

8 位微控制器

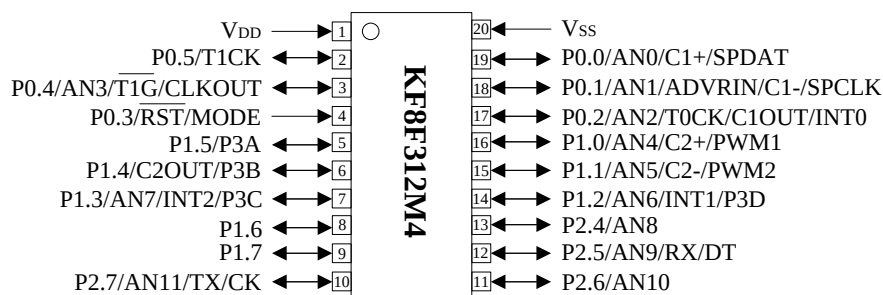
KF8F312M4 数据手册

产品订购信息

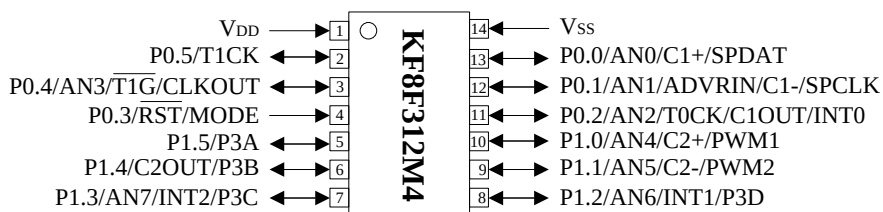
型号	订货号	FLASH	RAM	BLOCK EEPROM	最高内部 晶振	LDO	封装
KF8F312M4	KF8F312M4-SS20	4K×16 位	256×8 位	128X16 位	4MHz	无	SSOP20
	KF8F312M4-S20	4K×16 位	256×8 位	128X16 位	4MHz	无	SOIC20
	KF8F312M4-S14	4K×16 位	256×8 位	128X16 位	4MHz	无	SOIC14
	KF8F312M4-S8	4K×16 位	256×8 位	128X16 位	4MHz	无	SOIC8

引脚示意图

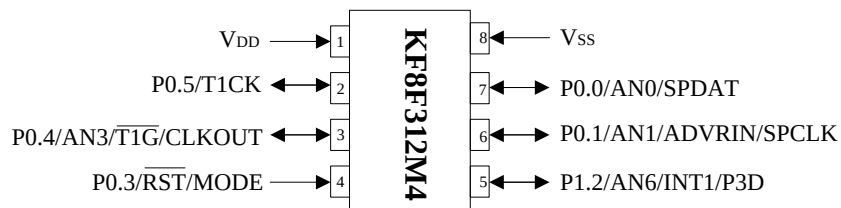
20 引脚 KF8F312M4 示意图:



14 引脚示意图:



8 引脚示意图:



芯片引脚说明

引脚名	I/O	引脚功能	引脚说明
1	P	VDD	电源
2	I/O	P0.5	带上拉和电平变化中断功能的双向输入输出端口
		T1CK	T1 时钟输入
3	I/O	P0.4	带上拉和电平变化中断功能的双向输入输出端口
		AN3	ADC 输入通道 3
		$\overline{\text{T1G}}$	T1 门控信号输入
		CLKOUT	系统时钟输出
4	I	P0.3	带电平变化中断的输入端口
		$\overline{\text{RST}}$	外部复位信号输入
		MODE	编程模式选择
5	I/O	P1.5	双向输入输出端口
		P3A	PWM3 输出
6	I/O	P1.4	双向输入输出端口
		C2OUT	模拟比较器 2 输出
		P3B	PWM3 输出
7	I/O	P1.3	双向输入输出端口
		AN7	ADC 输入通道 7
		INT2	外部中断 2 输入
		P3C	PWM3 输出
8	I/O	P1.6	双向输入输出端口
9	I/O	P1.7	双向输入输出端口
10	I/O	P2.7	双向输入输出端口
		AN11	ADC 输入通道 11
		TX	USART 全双工异步发送
		CK	USART 半双工同步时钟
11	I/O	P2.6	双向输入输出端口
		AN10	ADC 输入通道 10
12	I/O	P2.5	双向输入输出端口
		AN9	ADC 输入通道 9
		RX	USART 全双工异步接收
		DT	USART 半双工同步数据
13	I/O	P2.4	双向输入输出端口
		AN8	ADC 输入通道 8
14	I/O	P1.2	双向输入输出端口
		AN6	ADC 输入通道 6
		INT1	外部中断 1 输入
		P3D	PWM3 输出
15	I/O	P1.1	双向输入输出端口
		AN5	ADC 输入通道 5
		C2-	模拟比较器负端输入
		PWM2	PWM2 输出

16	I/O	P1.0	双向输入输出端口
		AN4	ADC 输入通道 4
		C2+	模拟比较器正端输入
		PWM1	PWM1 输出
17	I/O	P0.2	带上拉和电平变化中断功能的双向输入输出端口
		AN2	ADC 输入通道 2
		T0CK	T0 时钟输入
		C1OUT	模拟比较器 1 输出
		INT0	外部中断 0 输入
18	I/O	P0.1	带上拉和电平变化中断功能的双向输入输出端口
		AN1	ADC 输入通道 1
		ADVRIN	AD 外部参考电压输入
		C1-	模拟比较器 1 负端输入
		SPCLK	在线编程时钟输入
19	I/O	P0.0	带上拉和电平变化中断功能的双向输入输出端口
		AN0	ADC 输入通道 0
		C1+	模拟比较器 1 正端输入
		SPDAT	在线编程数据输入/输出
20	P	Vss	地，0V 参考点

目 录

产品订购信息	2
引脚示意图	3
芯片引脚说明	4
目 录	6
1 系统概述	12
1.1 芯片特征	13
1.2 系统框图	15
1.3 存储器	16
1.4 系统时钟	17
1.5 配置位	19
1.6 在线串行编程	20
2 I/O 端口介绍	21
2.1 P0 口	22
2.1.1 P0 口相关的寄存器	23
2.1.1.1 P0 口状态寄存器	23
2.1.1.2 P0 口方向控制寄存器	23
2.1.1.3 上拉功能控制寄存器	24
2.1.1.4 电平变化中断控制寄存器	24
2.1.2 P0 口各引脚内部原理功能框图	26
2.2 P1 口	28
2.2.1 P1 口相关的寄存器	29
2.2.1.1 P1 口状态寄存器	29
2.2.1.2 P1 口方向控制寄存器	29
2.2.2 P1 口原理功能框图	30
2.3 P2 口	31
2.3.1 P2 口相关的寄存器	31
2.3.1.1 P2 口状态寄存器(P2)	31
2.3.1.2 P2 口方向控制寄存器(TR2)	32
2.3.1.3 模拟/数字口配置寄存器(ANSEH)	32
2.3.2 P2 口原理功能框图	33
3 存储器	34
3.1 程序存储器(ROM)区	35
3.2 数据存储器(RAM)区	36
3.2.1 通用寄存器区	36

3.2.2	特殊功能寄存器(SFR)区	37
3.3	BLOCK EEPROM	38
3.3.1	寄存器 R6/R7	39
3.3.2	寄存器 BADDRH/BADDRL	39
3.3.3	寄存器 EECTL1/EECTL2	39
3.3.4	写 BLOCK EEPROM	40
3.3.5	读 BLOCK EEPROM	42
3.4	寄存器组 RN	42
3.5	ID 地址单元	42
4	汇编指令及寻址方式	43
4.1	寻址方式	43
4.1.1	寄存器寻址	43
4.1.2	直接寻址	43
4.1.3	立即数寻址	43
4.1.4	寄存器间接寻址	44
4.1.5	位寻址	44
4.2	汇编指令	44
5	中断	45
5.1	中断相关的寄存器	46
5.1.1	中断控制寄存器(INTCTL)	46
5.1.2	中断使能寄存器 EIE1	47
5.1.3	中断标志寄存器 EIF1	48
5.1.4	中断使能寄存器 EIE2	49
5.1.5	中断标志寄存器 EIF2	49
5.1.6	中断响应	49
5.2	INT 中断	51
5.2.1	INT0 中断	51
5.2.2	INT1 中断	51
5.2.3	INT2 中断	51
5.3	定时器中断	52
5.4	P0 口中断	52
5.5	PWM 中断	52
5.6	模拟比较器中断	52
5.7	USART 中断	52
5.8	中断现场保护	52
6	定时/计数器	53
6.1	定时/计数器 0(T0)	53
6.1.1	OPTR 选择寄存器	53
6.1.2	预分频器	54
6.1.3	定时模式	54
6.1.4	计数模式	54

6.1.5	T0 的使用.....	55
6.2	定时/计数器 1(T1).....	56
6.2.1	T1 控制寄存器.....	56
6.2.2	T1 预分频器.....	57
6.2.3	定时模式	57
6.2.4	计数模式	57
6.2.5	T1 在休眠模式下的运行.....	58
6.2.6	T1 分配给 PWM1/2.....	58
6.3	定时器 T2.....	59
6.3.1	T2 相关的寄存器.....	59
6.3.1.1	T2 控制寄存器 T2CTL.....	59
6.3.2	T2 的工作原理.....	60
6.3.3	T2 中断	61
6.3.4	T2 在休眠模式.....	61
6.3.5	T2 分配给 PWM3	61
7	模数(A/D)转换模块.....	62
7.1	与 AD 相关的寄存器.....	63
7.1.1	AD 控制寄存器 0(ADCCTL0)	63
7.1.2	AD 控制寄存器 1(ADCCTL1)	64
7.1.3	模拟/数字口选择寄存器(ANSEL、ANSEH)	64
7.2	通道的选择	66
7.3	模拟输入口的配置	66
7.4	转换时钟的选择	66
7.5	输出格式	66
7.6	A/D 转换的启动和完成.....	67
7.7	A/D 工作在休眠模式	67
7.8	复位的影响	67
7.9	使用 A/D 转换器的设置	67
8	PWM 模块.....	68
8.1	PWM1/2 模块.....	68
8.1.1	PWM1/2 相关的寄存器	69
8.1.2	PWM 控制寄存器	69
8.1.3	PWM1/2 周期	70
8.1.4	PWM1/2 占空比	70
8.1.5	PWM1/2 分辨率	71
8.1.6	PWM1/2 中断	71
8.1.7	休眠模式下的操作	71
8.1.8	系统时钟频率的改变	71
8.1.9	复位的影响	71
8.1.10	PWM1/2 使用方法	71
8.2	PWM3.....	73
8.2.1	PWM3 相关寄存器	74

8.2.2	PWM3CTL0 寄存器	74
8.2.3	PWM3CTL1 寄存器	75
8.2.4	P3ASCTL 寄存器	75
8.2.5	PATRCTL 寄存器	76
8.2.6	PWM3 的周期、占空比及分辨率	76
8.2.6.1	PWM3 周期.....	76
8.2.6.2	PWM3 占空比.....	76
8.2.6.3	PWM3 分辨率.....	77
8.2.6.4	PWM3 中断.....	77
8.2.7	单输出模式	78
8.2.8	半桥输出模式	80
8.2.8.1	死区延时.....	81
8.2.9	全桥输出模式	82
8.2.10	自动关断和自动重启模式	85
8.2.10.1	自动关断模式.....	85
8.2.10.2	自动重启模式.....	86
9	模拟比较器模块	87
9.1	模拟比较器原理	87
9.2	CMCTL0 寄存器	87
9.3	CMCTL1 寄存器	88
9.4	比较器中断	88
9.5	比较器的用法	88
10	通用全双工/ 半双工收发器	89
10.1	系统概述	89
10.1.1	相关寄存器	89
10.1.2	原理框图	90
10.2	波特率发生器	91
10.2.1	波特率控制寄存器 BRCTL	91
10.2.2	波特率的选择	92
10.2.3	自动波特率检测	98
10.3	USART 全双工模式.....	100
10.3.1	USART 全双工发送操作	101
10.3.1.1	发送状态和控制寄存器 TSCTL	101
10.3.1.2	发送数据	102
10.3.1.3	全双工发送的设置.....	103
10.3.2	USART 全双工接收操作	105
10.3.2.1	接收状态和控制寄存器 RSCTL	105
10.3.2.2	接收数据	106
10.3.2.3	接收错误	107
10.3.2.4	地址检测	107
10.3.2.5	全双工接收的设置.....	107
10.3.3	RS-485 发送/接收	109

10.3.3.1	RS-485 9 位地址检测模式设置	109
10.3.4	全双工操作时钟的精确性	109
10.4	USART 半双工模式	110
10.4.1	USART 半双工主控模式	110
10.4.1.1	半双工主控发送	110
10.4.1.2	半双工主控接收	112
10.4.2	USART 半双工从动模式	115
10.4.2.1	USART 半双工从动发送	115
10.4.2.2	USART 半双工从动接收	116
10.4.3	USART 半双工 RS-485 模式	116
10.4.4	USART 用作 SPI 模式	116
11	复位	118
11.1	电源控制状态寄存器(PCTL)	119
11.2	上电复位(POR)	120
11.3	WDT 复位	120
11.4	RST 复位	120
11.5	欠压检测复位(LVR)	121
11.6	上电延时定时器	121
11.7	不同复位条件下对寄存器的影响	122
12	休眠模式	124
13	看门狗定时器	125
14	电气规范	126
14.1	极限参数值	126
14.2	静态电流特性	127
14.3	外设电流特性	128
14.4	I/O 端口电平和芯片供电电压特性	129
14.5	比较器模块规范	130
14.6	A/D 转换器 (ADC) 特性	130
15	直流特性图表	131
16	封装信息	138
附录 1	KF8F312M4 SFR 地址映射及功能汇总	140
附录 2	汇编指令集	142
附录 3	寄存器全称表	144
产品标识体系		146
版本信息		147



ROHS 认证	148
声明及销售网络	149

1 系统概述

KF8F312M4 为哈佛结构的精简指令 CPU。在这种结构中，程序和数据总线是相互独立的。指令字节长度为 16 位，大多数指令能在一个机器周期内执行完成。一共有 68 条指令，效率高，容易进行指令扩展。芯片内集成了多种外设，包括 2 个 8 位定时器/计数器 T0 和 T2、1 个 16 位定时器/计数器 T1、2 路 8 位 PWM 模块、1 路 10 位增强型 PWM 模块、1 个通用全双工异步通讯模块、2 个模拟比较器模块、1 个 10 位 12/8/4 通道 AD 模块、硬件看门狗和低电压检测及低电压复位模块等。

芯片内集成了 256×8 位的数据存储器 RAM、4K×16 位的程序存储器。

1.1 芯片特征

● CPU

高性能哈佛结构的 RISC CPU
68 条精简指令
支持中断处理，共 14 个中断源
复位向量位于 0000H，中断向量位于 0004H
工作频率为 15.625kHz~4MHz，软件可选

● 存储器

4K×16 位 FLASH 程序存储器
256×8 位的数据存储器
128×16 位的 BLOCK EEPROM
工作寄存器组 R0~R7
FLASH 可经受 100 000 次写操作
BLOCK EEPROM 可经受 1 000 000 次写操作

● 特殊功能

内嵌上电复位电路
低电压检测及低电压复位
硬件看门狗
时钟精度 4MHz±1%(常温)
支持在线串行编程
低功耗休眠模式

● I/O 口配置

输入/输出：除 P0.3 只能作为输入口外其它端口均为双向口
内置弱上拉功能：P0 口带有弱上拉功能(P0.3 除外)
电平变化中断：P0 口均有电平变化中断功能

● 定时器/计数器

定时器 0：带有 8 位预分频器的 8 位定时器/计数器
定时器 1：带门控和预分频器的 16 位定时器/计数器
定时器 2：带 8 位周期寄存器、预分频器和后分频器的 8 位定时器

- 其它外设

- 2 路 8 位脉宽调制 PWM 模块

- 1 路 10 位增强型脉宽调制 PWM 模块

- 1 个 USART 模块

- 2 个模拟比较器模块

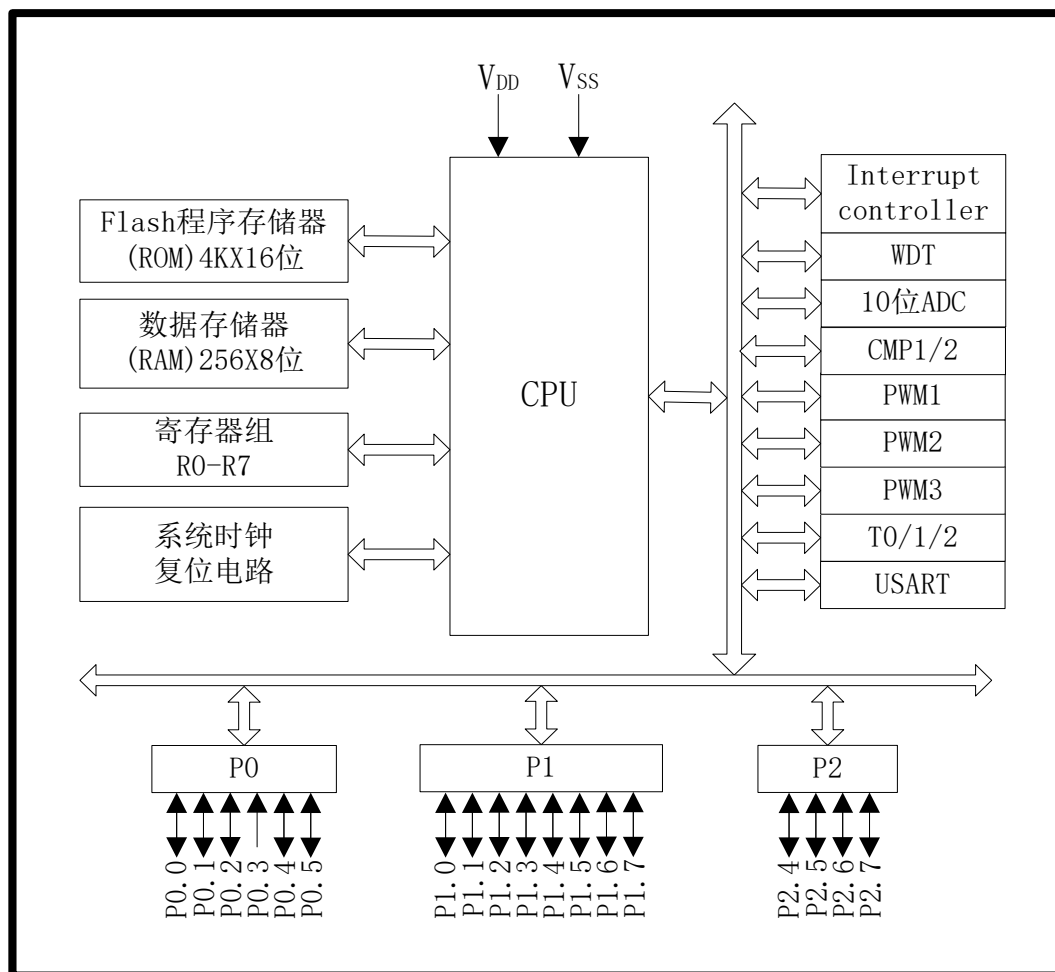
- 1 个 10 位 ADC 模块

- 工作条件

- 工作电压：2.3V~5.5V

- 工作温度范围：-40℃~85℃(工业级)

1.2 系统框图



1.3 存储器

KF8F312M4 单片机的存储器包含：程序存储器(ROM)、数据存储器(RAM)和 BLOCK EEPROM。

KF8F312M4 的程序存储器空间为 $4K \times 16$ 位，寻址范围为 $0000H \sim 0FFFH$ ，可擦写次数为 10 万次。数据存储器有三个存储区，这里称为特殊寄存器区(SFR)、通用存储器区 0 和通用存储器区 1，每个存储器区有 128×8 位的存储单元，这三个区的地址分别为 $00H \sim 7FH$ 、 $80H \sim 0FFH$ 和 $180H \sim 1FFH$ 。BLOCK EEPROM 是映射在程序存储器寻址范围 4K 内的存储单元，地址为 $0F70H \sim 0FEFH$ ，每个地址对应 16 位的数据。有关以上各种存储器的具体介绍请参考第 3 章和第 4 章。

