(T1CC2S = 01)，并选为上升沿捕获(T1CC2P = 0)；
4. 将计数控制模式配置为触发模式(T1SMS = 110)，触发源选为 TI2FP2 (T1TS = 110)；
5. 将通道 1 配置为输出通道(T1CC1S = 00)；
6. 将通道 1 配置为输出比较模式的 PWM2 模式(T1OC1M = 111)，输出极性配置为高电平有效(T1CC1P = 0)；
7. 使能通道 2 的输入捕获功能(T1CC2E=1)，以及通道 1 的输出比较功能 (T1CC1E)；
8. 使能主输出自动控制位(T1AOE = 1)，即硬件会在产生更新事件时自动使能主输出(T1MOE)；
9. 使能计数器(T1CEN = 1)；

快速输出 – 当 PWM 或单脉冲模式与触发模式配合使用时，触发输入有效沿会使能计数器计数，计数值与 T1CCRx 值的比较会使输出发生变化。但触发输入边沿发生变化到真正输出波形之间会有一段最小延时，可通过设置 T1OCxFE 来缩短延时。

同步模式 – 通道 1/2/3 的互补输出与其正向输出信号完全相同 (参阅 ”T1SMOD”)。同步模式下，硬件自动禁止死区功能。

群组模式 – 通道 1/2/3 及其互补输出的占空比均由 T1CCR1[15:0]决定，原配置 T1CCR2[15:0]和 T1CCR3[15:0]无效 (参阅 ”T1GP”)。

TIM1/TIM2 同步触发模式 – TIM2 的内部使能信号完全由 TIM1 的 TICEN 控制 (参阅 ”T1ENCTRL”)。配置过程如下：

1. 先配置 TIM1 和 TIM2 相应功能；
2. 配置 T1ENCTRL = 1，使能同步触发功能；
3. 配置 TICEN = 1，使 TIM1 和 TIM2 同时开启；

7.2.4.3. 故障刹车(Fault-Break)功能

4 路 PWM 均支持故障刹车功能。当刹车输入功能使能(T1BKE = 1)时，一旦发生故障刹车事件，且只要故障条件一直存在，则 PWM 输出引脚将根据其设置输出预设状态(参阅 TIM1OISR)。

TIM1 故障刹车事件可以为下列条件之一(参阅 BKS)：

- BKIN 管脚事件
- LVD 事件
- ADC 阈值比较事件

发生刹车事件时，PWM 输出及互补输出状态(参阅 **表 7-14**)如下：

- 如果 TIM1 的时钟源关闭，T1MOE 会被异步清零，强制输出为无效状态；
- 如果 TIM1 的时钟源打开，T1MOE 会被异步清零，输出在死区时间内为无效状态，死区时间后为空闲状态(此情况下，真正的死区时间比死区时间设置值长 2 个 CK_CNT 时钟)；

同时，刹车中断标志位 T1BIF 置位。是否触发中断和/或从睡眠中唤醒则取决于相应的使能控制位(GIE, PEIE, T1BIE)。此外，可使能刹车软件中断位 T1BG 来触发中断。

故障事件撤销后，如果主输出自动控制位 T1AOE = 1，那么 T1MOE 将在下次更新事件(UEV)到来时，由硬件自动置位(需同步 2 个 CK_CNT 时钟)，PWM 恢复正常输出。否则需由软件置位 T1MOE 以恢复输出。

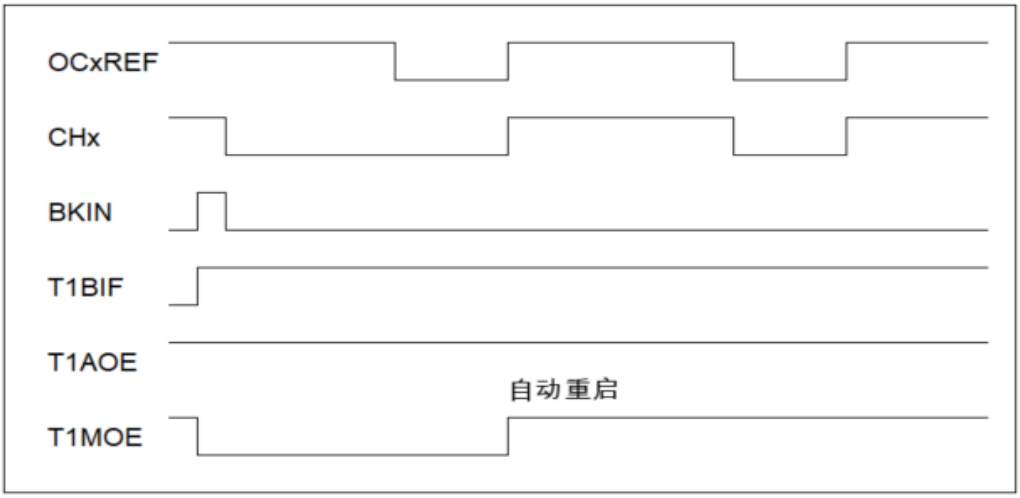


图 7-34 PWM 自动重启示意图

7.3. 通用定时器 2 (TIMER2)

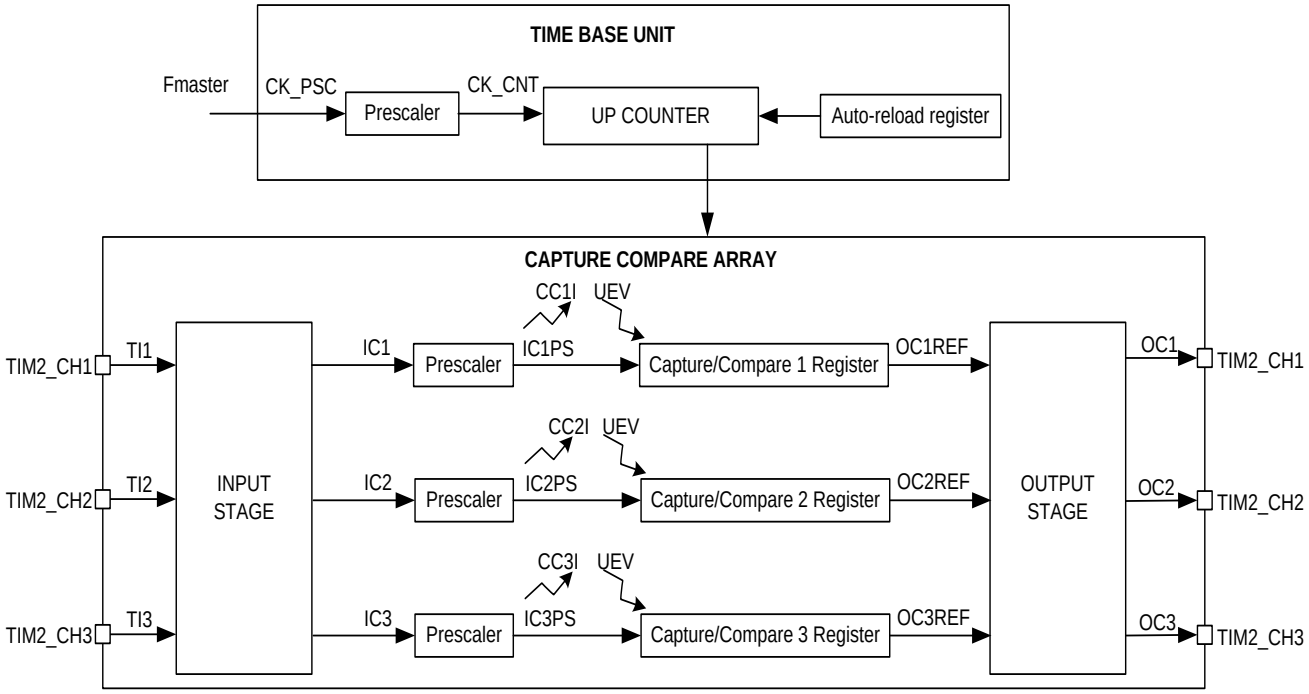


图 7-35 TIM2 原理框图

TIM2 特性：

- 16 位向上计数器，支持自动重载
- 4 位可编程预分频器
- 3 路极性可选的通道支持：
  - ✓ 输入捕获
  - ✓ 输出比较
  - ✓ 周期相同、独立占空比的 PWM 通道
  - ✓ 单脉冲模式
- 中断事件：更新事件、输入捕获、输出比较

7.3.1. Timer2 相关寄存器汇总

名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	复位值
PCKEN	0x9A	—	—	UARTEN	—	TIM4EN	TIM2EN	TIM1EN	ADCEN	0000 0000
CKOCON	0x95	SYSON	CCORDY	DTYSEL		CCOSEL[2:0]			CCOEN	0010 0000
TIM2CR1	0x30C	T2ARPE	—	—	—	T2OPM	T2URS	T2UDIS	T2CEN	0--- 000
TIM2IER	0x30D	—	—	—	—	T2CC3IE	T2CC2IE	T2CC1IE	T2UIE	---- 0000
TIM2SR1	0x30E	—	—	—	—	T2CC3IF	T2CC2IF	T2CC1IF	T2UIF	---- 0000
TIM2SR2	0x30F	—	—	—	—	T2CC3OF	T2CC2OF	T2CC1OF	—	---- 000-
TIM2EGR	0x310	—	—	—	—	T2CC3G	T2CC2G	T2CC1G	T2UG	---- 0000
TIM2CCMR1 (output mode)	0x311	—	T2OC1M[2:0]			T2OC1PE	—	T2CC1S[1:0]		-000 0-00
TIM2CCMR1 (input mode)		T2IC1F[3:0]				T2IC1PSC[1:0]		T2CC1S[1:0]		0000 0000
TIM2CCMR2 (output mode)	0x312	—	T2OC2M[2:0]			T2OC2PE	—	T2CC2S[1:0]		-000 0-00
TIM2CCMR2 (input mode)		T2IC2F[3:0]				T2IC2PSC[1:0]		T2CC2S[1:0]		0000 0000
TIM2CCMR3 (output mode)	0x313	—	T2OC3M[2:0]			OC3PE	—	T2CC3S[1:0]		-000 0-00
TIM2_CCMR3 (input mode)		T2IC3F[3:0]				T2IC3PSC[1:0]		T2CC3S[1:0]		0000 0000
TIM2CCER1	0x314	—	—	T2CC2P	T2CC2E	—	—	T2CC1P	T2CC1E	--00 --00
TIM2CCER2	0x315	—	—	—	—	—	—	T2CC3P	T2CC3E	---- --00
TIM2CNTRH	0x316	T2CNT[15:8]								0000 0000
TIM2CNTRL	0x317	T2CNT[7:0]								0000 0000
TIM2PSCR	0x318	—	—	—	—	T2PSC[3:0]				---- 0000
TIM2ARRH	0x319	T2ARR[15:8]								1111 1111
TIM2ARRL	0x31A	T2ARR[7:0]								1111 1111
TIM2CCR1H	0x31B	T2CCR1[15:8]								0000 0000
TIM2CCR1L	0x31C	T2CCR1[7:0]								0000 0000
TIM2CCR2H	0x31D	T2CCR2[15:8]								0000 0000
TIM2CCR2L	0x31E	T2CCR2[7:0]								0000 0000
TIM2CCR3H	0x29E	T2CCR3[15:8]								0000 0000
TIM2CCR3L	0x29F	T2CCR3[7:0]								0000 0000

表 7-23 Timer2 相关用户寄存器汇总（—保留位必须保持为复位值，不可更改）

名称	状态		寄存器	地址	复位值
T2CNT	TIM2 计数值	高 8 位	TIM2CNTRH[7:0]	0x316	RW-0000 0000
		低 8 位	TIM2CNTRL[7:0]	0x317	RW-0000 0000
T2PSC	TIM2 预分频器		TIM2PSCR[3:0]	0x318	RW-0000
T2ARR	计数周期的自动重载寄存器(预装载值) 注：此值为 0 时，计数器不工作；	高 8 位	TIM2ARRH[7:0]	0x319	RW-1111 1111
		低 8 位	TIM2ARRL[7:0]	0x31A	RW-1111 1111
T2CCR1	输入捕获模式: 上一次捕获事件(IC1)捕获的计数值	高 8 位	TIM2CCR1H[7:0]	0x31B	RO-0000 0000
		低 8 位	TIM2CCR1L[7:0]	0x31C	RO-0000 0000
	输出比较模式: TIM2_CH1 的输出比较值(预装载值)	高 8 位	TIM2CCR1H[7:0]	0x31B	RW-0000 0000
		低 8 位	TIM2CCR1L[7:0]	0x31C	RW-0000 0000
T2CCR2	输入捕获模式: 上一次捕获事件(IC2)捕获的计数值	高 8 位	TIM2CCR2H[7:0]	0x31D	RO-0000 0000
		低 8 位	TIM2CCR2L[7:0]	0x31E	RO-0000 0000
	输出比较模式: TIM2_CH2 的输出比较值(预装载值)	高 8 位	TIM2CCR2H[7:0]	0x31D	RW-0000 0000
		低 8 位	TIM2CCR2L[7:0]	0x31E	RW-0000 0000
T2CCR3	输入捕获模式: 上一次捕获事件(IC3)捕获的计数值	高 8 位	TIM2CCR3H[7:0]	0x29E	RO-0000 0000
		低 8 位	TIM2CCR3L[7:0]	0x29F	RO-0000 0000
	输出比较模式: TIM2_CH3 的输出比较值(预装载值)	高 8 位	TIM2CCR3H[7:0]	0x29E	RW-0000 0000
		低 8 位	TIM2CCR3L[7:0]	0x29F	RW-0000 0000

表 7-24 Timer2 周期相关寄存器

名称	状态	寄存器	地址	复位值
TIM2EN	TIM2 模块时钟 1 = 使能 0 = 关闭	PCKEN[2]	0x9A	RW−0
SYSON	睡眠模式下，系统时钟控制 1 = 使能 0 = 关闭	CKOCON[7]	0x95	RW−0
T2CKSRC	Timer2 时钟源 (Fmaster) 000 = Sysclk　100 = 2x (XT or EC) (*) 001 = HIRC　101 = LIRC 010 = XT or EC (*)　110 = LP or EC (*) 011 = 2x HIRC　111 = 2x (LP or EC) (*) (*) FOSC 应相应配置成 LP/XT/EC 模式，否则振荡器将不会运行。	TCKSRC[6:4]	0x31F	RW−000
DTYSEL	TIM1/TIM2 倍频时钟占空比调节位 00 = 2ns 延迟　10 = 4ns 延迟 01 = 3ns 延迟　11 = 7ns 延迟	CKOCON[5:4]	0x95	RW−10
T2ARPE	计数周期的自动预装载 1 = 使能 (T2ARR 预装载值在更新事件到来时被加载) 0 = 禁止 (T2ARR 立即被加载)	TIM2CR1[7]	0x30C	RW−0
T2OPM	单脉冲模式 1 = 使能 (下一次更新事件到来时，T2CEN 自动清零，计数器停止) 0 = 关闭 (发生更新事件时，计数器不停止)	TIM2CR1[3]		RW−0
T2URS	当 T2UDIS=0 时，更新事件中断源 1 / 0 = 计数器上溢	TIM2CR1[2]		RW−0
T2UDIS	产生更新事件控制 1 = 禁止 0 = 允许	TIM2CR1[1]		RW−0
T2CEN	TIM2 计数器 1 = 使能 0 = 关闭	TIM2CR1[0]		RW−0

表 7-25 Timer2 相关用户控制寄存器

名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	Bit0	复位值
TIM2CCMR1	0x311	T2IC1F[3:0]				T2IC1PSC[1:0]		T2CC1S[1:0]		RW-0000 0000
TIM2CCMR2	0x312	T2IC2F[3:0]				T2IC2PSC[1:0]		T2CC2S[1:0]		RW-0000 0000
TIM2CCMR3	0x313	T2IC3F[3:0]				T2IC3PSC[1:0]		T2CC3S[1:0]		RW-0000 0000

名称	状态			寄存器	地址	复位值
T2ICxF	通道 x 输入捕获的采样频率和数字滤波器长度			TIM2CCMRx[7:4] x = 1, 2, 3	0x311/ 0x312/ 0x313	RW−0000
	Value	采样频率 ($f_{SAMPLING}$)	数字滤波器 长度 (N)			
	0000	Fmaster /2	0			
	0001	Fmaster	2			
	0010	Fmaster	4			
	0011	Fmaster	8			
	0100	Fmaster / 2	6			
	0101	Fmaster / 2	8			
	0110	Fmaster / 4	6			
	0111	Fmaster / 4	8			
	1000	Fmaster / 8	6			
	1001	Fmaster / 8	8			
	1010	Fmaster / 16	5			
	1011	Fmaster / 16	6			
	1100	Fmaster / 16	8			
	1101	Fmaster / 32	5			
	1110	Fmaster / 32	6			
	1111	Fmaster / 32	8			
T2ICxPSC	通道 x 输入捕获预分频器（几个事件触发一次捕获） 00 = 1 个 01 = 2 个 10 = 4 个 11 = 8 个 注：当 T2CCxE = 0 时，该预分频器复位为 00			TIM2CCMRx[3:2]		RW−00
T2CC1S [8]	通道 1 模式选择	00 = 输出 01 = 输入，输入脚映射在 TI1FP1 10 = 输入，输入脚映射在 TI2FP1 11 = 保留		TIM2CCMR1[1:0]	0x311	RW−00

[8] 仅在通道 x 关闭时(即 T2CCxE = 0)可写，x = 1, 2, 3。

名称	状态		寄存器	地址	复位值
T2CC2S [9]	通道 2 模式选择	00 = 输出 01 = 输入，输入脚映射在 TI2FP2 10 = 输入，输入脚映射在 TI1FP2 11 = 保留	TIM2CCMR2[1:0]	0x312	RW−00
T2CC3S [9]	通道 3 模式选择	00 = 输出 01 = 输入，输入脚映射在 TI3FP3 1x = 保留	TIM2CCMR3[1:0]	0x313	RW−00

表 7-26 TIM2CCMRx 作为输入配置寄存器

名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	Bit0	复位值
TIM2CCMR1	0x311	−	T2OC1M[2:0]			T2OC1PE	−	T2CC1S[1:0]		RW−-000 0-00
TIM2CCMR2	0x312	−	T2OC2M[2:0]			T2OC2PE	−	T2CC2S[1:0]		RW−-000 0-00
TIM2CCMR3	0x313	−	T2OC3M[2:0]			T2OC3PE	−	T2CC3S[1:0]		RW−-000 0-00

T2OCxM	通道 x 输出比较模式		参考信号 OCxREF 的电平值
000	冻结 (不比较)		保持不变
001	当 T2CNT = CCRx_SHAD 时		1
010	当 T2CNT = CCRx_SHAD 时		0
011	当 T2CNT = CCRx_SHAD 时		电平翻转
100	强制无效		0
101	强制有效		1
110	PWM1 模式	T2CNT < CCRx_SHAD	1
		T2CNT > CCRx_SHAD	0
111	PWM2 模式	T2CNT < CCRx_SHAD	0
		T2CNT > CCRx_SHAD	1
注：输出参考信号 OCxREF 为高电平有效，其与极性选择 T2CCxP 共同决定引脚 OCx 的实际输出值；			

表 7-27 T2OCxM 配置为输出比较模式

[9] 仅在通道 x 关闭时(即 T2CCxE = 0)可写，x = 1, 2, 3。

名称	状态		寄存器	地址	复位值
T2OCxPE	通道 x 输出比较值的自动预装载 1 = 使能 (T2CCRx 预装载值在更新事件到来时加载) 0 = 禁止 (T2CCRx 立即被加载)		TIM2CCMRx[3] x = 1, 2, 3	0x311/ 0x312/ 0x313	RW−0
T2CC1S [10]	通道 1 模式选择	00 = 输出 01 = 输入，输入脚映射在 TI1FP1 10 = 输入，输入脚映射在 TI2FP1 11 = 保留	TIM2CCMR1[1:0]	0x311	RW−00
T2CC2S [10]	通道 2 模式选择	00 = 输出 01 = 输入，输入脚映射在 TI2FP2 10 = 输入，输入脚映射在 TI1FP2 11 = 保留	TIM2CCMR2[1:0]	0x312	RW−00
T2CC3S [10]	通道 3 模式选择	00 = 输出 01 = 输入，输入脚映射在 TI3FP3 1x = 保留	TIM2CCMR3[1:0]	0x313	RW−00

表 7-28 TIM2CCMRx 作为输出配置寄存器

名称	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	地址	复位值
TIM2CCER1	–	–	T2CC2P	T2CC2E	–	–	T2CC1P	T2CC1E	0x314	RW---00 --00
TIM2CCER2	–	–	–	–	–	–	T2CC3P	T2CC3E	0x315	RW----- --00

名称	功能	输入捕获模式 (T2CCxS = 01/10)	输出比较模式 (T2CCxS = 00)
T2CCxP	通道 x 输入/输出极性选择	1 = 捕获发生在 TIxF 的下降沿或低电平 0 = 捕获发生在 TIxF 的上升沿或高电平	1 = OCx 低电平有效 0 = OCx 高电平有效
T2CCxE	通道 x 输入/输出引脚功能	1 = 使能引脚的输入捕获功能 0 = 禁止	1 = 使能引脚的 OCx 输出功能 0 = 禁止
注：通道输出电平由 T2OISx 和 T2CCxE 位的值共同决定；			

表 7-29 Timer2 通道输出和极性选择

名称	状态		寄存器	地址	复位值
TIM2_CH1	通道 1 引脚重映射	1 = PB0 0 = PA5	AFP0[4]	0x19F	RW−0
TIM2_CH3	通道 3 引脚重映射	1 = PA3 0 = PB5	AFP0[3]		RW−0

表 7-30 Timer2 通道管脚功能重映射

[10] 仅在通道 x 关闭时(即 T2CCxE = 0)可写，x = 1, 2, 3。

名称	状态		寄存器	地址	复位值
GIE	全局中断 1 = 使能 (PEIE, T2CCxIE, T2CCxG, T2UIE 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 (T2CCxIE, T2CCxG, T2UIE 适用) 0 = 关闭 (无唤醒)		INTCON[6]		RW−0
T2CC3IE	通道 3 捕获/比较中断	1 = 使能 0 = 关闭	TIM2IER[3]	0x30D	RW−0
T2CC2IE	通道 2 捕获/比较中断		TIM2IER[2]		RW−0
T2CC1IE	通道 1 捕获/比较中断		TIM2IER[1]		RW−0
T2CC3G [11]	通道 3 捕获/比较软件中断		TIM2EGR[3]	0x310	WO−0
T2CC2G [11]	通道 2 捕获/比较软件中断		TIM2EGR[2]		WO−0
T2CC1G [11]	通道 1 捕获/比较软件中断		TIM2EGR[1]		WO−0
T2CC3IF [12]	通道 x 捕获/比较中断标志位 • 输出模式： 1 = T2CNT 与 T2CCRx 值匹配 0 = 不匹配 • 输入模式： 1 = 计数器值已被捕获至 T2CCRx (读 T2CCRx 时自动清零) 0 = 无捕获产生		TIM2SR1[3]	0x30E	R_W1C−0
T2CC2IF [12]			TIM2SR1[2]		R_W1C−0
T2CC1IF [12]			TIM2SR1[1]		R_W1C−0
T2CC3OF [12]	通道 x 重复捕获中断标志位 1 = 发生重复捕获 (计数器值被捕获到 T2CCRx 寄存器时，T2CCxIF 已经置 1) 0 = 无重复捕获		TIM2SR2[3]	0x30F	R_W1C−0
T2CC2OF [12]			TIM2SR2[2]		R_W1C−0
T2CC1OF [12]			TIM2SR2[1]		R_W1C−0
T2UIE	允许更新中断	1 = 使能 0 = 关闭	TIM2IER[0]	0x30D	RW−0
T2UG [11]	允许更新软件中断		TIM2EGR[0]	0x310	WO−0
T2UIF [12]	更新事件中断标志位 1 = 发生更新事件 0 = 无更新事件		TIM2SR1[0]	0x30E	R_W1C−0

表 7-31 Timer2 中断使能和状态位

[11] 软件置 1，硬件自动清 0。

[12] 写 1 清零，写 0 无效。建议只使用 STR、MOVWI 指令进行写操作，而不要用 BSR 或 IOR 指令。

7.3.2. 计数基本单元

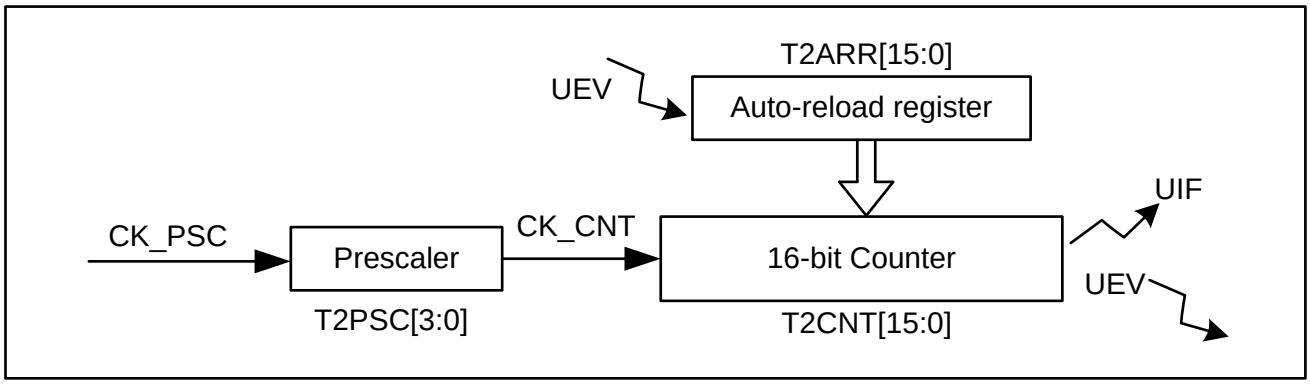


图 7-36 计数基本单元

TIM2 基本单元：

- 16 位向上计数器
- 4 位预分频器
- 16 位自动重载寄存器

向上计数模式：计数器从 0 开始向上计数，当 T2CNT = T2ARR 时，产生上溢事件，然后重新从 0 开始计数。

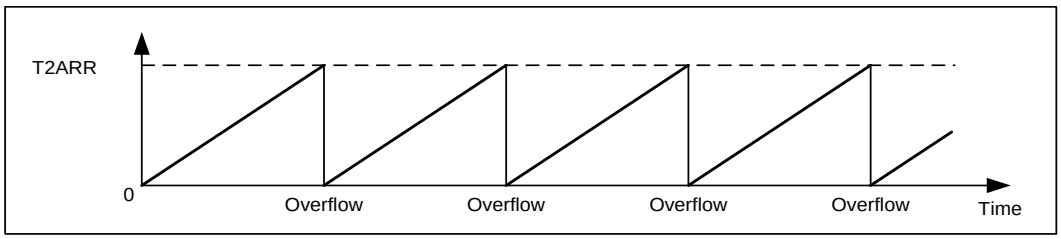


图 7-37 向上计数模式

预分频器、输出比较值和自动重载寄存器分别由预装载寄存器和影子寄存器组成。

	预分频器	输出比较值	自动重载寄存器
预装载使能位	当 T2CEN = 1 时默认使能	T2OCxPE	T2ARPE
预装载寄存器	T2PSC[3:0]	T2CCRx[15:0]	T2ARR[15:0]

表 7-32 有预装载功能的寄存器

TIM2 预分频时钟(CK_PSC)可选的 6 个时钟源(参阅 T2CKSRC)如下：

- Sysclk
- 1x or 2x HIRC
- LIRC
- 1x or 2x 外部时钟 (只有当 FOSC 相应地配置成 LP, XT 或 EC 模式时才有效)

4 位预分频器可对预分频时钟(CK_PSC)进行 1 ~ 32768 分频，从而产生计数时钟(CK_CNT)。

分频公式：$f_{CK_CNT} = f_{CK_PSC} / 2^{(PSCR[3:0])}$;　　(PSCR 为预分频器影子寄存器值)

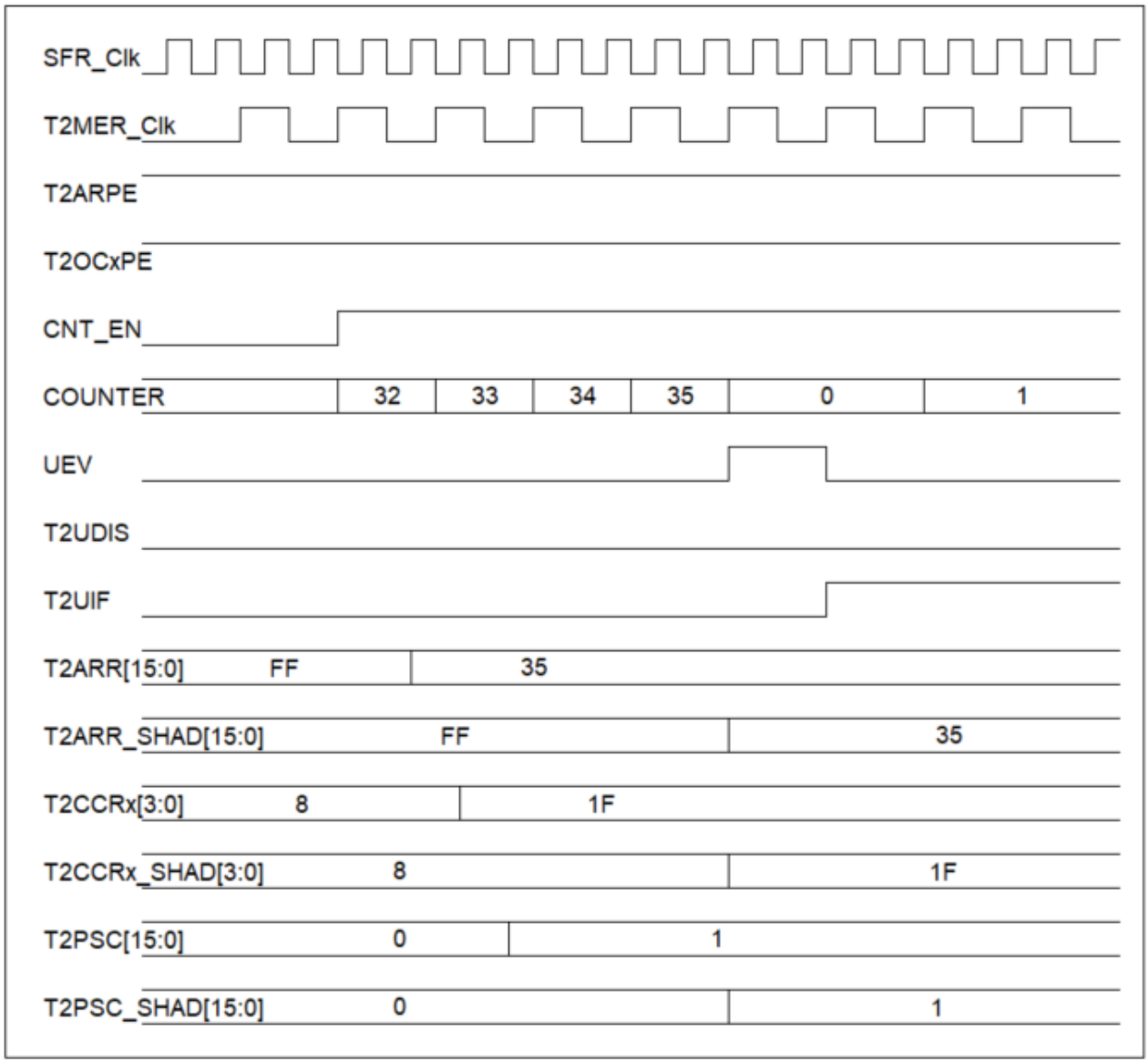
注：

1. 建议在计数器停止(T2CEN = 0)时，再对计数器值 T2CNT[15:0] 进行读写操作，避免出错。
2. 需先配置周期、输出比较值等寄存器，并在使能计数器(T2CEN = 1)之前配置预分频寄存器。

当 T2UDIS = 0 时，允许产生更新事件，更新事件源（参阅"T1URS"）如下：

- 计数器上溢

当产生更新事件时，更新事件标志位 T2UIF 置位，是否触发中断和/或从睡眠中唤醒则取决于相应的使能控制位(GIE, PEIE 和 T2UIE)。



- **图 7-38** 更新事件下，预装载寄存器的更新时序图

此外，根据不同配置，更新事件可触发以下情况：

1. 与预分频器、输出比较值和自动重载寄存器相关：

   1.1 当计数器使能(T2CEN = 1)，且其相应的预装载使能时(T2OCxPE / T2ARPE =1)，其影子寄存器将在产生更新事件时更新为预装载值，如**图 7-38**；

   1.3 当计数器关闭(T2CEN = 0)，或其相应的预装载关闭时(T2OCxPE / T2ARPE =0)，其影子寄存器将直接更新为预装载值；

2. 单次脉冲模式下，产生更新事件时，将自动关闭计数器(T2CEN = 0)，计数器停止计数；

7.3.3. 捕获/比较通道

TIM2 的 CH1~3 端口可配置为输入捕获或输出比较功能(参阅复用寄存器 TIM2CCMRx 的 T2CCxS 位)。T2CCRx 寄存器由一个预装载寄存器和一个影子寄存器组成。读写过程仅操作预装载寄存器。

- 输入捕获模式下：

  T2CCRx[15:0]为只读寄存器。当发生捕获事件时，捕获到的计数器值被写入到影子寄存器中，然后再复制到 T2CCRx 预装载寄存器中。

  读 T2CCRx[15:0]寄存器时，必须先读高 8 位，再读低 8 位。读高 8 位时，预装载寄存器被冻结，然后可读到正确的低 8 位。读完低 8 位后，预装载寄存器才能更新为最新一次的捕获值。

- 输出比较模式下：

  T2CCRx[15:0]为可读可写寄存器。写操作时 T2CCRx 预装载寄存器值被复制到影子寄存器中(参阅**章节 7.3.2**)，然后影子寄存器的内容和计数器进行比较。读操作时读出的值来自预装载寄存器。

7.3.3.1. 输入捕获模式

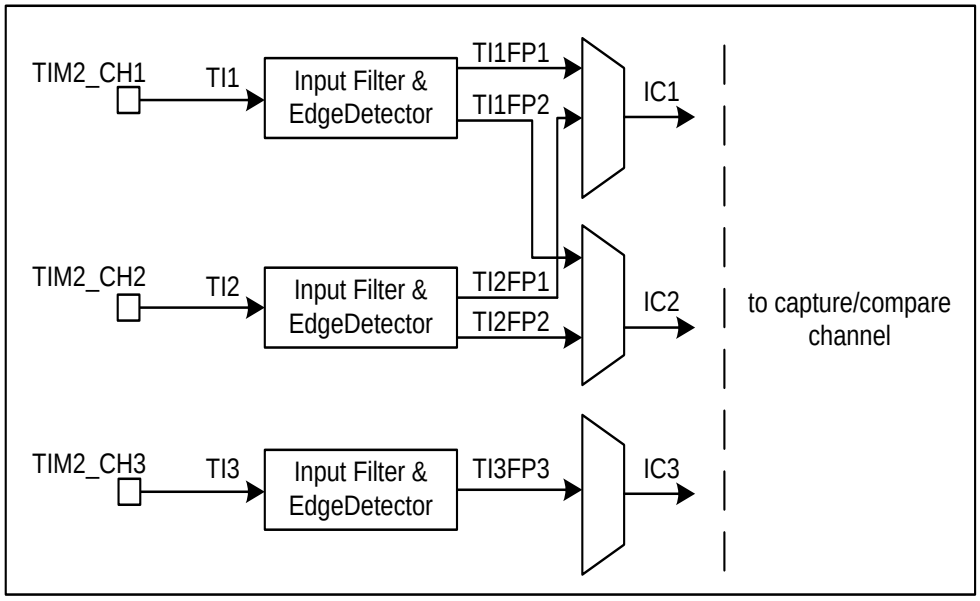


图 7-39 输入捕获通道框图

输入捕获模式下，当通道 x 发生输入捕获事件时，当前的计数值将被捕获到 T2CCRx[15:0]寄存器中，同

时输入捕获标志位 T2CCxIF 置位。如果当 T2CCxIF 保持为 1 时，再次发生输入捕获事件，则重复捕获标志位 T2CCxOF 将置位。是否触发中断和/或从睡眠中唤醒则取决于相应的使能控制位(GIE, PEIE 和 T2CCxIE)。此外，可使能输入捕获软件中断位 T2CCxG 来触发中断。

TIM2 各通道的输入捕获源(参阅 T2CCxS)如下：

T2CCxS	通道 1	通道 2	通道 4
01	TI1FP1	TI2FP2	TI3FP3
10	TI2FP1	TI1FP2	–

表 7-33 各通道输入捕获源

信号名称	详细说明
TIM2_CH1/2/3	通道 1/2/3/4 对应的 I/O 输入
IC1/2/3	通过选择后的通道捕获源
TI1FP1	通道 1 对应 I/O 的输入捕获信号，作为通道 1 的捕获源之一
TI1FP2	通道 1 对应 I/O 的输入捕获信号，作为通道 2 的捕获源之一
TI2FP2	通道 2 对应 I/O 的输入捕获信号，作为通道 2 的捕获源之一
TI2FP1	通道 2 对应 I/O 的输入捕获信号，作为通道 1 的捕获源之一
TI3FP3	通道 3 对应 I/O 的输入捕获信号，作为通道 3 的捕获源