1.4 系统时钟

振荡周期又叫时钟周期，是振荡器振荡频率的倒数。本芯片中一个机器周期等于四个时钟周期，如图 1.1 所示。本芯片除执行部分跳转指令需要两个机器周期外，其余指令仅需要一个机器周期。

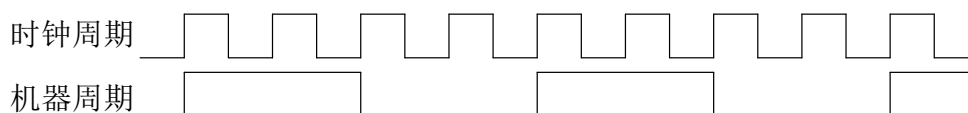


图 1.1 机器周期

KF8F312M4 单片机通过系统内部振荡器提供系统工作时钟，没有外部时钟输入和扩展口，时钟频率为 15.625kHz~4MHz，通过频率选择寄存器 OSCCTL(如寄存器 1.1 所示)选择系统工作频率。精度为±1%，时钟框图如图 1.2 所示。

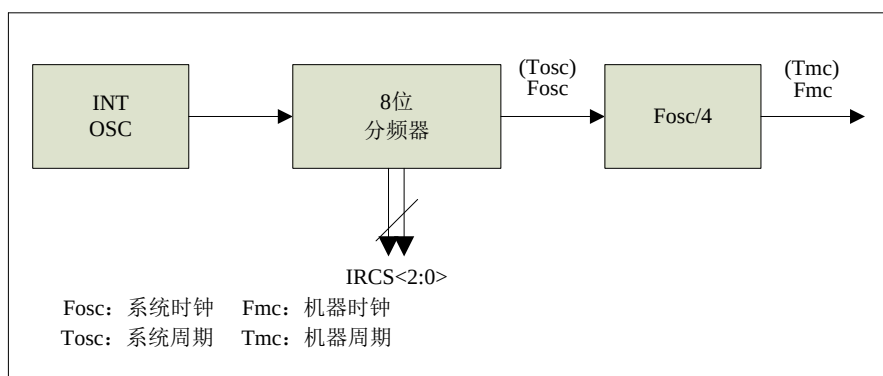


图 1.2 时钟框图

寄存器1.1: OSCCTL系统频率控制寄存器(地址:2FH)

	bit7							bit0
复位值 0011 ----	CKOEN	IRCF2	IRCF1	IRCF0	-	-	-	-
	R/W	R/W	R/W	R/W	U	U	U	U

注: R=可读 W=可写 U=未用 后面与此相同不再复述

- CKOEN: 系统时钟输出使能位
 CKOEN=1 使能系统时钟输出
 CKOEN=0 禁止系统时钟输出
- IRCF<2:0> 时钟频率选择位
 111=4MHz
 110=2MHz
 101=1MHz

100=500kHz

011=250kHz(默认)

010=125kHz

001=62.5kHz

000=15.625kHz

OSCCAL0、OSCCAL1 为内部振荡器时钟校准寄存器，用来存放系统时钟校准值。用户在编程时，需要在程序初始化部分将存放在程序空间 0FFFH 和 0FFEh 的晶振校准值读出来存放到 OSCCAL0 和 OSCCAL1，参考例子 1.1，否则会导致系统时钟频率不准。

例 1.1 读晶振校准值

```
CALL 0XFFF
```

```
MOV OSCCAL0, R0
```

```
NOP
```

```
NOP
```

```
CALL 0XFFE
```

```
MOV OSCCAL1, R0
```

```
NOP
```

```
NOP
```

1.5 配置位

如寄存器 1.2 所示，用户在烧写程序时，在编程器中通过对配置位进行设置，使单片机启用诸如看门狗、程序代码保护、欠压检测等功能。

寄存器1.2: CONFIG: 配置字

R/P	R/P	U	R/P	R/P	R/P	U	R/P	U	U	U
DEBUG	SWRTEN	-	CODEP	LVREN	RSTEN	-	WDTEN	-	-	-
bit10			bit7				bit3			bit0

注：R=可读 P=编程时可写 U=未用

- DEBUG:** 在线调试使能位
 $\overline{\text{DEBUG}}=1$ 禁止在线调试
 $\overline{\text{DEBUG}}=0$ 使能在线调试
- SWRTEN:** BLOCK EEPROM 写使能位
 $\text{SWRTEN}=1$ 使能写保护，此时 BLOCK EEPROM 不能写操作,(可读)
 $\text{SWRTEN}=0$ 禁止写保护，此时 BLOCK EEPROM 可写，(可读)
- CODEP:** 代码保护使能位
 $\overline{\text{CODEP}}=1$ 禁止程序存储器代码保护
 $\overline{\text{CODEP}}=0$ 使能程序存储器代码保护
- LVREN:** 欠压检测功能使能位
 $\text{LVREN}=1$ 使能欠压检查功能
 $\text{LVREN}=0$ 禁止欠压检查功能
- RSTEN:** P0.3/ $\overline{\text{RST}}$ 引脚功能选择
 $\text{RSTEN}=1$ P0.3/ $\overline{\text{RST}}$ 引脚配置为外部复位输入
 $\text{RSTEN}=0$ P0.3/ $\overline{\text{RST}}$ 引脚功能为数字输入口
- WDTEN:** 看门狗定时器(WDT)使能位
 $\text{WDTEN}=1$ 使能 WDT
 $\text{WDTEN}=0$ 禁止 WDT

1.6 在线串行编程

如图 1.2、1.3 所示，在最终应用电路中可对 KF8F312M4 单片机进行在线串行编程。实现编程仅需要五根线包括：时钟线(SPCLK)、数据线(SPDAT)、电源线(VDD)、地线(VSS)、编程电压线(MODE)。

开发人员和用户可以使用未编程的单片机来制造电路板，然后对其在线编程，调试等。只要有电脑、USB 下载线和编程器，即可在任何时候，任何地点，对电路板上的单片机程序进行更新。

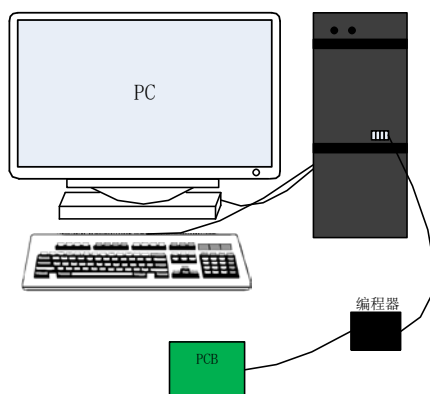


图 1.3 在线调试系统示意图

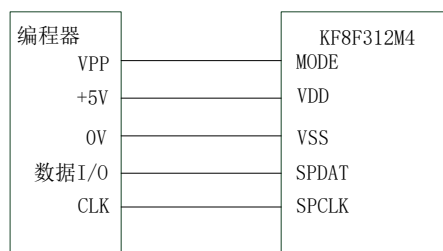


图 1.4 在线串行编程连接图

2 I/O端口介绍

KF8F312M4 单片机共有 20 个引脚，1 脚接电源正极，20 脚接电源负极，其余管脚均为 I/O 端口，包括 P0 口、P1 口、P2 口。P0 口共有 P0.0~P0.5 六个引脚，P1 口共有 P1.0~P1.7 八个引脚，P2 口共有 P2.4~P2.7 四个引脚。

注：用户在正常使用时，通常会有一些用不到的引脚，如果直接把这些管脚悬空，而不做其他处理可能使单片机功耗增大，因此建议将那些不用的引脚设置为数字输出模式，如果 P0.3 未用，应外接上拉电阻。

2.1 P0 口

如图 2.1 所示，P0 口共有 6 个引脚，对应管脚为 2~4、17~19。在线编程时 P0 口的 MODE、SPCLK、SPDAT 作为编程脚使用。P0.3 只能作为输入口且没有上拉功能，其它端口均可作为普通 I/O 口且带有上拉功能，P0 口所有引脚都有电平变化中断功能。各引脚功能表 2.1 所示。

表 2.1 P0 口各引脚功能介绍

引脚名	I/O	引脚功能	引脚说明
2	I/O	P0.5	带上拉和电平变化中断功能的双向输入输出端口
		T1CK	T1 时钟输入
3	I/O	P0.4	带上拉和电平变化中断功能的双向输入输出端口
		AN3	ADC 输入通道 3
		$\overline{T1G}$	T1 门控信号输入
		CLKOUT	系统时钟输出
4	I	P0.3	带电平变化中断的输入端口
		$\overline{RST}$	主复位信号输入
		MODE	编程模式选择
17	I/O	P0.2	带上拉和电平变化中断功能的双向输入输出端口
		AN2	ADC 输入通道 2
		T0CK	T0 时钟输入
		C1OUT	模拟比较器 1 输出
		INT0	外部中断 0 输入
18	I/O	P0.1	带上拉和电平变化中断功能的双向输入输出端口
		AN1	ADC 输入通道 1
		ADVRIN	AD 外部参考电压输入
		C1-	模拟比较器 1 负端输入
		SPCLK	在线编程时钟输入
19	I/O	P0.0	带上拉和电平变化中断功能的双向输入输出端口
		AN0	ADC 输入通道 0
		C1+	模拟比较器 1 正端输入
		SPDAT	在线编程数据输入/输出

2.1.1 P0 口相关的寄存器

表 2.2 与 P0 端口相关的寄存器

地址	名称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
05H	P0	-	-	P05	P04	P03	P02	P01	P00
25H	TR0	-	-	TR05	TR04	TR03	TR02	TR01	TR00
36H	IOCL	-	-	IOCL5	IOCL4	IOCL3	IOCL2	IOCL1	IOCL0
35H	PUR	-	-	PUR5	PUR4	-	PUR2	PUR1	PUR0
0BH	INTCTL	AIE	PUIE	T0IE	INT0IE	POIE	T0IF	INT0IF	P0IF
2EH	PCTL	-	-	-	SLVREN	-	-	POR	LVR
21H	OPTR	PUPH	INT0SE	T1CS	T1SE	PSA	PS2	PS1	PS0
31H	ANSEL	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0
2FH	OSCCTL	CKOEN	IRCS2	IRCS1	IRCS0	-	-	-	

2.1.1.1 P0 口状态寄存器

寄存器 P0 各位对应 P0 口相应引脚当前的状态，如寄存器 2.1 所示：

寄存器2.1: P0: P0口状态寄存器(地址: 05H)

	bit7							bit0
复位值 --xx xxxx	-	-	P05	P04	P03	P02	P01	P00
	U	U	R/W	R/W	R/W	R/W	R/W	R/W

P0<5:0>: P0 口各引脚状态位

P0x=1 对应引脚为逻辑高电平

P0x=0 对应引脚为逻辑低电平

2.1.1.2 P0 口方向控制寄存器

如寄存器 2.2 所示，TR0 为 P0 口方向控制寄存器，当 TR0 某位置 1 时，将该引脚设置为输入，此时引脚为三态(悬空)，TR0 某位清 0，对应引脚设置为输出。

寄存器2.2: TR0: P0口方向控制寄存器(地址: 25H)

	bit7							bit0
复位值 --11 1111	-	-	TR05	TR04	TR03	TR02	TR01	TR00
	U	U	R/W	R/W	R	R/W	R/W	R/W

TR0<5:4>: P0 口各引脚方向控制位

TR0<2:0>: P0 口各引脚方向控制位

TR0x=1 对应的引脚设置为输入

TR0x=0 对应的引脚设置为输出

TR03: P0.3 引脚方向控制位，始终为 1

2.1.1.3 上拉功能控制寄存器

KF8F312M4 中除了 P0.3 口没有内部上拉功能外，其它引脚均带有上拉功能，可通过上拉功能控制寄存器和 OPTR 寄存器中的 $\overline{\text{PUPH}}$ 来控制上拉功能是否打开。

如果要将某引脚的上拉功能打开，需要先将 $\overline{\text{PUPH}}$ (上拉功能总使能位) 位清 0，允许 P0 口上拉功能打开，然后再将要打开上拉功能的引脚，所对应的上拉功能控制位置 1 即可。寄存器 2.3 为上拉功能控制寄存器。

注：只有将引脚设置为数字输入时才开启上拉电阻功能，如果将某引脚设置为输出或者设置为模拟输入时将会自动禁止该引脚的上拉功能。

寄存器2.3: PUR: 弱上拉控制寄存器(地址: 35H)

bit7								bit0
复位值	-	-	PUR5	PUR4	-	PUR2	PUR1	PUR0
--11 -111	U	U	R/W	R/W	U	R/W	R/W	R/W

PUR<5:4>: 上拉功能使能位

PUR<2:0>: 上拉功能使能位

PURx=1 使能上拉功能

PURx=0 禁止上拉功能

2.1.1.4 电平变化中断控制寄存器

P0 口每个引脚都具有电平变化中断功能，当引脚的当前电平与上次读 P0 寄存器时的电平不匹配时将产生电平变化中断。如寄存器 2.4 所示，IOCL 为电平变化中断控制寄存器，将 IOCL 某位置 1 将开启对应引脚的电平变化中断功能，如果该引脚电平发生变化，不管电平变化中断是否使能，电平变化中断标志位(P0IF)都会置 1，如果全局中断使能位(AIE)和电平变化中断使能位(POIE)都已置 1，则会响应中断进入中断服务子程序。P0 口所有引脚的电平变化中断共用一个标志位 P0IF。

注：1. 只有将引脚设置为数字输入时才开启电平变化中断功能，如果将某引脚设置为输出或者设置为模拟输入时将会自动禁止该引脚的电平变化中断功能。
2. P0口各引脚的电平变化中断共用一个中断使能位和中断响应标志位。

寄存器2.4: IOCL: 电平变化中断控制寄存器(地址:36H)

复位值 --00 0000	bit7							bit0
	-	-	IOCL5	IOCL4	IOCL3	IOCL2	IOCL1	IOCL0
	U	U	R/W	R/W	R/W	R/W	R/W	R/W

IOCL<5:0>: P0 端口引脚电平变化中断使能控制位

IOCLx=1 使能对应引脚的电平变化中断

IOCLx=0 禁止对应引脚的电平变化中断

2.1.2 P0 口各引脚内部原理功能框图

如图 2.2、2.3 所示，为 P0 口引脚原理功能框图，各引脚的功能不同其原理功能图会有一点差别，但是原理是相通的。

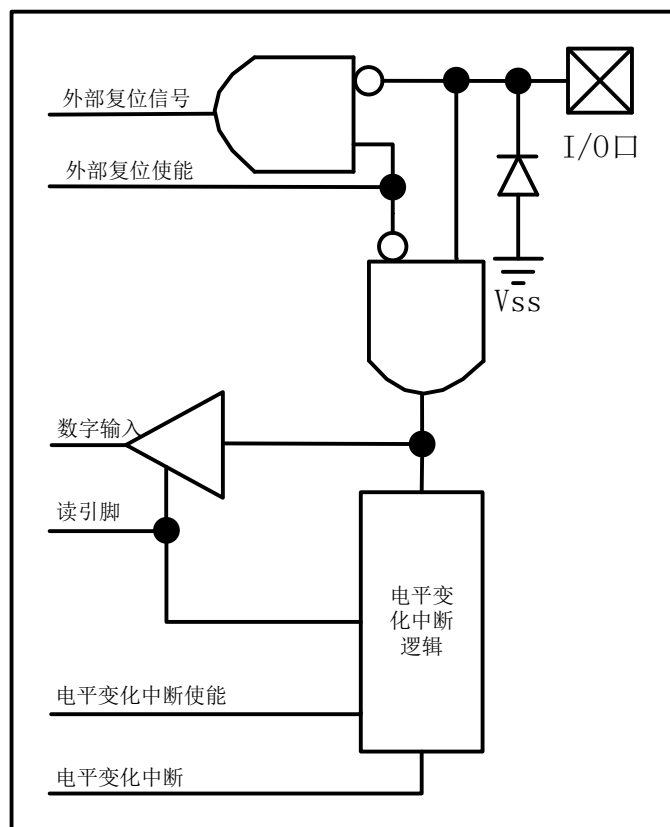


图 2.2 引脚 P0.3 原理功能框图

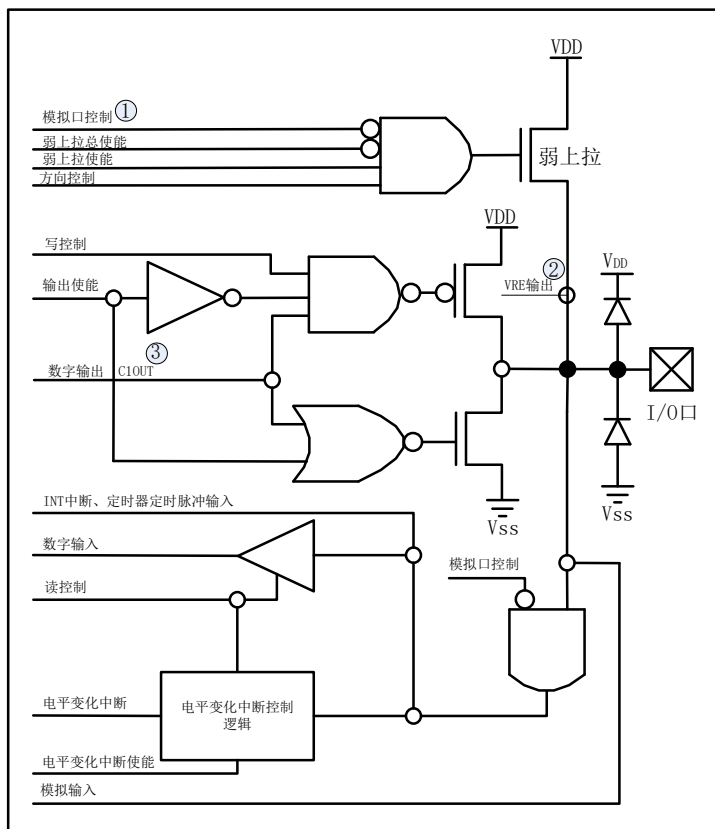


图 2.3 引脚 P0.0~P0.2、P0.4、P0.5 原理功能框图

- 注：
- 1、P0.5 没有模拟通道
 - 2、VRE 输出只针对 P0.4
 - 3、C1OUT 只针对 P0.2

2.2 P1 口

如图 2.1 所示，P1 口具有 8 个引脚，对应管脚为 5~9、14~16。所有管脚均可作为普通 I/O 口，部分引脚可作为 AD、外部中断 1/2 等的输入和 PWM 的输出。引脚功能如表 2.3 所示。

表 2.3 P1 口各引脚功能

引脚名	I/O	引脚功能	引脚说明
5	I/O	P1.5	双向输入输出端口
		P3A	PWM3 输出
6	I/O	P1.4	双向输入输出端口
		P3B	PWM3 输出
		C2OUT	模拟比较器 2 输出
7	I/O	P1.3	双向输入输出端口
		AN7	ADC 输入通道 7
		INT2	外部中断 2 输入
		P3C	PWM3 输出
8	I/O	P1.6	双向输入输出端口
9	I/O	P1.7	双向输入输出端口
14	I/O	P1.2	双向输入输出端口
		AN6	ADC 输入通道 6
		INT1	外部中断 1 输入
		P3D	PWM3 输出
15	I/O	P1.1	双向输入输出端口
		AN5	ADC 输入通道 5
		C2-	模拟比较器 2 负端输入
		PWM2	PWM2 输出
16	I/O	P1.0	双向输入输出端口
		AN4	ADC 输入通道 4
		C2+	模拟比较器正端输入
		PWM1	PWM1 输出