表 7-34 输入捕获信号说明

输入捕获通道的配置步骤请参阅 TIM1(**章节 7.2.4.1**)。

7.3.3.2. 输出比较模式

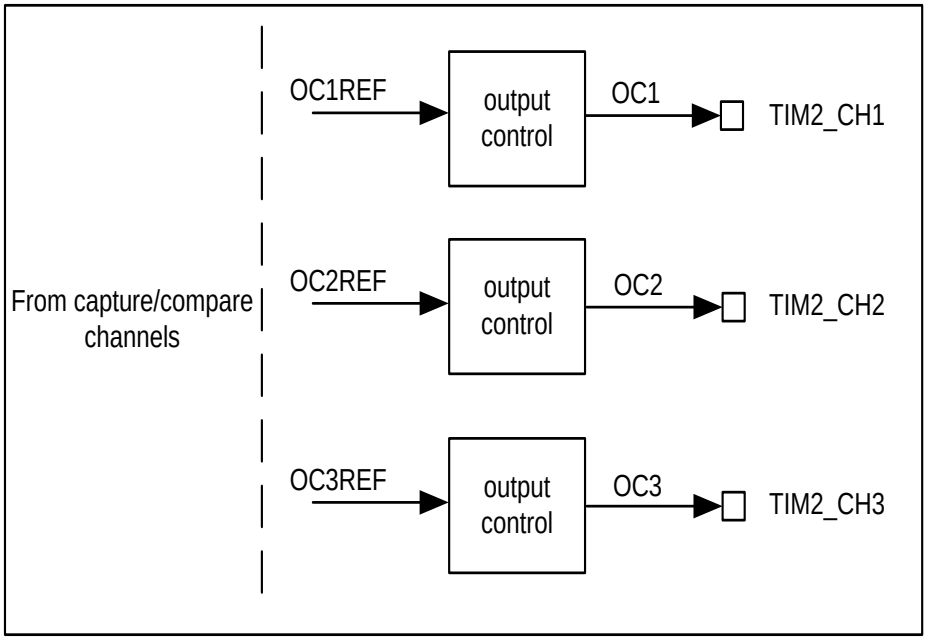


图 7-40 输出比较通道框图

输出比较模块通过对比计数值(T2CNT)与比较值(影子寄存器 CCRx_SHAD)，先产生输出参考信号 OCxREF(高电平有效)，再经过极性选择和输出控制位(T2CCxE) 将波形输出到端口。

参考信号 OCxREF 可通过 T2OCxM[2:0] 配置为 8 种输出模式 (参阅**表 7-27**)：

1. 冻结模式(T2OCxM = 000)：OCxREF 值保持不变；
2. 匹配有效(T2OCxM = 001)：当 T2CNT = CCRx_SHAD 时，OCxREF = 1；
3. 匹配无效(T2OCxM = 010)：当 T2CNT = CCRx_SHAD 时，OCxREF = 0；
4. 翻转模式(T2OCxM = 011)：当 T2CNT = CCRx_SHAD 时，OCxREF 值翻转；
5. 强制无效(T2OCxM = 100)：OCxREF 一直为 0；
6. 强制有效(T2OCxM = 101)：OCxREF 一直为 1；
7. PWM1 模式(T2OCxM = 110)：

a) 当 T2CNT < CCRx_SHAD 时，OCxREF = 1；当 T2CNT > CCRx_SHAD 时，OCxREF = 0；

8. PWM2 模式(T2OCxM = 111)：

a) 当 T2CNT < CCRx_SHAD 时，OCxREF = 0；当 T2CNT > CCRx_SHAD 时，OCxREF = 1；

当 T2CNT 与 CCRx_SHAD 匹配时，输出比较标志位 T1CCx2F 置位。是否触发中断和/或从睡眠中唤醒则取决于相应的使能控制位(GIE, PEIE, T2CCxIE)。此外，可使能输出比较软件中断位 T2CCxG 来触发中断。

输出比较通道的配置步骤请参阅 TIM1(**章节 7.2.4.2**)。

PWM 模式 – PWM1/PWM2 周期由 T2ARR 决定、占空比由 T2CCRx 决定。

公式 7-3 *PWM1/2 周期* $= (T2ARR+1) * T_{CK_CNT}$

公式 7-4 *PWM1/2 占空比* $= T2CCRx \div (T2ARR+1)$

TIM1_CH1/2/3 通道可独立使能输出 PWM 信号，输出信号极性可选(参阅 T2CCxP)。

单脉冲模式 – 单脉冲模式下(T2OPM = 1)，当产生下一次更新事件时，硬件会自动关闭计数器使能位(T2CEN = 0)，计数器停止计数。

为产生一个正确的脉冲，计数器初始值(T2CNT) 必须与比较值(T2CCRx) 不同。即启动计数前，需满足以下配置：

- 向上计数模式：T2CNT < T2CCRx ≤ T2ARR

7.4. 基本定时器 4 (TIMER4)

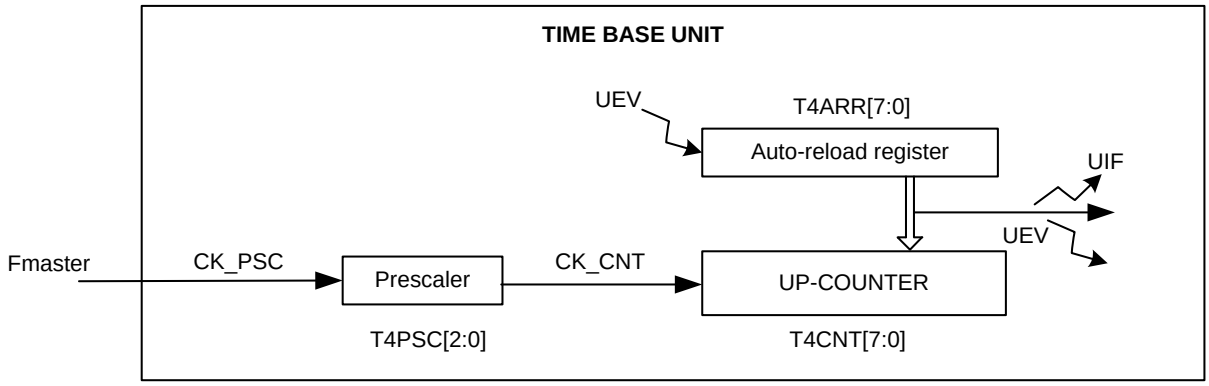


图 7-41 TIM4 原理框图

TIM4 为 8 位向上计数器：计数器从 0 开始向上计数，当 T4CNT = T4ARR 时，产生上溢事件，然后重新从 0 开始计数。自动重载寄存器 T4ARR 由预装载寄存器和影子寄存器组成。

TIM4 预分频时钟(CK_PSC)可选的 4 个时钟源(参阅 T4CKSRC)如下：

- Sysclk
- HIRC
- 外部时钟 LP/XT (只有当 FOSC 相应地配置成 LP 或 XT 模式时才有效)

3 位预分频器可对预分频时钟(CK_PSC)进行 1 ~ 128 分频，从而产生计数时钟(CK_CNT)。

分频公式：$f_{CK_CNT} = f_{CK_PSC} / 2^{(PSCR[2:0])}$;　　(PSCR 为预分频器影子寄存器值)

注：

1. 建议在计数器停止(T4CEN = 0)时，再对计数器值 T4CNT[7:0] 进行读写操作，避免出错。
3. 需先配置周期等寄存器，并在使能计数器(T4CEN = 1)之前配置预分频寄存器。

当 T4UDIS = 0 时，允许产生更新事件，更新事件源（参阅"T1URS"）如下：

- 计数器上溢
- 软件设置 T4UG 位或计数器上溢

当产生更新事件时，更新事件标志位 T4UIF 置位，是否触发中断和/或从睡眠中唤醒则取决于相应的使能控制位(GIE, PEIE 和 T4UIE)。

此外，根据不同配置，更新事件可触发以下情况：

1. 与预分频器寄存器相关：
   - 1.2 当计数器使能(T4CEN = 1)，且其相应的预装载使能时(T4ARPE =1)，其影子寄存器将在产生更新事件时更新为预装载值；
   - 1.4 当计数器关闭(T4CEN = 0)，或其相应的预装载关闭时(T4ARPE =0)，其影子寄存器将直接更新为预装载值；

2 单次脉冲模式下，产生更新事件时，将自动关闭计数器(T4CEN = 0)，计数器停止计数；

7.4.1. Timer4 相关寄存器汇总

名称	状态		寄存器	地址	复位值
TIM4EN	TIM4 模块时钟	1 = 使能 0 = 关闭	PCKEN[3]	0x9A	RW−0
SYSON	睡眠模式下，系统时钟控制	1 = 使能 0 = 关闭	CKOCON[7]	0x95	RW−0
T4ARPE	计数周期的自动预装载 1 = 使能 (T4ARR 预装载值在更新事件到来时被加载) 0 = 禁止 (T4ARR 立即被加载)		TIM4CR1[7]	0x111	RW−0
T4CKS	Timer4 时钟源 00 = Sysclk 10 = LP (*) 01 = HIRC 11 = XT (*) (*) FOSC 需相应配置成 LP/XT 模式，否则振荡器将不会运行。		TIM4CR1[5:4]		RW−00
T4OPM	单脉冲模式 1 = 使能 (下一次更新事件到来时，T4CEN 自动清零，计数器停止) 0 = 关闭 (发生更新事件时，计数器不停止)		TIM4CR1[3]		RW−0
T4URS	当 T4UDIS = 0 时，更新事件中断源 1 = 计数器上溢 0 = 软件设置 T4UG 位或计数器上溢		TIM4CR1[2]		RW−0
T4UDIS	产生更新事件控制	1 = 禁止 0 = 允许	TIM4CR1[1]		RW−0
T4CEN	TIM4 计数器	1 = 使能 0 = 关闭	TIM4CR1[0]		RW−0
T4PSC	Timer4 预分频器 000 = 1 100 = 16 001 = 2 101 = 32 010 = 4 110 = 64 011 = 8 111 =128		TIM4PSCR[2:0]	0x116	RW-000
T4CNT	Timer4 计数值		TIM4CNTR[7:0]	0x115	RW−0000 0000
T4ARR	计数周期的自动重载寄存器(预装载值) 注：此值为 0 时，计数器不工作		TIM4ARR[7:0]	0x117	RW−1111 1111

表 7-35 Timer4 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, T4UIE, T4UG 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 (T4UIE, T4UG 适用) 0 = 关闭 (无唤醒)		INTCON[6]		RW-0
T4UIE	允许更新中断	1 = 使能 0 = 关闭	TIM4IER[0]	0x112	RW-0
T4UG [13]	允许更新软件中断		TIM4EGR[0]	0x114	WO-0
T4UIF [14]	更新事件中断标志位	1 = 发生更新事件 0 = 无更新事件	TIM4SR[0]	0x113	R_W1C-0

表 7-36 Timer4 中断使能和状态位

名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	复位值
PCKEN	0x9A	—	—	UARTEN	—	TIM4EN	TIM2EN	TIM1EN	ADCEN	--0- 0000
CKOCON	0x95	SYSON	CCORDY	DTYSEL		CCOSEL[2:0]			CCOEN	0010 0000
TIM4CR1	0x111	T4ARPE	—	T4CKS[1:0]		T4OPM	T4URS	T4UDIS	T4CEN	0-00 0000
TIM4IER	0x112	—	—	—	—	—	—	—	T4UIE	---- ---0
TIM4SR	0x113	—	—	—	—	—	—	—	T4UIF	---- ---0
TIM4EGR	0x114	—	—	—	—	—	—	—	T4UG	---- ---0
TIM4CNTR	0x115	T4CNT[7:0]								0000 0000
TIM4PSCR	0x116	—	—	—	—	—	T4PSC[2:0]			---- -000
TIM4ARR	0x117	T4ARR[7:0]								1111 1111

表 7-37 Timer4 相关用户寄存器汇总（—保留位必须保持为复位值，不可更改）

[13] 软件置 1，硬件自动清 0。

[14] 写 1 清零，写 0 无效。建议只使用 STR、MOVWI 指令进行写操作，而不要用 BSR 或 IOR 指令。

8. SLEEP 睡眠模式 (POWER-DOWN)

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 8-1 所示，FT61F14x 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、ADC 和 USART 能在 SLEEP 模式下保持运行。有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 (配置 LVREN)	使能或通过指令控制 (SLVREN = 1)	非 SLEEP 模式下使能
LVD	LVDEN = 1	No
WDT	WDTE or SWDTEN	No
TIMER1	SYSON = 1 & TIM1EN = 1 & T1CEN = 1	SYSON = 0
TIMER2	SYSON = 1 & TIM2EN = 1 & T2CEN = 1	SYSON = 0
TIMER4	SYSON = 1 & TIM4EN = 1 & T4CEN = 1	SYSON = 0
时钟输出	SYSON = 1 & CCOEN = 1	SYSON = 0
ADC	SYSON = 1 & ADCEN = 1 & ADON = 1	Yes: SYSON = 0 & ADCS ≠ x11 No: ADCS = x11
USART	SYSON = 1 & UARTEN = 1 & TXEN / RXEN = 1	SYSON = 0
PWM	(跟随 Timer1 或 Timer2)	
HIRC / LIRC / EC / LP / XT	(跟随使用它们的外设状态)	
I/O	(除非 SLEEP 时使能 PWM、时钟输出或其他外设，否则 I/O 将保持其进入 SLEEP 前的状态)	

表 8-1 除指令时钟外，其他模块可根据需求在 SLEEP 模式下保持运行

8.1. 进入 SLEEP

CPU 通过执行 SLEEP 指令进入睡眠模式。进入睡眠时：

1. 如果 WDT 使能，则 WDT 的预分频器和定时器将被清零，并重新开始计时。
2. 超时标志位 (/TO) = 1。
3. 掉电标志位 (/PD) = 0。
4. 时钟源
   - 当 SYSON = 0 时，指令时钟自动关闭；
   - 当 SYSON = 1 时，指令时钟保持运行，使能模块系统时钟(参阅 PCKEN)的相应外设及其所选时钟源(HIRC, LIRC, EC, LP 或 XT)也将保持运行；
   - 当 SYSON = 1 时，时钟输出所选时钟源(参阅 CCOSEL)将保持运行，时钟输出继续；
5. I/O 端口
   - 当 SYSON = 1 & TIMxEN = 1 & TxCEN = 1 时，即 Timers 在 SLEEP 下保持运行，则 PWM 输出也将保持。当 SYSON = 0 时，Timers 自动关闭，那么 PWM 输出将保持其进入 SLEEP 前的状态。

- 当 SYSON = 1，且 ADC 或 USART 的模块系统时钟和模块功能同时使能时，则模块功能既可保持运行。当 SYSON = 0 时，ADC (ADCS ≠ x11) 或 USART 将自动关闭，但 ADC 时钟源选择 LIRC (ADCS = x11) 除外。
- 其他数字输出端口，将保持其进入 SLEEP 之前的状态(高阻态，低电平或高电平)。

有关各个外设在 SLEEP 状态下如何工作的详细信息，请查阅相应章节。

注：

1. 如果在执行 SLEEP 指令之前发生中断(中断标志位置 1 且其中断使能，但全局中断 GIE 禁止)，则 SLEEP 指令将作为 NOP 执行，不会进入睡眠模式。
2. 由于同步延时，将中断标志位清 0 之后，至少要等两条指令才可以执行 SLEEP 指令，否则不会进入睡眠模式。

8.2. 从 SLEEP 中唤醒

从睡眠中唤醒有 2 个基本原则：

- 基于时间，即 CPU 在一定的时间后醒来。建议选择 LIRC 作为计时的时钟源，因为 LIRC 比 HIRC 的功耗更低。
- 基于事件，即触发 POR，系统复位，仅唤醒而不产生中断，以及产生中断的事件，如 LVD，ADC，EEPROM 写完成，外部管脚中断。

从睡眠中唤醒，分为以下几种情况：

1. 如果使能，看门狗定时器可触发唤醒 (参阅 **章节 7.1** 看门狗定时器)。
2. 完全复位和系统复位：
   - POR 完全复位 (不能关闭)
   - 通过 /MCLR 进行外部系统复位 (如果使能)
   - LVR 复位 (如果使能)
3. 使能中断 (关闭“全局中断使能”不影响唤醒功能)。请参阅 **章节 9** 中断。

注：

1. 从睡眠中唤醒将清零 WDT。
2. <u>紧跟 SLEEP 指令后必须写为 NOP 指令</u>

   使用非中断方式 (即未执行“中断服务程序”) 从 SLEEP 中唤醒时，比如 WDT 唤醒或全局中断控制位(GIE)未使能时的中断事件唤醒，将继续执行下一条指令。

   使用中断方式从 SLEEP 中唤醒时，将先执行下一条指令，然后再调用中断服务程序。为了避免重复执行，建议紧跟 SLEEP 的后面写为 NOP 指令。

   SLEEP
   NOP // 中断方式唤醒

9. 中断 (INTERRUPTS)

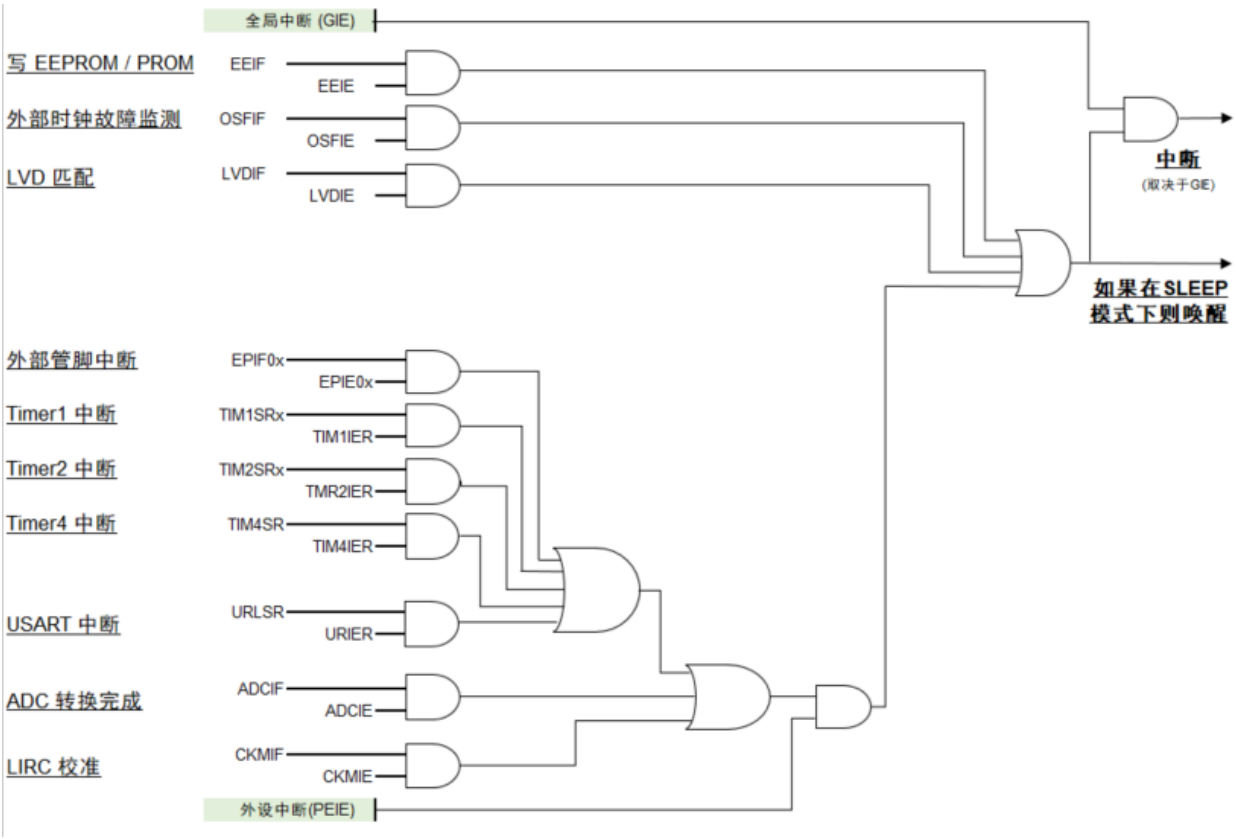


图 9-1 中断结构框图

CPU 支持 10 个中断源，分为 2 组：

1) 非外设中断

- DATA EEPROM / PROM 写完成
- 故障保护时钟监控器
- LVD 条件匹配

2) 外设中断

- 外部管脚中断
- Timer1 中断
- Timer2 中断
- Timer4 中断
- USART 中断
- LIRC 和 HIRC 交叉校准完成
- ADC 转换完成

与其他 Timers 不同，WDT 溢出不会产生中断。除外部 I/O 中断外，其他中断请参阅相应章节。

产生中断时，PC 跳转并执行 ”中断服务程序(ISR)”。中断的关闭/使能有多层控制：

- 每个中断源均有其独立的中断使能位：EEIE, LVDIE, OSFIE, CKMIE, ADCIE, TIM1IER, TIM2IER, TIM4IER, URIER。
- 同时最多有 8 个外部管脚中断输入，共用一个端口中断使能位：EPIE0x (External PORTx Interrupt Enable)。
- 外设中断有一个总中断使能位：PEIE (PEripheral Interrupt Enable)。
- 如果关闭以上所有控制位，将不会执行从睡眠中唤醒。
- 所有中断都由全局中断使能位控制：GIE (Global Interrupt Enable)。与其他使能位不同，当关闭全局中断使能位时，依然允许从睡眠中唤醒。
- 关闭中断使能位并不影响中断标志位的置位。

中断处理时序如下：

- 自动设置 “GIE = 0”，从而关闭中断。
- 返回地址被压入堆栈，程序指针 PC 加载 0x0004 地址。
- 发生中断后的 3 – 5 个指令周期，跳转到 ”中断服务程序(ISR)” 开始处理中断。
- 执行 “从中断返回(RETI)” 指令退出 ISR。在 RETI 之前必须清除当前所处理的中断标志位。
- 当 ISR 完成时，PC 返回到中断前的地址，如果在 SLEEP 模式下，则返回到 SLEEP 指令后紧跟的地址。
- 在执行 RETI 时自动设置 “GIE = 1”，从而使能中断。

注： 中断过程中，返回的 PC 地址自动保存在堆栈中，其他重要的寄存器值 [W, STATUS (/TO 和/PD 除外), BSREG, FSR, PCLATH] 自动保存在相应的影子寄存器 (可读写，参阅 bank31) 中。退出中断服务程序时，这些寄存器值将自动从影子寄存器中恢复。如果用户需要保存其他寄存器值，必须通过指令将这些值正确地写入临时寄存器中，建议使用 SRAM 的最后 16 个 bytes 作为临时寄存器，因为所有 bank 共用这 16 个 bytes，而不需要切换 bank 以节省代码。

9.1. 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	复位值(RW)
INTCON	0x0B	GIE	PEIE	EEIE	LVDIE	OSFIE	EEIF	LVDIF	OSFIF	0000 0000
PIE1	0x91	–	–	–	–	–	–	CKMIE	ADCIE	---- --00
PIR1	0x11	–	–	–	–	–	–	CKMIF	ADCIF	---- --00
TIM1IER	0x215	T1BIE	T1TIE	–	T1CC4IE	T1CC3IE	T1CC2IE	T1CC1IE	T1UIE	00-0 0000
TIM1SR1	0x216	T1BIF	T1TIF	–	T1CC4IF	T1CC3IF	T1CC2IF	T1CC1IF	T1UIF	00-0 0000
TIM1SR2	0x217	–	–	–	T1CC4OF	T1CC3OF	T1CC2OF	T1CC1OF	–	---0 000-
TIM1EGR	0x218	–	–	–	T1CC4G	T1CC3G	T1CC2G	T1CC1G	–	---0 000-
TIM2IER	0x30D	–	–	–	–	T2CC3IE	T2CC2IE	T2CC1IE	T2UIE	---- 0000
TIM2SR1	0x30E	–	–	–	–	T2CC3IF	T2CC2IF	T2CC1IF	T2UIF	---- 0000
TIM2SR2	0x30F	–	–	–	–	T2CC3OF	T2CC2OF	T2CC1OF	–	---- 000-
TIM2EGR	0x310	–	–	–	–	T2CC3G	T2CC2G	T2CC1G	T2UG	---- 0000
TIM4IER	0x112	–	–	–	–	–	–	–	T4UIE	---- ---0
TIM4SR	0x113	–	–	–	–	–	–	–	T4UIF	---- ---0
TIM4EGR	0x114	–	–	–	–	–	–	–	T4UG	---- ---0
URIER	0x48E	–	–	TCEN	–	IDELE	RXSE	URTE	URRXNE	--0- 0000
URLSR	0x492	ADDRF	IDLEF	TXEF	BKF	FEF	PEF	OVERF	RXNEF	0010 0000
URTC	0x49C	–	–	–	–	–	–	–	TCF	---- ---1
EPIE0	0x94	外部中断控制寄存器								0000 0000
EPIF0	0x14	外部中断标志位寄存器								0000 0000
TRISA	0x8C	PORTA 方向控制								1111 1111
TRISB	0x8D	PORTB 方向控制								1111 1111
TRISC	0x8E	PORTC 方向控制								1111 1111
EPS0	0x118	外部中断 EINT3 ~ 0 管脚选择寄存器								0000 0000
EPS1	0x119	外部中断 EINT7 ~ 4 管脚选择寄存器								0000 0000

表 9-1 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
EEIE	EEPROM / PROM 写完成中断	1 = 使能 0 = 关闭 (无唤醒)	INTCON[5]		RW−0
LVDIE	LVD 中断		INTCON[4]		RW−0
OSFIE	外部振荡器故障中断		INTCON[3]		RW−0
EEIF [1]	EEPROM / PROM 写完成标志位	1 = Yes (锁存) 0 = No	INTCON[2]		R_W1C−0
LVDIF [1]	LVD 中断标志位		INTCON[1]		R_W1C−0
OSFIF [1]	外部振荡器故障中断标志位		INTCON[0]		R_W1C−0

表 9-2 INTCON 寄存器

名称	状态		寄存器	地址	复位值
CKMIE	LIRC 和 HIRC 交叉校准完成中断	1 = 使能 0 = 关闭 (无唤醒)	PIE1[1]	0x91	RW−0
ADCIE	ADC 转换完成中断		PIE1[0]		RW−0
CKMIF [1]	LIRC 和 HIRC 交叉校准完成标志位	1 = Yes (锁存) 0 = No	PIR1[1]	0x11	R_W1C−0
ADCIF [1]	ADC 转换完成标志位		PIR1[0]		R_W1C−0

表 9-3 PIE1 和 PIR1 寄存器

[1] 写 1 清零，写 0 无效。建议只使用 STR、MOVWI 指令进行写操作，而不要用 BSR 或 IOR 指令。

名称	状态		寄存器	地址	复位值
T1BIE	Timer1 刹车中断	1 = 使能 0 = 关闭 (无唤醒)	TIM1IER[7]	0x215	RW-0
T1TIE	Timer1 触发中断		TIM1IER[6]		RW-0
T1COMIE	Timer1 换相 (COM) 中断		TIM1IER[5]		RW-0
T1CC4IE	Timer1 捕获/比较通道 4 中断		TIM1IER[4]		RW-0
T1CC3IE	Timer1 捕获/比较通道 3 中断		TIM1IER[3]		RW-0
T1CC2IE	Timer1 捕获/比较通道 2 中断		TIM1IER[2]		RW-0
T1CC1IE	Timer1 捕获/比较通道 1 中断		TIM1IER[1]		RW-0
T1UIE	Timer1 更新事件中断		TIM1IER[0]		RW-0
T1BG [2]	Timer1 刹车软件中断		TIM1EGR[7]	0x218	WO-0
T1TG [2]	Timer1 触发软件中断		TIM1EGR[6]		WO-0
T1COMG [2]	Timer1 换相 (COM) 软件中断		TIM1EGR[5]		WO-0
T1CC4G [2]	Timer1 捕获/比较通道 4 软件中断		TIM1EGR[4]		WO-0
T1CC3G [2]	Timer1 捕获/比较通道 3 软件中断		TIM1EGR[3]		WO-0
T1CC2G [2]	Timer1 捕获/比较通道 2 软件中断		TIM1EGR[2]		WO-0
T1CC1G [2]	Timer1 捕获/比较通道 1 软件中断		TIM1EGR[1]		WO-0
T1UG [2]	Timer1 更新事件软件中断		TIM1EGR[0]		WO-0
T2CC3IE	Timer2 捕获/比较通道 3 中断		TIM2IER[3]	0x30D	RW-0
T2CC2IE	Timer2 捕获/比较通道 2 中断		TIM2IER[2]		RW-0
T2CC1IE	Timer2 捕获/比较通道 1 中断		TIM2IER[1]		RW-0
T2UIE	Timer2 更新事件中断		TIM2IER[0]		RW-0
T2CC3G [2]	Timer2 捕获/比较通道 3 软件中断		TIM2EGR[3]	0x310	WO-0
T2CC2G [2]	Timer2 捕获/比较通道 2 软件中断		TIM2EGR[2]		WO-0
T2CC1G [2]	Timer2 捕获/比较通道 1 软件中断		TIM2EGR[1]		WO-0
T2UG [2]	Timer2 更新事件软件中断		TIM2EGR[0]		WO-0
T4UIE	Timer4 更新事件中断		TIM4IER[0]	0x112	RW-0
T4UG [2]	Timer4 更新事件软件中断		TIM4EGR[0]	0x114	WO-0

表 9-4 TIMx 中断控制寄存器

[2] 软件置 1，硬件自动清 0。

名称	状态		寄存器	地址	复位值
T1BIF [3]	Timer1 刹车中断标志位	1 = Yes (锁存) 0 = No	TIM1SR1[7]	0x216	R_W1C-0
T1TIF [3]	Timer1 触发事件中断标志位		TIM1SR1[6]		R_W1C-0
T1COMIF [3]	Timer1 换相 (COM) 中断标志位		TIM1SR1[5]		R_W1C-0
T1CC4IF [3]	Timer1 捕获/比较通道 4 中断标志位		TIM1SR1[4]		R_W1C-0
T1CC3IF [3]	Timer1 捕获/比较通道 3 中断标志位		TIM1SR1[3]		R_W1C-0
T1CC2IF [3]	Timer1 捕获/比较通道 2 中断标志位		TIM1SR1[2]		R_W1C-0
T1CC1IF [3]	Timer1 捕获/比较通道 1 中断标志位		TIM1SR1[1]		R_W1C-0
T1UIF [3]	Timer1 更新事件中断标志位		TIM1SR1[0]		R_W1C-0
T1CC4OF [3]	Timer1 捕获/比较通道 4 重复捕获中断标志位		TIM1SR2[4]	0x217	R_W1C-0
T1CC3OF [3]	Timer1 捕获/比较通道 3 重复捕获中断标志位		TIM1SR2[3]		R_W1C-0
T1CC2OF [3]	Timer1 捕获/比较通道 2 重复捕获中断标志位		TIM1SR2[2]		R_W1C-0
T1CC1OF [3]	Timer1 捕获/比较通道 1 重复捕获中断标志位		TIM1SR2[1]		R_W1C-0
T2CC3IF [3]	Timer2 捕获/比较通道 3 中断标志位		TIM2SR1[3]	0x30E	R_W1C-0
T2CC2IF [3]	Timer2 捕获/比较通道 2 中断标志位		TIM2SR1[2]		R_W1C-0
T2CC1IF [3]	Timer2 捕获/比较通道 1 中断标志位		TIM2SR1[1]		R_W1C-0
T2UIF [3]	Timer2 更新事件中断标志位		TIM2SR1[0]		R_W1C-0
T2CC3OF [3]	Timer2 捕获/比较通道 3 重复捕获中断标志位		TIM2SR2[3]	0x30F	R_W1C-0
T2CC2OF [3]	Timer2 捕获/比较通道 2 重复捕获中断标志位		TIM2SR2[2]		R_W1C-0
T2CC1OF [3]	Timer2 捕获/比较通道 1 重复捕获中断标志位		TIM2SR2[1]		R_W1C-0
T4UIF [3]	Timer4 更新事件中断标志位		TIM4SR[0]	0x113	R_W1C-0

表 9-5 TIMx 中断标志寄存器

[3] 写 1 清零，写 0 无效。建议只使用 STR、MOVWI 指令进行写操作，而不要用 BSR 或 IOR 指令。

名称	状态		寄存器	地址	复位值
URTE	发送 BUF 为空中断	1 = 使能 0 = 关闭	URIER[1]	0x48E	RW-0
TXEF	发送 BUF 状态	1 = 空 0 = 非空 注：写 DATAL(8bit) / DATAH(9bit) 清零	URLSR[5]	0x492	RO-1
URRXNE	接收 BUF 非空中断	1 = 使能 0 = 关闭	URIER[0]	0x48E	RW-0
RXNEF	接收 BUF 状态	1 = 非空 0 = 空，或已被清零 注：读 DATAL(8bit) / DATAH(9bit) 清零	URLSR[0]	0x492	RO-0
TCEN	发送完成中断	1 = 使能 0 = 关闭	URIER[5]	0x48E	RW-0
TCF	发送完成标志	1 = 完成 0 = 未完成 注：写 1 清零，或写 DATAL(8bit) /DATAH(9bit)后清零	URTC[0]	0x49C	R_W1C-1
IDELE	空闲帧中断	1 = 使能 0 = 关闭	URIER[3]	0x48E	RW-0
IDLEF [4]	检测到空闲帧标志	1 = 检测到 0 = 未检测到	URLSR[6]	0x492	RW0-0
RXSE [5]	接收状态中断	1 = 使能 0 = 关闭 接收状态中断产生条件： BKF = 1 FEF = 1 PEF = 1 OVERF = 1	URIER[2]	0x48E	RW-0
BKF [4]	接收到断开帧标志	1 = 接收到 0 = 未接收到，或已被清零	URLSR[4]	0x492	RW0-0
FEF [4]	接收到帧错误标志	1 = 错误 0 = 正确，或已被清零	URLSR[3]	0x492	RW0-0
PEF [4]	接收到奇偶校验错误标志	1 = 错误 0 = 正确，或已被清零	URLSR[2]	0x492	RW0-0

[4] 写 0 清零，写 1 无效。

[5] USART 接收到断开帧，帧错误，奇偶校验错误，接收溢出错误状态。

名称	状态		寄存器	地址	复位值
OVERF [6]	接收 BUF 溢出标志	1 = 溢出 0 = 正常，或已被清零	URLSR[1]	0x492	RW0-0
WAKE	哑模式唤醒方式选择	1 = 地址匹配 0 = IDLE 帧	URMCR[2]	0x491	RW-0
ADDRF	哑模式地址匹配标志	1 = 匹配 0 = 未匹配	URLSR[7]	0x492	RO-0

表 9-6 USART 中断使能和状态位

名称	状态	寄存器	地址	复位值
EINT0	00 = PA0 01 = PB0 10 = PC0 11 = 保留	EPS0[1:0]	0x118	RW-00
EINT1	00 = PA1 01 = PB1 10 = PC1 11 = 保留	EPS0[3:2]		RW-00
EINT2	00 = PA2 01 = PB2 1x = 保留	EPS0[5:4]		RW-00
EINT3	00 = PA3 01 = PB3 1x = 保留	EPS0[7:6]		RW-00
EINT4	00 = PA4 01 = PB4 1x = 保留	EPS1[1:0]	0x119	RW-00
EINT5	00 = PA5 01 = PB5 1x = 保留	EPS1[3:2]		RW-00
EINT6	00 = PA6 01 = PB6 1x = 保留	EPS1[5:4]		RW-00
EINT7	00 = PA7 01 = PB7 1x = 保留	EPS1[7:6]		RW-00

表 9-7 外部中断管脚选择寄存器

名称	状态		寄存器	地址	复位值
ITYPE0[1:0]	PORTx.0	外部中断管脚 EINTx 触发类型 00 = 低电平 01 = 上升沿 10 = 下降沿 11 = 双边沿	ITYPE0[1:0]	0x11E	RW-00
ITYPE0[3:2]	PORTx.1		ITYPE0[3:2]		RW-00
ITYPE0[5:4]	PORTx.2		ITYPE0[5:4]		RW-00
ITYPE0[7:6]	PORTx.3		ITYPE0[7:6]		RW-00
ITYPE1[1:0]	PORTx.4		ITYPE1[1:0]	0x11F	RW-00
ITYPE1[3:2]	PORTx.5		ITYPE1[3:2]		RW-00
ITYPE1[5:4]	PORTy.6		ITYPE1[5:4]		RW-00
ITYPE1[7:6]	PORTy.7		ITYPE1[7:6]		RW-00

表 9-8 外部管脚中断触发类型寄存器 (x = A, B, C； y = A, B)

[6] 写 0 清零，写 1 无效。

名称	状态		寄存器	地址	复位值
EPIE0x	外部管脚中断	1 = 使能 0 = 禁止	EPIE0[7:0]	0x94	RW−00000000
EPIF0x [7]	外部管脚中断标志位	1 = Yes (锁存) 0 = No	EPIF0[7:0]	0x14	R_W1C−00000000