2.2.1 P1 口相关的寄存器

表 2.4 与 P1 口相关的寄存器

地址	名称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
07H	P1	P17	P16	P15	P14	P13	P12	P11	P10
27H	TR1	TR17	TR16	TR15	TR14	TR13	TR12	TR11	TR10
2CH	EIE1	-	ADIE	INT2IE	INT1IE	C1IE	PWM2IE	T2IE	T1IE
0CH	EIF1	-	ADIF	INT2IF	INT1IF	C1IF	PWM2IF	T2IF	T1IF
15H	PWMCTL	INT2SE	INT1SE	-	-	-	-	PWM2ON	PWM1ON
31H	ANSEL	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0
57H	PWM3CTL0	P1M1	P1M0	PDT1	PDT0	PWM3M3	PWM3M2	PWM3M1	PWM3M0

2.2.1.1 P1 口状态寄存器

寄存器 P1 对应端口 P1 引脚作为普通 I/O 口时的状态。如寄存器 2.5 所示

寄存器2.5: P1: P1口状态寄存器(地址: 07H)

bit7									bit0
复位值 xxxx xxxx	P17	P16	P15	P14	P13	P12	P11	P10	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

P1<7:0>: P1 口各引脚状态位

P1x=1 对应引脚为逻辑高电平

P1x=0 对应引脚为逻辑低电平

2.2.1.2 P1 口方向控制寄存器

如寄存器 2.6 所示, 通过将寄存器 TR1 中的某位置 1, 将对应管脚设置为输入口。清 0 设置为输出口, 系统复位时, P1 口各引脚默认为输入口。

寄存器2.6: TR1: P1口方向控制寄存器(地址: 27H)

bit7									bit0
复位值 1111 1111	TR17	TR16	TR15	TR14	TR13	TR12	TR11	TR10	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

TR1<7:0>: P1 口引脚方向控制位

TR1x=1 P1 口对应引脚被配置为输入端口

TR1x=0 P1 口对应引脚被配置为输出端口

2.2.2 P1 口原理功能框图

如图 2.4 所示，P1 口共有 8 个引脚，根据各引脚的作用不同，P1 口引脚原理功能框图中的模拟、PWM1/2/3、INT1/2 等部分有微小的变动。

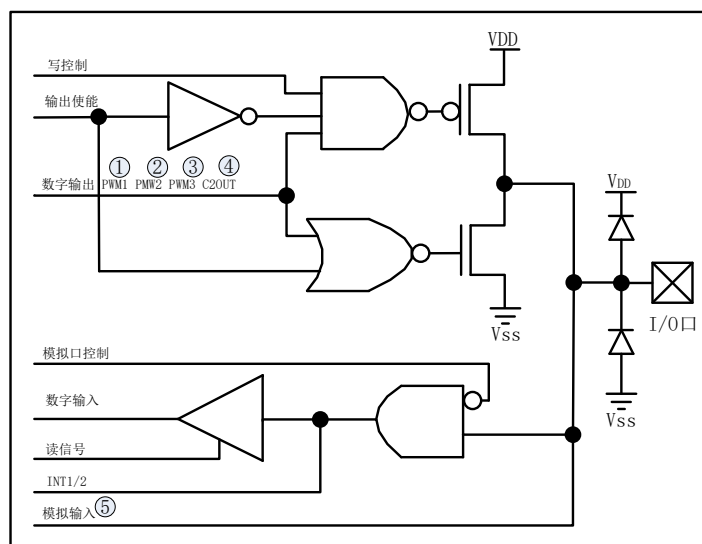


图 2.4 P1 口引脚原理功能框图

- 注：
- 1、PWM1 只针对 P1.0
 - 2、PWM2 只针对 P1.1
 - 3、PWM3 分别为 P1.5、P1.4、P1.3 和 P1.2
 - 4、C2OUT 只针对 P1.4
 - 5、P1.4/P1.5/P1.6/P1.7 没有模拟功能

2.3 P2 口

如图 2.1 所示，P2 口共有 4 个引脚，对应管脚 10~13。所有管脚均可作为普通 I/O 口的输入或者输出口。

表 2.5 P2 口各引脚功能

引脚名	I/O	引脚功能	引脚说明
10	I/O	P2.7	双向输入输出端口
		AN11	ADC 输入通道 11
		TX	USART 全双工异步发送
		CK	USART 半双工同步时钟
11	I/O	P2.6	双向输入输出端口
		AN10	ADC 输入通道 10
12	I/O	P2.5	双向输入输出端口
		AN9	ADC 输入通道 9
		RX	USART 全双工异步接收
		DT	USART 半双工同步数据
13	I/O	P2.4	双向输入输出端口
		AN8	ADC 输入通道 8

2.3.1 P2 口相关的寄存器

表 2.6 与 P2 口相关的寄存器

地址	名称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
06H	P2	P27	P26	P25	P24	-	-	-	-
26H	TR2	TR27	TR26	TR25	TR24	-	-	-	-
1DH	ANSEH	-	-	-	-	ANS11	ANS10	ANS9	ANS8

2.3.1.1 P2 口状态寄存器(P2)

寄存器 P2 各位对应端口 P2 口各引脚作为普通 I/O 口时的状态。如寄存器 2.7 所示

寄存器2.7: P2: P2口状态寄存器(地址: 06H)

复位值 xxxx ----	bit7				bit0			
	P27	P26	P25	P24	-	-	-	-
	R/W	R/W	R/W	R/W	U	U	U	U

P2<7:4>: P2 口各引脚状态位

P2x=1 对应引脚为逻辑高电平

P2x=0 对应引脚为逻辑低电平

2.3.1.2 P2 口方向控制寄存器(TR2)

如寄存器 2.8 所示，通过将寄存器 TR2 中的某位置 1，将对应管脚设置为输入口。清 0 设置为输出口。

寄存器2.8: TR2: P2口方向控制寄存器(地址: 26H)

bit7				bit0			
复位值 1111 ----	TR27	TR26	TR25	TR24	-	-	-
	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0

TR2<7:4>: P2 口各引脚方向控制位

TR2x=1 P2 口对应引脚被配置为输入端口

TR2x=0 P2 口对应引脚被配置为输出端口

2.3.1.3 模拟/数字口配置寄存器(ANSEH)

ANSEH 用来将 P2 口配置为模拟或数字 I/O 口，ANS8 对应 P2.4，以此类推 ANS11 对应 P2.7。ANSx=0 将对应引脚配置为数字 I/O 口。ANSx=1 将对应的引脚配置为模拟口。

寄存器2.9: ANSEH: 模拟/数字口设置寄存器(地址: 1DH)

bit7				bit0			
复位值 ---- 0000	-	-	-	ANS11	ANS10	ANS9	ANS8
	U	U	U	R/W	R/W	R/W	R/W

ANS<11:8>: P2 口模拟/数字口配置位

ANSx=1 将对应引脚配置为模拟口

ANSx=0 将对应引脚配置为数字 I/O 口

2.3.2 P2 口原理功能框图

如图 2.5 所示，P2 口共有 4 个引脚。

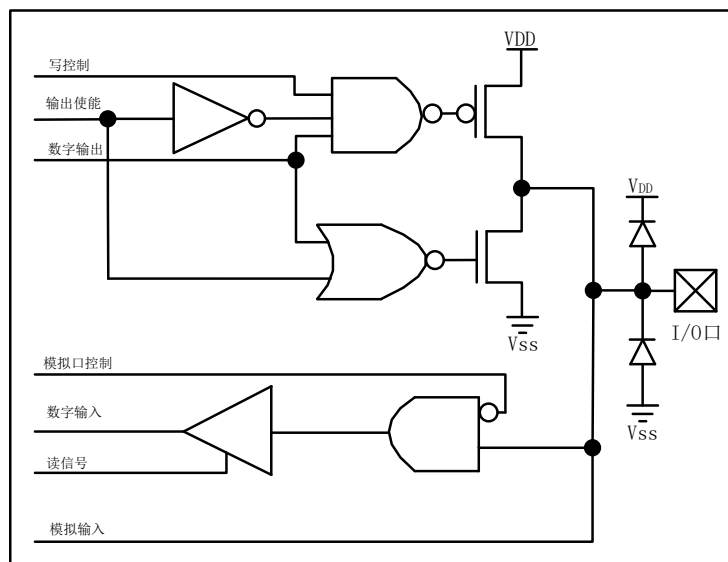


图 2.5 P2 口引脚原理功能框图

3 存储器

如图 3.1 所示，KF8F312M4 中存储器主要由程序存储器(ROM)和数据存储器(RAM)组成，程序存储器和数据存储器地址空间相互独立。其中程序存储器为 4K×16 位的 FLASH 存储器；数据存储器由特殊功能寄存器和通用寄存器组成，特殊功能寄存器空间为 128×8 位，通用数据寄存器空间为 256×8 位。另外 KF8F312M4 中还有一些其它存储器，包括:BLOCK EEPROM，寄存器组 R0~R7、8 级硬件堆栈、ID 地址单元等。

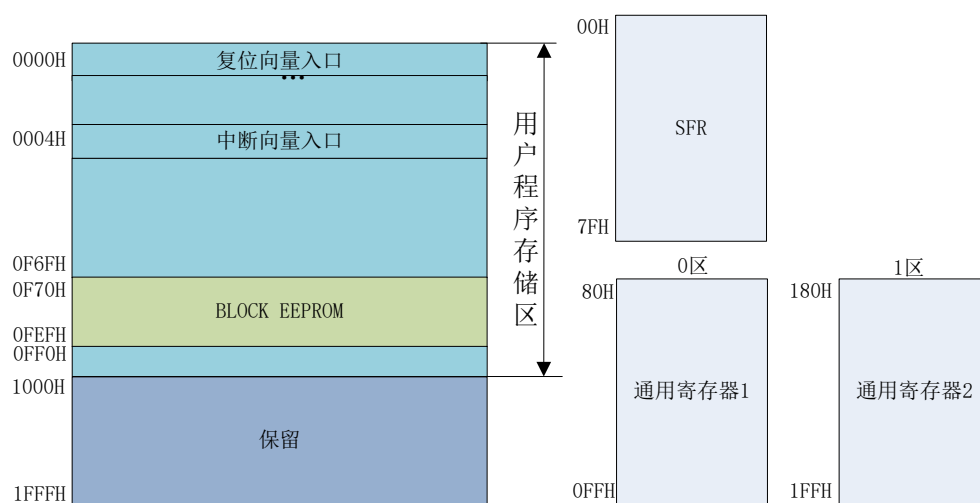


图 3.1 存储器组织图

3.1 程序存储器(ROM)区

KF8F312M4 有一个 13 位的程序计数器，最大可寻址 8K×16 的程序存储空间。而在 KF8F312M4 中实际只实现了 4K×16 的程序空间，地址为 0000H~0FFFH，其中 0F70H~0FEFH 共 128 个字可复用为 BLOCK EEPROM。复位向量入口地址为 0000H，中断向量入口地址为 0004H。

如图 3.2 所示，程序计数器(PC)的低 8 位(PC<7:0>)来自特殊功能寄存器 PCL,高 5 位(PC<12:8>)来自 PCH 寄存器。在任何复位发生后 PC 值将被清 0。在有任何未屏蔽中断发生后 PC 值将指向 0004H 地址。图 3.3 为程序存储器区的地址映射图。

在用户的程序中，每当执行一条汇编指令 PC 值会自动加 1，指向下一条要执行的指令。当有子程序调用或响应中断时，CPU 会将 PC+1 后的值压入堆栈进行保存，然后将子程序或中断入口地址送到 PC 中，CPU 根据 PC 的值跳转到对应的地址执行命令。

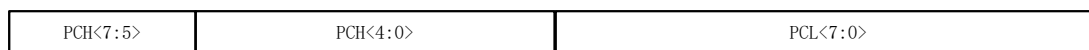


图 3.2 程序计数器 (PC)

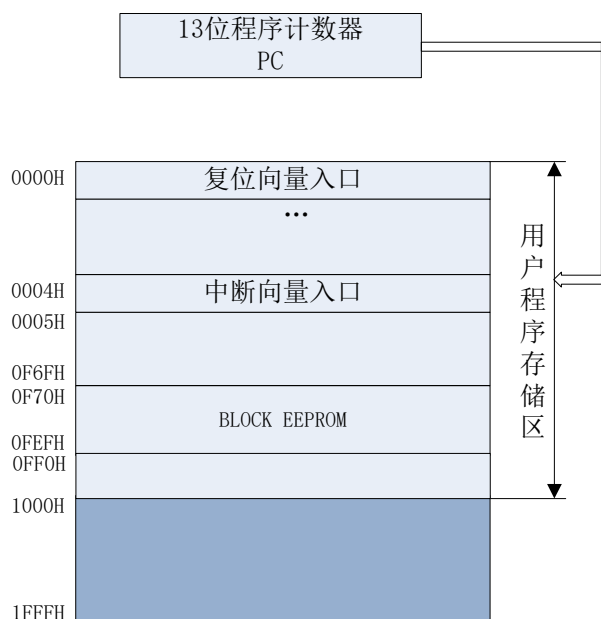


图 3.3 KF8F312M4 程序存储器映射

3.2 数据存储器(RAM)区

如图 3.4 所示，KF8F312M4 中的数据存储器由三个区组成，每个区的空间都是 128 字节，其中一个区用作特殊功能寄存器区(SFR)使用；另外两个存储区为通用寄存器区，由用户支配。SFR 地址空间为 00H~7FH；通用寄存器区地址为 80H~0FFH 和 180H~1FFH，在这里将两个通用寄存器区称作 0 区和 1 区。

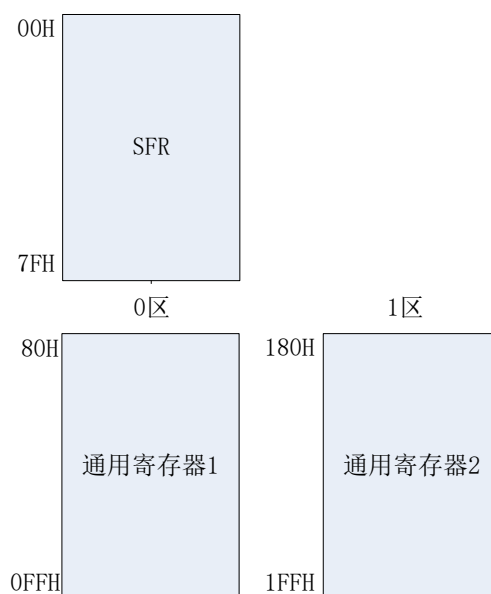


图 3.4 数据存储器地址映射图

3.2.1 通用寄存器区

如图 3.4 所示，通用寄存器的空间为 128×2 字节，0 区和 1 区通过 PSW 中的 RP0(PSW.5) 位进行选择，如表 3.1 所示。

表 3.1 通用寄存器区的选择

RP0(PSW.5)	当前使用的通用寄存器区	地址
0	0 区	80H~0FFH
1	1 区	180H~1FFH

3.2.2 特殊功能寄存器(SFR)区

KF8F312M4 内部的 I/O 口控制、定时/计数器、PWM、中断等各种控制寄存器和状态寄存器都称为特殊功能寄存器。附录 1 列出 SFR 的地址映射及复位初始值等。

状态字寄存器(PSW): 如寄存器 3.1 所示, PSW 的低三位是算术运算标志位, 在进行加、减等运算时对它们产生影响(具体请参考汇编指令部分)。TO 和 PD 是复位状态位, 当单片机有复位或看门狗超时、执行休眠等指令时, 会对这两位产生影响。RP0 为通用存储器区选择位。

寄存器3.1: PSW: 状态字寄存器(地址: 03H)

bit7				bit0			
-	-	RP0	TO	PD	Z	DC	CY
保留	保留	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x

RP0: 通用寄存器区选择位

RP0=0 选择存储器区 0

RP0=1 选择存储器区 1

TO: 超时标志位

TO=1 在上电复位、CWDT 指令或 IDLE 指令执行之后

TO=0 WDT 超时被清 0

PD: 上电复位标志位

PD=1 上电复位或执行 CWDT 指令后

PD=0 执行 IDLE 指令后被清 0

Z: 零状态标志位

Z=1 算术运算或者逻辑运算的运行结果为 0

Z=0 算术运算或者逻辑运算的运行结果不为 0

DC: 辅助进/借位标志位

DC=1 执行结果的低 4 位向高 4 位有进位(加指令)或没有借位(减指令)

DC=0 执行结果的低 4 位向高 4 位没有进位(加指令)或有借位(减指令)

CY: 进位/借位标志位

CY=1 执行结果(8 位)向高位有进位时(加指令)或没有借位(减指令)

CY=0 执行结果(8 位)向高位无进位时(加指令)或有借位(减指令)

注: 对于借位的情况, 当指令执行后, 低四位(或高四位)向高位有借位时, DC(或CY)标志为0, 当没有借位时其值为1。关于对标志位是否产生影响的指令请参考“汇编指令集”部分。

3.3 BLOCK EEPROM

KF8F312M4 在程序存储区后面的地址开辟了一个 128X16 位的 BLOCK EEPROM。如图 3.5 所示，写 BLOCK EEPROM 时，BLOCK EEPROM 中所有内存单元以连续四个地址为一个数据块，四个数据块为一页。

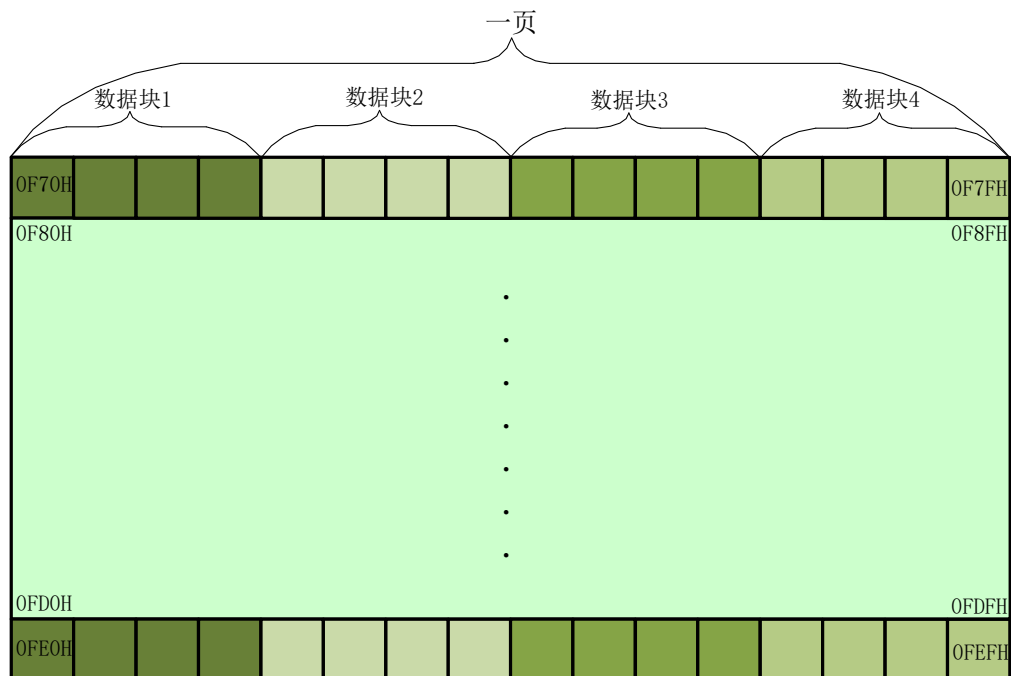


图 3.5 BLOCK EEPROM 地址映射图

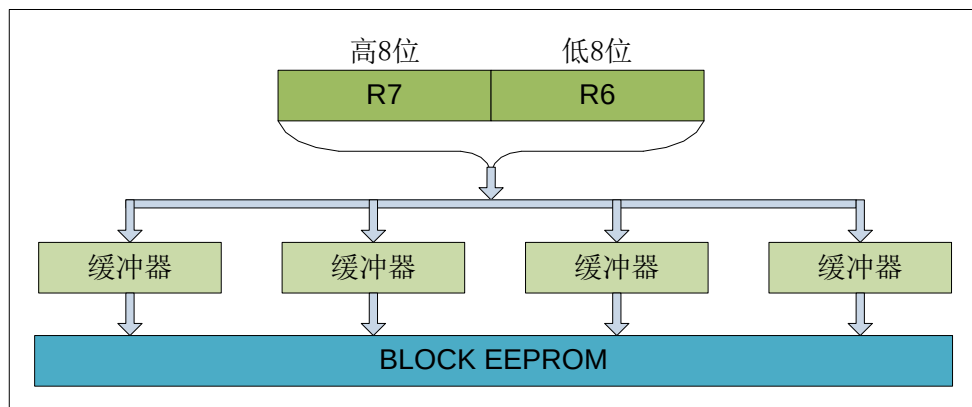


图 3.6 写操作

如图 3.6 所示，在写 BLOCK EEPROM 时，有四个 16 位的缓冲寄存器，用来临时存放要写入 BLOCK EEPROM 中的数据

表 3.2 与 BLOCK EEPROM 相关的寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
3AH	BADDRH	-	-	-	BLOCK EEPROM 地址指针高 5 位				
3BH	BADDRL	BLOCK EEPROM 地址指针低 8 位							
3CH	EECTL1	控制寄存器 1							
3DH	EECTL2	控制寄存器 2							
-	R6	数据寄存器 R6							
-	R7	数据寄存器 R7							