表 9-9 外部管脚中断使能和标志寄存器

9.2. 外部管脚中断

所有 I/O 均支持外部管脚中断，同时最多有 8 个外部管脚中断输入 (参阅 EINT0~7)，同时 I/O 需设置成输入(TRISA/B/C[x] = 1，且 ANSELA[x] = 0)，触发源可选上升沿、下降沿、双边沿和低电平方式 (参阅 ITYPEx)。

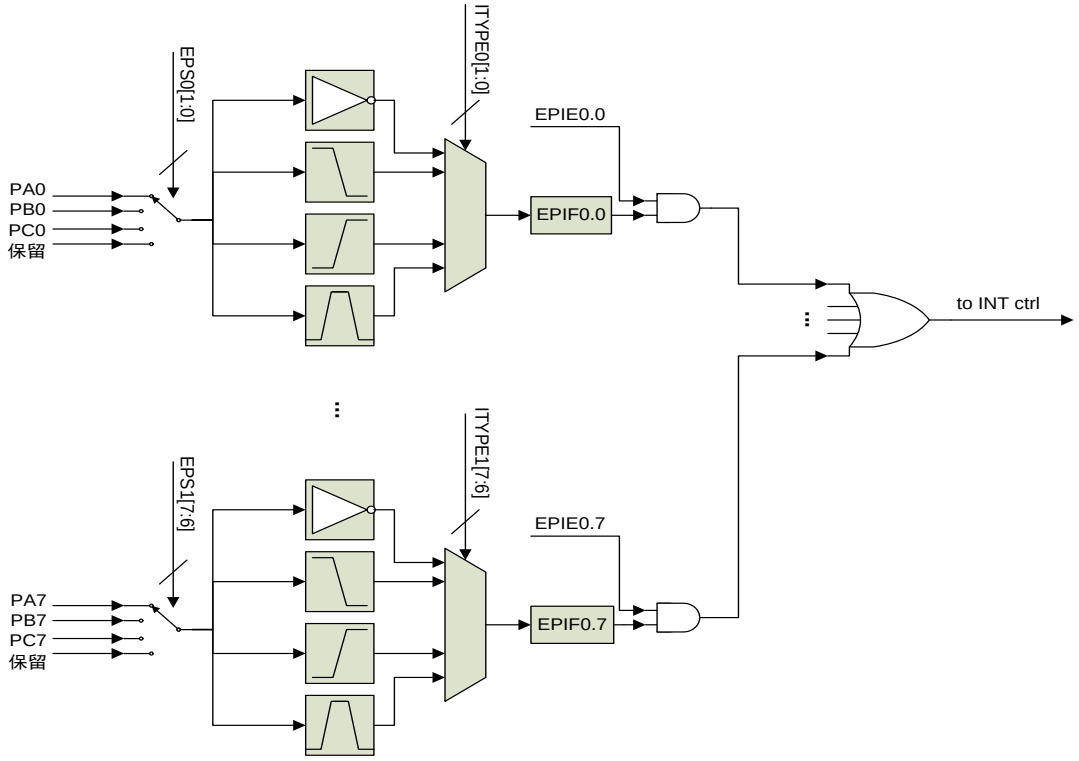


图 9-2 外部管脚中断结构框图

[7] 写 1 清零，写 0 无效。建议只使用 STR、MOVWI 指令进行写操作，而不要用 BSR 或 IOR 指令。

10. 数据 EEPROM 和程序 PROM

FT61F14x 片内集成的非易失性 DATA EEPROM 存储区和程序存储器 PROM 均可通过指令进行读/写访问，由 ”CFGS” 和 ”EEPGD” 选择所访问的存储区。128 x 8−bit 的 DATA EEPROM 和 4k x 14−bit (128 page x 32 words) 的程序 PROM 相互独立。

DATA EEPROM 存储区的典型擦写次数可达 100 万次。地址范围为 0x00 ~ 0x7F，每次可读取或写入的单位为 1 个 byte (8−bit)，没有页模式(page mode)。程序 PROM 存储区的典型擦写次数为 10 万次。地址范围为 0x0000 ~ 0x0FFF，每次可读取或写入的单位为 1 个 word (14−bit)，

DATA EEPROM 擦除/编程实了硬件自定时，无需软件查询，以节省有限的代码空间，因此写操作可在后台运行，不影响 CPU 执行其他指令，甚至可进入 SLEEP 状态。但 PROM 擦除/编程时，CPU 将停止执行指令。

读操作需要 1 个指令时钟周期，而 EEPROM 写操作需要的时间为 $T_{WRITE\text{-}EEPROM}$ (3 ~ 5 ms)，PROM 擦除和写操作的时间分别为 $T_{ERASE\text{-}PROM}$ (0.75 ~ 1.25 ms)和 $T_{WRITE\text{-}PROM}$ (0.75 ~ 1.25 ms)。芯片内置有电荷泵，因此不需要提供外部高压，即可对 EEPROM 和 PROM 区进行擦除和编程。编程前，DATA EEPROM 可配置成自动擦除，但 PROM 必须由软件擦除。EEPROM 写操作完成时将置位相应的中断标志位 EEIF。

不支持连续读(sequential READ) 或连续写(sequential WRITE)，因此每次读/写都必须更新相应的地址。

只要 $V_{DD} \geq V_{POR}$，CPU 即可在 8 MHz / 2T 的速度下运行，在高温下甚至可低至 1.5V 左右。而写 DATA EEPROM 和 PROM 所需的电压($V_{DD\text{-}WRITE}$) 较高。对于 DATA EEPROM，工作温度等级 2 和等级 1 的最低 $V_{DD\text{-}WRITE}$ 分别为 1.9V 和 2.2V。对于 PROM，最低 $V_{DD\text{-}WRITE}$ 为 2.7V。读 DATA EEPROM 和 PROM 没有此最低电压限制(参阅 $V_{DD\text{-}READ}$)。

10.1. DATA EEPROM 和 PROM 相关寄存器汇总

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 / PROM 写完成中断	1 = 使能 0 = 关闭 (无唤醒)	INTCON[5]		RW−0
EEIF [1]	EEPROM / PROM 写完成中断标志位	1 = Yes (锁存) 0 = No	INTCON[2]		R_W1C−0

表 10-1 EEPROM 和 PROM 中断使能和状态位

[1] 写 1 清零，写 0 无效。建议只使用 STR、MOVWI 指令进行写操作，而不要用 BSR 或 IOR 指令。

名称	状态		寄存器	地址	复位值
SYSON	睡眠模式下，系统时钟控制	1 = 使能 0 = 关闭	CKOCON[7]	0x95	RW−0
EEADR [2]	当 EEPGD = 0 时	DATA EEPROM 地址	EEADRL[7:0]	0x191	RW−0000 0000
	当 EEPGD = 1 时	PROM 地址低 8 位			
		PROM 地址高 4 位	EEADRH[3:0]	0x192	RW−---- 0000
EEDAT [2]	当 EEPGD = 0 时	DATA EEPROM 数据	EEDATL[7:0]	0x193	RW−xxxx xxxx
	当 EEPGD = 1 时	PROM 数据低 8 位			
		PROM 数据高 6 位	EEDATH[5:0]	0x194	RW−xx xxxx
EEPGD	当 CFGS = 0 时	1 = 访问 PROM 0 = 访问 DATA EEPROM	EECON1[7]	0x195	RW−0
CFGS	1 = 访问配置寄存器 (读访问) 0 = 访问 PROM 或 DATA EEPROM		EECON1[6]		RW−0
FREE	下一条 WR 命令所执行的 PROM 操作 1 = 擦除 (擦除完成后由硬件清零) 0 = 写 注：仅当 CFGS = 0 且 EEPGD = 1 时有效		EECON1[4]		RW−0
WRERR	PROM / DATA EEPROM 擦除/写错误标志位 1 = 中止 (除 POR 之外的任何复位) 0 = 正常完成		EECON1[3]		RW−x
WREN	PROM / DATA EEPROM 写使能位	1 = 使能 0 = 禁止	EECON1[2]		RW−0
WR	PROM / DATA EEPROM 写控制位 1 = 启动一次写或写正在进行中 (置 1 后至少等 1 个 Sysclk 才能回读，写完成后自动清零) 0 = 完成		EECON1[1]		RW1−0
RD	PROM / DATA EEPROM 读控制位 1 = Yes (保持 4 个指令周期后自动清零) 0 = No		EECON1[0]		RW1−0
EECON2	PROM / DATA EEPROM 写操作解锁控制 必须先写 0x55，再写 0xAA，用于解锁操作，然后将 WR 置 1。 注：这些写操作必须在连续的指令周期内完成，不能被打断；		EECON2[7:0]	0x196	WO−xxxx xxxx

表 10-2 EEPROM 和 PROM 相关用户控制寄存器

[2] 在写周期 (参阅 $T_{WRITE\text{-}EEPROM}$ 和 $T_{WRITE\text{-}PROM}$) 内，该寄存器不可写。

10.2. DATA EEPROM

10.2.1. 写 DATA EEPROM

1. 设置“GIE = 0”；
2. 判断 GIE，如果“GIE = 1”，则重复步骤 (1)；
3. 将目标地址写入 EEADRL；
4. 将目标数据写入 EEDATL；
5. 设置“CFGS = 0”和“EEPGD = 0”，选择访问 DATA EEPROM；
6. 设置“WREN = 1”；
7. 向 EECON2 中顺序写入 0x55 和 0xAA；
8. 设置“WR = 1”以启动写；
9. 编程完成 (编程时间请参阅 $T_{WRITE\text{-}EEPROM}$) 后，”WR”将由硬件自动清 0；

示例程序：

```
BCR INTCON, GIE
NOP
NOP                          ; 中断响应延时为 2 个 NOP
BTSC INTCON, GIE
LJUMP $-4

BANKSEL EEADRL
LDWI DATA_EE_ADDR
STR EEADRL                   ; 写入目标地址
LDWI DATA_EE_DATA
STR EEDATL                   ; 写入目标数据
BCR EECON1, CFGS
BCR EECON1, EEPGD            ; 选择访问 DATA EEPROM
BSR EECON1, WREN

LDWI 55H
STR EECON2                   ; 向 EECON2 写入 0x55
LDWI AAH
STR EECON2                   ; 向 EECON2 写入 0xAA

BSR EECON1, WR               ; 启动写
BSR INTCON, GIE              ; GIE 置 1
BCR EECON1, WREN             ; 关闭写使能
BTSC EECON1, WR
LJUMP $-2
```

注：

1. 当编程正在进行中时，对 DATA EEPROM 进行读操作将导致读取结果错误。
2. 在启动 DATA EEPROM 的写操作前，需要先解锁，即向 EECON2 中顺序写入 0x55 和 0xAA，且不能被打断，因此解锁前需关闭所有中断。
3. GIE 清零后，需等待 2 个 NOP 的中断响应延时，再次判断 GIE 是否被清零。
4. WR 置 1 后，需至少等待一个 Sysclk (NOP 或任何其他指令)，软件才能读到正确的 WR 值，否则将读回 0 (误认为写结束)。
5. WR 置 1 后，清零 WREN 不会影响当前的写周期。
6. 当写 DATA EEPROM 结束后，WREN 需由软件清零，此保护机制可防止误写操作。此外，上电延时定时器 PWRT(~64ms)也会阻止对 DATA EEPROM 进行写操作。

10.2.2. 读 DATA EEPROM

1. 设置“GIE = 0”；
2. 判断 GIE，如果“GIE = 1”，则重复步骤 (1)；
3. 将目标地址写入 EEADRL；
4. 设置“CFGS = 0”和“EEPGD = 0”，选择访问 DATA EEPROM；
5. 设置“RD = 1”以启动读；
6. 等待 4 个指令周期后，从 EEDATL 中读取目标数据。EEDATL 寄存器将保持此值直至下一次读或写操作。”RD”将由硬件自动清 0；

读 DATA EEPROM 的示例程序如下：

```
BCR INTCON, GIE
NOP
NOP                          ; 中断响应延时为 2 个 NOP
BTSC INTCON, GIE
LJUMP $-4

BANKSEL EEADRL
LDWI DATA_EE_ADDR
STR EEADRL                   ; 写入目标地址
BCR EECON1, CFGS
BCR EECON1, EEPGD            ; 选择访问 DATA EEPROM
BSR EECON1, RD               ; 启动读
NOP
NOP
NOP
NOP
LDR EEDATL, W                ; 此时，数据可由指令读取
```

注：无论 CPB 为何值，软件总是可以读取 EEPROM。

10.3 程序 PROM

程序地址计数器 PC 为 15 位(0x0000 ~ 0x7FFF)，最多支持 32k 地址空间。FT61F14x 实现了 4k 的程序 PROM，共分为 128 page x 32 words (1 word = 14 bits)，地址范围为 0x0000 ~ 0x0FFF，当程序地址超过 0x0FFF 将导致回卷到 0x0000。

软件需先对程序 PROM 进行擦除，再执行编程操作。

注：

1. 当初始化配置寄存器 (参阅 FSECPB0) 设置成扇区加密时，被加密的 PROM 扇区只能读，不能擦除或写。

10.3.1 擦除程序 PROM

软件擦除 PROM 的单位为 1 page (32 words)。

1. 设置 “GIE = 0”；
2. 判断 GIE，如果 “GIE = 1”，则重复步骤 (1)；
3. 将目标地址写入 EEADRL 和 EEADRH；
4. 设置 “CFGS = 0” 和 “EEPGD = 1”，选择访问 PROM；
5. 设置 “FREE = 1” 以及 “WREN = 1”；
6. 向 EECON2 中顺序写入 0x55 和 0xAA；
7. 设置 “WR = 1” 以启动擦除；
8. 擦除完成 (擦除时间 $T_{ERASE\text{-}PROM}$ 为 0.75 ~ 1.25 ms) 后，”FREE” 将由硬件自动清 0；

示例程序：

```
BCR INTCON, GIE
NOP
NOP                         ; 中断响应延时为 2 个 NOP
BTSC INTCON, GIE
LJUMP $-4

BANKSEL EEADRL
LDR ADDRL,W
STR EEADRL                  ; 写入目标地址低 8 位
LDR ADDRH,W
STR EEADRH                  ; 写入目标地址高 7 位
BCR EECON1, CFGS
BCR EECON1, EEPGD           ; 选择访问 PROM
BSR EECON1, FREE
BSR EECON1, WREN

LDWI 55H
STR EECON2                  ; 向 EECON2 写入 0x55
LDWI AAH
```

```
STR EECON2              ; 向 EECON2 写入 0xAA

BSR EECON1, WR          ; 启动擦除
NOP
NOP                     ; 设置擦除操作需等待 2 个 NOP
BCR EECON1, WREN        ; 关闭写使能
BSR INTCON, GIE         ; GIE 置 1
```

注：

1. 在启动 PROM 的擦除操作前，需要先解锁，即向 EECON2 中顺序写入 0x55 和 0xAA，且不能被打断，因此解锁前需关闭所有中断。
2. WR 置 1 后，处理器设置擦除操作需要 2 个指令周期，因此必须在擦除指令之后紧跟 2 条 NOP 指令。
3. 擦除周期 $T_{ERASE\text{-}PROM}$ 内，CPU 将暂停执行指令，时钟和外设将继续运行。
4. 擦除完成后，程序将从擦除指令后的第三条指令继续执行。

10.3.2 写程序 PROM

软件编程 PROM 的单位为 1 word (14 bits)，因此 1 page 需要重复编程 32 次。编程前，对应地址需已擦除或未编程过。

1. 设置 “GIE = 0”；
2. 判断 GIE，如果 “GIE = 1”，则重复步骤 (1)；
3. 将目标地址写入 EEADRL 和 EEADRH；
4. 设置 “CFGS = 0” 和 “EEPGD = 1”，选择访问 PROM；
5. 设置 “FREE = 0” 以及 “WREN = 1”；
6. 将目标数据写入 EEDATL 和 EEDATH；
7. 向 EECON2 中顺序写入 0x55 和 0xAA；
8. 设置 “WR = 1” 以启动写；
9. 写完成 (编程时间 $T_{WRITE\text{-}PROM}$ 为 0.75 ~ 1.25 ms)后，”WR” 将由硬件自动清 0；
10. 重复以上步骤，直至写入所有数据；

示例程序 (目标数据通过间接寻址加载)：

```
BCR INTCON, GIE
NOP
NOP                     ; 中断响应延时为 2 个 NOP
BTSC INTCON, GIE
LJUMP $-4

BANKSEL EEADRL
LDR ADDRL,W
STR EEADRL              ; 写入目标地址低 8 位
```

```
LDR ADDRH,W
STR EEADRH              ; 写入目标地址高 7 位
LDWI LOW DATA_ADDR      ; 加载目标数据的地址低 8 位
STR FSR0L
LDWI HIGH DATA_ADDR     ; 加载目标数据的地址高 8 位
STR FSR0H
BCR EECON1, CFGS
BCR EECON1, EEPGD       ; 选择访问 PROM
BCR EECON1, FREE
BSR EECON1, WREN

MOVIW FSR++
STR EEDATL              ; 写入目标数据低 8 位
MOVIW FSR++
STR EEDATH              ; 写入目标数据高 6 位

LDWI 55H
STR EECON2              ; 向 EECON2 写入 0x55
LDWI AAH
STR EECON2              ; 向 EECON2 写入 0xAA

BSR EECON1, WR          ; 启动写
NOP
NOP                     ; 设置写操作需等待 2 个 NOP
BCR EECON1, WREN        ; 关闭写使能
BSR INTCON, GIE         ; GIE 置 1
```

注：

1. 将目标数据写入 EEDATL 和 EEDATH 时，将被加载到 14-bit 的写锁存器。写操作完成后，写锁存器将复位为 0x3FFF。
2. 在启动 PROM 的写操作前，需要先解锁，即向 EECON2 中顺序写入 0x55 和 0xAA，且不能被打断，因此解锁前需关闭所有中断。
3. WR 置 1 后，处理器设置写操作需要 2 个指令周期，因此必须在写指令之后紧跟 2 条 NOP 指令。
4. 写周期 $T_{WRITE\text{-}PROM}$ 内，CPU 将暂停执行指令，时钟和外设将继续运行。
5. 写完成后，程序将从写指令后的第三条指令继续执行。
6. 当需要更改某些 PROM 数据，且当前 page 的其他数据需要保留时，按以下步骤修改：
   a) 加载要修改的 page 的起始地址；
   b) 读取当前 page 所有数据并保存到 RAM 映射区；
   c) 修改 RAM 映射区需要更改的新数据；
   d) 加载要修改的 page 的起始地址，并执行擦除操作，以擦除当前 page；
   e) 按编程步骤，将 RAM 映射区的数据逐一写入当前 page；

10.3.3 读程序 PROM

1. 设置“GIE = 0”；
2. 判断 GIE，如果“GIE = 1”，则重复步骤 (1)；
3. 将目标地址写入 EEADRL 和 EEADRH；
4. 设置“CFGS = 0”和“EEPGD = 1”，选择访问 PROM；
5. 设置“RD = 1”以启动读；
6. 等待 2 个指令周期后，PROM 数据被写入 EEDATH:EEDATL 寄存器，因而必须在读指令之后紧跟 2 条 NOP 指令。”RD”将由硬件自动清 0。EEDATH:EEDATL 寄存器将保持此值直至下一次读或写操作；

读 PROM 的示例程序如下：

```
BCR INTCON, GIE
NOP
NOP                          ; 中断响应延时为 2 个 NOP
BTSC INTCON, GIE
LJUMP $-4

BANKSEL    EEADRL
LDWI PROG_ADDR_LO
STR EEADRL                   ; 写入目标地址低 8 位
LDWI PROG_ADDR_HI
STR EEADRH                   ; 写入目标地址高 7 位
BCR EECON1, CFGS
BSR EECON1, EEPGD            ; 选择访问 PROM
BSR EECON1, RD               ; 启动读
NOP
NOP                          ; 读等待 2 个 NOP
LDR EEDATL, W                ; 读取数据低 8 位
STR PROG_DATA_LO
LDR EEDATH, W                ; 读取数据高 6 位
STR PROG_DATA_HI
```

注意：

1. 无论 CPB 为何值，软件总是可以读取 PROM。

10.4 读初始化配置寄存器 UCFGx

当 CFGS = 1 时，软件可读取初始化配置寄存器区 UCFGx (具体请参阅 **章节 14.1**)。UCFGx 和程序 PROM 相互独立，地址从 0x8000 开始，对于未实现的单元，读返回未定义。

11 12-bit 模/数转换器 (ANALOG TO DIGITAL CONVERTER, ADC)

ADC 模块可将模拟输入信号转换成 12-bit 的数字信号。ADC 可在不同的时钟速度下运行，并且在高达 2MHz 的时钟速度下 (即 100 kHz 的采样率，10 μs/采样) 仍具有 11-bit 精度。

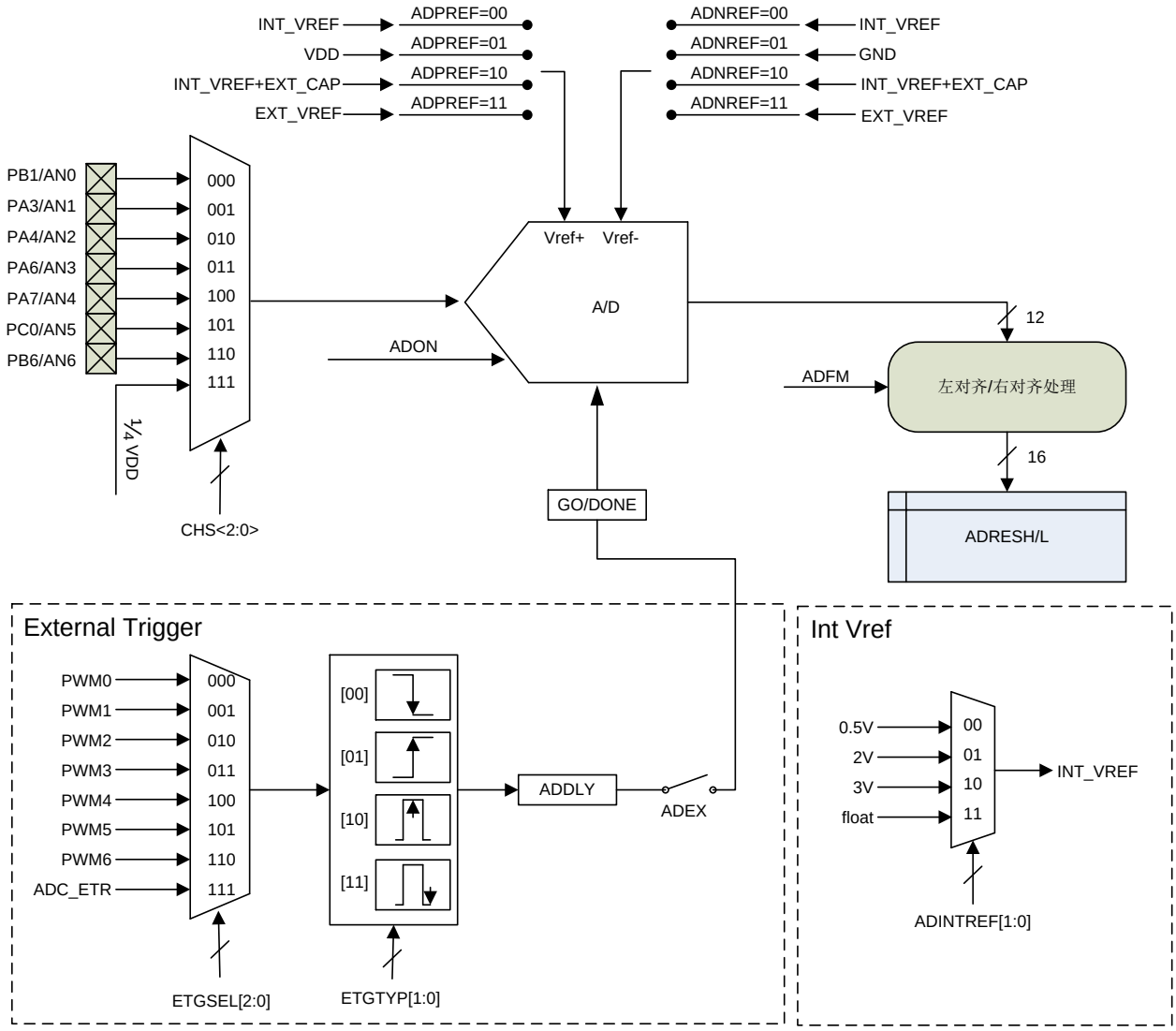


图 11-1 ADC 结构框图

模拟输入信号可选择为 7 个 I/O (ANx) 通道之一或 1 个内部通道 (1/4VDD)。ADC 由指令、I/O (PA4 / PB3) 或 PWM 触发。在触发和 ADC 采样之间可增加延时或前沿消隐(Leading Edge Blanking, LEB)。

当 ADC 转换完成时，将置位相应的中断标志位，并可触发中断和/或从睡眠中唤醒。

ADC 参考电压($V_{ADC-REF}$) 可通过指令选择为 V_{DD}，3 个内部参考电压(0.5V, 2V, 3V) 之一，或通过 I/O 输入外部参考电压。

ADC 不需要校准。另外，ADC 转换过程在后台运行，转换期间 CPU 可执行其他指令。

如果 ADC 需要在 SLEEP 下保持运行：

1. 需设置 SYSON = 1，以使 Sysclk 保持运行；

2. 当 ADC 转换时钟源为 LIRC 时，进入 SLEEP 后 LIRC 将保持运行，与 SYSON 无关；

当 ADC 配置为硬件触发(PA4/PB3 或 PWM)时，GO/DONE 由硬件触发事件直接置位并启动 A/D 转换，软件置位 GO/DONE 将被忽略。

在高采样率的应用中，使用 ADC 时需注意 3 个时间点：

1. 所选通道开始采样的时刻。
2. 结束采样的时刻。采样保持电路断开前的瞬间，所选通道上的电压值被用于测量转换。
3. 数据转换完成时间。

11.1 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
PCKEN	0x9A	–	–	UARTEN	–	TIM4EN	TIM2EN	TIM1EN	ADCEN	––00 0000
CKOCON	0x95	SYSON	CCORDY	DTYSEL		CCOSEL[2:0]			CCOEN	0010 0000
ADRESL	0x9B	A/D 转换结果低有效位								xxxx xxxx
ADRESH	0x9C	A/D 转换结果高有效位								xxxx xxxx
ADCON0	0x9D	–	CHS[2:0]			–	ADEX	GO/DONE	ADON	–000 –000
ADCON1	0x9E	ADFM	ADCS[2:0]			ADNREF[1:0]		ADPREF[1:0]		0000 0000
ADCON2 [1]	0x9F	ADINTREF[1:0]		ETGTYP[1:0]		ADDLY.8	ETGSEL[2:0]			0000 0000
ADDLY [1]	0x1F	ADDLY[7:0] / LEBPRL[7:0]								0000 0000
ADCON3 [1]	0x41A	ADFBEN	ADCMPOP	ADCMPEN	ADCMPO	LEBADT	–	ELVDS[1:0]		0000 0–00
ADCMPH	0x41B	ADCMPH[7:0]								0000 0000
LEBCON [1]	0x41C	LEBEN	LEBCH		–	EDGS	BKS2	BKS1	BKS0	000– 0000

表 11-1 ADC 相关用户寄存器地址

名称	状态		寄存器	地址	复位值
GIE	全局中断	1 = 使能 (PEIE, ADCIE 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 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]	0x91	RW–0
ADCIF [2]	ADC 转换完成中断标志位	1 = Yes (锁存) 0 = No	PIR1[0]	0x11	R_W1C–0

表 11-2 ADC 中断使能和状态位

[1] 当 ADCEN = 0 时，此寄存器仍可读写。

[2] 写 1 清零，写 0 无效。建议只使用 STR、MOVWI 指令进行写操作，而不要用 BSR 或 IOR 指令。

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]	0x9B	RW−0000 0000
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]	0x9C	RW−0000 0000
SYSON	睡眠模式下，系统时钟控制 1 = 使能 0 = 关闭	CKOCON[7]	0x95	RW−0
ADCEN	ADC 模块时钟 1 = 使能 0 = 关闭	PCKEN[0]	0x9A	RW−0
CHS	ADC 模拟输入通道 000 = AN0 100 = AN4 001 = AN1 101 = AN5 010 = AN2 110 = AN6 011 = AN3 111 = 1/4 V_{DD}	ADCON0[6:4]	0x9D	RW−000
ADEX	ADC 触发条件 (GO/DONE) 1 = 由 PA4/PB3 或 PWM 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, PA4/PB3 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	TCKSRC[7]	0X31F	RW−0
ADFM	A/D 转换结果格式 (参阅 “ADRESH”) 1 = 右对齐 0 = 左对齐	ADCON1[7]	0x9E	RW−0
ADCS	ADC 转换时钟源 000 = SysClk/2 100 = SysClk/4 001 = SysClk/8 101 = SysClk/16 010 = SysClk/32 110 = SysClk/64 011 = LIRC 111 = LIRC	**ADCON1[6:4]**		RW−000

名称	状态	寄存器	地址	复位值
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 C_{EXT} 11 = 外部参考电压 (I/O)	ADCON1[3:2]	0x9E	RW-00
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 C_{EXT} 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]	0x9F	RW-00
ETGTYP	外部触发沿 (当 ADEX=1 时适用) 00 = (PWM 或 PA4/PB3-ADC_ETR) 下降沿 01 = (PWM 或 PA4/PB3-ADC_ETR) 上升沿 10 = 一个 PWM 周期的中点 (*) 11 = 一个 PWM 周期的终点 (*) (*)：默认选择 TIM1 的中心对齐 PWM 模式；	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 = PWM1, TIM1_CH1 100 = PWM5, TIM2_CH1 001 = PWM2, TIM1_CH2 101 = PWM6, TIM2_CH2 010 = PWM3, TIM1_CH3 110 = PWM7, TIM2_CH3 011 = PWM4, TIM1_CH4 111 = ADC_ETR	ADCON2[2:0]		RW-000
ADDLY / LEBPRL	ADC 延迟/ LEB (非软件触发，即 ADEX = 1 时有效) (此为低 8 位, ADDLY.8 为高有效位) 延迟时间 = (ADDLY+6) x T_{AD} (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]	0x1F	RW-0000 0000

名称	状态	寄存器	地址	复位值
ADFBEN	ADC 阈值比较结果匹配事件触发 PWM 故障刹车 1 = 使能 0 = 关闭	ADCON3[7]	0x41A	RW-0
ADCMPOP	ADC 阈值比较的极性 1 = ADC 结果的高 8 位 < ADCMPH[7:0] 0 = ADC 结果的高 8 位 ≥ ADCMPH[7:0]	ADCON3[6]		RW-0
ADCMPEN	ADC 阈值比较 1 = 使能 0 = 关闭 (将清除 ADCMP 产生的刹车事件)	ADCON3[5]		RW-0
ADCMPO	ADC 比较结果输出位 (每次 A/D 转换完成均更新) 当 ADCMPOP = 1 时 1 = ADC 结果的高 8 位 < ADCMPH[7:0] (不锁存) 0 = ADC 结果的高 8 位 ≥ ADCMPH[7:0] 当 ADCMPOP = 0 时 1 = ADC 结果的高 8 位 ≥ ADCMPH[7:0] (不锁存) 0 = ADC 结果的高 8 位 < ADCMPH[7:0]	ADCON3[4]		RO-0
LEBADT	LEB 结束后，ADC 开始自动转换 1 = 触发 ADC 转换 0 = 不触发 ADC 转换	ADCON3[3]		RW-0
ADCMPH	ADC 比较阈值 (仅高 8 位，0.4% steps)	ADCMPH[7:0]	0x41B	RW-0000 0000
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]	0x41C	RW-0
LEBCH	LEB 信号源 00 = TIM1_CH1 10 = TIM1_CH3 01 = TIM1_CH2 11 = TIM1_CH4	LEBCON[6:5]		RW-00
EDGS	LEB 触发沿 1 = 下降沿 0 = 上升沿	LEBCON[3]		RW-0

表 11-3 ADC 相关用户寄存器

11.2 ADC 配置

配置 ADC 包括以下设置 (更改配置时需设置 ADON = 0 以关闭 A/D 转换或外部触发)：

- ADC 模块时钟
- 通道选择
- ADC 参考电压
- ADC 转换时钟源
- 转换结果格式
- 触发源
- ADC 延时或前沿消隐 (LEB)
- 阈值比较 (可选)
- 响应 (中断设置)

通道选择 – 由 CHS 寄存器选择输入通道，连接到用于 ADC 转换的采样保持电路。相应的 I/O 需设置 TRISx = 1 和 ANSELAx = 1 来配置成模拟输入。

ADC 参考电压($V_{ADC-REF}$) – ADC 以 2 个参考电压作为相对值来测量输入模拟电压：V_{REF}+和 V_{REF}–。参考电压可以选择为：

- V_{REF}+可选 VDD，V_{REF}–可选 GND
- 内部参考电压
- 内部参考电压加外部电容 C_{EXT}
- 外部参考电压 (V_{REF}+为 PA1, V_{REF}–为 PA0)

V_{REF}+和 V_{REF}–可以为上述选择的不同组合，但不可以同时选择内部参考电压，否则 V_{REF}–将强制连接到 GND。

内部参考电压可以为 0.5V，2.0V，3.0V 或 ”未连接” (参阅 ”ADINTREF”)。

ADC 转换时钟选择 – ADC 可通过指令选择 7 种时钟频率 (参阅 ”ADCS”)：

- SysClk/N；N = 2, 4, 8, 16, 32, 64
- LIRC (256 kHz 或 32 kHz，参阅 “LFMOD”)

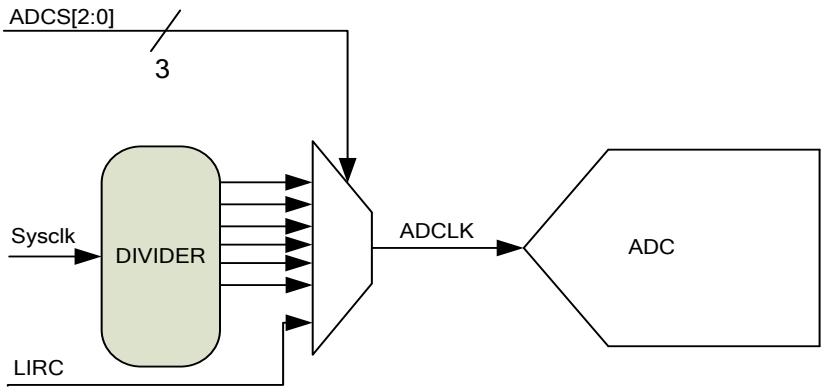


图 11-2 ADC 时钟配置

转换结果格式 – A/D 转换结果可储存为左对齐或右对齐两种格式 (参阅 ”ADFM”)。A/D 自动校准值也受此格式影响。

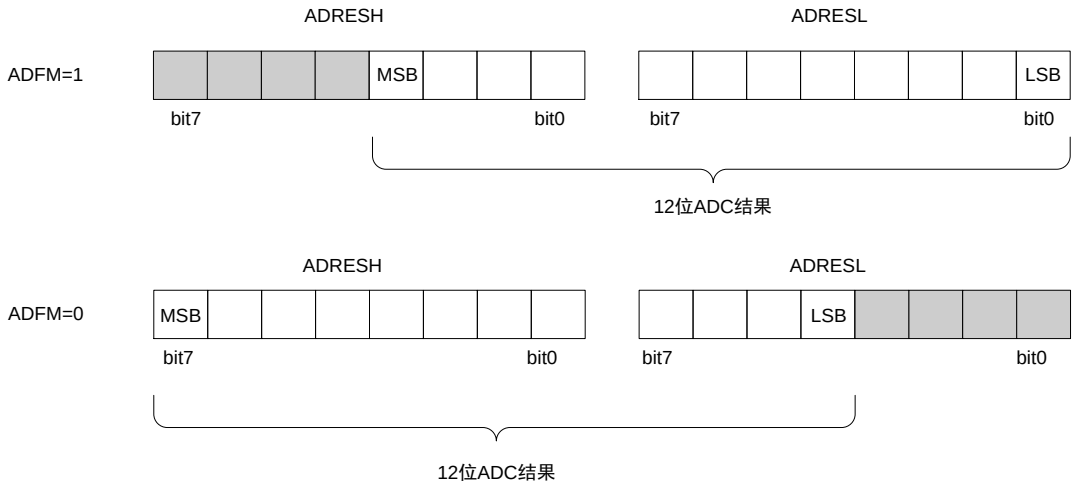


图 11-3 ADC 转换结果格式

11.2.1 ADC 触发和延时配置

ADC 转换可由指令 (ADEX = 0)、PWM (边沿/周期) 或 I/O (PA4/PB3) 转变沿 (ADEX = 1) 触发。其中，PWM 的触发类型可选择为”上升沿”、”下降沿”、”周期的中点” 或 ”周期的终点”，PA4/PB3 的触发沿可选择为 ”上升沿” 或 ”下降沿” (参阅 ”ETGTYP” 和 ”ETGSEL”)。