3.3.1 寄存器R6/R7

使用 BLOCK EEPROM 时，用来存放要写入或者读出 BLOCK EEPROM 的数据，R6 存放数据的低 8 位，R7 存放数据的高 8 位。不使用 BLOCK EEPROM 时，R6/R7 由用户自由分配。

3.3.2 寄存器BADDRH/BADDRL

如寄存器 3.1 所示，BADDRH/BADDRL 地址位于特殊功能寄存器区的 3AH/3BH。用来存放要写入 BLOCK EEPROM 的 13 位的地址信息，BADDRH 存放地址的高 5 位，BADDRL 存放地址的低 8 位。

寄存器3.1: BADDRH: 数据指针高5位(地址: 3AH)

复位值 ---0 0000	-	-	-	bit4	bit3	bit2	bit1	bit0
	U	U	U	R/W	R/W	R/W	R/W	R/W

BADDRL: 数据指针低8位(地址: 3BH)

复位值 0000 0000	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

3.3.3 寄存器EECTL1/EECTL2

EECTL1/EECTL2 为写 BLOCK EEPROM 控制寄存器，地址位于特殊功能寄存器区的 3CH/3DH。用户在写 BLOCK EEPROM 时，将 R6/R7 中送入要写入的数据，BADDRH/BADDRL 中送入要写入的地址，然后通过向 EECTL1 和 EECTL2 送入固定的写命令，将要写入 BLOCK 的数据存到对应的缓冲寄存器中。在读 BLOCK EEPROM 时，将要读的地址送到 BADDRH/BADDRL 中，然后向 EECTL1 写入固定的读命令，把要读的数据送到 R6/R7 中。

3.3.4 写BLOCK EEPROM

写 BLOCK EEPROM 时，只能对 BLOCK EEPROM 成块写入数据，不能单独将一个字节(或字)的数据写入某块的一个字节(或字)中，如果实际上写入 BLOCK EEPROM 中的数据没有 4 个字或不能被 4 整除(例如要写入一组 7 个字的数据),需要将块中不需要写入数据的单元写入 0 或者其它值，否则可能会导致写入的数据出错。如果原来的 BLOCK EEPROM 保存有数据，现在需要修改原数据中的一个字或者几个字，其它单元的值不变，则需要先将其对应块中其它数据读出来保存，然后再根据实际情况将需要修改的值和之前读出的值写入即可。

在写 BLOCK EEPROM 时，必须先对每个页的第一块进行写操作，以擦除本页的数据，如果没有对第一块进行写操作，直接写后面块则本页的所有数据都不会被擦除。即只有对每个页的第一块进行写操作后才会擦除本页之前的数据，对其它块写操作不会产生擦除本页数据的命令，可能导致写入数据出错。

注：1. 写BLOCK EEPROM时，从EEPROM地址开始处，连续的4个字作为一个数据块，连续的4个数据块作为一个页，因此整个BLOCK EEPROM共有8页数据。

2. 写BLOCK EEPROM时，不管其存储单元是否有数据，都要先执行一次擦除命令，且擦除命令只能在每页的第一个数据块写完后才会执行擦除命令，将本页所有单元数据擦除。而对每页的其他数据块写操作时不会有擦除命令发出。

3. 将各页第一个数据块写完后，CPU将停止6ms执行擦除和写命令，写其他块时，停止3ms执行写命令。

在写 BLOCK EEPROM 时，将要写入的数据送到 R6/7，地址送到 BADDRH/L 后，通过执行以下指令完成写操作：


```
MOV R5, INTCTL           ;保存当前的中断状态
MOV DATA_INTCTL, R5     ;该样例要求DATA_INTCTL 在0区,否则添加切区
CLR INTCTL, 7            ;关闭总中断
JNB INTCTL, 7
JMP $-2

MOV R5, OSCCTL           ;保存当前的时钟状态
MOV DATA_OSCCTL, R5     ;该样例要求DATA_OSCCTL在0区,否则添加切区
MOV R5, #0X50            ;切换到1M
MOV OSCCTL, R5

;;以下时序不可更改
MOV R5, #0X84
MOV EECTL1, R5
MOV R5, #0X69
MOV EECTL2, R5
MOV R5, #0X96
MOV EECTL2, R5
SET EECTL1, 1
NOPZ
NOPZ
NOPZ
NOPZ
NOPZ
NOPZ
NOPZ
NOPZ
NOPZ
NOPZ
MOV R5, #0X80            ;关闭Flash的写操作, 防止意外写
MOV EECTL1, R5

MOV R5, DATA_OSCCTL     ;恢复时钟状态
MOV OSCCTL, R5

JNB DATA_INTCTL, 7      ;恢复中断状态
SET INTCTL, 7
```

写 BLOCK EEPROM 的步骤为:

将要写入的数据的送到 R6/R7;

将对应的 BLOCK EEPROM 地址送到 BADDRH/BADDRL;

执行上面的写命令, 此时, CPU 将要写入数据的一个字保存到 BLOCK EEPROM 的数据缓冲器中;

重复执行步骤 1、2、3 四次, 此时 CPU 自动将要写入第一块的数据分别存入对应的 BLOCK EEPROM 的数据缓冲器中;

当上边第四次写命令执行完后, CPU 发出擦除本页的命令, 将本页原来的数据全部擦除, 擦除完毕后, 将数据缓冲器中的数据送到对应的地址中。在这个过程中 CPU 停止其它工作 6ms 用来执行擦除和写入数据的命令。

重复执行步骤 1、2、3 四次, 将数据写入本页的第二块。当执行完第四次写命令后, 因本次写的不是页的第一块, CPU 不会执行擦除命令, 仅将数据缓冲器中的数据写入对应的存储单元, 写操作耗时 3ms。

3.3.5 读BLOCK EEPROM

在读 BLOCK EEPROM 时，将要读取的地址送到 BADDRH/L 后，通过执行以下操作完成读操作：

```
MOV R5,#0X81
MOV EECTL1,R5
NOPZ
NOPZ
```

上面指令中的立即数 0X81 是固定不变的。此时，该地址的数据高 8 位被送到 R7，低 8 位送到 R6。

读 BLOCK EEPROM 是逐字读取的，不要求一块一块的读。读 BLOCK EEPROM 时通过向 EECTL1 写入 0X81 来执行读命令。

读 BLOCK EEPROM 的步骤如下：

- 1、将要读的数据单元的地址送到 BADDRH/L 中；
- 2、向 EECTL1 写入读命令；
- 3、两个指令周期后该单元的数据被送到 R6/R7。

3.4 寄存器组Rn

KF8F312M4 芯片中有一个工作寄存器组 R0~R7，可用做间接寻址的中间寄存器，存放操作数的地址；在读和写 BLOCK EEPROM 时，R6 和 R7 用来存放要写的数据和读出的数据；隐含目的操作数的指令中，默认 R0 作为目的操作数(如: RRCR 0X81)；在读晶振校准值时，默认将读到的值送到 R0 中。

3.5 ID地址单元

KF8F312M4 的程序存储器空间的最后 4 个地址单元被指定为 ID 地址单元，地址为 FFCH~FFFH。用户可在其中存放校准值(内部振荡器和 VRE)和或其它信息。正常运行时不能对这些地址单元进行访问，但在编程/校验时这些地址单元是可读写的。

4 汇编指令及寻址方式

4.1 寻址方式

KF8F312M4 系列单片机提供 5 种寻址方式，分别为：寄存器寻址、直接寻址、立即数寻址、寄存器间接寻址和位寻址。KF8F312M4 的指令可以没有操作数、一个操作数、两个操作数。

4.1.1 寄存器寻址

采用这种寻址方式的指令中的操作数为寄存器组 R0-R7 的一个。

例：

CLR R0 ; R0←0 将寄存器 R0 清 0
只有一个操作数(R0 的值)，寻址方式为寄存器寻址。

ADD R0, R1
两个操作数 (R0 和 R1)，寻址方式为寄存器寻址。

4.1.2 直接寻址

在指令中的操作数为某个寄存器的直接地址，该地址指出其参与运算的数据所在的地址。直接寻址可以是：特殊功能寄存器、通用数据存储器。

例：

MOV R0,0X81 ; R0←(81H) 将 81H 单元的数据送到 R0 中
指令中，源操作数寻址方式为直接寻址，目的操作数为寄存器寻址。

INC 0X3B ; 3BH←(3BH)+1 将地址 3BH 里的值加 1，3BH 即 BADDRL。
指令中含有一个操作数，寻址方式为直接寻址。

4.1.3 立即数寻址

在指令中的操作数为立即数。

例：

MOV R0,#0X20 ; R0←0X20 将立即数 0X20 送到寄存器 R0 中
ADD R0,#0X20 ; R0←(R0)+0X20 寄存器 R0 的值与 0X20 相加结果送到 R0
AND R0,#0X20 ; R0←(R0)&0X20 寄存器 R0 的值与 0X20 相与结果送到 R0
以上三条指令中源操作数都是#0X20，为立即数寻址，目的操作数为寄存器寻址。

4.1.4 寄存器间接寻址

这种寻址方式中,寄存器的内容指定操作数的地址,即寄存器中存放的是操作数的地址。间接寻址只有两条指令 LD 和 ST。

例:

LD R0, [R1] ; R0←((R1)) 将 R1 的内容所指地址单元的数据送到 R0
指令中源操作数的寻址方式为寄存器间接寻址,目的操作数为寄存器寻址。

ST [R0], R1 ; (R0)←(R1) 将 R1 的内容送到 R0 的内容所指向的地址单元
指令中目的操作数的寻址方式为寄存器间接寻址,源操作数为寄存器寻址。

4.1.5 位寻址

指令中的操作数是寄存器的某位,这样的寻址方式称为位寻址。

例:

CLR INTCTL,1 ; 将 INTCTL 的第 1 位清 0
CLR 0X80,1 ; 将 80H 的第 1 位清 0
JNB 0X80,1 ; 如果 80H 的第 1 位为 0 则跳过下一条指令执行后面的程序

4.2 汇编指令

KF8F312M4 系列单片机汇编指令共有 68 条,除子程序调用、子程序返回、中断返回、部分跳转指令为双周期指令外,其余指令均为单周期指令。所有指令都占两个字节。

按照指令的功能可将其分为:数据传送指令、算术运算指令、逻辑运算指令、位操作指令和转移指令和特殊指令。具体指令集请参考附录 2。

5 中断

KF8F312M4 单片机的中断源有：

- INT0/1/2
- T0/1 溢出中断
- P0 口引脚电平变化中断
- A/D 中断
- CMP1/2 中断
- PWM1/2/3 中断
- T2 中断
- 通用串行通讯接口 USART 的发送和接收中断

在本系列单片机中只有一个中断优先级。所有中断发生后都会跳转到中断入口地址 0004H，在中断服务子程序中通过检测相应的中断标志位来确定具体是哪个中断源触发发生。

KF8F312M4 中的 INT1/2、T1、T2、PWM1/2/3、ADC、USART、CMP1 和 CMP2 都属于外设，因此对应的中断称作外设中断，其它中断源产生的中断属于内设中断。中断逻辑如图 5.1 所示。

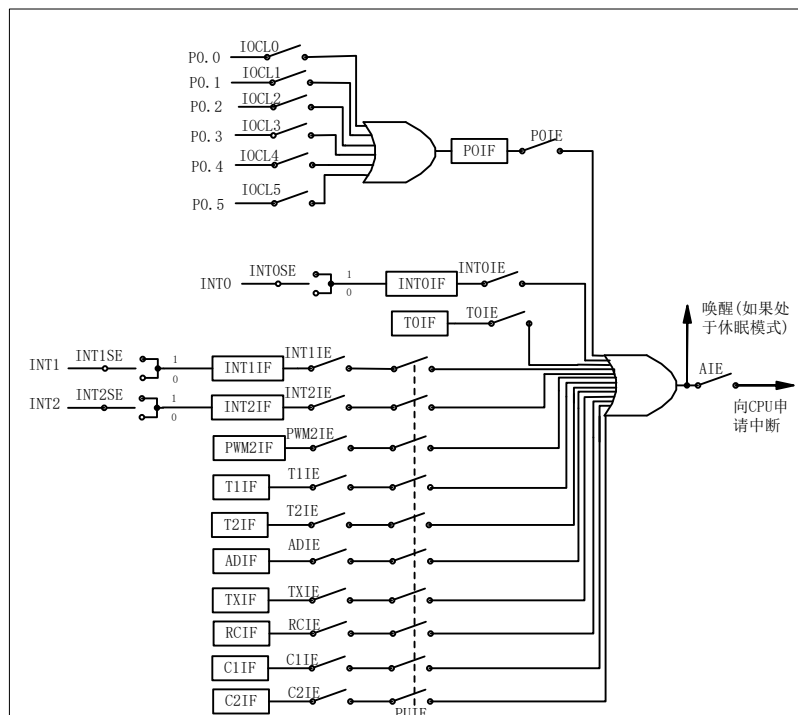


图 5.1 中断逻辑

5.1 中断相关的寄存器

表 5.1 与中断相关的寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
0BH	INTCTL	AIE	PUIE	TOIE	INT0IE	POIE	TOIF	INT0IF	POIF
0CH	EIF1	-	ADIF	INT2IF	INT1IF	C1IF	PWM2IF	T2IF	T1IF
0DH	EIF2	-	C2IF	RXIF	TXIF	-	-	-	-
10H	T1CTL	-	T1GC	T1CKS1	T1CKS0	-	T1SY	T1CS	T1ON
15H	PWMCTL	INT2SE	INT1SE	-	-	-	-	PWM2ON	PWM1ON
21H	OPTR	PUPH	INT0SE	T0CS	T0SE	PSA	PS2	PS1	PS0
2CH	EIE1	-	ADIE	INT2IE	INT1IE	C1IE	PWM2IE	T2IE	T1IE
2DH	EIE2	-	C2IE	RXIE	TXIE	-	-	-	-
36H	IOCL	-	-	IOCL5	IOCL4	IOCL3	IOCL2	IOCL1	IOCL0

5.1.1 中断控制寄存器(INTCTL)

如寄存器 5.1 所示，它包含：

- T0 寄存器溢出中断使能控制和标志位
- P0 口电平变化中断使能控制和标志位
- INT0 中断使能控制和标志位
- 外设中断和总中断使能位

AIE 为全局中断使能位，当其被清 0 时，禁止所有中断。PUIE 为外设中断使能位，当其被清 0 时禁止所有外设中断。具体的中断逻辑如图 5.1 所示。

注：1. 当中断条件满足时，无论相应的中断使能位或者全局中断使能位AIE的状态如何，中断标志位将被硬件置1。
2. 中断条件满足时，中断标志位通过硬件置1, 而清零则需要软件完成。

寄存器5.1: INTCTL: 中断控制寄存器(地址: 0BH)

复位值	bit7				bit0			
0000 0000	AIE	PUIE	TOIE	INT0IE	POIE	TOIF	INT0IF	POIF
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

AIE: 全局中断使能位

AIE=1 使能所有未屏蔽的中断

AIE=0 禁止所有中断

PUIE: 外设中断使能位

PUIE=1 使能所有未屏蔽的外设中断

PUIE=0 禁止所有外设中断

TOIE: T0 溢出中断使能位

TOIE=1 使能 T0 中断

TOIE=0 禁止 T0 中断

INT0IE: INT0 中断使能位

INT0IE=1 使能 INT0 中断

INT0IE=0 禁止 INT0 中断

P0IE: P0 口电平变化中断使能位
P0IE=1 使能 P0 口电平变化中断
P0IE=0 禁止 P0 口电平变化中断

T0IF: T0 溢出中断标志位
T0IF=1 T0 寄存器溢出
T0IF=0 T0 寄存器未溢出

INT0IF: INT0 中断标志位
INT0IF=1 INT0/P0.2 产生外部中断
INT0IF=0 INT0/P0.2 未产生外部中断

P0IF: P0 口电平变化中断标志位
P0IF=1 引脚 P0.0~P0.5 至少有一个电平状态发生变化
P0IF=0 引脚 P0.0~P0.5 电平状态未发生变化

5.1.2 中断使能寄存器EIE1

如寄存器 5.2 所示，EIE1 是一个可读写的寄存器，它包含：

- ADC 中断使能位
- INT1/2 中断使能位
- CMP1 中断使能位
- PWM2 中断使能位
- T2 中断使能位
- T1 中断使能位

寄存器5.2: EIE1: 中断使能寄存器(地址: 2CH)

bit7								bit0
复位值	-	ADIE	INT2IE	INT1IE	C1IE	PWM2IE	T2IE	T1IE
-000 0000	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ADIE: AD 中断使能位
ADIE=1 使能 AD 中断
ADIE=0 禁止 AD 中断

INT2IE: INT2 中断使能位
INT2IE=1 使能 INT2 中断
INT2IE=0 禁止 INT2 中断

INT1IE: INT1 中断使能位
INT1IE=1 使能 INT1 中断
INT1IE=0 禁止 INT1 中断

C1IE: CMP1 中断使能位
C1IE=1 使能 CMP1 中断
C1IE=0 禁止 CMP1 中断

PWM2IE: PWM2 中断使能位
PWM2IE=1 使能 PWM2 中断
PWM2IE=0 禁止 PWM2 中断

T2IE: T2 与 PP3 匹配中断允许位

T2IE=1 允许 T2 与 PP3 匹配中断
T2IE=0 禁止 T2 与 PP3 匹配中断
T1IE: T1 中断使能位
T1IE=1 使能 T1 中断
T1IE=0 禁止 T1 中断

5.1.3 中断标志寄存器EIF1

如寄存器 5.3 所示，中断标志寄存器包含：

- AD 中断标志位
- INT1/2 中断标志位
- CMP1 中断标志位
- PWM2 中断标志位
- T2 中断标志位
- T1 中断标志位

寄存器5.3: EIF1: 外设中断标志寄存器 (地址0CH)

复位值	bit7						bit0	
-000 0000	-	ADIF	INT2IF	INT1IF	C1IF	PWM2IF	T2IF	T1IF
	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ADIF: AD 完成中断标志位