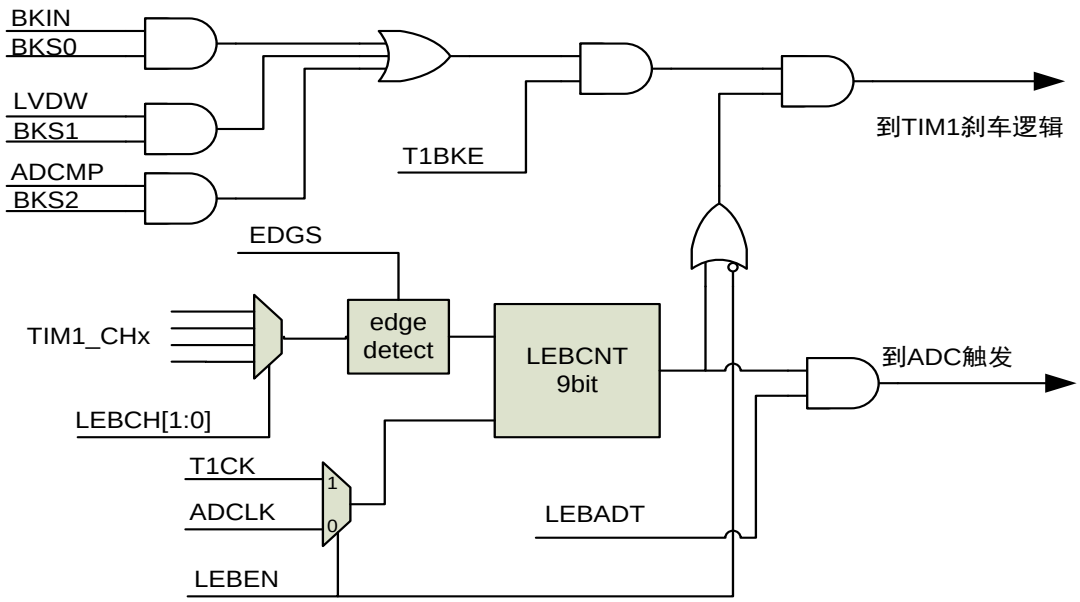


图 11-4 LEB 结构框图

在高速开关应用中，开关(如 MOSFETs/IGBTs)导通瞬间通常会产生极大的瞬变电流，而这些瞬变会导致测量误差。利用前沿消隐(LEB)功能，应用程序可忽略 PWM 输出边沿附近产生的由 MOSFETs/IGBTs 开关所引起的预期瞬变。

LEB 和 PWM 的时钟源均为 T1CK (Timer1 时钟源)。LEB 计时期间，ADC 保持采样，直至 LEB 计时溢

出(参阅 “LEBPR”，**表 11-3**)。在 LEB 计时周期内如果再次发生有效的 LEB 触发沿，则 LEB 定时器将清 0 并重新开始计数。

触发条件	延迟 / 消隐	触发通道
指令	(无延迟)	(N/A)
I/O (PA4/PB3)	(ADDLY+6) x T_{AD}; ADDLY = LEBPR	I/O (PA4或PB2)
PWM	(LEBPR+6) x T_{AD}	LEBEN = 0; ETGSEL (LEBCH 忽略)
	(LEBPR+3) x T_{T1CK} + 3 x T_{AD} (T_{T1CK} = Timer1 period)	LEBEN = 1; LEBCH (ETGSEL 忽略)

表 11-4 ADC 触发，延迟和通道设置

如果由软件触发(ADEX = 0)，GO/DONE 由指令置位后立即启动 A/D 转换。如果由 PA4/PB2 或 PWM 触发，则有一定的延迟时间(“6 x TAD” 或 “3 x T_{T1CK} + 3 x T_{AD}”，参阅 **表 11-4**)。另外可通过设置 ADDLY / LEBPR 寄存器在 GO/DONE 置位前增加额外的延迟。ADC 延时定时器(ADDLY)和 LEB 定时器(LEBPR)共用同一个 9-bit 计数器，此计数器由 LEBPR9 和 LEBPRL[7:0]组成。延迟结束后采样保持电路将在 “3.5 x T_{AD} – 4.5 x T_{AD}” 时间内断开。

注：

1. 在使能 LEB 前，需先设置 ADEX 和 ADON 寄存器。
2. ADC 转换完成前将忽略新的触发条件。
3. 如果 LEBEN=1，则将忽略 ETGSEL，触发源即为 LEB 的触发源。此时由 LEB 定时器溢出触发 ADC 自动转换 (参阅 “LEBADT”，**表 11-3**)。

11.2.2 ADC 中止转换

有时需中止 ADC 转换，比如需启动新的 ADC 采样。

- 当 ADEX = 0 (指令触发)时，可通过软件设置 GO/DONE = 0 来中止 ADC。
- 当 ADEX = 1 时，必须通过关闭 ADC 模块(ADON = 0)来中止 ADC。
- 当 ADC 转换被中止时，中止操作需等待 4 x T_{AD} 的处理时间，而后 ADRESH 和 ADRESL 将部分更新为已转换完成位的值，未完成位将全部用最后转换的那 1 位的值来填充。
- 系统复位时，由于相应的寄存器被复位，因此 ADC 将中止，且 ADC 模块被关闭。

11.2.3 阈值比较

ADC 转换完成后可自动将结果与 ADCMPH 寄存器里的阈值进行比较(参阅 ”ADCMPEN”)。由 ADCMPOP 设置比较极性，ADCMPO 输出比较结果。当产生相应的匹配条件时可触发 PWM 故障刹车(参阅 ”ADFBEN”)。仅转换结果的高 8 位用于阈值比较，因此 V_{REF}+和 V_{REF}−之间的比较 step 为 0.4%。

注：

1. 当 ADCMPEN = 0 或 ADON = 0 时，ADCMPO 将清零；进入睡眠模式，ADCMPO 不清零。

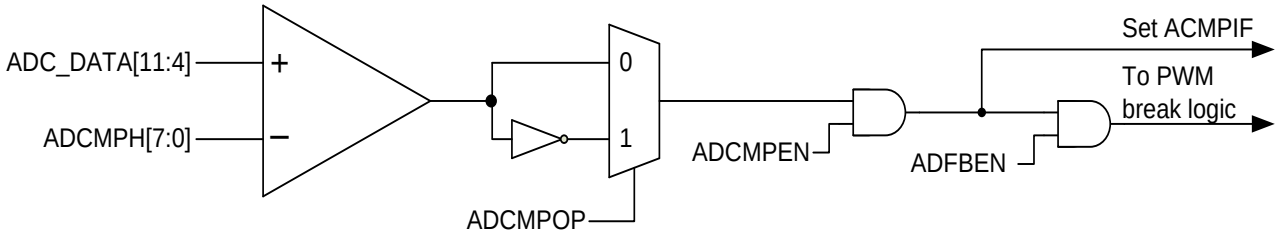


图 11-5 ADC 阈值比较结构框图

11.2.4 中断

ADC 转换完成时将置位相应的中断标志位 ADCIF。是否触发中断和/或从睡眠中唤醒则取决于相应的中断使能控制位(GIE, PEIE, ADCIE)。

注：

1. 无论中断使能位是否打开，ADC 每次正常转换完成时都将置位 ADCIF。
2. 自动校准完成、软件中止 AD 转换均不会置位 ADCIF。

11.3 ADC 采样保持时间

采样保持时间 T_{ACQ}，必须足够长以保证内部 ADC 电压稳定在输入通道电压的 0.01%误差以内，从而达到 12bit 的精度(0.024%)。采样保持时间和外部串联电阻的关系如下(**表 11-5**)：

$$T_{ACQ} > 0.09 \times (R + 1)\ (\mu s);\ R \text{ 的单位为 } k\Omega$$

当采样保持时间 T_{ACQ} 为 2μs 时，外部串联电阻必须≤ 21 kΩ。如果使用更大的串联电阻，则 T_{ACQ} 将成比例增加。结点漏电流限制了允许使用的最大串联电阻值。对于 5nA 的结点漏电流，在 50 kΩ 的串联电阻上将产生 0.25mV (2V 参考电压的 0.0125%)的压降。而当温度超过 100°C 时，结点漏电流将大幅提高。因此，串联电阻越小越好。

串联电阻值	T_{ACQ}
> 50 kΩ	(不推荐)
43 kΩ	≥ 4.0 μs
21 kΩ	≥ 2.0 μs
< 21 kΩ	≥ 2.0 μs

表 11-5 不同的外部串联电阻与最短 T_{ACQ} 的对应关系

采样保持时间即为内部 ADC 观测输入通道电压的时间。

采样保持时间的开始 = 通道切换(参阅"CHS")后或 ADC 稳定(参阅"T_{ST}")后，以时间较长者为准。

采样保持时间的结束 = 硬件触发延迟结束或 GO/DONE 置 1 后的 0 – 1 x T_{AD} 时间内，延迟时间由触发条件决定(参阅 **表 11-4**)，同时采样保持电路断开。

采样点 = 采样保持电路断开前的瞬间，有 0 – 1 x T_{AD} 时间的不确定性。

采样断开后开始数据转换，转换过程需 16 x T_{AD} 时间。因此从硬件触发延迟结束或软件 GO/DONE 置 1 后到数据转换完成需要 16 x T_{AD} 到 17 x T_{AD} 时间。数据转换完成后，采样保持电路重新闭合，开始下一个采样周期，同样需等待足够长的采样时间 T_{ACQ} 后，才能再次启动 A/D 转换。

11.4 ADC 最短采样时间

T_{AD} 为 ADC 的时钟周期。完整的 12-bit 转换所需最短时间：

T_{ACQ} + 17 x T_{AD}

可保证 11-bit 精度的最高转换采样率为 100 kHz (~10 μs/采样)。

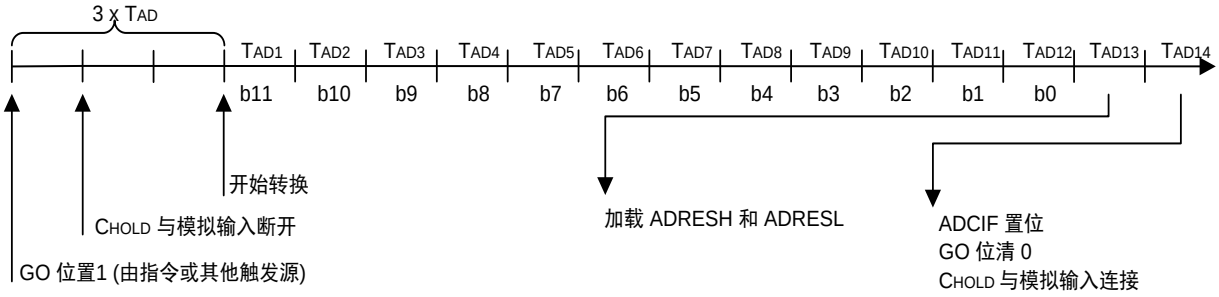


图 11-6 模数转换 T_{AD} 周期

11.5 ADC 转换步骤示例

设置 ADC：

1. 设置 ADCEN = 1，打开 ADC 模块时钟；
2. 配置端口：
   a. 设置 TRISx = 1，禁止引脚输出驱动；
   b. 设置 ANSELAx = 1，关闭数字输入、弱上拉和弱下拉功能；
3. 配置 ADC 模块：
   a. 选择 ADC 转换时钟源；
   b. 选择 ADC 参考电压；
   c. 选择 ADC 触发条件：软件、PA4/PB3-ADC_ETR 或 PWM，有或无 LEB；
   d. 选择转换结果格式；
   e. 使能阈值比较(可选)；
4. 配置 ADC 中断（可选）：
   a. 使能 ADC 转换完成中断；
   b. 使能外设总中断；
   c. 关闭全局中断(如需执行中断服务程序则使能)；
5. 打开 ADC 模块。然后等待所需稳定时间 T_{ST} (~15 μs)，当 $V_{ADC\text{-}REF}$ 选择内部参考电压时，则需等待内部参考电压的稳定时间 T_{VRINT} (参阅" T_{VRINT}"，**章节 16.7**) 和 T_{ST} 时间的较长者，即 max(T_{VRINT},T_{ST})。

至此，ADC 已准备好对不同的通道进行采样。对输入通道采样时：

1. ADC 输入选择为需测量的通道（参阅"CHS"）。
2. 如有必要，需清除 ADC 转换完成中断标志位 ADCIF。
3. 对采样时间 T_{ACQ} 有最低要求，T_{ACQ} 需足够长以保证内部 ADC 输入电容充满至输入通道电压的 0.01%

误差以内。另外取决于触发类型，切换通道后或 ADC 稳定后(以时间较长者为准)可能会有一定的延迟再触发。

a. 对于软件触发，需要额外的 T_{ACQ} 时间。

b. 对于 PA4/PB3-ADC_ETR 或 PWM 触发，除非使用非常大的串联电阻，否则内部延迟时间 (ADDLY+6) x T_{AD} 通常大于 T_{ACQ}，因此不需要再额外延迟 T_{ACQ}。

6. 等待所需的延迟后，由指令置位 GO/DONE，或等待硬件触发事件自动置位 GO/DONE，以启动 A/D 转换。

7. 通过以下方式等待 ADC 转换完成：

   a. 等待 1 个 sysclk 周期后，查询 GO/DONE 位；

   b. 等待 ADC 中断 (使能中断时)；

8. 读取 ADC 转换结果。

9. 如有必要，清除 ADC 转换完成中断标志位 ADCIF。

注：

1. 虽然 GO/DONE 和 ADON 在同一个寄存器(ADCON0)中，但不应同时设置。

2. ADC 转换过程中或等待外部触发时，不可更改配置。建议在 ADON = 0 时进行更改。

以下为 ADC 程序示例 (输入采样通道为 PB1，ADC 时钟为 LIRC)：

```
BANKSEL PCKEN
BSR PCKEN,0                 ; ADC module clock
BANKSEL TRISB
BSR TRISB,1                 ; Set PB1 to input
BANKSEL ANSELA
BSR ANSELA, 0               ; Set PB1 to analog
BANKSEL ADCON1
LDWI B'11110101'            ; Right justify, ADC LIRC clock
STR ADCON1                  ; Vref+: VDD , Vref-: GND
BANKSEL ADCON0
LDWI B'00000000'            ; Select channel AN0
STR ADCON0
BSR ADCON0,ADON             ; Turn ADC On
CALL StableTime             ; ADC stable time
BSR ADCON0,GO               ; Start conversion
NOP                         ; GO/DONE ReadBack WaitTime
BTSC ADCON0,GO              ; Conversion done?
LJUMP $-1                   ; No, test again
BANKSEL ADRESH
LDR ADRESH,W                ; Read upper 4 bits
STR RESULTHI                ; store in SRAM space
BANKSEL ADRESL
LDR ADRESL,W                ; Read lower 8 bits
STR RESULTLO                ; Store in SRAM space
```

12 USART 接口

通用同步/异步收发器 USART 可与使用工业标准 NRZ 串行数据格式的外设进行通信，特性如下：

- 全双工、单线半双工异步模式
- 全双工同步模式
  - ✓ 同步时钟输出：可编程时钟极性和相位
- 红外 1.0 模式
  - ✓ 8 位预分频波特率发生器
  - ✓ 低功耗模式
- 智能卡模式
  - ✓ 8 位预分频波特率发生器
  - ✓ 停止位：1.5 位
  - ✓ 可编程保护时间
- LIN 主机模式
  - ✓ 支持断开帧的发送与检测
- 多芯片通信模式
  - ✓ 可通过地址匹配或 IDLE 帧唤醒哑模式，唤醒后开始接收数据
- 数据传输长度：7、8 或 9 位
- 奇偶校验位
- 停止位：1 或 2 位
- 16 位可编程波特率发生器，高达 1Mbit/s
- 数据从低位开始发送/接收
- 发送器和接收器可单独使能
- 自动波特率检测
- 支持 USART 接口 TX 开漏输出
- 发送 BUF 为空中断、接收 BUF 为非空中断
- 发送完成中断
- 空闲帧中断
- 接收状态中断：帧断开、帧错误、奇偶校验错误或接收溢出

注：默认为异步全双工模式，当选定一种工作模式后，请关闭其他模式的使能位。

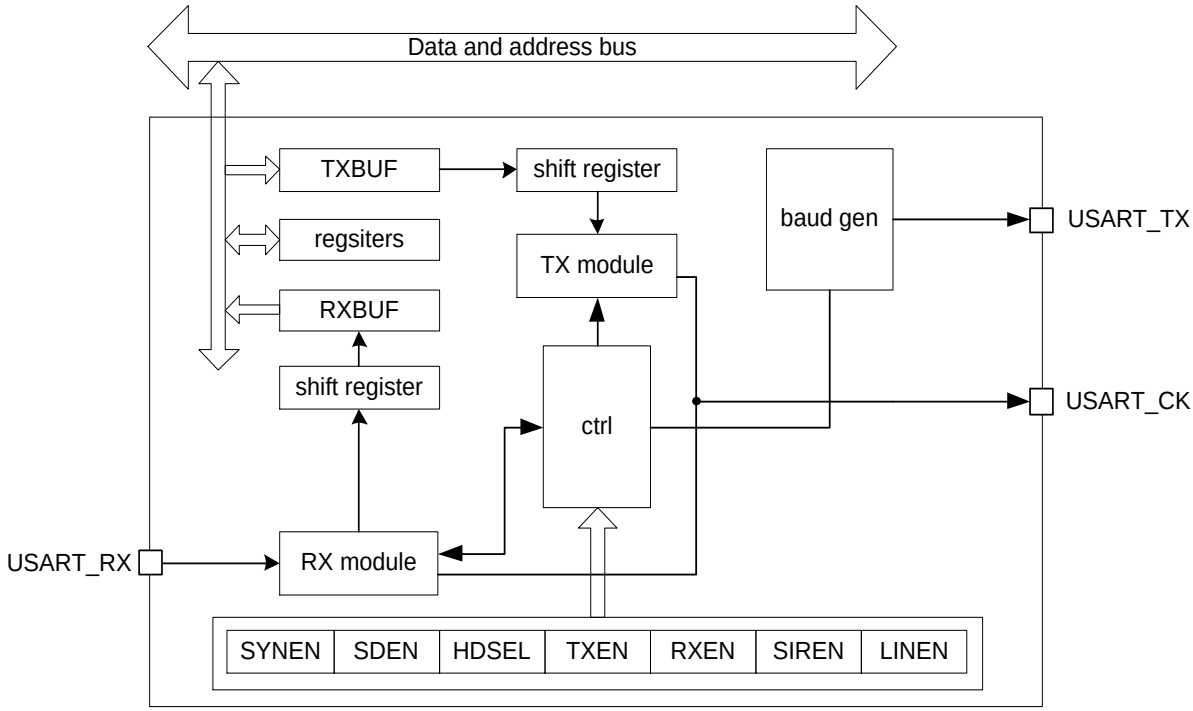


图 12-1 USART 结构框图

USART 串口模块有 3 个引脚：

USART_RX：串行数据输入。

USART_TX：串行数据输出。单线半双工模式下，TX 引脚既用于数据输入也用于数据输出 (需配置成开漏模式)。

USART_CK：同步模式下用作同步时钟输出，智能卡模式下用作系统时钟分频输出。

注：

- 当发送器使能，但不发送数据时，TX 引脚处于高电平。
- 当发送器使能，且发送数据时，TX 引脚在起始位期间处于低电平，在停止位期间处于高电平。

12.1 USART 接口相关寄存器汇总

名称	状态		寄存器	地址	复位值
UARTEN	USART 模块时钟	1 = 打开 0 = 关闭	PCKEN[6]	0x9A	RW-0
SYSON	睡眠模式下，系统时钟控制	1 = 保持运行 0 = 关闭	CKOCON[7]	0x95	RW-0
DATAL	数据发送 / 接收 BUF 低 8 位 (不适宜位操作)		URDATAL[7:0]	0x48C	RW-0000 0000
DATAH	数据发送 / 接收 BUF 高 1 位 (当 EXTEN=1 时) 注：需先写 DATAL，再 DATAH；		URDATAH[0]	0x48D	RW-0
BKREQ	发送断开帧 1 = 使能，或正在发送中 0 = 关闭，或已发送完成 注： 1. 发送断开帧之前请先设置断开帧的长度； 2. 此位在发送完成后自动清零，禁止在发送过程中对其写 0；		URLCR[6]	0x48F	RW-0
EVEN	奇/偶校验	1 = 偶校验 0 = 奇校验	URLCR[4]		RW-0
PEN	校验位	1 = 使能 0 = 关闭	URLCR[3]		RW-0
URSTOP	停止位长度 1 = 1.5 bit (智能卡模式) 或 2 bit 0 = 1 bit		URLCR[2]		RW-0
LTH	通信数据长度控制位(不包括校验位) 1 = 8 bit 0 = 7 bit		URLCR[0]		RW-0
RWU	多处理器模式下，进入哑模式 1 = 使能 0 = 关闭，或已退出		URLCREXT[1]	0x490	RW-0
EXTEN	通信数据长度总控制位 (不包括校验位) 1 = 9 bit 0 = 7 bit 或 8 bit (由 LTH 决定)		URLCREXT[0]		RW-0
SIRLP	红外低功耗模式	1 = 使能 0 = 关闭	URMCR[5]	0x491	RW-0

名称	状态		寄存器	地址	复位值
TXEN	串口发送 1 = 使能 (引脚 TX 功能自动使能) 0 = 关闭		URMCR[4]		RW-0
RXEN	串口接收 1 = 使能 (引脚 RX 功能自动使能) 0 = 关闭		URMCR[3]		RW-0
WAKE	哑模式唤醒方式	1 = 地址匹配 0 = IDLE 帧	URMCR[2]		RW-0
HDSEL	半双工	1 = 使能 0 = 关闭	URMCR[1]		RW-0
SIREN	红外模式	1 = 使能 0 = 关闭	URMCR[0]		RW-0
RAR	多处理器模式下的本机地址[3:0]		URRAR[3:0]	0x493	RW-0000
DLL	波特率分频计数器低 8 位和高 8 位 波特率 = Fmaster / (16 * {DLH, DLL}) 注：Fmaster = Sysclk；{DLH, DLL} 最小值为 0x0001，当其为 0x0000 时, USART 不工作；		URDLL[7:0]	0x494	RW-0000 0000
DLH			URDLH[7:0]	0x495	RW-0000 0000
ABRE	波特率检测溢出标志	1 = 溢出 0 = 正常	URABCR[3]	0x496	RW-0
ABRM	波特率检测模式 1 = 检测长度为 [(起始位+第 1bit 数据) / 2] (数据的第 1bit 必须为 1，第 2bit 必须为 0) 0 = 只检测起始位长度 (第 1bit 数据必须为 1)		URABCR[2]		RW-0
ABRF	检测到波特率标志位 1 = 检测到 0 = 未检测到 注：写 0 清零，该位清零后，会立即再次进入波特率检测，为了保证每次检测到的都是起始位，建议在 RXNEF 被置位后，再清零此位；		URABCR[1]		RW-0
ABREN	自动波特率检测	1 = 使能 0 = 关闭	URABCR[0]		RW-0
LBCL	同步模式下，发送最后 1bit 数据(MSB)对应的时钟输出 1 = 使能 0 = 关闭		URSYNCR[3]	0x497	RW-0

名称	状态	寄存器	地址	复位值
URCPHA	同步模式时钟相位（数据采样点） 1 = 第 2 个时钟转换沿 0 = 第 1 个时钟转换沿	URSYNCR[2]		RW-0
URCPOL	同步模式时钟极性（总线空闲时，SCK 的状态） 1 = 高电平 0 = 低电平	URSYNCR[1]		RW-0
SYNEN	同步模式 1 = 使能（引脚 CK 自动输出同步时钟） 0 = 关闭	URSYNCR[0]		RW-0
LINEN	LIN Master 模式 1 = 使能 0 = 关闭	URLINCR[4]	0x498	RW-0
BLTH	断开帧长度（bit） 注: BLTH>0 有效，建议设置为 12bit 或 13bit，太短会将接收到的误判为正常帧；	URLINCR[3:0]		RW-0000
NACK	智能卡模式，检测到奇偶校验出错时回复 NACK 1 = 发送 NACK 0 = 不发送 NACK	URSDCR0[6]	0x499	RW-0
CKOE	智能卡时钟源 1 = 使能（需配置 PSC 寄存器为有效值） 0 = 关闭	URSDCR0[5]		RW-0
SDEN	智能卡模式 1 = 使能（停止位必须为 1.5bit） 0 = 关闭	URSDCR0[4]		RW-0
GT	智能卡模式，保护时间（两字符之间的波特时钟间隔） 注意：最小值为 1 (0 无效)，保护时间过后，发送完成标志才被置位	URSDCR1[7:0]	0x49A	RW-0000 0000
PSC	对系统时钟进行分频，给智能卡或红外低功耗提供时钟	URSDCR2[7:0]	0x49B	RW-0000 0000

	智能卡时钟源	红外低功耗时钟源
0	无效	无效
1	2 分频	1 分频
2	3 分频	2 分频
3	4 分频	3 分频
…	…	…
255	256 分频	255 分频

表 12-1 USART 相关用户寄存器

名称	状态		寄存器	地址	复位值
AFP1[5:4]	USART_TX	00 = PA6 10 = PA7 01 = PB6 11 = PA2	AFP1[5:4]	0x19F	RW-00
AFP1[3:2]	USART_RX	00 = PA7 10 = PA6 01 = PA2 11 = PB6	AFP1[3:2]		RW-00
UROD	USART_TX 开漏输出	1 = 使能 0 = 关闭	ODCON0[0]	0x21F	RW-0

表 12-2 USART 接口引脚控制

名称	状态		寄存器	地址	复位值
GIE	全局中断	1 = 使能 (PEIE, URTE, URRXNE, TCEN, IDELE, RXSE 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 (URTE, URRXNE, TCEN, IDELE, RXSE 适用) 0 = 关闭 (无唤醒)	INTCON[6]		RW-0
URTE	发送 BUF 为空中断	1 = 使能 0 = 关闭	URIER[1]	0x48E	RW-0
TXEF	发送 BUF 状态	1 = 空 0 = 非空 注：写 DATAL(8bit) / DATAH(9bit) 清零；	URLSR[5]	0x492	RO-1
URRXNE	接收 BUF 为非空中断	1 = 使能 0 = 关闭	URIER[0]	0x48E	RW-0
RXNEF	接收 BUF 状态	1 = 非空 0 = 空，或已被清零 注：读 DATAL(8bit) / DATAH(9bit) 清零；	URLSR[0]	0x492	RO-0
TCEN	发送完成中断	1 = 使能 0 = 关闭	URIER[5]	0x48E	RW-0
TCF	发送完成标志	1 = 完成 0 = 未完成 注：写 1 清零，或写 DATAL(8bit) /DATAH(9bit)后清零；	URTC[0]	0x49C	R_W1C-1

名称	状态		寄存器	地址	复位值
IDELE	空闲帧中断	1 = 使能 0 = 关闭	URIER[3]	0x48E	RW-0
IDLEF [1]	检测到空闲帧标志	1 = 检测到 0 = 未检测到	URLSR[6]	0x492	RW0-0
RXSE	接收错误中断	1 = 使能 0 = 关闭 注：接收错误中断产生条件如下 BKF = 1 FEF = 1 PEF = 1 OVERF = 1	URIER[2]	0x48E	RW-0
BKF [1]	接收到断开帧标志	1 = 接收到 0 = 未接收到，或已被清零	URLSR[4]	0x492	RW0-0
FEF [1]	接收到帧错误标志	1 = 错误 0 = 正确，或已被清零	URLSR[3]	0x492	RW0-0
PEF [1]	接收到奇偶校验错误标志	1 = 错误 0 = 正确，或已被清零	URLSR[2]	0x492	RW0-0
OVERF [1]	接收 BUF 溢出错误标志	1 = 溢出 0 = 正常，或已被清零	URLSR[1]	0x492	RW0-0
WAKE	哑模式唤醒方式选择	1 = 地址匹配 0 = IDLE 帧	URMCR[2]	0x491	RW-0
ADDRF	哑模式地址匹配标志	1 = 匹配 0 = 未匹配	URLSR[7]	0x492	RO-0

表 12-3 USART 中断使能和状态位

[1] 写 0 清零，写 1 无效。

名称	地址	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	复位值
PCKEN	0x9A	—	—	UARTEN	—	TIM4EN	TIM2EN	TIM1EN	ADCEN	--00- 0000
CKOCON	0x95	SYSON	CCORDY	DTYSEL		CCOSEL[2:0]			CCOEN	0010 0000
URDATAL	0x48C	DATAL[7:0]								0000 0000
URDATAH	0x48D	—							DATAH	---- ---0
URIER	0x48E	—		TCEN	—	IDELE	RXSE	URTE	URRXNE	--0- 0000
URLCR	0x48F	—	BKREQ	—	EVEN	PEN	STOP	—	LTH	-0-0 00-0
URLCREXT	0x490	—						RWU	EXTEN	---- --00
URMCR	0x491	—		SIRLP	TXEN	RXEN	WAKE	HDSEL	SIREN	--00 0000
URLSR	0x492	ADDRF	IDLEF	TXEF	BKF	FEF	PEF	OVERF	RXNEF	0000 0000
URRAR	0x493	—				RAR[3:0]				---- 0000
URDLL	0x494	DLL[7:0]								0000 0000
URDLH	0x495	DLH[7:0]								0000 0000
URABCR	0x496	—				ABRE	ABRM	ABRF	ABREN	---- 0000
URSYNCR	0x497	—				LBCL	CPHA	CPOL	SYNEN	---- 0000
URLINCR	0x498	—			LINEN	BLTH[3:0]				---0 0000
URSDCR0	0x499	—	NACK	CKOE	SDEN	—				-000 ----
URSDCR1	0x49A	GT[7:0]								0000 0000
URSDCR2	0x49B	PSC[7:0]								0000 0000
URTC	0x49C	—							TCF	---- ---1

表 12-4 USART 相关寄存器地址

12.2 USART 功能

12.2.1 异步工作模式

全双工和半双工的配置流程：

1. 设置 UARTEN = 1，使能 USART 模块时钟；
2. 设置通信波特率 = Fmaster / (16 * {DLH, DLL}) (参阅 ”DLH”, “DLL”)；
3. 设置通信数据长度为 7, 8 或 9 位 (参阅 “EXTEN”, “LTH”)；
4. 设置奇偶校验位 (参阅 “PEN”, “EVEN”)；
5. 设置停止位长度为 1 或 2 位 (参阅 “URSTOP”)；
6. 选择全双工 (默认) 或半双工工作模式 (参阅 “HDSEL”)；
7. 如需要，可使能相应的中断 (参阅” GIE”, ” PEIE”, ”URTE” , ”URRXNE”, “TCEN” 和 ”RXSE” 等)；
8. 根据需要，设置 TXEN = 1 或 RXEN = 1，使能发送或接收功能；

注：

- USART 外设时钟 Fmaster = Sysclk；
- 半双工模式下，如果同时使能发送和接收功能，则发送的数据也会被本机接收到；

异步模式的数据通信格式为先发送低位，后发送高位。有无奇偶校验位的数据帧格式对比如下图：

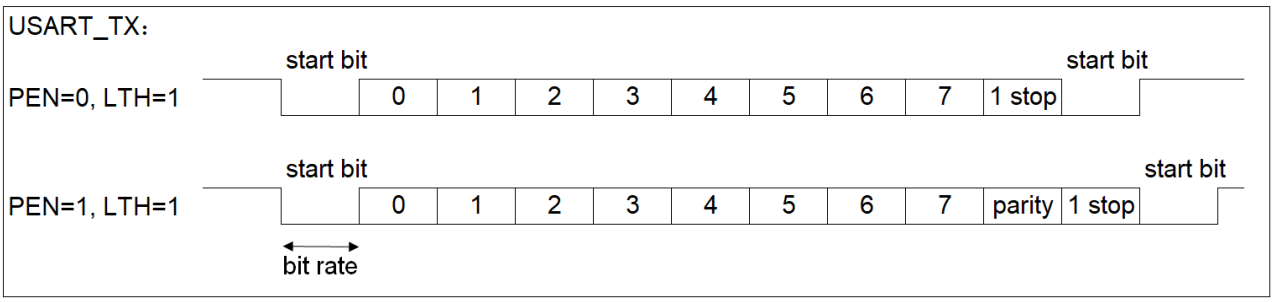


图 12-2 异步模式通信格式 (以 8bit 长度为例)

数据处理流程包括阻塞模式和非阻塞模式：

	阻塞模式	非阻塞模式
发送数据	向 DATAL/H(TXBUF)写入数据后，查询 TXEF 或 TCF，当其置 1 时，写入下一个数据	当 URTE = 1 或 TCEN = 1 时，向 DATAL/H(TXBUF)写入数据后，TXEF 或 TCF 置 1 则进入中断
接收数据	查询 RXNEF，当其置 1 时，则可读取 DATAL/H(RXBUF)的值	当 URRXNE = 1 时，RXNEF 置 1 后进入中断； 此外，建议使能 RXSE 中断，当接收错误时则进入相应的中断进行处理；
备注	-	进入中断后，查询相应的状态标志位并处理发送接收流程，处理完成后退出中断

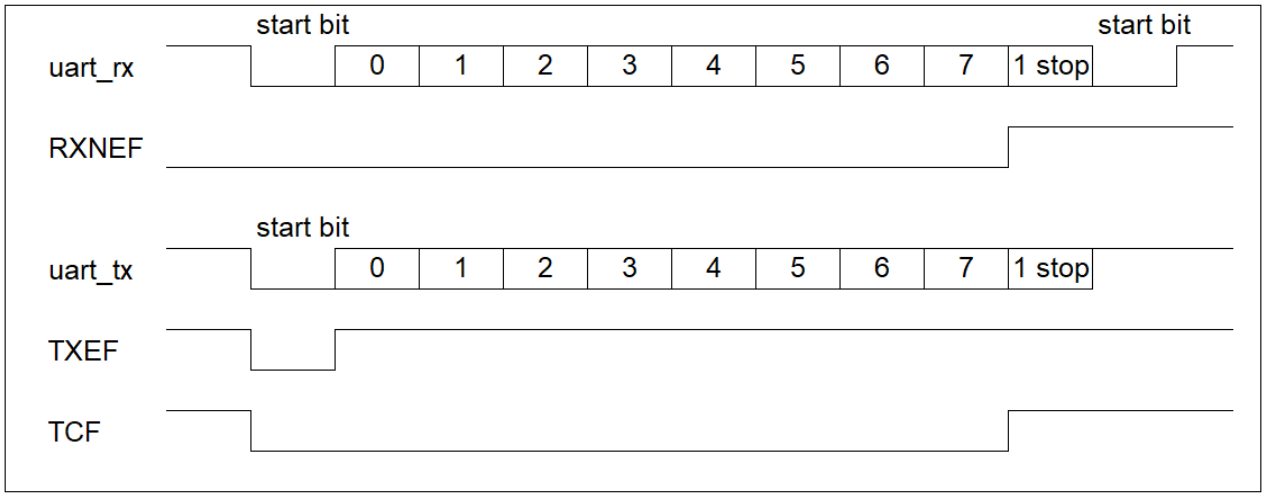


图 12-3 异步模式标志位时序图

12.2.2 同步工作模式

同步模式用于模拟 SPI 主机模式下的通信功能。当 SYNEN = 1 时，USART_CK 引脚将输出与数据同步的时钟。数据输出先发送低位，后发送高位。

此外，可选择同步时钟的极性和相位（参阅 “CPOL”, “CPHA”)。在起始位和停止位期间，USART_CK 引脚上没有时钟脉冲。发送最后 1bit 数据时是否输出同步时钟由 LBCL 决定。

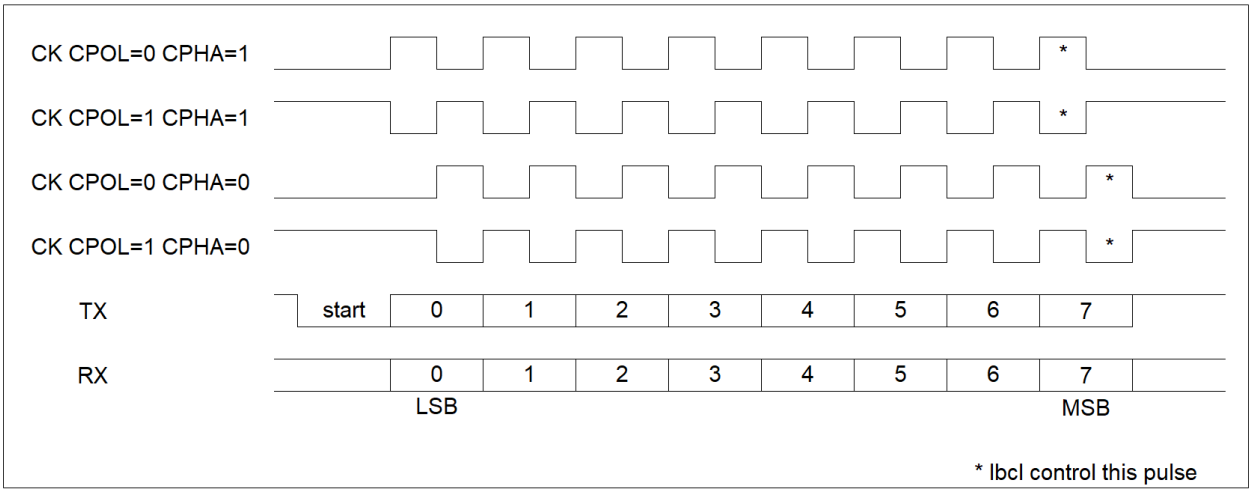


图 12-4 同步模式通信格式（8bit 数据长度）

注：

- 同步时钟速率与波特率设置相同，即 Fmaster / (16 * {DLH, DLL})；
- 当 TXEN = 0，且 RXEN = 1 时，仍会输出同步时钟，此时只用于接收数据，TX 引脚保持为高电平；