ADIF=1 AD 转换完成

ADIF=0 AD 转换没有完成

INT2IF: INT2 中断标志位

INT2IF=1 INT2/P1.3 产生外部中断

INT2IF=0 INT2/P1.3 未产生外部中断

INT1IF: INT1 中断标志位

INT1IF=1 INT1/P1.2 产生外部中断

INT1IF=0 INT1/P1.2 未产生外部中断

C1IF: 模拟比较器 CMP1 中断标志位

C1IF=1 模拟比较器 CMP1 输出发生改变(必须软件清 0)

C1IF=0 模拟比较器 CMP1 输出未发生改变

PWM2IF: PWM2 中断标志位

PWM2IF=1 PWM1/2 使能时，T1H 和 PP2 匹配

PWM2IF=0 PWM1/2 使能时，T1H 和 PP2 不匹配

T2IF: T2 与 PP3 匹配中断标志位

T2IF=1 发生了 T2 与 PP3 匹配

T2IF=0 未发生了 T2 与 PP3 匹配

T1IF: T1 寄存器溢出标志位

T1IF=1 T1 寄存器溢出或 PWM1/2 使能时，T1L 与 PP1 匹配

T1IF=0 T1 寄存器未溢出或 PWM1/2 使能时，T1L 与 PP1 不匹配

5.1.4 中断使能寄存器EIE2

寄存器5.4: EIE2: 外设中断使能寄存器2(地址:2DH)

	bit7					bit0		
复位值	-	C2IE	RXIE	TXIE	-	-	-	-
-000----	U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0

C2IE: 比较器 2 中断允许位

1 = 允许比较器 2 中断

0 = 禁止比较器 2 中断

RXIE: USART 接收中断允许位

1 = 允许 USART 接收中断

0 = 禁止 USART 接收中断

TXIE: USART 发送中断允许位

1 = 允许 USART 发送中断

0 = 禁止 USART 发送中断

5.1.5 中断标志寄存器EIF2

寄存器5.5: EIF2: 外设中断标志寄存器2(地址:0DH)

	bit7					bit0		
复位值	-	C2IF	RXIF	TXIF	-	-	-	-
-000----	U-0	R/W-0	R-0	R-0	U-0	U-0	U-0	U-0

C2IF: 比较器 2 中断标志位

1 = 比较器 2 输出发生改变(必须用软件清 0)

0 = 比较器 2 输出未发生改变

RXIF: USART 接收中断标志位

1 = USART 接收缓冲器满(通过读 RXSDR 清 0)

0 = USART 接收缓冲器空

TXIF: USART 发送中断标志位

0 = USART 发送缓冲器满

1 = USART 发送缓冲器空(通过写 TXSDR 清 0)

5.1.6 中断响应

当中断被响应后:

1. AIE 位将被硬件清 0 以禁止其它中断;
2. 返回地址压入堆栈;
3. 中断入口地址载入 PC;
4. 执行该中断服务子程序;
5. 执行指令 IRET 退出中断服务子程序并将 AIE 置 1, 重新使能未屏蔽的中断;

6. 跳转到中断发生处继续执行下面的程序。

进入中断服务程序后，首先保存 PSW 和其它寄存器的值，然后通过查询中断标志位确定中断源。在重新使能中断之前，应在软件中将相应的中断标志位清 0，以避免出错。

- 注 1：中断条件满足时，无论相应的中断使能位或AIE位的状态为何，中断标志位都将被置1。

2：当执行一条清除AIE位的指令后，任何在下一周期等待响应的中断都将被忽略。当AIE位重新置1时，被忽略的中断请求将继续等待被响应。

3：当对中断进行响应，进入中断服务子程序的时候硬件会将AIE位清零关闭总中断，当中断程序执行完，中断返回指令跳出中断子程序时，硬件将AIE位置1打开总中断。

5.2 INT中断

INT 中断有三个中断源: INT0、INT1 和 INT2, 都采用边沿触发方式,如果触发边沿选择位(INTxSE)置 1, 则采用上升沿触发; 如果触发边沿选择位清 0, 则采用下降沿触发。

表 5.2 与 INT 中断有关的寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
0BH	INTCTL	AIE	PUIE	TOIE	INT0IE	POIE	TOIF	INT0IF	POIF
2CH	EIE1	-	ADIE	INT2IE	INT1IE	C1IE	PWM2IE	T2IE	T1IE
0CH	EIF1	-	ADIF	INT2IF	INT1IF	C1IF	PWM2IF	T2IF	T1IF
15H	PWMCTL	INT2SE	INT1SE	-	-	-	-	PWM2ON	PWM1ON
21H	OPTR	PUPH	INT0SE	T0CS	T0SE	PSA	PS2	PS1	PS0

5.2.1 INT0 中断

INT0 中断通过寄存器 INTCTL 中的 INT0IE 位置 1 使能 INT0 中断。通过 OPTR 中的 INT0SE 位设置触发边沿, INT0SE 置 1, 将 INT0 设置为上升沿触发, 清 0 设置为下降沿触发。INTCTL 中的 INT0IF 为 INT0 的中断标志位。

INT0 引脚有触发脉冲时, INT0IF 被自动置 1, 如果 INT0IE 和 AIE 位为 1, 则响应 INT0 中断。

5.2.2 INT1 中断

INT1 中断通过寄存器 EIE1 中的 INT1IE 位置 1 使能 INT1 中断。通过 PWMCTL 中的 INT1SE 位设置触发边沿, INT1SE 置 1, 将 INT1 设置为上升沿触发, 清 0 设置为下降沿触发。EIF1 中的 INT1IF 为 INT1 的中断标志位。

INT1 引脚有触发脉冲时, INT1IF 被自动置 1, 如果 INT1IE、PUIE 和 AIE 位为 1, 则响应 INT1 中断。

5.2.3 INT2 中断

INT2 中断通过寄存器 EIE1 中的 INT2IE 位置 1 使能 INT2 中断。通过 PWMCTL 中的 INT2SE 位设置触发边沿, INT2SE 置 1, 将 INT2 设置为上升沿触发, 清 0 设置为下降沿触发。EIF1 中的 INT2IF 为 INT2 的中断标志位。

INT2 引脚有触发脉冲时, INT2IF 被自动置 1, 如果 INT2IE、PUIE 和 AIE 位为 1, 则响应 INT2 中断。

使用 INT 中断时的设置:

1. 将对应的 INTx 引脚设置为数字输入口。
2. 选择触发脉冲边沿时上升沿还是下降沿(INT0/1/2SE 置 1 为上升沿触发);
3. 将相应的外部中断使能位置 1(INTxIE);

5.3 定时器中断

T0/1 寄存器发生溢出时，T0IF/T1IF 位将会被置 1。通过将 T0IE/T1IE 位置 1/清 0 可使能/禁止该中断。有关定时/计数器模块的操作，请参考定时/计数器部分。

当 T2 与 PP3 匹配时，T2IF 将被置 1。

5.4 P0 口中断

P0 口引脚的输入电平变化将使 P0IF(INTCTL.0)位置 1。通过设置/清除 P0IE(INTCTL.3)位，可使能/禁止该中断。且该端口各引脚可通过 IOCL 寄存器来对每个引脚进行配置。有关 P0 口的操作，请参考 P0 口部分。

5.5 PWM中断

使能 PWM1/2 后，T1L 分配给 PWM1 进行计数，T1H 分配给 PWM2 进行计数，当 T1L/H 与 PP1/2 匹配时，会触发相应的中断标志位 T1IF 和 PWM2IF。如果使能 T1IE 或者 PWM2IE，则会触发中断（AIE、PUIE 置 1）。

使能 PWM3 后，当 T2 和 PP3 匹配时，会触发相应的中断标志 T2IF。如果 T2IE 使能，则会触发中断（AIE、PUIE 置 1）。详见 PWM 部分。

5.6 模拟比较器中断

当比较器 1 输出变化引起寄存器 C1OUT 变化时，将产生失配条件，C1OUT 和失配信号不相等，产生比较器 CMP1 的中断标志 C1IF 置 1。失配条件将持续到读 CMCTL0 寄存器或比较器输出返回到先前状态为止。如果 CMP1 的中断使能，且 AIE(全局中断允许位)和 PUIE(外设中断允许位)置 1，程序将相应中断。

当比较器 2 输出寄存器 C2OUT 为 1 时，比较器 CMP2 的中断标志 C2IF 将一直保持为 1。如果 CMP2 的中断使能，且 AIE(全局中断允许位)和 PUIE(外设中断允许位)置 1，程序将相应中断。

5.7 USART中断

通用串行通讯模块 USART 的中断分为接收中断和发送中断。
详见通用串行通讯模块部分

5.8 中断现场保护

在中断响应时，硬件会把当前 PC 值加 1 入栈保存，中断结束后，硬件在将本次中断入栈时的值弹出载入 PC，继续执行后面的程序。通常，用户可能希望在中断时对一些关键寄存器的内容进行保存(例如，Rn 和 PSW)。这些都需通过软件方式实现。

6 定时/计数器

KF8F312M4 单片机提供一个 8 位的定时/计数器 T0、1 个 16 位的定时/计数器 T1 和 1 个 8 位定时器 T2。

6.1 定时/计数器 0(T0)

T0 是一个 8 位的定时/计数器，当 T0 寄存器值加到 255 时，再加 1，则会产生溢出，T0 寄存器的值返回到 0 开始重新计数。

表 6.1 与 T0 相关的寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
0BH	INTCTL	AIE	PUIE	TOIE	INT0IE	POIE	TOIF	INT0IF	POIF
01H	T0	8 位计数器							
21H	OPTR	PUPH	INT0SE	TOCS	TOSE	PSA	PS2	PS1	PS0
25H	TR0	-	-	TR05	TR04	TR03	TR02	TR01	TR00

6.1.1 OPTR选择寄存器

如寄存器 6.1 所示，T0CS 为定时/计数模式选择位，T0SE 为计数模式时外部触发脉冲边沿选择位，PSA 用来将预分频器分配给 WDT 或者 T0,PS<2:0>对分频比分配。

寄存器6.1: OPTR: 选择寄存器(地址: 21H)

复位值	bit7				bit0			
1111 1111	PUPH	INT0SE	TOCS	TOSE	PSA	PS2	PS1	PS0
	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1

PUPH: P0 口上拉功能总使能位

$\overline{\text{PUPH}}=1$ 禁止 P0 口所有上拉功能

$\overline{\text{PUPH}}=0$ 允许 P0 口使用上拉功能

INT0SE: INT0 中断触发脉冲边沿选择位

INT0SE=1 INT0/P0.2 为上升沿触发

INT0SE=0 INT0/P0.2 为下降沿触发

TOCS: T0 模式选择位

TOCS=1 计数模式，T0 的时钟为外部时钟 T0CK/P0.2

TOCS=0 定时模式，T0 的时钟为机器内部时钟 FOSC/4

TOSE: T0 计数脉冲信号边沿选择位

TOSE=1 下降沿触发

TOSE=0 上升沿触发

PSA: 预分频器分配控制位

PSA=1 预分频器用于 WDT

PSA=0 预分频器用于 T0

PS<2:0>: 预分频器分频比选择位

PS<2:0>	WDT 分频比	T0 分频比
000	1 : 1	1 : 2
001	1 : 2	1 : 4
010	1 : 4	1 : 8
011	1 : 8	1 : 16
100	1 : 16	1 : 32
101	1 : 32	1 : 64
110	1 : 64	1 : 128
111	1 : 128	1 : 256

6.1.2 预分频器

图 6.1 为预分频器的结构框图。T0 模块使用一个 8 位计数器作为预分频器。如寄存器 6.1 所示，通过软件设定 PSA 位(OPTR.3)的状态可对预分频器的分配进行控制，PSA 位清 0 可将预分频器分配给 T0 模块。通过设置 PS<2:0>位可选择预分频器的分频比。预分频器是不可读写的。当预分频器用于 T0 模块时，所有写入 T0 寄存器的指令都会将预分频器清 0。当预分频器用于 WDT 时，CWDT 指令会同时将预分频器和看门狗定时器清 0。

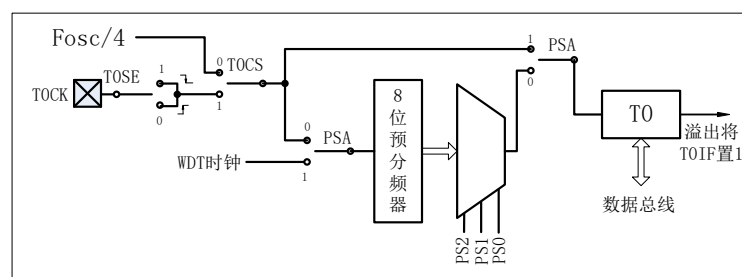


图 6.1 预分频器框图

6.1.3 定时模式

通过将 T0CS 位(OPTR.5)清 0 可选择定时器模式。在定时模式中，如果不使用预分频器，每一个机器周期 T0 寄存器的值加 1。如果 T0 寄存器被写入初始值，则在接下来的两个机器周期将不执行递增操作，用户可通过将校正值写入 T0 寄存器进行修正。

6.1.4 计数模式

通过将 T0CS 位(OPTR.5)置 1 可选择计数模式。在该模式下，T0 模块在 T0CK 引脚信号的每一次上升沿(T0SE 位清 0)或下降沿(T0SE 位置 1)递增计数。

当不使用预分频器时，在内部相位时钟的 Q2 和 Q4 周期对 T0CK 引脚电平进行采样，可以实现 T0CK 与内部相位时钟的同步。因此，要求 T0CK 的高电平状态和低电平状态分别保持至少 2Tosc 的时间。

6.1.5 T0 的使用

T0 在使用时通过以下步骤进行设置:

1. 通过将 T0CS 位清 0/置 1 选择定时/计数模式(如果是计数模式,再设置 T0SE 选择脉冲触发边沿, 将对应的计数脉冲输入脚 T0CK 设置为输入);
2. 如果需要分频, 则将预分频器分配给 T0, 并设置分频比;
3. 给 T0 寄存器设置初始值;
4. 如果使用中断方式则将 T0IE 和 AIE 位置 1。

6.2 定时/计数器 1(T1)

T1 是一个 16 位的定时/计数器，T1 的低 8 位在寄存器 T1L 中，高 8 位在寄存器 T1H 中，当 T1 计数值达到 65535 后，T1 的值再加 1 就会产生溢出，将 T1 中断标志位置 1。T1 属于外部单元，因此在使用 T1 中断时，需将 PUIE 位置 1，使能外设中断。如图 6.2 所示为 T1 的原理框图。

表 6.2 与 T1 相关的寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
0BH	INTCTL	AIE	PUIE	TOIE	INT0IE	POIE	TOIF	INT0IF	POIF
0EH	T1L	T1 低 8 位							
0FH	T1H	T1 高 8 位							
0CH	EIF1	-	ADIF	INT2IF	INT1IF	C1IF	PWM2IF	T2IF	T1IF
2CH	EIE1	-	ADIE	INT2IE	INT1IE	C1IE	PWM2IE	T2IE	T1IE
10H	T1CTL	-	T1GC	T1CKS1	T1CKS0	-	$\overline{T1SY}$	T1CS	T1ON
25H	TR0	-	-	TR05	TR04	TR03	TR02	TR01	TR00

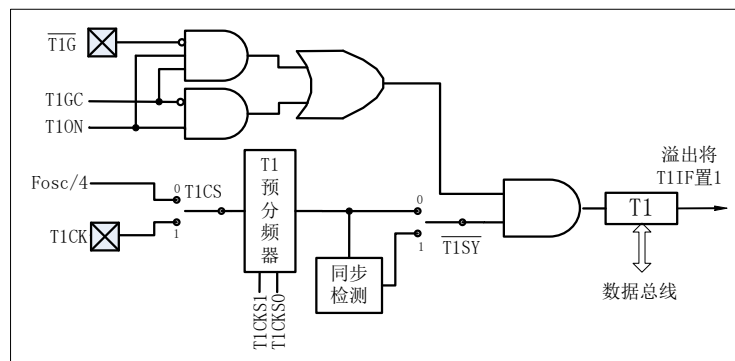


图 6.2 T1 原理框图

6.2.1 T1 控制寄存器

如寄存器 6.2 所示，T1 控制寄存器（T1CTL）用于启动/禁止 T1 以及选择 T1 模块的不同功能特性。

寄存器 6.2: T1CTL: T1 控制寄存器 (地址: 10H)

	bit7						bit0	
复位值	-	T1GC	T1CKS1	T1CKS0	-	$\overline{T1SY}$	T1CS	T1ON
-000 -000								
	U	R/W	R/W	R/W	U	R/W	R/W	R/W

T1GC: T1 门控使能位

如果 T1ON=0 则该位被忽略

如果 T1ON=1 则:

T1GC = 1 使能 $\overline{T1G}$ 引脚控制(如果 $\overline{T1G}$ 引脚为低电平，启动 T1，为高电平，关闭 T1)

T1GC = 0 禁止 $\overline{T1G}$ 引脚控制

T1CKS<1:0>: T1 输入时钟预分频比选择位

T1CKS<1:0>=11 1/8 倍预分频比

T1CKS<1:0>=10 1/4 倍预分频比

T1CKS<1:0>=01 1/2 倍预分频比

T1CKS<1:0>=00 1/1 倍预分频比

$\overline{T1SY}$: T1 计数模式外部触发脉冲输入同步控制位

T1CS=1:

$\overline{T1SY}$ =1 不与外部触发脉冲输入同步

$\overline{T1SY}$ =0 与外部触发脉冲输入同步

T1CS=0: 该位被忽略, T1 使用内部时钟

T1CS: T1 定时/计数模式选择

T1CS=1 计数模式, T1 时钟为外部时钟 T1CK/P0.5

T1CS=0 定时模式, T1 时钟为内部时钟 FOSC/4

T1ON: T1 启动控制位

T1ON=1 启动 T1

T1ON=0 停止 T1

6.2.2 T1 预分频器

如寄存器 6.2 所示, T1 具有四个预分频器选择项, 允许对时钟输入进行 1、2、4、或 8 倍分频。T1CKS 位(T1CTL<5:4>)对预分频计数器进行控制。T1 预分频计数器不能直接进行读写操作, 可通过写入 T1H 或 T1L 使预分频计数器清 0。

6.2.3 定时模式

通过将 T1CS 位清 0 将 T1 设定为定时模式, T1 工作在定时模式时, 对单片机内部时钟进行计数, 当不使用预分频器时, 每个机器周期 T1 寄存器自加 1, 加到 0FFFFH 后再加 1, T1 溢出, 将 T1 中断标志位 T1IF 置 1。

如果使能 T1 门控引脚, 且 T1ON=1, 则在 $\overline{T1G}$ 引脚为低电平时, 启动 T1, 如果 $\overline{T1G}$ 引脚为高电平, 禁止 T1。使用该方式可粗略的对 $\overline{T1G}$ 引脚的低电平持续时间进行计算。

6.2.4 计数模式

通过将 T1CS 位置 1 将 T1 设定为计数模式, 在计数模式时, T1 在计数脉冲 T1CK 的上升沿进行递增计数。T1 的计数模式又有同步计数和异步计数两种方式,

如果控制位 $\overline{T1SY}$ (T1CTL.2)置 1, 则 T1 工作在异步计数模式。计数器根据 T1CK 引脚的脉冲进行递增计数。在休眠模式下, 计数器将继续递增并在溢出时产生中断以唤醒处理器。

如果控制位 $\overline{T1SY}$ (T1CTL.2)清 0, 则 T1 工作在同步计数模式。在内部相位时钟的 Q2 和 Q4 周期对 T1CK 引脚电平进行采样, 可以实现 T1CK 与内部相位时钟的同步。

6.2.5 T1 在休眠模式下的运行

只有设定在异步计数器模式时, T1 才能在休眠模式下工作。在该模式下, 计数脉冲 T1CK 使计数器递增。通过如下步骤设定定时器以唤醒器件:

- 使能 T1(T1ON/T1CTL.0 置 1)
- 将 T1IE 位(EIE1.0)置 1
- 将 PUIE 位(INTCTL.6)置 1

器件将在溢出时被唤醒。如果 AIE 位(INTCTL.7)置 1, 器件将被唤醒并跳转至中断服务程序。

6.2.6 T1 分配给PWM1/2

当使用 PWM1/2 时需要用到 T1, 单片机将 T1L, T1IE, T1IF 分配给 PWM1, T1H 分配给 PWM2, 具体使用方法参见 PWM1/2 部分。

6.3 定时器T2

T2 与 T0/1 有所不同，T2 是一个 8 位的定时器，没有外部计数时钟输入脚，T2 的原理框图如图 6.3 所示。

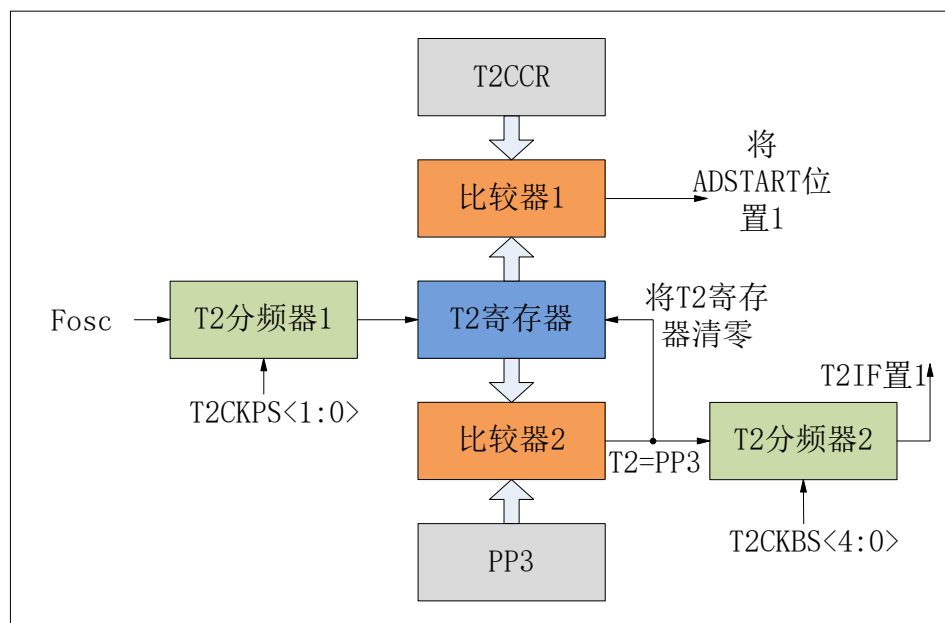


图 6.3 定时器 T2 原理框图

6.3.1 T2 相关的寄存器

表 6.3 与 T2 相关的寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
0BH	INTCTL	AIE	PUIE	TOIE	INT0IE	P0IE	T0IF	INT0IF	P0IF
0CH	EIF1	-	ADIF	INT1IF	INT1IF	C1IF	PWM2IF	T2IF	T1IF
2CH	EIE1	-	ADIE	INT1IE	INT1IE	C1IE	PWM2IE	T2IE	T1IE
12H	T2CTL	-	T2CKBS3	T2CKBS2	T2CKBS1	T2CKBS0	T2ON	T2CKPS1	T2CKPS0
11H	T2	T2 寄存器							
52H	PP3	PWM3 周期寄存器							
54H	T2CCR	T2 触发 AD 启动寄存器							

6.3.1.1 T2 控制寄存器T2CTL

寄存器6.3: T2CTL: T2控制寄存器(地址: 12H)

复位值	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
-000 0000	-	T2CKBS3	T2CKBS2	T2CKBS1	T2CKBS0	T2ON	T2CKPS1	T2CKPS0
	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W

T2CKBS<3:0>: T2 分频器 2 分频比选择位
T2CKBS<3:0>=0000: 分频比为 1:1

	T2CKBS<3:0>=0001: 分频比为 1:2
	T2CKBS<3:0>=0010: 分频比为 1:3
	T2CKBS<3:0>=0011: 分频比为 1:4
	T2CKBS<3:0>=0100: 分频比为 1:5
	T2CKBS<3:0>=0101: 分频比为 1:6
	T2CKBS<3:0>=0110: 分频比为 1:7
	T2CKBS<3:0>=0111: 分频比为 1:8
	T2CKBS<3:0>=1000: 分频比为 1:9
	T2CKBS<3:0>=1001: 分频比为 1:10
	T2CKBS<3:0>=1010: 分频比为 1:11
	T2CKBS<3:0>=1011: 分频比为 1:12
	T2CKBS<3:0>=1100: 分频比为 1:13
	T2CKBS<3:0>=1101: 分频比为 1:14
	T2CKBS<3:0>=1110: 分频比为 1:15
	T2CKBS<3:0>=1111: 分频比为 1:16
T2ON:	T2 使能位
	T2ON=1: 使能 T2
	T2ON=0: 禁止 T2
T2CKPS<1:0>:	T2 分频器 1 分频比选择位
	T2CKPS<1:0>=00: 分频比为 1:1
	T2CKPS<1:0>=01: 分频比为 1:4
	T2CKPS<1:0>=1x: 分频比为 1:16

6.3.2 T2 的工作原理

通过 T2CTL 的 T2ON 启动/禁止 T2。如图 6.3 所示，为 T2 的原理框图，T2 的计数时钟为系统时钟，每个机器周期(分频器 1 分频比为 1:1 时)寄存器 T2 的值自动加 1。

Fosc/4 先经过分频器 1 分频后送到 T2 寄存器，进行加 1 计数，分频器 1 通过设置寄存器 T2CTL 中的 T2CKPS<1:0>位，可将 Fosc/4 进行 1:1、1:4 和 1:16 分频。

当 T2 寄存器与 PP3 相等时，T2 自动清 0，发出相等信号给分频器 2，分频器 2 递增。同时 T2 的值与寄存器 T2CCR 相比较，如果等于 T2CCR，且对应的控制位置 1，则会发出信号，使 ADCCTL0 的 START 位置 1。

分频器 2 的分频比可通过寄存器 T2CTL 中的 T2CKBS<3:0>位设置为 1:1 至 1:16。当其设置为 1:1 时，每次寄存器 T2 与 PP3 相等，将会使 T2 中断标志位 T2IF 置 1；当其设置为 1:2 时，寄存器 T2 与 PP3 相等累计两次才会使 T2IF 置 1，以此类推。

T2 的分频器 1 的分频比由寄存器 T2CTL 的 T2CKPS<1:0>位设置。分频器 2 由寄存器 T2CTL 的 T2CKBS<3:0>位设置。如果对 T2 寄存器执行写操作、对寄存器 T2CTL 执行写操作或发生任何器件复位事件(上电复位、RST 复位、看门狗复位或欠压复位)，分频器 1/2 的计数器将被清 0。

寄存器 T2、PP3 和 T2CCR 均可读写。任何复位时，寄存器 T2 均被设置为 00H，T2CCR 被设置为 XXH，PP3 被设置为 0FFH。

6.3.3 T2 中断

如图 6.3 所示，预分频器 2 的计数器达到设定值后将会使中断标志位置 1，如果使能位 T2IE 为 1，且全局中断和外设中断允许位为 1，将会相应 T2 中断。

6.3.4 T2 在休眠模式

单片机进入休眠模式后，因 T2 使用系统时钟，因此 T2 将停止工作。

6.3.5 T2 分配给PWM3

PWM3 在使用时，需要使用 T2 进行定时，PWM3 的详细内容请参考 PWM3 部分。

7 模数(A/D)转换模块

模数(A/D)转换模块可将模拟输入信号转换为十位二进制值。KF8F312M4 拥有 12 路模拟输入通道，转换器通过逐次逼近法将模拟输入信号转换为二进制值，并将转换结果存放到 10 位寄存器中。可通过软件方式选择 VDD 或施加在 ADVRIN 引脚上的电压作为转换使用的参考电压。图 7.1 显示了 KF8F312M4 中 A/D 转换模块的结构框图。

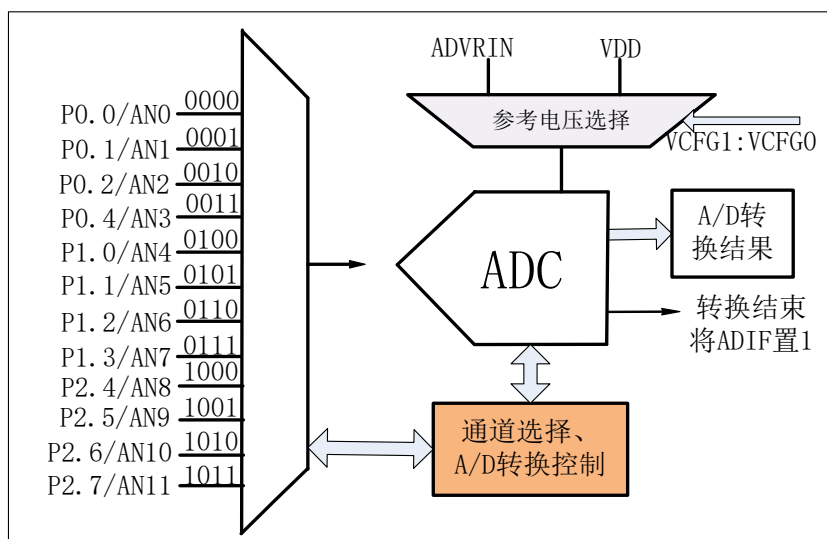


图 7.1 AD 模块结构框图

7.1 与AD相关的寄存器

表 7.1 与 AD 转换相关的寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
0BH	INTCTL	AIE	PUIE	T1IE	INT0IE	P0IE	T0IF	INT0IF	P0IF
2CH	EIE1	-	ADIE	INT2IE	INT1IE	C1IE	PWM2IE	T2IE	T1IE
0CH	EIF1	-	ADIF	INT2IF	INT1IF	C1IF	PWM2IF	T2IF	T1IF
1FH	ADCCTL0	ADLR	T2CCRON	CHS3	CHS2	CHS1	CHS0	START	ADEN
3FH	ADCCTL1	-	ADCS2	ADCS1	ADCS0	VCFG1	VCFG0	-	-
31H	ANSEL	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0
1DH	ANSEH	-	-	-	-	ANS11	ANS10	ANS9	ANS8
1EH	ADCDATAH	AD 高 8 位							
3EH	ADCDATAH	AD 低 8 位							

7.1.1 AD控制寄存器 0(ADCCTL0)

如寄存器 7.1 所示，AD 控制寄存器 0 包括：转换结果输出格式选择、模拟通道选择和 AD 启动。

寄存器 7.1: ADCCTL0: A/D控制寄存器0(地址: 1FH)

bit7								bit0	
复位值	ADLR	T2CCRON	CHS3	CHS2	CHS1	CHS0	START	ADEN	
0000 0000	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

ADLR: A/D 转换结果输出格式选择位

ADLR=1 结果右对齐

ADLR=0 结果左对齐

T2CCRON: T2 触发 AD 启动使能位

T2CCRON=1 使能 T2 触发 AD 启动

T2CCRON=0 禁止 T2 触发 AD 启动

CHS<3:0>: 模拟通道选择位

CHS<3:0>=0000 通道 00(AN0)

CHS<3:0>=0001 通道 01(AN1)

CHS<3:0>=0010 通道 02(AN2)

CHS<3:0>=0011 通道 03(AN3)

CHS<3:0>=0100 通道 04(AN4)

CHS<3:0>=0101 通道 05(AN5)

CHS<3:0>=0110 通道 06(AN6)

CHS<3:0>=0111 通道 07(AN7)

CHS<3:0>=1000 通道 08(AN8)

CHS<3:0>=1001 通道 09(AN9)

CHS<3:0>=1010 通道 010(AN10)

CHS<3:0>=1011 通道 011(AN11)

START: A/D 转换状态位 S

START=1 A/D 转换正在进行,该位置 1 将启动 A/D 转换,在转换结束后该位将被硬件自动清 0。

START=0 A/D 转换结束或者未进行

ADEN: A/D 转换使能位

ADEN=1 A/D 转换模块正在工作

ADEN=0 A/D 转换器关闭且不消耗工作电流

7.1.2 AD控制寄存器 1(ADCCTL1)

如寄存器 7.2 所示，AD 控制寄存器 1 包含 AD 时钟选择和参考电压选择位。

寄存器7.2: ADCCTL1: A/D控制寄存器1(地址: 3FH)

复位值	bit7						bit0
-000 00--	-	ADCS2	ADCS1	ADCS0	VCFG1	VCFG0	-
	U	R/W	R/W	R/W	R/W	R/W	U

ADCS<2:0>: A/D 转换时钟选择位

ADCS<2:0>=000 $F_{ad}=F_{osc}/2$

ADCS<2:0>=001 $F_{ad}=F_{osc}/8$

ADCS<2:0>=010 $F_{ad}=F_{osc}/32$

ADCS<2:0>=x11 F_{ad} =内部专用 500Khz 时钟

ADCS<2:0>=100 $F_{ad}=F_{osc}/4$

ADCS<2:0>=101 $F_{ad}=F_{osc}/16$

ADCS<2:0>=110 $F_{ad}=F_{osc}/64$

VCFG<1:0>: A/D 转换参考电压选择位

VCFG<1:0>=0X VDD 作为 ADC 参考电压

VCFG<1:0>=10 ADVRIN 作为 ADC 参考电压

7.1.3 模拟/数字口选择寄存器(ANSEL、ANSEH)

如寄存器 7.3 所示，ANSEL 和 ANSEH 寄存器用于将 A/D 转换输入引脚设置为模拟口，通过将 ANSEL 和 ANSEH 某位置 1，将对应的引脚设置为模拟口，清 0 设置为数字 I/O 口。

寄存器7.3: ANSEL: 模拟/数字口设置寄存器(地址: 31H)

复位值	bit7						bit0
0000 0000	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ANS<7:0>: 引脚 AN7~AN0 分别配置为模拟或数字 I/O 口的控制位

ANSx=1 将对应引脚配置为模拟口

ANSx=0 将对应引脚配置为数字 I/O 口或者特殊功能引脚

寄存器7. 4: ANSEH: 模拟/数字口设置寄存器(地址: 1DH)

	bit7							bit0
复位值	-	-	-	-	ANS11	ANS10	ANS9	ANS8
---- 0000	U	U	U	U	R/W	R/W	R/W	R/W

ANS<11:8>: 引脚 AN11~AN8 分别配置为模拟或数字 I/O 口的控制位

ANSx=1 将对应引脚配置为模拟口

ANSx=0 将对应引脚配置为数字 I/O 口

7.2 通道的选择

如图 7.1 所示，KF8F312M4 中的 A/D 转换模块的输入可以选择 12 路，12 路全部来自外部的模拟信号。通过寄存器 ADCCTL0(如寄存器 7.1 所示)进行通道的选择。

7.3 模拟输入口的配置

当选择 AN0~AN11 作为 A/D 转换的输入时，需要将对应的引脚配置为模拟输入口。通过将寄存器 ANSEL 或 ANSEH 的某位置 1 将对应的引脚配置为模拟口，然后把寄存器 TR0(或 TR1)的对应位置 1 把该引脚配置为输入口，此时该引脚被设置为模拟输入口。

注：如果某引脚被配置为模拟输入口，将会自动禁止有效地数字I/O、上拉电阻和电平变化中断。

7.4 转换时钟的选择

完成一次 A/D 转换所需要的时间为 $11T_{ad}$ 。如寄存器 7.2 所示，可通过软件方式设置 ADCS 位(ADCCTL1<6:4>)选择转换时钟源，共有 7 种时钟选项。 T_{ad} 和 F_{ad} 分别为 A/D 转换时钟周期和频率。

为保证 A/D 转换的正确进行，所选择的 A/D 转换时钟周期(T_{ad})典型值应在 1 μ s 左右。

7.5 输出格式

KF8F312M4 中 A/D 转换的结果为 10 位二进制数，A/D 转换结果寄存器为两个 8 位的寄存器。用户可以通过 ADLR(ADCCTL0.7)设置转换结果输出格式，ADLR 置 1 输出为右对齐，ADLR 清 0 输出为左对齐。如图 7.2 所示。

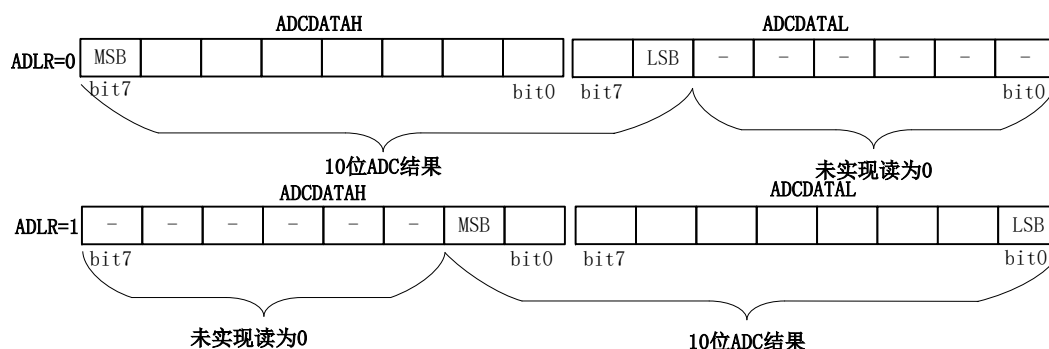


图 7.2 ADC 结构对其方式

7.6 A/D转换的启动和完成

先将 ADEN 位置 1，然后将 START 位(ADCCR0.1)置 1 即可启动 A/D 转换。当转换结束时，A/D 模块将：

1. 将 START 位清 0
2. 将 ADIF 位置 1
3. 如果使能 A/D 转换中断，则响应中断

可以采取在程序中将 START 位清 0 的方法中止当前的转换操作。在 A/D 转换采样全部结束之前，ADCDATAH:ADCDATAL 寄存器中的内容将不会被更新，而是仍旧保留前一次的转换结果。A/D 转换被中止后，需至少等待 2Tad 的延时时间后才能开始下一次数据采集。

7.7 A/D 工作在休眠模式

A/D 转换器模块可以在休眠模式下工作。这需把 AD 的时钟源设定为 A/D 专用的内部振荡器。当选择了专用内部时钟源后，A/D 需等待一个指令周期后才能启动转换操作，转换结束后，START 位将被清 0，且转换结果将被载入 ADCDATAH:ADCDATAL 寄存器。如果 A/D 中断被使能，器件将从休眠状态唤醒。如果 A/D 中断被禁止，A/D 转换模块在转换完成后被关闭。

如果 A/D 时钟源为不是 AD 专用内部振荡器，执行 IDLE 指令将导致当前转换操作中止，并使 A/D 模块关闭。

7.8 复位的影响

器件复位将强制所有寄存器进入复位状态。因此，A/D 模块将被关闭，任何进行中的转换操作被中止。ADCDATAH:ADCDATAL 寄存器中的值不变。

7.9 使用A/D转换器的设置

启动 A/D 转换器时的设置：

1. 选择 A/D 采样输入通道，设置 A/D 转换结果对齐方式；
2. 将对应的 A/D 采样输入通道设置为模拟输入模式；
3. 选择参考电压和 A/D 采样时钟频率，打开 A/D 转换；
4. 如果采用中断方式，使能 A/D 转换中断；
5. 等待 AD 所需的采集时间；
6. 启动 A/D 转换；
7. 查询 AD 是否转换完成(START=0)或进入 AD 中断；
8. 读取 AD 转换结果。

8 PWM模块

KF8F312M4 单片机具有 2 路 8 位的 PWM 模块 PWM1/PWM2 和 1 路 10 位增强型 PWM3 模块。其中 PWM1 和 PWM2 结构相同。PWM3 模块有 4 个输出引脚、带有死区延时、自动关断、自动重启等功能。

8.1 PWM1/2 模块

启动 PWM 后，在对应的 PWM1(或 PWM2)引脚输出 PWM 脉冲。PWM 脉冲的频率和占空比通过 PP1(或 PP2)和 PWM1L(或 PWM2L)设置。