12.2.3 红外工作模式

红外模式用于红外通信。当 SIREN = 1 使能红外模式时，通信数据长度默认为 8 位。

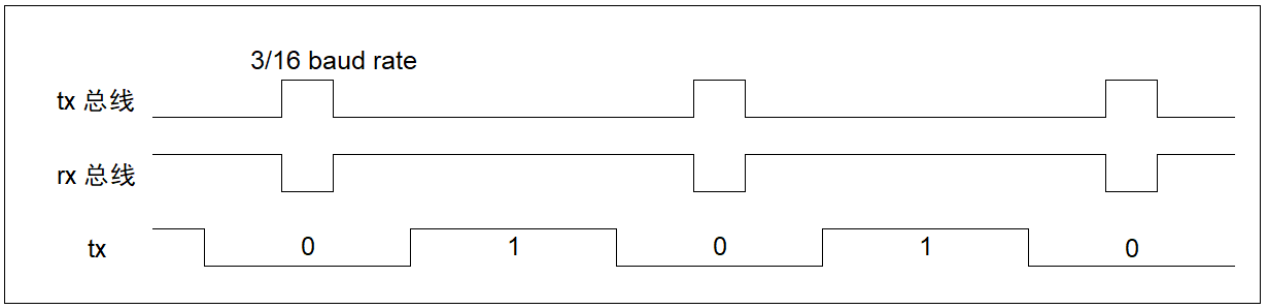


图 12-5 红外模式通信时序图

如图 12-5 所示，红外模块发送或接收总线上的数据脉冲宽度为正常模式时比特周期的 3/16。当发送数据为零时会产生一个高脉冲，而接收时的低脉冲会被译码成零。发送与接收总线极性相反，发送空闲时总线保持低电平，接收空闲时总线保持高电平。

正常模式下的红外模块，通信波特率 = Fmaster / (16 * [DLH:DLL])；

低功耗模式（参阅” SIRLP”)下，通信波特率 = Fmaster / (PSC * 16 *[DLH:DLL])；

12.2.4 智能卡模式

智能卡模式属于半双工模式，支持 ISO7816-3 标准。当 SDEC = 1 使能智能卡模式时，根据协议要求需设置数据长度为 8 位(参阅” LTH”)，使能奇偶校验位(参阅” PEN”)，且设置停止位为 1.5 位(参阅” URSTOP”) ，同时需配置相应的 IO 为开漏模式。

智能卡的时钟源及分频输出由 CKOE 和 PSC 设置。

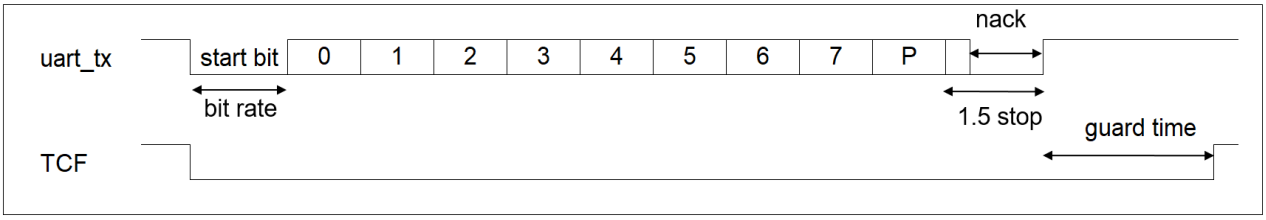


图 12-6 智能卡模式通信时序图

当 NACK = 1 时，接收方在检测到奇偶校验出错之后，会在 0.5 个停止位后拉低总线 1 比特周期，同时发送方会在停止位处检测总线是否被拉低，若检测到总线被拉低，则帧错误标志 FEF 将置 1。发送方根据要求可以选择重发当前的数据，发送次数由用户决定。

当 NACK = 0 时，接收方在检测到奇偶校验出错之后，不会拉低总线，此时奇偶检验错误标志位 PEF 置 1。

此外，智能卡模式可设置保护时间(参阅”GT”)，当发送方发送数据完成后，等待 GT 个波特时钟周期后 TCF 置位。

12.2.5 LIN Master 模式

LINEN 置位后进入 LIN Master 模式。

发送端需先配置断开帧的长度(参阅” BLTH”)。当设置 BKREQ = 1 使能断开帧发送时，TX 引脚会连续发送 BLTH 个低电平，发送完成后 BKREQ 自动清零。

接收端在接收到大于(起始位+数据长度+停止位)个数的连续低电平以后，会认为接收到了断开帧，断开帧标志位 BKF 将置 1。

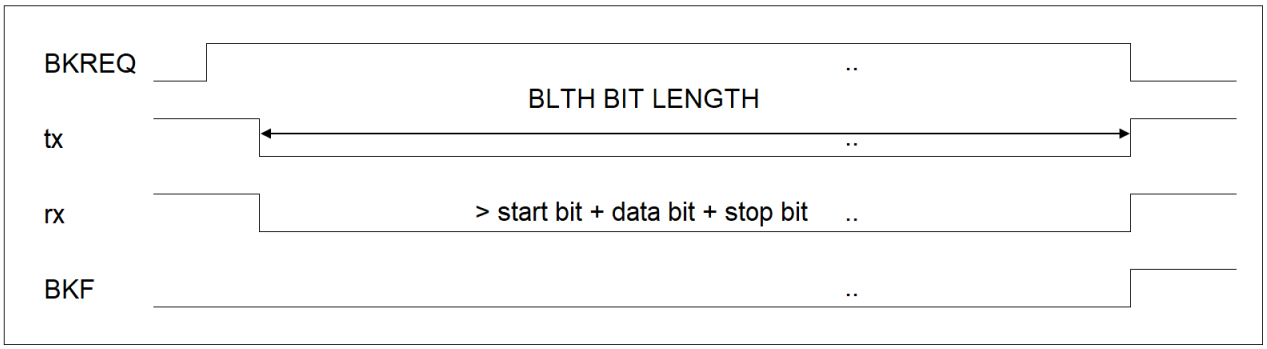


图 12-7 LIN Master 模式

注：断开帧的接收与发送不仅适用于 LIN Master 模式，其他异步模式、红外模式等也同样适用。

12.2.6 多处理器通信模式

多处理器通信模式，例如某个 USART 用作主机模式，其他 USART 用作从机模式，从机的 TX 输出通过逻辑与的方式连接到主机的 RX 输入。

当 RWU 置 1 后，主机进入哑模式，屏蔽一切接收。根据 WAKE 的设置，USART 唤醒主机接收数据或退出哑模式有以下两种方式：

- WAKE = 0：地址空闲唤醒。当检测到空闲帧后唤醒，并开始接收数据。若总线数据一直繁忙，则不唤醒。

  注：空闲帧，即完全由'1'组成的一个完整的数据帧（连续'1'的位数 ≥ (起始位+数据位+停止位) 的位数)。

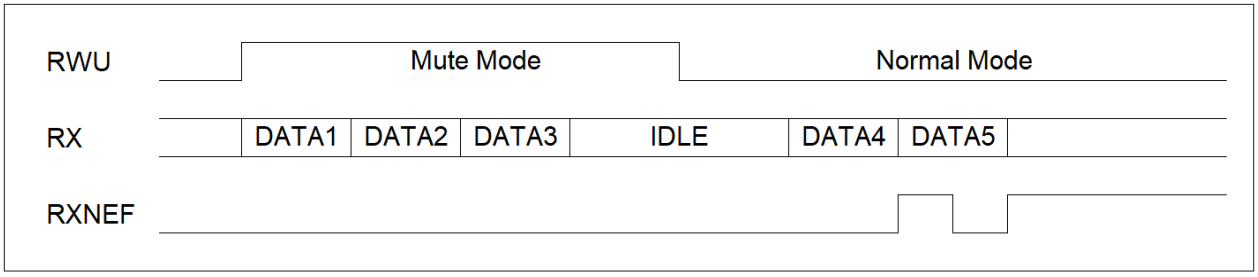


图 12-8 哑模式地址空闲唤醒

- WAKE = 1：地址匹配唤醒。每次接收到数据后会判断高位是否为 1(表示接收到的数据为地址数据)，若高位为 1 则继续对比数据的低四位和 URRAR 的值，相等则地址匹配标志位 ADDRF 置 1，退出哑模式，并开始接收之后的数据，若不相等则立即进入哑模式。

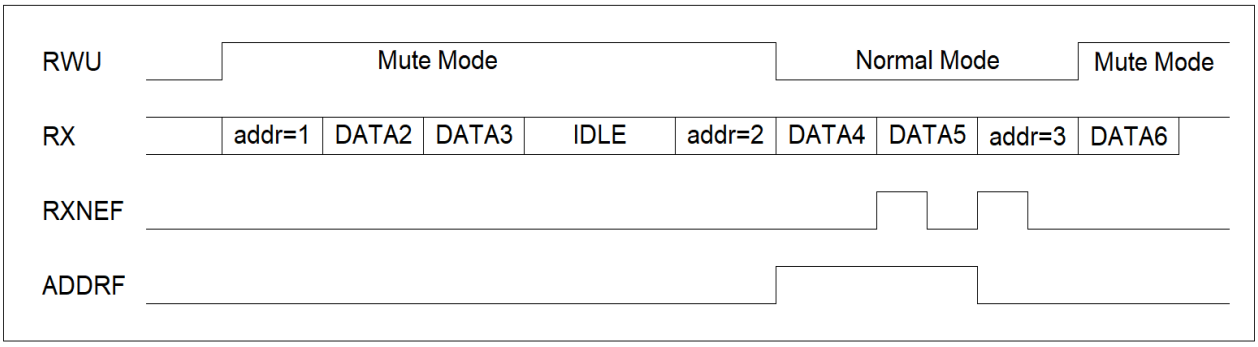


图 12-9 哑模式地址匹配唤醒

12.2.7 自动波特率检测

自动波特率检测功能用于接收端校准通信波特率，从而保持与发送端波特率相同。波特率检测模块有两种模式：

1. ABRM = 0：只检测起始位的长度，并要求数据的第 1bit = 1。例如数据 0x03、0x55 等。
2. ABRM = 1：检测起始位和第 1bit 的长度，并要求第 1bit = 1，第 2bit = 0。例如数据 0x55，0x01 等。

波特率检测数据用来自动配置 DLL/DLH。若发送端的波特率数据不靠近接收端的 Fbaudrate = Fmaster / (16 * {DLH, DLL})，则波特率检测模块会自动配置为较为靠近的波特率。串口模块并不支持小数波特率，因此该模块的波特率检测存在误差。

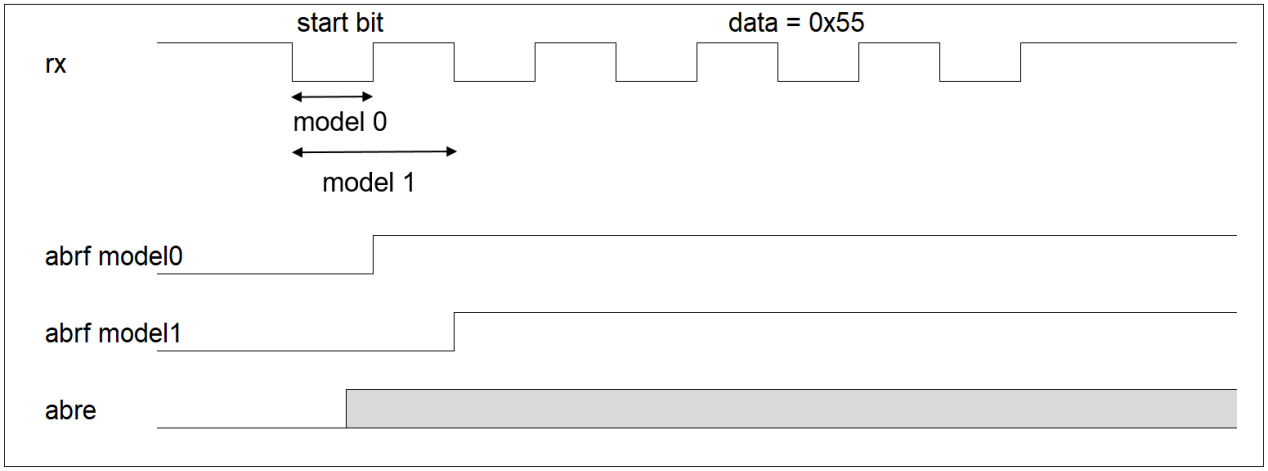


图 12-10 自动波特率检测

自动波特率检测流程：

1. 选择检测模式 ABRM；
2. 配置 ABREN = 1，使能自动波特率检测；
3. 读取检测到波特率标志位 ABRF 是否为 1 (上次未清零)，如果为 1，则写 0 清零；
4. 开始接收数据，波特率检测完成后 ABRF 置 1；
5. 当前数据接收完成后，接收 BUF 为非空标志位 RXNEF 置 1；
6. 开始下一次波特率的检测前，需先清零 ABRF；

注：

- 波特率检测完成，ABRF 置 1 后不可立即清零 ABRF。因为清零 ABRF 会立即在当前传输的位置（可能已经不是起始比特的位置）进行波特率检测，导致结果错误。
- 当波特率检测超出范围时，波特率检测溢出标志位 ABRE 将置 1。

13 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 1k x 14)。这些保护功能均由 IDE 界面进行选择配置。

名称	功能	默认
CPB	PROM 全区读保护	关闭
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

表 13-1 存储区读/写保护初始化配置寄存器

全加密和分扇区加密区别如下：

加密方式	CPU 取指	软件读	软件写	串口读	串口写
无	√	√	√(2)	√	√
全区	√	√	√(2)	×(1)	×(4)
分扇区	√	×(1)	×(3)	×(3)	×(5)

注：

1. EEDAT 保持旧值不变；
2. 软件不可以编程或擦除 UCFG 页；
3. 只可以读或写未加密的扇区；
4. 只允许串口做包括 UCFG 在内的全芯片擦除(解除加密)；
5. 只允许串口做包括 UCFG 在内的全芯片擦除(解除加密)，或者对未加密的扇区做页擦除，编程；
6. 任何情况下，软件都不可以做包括 UCFG 在内的全芯片擦除；

14 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；

14.1 初始化配置寄存器

图 14-1 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

名称	功能	默认
FOSC	• LP：PC1 (+) 和 PB7 (−) 接外部低速晶振 • XT：PC1 (+) 和 PB7 (−) 接外部高速晶振 • EC：PC1 (+) 接外部时钟输入，PB7 为 I/O • INTOSCIO：PC1 和 PB7 为 I/O	INTOSCIO
STVREN	堆栈上/下溢复位 • 使能 • 关闭	使能
OSTPER	OST 定时器周期选择 (XT / LP 适用) • 512 • 1024 • 2048 • 4096 (LP 模式时为 32768)	1024
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
FSCMEN	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
LVREN	LVR • 使能 • 关闭 • 非 SLEEP 模式下使能 • 通过指令控制 (SLVREN)	关闭
LVRS	7 档 V_{BOR} 电压(V)： 2.0 / 2.2 / 2.5 / 2.8 / 3.1 / 3.6 / 4.1	2.5
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

表 14-1 初始化配置寄存器 (由 IDE 设置)

14.2 用户寄存器

用户寄存器，即特殊功能寄存器(SFR)和 SRAM 分布在 12 个 bank (bank0~10, bank31) 中，每个 bank 大小为 128 字节。在访问寄存器前，必须先切换到相应的 bank。可通过将 bank 存储区号 (0~10, 31) 写入存储区选择寄存器 (Bank Select Register, BSREG) 来选择有效 bank。用户寄存器的地址为 12 位，地址范围为 0x000 ~ 0xFFF，高 5 位为 bank 区地址，低 7 位为 SFR/SRAM 地址。

所有用户寄存器都可以通过 INDFn 直接访问，或通过 FSRn 文件选择寄存器间接访问 (请参阅 **章节 14.5** 间接寻址)。

由于切换 bank 需要额外的指令，因此一些常用的 SFR 同时存储在 12 个 bank 中，以减少切换操作，这 12 个 bank 所共有的寄存器值是同步的。

Bank	首地址	Bank	首地址
Bank0	000H	Bank6	300H
Bank1	080H	Bank7	380H
Bank2	100H	Bank8	400H
Bank3	180H	Bank9	480H
Bank4	200H	Bank10	500H
Bank5	280H	Bank31	F80H

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	INDF0	使用 FSR0 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
首地址 + 1H	INDF1	使用 FSR1 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	–	–	–	/TO	/PD	Z	DC	C	0001 1xxx
首地址 + 4H	FSR0L	间接寻址指针寄存器 FSR0 低 8 位								xxxx xxxx
首地址 + 5H	FSR0H	间接寻址指针寄存器 FSR0 高 8 位								xxxx xxxx
首地址 + 6H	FSR1L	间接寻址指针寄存器 FSR1 低 8 位								xxxx xxxx
首地址 + 7H	FSR1H	间接寻址指针寄存器 FSR1 高 8 位								xxxx xxxx
首地址 + 8H	BSREG	存储区选择寄存器								xxxx xxxx
首地址 + 9H	WREG	工作寄存器 W								xxxx xxxx
首地址 + AH	PCLATH	–	程序计数器 (PC) 高 7 位锁存器							---0 0000
首地址 + BH	INTCON	GIE	PEIE	EEIE	LVDIE	OSFIE	EEIF	LVDIF	OSFIF	0000 0000
首地址 + (70 – 7F)		公共 BANK SRAM 区								xxxx xxxx

表 14-2 12 个 BANK 共有的寄存器

地址	名称	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
000	INDF0	使用 FSR0 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
001	INDF1	使用 FSR1 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
002	PCL	程序计数器 (PC) 低 8 位								0000 0000
003	STATUS	–	–	–	/TO	/PD	Z	DC	C	0001 1xxx
004	FSR0L	间接寻址指针寄存器 FSR0 低 8 位								xxxx xxxx
005	FSR0H	间接寻址指针寄存器 FSR0 高 8 位								xxxx xxxx
006	FSR1L	间接寻址指针寄存器 FSR1 低 8 位								xxxx xxxx
007	FSR1H	间接寻址指针寄存器 FSR1 高 8 位								xxxx xxxx
008	BSREG	存储区选择寄存器								xxxx xxxx
009	WREG	工作寄存器 W								xxxx xxxx
00A	PCLATH	–	程序计数器 (PC) 高 7 位锁存器							–000 0000
00B	INTCON	GIE	PEIE	EEIE	LVDIE	OSFIE	EEIF	LVDIF	OSFIF	0000 0000
00C	PORTA	PORTA [7:0]								xxxx xxxx
00D	PORTB	PORTB [7:0]								xxxx xxxx
00E	PORTC	–	–	–	–	–	–	PORTC[1:0]		xxxx xxxx
011	PIR1	–	–	–	–	–	–	CKMIF	ADCIF	–––– ––00
014	EPIF0	外部管脚中断标志位								0000 0000
01F	ADDLY /LEBPRL	ADC 外部触发启动延时计数器阈值低 8 位 / 复用为前沿消隐计数阈值								0000 0000
020–06F		SRAM BANK0 (80Bytes)								xxxx xxxx
070–07F		SRAM BANK0 (16Bytes), 物理地址 0x70–0x7F								xxxx xxxx

表 14-3 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
080	INDF0	使用 FSR0 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
081	INDF1	使用 FSR1 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
082	PCL	程序计数器 (PC) 低 8 位								0000 0000
083	STATUS	–	–	–	/TO	/PD	Z	DC	C	0001 1xxx
084	FSR0L	间接寻址指针寄存器 FSR0 低 8 位								xxxx xxxx
085	FSR0H	间接寻址指针寄存器 FSR0 高 8 位								xxxx xxxx
086	FSR1L	间接寻址指针寄存器 FSR1 低 8 位								xxxx xxxx
087	FSR1H	间接寻址指针寄存器 FSR1 高 8 位								xxxx xxxx
088	BSREG	存储区选择寄存器								xxxx xxxx
089	WREG	工作寄存器 W								xxxx xxxx
08A	PCLATH	–	程序计数器 (PC) 高 7 位锁存器							–000 0000
08B	INTCON	GIE	PEIE	EEIE	LVDIE	OSFIE	EEIF	LVDIF	OSFIF	0000 0000
08C	TRISA	TRISA[7:0]								1111 1111
08D	TRISB	TRISB[7:0]								1111 1111
08E	TRISC	–	–	–	–	–	–	TRISC[1:0]		---- --11
091	PIE1	–	–	–	–	–	–	CKMIE	ADCIE	---- --00
094	EPIE0	外部管脚中断使能位								0000 0000
095	CKOCON	SYSON	CCORDY	DTYSEL		CCOSEL[2:0]			CCOEN	0010 0000
096	PCON	STKOVF	STKUNF	EMCF	IERRR	/MCLRR	/SRSTF	/PORF	/BORF	qqqq qqqq
097	WDTCON	WDTPRE[2:0]			WDTPS[3:0]				SWDTEN	1110 1000
098	OSCTUNE	HIRC 时钟频率调节位								–xxx xxxx
099	OSCCON	MCKCF[3:0]				OSTS	HTS	LTS	SCS	0100 x000
09A	PCKEN	–	–	UARTEN	–	TIM4EN	TIM2EN	TIM1EN	ADCEN	0000 0000
09B	ADRESL	ADC 结果寄存器低 8 位								0000 0000
09C	ADRESH	ADC 结果寄存器高 8 位								0000 0000
09D	ADCON0	–	CHS[2:0]			–	ADEX	GO/DONE	ADON	0000 0000
09E	ADCON1	ADFM	ADCS[2:0]			ADNREF[1:0]		ADPREF[1:0]		0000 0000
09F	ADCON2	ADINTREF[1:0]		ETGTYP[1:0]		ADDLY.8	ETGSEL[2:0]			0000 0000
0A0–0EF		SRAM BANK1 (80Bytes)								xxxx xxxx
0F0–0FF		SRAM BANK1 (16Bytes), 访问 BANK0's 物理地址 0x70–0x7F								xxxx xxxx

表 14-4 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

地址	名称	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
100	INDF0	使用 FSR0 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
101	INDF1	使用 FSR1 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
102	PCL	程序计数器 (PC) 低 8 位								0000 0000
103	STATUS	–	–	–	/TO	/PD	Z	DC	C	0001 1xxx
104	FSR0L	间接寻址指针寄存器 FSR0 低 8 位								xxxx xxxx
105	FSR0H	间接寻址指针寄存器 FSR0 高 8 位								xxxx xxxx
106	FSR1L	间接寻址指针寄存器 FSR1 低 8 位								xxxx xxxx
107	FSR1H	间接寻址指针寄存器 FSR1 高 8 位								xxxx xxxx
108	BSREG	存储区选择寄存器								xxxx xxxx
109	WREG	工作寄存器 W								xxxx xxxx
10A	PCLATH	–	程序计数器 (PC) 高 7 位锁存器							–000 0000
10B	INTCON	GIE	PEIE	EEIE	LVDIE	OSFIE	EEIF	LVDIF	OSFIF	0000 0000
10C	LATA	LATA[7:0]								xxxx xxxx
10D	LATB	LATB[7:0]								xxxx xxxx
10E	LATC	LATC[7:0]								xxxx xxxx
10E	LATC	–	–	–	–	–	–	LATC[1:0]		–––– ––xx
111	TIM4CR1	T4ARPE	–	T4CKS[1:0]		T4OPM	T4URS	T4UDIS	T4CEN	0–00 0000
112	TIM4IER	–	–	–	–	–	–	–	T4UIE	---- ---0
113	TIM4SR	–	–	–	–	–	–	–	T4UIF	---- ---0
114	TIM4EGR	–	–	–	–	–	–	–	T4UG	---- ---0
115	TIM4CNTR	T4CNT[7:0]								0000 0000
116	TIM4PSCR	–	–	–	–	–	T4PSC[2:0]			---- -000
117	TIM4ARR	T4ARR[7:0]								1111 1111
118	EPS0	EPS0[7:0]								0000 0000
119	EPS1	EPS1[7:0]								0000 0000
11A	PSRC0	PSRCA[7:0]								1111 1111
11B	PSRC1	PSRCB[7:0]								1111 1111
11C	PSRC2					PSRCC[3:2]		PSRCC[1:0]		---- 1111
11E	ITYPE0	ITYPE0[7:0]								0000 0000
11F	ITYPE1	ITYPE1[7:0]								0000 0000
120–16F		SRAM BANK2 (80Bytes)								xxxx xxxx
170–17F		SRAM BANK2 (16Bytes), 访问 BANK0's 物理地址 0x70–0x7F								xxxx xxxx

表 14-5 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	INDF0	使用 FSR0 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
181	INDF1	使用 FSR1 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 (PC) 低 8 位								0000 0000
183	STATUS	–	–	–	/TO	/PD	Z	DC	C	0001 1xxx
184	FSR0L	间接寻址指针寄存器 FSR0 低 8 位								xxxx xxxx
185	FSR0H	间接寻址指针寄存器 FSR0 高 8 位								xxxx xxxx
186	FSR1L	间接寻址指针寄存器 FSR1 低 8 位								xxxx xxxx
187	FSR1H	间接寻址指针寄存器 FSR1 高 8 位								xxxx xxxx
188	BSREG	存储区选择寄存器								xxxx xxxx
189	WREG	工作寄存器 W								xxxx xxxx
18A	PCLATH	–	程序计数器 (PC) 高 7 位锁存器							–000 0000
18B	INTCON	GIE	PEIE	EEIE	LVDIE	OSFIE	EEIF	LVDIF	OSFIF	0000 0000
18C	WPUA	WPUA[7:0]								0000 0000
18D	WPUB	WPUB[7:0]								0000 0000
18E	WPUC	–	–	–	–	–	–	WPUC[1:0]		–––– ––00
191	EEADRL	EEADR[7:0]								0000 0000
192	EEADRH	–	–	–	–	EEADR[11:8]				-000 0000
193	EEDATL	EEDAT[7:0]								xxxx xxxx
194	EEDATH	–	EEDAT[13:8]							-xxx xxxx
195	EECON1	EEPGD	CFGS	–	FREE	WRERR	WREN	WR	RD	00-0 x000
196	EECON2	EEPROM 控制寄存器 2								xxxx xxxx
197	ANSELA	模拟管脚设置寄存器								0000 0000
199	LVDCON	SLVREN	LVDM	–	LVDEN	LVDW	LVDL[3:0]			0000 0000
19A	PSINK0	PSINK0[7:0]								0000 0000
19B	PSINK1	PSINK1[7:0]								0000 0000
19C	PSINK2	–	–	–	–	–	–	PSINK2[1:0]		---- --00
19D	MISC0	–	–	–	–	–	–	WCKSEL[1:0]		---- --00
19E	AFP0	–	–	AFP0[5:1]					–	--00 000-
19F	AFP1	–	–	AFP1[5:0]						---00 0000
1A0–1EF		SRAM BANK3 (80Bytes)								xxxx xxxx
1F0–1FF		SRAM BANK3 (16Bytes), 访问 BANK0's 物理地址 0x70–0x7F								xxxx xxxx

表 14-6 SFR, BANK3

地址	名称	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
200	INDF0	使用 FSR0 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
201	INDF1	使用 FSR1 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
202	PCL	程序计数器 (PC) 低 8 位								0000 0000
203	STATUS	–	–	–	/TO	/PD	Z	DC	C	0001 1xxx
204	FSR0L	间接寻址指针寄存器 FSR0 低 8 位								xxxx xxxx
205	FSR0H	间接寻址指针寄存器 FSR0 高 8 位								xxxx xxxx
206	FSR1L	间接寻址指针寄存器 FSR1 低 8 位								xxxx xxxx
207	FSR1H	间接寻址指针寄存器 FSR1 高 8 位								xxxx xxxx
208	BSREG	存储区选择寄存器								xxxx xxxx
209	WREG	工作寄存器 W								xxxx xxxx
20A	PCLATH	–	程序计数器 (PC) 高 7 位锁存器							–000 0000
20B	INTCON	GIE	PEIE	EEIE	LVDIE	OSFIE	EEIF	LVDIF	OSFIF	0000 0000
20C	WPDA	WPDA[7:0]								0000 0000
20D	WPDB	WPDB[7:0]								0000 0000
20E	WPDC	–	–	–	–	–	–	WPDC[1:0]		---- --00
211	TIM1CR1	T1ARPE	T1CMS[1:0]		TIDIR	T1OPM	TIURS	T1UDS	T1CEN	0000 0000
212	TIM1CR2	T1ENCTRL	–	–	–	–	T1COMS	–	T1CCPC	0--- -0-0
213	TIM1SMCR	–	T1TS[2:0]			–	T1SMS[2:0]			-000 -000
215	TIM1IER	T1BIE	T1TIE	T1COMIE	T1CC4IE	T1CC3IE	T1CC2IE	T1CC1IE	T1UIE	0000 0000
216	TIM1SR1	T1BIF	T1TIF	T1COMIF	T1CC4IF	T1CC3IF	T1CC2IF	T1CC1IF	T1UIF	0000 0000
217	TIM1SR2	-	-	-	T1CC4OF	T1CC3OF	T1CC2OF	T1CC1OF	-	---0 000-
218	TIM1EGR	T1BG	T1TG	T1COMG	T1CC4G	T1CC3G	T1CC2G	T1CC1G	T1UG	0000 0000
219	TIM1CCMR1 (output mode)	T1OC1CE	T1OC1M[2:0]			T1OC1PE	T1OC1FE	T1CC1S[1:0]		0000 0000
	TIM1CCMR1 (input mode)	T1IC1F[3:0]				T1IC1PSC[1:0]		T1CC1S[1:0]		0000 0000
21A	TIM1CCMR2 (output mode)	T1OC2CE	T1OC2M[2:0]			T1OC2PE	T1OC2FE	T1CC2S[1:0]		0000 0000
	TIM1CCMR2 (input mode)	T1IC2F[3:0]				T1IC2PSC[1:0]		T1CC2S[1:0]		0000 0000
21B	TIM1CCMR3 (output mode)	T1OC3CE	T1OC3M[2:0]			T1OC3PE	T1OC3FE	T1CC3S[1:0]		0000 0000
	TIM1CCMR3 (input mode)	T1IC3F[3:0]				T1IC3PSC[1:0]		T1CC3S[1:0]		0000 0000
21C	TIM1CCMR4 (output mode)	T1OC4CE	T1OC4M[2:0]			T1OC4PE	T1OC4FE	T1CC4S[1:0]		0000 0000
	TIM1CCMR4 (input mode)	T1IC4F[3:0]				T1IC4PSC[1:0]		T1CC4S[1:0]		0000 0000
21D	TIM1CCER1	T1CC2NP	T1CC2NE	T1CC2P	T1CC2E	T1CC1NP	T1CC1NE	T1CC1P	T1CC1E	0000 0000
21E	TIM1CCER2	T1GP	T1SMOD	T1CC4P	T1CC4E	T1CC3NP	T1CC3NE	T1CC3P	T1CC3E	0000 0000
21F	ODCON0	–	–	–	–	–	–	–	UROD	---- ---0
220–26F		SRAM BANK4 (80Bytes)								xxxx xxxx
270–27F		SRAM BANK4 (16Bytes), 访问 BANK0's 物理地址 0x70–0x7F								xxxx xxxx

表 14-7 SFR, BANK4

地址	名称	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
280	INDF0	使用 FSR0 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
281	INDF1	使用 FSR1 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
282	PCL	程序计数器 (PC) 低 8 位								0000 0000
283	STATUS	–	–	–	/TO	/PD	Z	DC	C	0001 1xxx
284	FSR0L	间接寻址指针寄存器 FSR0 低 8 位								xxxx xxxx
285	FSR0H	间接寻址指针寄存器 FSR0 高 8 位								xxxx xxxx
286	FSR1L	间接寻址指针寄存器 FSR1 低 8 位								xxxx xxxx
287	FSR1H	间接寻址指针寄存器 FSR1 高 8 位								xxxx xxxx
288	BSREG	存储区选择寄存器								xxxx xxxx
289	WREG	工作寄存器 W								xxxx xxxx
28A	PCLATH	–	程序计数器 (PC) 高 7 位锁存器							–000 0000
28B	INTCON	GIE	PEIE	EEIE	LVDIE	OSFIE	EEIF	LVDIF	OSFIF	0000 0000
28C	TIM1CNTRH	T1CNT[15:8]								0000 0000
28D	TIM1CNTRL	T1CNT [7:0]								0000 0000
28E	TIM1PSCRH	T1PSC[15:8]								0000 0000
28F	TIM1PSCRL	T1PSC[7:0]								0000 0000
290	TIM1ARRH	T1ARR[15:8]								1111 1111
291	TIM1ARRL	T1ARR[7:0]								1111 1111
292	TIM1RCR	TIREP[7:0]								0000 0000
293	TIM1CCR1H	T1CCR1[15:8]								0000 0000
294	TIM1CCR1L	T1CCR1[7:0]								0000 0000
295	TIM1CCR2H	T1CCR2[15:8]								0000 0000
296	TIM1CCR2L	T1CCR2[7:0]								0000 0000
297	TIM1CCR3H	T1CCR3[15:8]								0000 0000
298	TIM1CCR3L	T1CCR3[7:0]								0000 0000
299	TIM1CCR4H	T1CCR4[15:8]								0000 0000
29A	TIM1CCR4L	T1CCR4[7:0]								0000 0000
29B	TIM1BKR	T1MOE	T1AOE	T1BKP	T1BKE	T1OSSR	T1OSSI	T1LOCK[1:0]		0000 0000
29C	TIM1DTR	T1DGT[7:0]								0000 0000
29D	TIM1OISR	–	T1OIS4	T1OIS3N	T1OIS3	T1OIS2N	T1OIS2	T1OIS1N	T1OIS1	-000 0000
29E	TIM2CCR3H	T2CCR3[15:8]								0000 0000
29F	TIM2CCR3L	T2CCR3 [7:0]								0000 0000
2A0–2EF		SRAM BANK5 (80Bytes)								xxxx xxxx
2F0–2FF		SRAM BANK5 (16Bytes), 访问 BANK0's 物理地址 0x70–0x7F								xxxx xxxx

表 14-8 SFR, BANK5

地址	名称	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
300	INDF0	使用 FSR0 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
301	INDF1	使用 FSR1 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
302	PCL	程序计数器 (PC) 低 8 位								0000 0000
303	STATUS	–	–	–	/TO	/PD	Z	DC	C	0001 1xxx
304	FSR0L	间接寻址指针寄存器 FSR0 低 8 位								xxxx xxxx
305	FSR0H	间接寻址指针寄存器 FSR0 高 8 位								xxxx xxxx
306	FSR1L	间接寻址指针寄存器 FSR1 低 8 位								xxxx xxxx
307	FSR1H	间接寻址指针寄存器 FSR1 高 8 位								xxxx xxxx
308	BSREG	存储区选择寄存器								xxxx xxxx
309	WREG	工作寄存器 W								xxxx xxxx
30A	PCLATH	–	程序计数器 (PC) 高 7 位锁存器							–000 0000
30B	INTCON	GIE	PEIE	EEIE	LVDIE	OSFIE	EEIF	LVDIF	OSFIF	0000 0000
30C	TIM2CR1	T2ARPE	–	–	–	T2OPM	T2URS	T2UDIS	T2CEN	0--- 0000
30D	TIM2IER	–	–	–	–	T2CC3IE	T2CC2IE	T2CC1IE	T2UIE	---- 0000
30E	TIM2SR1	–	–	–	–	T2CC3IF	T2CC2IF	T2CC1IF	T2UIF	---- 0000
30F	TIM2SR2	–	–	–	–	T2CC3OF	T2CC3OF	T2CC3OF	–	---- 000-
310	TIM2EGR	–	–	–	–	T2CC3G	T2CC2G	T2CC1G	T2UG	---- 0000
311	TIM2CCMR1 (output mode)	–	T2OC1M[2:0]			T2OC1PE	-	T2CC1S[1:0]		-000 0-00
	TIM2CCMR1 (input mode)	T2IC1F[3:0]				T2IC1PSC[1:0]		T2CC1S[1:0]		0000 0000
312	TIM2CCMR2 (output mode)	–	T2OC2M[2:0]			T2OC2PE	–	T2CC2S[1:0]		-000 0-00
	TIM2CCMR2 (input mode)	T2IC2F[3:0]				T2IC2PSC[1:0]		T2CC2S[1:0]		0000 0000
313	TIM2CCMR3 (output mode)	–	T2OC3M[2:0]			T2OC3PE	–	T2CC3S[1:0]		-000 0-00
	TIM2CCMR3 (input mode)	T2IC3F[3:0]				T2IC3PSC[1:0]		T2CC3S[1:0]		0000 0000
314	TIM2CCER1	–	–	T2CC2P	T2CC2E	–	–	T2CC1P	T2CC1E	--00 --00
315	TIM2CCER2	–	–	–	–	–	–	T2CC3P	T2CC3E	---- --00
316	TIM2CNTRH	T2CNT[15:8]								0000 0000
317	TIM2CNTRL	T2CNT[7:0]								0000 0000
318	TIM2PSCR	–	–	–	–	T2PSC[3:0]				---- 0000
319	TIM2ARRH	T2ARR[15:8]								1111 1111
31A	TIM2ARRL	T2ARR[7:0]								1111 1111
31B	TIM2CCR1H	T2CCR1[15:8]								0000 0000
31C	TIM2CCR1L	T2CCR1[7:0]								0000 0000
31D	TIM2CCR2H	T2CCR2[15:8]								0000 0000
31E	TIM2CCR2L	T2CCR2[7:0]								0000 0000
31F	TCKSRC	LFMOD	T2CKSRC[2:0]			–	T2CKSRC[2:0]			0000 -000
320–32F		SRAM BANK6 (16Bytes)								xxxx xxxx
370–37F		SRAM BANK6 (16Bytes), 访问 BANK0's 物理地址 0x70–0x7F								xxxx xxxx