图 8.1 显示了 PWM 逻辑框图。其中 PP1 为 PWM1 模块的周期寄存器，PWM1L 为 PWM1 模块占空比设置寄存器,使用 PWM 时需要将定时器 1 配置给 PWM 做定时用，其中 T1L、T1IE 和 T1IF 分配给 PWM1,T1H 分配给 PWM2。启动 PWM1 后，当 T1L 计数值和 PP1 相等时，P1.0 引脚被置 1，此时 T1L 被清 0，重新开始计数，当 T1L 的计数值和 PWM1L 相等时，P1.0 引脚清 0(如图 8.2 所示)。改变 PP1 和 PWM1L 的值可产生不同的 PWM1 周期和 PWM1 占空比。PWM2 模块的工作原理和 PWM1 模块完全一致。

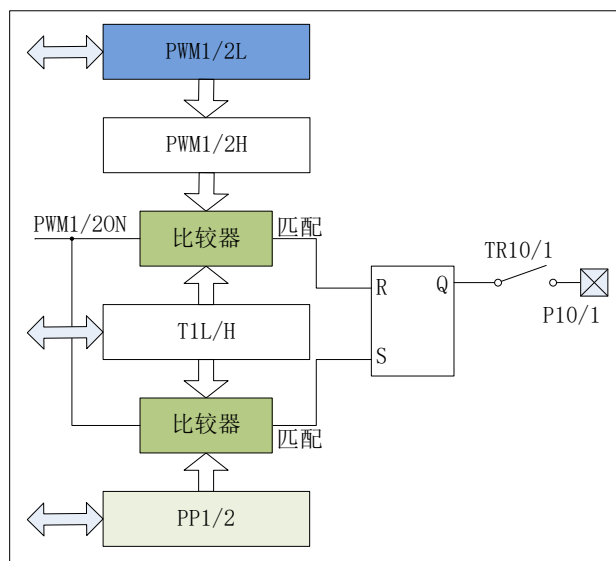


图 8.1 PWM1/2 逻辑框图

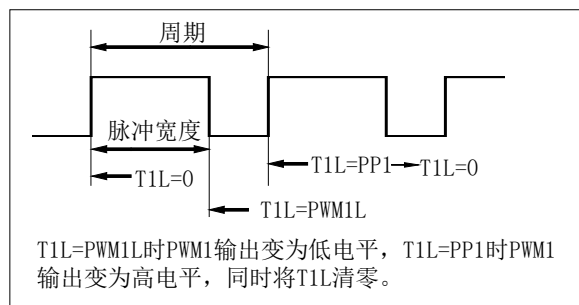


图 8.2 PWM1 输出波形图

8.1.1 PWM1/2 相关的寄存器

表 8.1 与 PWM 相关的寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
0BH	INTCTL	AIE	PUIE	T0IE	INT0IE	POIE	T0IF	INT0IF	POIF
2CH	EIE1	-	ADIE	INT2IE	INT1IE	C1IE	PWM2IE	T2IE	T1IE
0CH	EIF1	-	ADIF	INT2IF	INT1IF	C1IF	PWM2IF	T2IF	T1IF
0EH	T1L	定时器 1 低 8 位							
0FH	T1H	定时器 1 高 8 位							
13H	PWM1L	PWM1 占空比寄存器							
14H	PWM1H	PWM1 寄存器(与用户无关)							
16H	PP1	PWM1 周期寄存器							
32H	PP2	PWM2 周期寄存器							
33H	PWM2L	PWM2 占空比寄存器							
34H	PWM2H	PWM2 寄存器(与用户无关)							
10H	T1CTL	-	T1GC	T1CKS1	T1CKS0	-	$\overline{\text{T1SY}}$	T1CS	T1ON
15H	PWMCTL	INT2SE	INT1SE	-	-	-	-	PWM2ON	PWM1ON
27H	TR1	TR17	TR16	TR15	TR14	TR13	TR12	TR11	TR10
31H	ANSEL	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0

8.1.2 PWM控制寄存器

PWM 控制寄存器包括 PWM1/2 的启动控制位和 INT1/2 触发脉冲边沿选择。

寄存器8. 1: PWMCTL: PWM启动控制寄存器(地址: 15H)

复位值		bit7						bit0	
11-- --00		INT2SE	INT1SE	-	-	-	-	PWM2ON	PWM1ON
		R/W	R/W	U	U	U	U	R/W	R/W

INT2SE: INT2 触发脉冲边沿选择位

INT2SE=1 上升沿触发

INT2SE=0 下降沿触发

INT1SE: INT1 触发脉冲边沿选择位

INT1SE=1 上升沿触发

INT1SE=0 下降沿触发

PWM2ON: PWM2 启动控制位

PWM2ON=1 启动 PWM2

PWM2ON=0 禁止 PWM2

PWM1ON: PWM1 启动控制位

PWM1ON=1 启动 PWM1

PWM1ON=0 禁止 PWM1

8.1.3 PWM1/2 周期

PWM 周期通过 PP1/2 (地址: 16H/32H, 如寄存器 8.2 所示)进行设置,PP1/2 是一个 8 位的寄存器, 其值可设置为 0~255。PWM 周期通过式 8.1 进行计算。

$$\star \text{ 式8.1: } \text{PWM周期} = (\text{PP}_x + 1) \cdot 4 \cdot \text{Tosc} \cdot (\text{T1预分频比}) \quad (x=1、2)$$

寄存器8.2: PP1: PWM1周期控制寄存器(地址:16H)

bit7				bit0			
复位值 1111 1111	PP17	PP16	PP15	PP14	PP13	PP12	PP11
	R/W	R/W	R/W	R/W	R/W	R/W-1	R/W

PP2: PWM2周期控制寄存器(地址:32H)

bit7				bit0			
复位值 1111 1111	PP27	PP26	PP25	PP24	PP23	PP22	PP21
	R/W	R/W	R/W	R/W	R/W	R/W	R/W

8.1.4 PWM1/2 占空比

PWM 占空比通过 PWM1/2L(地址: 13H/33H, 如寄存器 8.3 所示)设置, 可写入一个 8 位的值到 PWM1/2L 来设置占空比。脉冲宽度和占空比通过式 8.2 和式 8.3 计算:

$$\star \text{ 式8.2: } \text{脉冲宽度} = \text{PWM}_x\text{L} \cdot 4 \cdot \text{Tosc} \cdot (\text{T1预分频比}) \quad (x=1、2)$$

$$\star \text{ 式8.3: } \text{占空比} = \frac{\text{脉冲宽度}}{\text{PWM周期}} = \frac{\text{PWM}_x\text{L}}{\text{PP}_x + 1} \quad (X=1、2)$$

寄存器8.3: PWM1L: PWM1占空比设置寄存器(地址: 13H)

bit7				bit0			
复位值 xxxx xxxx							
	R/W	R/W	R/W	R/W	R/W	R/W	R/W

PWM2L: PWM2占空比设置寄存器(地址: 33H)

bit7				bit0			
复位值 xxxx xxxx							
	R/W	R/W	R/W	R/W	R/W	R/W	R/W

8.1.5 PWM1/2 分辨率

分辨率决定在给定周期内的占空比数。例如, 10 位分辨率将产生 1024 个离散的占空比, 8 位分辨率产生 256 个离散的占空比。KF8F312M4 中当 PP1/2 为 255 时, PWM 的最大分辨率为 8 位。分辨率的计算公式如式 8.4 所示。

$$\star \text{ 式8.4: } \text{分辨率} = \frac{\text{Log}[(PP_x+1)]}{\log 2} \text{ 位} \quad (x=1、2)$$

8.1.6 PWM1/2 中断

PWM2 有一个专门的中断使能位 PWM2IE 和中断标志位 PWM2IF, 当启动 PWM2 后, 会对 PWM2IF 产生影响, 而 PWM1 和定时器 T1 共用中断使能位 T1IE 和中断标志位 T1IF。在 PWM1/2 启用后, 当 T1L/H 的计数值与 PWM1L/2L 的值匹配后, 其对应的输出引脚变为低电平。当 T1L/H 的计数值与 PP1/2 的值匹配后, 其对应的输出引脚变为高电平, 同时将 T1L/H 清 0, 将 T1IF/PWM2IF 置 1, 如果允许 T1 或 PWM2 中断, 将会转入对应的中断子程序中。

8.1.7 休眠模式下的操作

在休眠模式下, T1 寄存器将不会递增并且模块的状态将保持不变。PWM1/2 输出引脚电平保持不变(如果输出为高电平, 则保持高电平, 如果为低电平保持低电平)。当器件被唤醒时, T1 将从原来的状态继续工作。

8.1.8 系统时钟频率的改变

PWM1/2 的输出频率是通过 T1L/H 定时产生的, 而定时器的计数时钟为系统时钟。因此系统时钟频率发生任何改变都会使 PWM 频率发生变化。

8.1.9 复位的影响

任何复位都会将所有端口强制为输入模式, 并强制 PWM1/2 使用的寄存器进入其复位状态。

8.1.10 PWM1/2 使用方法

PWM1/2 工作的设置应按照以下步骤:

- 1、 将 TR10 或 TR11 置 1，禁止引脚 P1.0/PWM1 或 P1.1/PWM2 的输出驱动器。
- 2、 赋 PP1 或 PP2 寄存器的初值以设置 PWM1 或 PWM2 的 PWM 周期。
- 3、 赋 PWM1L 或 PWM2L 寄存器的初值以设置 PWM1 或 PWM2 的占空比。
- 4、 配置并启动定时器/计数器 T1:
 - 配置 T1CTL 寄存器的 T1CKS1 和 T1CKS0 以选择 T1 的预分频比；
 - 将 T1L/H 清 0；
 - 将 T1CTL 寄存器的 T1ON 位置 1 以启动 T1。
- 5、 将 PWMCTL 寄存器的 PWM1ON 或 PWM2ON 置 1 以启动 PWM1 或 PWM2。
- 6、 将 TR10 或 TR11 清 0 使能引脚 P1.0/PWM1 或 P1.1/PWM2 的输出驱动器。

8.2 PWM3

PWM3 与 PWM1/2 不同: 带有死区控制延时功能, 占空比设置寄存器为 10 位, PWM3 最多可在 4 个不同的引脚(P3A、P3B、P3C 和 P3D)输出 PWM 信号, 分辨率最高 10 位; PWM3 有 4 种输出模式: 单输出、半桥输出、全桥正向输出模式和全桥反向输出模式, 通过寄存器 PWM3CTL0 中的 P3M<1:0>位选择 4 种输出模式之一(如表 8.3), 通过 PWM3M<1:0>位设置各引脚的有效电平(可设置为高电平有效和低电平有效)。PWM3 的原理框图如图 8.3 所示。

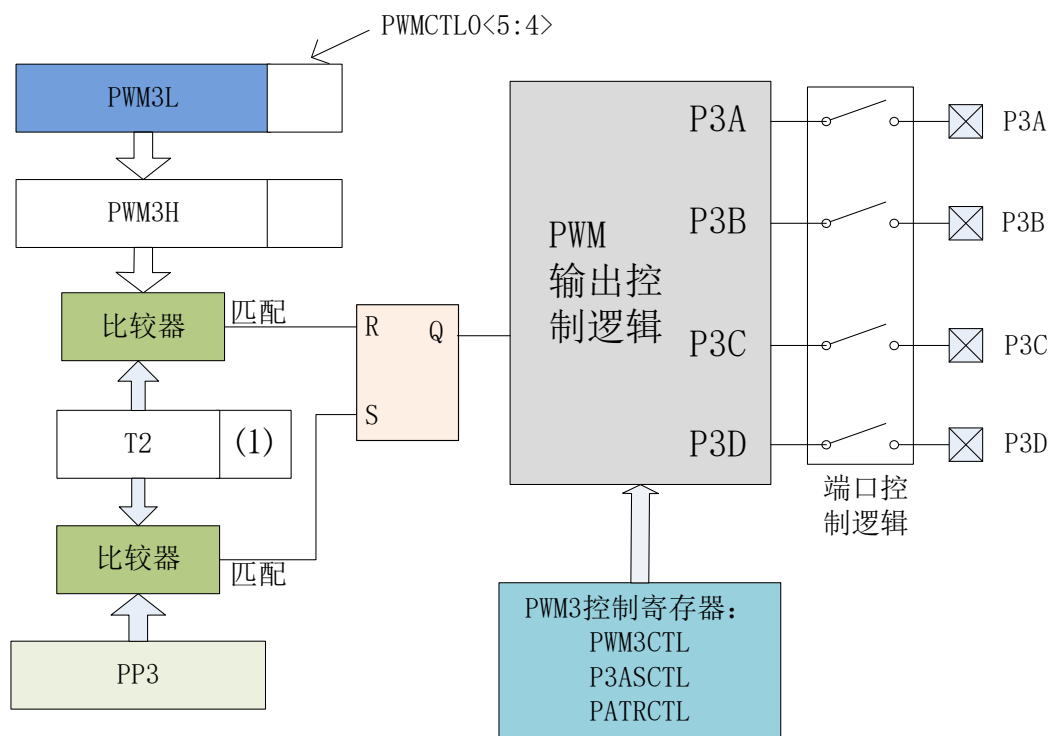


图 8.3 PWM3 原理框图

表 8.3 PWM3 输出模式控制及有效引脚

PWM3 输出模式	P3M<1:0>	有效引脚
单输出模式	00	默认将 P3A 配置为 PWM 输出, P3B、P3C 和 P3D 配置为端口引脚; 可通过 PATRCTL 寄存器中的 STR<D:A>各位置 1, 分别将 P3A、P3B、P3C 和 P3D 配置为 PWM 输出; PWM3 最多可在 4 个引脚输出 PWM 信号。
半桥输出模式	10	P3A 和 P3B 配置为调制输出; P3C 和 P3D 配置为端口引脚; 半桥输出模式带有死区控制功能
全桥正向输出模式	01	P3D 配置为 PWM 调制输出; P3A 为有效电平; P3B 和 P3C 为无效电平
全桥反向输出模式	11	P3B 配置为 PWM 调制输出; P3C 为有效电平, P3A 和 P3D 为无效电平

注: 各种输出模式的详细介绍请参考本节相应部分。

8.2.1 PWM3 相关寄存器

表 8.3 与 PWM3 相关的寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
0BH	INTCTL	AIE	PUIE	TOIE	INT0IE	POIE	TOIF	INT0IF	POIF
11H	T2	T1 低 8 位							
12H	T2CTL	-	T2CKBS3	T2CKBS2	T2CKBS1	T2CKBS0	T2ON	T2CKPS1	T2CKPS0
54H	T2CCR	T2 触发 AD 启动寄存器							
52H	PP3	PWM3 周期设置寄存器							
55H	PWM3L	PWM3 占空比设置寄存器							
56H	PWM3H	PWM3 寄存器							
57H	PWM3CTL0	P3M1	P3M0	PDT1	PDT0	P3ON1	P3ON0	PWM3M1	PWM3M0
5BH	PWM3CTL1	PRSEN	PDC6	PDC5	PDC4	PDC3	PDC2	PDC1	PDC0
5CH	P3ASCTL	P3ASE	P3ASS2	P3ASS1	P3ASS0	P3SSAC1	P3SSAC0	P3SSBD1	P3SSBD0
5DH	PATRCTL	-	-	-	STRSYNC	STREND	STRENC	STRENB	STRENA

8.2.2 PWM3CTL0 寄存器

寄存器 8.4: PWM3CTL0: PWM3 控制寄存器 0 (地址: 57H)

复位值	bit7						bit0	
0000 0000	P3M1	P3M0	PDT1	PDT0	P3ON1	P3ON0	PWM3M1	PWM3M0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

P3M<1:0>: PWM3 输出配置位

如果 P3ON<1:0>=00、01 或 10, P3A、P3B、P3C 和 P3D 为端口引脚

如果 P3ON<1:0>=11:

P3M<1:0>=00: 单输出模式; 默认 P3A 配置为 PWM 输出, P3B、P3C 和 P3D 为端口引脚; 可通过 PATRCTL 寄存器中的 STR<A:D>各位置 1, 分别将 P3A、P3B、P3C 和 P3D 配置为 PWM 输出; PWM3 最多可提供 4 路 PWM 输出

P3M<1:0>=01: 全桥正向输出模式; P3D 配置为 PWM 调制输出, P3A 为有效电平, P3B 和 P3C 为无效电平

P3M<1:0>=10: 半桥输出模式; P3A 和 P3B 配置为调制输出; P3C 和 P3D 被分配为端口引脚, 此模式带有死区控制功能

P3M<1:0>=11: 全桥反向输出模式; P3B 配置为调制输出; P3C 为有效电平; P3A 和 P3D 为无效电平

PDT1<1:0>: PWM3 十位占空比的低 2 位, 高 8 位由寄存器 PWM3L 提供

P3ON<1:0>: PWM3 开关位

P3ON<1:0>=11: 打开 PWM3

P3ON<1:0>=00、01 或 10: 关闭 PWM3

PWM3M<1:0>: PWM3 输出有效电平设置位

PWM3M<1:0>=00: P3A、P3B、P3C 和 P3D 均为高电平有效

PWM3M<1:0>=01: P3A 和 P3C 为高电平有效; P3B 和 P3D 为低电平有效

PWM3M<1:0>=10: P3A 和 P3C 为低电平有效; P3B 和 P3D 为高电平有效
PWM3M<1:0>=11: P3A、P3B、P3C 和 P3D 均为低电平有效

8.2.3 PWM3CTL1 寄存器

寄存器8.5: PWM3CTL1: PWM3控制寄存器1(地址: 5BH)

复位值	bit7						bit0	
0000 0000	PRSEN	PDC6	PDC5	PDC5	PDC3	PDC2	PDC1	PDC0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

PRSEN: PWM 重启使能位

PRSEN=1: 自动关闭时, 一旦关闭事件消失, P3ASE 位将自动清 0, PWM 自动重启

PRSEN=0: 自动关闭时, P3ASE 由软件清 0, 以重启 PWM

PDC<6:0>: PWM 死区延时时间设置位, 用于设置死区延时的时间

8.2.4 P3ASCTL寄存器

寄存器8.6: P3ASCTL: PWM3自动关闭控制寄存器(地址: 5CH)

复位值	bit7						bit0	
0000 0000	P3ASE	P3ASS2	P3ASS1	P3ASS0	P3SSAC1	P3SSAC0	P3SSBD1	P3SSBD0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

P3ASE: 自动关闭事件状态位

P3ASE=1: 发生了关闭事件; 四路输出为关闭状态

P3ASE=0: 四路输出正常工作

P3ASS<2:0>: 自动关闭源选择位

P3ASS<2:0>=0XX: 禁止自动关闭

P3ASS<2:0>=1XX: INT 引脚电压为低电平

P3SSAC<1:0>: 引脚 P3A 和 P3C 关闭状态控制位

P3SSAC<1:0>=00: 驱动引脚 P3A 和 P3C 为 0

P3SSAC<1:0>=01: 驱动引脚 P3A 和 P3C 为 1

P3SSAC<1:0>=1x: 引脚 P3A 和 P3C 为三态

P3SSBD<1:0>: 引脚 P3B 和 P3D 关闭状态控制位

P3SSBD<1:0>=00: 驱动引脚 P3B 和 P3D 为 0

P3SSBD <1:0>=01: 驱动引脚 P3B 和 P3D 为 1

P3SSBD <1:0>=1x: 引脚 P3B 和 P3D 为三态

8.2.5 PATRCTL寄存器

寄存器8.7: PATRCTL: 脉冲转向控制寄存器(地址: 5DH)

复位值 —0 0001	bit7 —	—	—	STRSYNC	STREND	STRENC	STRENB	bit0 STRENA
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