表 14-9 SFR, BANK6

地址	名称	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
380	INDF0	使用 FSR0 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
381	INDF1	使用 FSR1 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
382	PCL	程序计数器 (PC) 低 8 位								0000 0000
383	STATUS	–	–	–	/TO	/PD	Z	DC	C	0001 1xxx
384	FSR0L	间接寻址指针寄存器 FSR0 低 8 位								xxxx xxxx
385	FSR0H	间接寻址指针寄存器 FSR0 高 8 位								xxxx xxxx
386	FSR1L	间接寻址指针寄存器 FSR1 低 8 位								xxxx xxxx
387	FSR1H	间接寻址指针寄存器 FSR1 高 8 位								xxxx xxxx
388	BSREG	存储区选择寄存器								xxxx xxxx
389	WREG	工作寄存器 W								xxxx xxxx
38A	PCLATH	–	程序计数器 (PC) 高 7 位锁存器							–000 0000
38B	INTCON	GIE	PEIE	EEIE	LVDIE	OSFIE	EEIF	LVDIF	OSFIF	0000 0000
3F0–3FF		SRAM BANK7 (16Bytes), 访问 BANK0's 物理地址 0x70–0x7F								xxxx xxxx

表 14-10 SFR, BANK7

地址	名称	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
400	INDF0	使用 FSR0 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
401	INDF1	使用 FSR1 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
402	PCL	程序计数器 (PC) 低 8 位								0000 0000
403	STATUS	–	–	–	/TO	/PD	Z	DC	C	0001 1xxx
404	FSR0L	间接寻址指针寄存器 FSR0 低 8 位								xxxx xxxx
405	FSR0H	间接寻址指针寄存器 FSR0 高 8 位								xxxx xxxx
406	FSR1L	间接寻址指针寄存器 FSR1 低 8 位								xxxx xxxx
407	FSR1H	间接寻址指针寄存器 FSR1 高 8 位								xxxx xxxx
408	BSREG	存储区选择寄存器								xxxx xxxx
409	WREG	工作寄存器 W								xxxx xxxx
40A	PCLATH	–	程序计数器 (PC) 高 7 位锁存器							–000 0000
40B	INTCON	GIE	PEIE	EEIE	LVDIE	OSFIE	EEIF	LVDIF	OSFIF	0000 0000
41A	ADCON3	ADFBEN	ADCMPOP	ADCMPEN	ADCMPO	LEBADT	–	ELVDS[1:0]		0000 0-00
41B	ADCMPH	ADCMPH[7:0]								0000 0000
41C	LEBCON	LEBEN	LEBCH[1:0]		–	EDGS	BKS[2:0]			000- 0000
41D	MSCKCON	–	–	–	–	–	–	CKMAVG	CKCNTI	---- --01
41E	SOSCPRL	SOSCPR[7:0]								1111 1111
41F	SOSCPRH	–	–	–	–	SOSCPR[11:8]				---- 1111
470–47F		SRAM BANK8 (16Bytes), 访问 BANK0's 物理地址 0x70–0x7F								xxxx xxxx

表 14-11 SFR, BANK8

地址	名称	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
480	INDF0	使用 FSR0 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
481	INDF1	使用 FSR1 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
482	PCL	程序计数器 (PC) 低 8 位								0000 0000
483	STATUS	–	–	–	/TO	/PD	Z	DC	C	0001 1xxx
484	FSR0L	间接寻址指针寄存器 FSR0 低 8 位								xxxx xxxx
485	FSR0H	间接寻址指针寄存器 FSR0 高 8 位								xxxx xxxx
486	FSR1L	间接寻址指针寄存器 FSR1 低 8 位								xxxx xxxx
487	FSR1H	间接寻址指针寄存器 FSR1 高 8 位								xxxx xxxx
488	BSREG	存储区选择寄存器								xxxx xxxx
489	WREG	工作寄存器 W								xxxx xxxx
48A	PCLATH	–	程序计数器 (PC) 高 7 位锁存器							–000 0000
48B	INTCON	GIE	PEIE	EEIE	LVDIE	OSFIE	EEIF	LVDIF	OSFIF	0000 0000
48C	URDATAL	DATAL[7:0]								0000 0000
48D	URDATAH	–							DATAH	---- ---0
48E	URIER	–	–	TCEN	–	IDELE	RXSE	URTE	URRXNE	--0- 0000
48F	URLCR	–	BKREQ	–	EVEN	PEN	URSTOP	–	LTH	-0-0 00-0
490	URLCREXT	–	–	–	–	–	–	RWU	EXTEN	---- --00
491	URMCR	–	–	SIRLP	TXEN	RXEN	WAKE	HDSEL	SIREN	--00 0000
492	URLSR	ADDRF	IDLEF	TXEF	BKF	FEF	PEF	OVERF	RXNEF	0000 0000
493	URRAR	–	–	–	–	RAR[3:0]				---- 0000
494	URDLL	DLL[7:0]								0000 0000
495	URDLH	DLH[7:0]								0000 0000
496	URABCR	–	–	–	–	ABRE	ABRM	ABRF	ABREN	---- 0000
497	URSYNCR	–	–	–	–	LBCL	URCPHA	URCPOL	SYNEN	---- 0000
498	URLINCR	–	–	–	LINEN	BLTH[3:0]				---0 0000
499	URSDCR0	–	NACK	CKOE	SDEN	–	–	–	–	-000 0000
49A	URSDCR1	GT[7:0]								0000 0000
49B	URSDCR2	PSC[7:0]								0000 0000
49C	URTC	–	–	–	–	–	–	–	TCF	---- ---1
4F0–4FF		SRAM BANK9 (16Bytes), 访问 BANK0's 物理地址 0x70–0x7F								xxxx xxxx

表 14-12 SFR, BANK9

地址	名称	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
500	INDF0	使用 FSR0 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
501	INDF1	使用 FSR1 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
502	PCL	程序计数器 (PC) 低 8 位								0000 0000
503	STATUS	–	–	–	/TO	/PD	Z	DC	C	0001 1xxx
504	FSR0L	间接寻址指针寄存器 FSR0 低 8 位								xxxx xxxx
505	FSR0H	间接寻址指针寄存器 FSR0 高 8 位								xxxx xxxx
506	FSR1L	间接寻址指针寄存器 FSR1 低 8 位								xxxx xxxx
507	FSR1H	间接寻址指针寄存器 FSR1 高 8 位								xxxx xxxx
508	BSREG	存储区选择寄存器								xxxx xxxx
509	WREG	工作寄存器 W								xxxx xxxx
50A	PCLATH	–	程序计数器 (PC) 高 7 位锁存器							---0 0000
50B	INTCON	GIE	PEIE	EEIE	LVDIE	OSFIE	EEIF	LVDIF	OSFIF	0000 0000
570–57F		SRAM BANK10 (16Bytes), 访问 BANK0's 物理地址 0x70–0x7F								xxxx xxxx

表 14-13 SFR, BANK10

地址	名称	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
F80	INDF0	使用 FSR0 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
F81	INDF1	使用 FSR1 的内容对数据存储器进行访问 (非物理寄存器)								xxxx xxxx
F82	PCL	程序计数器 (PC) 低 8 位								0000 0000
F83	STATUS	–	–	–	/TO	/PD	Z	DC	C	0001 1xxx
F84	FSR0L	间接寻址指针寄存器 FSR0 低 8 位								xxxx xxxx
F85	FSR0H	间接寻址指针寄存器 FSR0 高 8 位								xxxx xxxx
F86	FSR1L	间接寻址指针寄存器 FSR1 低 8 位								xxxx xxxx
F87	FSR1H	间接寻址指针寄存器 FSR1 高 8 位								xxxx xxxx
F88	BSREG	存储区选择寄存器								xxxx xxxx
F89	WREG	工作寄存器 W								xxxx xxxx
F8A	PCLATH	–	程序计数器 (PC) 高 7 位锁存器							–000 0000
F8B	INTCON	GIE	PEIE	EEIE	LVDIE	OSFIE	EEIF	LVDIF	OSFIF	0000 0000
FE4	STATUS_SHAD	STATUS 影子寄存器								xxxx xxxx
FE5	WREG_SHAD	WREG 影子寄存器								xxxx xxxx
FE6	BSREG_SHAD	BSREG 影子寄存器								xxxx xxxx
FE7	PCLATH_SHAD	PCLATH 影子寄存器								xxxx xxxx
FE8	FSR0L_SHAD	FSR0L 影子寄存器								xxxx xxxx
FE9	FSR0H_SHAD	FSR0H 影子寄存器								xxxx xxxx
FEA	FSR1L_SHAD	FSR1L 影子寄存器								xxxx xxxx
FEB	FSR1H_SHAD	FSR1H 影子寄存器								xxxx xxxx
FEC	_	_								_
FED	STKPTR	STKPTR								xxxx xxxx
FEE	TOSL	TOSL								xxxx xxxx
FEF	TOSH	TOSH								xxxx xxxx
FF0–FFF		SRAM BANK31 (16Bytes), 访问 BANK0's 物理地址 0x70–0x7F								xxxx xxxx

表 14-14 SFR, BANK31

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；

14.3 STATUS 寄存器

名称	状态	寄存器	地址	复位值
/ TO	超时标志位 1 = 上电后，执行了 CLRWDT 或 SLEEP 指令 0 = 发生 WDT 超时溢出	STATUS[4]	Bank 首地址 + 0x03	RO−1
/PD	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
DC	半进位 / 半借位 (ADDWR, ADDWI, SUBWI, SUBWR)：结果的第 4 低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 (ADDWR, ADDWI, SUBWI, SUBWR)：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 14-15 STATUS 寄存器

注：

1. 同其他寄存器一样，STATUS 状态寄存器也可以作为任何指令的目标寄存器。但如果一条影响 Z、DC 或 C 位的指令以 STATUS 作为目标寄存器，那么对这三位的写操作将被禁止，Z、DC 和 C 位只受运算结果影响从而被置 1 或清 0。此时，当执行一条以 STATUS 作为目标寄存器的指令后，STATUS 的内容可能与预期不一致。
2. 建议只使用 BCR、BSR、SWAPR 和 STR 指令来操作 STATUS 寄存器。

14.4 堆栈

FT61F14x 的硬件堆栈为 16 级深 x 15 位宽。堆栈空间与程序 PROM、数据 EEPROM 或数据存储区 SRAM 相互独立。

TOSH:TOSL 指向栈顶，STKPTR 为堆栈指针的当前值。访问堆栈时，可调整用来定位 TOSH:TOSL 的 STKPTR 值，然后对 TOSH:TOSL 执行读/写操作。

正常程序运行期间，LCALL、CALLW 和中断使 STKPTR 值递增 1，PC 值被压入堆栈。而当执行 RETW、RET 和 RETI 指令时，PC 值从堆栈弹出，STKPTR 值递减 1。PCLATH 值不受压栈或出栈操作的影响。可通过读取 STKPTR 以查看可用堆栈空间。

STKPTR 为 5 位，允许检测上溢和下溢。在压满 16 级后再执行压栈操作，将产生上溢，STKOVF 标志位将被置 1。在弹出第 1 级后再执行出栈操作，将产生下溢，STKUNF 标志位将被置 1。上溢或下溢事件都将导致系统复位，且 16 级堆栈将全部清 0。

注意：在允许中断的情况下修改 STKPTR 时需谨慎。

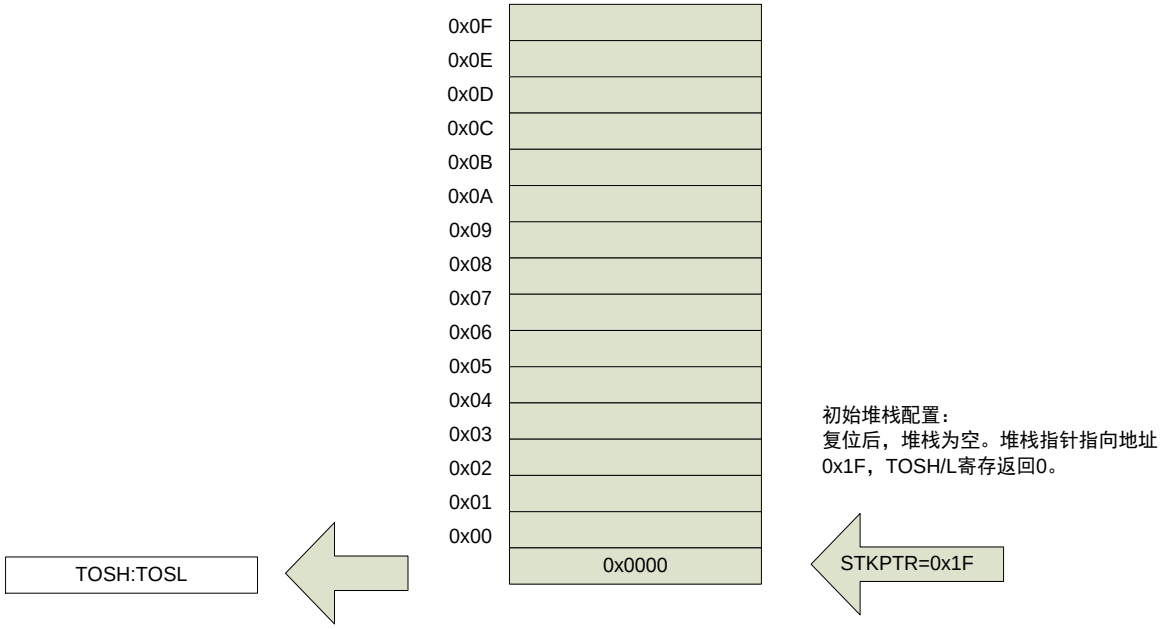


图 14-2 软件访问堆栈

14.5 间接寻址

INDFn 不是物理存在的寄存器，对 INDFn 进行寻址将产生间接寻址。任何访问 INDFn 寄存器的指令，实际上是对文件选择寄存器(File Select Register, FSRn) 所指向的单元进行存取。间接对 INDF 进行读操作将返回 0，间接对 INDF 进行写操作将导致空操作(可能会影响状态标志位)。FSRnH:FSRnL 组成的 16 位地址允许对 65536 个地址单元进行寻址，可划分为 3 个存储区：

- 传统数据存储器
- 线性数据存储器
- 闪存程序存储器

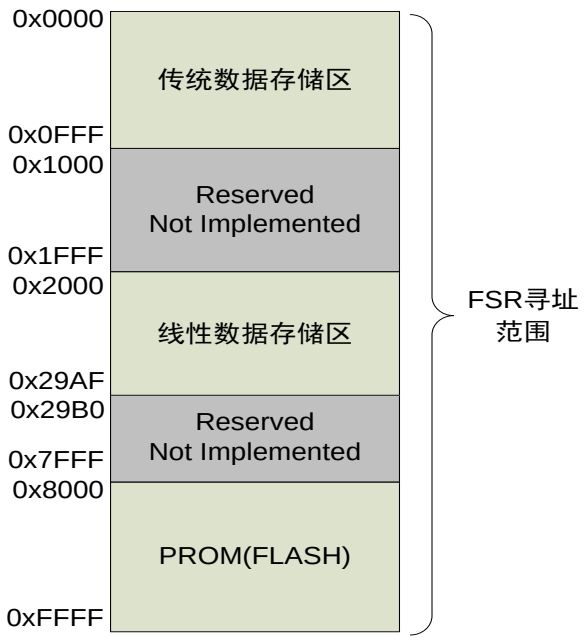


图 14-3 间接寻址

14.5.1 传统数据存储器

传统数据存储器，即用户寄存器，地址范围为 0x0000 ~ 0x0FFF，对应所有 SFR 和 SRAM 的绝对地址。

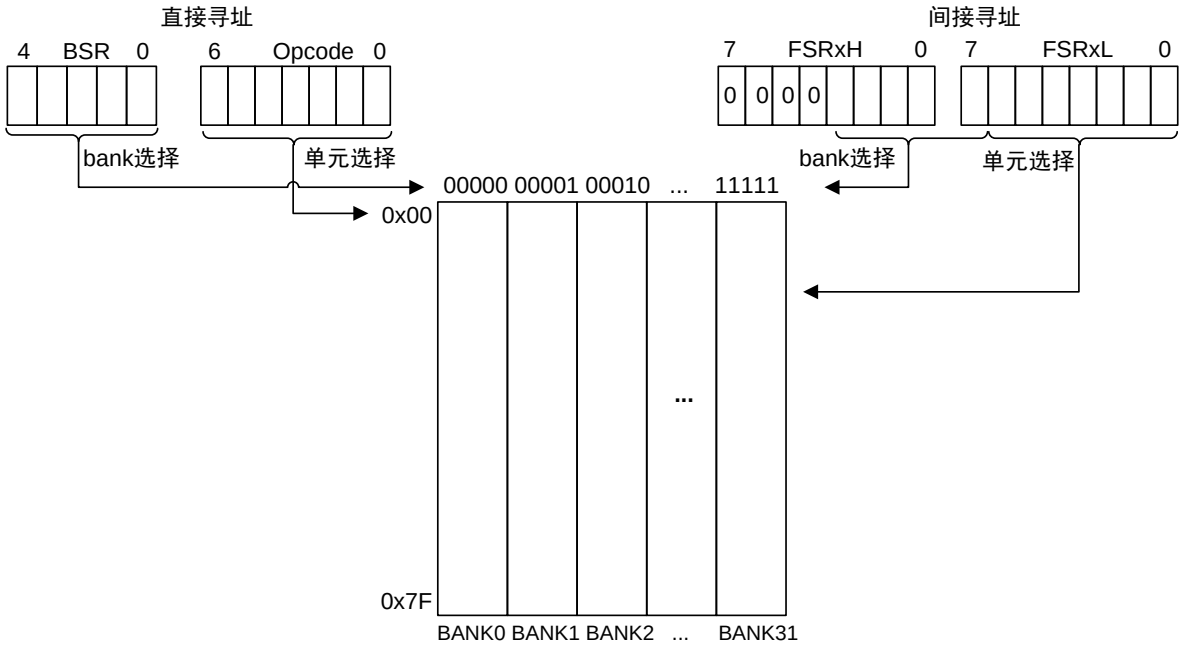


图 14-4 传统数据存储器映射

14.5.2 线性数据存储器

线性数据存储器，地址范围为 0x2000 ~ 0x29AF。该区域为虚拟区域，指向所有 Bank 中的 80 字节 SRAM 存储区（不包括 16 字节的公共 SRAM），未实现的存储区（Bank6~30）读为 0x00。

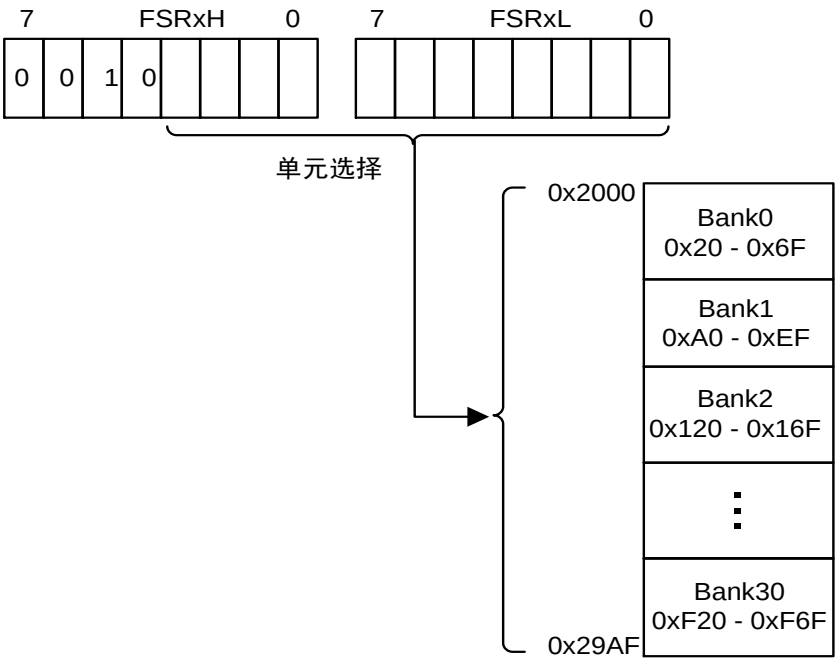


图 14-5 线性数据存储区映射

14.5.3 闪存程序存储器

当 FSRnH 的 MSB 置 1 时，FSRnH:FSRnL 的低 15 位为需要访问的程序 PROM 存储区的地址，相应的低 8 位数据可通过 INDFn 进行读取。通过 FSR/INDF 无法对程序 PROM 执行写操作，而读操作需要 2 个指令周期。

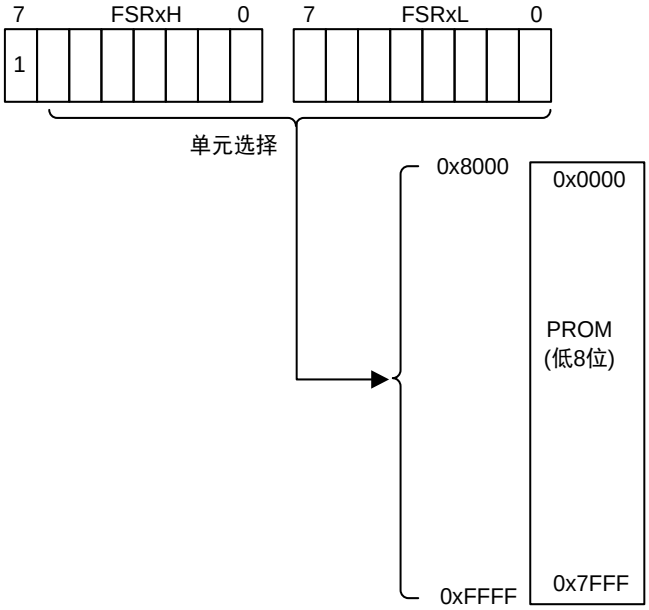


图 14-6 程序存储区映射

程序 PROM 存储器中的常数，可通过 FSR 间接访问，或通过 RETW 指令进行读取。

通过 FSR 间接访问的示例程序：

```
constants
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
```

通过 RETW 和 BRW 指令对常数表进行读取的示例程序：

```
constants
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
```

15 指令集 (INSTRUCTION SET)

汇编语法	功能	运算	指令周期	状态位
NOP	空操作	None	1	NONE
SLEEP	进入 SLEEP 模式	0 → WDT; Stop OSC	1	/PF, /TF
RESET	软件复位	复位寄存器 PCON	1	NONE
CLRWDT	清看门狗 (喂狗)	0 → WDT	1	/PF, /TF
LJUMP N	无条件跳转	N → PC	2	NONE
BRA k	相对跳转 (地址范围有限制)	PC + 1 + k → PC	2	NONE
BRW	将寄存器 W 的值作为偏移量进行相对跳转	PC + w → PC	2	NONE
LCALL N	调用子程序	N → PC; PC + 1 → Stack	2	NONE
CALLW	调用地址由寄存器 W 指定的子程序	W → PC; PC + 1 → Stack	2	NONE
RETW	将立即数 I 送入 W 并返回	I → W, Stack → PC	2	NONE
RETI	从中断返回	Stack → PC; 1 → GIE	2	NONE
RET	从子程序返回	Stack → PC	2	NONE
BCR R, b	将寄存器 R 的 b 位清 0	0 → R(b)	1	NONE
BSR R, b	将寄存器 R 的 b 位置 1	1 → R(b)	1	NONE
CLRR R	将寄存器 R 清 0	0 → R	1	Z
LDR R, d (MOVF)	将 R 存到 d	R → d	1	Z
COMR R, d	R 的反码	/R → d	1	Z
INCR R, d	R + 1	R + 1 → d	1	Z
INCRSZ R, d	R + 1，结果为 0 则跳过	R + 1 → d	1	NONE
DECR R, d	R − 1	R − 1 → d	1	Z
DECRSZ R, d	R − 1，结果为 0 则跳过	R − 1 → d	1	NONE
SWAPR R, d	将寄存器 R 的半字节交换	R(0-3)R(4−7) → d	1	NONE
RRR R, d	R 带进位循环右移	R(0) → C; R(n) → R(n−1); C → R(7);	1	C
RLR R, d	R 带进位循环左移	R(7) → C; R(n) → R(n+1); C → R(0);	1	C
LSRF f, d	F 逻辑右移	0→ f(7); f(n+1) → R(n); f(0)→ C;	1	C, Z
LSLF f, d	F 逻辑左移	f(7) → C; f(n) → R(n+1); 0→ R(0);	1	C, Z
ASRF f, d	算术右移	f(7) → R(7); f(n+1) →R(n); f(0) →C;	1	C, Z
BTSC R, b	位测试，结果为 0 则跳过	Skip if R(b)=0	1	NONE

汇编语法	功能	运算	指令周期	状态位
BTSS R, b	位测试，结果为 1 则跳过	Skip if R(b)=1	1	NONE
CLRW	将工作寄存器 W 清 0	0 → W	1	Z
STR R (MOVWF)	将 W 存到 R	W → R	1	NONE
ADDWR R, d	W 与 R 相加	W + R → d	1	C, DC, Z
ADDWFC R, d	W 与 R 相加(带进位)	W + R + C → d	1	C, DC, Z
SUBWR R, d	R 减 W	R − W → d	1	C, DC, Z
SUBWFB R, d	R 减 W (带借位)	R − W – (/B) → d	1	C, DC, Z
ANDWR R, d	W 与 R 相与	R & W → d	1	Z
IORWR R, d	W 与 R 相或	W | R → d	1	Z
XORWR R, d	W 与 R 异或	W ^ R → d	1	Z
LDWI I (MOVLW)	将立即数存到 W	I → W	1	NONE
ANDWI I	W 与立即数 I 相与	I & W → W	1	Z
IORWI I	W 与立即数 I 相或	I | W → W	1	Z
XORWI I	W 与立即数 I 异或	I ^ W → W	1	Z
ADDWI I	W 与立即数 I 相加	I + W → W	1	C, DC, Z
SUBWI I	立即数 I 减 W	I − W → W	1	C, DC, Z
RETW I	返回，将立即数 I 存到 W	Stack → PC; I → W		NONE
MOVLB k	将立即数 I 送到存储区选择寄存器 BSR	K → BSR	1	NONE
ADDFSR FSRn, k	立即数 k 与 FSRn 相加	FSRn + k → FSRn	1	NONE
MOVLP	将立即数 k 送到 PCLATH	k → PCLATH	1	NONE
MOVIW mm	将 FSRn 的内容送到 W	FSRn → W	1	Z
MOVWI mm	将 W 的内容送到 FSRn	W → FSRn	1	NONE

表 15-1 49 条指令集

字段	描述	
R(f)	SFR/SRAM 地址	
W	工作寄存器	
b	8-bit 寄存器 R / RAM 中的位地址	
I/ Imm (k)	立即数字段、常数或标号	
x	不关心，值可以为 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
mm	预/后增/减模式选择 (++FSRn, --FSRn, FSRn++ , FSRn--, k[FSRn])	
N	程序绝对地址	
PC	程序计数器	

字段	描述
/PD	掉电标志位
/TO	超时标志位
C	进位 / 借位
DC	半进位 / 半借位
Z	0 标志位

表 15-2 操作码字段

名称	状态	寄存器	地址	复位值
Z	0 标志位：算术或逻辑运算的结果为零？ 1 = Yes 0 = No	STATUS[2]	Bank 首地址 +0x03	RW-x
DC	半进位 / 半借位 (ADDWR, ADDWI, SUBWI, SUBWR)：结果的第 4 低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 (ADDWR, ADDWI, SUBWI, SUBWR)：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 15-3 计算状态标志位

15.1 读-修改-写 (RMW)指令

所有需要使用文件寄存器 (**表 15-1** 中助记符带 R 的指令) 的指令都会执行读-修改-写 (RMW)操作，即先把目标寄存器内容取出，根据指令修改数据，再把数据写回到目标寄存器或 W (取决于 d 和具体指令)。

举例说明：

```
BSR     FSR0L, 0;
```

上述指令在 CPU 的执行过程如下：

1）把 FSR0L 读出到临时寄存器 T；

2）把寄存器 T 或上”0000 0001”形成新数据；

3）再把新数据写回 FSR0L；

15.2 指令详细描述

ADDFSR	立即数与 FSRn 相加
语法：	[标号] ADDFSR FSRn, k
操作数：	-32 ≤ k ≤ 31 n∈[0,1]
操作：	FSR(n)+k → FSR(n)
受影响的状态位：	无
说明：	将有符号 6 位立即数 k 与 FSRnH:FSRnL 寄存器对的内容相加。 FSRn 地址范围限制为 0000h-FFFFh。传送地址超出该边界时，FSR 会发生折回。

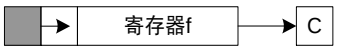
ADDWI	立即数与 W 相加
语法：	[标号] ADDWI k
操作数：	0 ≤ k ≤ 255
操作：	(W)+k → (W)
受影响的状态位：	C、DC 和 Z
说明：	将 W 寄存器的内容与 8 位立即数 k 相加，结果存入 W 寄存器。

ADDWR	W 与 f 相加
语法：	[标号]ADDWR f,d
操作数：	0≤f≤127 d∈[0,1]
操作：	(W)+(f)→(目标寄存器)
受影响的状态位：	C、DC 和 Z
说明：	将 W 寄存器的内容与寄存器 f 的内容相加。如果 d 为 0，结果存入 W 寄存器。如果 d 为 1，结果存回寄存器 f。

ANDWI	立即数和 W 作逻辑与运算
语法：	[标号] ANDWI k
操作数：	0 ≤ k ≤ 255
操作：	(W).AND.(k) → (W)
受影响的状态位：	Z
说明：	将 W 寄存器的内容与 8 位立即数 k 进行逻辑与运算。结果存入 W 寄存器。

ANDWR	W 和 f 作逻辑与运算
语法：	[标号] ANDWR f,d
操作数：	0 ≤ f ≤ 127 d∈[0,1]
操作：	(W).AND.(f) →(目标寄存器)
受影响的状态位：	Z
说明：	将 W 寄存器的内容与寄存器 f 的内容进行逻辑与运算。如果 d 为 0，结果存入 W 寄存器。如果 d 为 1，结果存回寄存器 f。

ASRF	算术右移
语法：	[标号] ASRF f {,d}
操作数：	0≤f≤127 d∈[0,1]
操作：	(f[7])→目标寄存器[7] (f[7:1])→目标寄存器[6:0], (f[0])→C
受影响的状态位：	C 和 Z
说明：	将寄存器 f 的内容连同进位标志位一起右移 1 位。MSb 保持不变。如果 d 为 0，结果存入 W 寄存器。如果 d 为 1,结果存回寄存器 f。



ADDWFC W 与 f 相加（带进位）

语法：	[标号] ADDWFC f {,d}
操作数：	0≤f≤127 d∈[0,1]
操作：	(W)+(f)+(C)→目标寄存器
受影响的状态位：	C、DC 和 Z
说明：	将 W 的内容、进位标志位与数据存储单元 f 的内容相加。如果 d 为 0，结果存入 W。如果 d 为 1，结果存入数据存储单元 f。

BCR 将 f 寄存器中的某位清零

语法：	[标号] BCR f, b
操作数：	0≤f≤127 0≤b≤7
操作：	0→(f[b])
受影响的状态位：	无
说明：	将寄存器 f 中的位 b 清零。

BRA 相对跳转

语法：	[标号] BRA 标号 [标号] BRA $+k
操作数：	-256≤标号 - PC+1≤255 -256≤k≤255
操作：	(PC)+1+k→PC
受影响的状态位：	无
说明：	将有符号 9 位立即数 k 与 PC 相加。由于 PC 将递增 1 以便取下一条指令，所以新地址将为 PC+1+k。该指令为一条双周期指令。该跳转的地址范围存在限制。

BRW 将 W 寄存器的内容作为偏移量进行相对跳转

语法：	[标号] BRW
操作数：	无
操作：	(PC)+(W)→PC
受影响的状态位：	无
说明：	将 W 的内容（无符号）与 PC 相加。由于 PC 将递增 1 以取出下一条指令，所以新地址将为 PC+1+(W)。该指令为一条双周期指令。

BSR 将 f 中的某位置 1

语法：	[标号] BSR f,b
操作数：	0≤f≤127 0≤b≤7
操作：	1→(f[b])
受影响的状态位：	无
说明：	将寄存器 f 的位 b 置 1。

BTSC 测试 f 中某位，为 0 则跳过

语法：	[标号] BTSC f,b
操作数：	0≤f≤127 0≤b≤7
操作：	如果(f[b])=0，则跳过
受影响的状态位：	无
说明：	如果寄存器 f 的位 b 为 1，则执行下一条指令。如果寄存器 f 的位 b 为 0，则丢弃下一条指令，转而执行一条 NOP 指令，从而使该指令成为双周期指令。

BTSS 测试 f 中某位,为 1 则跳过

语法：	[标号] BTSS f,b
操作数：	0≤f≤127

	0≤b≤7
操作：	如果(f[b])=1，则跳过
受影响的状态位：	无
说明：	如果寄存器 f 的位 b 为 0，则执行下一条指令。如果位 b 为 1，则丢弃下一条指令，转而执行一条 NOP 指令，从而使该指令成为双周期指令。

LCALL 调用子程序

语法：	[标号] LCALL k
操作数：	0≤k≤2047
操作：	(PC)+1→TOS， k→PC[10:0]， (PCLATH[4:3])→PC[12:11]
受影响的状态位：	无
说明：	调用子程序。首先，将返回地址(PC+1) 压入堆栈。将 11 位立即数地址装入 PC 的[10:0]位。将 PCLATH 的内容装入 PC 的高位。LCALL 是双周期指令。

CLRR 将 f 清零

语法：	[标号] CLRR f
操作数：	0≤f≤127
操作：	00h→(f) 1→Z
受影响的状态位：	Z
说明：	寄存器 f 的内容被清零，并且 Z 位被置 1。

CLRW 将 W 寄存器清零

语法：	[标号] CLRW
操作数：	无
操作：	00h→(W) 1→Z
受影响的状态位：	Z
说明：	W 寄存器被清零。全零位（Z）被置 1。

CALLW 调用地址由 W 寄存器指定的子程序

语法：	[标号] CALLW
操作数：	无
操作：	(PC)+1→TOS， (W)→PC[7:0]， (PCLATH[6:0])→PC[14:8]
受影响的状态位：	无
说明：	调用地址由 W 寄存器指定的子程序。首先，将返回地址 (PC+1) 压入返回堆栈。然后，W 的内容被装入 PC[7:0]，将 PCLATH 的内容装入 PC[14:8]。CALLW 是双周期指令。

CLRWDT 将看门狗定时器清零

语法：	[标号] CLRWDT
操作数：	无
操作：	00h→WDT 0→WDT 预分频器 1→/TO 1→/PD
受影响的状态位：	/TO 和/PD
说明：	CLRWDT 指令复位看门狗定时器及其预分频器。 状态位/TO 和/PD 均被置 1。

COMR f 取反

语法：	[标号] COMR f,d
操作数：	0≤f≤127 d∈[0,1]

操作： $(\bar{f})$→(目标寄存器)
受影响的状态位：Z
说明： 将寄存器 f 的内容取反。如果 d 为 0，结果存入 W 寄存器。如果 d 为 1，结果存回寄存器 f。

DECR f 递减 1

语法： [标号] DECR f,d
操作数： 0≤f≤127
d∈[0,1]
操作： (f)-1→(目标寄存器)
受影响的状态位：Z
说明： 将寄存器 f 的内容递减 1。如果 d 为 0，结果存入 W 寄存器。如果 d 为 1，结果存回寄存器 f。

DECRSZ f 递减 1，为 0 则跳过

语法： [标号]DECRSZ f,d
操作数： 0≤f≤127
d∈[0,1]
操作： (f)-1→(目标寄存器)；
结果=0 则跳过
受影响的状态位：无
说明： 将寄存器 f 的内容递减 1。如果 d 为 0，结果存入 W 寄存器。如果 d 为 1，结果存回寄存器 f。如果结果为 1，则执行下一条指令。如果结果为 0，则转而执行一条 NOP 指令，从而使该指令成为双周期指令。

LJUMP 无条件跳转

语法： [标号] LJUMP k
操作数： 0≤k≤2047
操作： k→PC[10:0]
PCLATH[4:3]→PC[12:11]
受影响的状态位：无
说明： LJUMP 是无条件跳转指令。将 11 位立即数值装入 PC 的[10:0]位。PC 的高位从 PCLATH[4:3]装入。LJUMP 是双周期指令。

INCR f 递增 1

语法： [标号] INCR f,d
操作数： 0≤f≤127
d∈[0,1]
操作： (f)+1→(目标寄存器)
受影响的状态位：Z
说明： 将寄存器 f 的内容递增 1。如果 d 为 0，结果存入 W 寄存器。如果 d 为 1，结果存回寄存器 f。

INCRSZ f 递增 1，为 0 则跳过

语法： [标号] INCRSZ f,d
操作数： 0≤f≤127
d∈[0,1]
操作： (f)+1→(目标寄存器)，
结果=0 则跳过
受影响的状态位：无
说明： 将寄存器 f 的内容递增 1。如果 d 为 0，结果存入 W 寄存器。如果 d 为 1，结果存回寄存器 f。如果结果为 1，则执行下一条指令。如果结果为 0，则转而执行 NOP 指令，从而使该指令成为双周期指令。

IORWI 立即数和 W 作逻辑或运算

语法： [标号] IORWI k
操作数： 0≤k≤255
操作： (W).OR.k→(W)
受影响的状态位：Z
说明： 将 W 寄存器的内容与 8 位立即

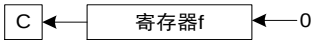
数 k 进行逻辑或运算。结果存入 W 寄存器。

IORWR W 和 f 作逻辑或运算

语法：	[标号] IORWR f,d
操作数：	0≤f≤127 d∈[0,1]
操作：	(W).OR.(f)→(目标寄存器)
受影响的状态位：	Z
说明：	将 W 寄存器的内容与寄存器 f 的内容进行逻辑或运算。如果 d 为 0，结果存入 W 寄存器。如果 d 为 1，结果存回寄存器 f。

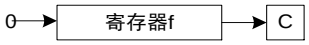
LSLF 逻辑左移

语法：	[标号] LSLF f {,d}
操作数：	0≤f≤127 d∈[0,1]
操作：	(f[7])→C (f[6:0])→目标寄存器[7:1] 0→目标寄存器[0]
受影响的状态位：	C 和 Z
说明：	将寄存器 f 的内容连同进位标志位一起左移 1 位。0 移入 LSb。如果 d 为 0，结果存入 W。如果 d 为 1，结果存回寄存器 f。



LSRF 逻辑右移

语法：	[标号] LSRF f {,d}
操作数：	0≤f≤127 d∈[0,1]
操作：	0→目标寄存器[7] (f[7:1])→目标寄存器[6:0] (f[0])→C
受影响的状态位：	C 和 Z
说明：	将寄存器 f 的内容连同进位标志位一起右移 1 位。0 移入 MSb。如果 d 为 0，结果存入 W。如果 d 为 1，结果存回寄存器 f。



LDR 传送 f

语法：	[标号] LDR f,d
操作数：	0≤f≤127 d∈[0,1]
操作：	(f)→(目标寄存器)
受影响的状态位：	Z
说明：	根据 d 的状态，将寄存器 f 的内容传送到目标寄存器。如果 d=0，目标寄存器为 W 寄存器。如果 d=1，目标寄存器为文件寄存器 f。由于状态标志位 Z 要受影响，可用 d=1 检测文件寄存器。
指令字数：	1
指令周期数：	1
示例：	LDR FSR, 0 执行指令后 W=FSR 寄存器的值 Z=1

MOVIW 将 INDFn 的内容传送到 W

语法：	[标号] MOVIW ++FSRn [标号] MOVIW --FSRn [标号] MOVIW FSRn++ [标号] MOVIW FSRn-- [标号] MOVIW k[FSRn]
操作数：	n∈[0,1] mm∈[00,01,10,11] -32≤k≤31
操作：	INDFn→W 有效地址通过以下方式确定：

•FSR+1 (预递增 1)
•FSR-1 (预递减 1)
•FSR+k (相对偏移)
执行传送指令后，FSR 值为以下任一项：
•FSR+1 (所有值都加 1)
•FSR-1 (所有值都减 1)
•不变

受影响的状态位：Z

模式	语法	mm
预递增	++FSRn	00
与递减	--FSRn	01
后递增	FSRn++	10
后递减	FSRn--	11

说明：该指令用于在 W 寄存器和任一间接寄存器 (INDFn) 之间传送数据。执行该传送指令之前/之后，将通过预/后增/减来更新指针 (FSRn)。

注：INDFn 寄存器不是物理寄存器。访问 INDFn 寄存器的指令实际上访问的是由 FSRn 指定的地址处的寄存器。

FSRn 地址范围限制为 0000h-FFFFh。地址递增/递减到超出边界时，将导致它发生折回。

•FSR+1 (预递增 1)
•FSR-1 (预递减 1)
•FSR+k (相对偏移)
执行传送指令后，FSR 值为以下任一项：
•FSR+1 (所有值都加 1)
•FSR-1 (所有值都减 1)
不变

受影响的状态位：无

模式	语法	mm
预递增	++FSRn	00
与递减	--FSRn	01
后递增	FSRn++	10
后递减	FSRn--	11

说明：该指令用于在 W 寄存器和任一个间接寄存器 (INDFn) 之间传送数据。执行该传送指令之前/之后，将通过预/后增/减来更新指针 (FSRn)。

注：INDFn 寄存器不是物理寄存器。访问 INDFn 寄存器的指令实际上访问的是由 FSRn 指定的地址处的寄存器。

FSRn 地址范围限制为 0000h-FFFFh。地址递增/递减到超出边界时，将导致它发生折回。
对于 FSRn 的递增/递减操作不会影响任何状态位。

MOVWI 将 W 的内容传送到 INDFn

语法：[标号] MOVWI ++FSRn
[标号] MOVWI --FSRn
[标号] MOVWI FSRn++
[标号] MOVWI FSRn--
[标号] MOVWI k[FSRn]

操作数：n∈[0,1]
mm∈[00,01,10,11]
-32≤k≤31

操作：W→INDFn
有效的地址由以下项决定:

MOVLB 将立即数传送到 BSR

语法：[标号] MOVLB k
操作数：0≤k≤15
操作：k→BSR
受影响的状态位：无
说明：将 5 位立即数 k 装入存储区选择寄存器 (BSR)。

MOVLP 将立即数传送到 PCLATH

语法：[标号] MOVLP k

操作数：	0≤k≤127
操作：	k→PCLATH
受影响的状态位：	无
说明：	将 7 位立即数 k 装入 PCLATH 寄存器。

LDWI	将立即数传送到 W
语法：	[标号] MOVLW k
操作数：	0≤k≤255
操作：	k→(W)
受影响的状态位：	无
说明：	将 8 位立即数 k 装入 W 寄存器。其余无关位均汇编为 0。
指令字数：	1
指令周期数：	1
示例：	LDWI 0x5A 执行指令后 W = 0x5A

NOP	空操作
语法：	[标号] NOP
操作数：	无
操作：	空操作
受影响的状态位：	无
说明：	不执行任何操作。
指令字数：	1
指令周期数：	1
示例：	NOP

STR	将 W 的内容传送到 f
语法：	[标号] STR f
操作数：	0≤f≤127
操作：	(W) →(f)
受影响的状态位：	无
说明：	将 W 寄存器的数据传送到寄存器 f。
指令字数：	1
指令周期数：	1
示例：	STR OPTION 执行指令前 OPTION = 0xFF W = 0x4F 执行指令后 OPTION = 0x4F W = 0x4F

RESET	软件复位
语法：	[标号] RESET
操作数：	无
操作：	执行器件复位。复位 PCON 寄存器的 nRI 标志。
受影响的状态位：	无
说明：	此指令可实现用软件执行硬件复位。

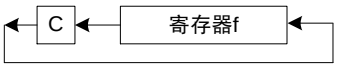
RET	从子程序返回
语法：	[标号] RET
操作数：	无
操作：	TOS→PC
受影响的状态位：	无
说明：	从子程序返回。执行出栈操作，将栈顶 (TOS) 内容装入程序计数器。这是一条双周期指令。

RETI	从中断返回
语法：	[标号] RETI
操作数：	无
操作：	TOS→PC， 1→GIE
受影响的状态位：	无
说明：	从中断返回。执行出栈操作，将栈顶 (Top-of-Stack，TOS) 的内容装入 PC。通过将全局中断允许位 GIE (INTCON[7]) 置 1，来允许中断。这是一条双周期指令。

指令字数： 1
指令周期数： 2
示例： RETI
中断后
PC = TOS
GIE = 1

RLR 对 f 执行带进位的循环左移

语法： [标号] RLR f,d
操作数： 0≤f≤127
d∈[0,1]
操作： 参见如下说明
受影响的状态位： C
说明： 将寄存器 f 的内容连同进位标志位一起循环左移 1 位。如果 d 为 0，结果存入 W 寄存器。如果 d 为 1，结果存回寄存器 f。
指令字数： 1
指令周期数： 1
示例： RLF REG1,0
执行指令前:
REG1 = 1110 0110
C = 0
执行指令后:
REG1 = 1110 0110
W = 1100 1100
C = 1



RETW 返回并将立即数送入 W

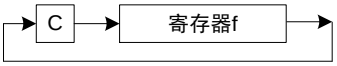
语法： [标号] RETW k
操作数： 0≤k≤255
操作： k→(W)；
TOS→PC
受影响的状态位： 无
说明： 将 8 位立即数 k 装入 W 寄存器。将栈顶内容（返回地址）装入程序计数器。这是一条双周期指令。
指令字数： 1
指令周期数： 2
示例：

```
        LCALL TABLE;W contains
        ;table offset value
        • ;W now has table value
        •
TABLE   •
        ADDWR   PC   ;W = offset
        RETW    k1   ;Begin table
        RETW    k2   ;
        •
        •
        •
        RETW    kn   ;End of table
```

执行指令前
W = 0x07
执行指令后
W = k8 的值

RRR 对 f 执行带进位的循环右移

语法： [标号] RRR f,d
操作数： 0≤f≤127
d∈[0,1]
操作： 参见如下说明
受影响的状态位： C
说明： 将寄存器 f 的内容连同进位标志位一起循环右移 1 位。如果 d 为 0，结果存入 W 寄存器。如果 d 为 1，结果存回寄存器 f



SLEEP 进入休眠模式

语法： [标号] SLEEP
操作数： 无
操作： 00h→WDT，
0→WDT 预分频器，
1→/TO，
0→/PD
受影响的状态位： /TO 和/PD
说明： 掉电状态位/PD 被清零。超时状态位/TO 被置 1。看门狗定时器及其预分频器被清零。振荡器停振，处理器进入休眠模式。

SUBWI 从立即数中减去 W

语法： [标号] SUBWI k
操作数： 0≤k≤255
操作： k-(W)→(W)
受影响的状态位： C、DC 和 Z
说明： 用 8 位立即数 k 减去 W 寄存器的内容 (通过二进制补码方式进行运算) 。结果存入 W 寄存器。

C= 0	W＞k
C= 1	W≤k
DC=0	W[3:0] > k[3:0]
DC=1	W[3:0] ≤ k[3:0]

SUBWFB f 减去 W (带借位)

语法： SUBWFB f {,d}
操作数： 0≤f≤127
d∈[0,1]
操作： (f)–(W)–(/B)→目标寄存器
受影响的状态位： C、DC 和 Z
说明： 用 f 寄存器的内容减去 W 的内容和借位标志 (进位) (通过二进制补码方式进行运算)。如果 d 为 0，结果存入 W。如果 d 为 1，结果存回寄存器 f。

SWAPR 将 f 中的两个半字节交换

语法： [标号] SWAPR f,d
操作数： 0≤f≤127
d∈[0,1]
操作： (f[3:0])→(目标寄存器[7:4])，(f[7:4])→(目标寄存器[3:0])
受影响的状态位： 无
说明： 寄存器 f 的高半字节和低半字节相互交换。如果 d 为 0，结果存入 W 寄存器。如果 d 为 1，结果存回寄存器 f。

SUBWR f 减去 W

语法： [标号] SUBWR f,d
操作数： 0≤f≤127
d∈[0,1]
操作： (f)-(W)→(目标寄存器)
受影响的状态位： C、DC 和 Z
说明： 用寄存器 f 的内容减去 W 寄存器的内容 (通过二进制补码方式进行运算)。如果 d 为 0，结果存入 W 寄存器。如果 d 为 1，结果存回寄存器 f。

C= 0	W＞f
C= 1	W≤f
DC=0	W[3:0] > f[3:0]
DC=1	W[3:0] ≤ f[3:0]

XORWR W 和 f 作逻辑异或运算

语法： [标号] XORWR f,d
操作数： 0≤f≤127
d∈[0,1]
操作： (W).XOR.(f)→(目标寄存器)
受影响的状态位： Z
说明： 将 W 寄存器的内容与寄存器 f 的内容进行逻辑异或运算。如果 d 为 0，结果存入 W 寄存器。如果 d 为 1，结果存回寄存器 f。

XORWI 立即数和 W 作逻辑异或运算

语法： [标号] XORWI k
操作数： 0≤k≤255
操作： (W).XOR.k→(W)
受影响的状态位： Z
说明： 将 W 寄存器的内容与 8 位立即数 k 进行逻辑异或运算。结果存入 W 寄存器。

16 电气特性

16.1 极限参数

工作环境温度等级 3…… $^{-}40 – ^{+}85°C$
工作环境温度等级 2…… $^{-}40 – ^{+}105°C$
工作环境温度等级 1…… $^{-}40 – ^{+}125°C$
存储温度…… $^{-}40 – ^{+}125°C$
结工作温度范围(Tj)…… $^{-}40 – ^{+}150°C$
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。
4. 除非另作说明，生产测试温度为 25°C。由于高温筛选不是常规测试流程，超出上述工作环境温度时，芯片的某些性能参数将不能保证。
5. 150°C 下，未经压力测试的典型数据保持时间大于 10 年。

16.2 工作特性

参数		最小值	典型值	最大值	单位	条件
Fsys (SysClk)	2T /4T	–	–	8	MHz	-40~85°C , V_{DD} = 1.9~5.5V
		–	–	16	MHz	-40~85°C , V_{DD} = 2.7~5.5V
指令周期 ($T_{INSTRCLK}$)	2T	–	125	–	ns	SysClk = HIRC
	4T	–	250	–	ns	
	2T	–	61	–	μs	SysClk = 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

16.3 POR, LVR, LVD

上电复位 (POR)

参数	最小值	典型值	最大值	单位	条件
I_{POR} 工作电流	–	0.14	–	μA	25°C, V_{DD} = 3.3V
V_{POR}	–	1.65	–	V	25°C

低电压复位 (LVR)

参数	最小值	典型值	最大值	单位	条件
I_{LVR} 工作电流	–	15.9	–	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	94	–	125	μs	25°C, V_{DD} = 1.9 – 5.5V

低电压检测 (LVD)

参数	最小值	典型值	最大值	单位	条件
I_{LVD} 工作电流	–	22.7	–	μA	25°C, V_{DD} = 3.3V
V_{LVD}，LVD 阈值	1.94	2.0	2.06	V	25°C
	2.33	2.4	2.47		
	2.72	2.8	2.88		
	2.91	3.0	3.09		
	3.49	3.6	3.71		
	3.88	4.0	4.12		
LVD delay	94	–	125	μs	25°C, V_{DD} = 1.9 – 5.5V

16.4 I/O 端口电路

参数		最小值	典型值	最大值	单位	条件
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	—	-4	—	mA	25°C, V_{DD} = 5V, V_{OH} = 4.5V
	L1	—	-8	—		
	L2	—	-26	—		
灌电流(sink)	L0	—	53	—	mA	25°C, V_{DD} = 5V, V_{OL}= 0.5V
	L1	—	62	—		
上拉电阻		—	20	—	kΩ	
下拉电阻		—	20	—	kΩ	

16.5 工作电流（I_{DD}）

参数	Sysclk	典型值@V_{DD}			单位
		2.0V	3.0V	5.5V	
正常模式(2T)，I_{DD}	16MHz	—	2.181	2.286	mA
	8MHz	1.078	1.522	1.592	
	4MHz	0.826	1.178	1.214	
	2MHz	0.624	0.716	0.719	
	1MHz	0.400	0.460	0.461	
	500kHz	0.294	0.361	0.369	
	250kHz	0.244	0.317	0.327	
	125kHz	0.219	0.295	0.306	
	32kHz	0.034	0.046	0.048	
Sleep 模式（~~WDT OFF~~, ~~LVR OFF~~），I_{SB}	—	0.211	0.274	0.423	μA
Sleep 模式（WDT ON, ~~LVR OFF~~）	—	1.383	2.499	3.062	
Sleep 模式（~~WDT OFF~~, LVR ON）	—	11.618	15.951	21.956	
Sleep 模式（WDT ON, LVR ON）	—	12.793	18.156	24.469	
Sleep 模式（~~WDT OFF~~, ~~LVR OFF~~, LVD ON）	—	18.621	22.774	28.738	

注： Sleep 模式 I_{SB} 的测试条件为所有 I/O 设置成输入模式并外部下拉到 GND；

16.6 内部振荡器

内部低频振荡器 (LIRC)

测试条件为 LIRC 选择 32 kHz (LFMOD = 0)。

参数	最小值	典型值	最大值	单位	条件
频率范围	30.4	32	33.6	kHz	25°C, V_{DD} = 2.5V
随温度变化范围	-3.0%	–	2.0%	–	-40 ~ 85°C, V_{DD} = 2.5V
随电源电压变化范围	-4.5%	–	2.0%	–	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

内部高频振荡器(HIRC)

参数	最小值	典型值	最大值	单位	条件
频率范围	15.84	16	16.16	MHz	25°C, V_{DD} = 2.5V
随温度变化范围	-2.0%	–	2.0%	–	-40 ~ 85°C, V_{DD} = 2.5V
随电源电压变化范围	-0.5%	–	0.5%	–	25°C, V_{DD} = 1.9 ~ 5.5V
I_{HIRC} 工作电流	–	40	–	μA	25°C, V_{DD} = 3.0V
启动时间	–	2.5	–	μs	25°C, V_{DD} = 3.0V

16.7 ADC (12bit) 和 ADC VREF

ADC (12bit)

参数	最小值	典型值	最大值	单位	条件
ADC 工作电压 V_{DD}	2.7	–	5.5	V	
ADC 工作电流 I_{VDD}	–	100	–	μA	V_{REF}+ = V_{DD} = 2.7V
	–	110	–	μA	V_{REF}+ = V_{DD} = 3.0V
	–	140	–	μA	V_{REF}+ = V_{DD} = 5.5V
模拟输入电压 V_{AIN}	V_{REF}−	–	V_{REF}+	V	
外部参考电压 V_{REF}	–	–	V_{DD}	V	
分辨率	–	–	12	位	
积分误差 E_{IL}	–	±1.5	–	LSB	V_{REF}+ = V_{DD} = 5.0V, V_{REF}− = GND, F_{ADCLK} = 250kHz
微分误差 E_{DL}	–	±1.0	–	LSB	
偏移误差 E_{OFF}	–	±3.0	–	LSB	V_{REF}+ = V_{DD} = 5.0V, V_{REF}− = GND, F_{ADCLK} = 250kHz
增益误差 E_{GN}	–	±5.0	–	LSB	
转换时钟周期 T_{AD}	–	2	–	μs	V_{REFP} > 3.0V, V_{DD} > 3.0V
转换时钟数	–	17	–	T_{AD}	
稳定时间 (T_{ST})	–	15	–	μs	
采样时间 (T_{ACQ})	–	≥ 2	–	μs	
模拟电压源阻抗(ZAI)	–	–	10	kΩ	(推荐)

微分误差 DNL

typical DNL Error (LSB) @ V_{DD} = 5 V				
F_{ADCLK} \ V_{REF+}	0.5	2	3	V_{DD}
≤ 500 kHz	±3.0	±1.5	±1.0	±1.0
1 MHz	±3.0	±1.5	±1.0	±1.0
2 MHz	±6.0	±2.0	±1.5	±1.5
4 MHz	–	–	–	±4.0

积分误差 INL

typical INL Error (LSB) @ V_{DD} = 5 V				
F_{ADCLK} \ V_{REF+}	0.5	2	3	V_{DD}
≤ 500 kHz	±3.0	±1.5	±1.5	±1.5
1 MHz	±3.5	±2.0	±2.0	±2.0
2 MHz	±7.0	±2.5	±2.5	±2.5
4 MHz	–	–	–	±4.5

ADC V_{REF}

参数		最小值	典型值	最大值	单位	条件
内部参考电压 $V_{ADC-REF}$	$V_{ADC-REF}$ = 0.5V	0.492	0.5	0.508	V	
	$V_{ADC-REF}$ = 2.0V	1.990	2	2.010	V	
	$V_{ADC-REF}$ = 3.0V	2.985	3	3.015	V	
稳定时间 T_{VRINT}	$V_{ADC-REF}$ = 0.5V	–	400	–	μs	
		–	600	–	μs	C_{EXT} = 1μF
	$V_{ADC-REF}$ = 2.0V	–	450	–	μs	
		–	800	–	μs	C_{EXT} = 1μF
	$V_{ADC-REF}$ = 3.0V	–	450	–	μs	
		–	1200	–	μs	C_{EXT} = 1μF

注：

1. 除非另作说明，典型值的测试条件为 25°C, V_{DD} = 5.0V。
2. C_{EXT} 为内部参考电压 $V_{ADC-REF}$ 所接的外部电容 (当 ADPREF 或 ADNREF 配置成 10 时，参阅**表 11-3**)。

16.8 Program 和 Data EEPROM

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	2.7	–	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	–	4.0	–	ms	使能自动擦除
		–	2.0	–		关闭自动擦除
I_{PROG}	Data EE 编程电流	–	350	–	μA	25 °C, V_{DD} = 3 V, 16MHz / 2T

16.9 EMC 特性

ESD

参数		最小值	典型值	最大值	单位	条件
V_{ESD}	HBM	7000	–	–	V	MIL-STD-883H Method 3015.8
V_{ESD}	MM	400	–	–	V	JESD22-A115

Latch-up

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

EFT

参数	最小值	典型值	最大值	单位	条件
V_{EFT}	5.5	–	–	kV	V_{DD} (5V) 与 GND 间的电容：1μF

17 特性图

注： 特性图基于特性值，仅供参考，未经生产测试。

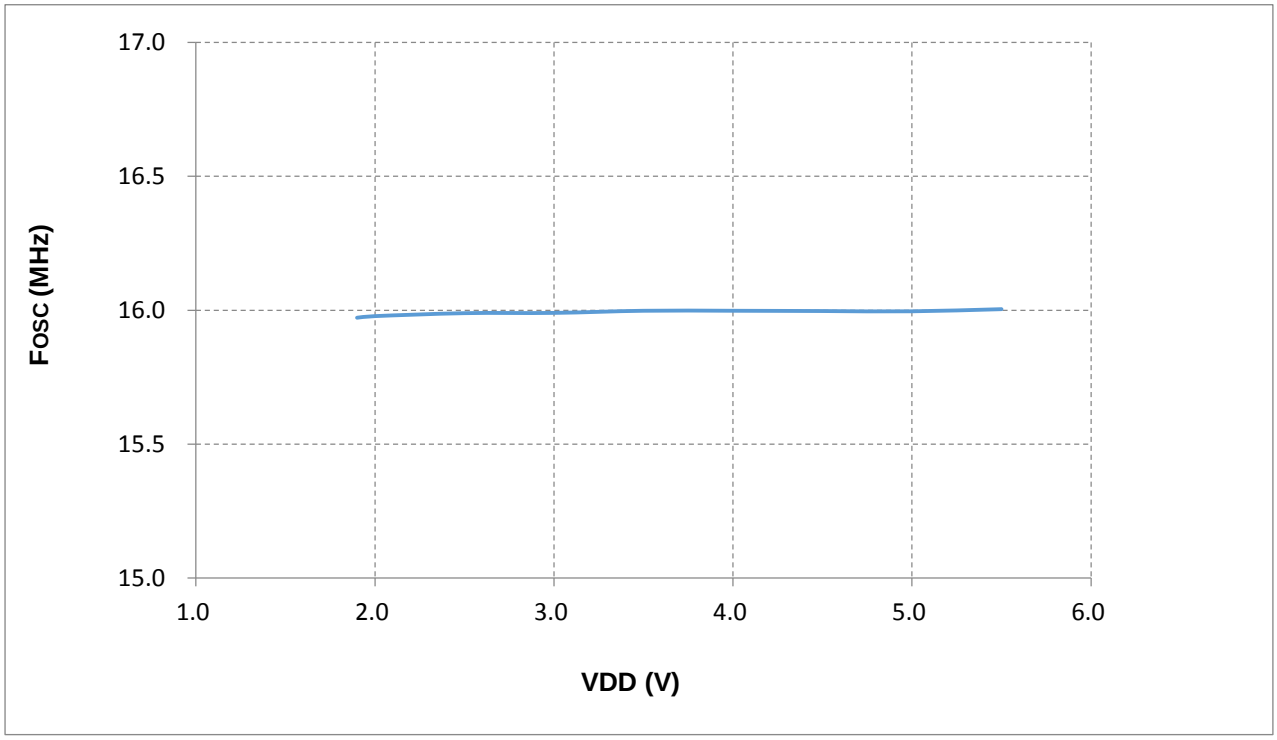


图 17-1 HIRC vs.V_{DD} (T_A = 25°C)

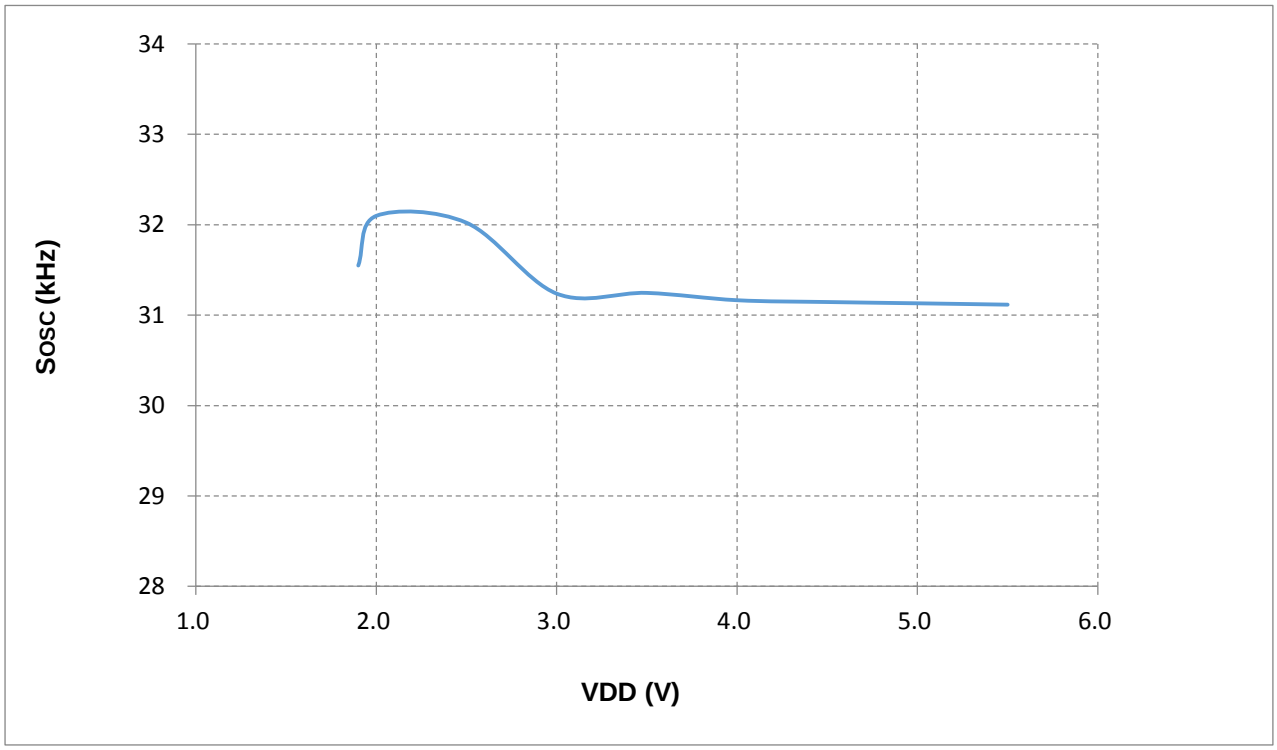


图 17-2 LIRC vs. V_{DD} (T_A = 25°C)

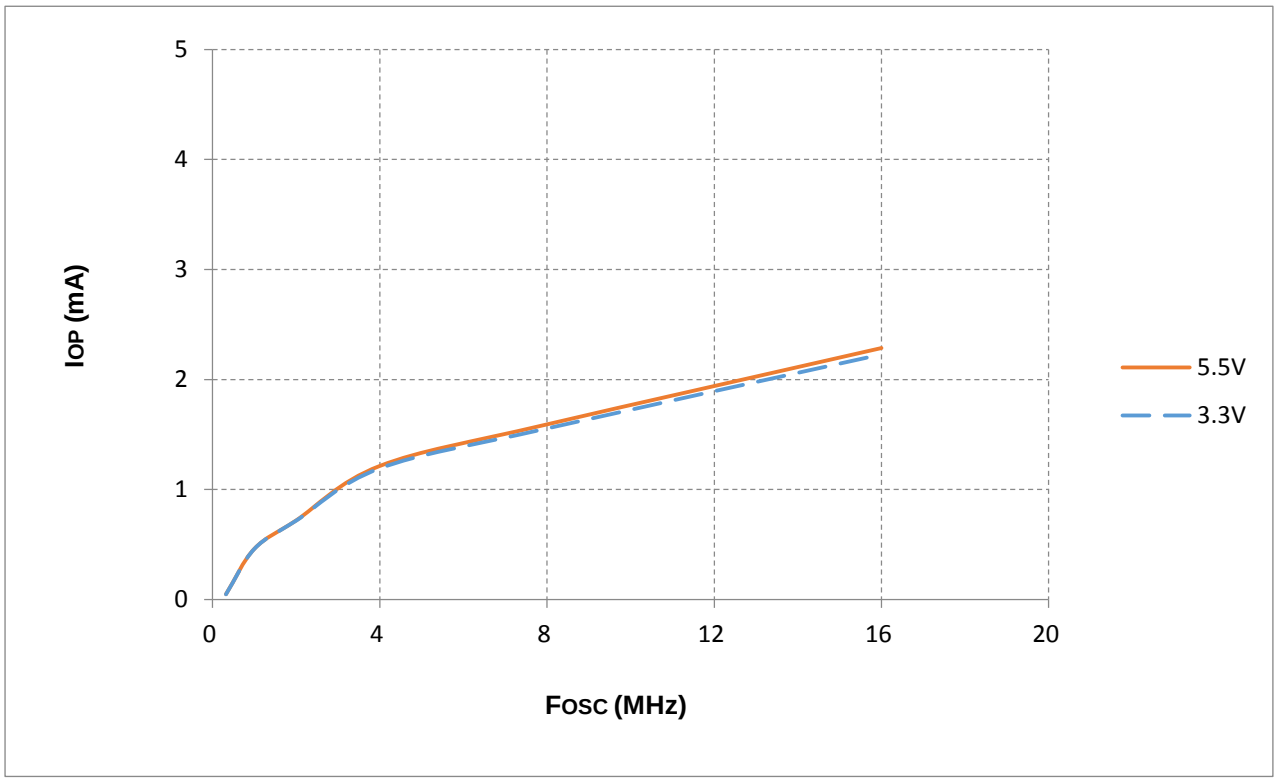


图 17-3 I_{DD} vs Freq (2T, T_A=25°C)

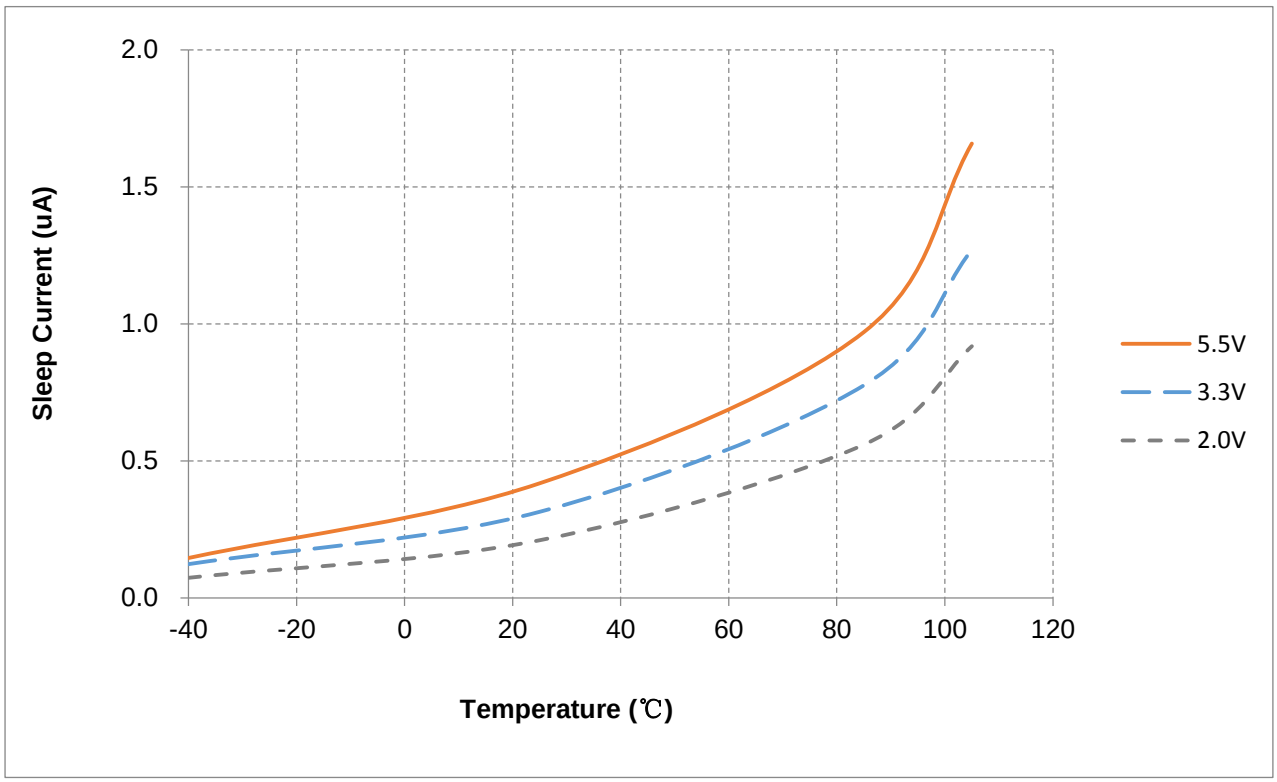


图 17-4 Sleep Current (I_{SB}) vs. Temperature

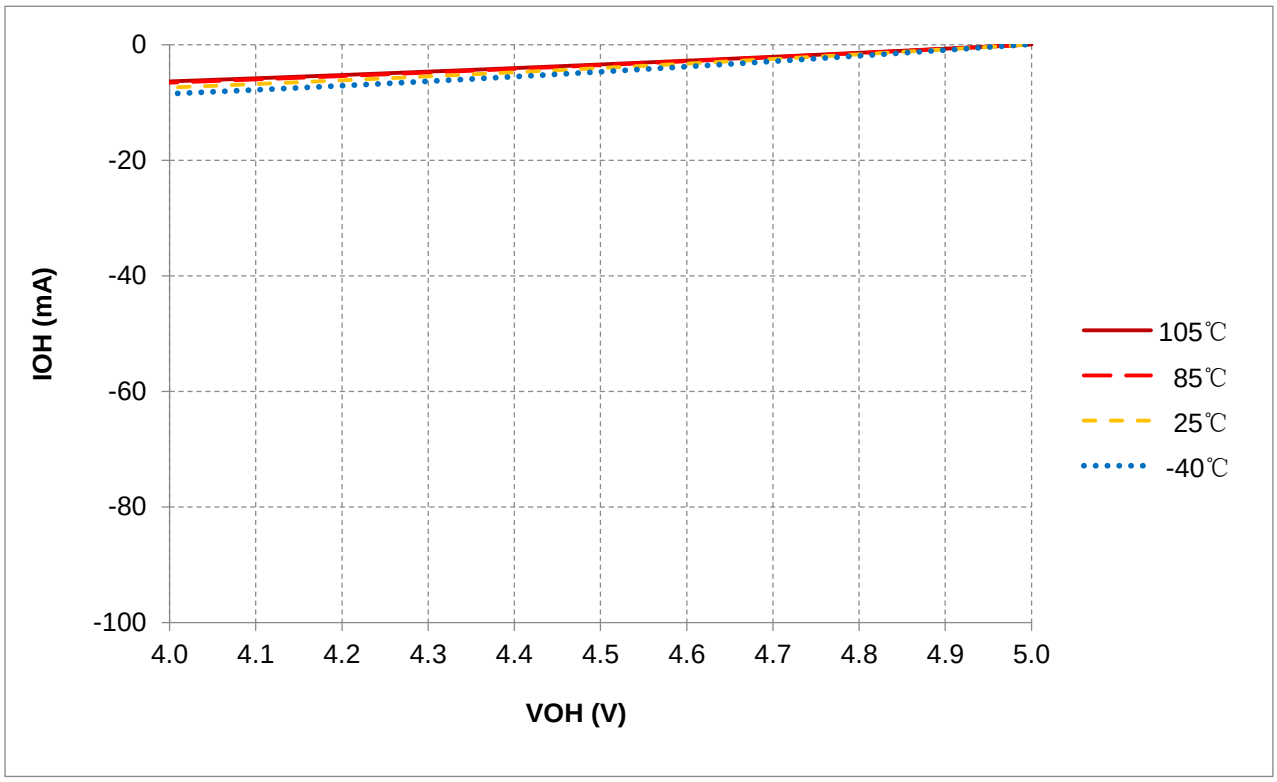


图 17-5 I_{OH} vs V_{OH} @L0 = -4mA , V_{DD} = 5V

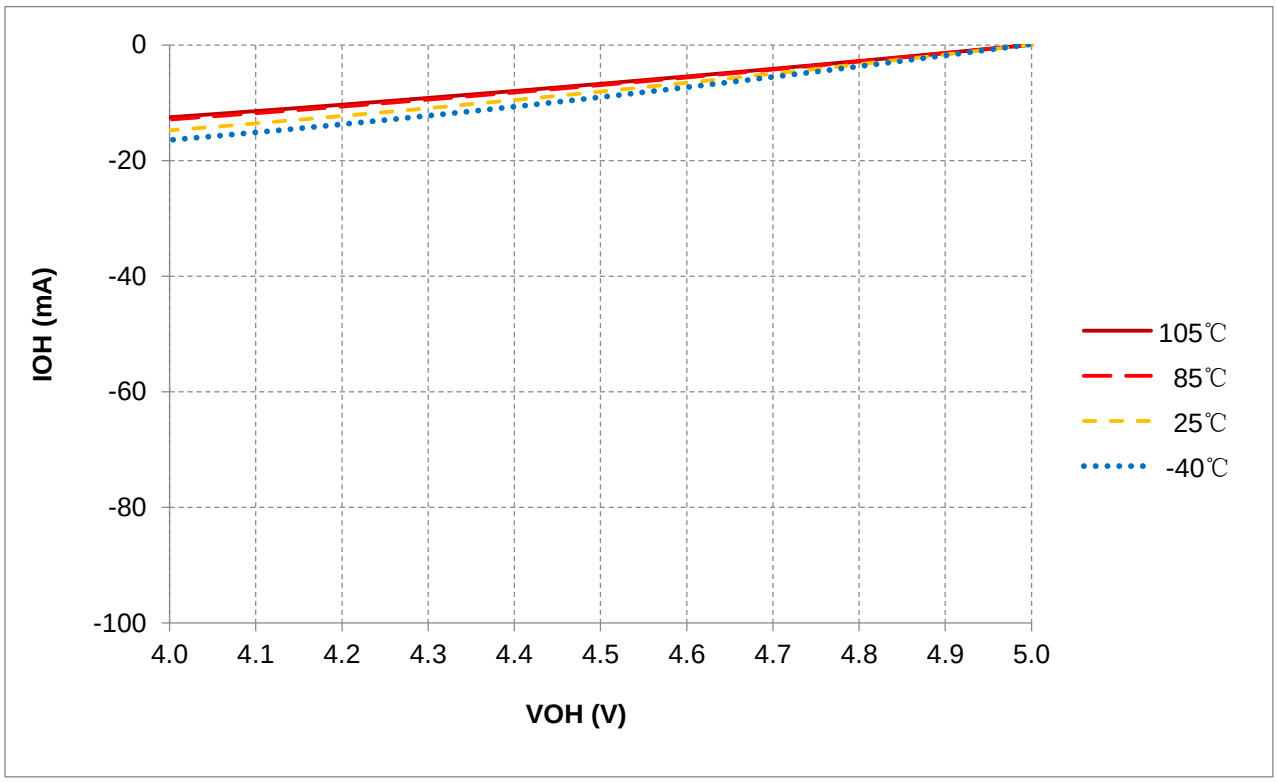


图 17-6 I_{OH} vs V_{OH} @L1 = -8mA , V_{DD} = 5V

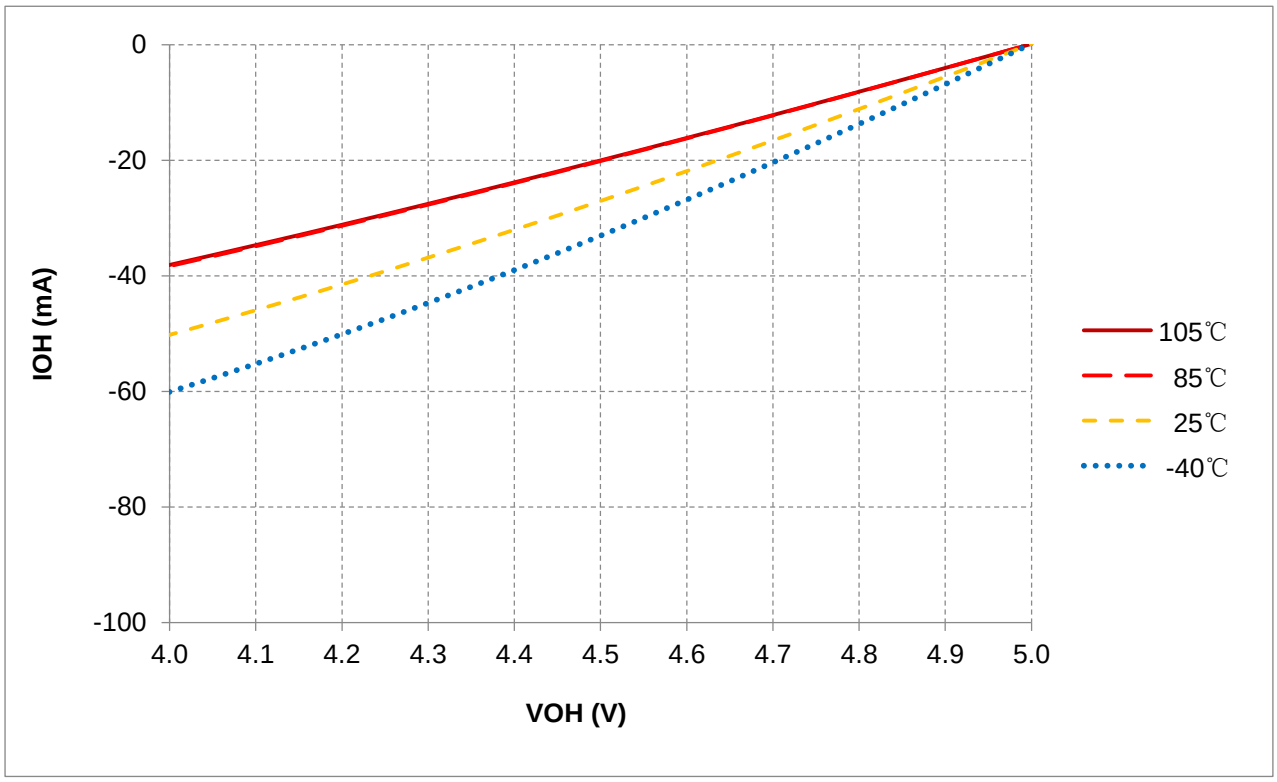


图 17-7 I_{OH} vs V_{OH} @L2 = -26mA , V_{DD} = 5V

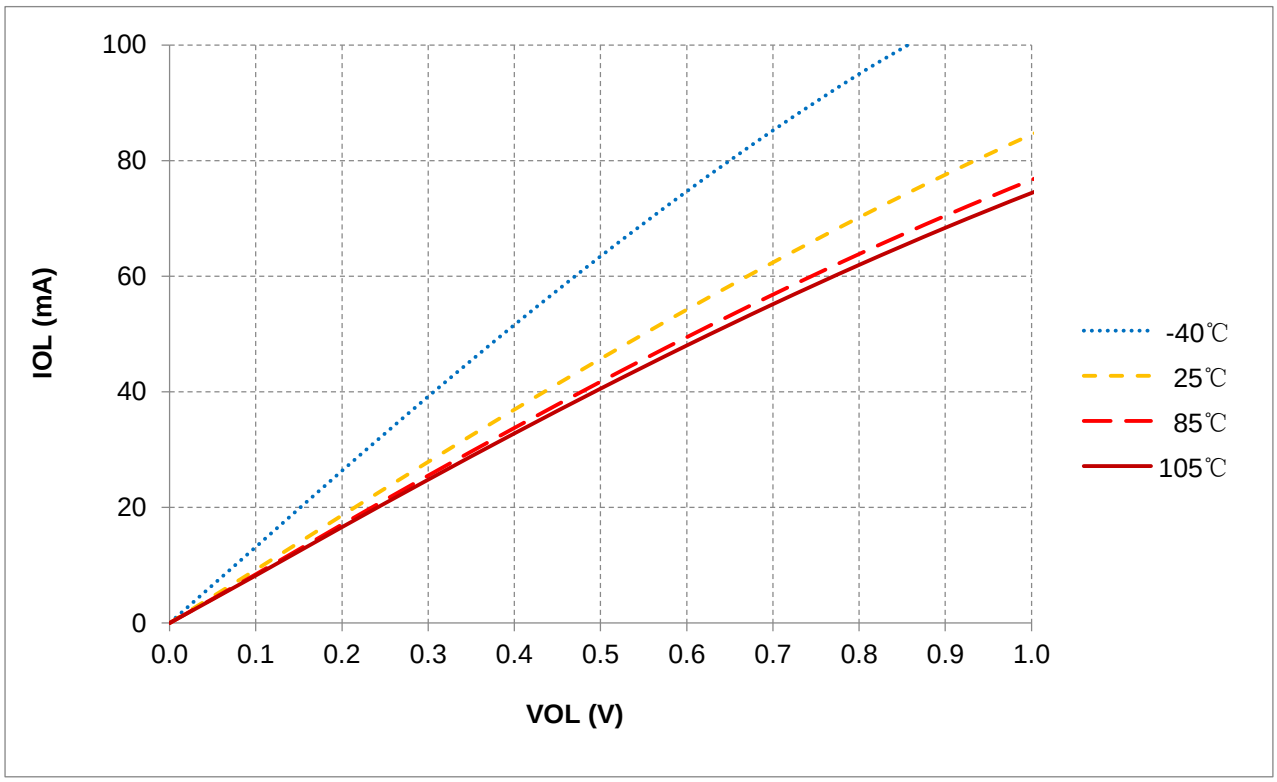


图 17-8 I_{OL} vs V_{OL} @L0 = 53mA , V_{DD} = 5V

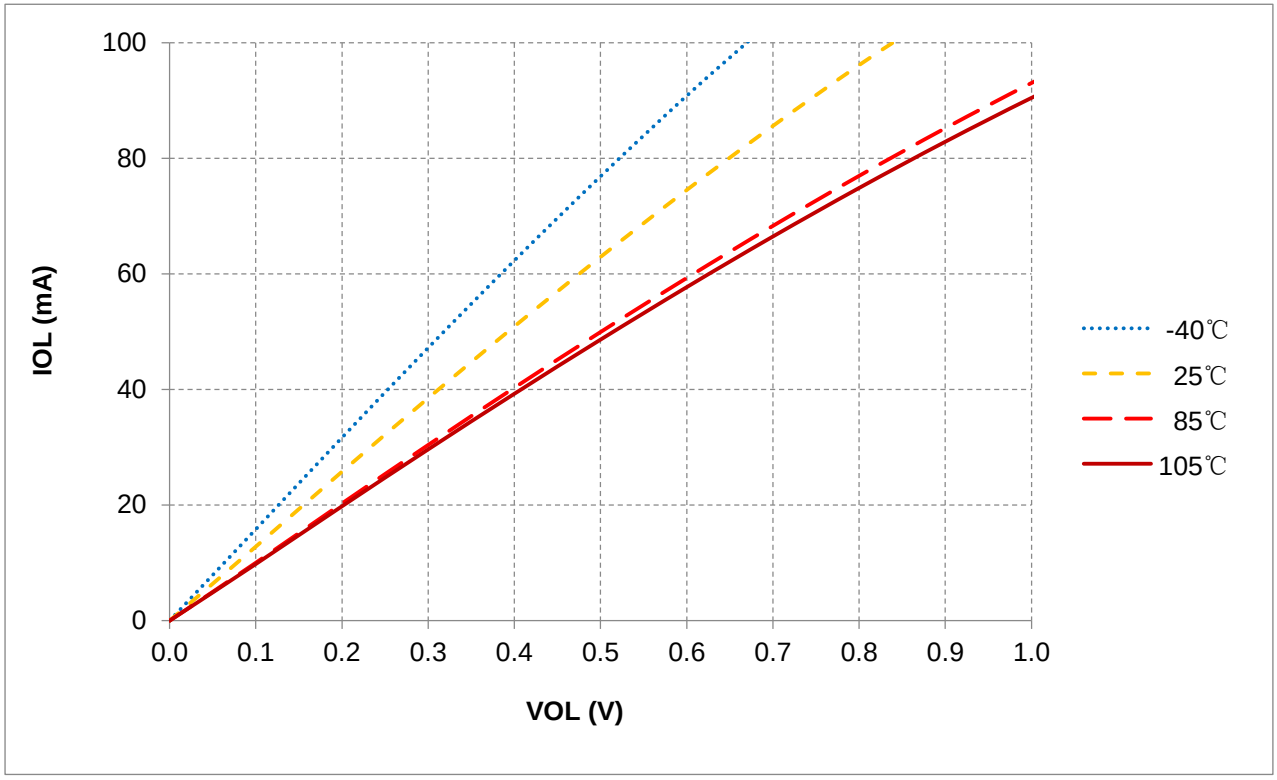
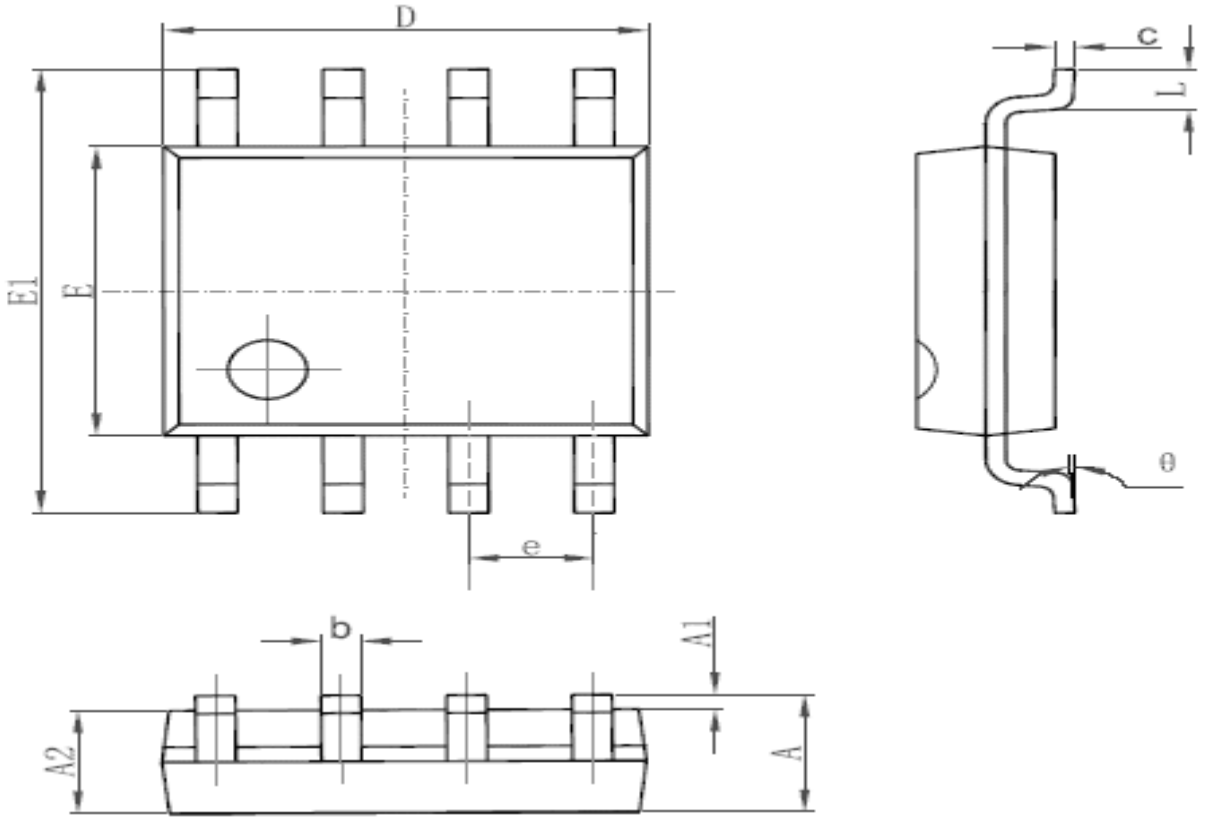


图 17-9 I_{OL} vs V_{OL} @L01 = 62mA , V_{DD} = 5V

18 封装信息

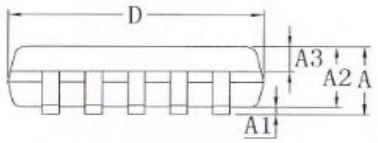
本芯片的封装形式有 SOP8、MSOP10、SOP14、SOP16、SOP20、QFN20 和 TSSOP20 封装。具体封装尺寸信息如下：

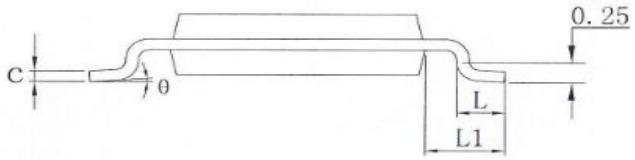
SOP8

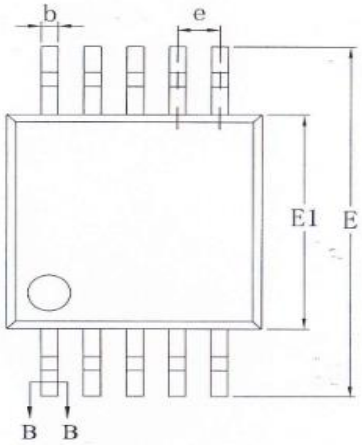


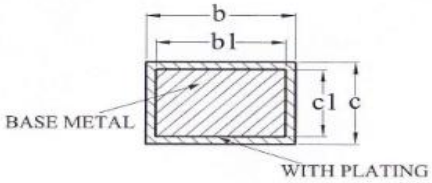
Symbol	Dimensions (mm)		Dimensions (inches)	
	Min	Max	Min	Max
A	1.350	1.750	0.053	0.069
A1	0.100	0.250	0.004	0.010
A2	1.350	1.550	0.053	0.061
b	0.330	0.510	0.013	0.020
c	0.170	0.250	0.006	0.010
D	4.700	5.100	0.185	0.200
E	3.800	4.000	0.150	0.157
E1	5.800	6.200	0.228	0.244
e	1.270 (BSC)		0.050 (BSC)	
L	0.400	1.270	0.016	0.050
θ	0°	8°	0°	8°

MSOP10





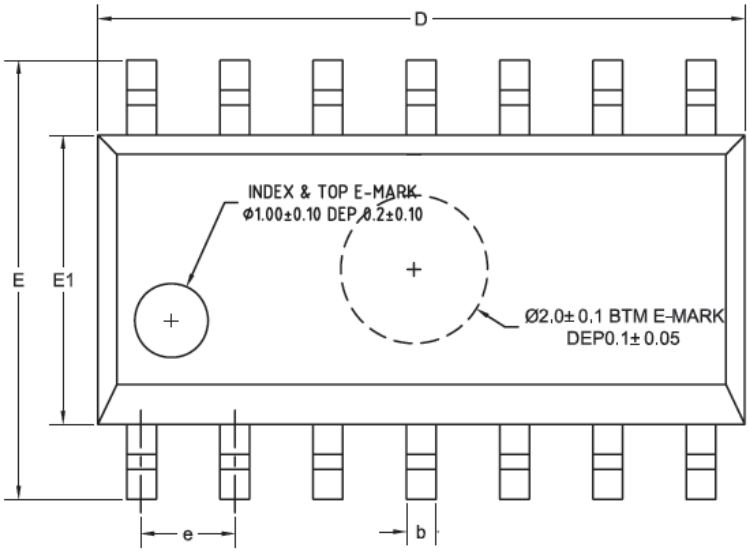


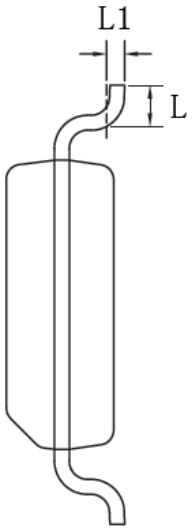


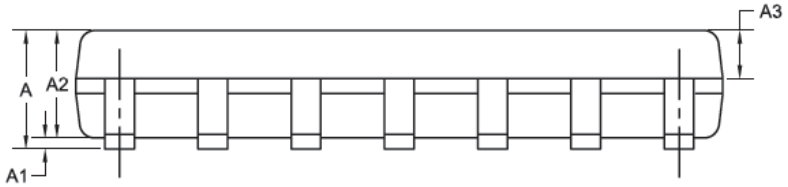
SECTION B-B

Symbol	Dimensions (mm)		Dimensions (inches)	
	Min	Max	Min	Max
A	-	1.100	-	0.043
A1	0.050	0.150	0.002	0.006
A2	0.750	0.950	0.030	0.037
A3	0.300	0.400	0.012	0.016
b	0.180	0.260	0.007	0.010
b1	0.170	0.230	0.007	0.009
c	0.150	0.190	0.006	0.007
c1	0.140	0.160	0.006	0.006
D	2.900	3.100	0.114	0.122
E	4.700	5.100	0.185	0.201
E1	2.900	3.100	0.114	0.122
e	0.500(BSC)		0.020(BSC)	
L	0.400	0.700	0.016	0.028
L1	0.950(REF)		0.037(REF)	
θ	0	8°	0	8°

SOP14

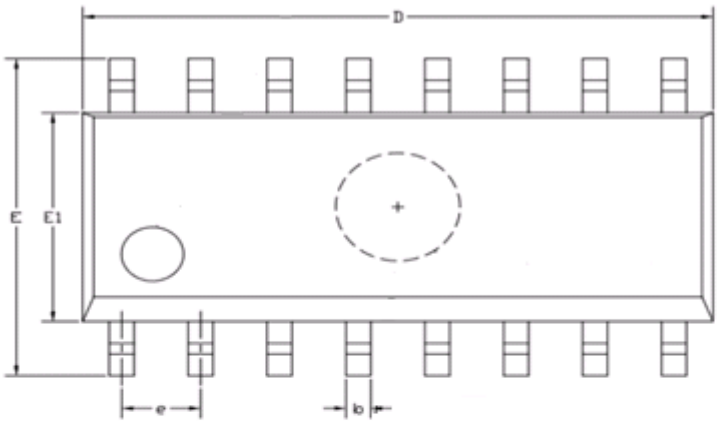


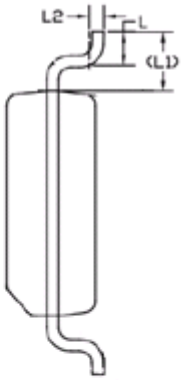


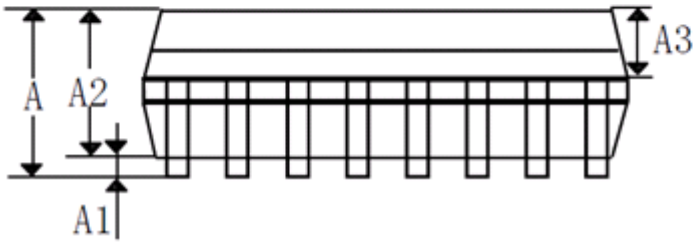


Symbol	Dimensions (mm)		Dimensions (inches)	
	Min	Max	Min	Max
A	-	1.700	-	0.066
A1	0.100	0.200	0.004	0.008
A2	1.400	1.500	0.055	0.059
A3	0.620	0.680	0.024	0.027
b	0.370	0.420	0.015	0.016
D	8.710	8.910	0.343	0.347
E	5.900	6.100	0.232	0.238
E1	3.800	3.950	0.150	0.156
e	1.270(BSC)		0.050(BSC)	
L	0.500	0.700	0.020	0.027
L1	0.250(BSC)		0.010(BSC)	

SOP16

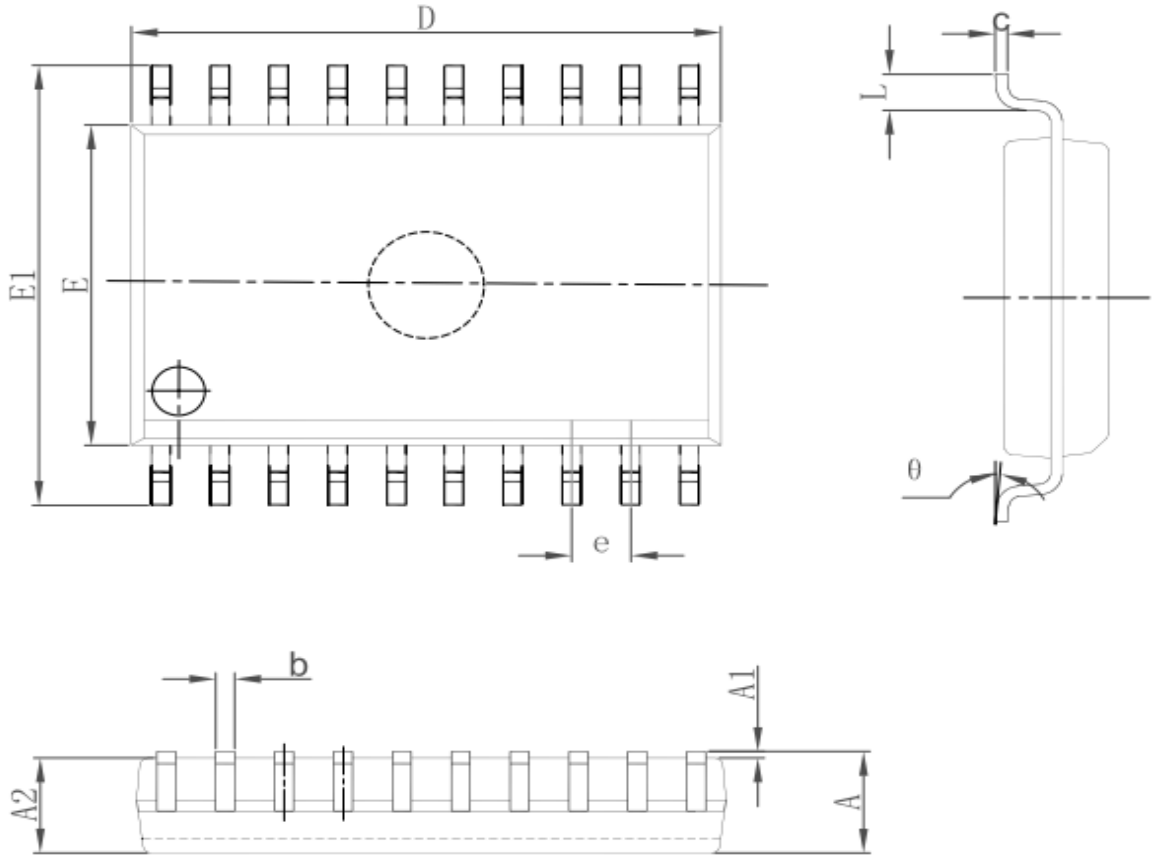






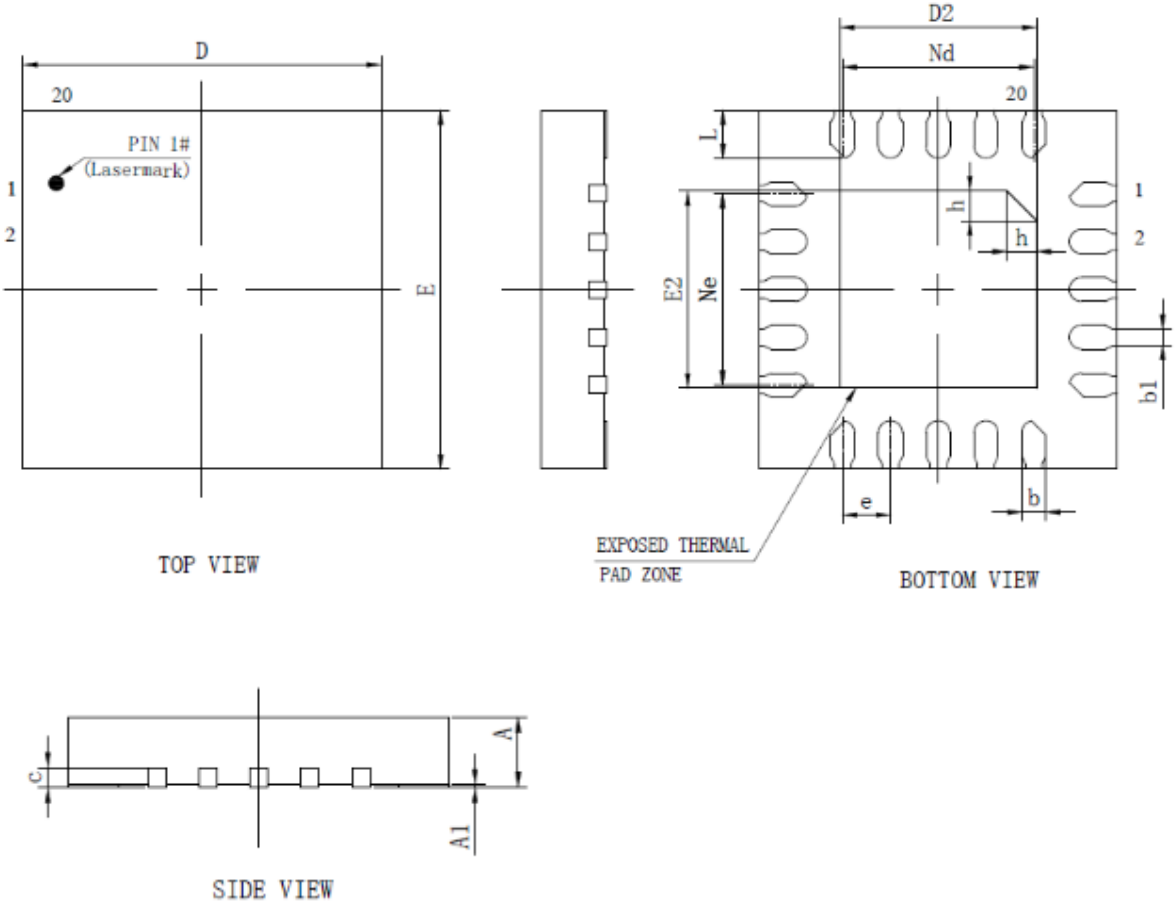
Symbol	Dimensions (mm)		Dimensions (inches)	
	Min	Max	Min	Max
A	-	1.750	-	0.069
A1	0.100	0.250	0.004	0.010
A2	1.250	1.550	0.049	0.061
A3	0.550	0.750	0.022	0.030
D	9.800	10.160	0.386	0.400
E	5.800	6.200	0.228	0.244
E1	3.800	4.000	0.150	0.157
b	0.310	0.510	0.012	0.020
e	1.270(BSC)		0.050(BSC)	
L	0.400	1.270	0.016	0.050
L1	1.04(REF)		0.04(REF)	
L2	0.25(BSC)		0.01(BSC)	

SOP20



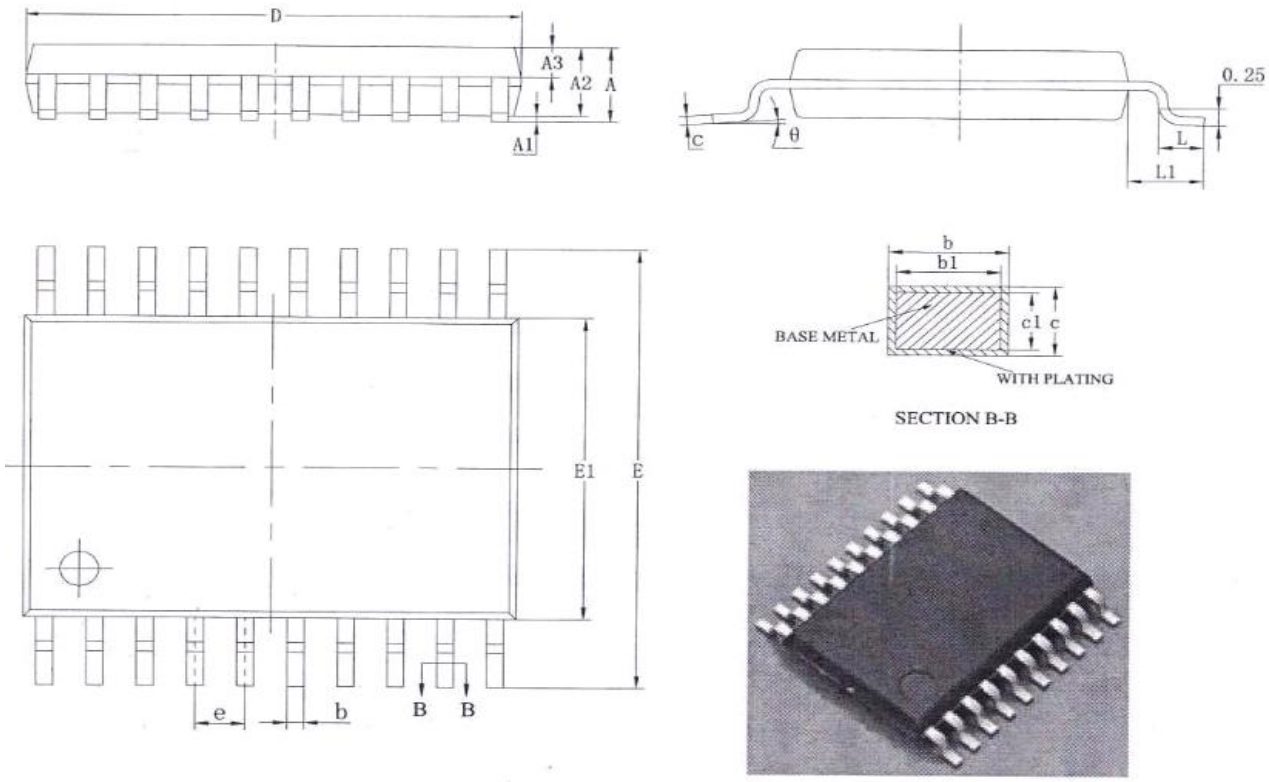
Symbol	Dimensions (mm)		Dimensions (inches)	
	Min	Max	Min	Max
A	2.350	2.650	0.093	0.104
A1	0.100	0.300	0.004	0.012
A2	2.100	2.500	0.083	0.098
b	0.330	0.510	0.013	0.020
c	0.204	0.330	0.008	0.013
D	12.520	13.000	0.493	0.512
E	7.400	7.600	0.291	0.299
E1	10.210	10.610	0.402	0.418
e	1.270 (BSC)		0.050 (BSC)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

QFN20



Symbol	Dimensions (mm)		Dimensions (inches)	
	Min	Max	Min	Max
A	0.500	0.600	0.020	0.024
A1	-	0.050	-	0.002
b	0.150	0.250	0.006	0.010
b1	0.140 (REF)		0.006 (REF)	
c	0.100	0.200	0.004	0.008
D	2.900	3.100	0.114	0.122
D2	1.550	1.750	0.061	0.069
e	0.400 (BSC)		0.016 (BSC)	
Ne	1.600 (BSC)		0.063 (BSC)	
Nd	1.600 (BSC)		0.063 (BSC)	
E	2.900	3.100	0.114	0.122
E2	1.550	1.750	0.061	0.069
L	0.350	0.450	0.014	0.018
h	0.200	0.300	0.008	0.012

TSSOP20



Symbol	Dimensions (mm)		Dimensions (inches)	
	Min	Max	Min	Max
A	-	1.20	-	0.047
A1	0.05	0.15	0.002	0.006
A2	0.80	1.05	0.031	0.041
A3	0.39	0.49	0.015	0.019
b	0.20	0.28	0.008	0.011
b1	0.19	0.25	0.007	0.010
c	0.13	0.17	0.005	0.007
c1	0.12	0.14	0.005	0.006
D	6.40	6.60	0.252	0.260
E1	4.30	4.50	0.169	0.177
E	6.20	6.60	0.244	0.259
e	0.65(BSC)		0.026(BSC)	
L	0.45	0.75	0.018	0.030
L1	1.00REF		0.039REF	
θ	0	8°	0	8°

19 附录：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

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.