STRSYNC: 转向同步位

STRSYNC=1: 在下一个 PWM 周期发生输出转向更新

STRSYNC=0: 在指令周期边界的开始发生输出转向更新

STREND: 转向使能位 D

STREND=1: P3D 引脚输出 PWM 波形, 其极性由 PWM3M<1:0>控制

STREND=0: P3D 引脚被分配为端口引脚

STRENC: 转向使能位 C

STRENC=1: P3C 引脚输出 PWM 波形, 其极性由 PWM3M<1:0>控制

STRENC=0: P3C 引脚被分配为端口引脚

STRENB: 转向使能位 B

STRENB=1: P3B 引脚输出 PWM 波形, 其极性由 PWM3M<1:0>控制

STRENB=0: P3B 引脚被分配为端口引脚

STRENA: 转向使能位 A

STRENA=1: P3A 引脚输出 PWM 波形, 其极性由 PWM3M<1:0>控制

STRENA=0: P3A 引脚被分配为端口引脚

8.2.6 PWM3 的周期、占空比及分辨率

8.2.6.1 PWM3 周期

PWM3 的周期通过 8 位的寄存器 PP3 (地址:52H)进行设置,其值可设置为 0~255,PWM3 的周期通过式 8.5 进行计算。

★ 式 8.5:
$$\text{PWM周期} = (\text{PP3} + 1) \cdot 4 \cdot T_{\text{osc}} \cdot (T2\text{预分频比})$$

8.2.6.2 PWM3 占空比

PWM3 占空比设置寄存器为 10 位, 通过寄存器 PWM3L(地址:55H)和 PWM3CTL0 的 PDT<1:0>位进行设置, PWM3L 为占空比的高 8 位, PDT<1:0>为低两位。脉冲宽度和占空比通过式 8.6 和式 8.7 计算:

$$\star \text{ 式8.6: } \quad \text{脉冲宽度} = (\text{PWM3L:PWM3CTL0} \langle 5:4 \rangle) \cdot T_{\text{osc}} \cdot (\text{T2预分频比})$$

$$\star \text{ 式8.7: } \quad \text{占空比} = \frac{\text{脉冲宽度}}{\text{PWM周期}} = \frac{\text{PWM3L:PWM3CTL0} \langle 5:4 \rangle}{4(\text{PP3}+1)}$$

8.2.6.3 PWM3 分辨率

当 PP3 为 255 时，PWM3 的最大分辨率为 10 位。分辨率的计算公式如式 8.8 所示。

$$\star \text{ 式8.8: } \quad \text{分辨率} = \frac{\text{Log}[4(\text{PP3}+1)]}{\text{log}2} \text{ 位}$$

8.2.6.4 PWM3 中断

当 PWM3 调制输出满一个周期时，将中断标志位 T2IF 置 1，如果 PWM3 中断使能，且 AIE(全局中断允许位)和 PUIE(外设中断允许位)置 1，程序将相应中断。

8.2.7 单输出模式

通过将寄存器 PWM3CTL0 中的 P3M<1:0>位设置为 00，选择单输出模式，在此模式下，默认从 P3A 引脚输出 PWM 信号，P3B、P3C 和 P3D 引脚为通用端口引脚。

在使能相应的 PWM3 引脚时，应将对应的 TRx.x 位清 0，以将此引脚设置为输出模式；如果相应引脚为模拟数字共用引脚，应将该引脚设置为通用数字 I/O 口。

可通过设置寄存器 PATRCTL(参考寄存器 8.7)的 STREN<D:A>位，使能或禁止 PWM3 的 4 路输出，PWM3 使用单输出模式时，最多可同时在 4 个引脚输出同一个 PWM 信号。如图 8.4 所示，在单输出模式下，将 P3A 一路设置为 PWM 输出和 P3A 和 P3B 两路同时设置为 PWM 输出的示例，其它设置情况与此类似。

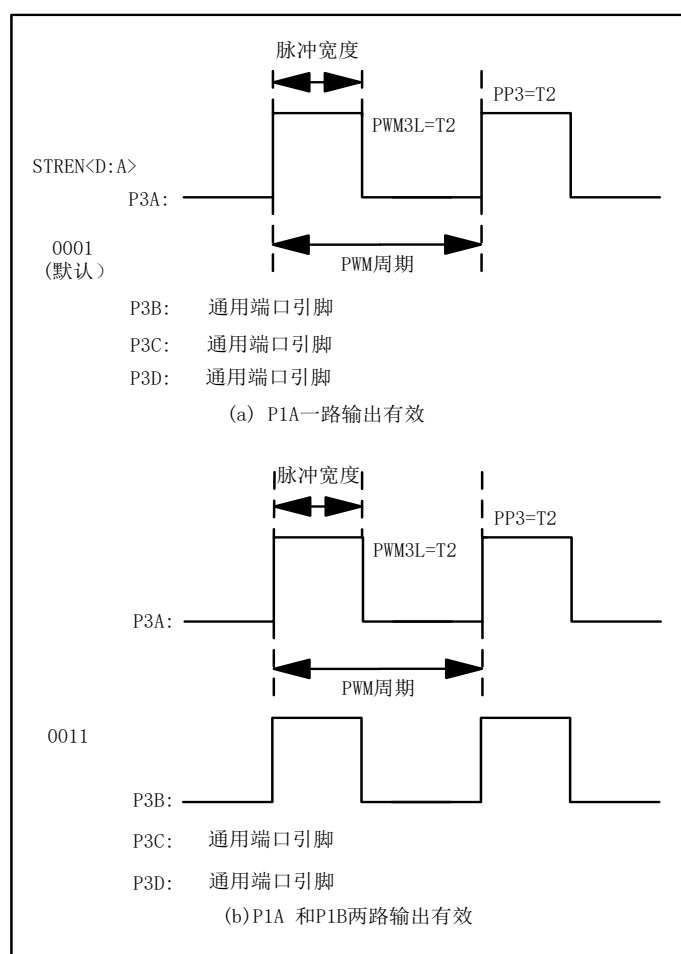


图 8.4 PWM3 的输出示例

在单输出模式过程中，如果开始时某路输出 PWM 信号无效(为通用端口)，现在要将其设置为有效 PWM 输出,此时可通过寄存器 PATRCTL 中的 STRSYNC 位进行设置引脚输出切换时是否与指令同步。如图 8.5 所示，STRSYNC=1 时，对应引脚 P3x 输出的 PWM 信号在 STRENx 置 1 后且的 PWM3 输出信号周期结束时输出；STRSYNC=0 时，对应引脚 P3x 输出的 PWM 信号在 STRENx 置 1 后立即输出；

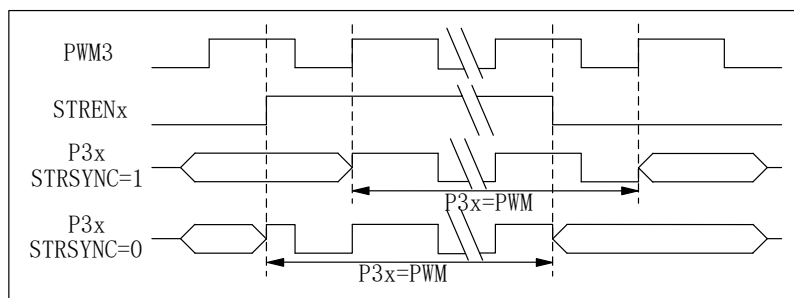


图 8.5 单输出模式 PWM 输出引脚切换

8.2.8 半桥输出模式

通过将寄存器 PWM3CTL0 的 P3M<1:0>位设置为 10，把 PWM3 设置为半桥输出模式。在此模式下，P3A 和 P3B 被配置为调制输出，来驱动推挽式负载，P3C 和 P3D 被配置为通用端口。PWM 输出信号在 P3A 引脚上输出，而互补的 PWM 输出信号在 P3B 引脚上输出，如图 8.6 所示。

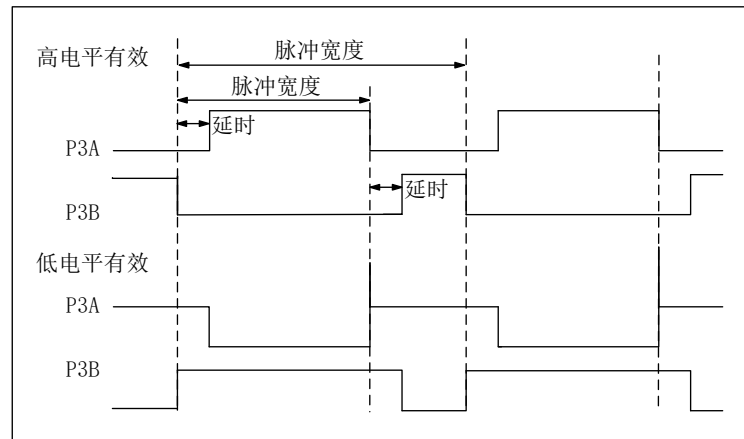


图 8.6 半桥输出模式输出信号示例

半桥输出模式可用于控制半桥和全桥控制电路，如图 8.7 所示，为半桥输出模式应用于半桥和全桥控制电路的示例。半桥输出模式应用于两个开关管的半桥控制电路，或使用 2 个 PWM 信号来控制 4 个开关管的全桥控制电路。

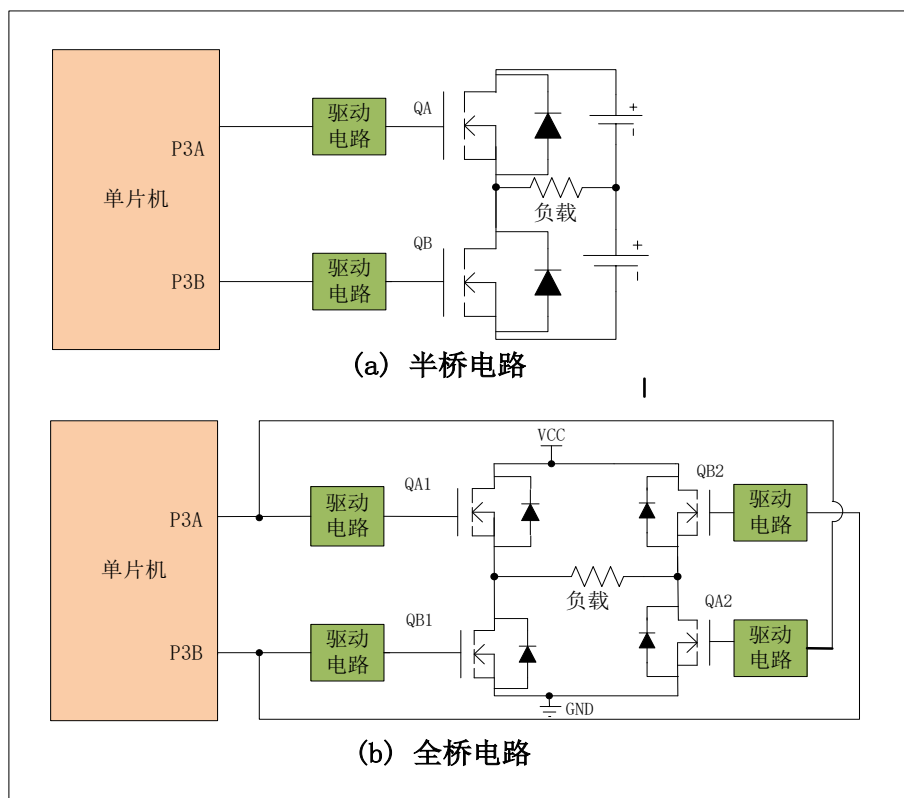


图 8.7 半桥输出模式应用举例

在使用半桥输出模式时，需将 P3A 和 P3B 引脚对应的方向控制位 TR1x 清 0，设置为输出，将对应 ANSEx 位清 0，设置为数字 I/O 口。

半桥输出模式具有可编程的死区延时功能，由于外部电路中的开关管等元件导通和截止时间存在差异，可用来防止在半桥驱动电路中产生直通电流，损坏相关电路。PWM3CTL1 寄存器中 PDC<6:0> 位的值用来设置死区延时时间。如果该值大于脉冲宽度，在整个周期内对应的输出将保持无效。

8.2.8.1 死区延时

如图 8.7(a)、8.8 所示，在半桥输出模式应用中，P3A 和 P3B 一直以 PWM 频率调制两个开关管，通常开关管的截止比导通需要更多的时间。如果 QA 和 QB 同时导通，两个管子可能会在一段很短的时间内都处于导通状态，在这很短的时间内，将会产生很大的电流流过两个管子，从而可能导致电路损坏。直到一个管子完全截止才会退出此状态(图 8.8(a)中的(1)、(2)、(3)和(4)处所示)。

为了避免开关期间产生这种具有破坏性的直通电流，可使其中一个管子关闭后再打开另一个管子。在半桥输出模式下，使用一个可编程死区延时模块，来避免产生的直通电流破坏电路。如图 8.8 所示，该延时在 PWM3 信号从非有效电平到有效电平转换时发生。延时时间通过寄存器 PWM3CTL1 的低 7 位进行设置。延时时间计数公式如式 8.9 所示。

★ 式8.9: 延时时间= PDC<6:0> • T_{osc}

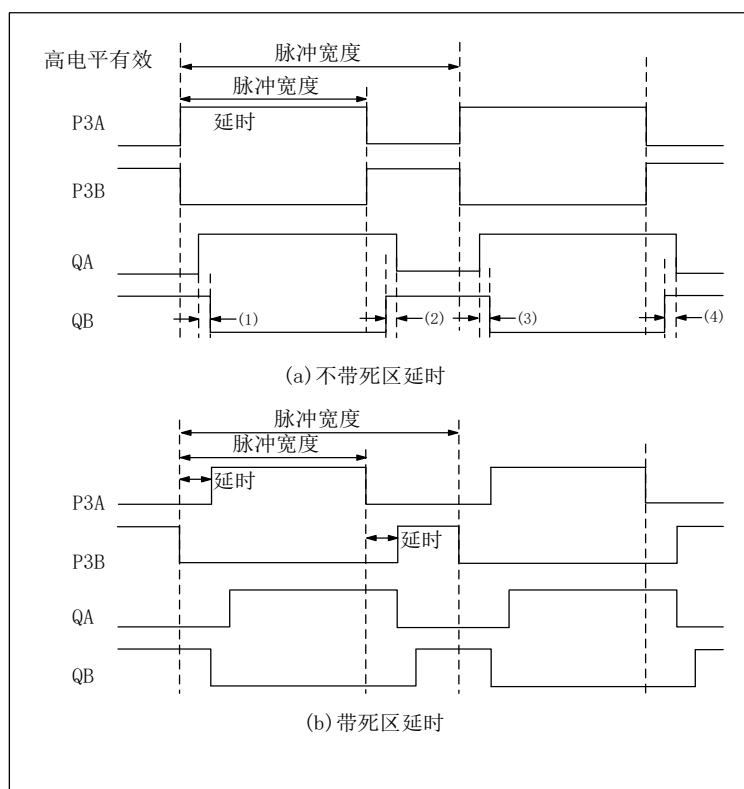
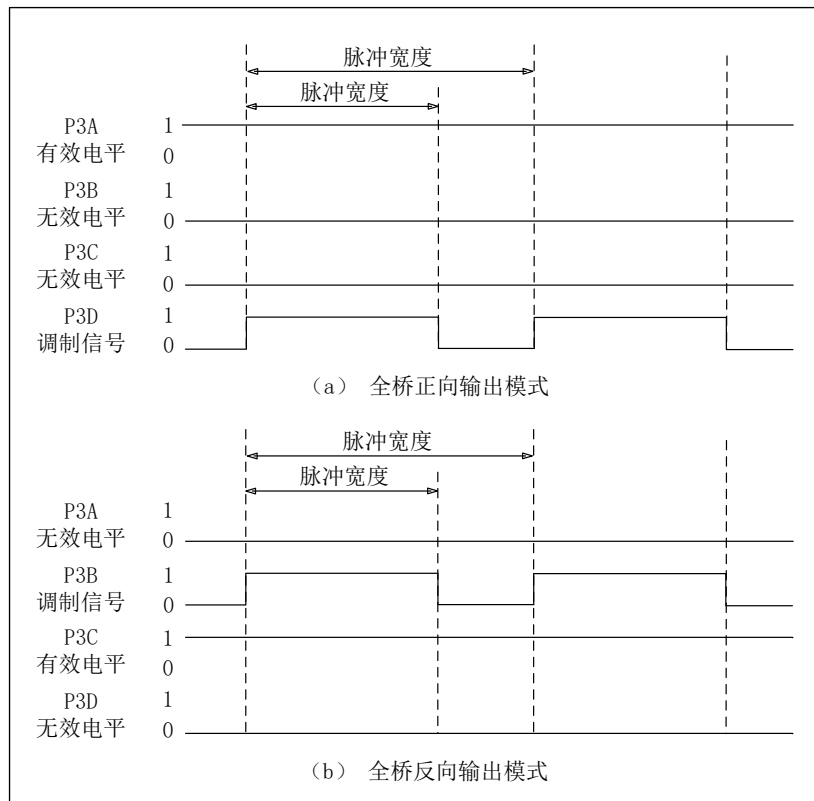


图 8.8 带死区控制和不带死区控制时的信号示例

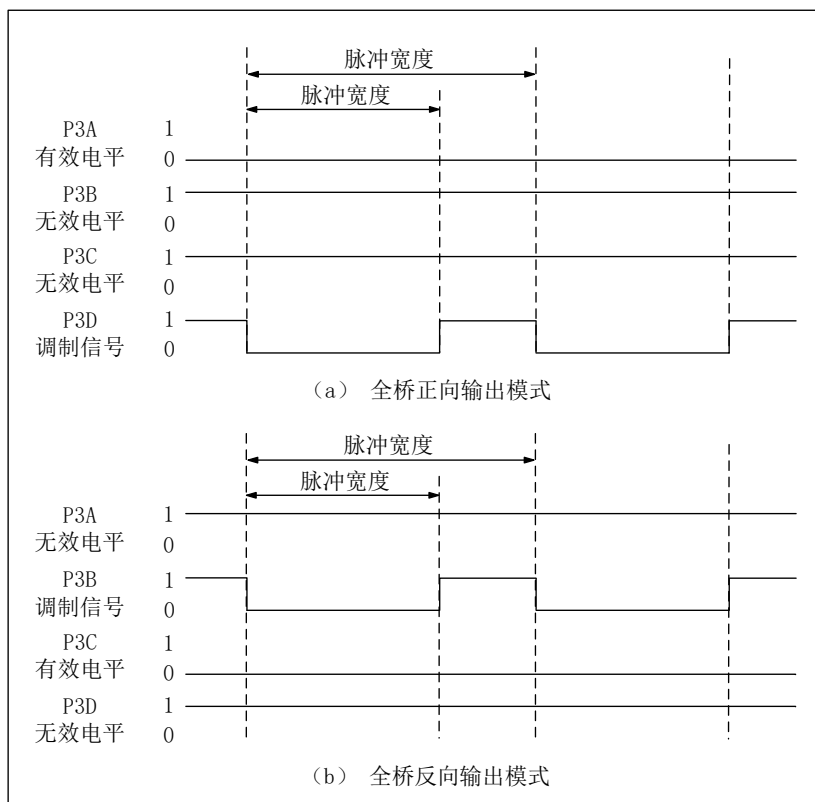
8.2.9 全桥输出模式

全桥输出模式有全桥正向输出模式和全桥反向输出模式两种。通过将寄存器 PWM3CTL0 的 P3M<1:0>设置为 01，把 PWM3 设置为全桥正向输出模式；将其设置为 11，把 PWM3 设置为全桥反向输出模式。

在全桥输出模式下，P3A、P3B、P3C 和 P3D 四个引脚都用作输出。将其设置为全桥正向输出模式时，引脚 P3A 被设置为有效电平，引脚 P3D 为 PWM 调制信号，P3B 和 P3C 为无效电平，图 8.9(a)和 8.10(a)为全桥正向输出模式引脚信号示例。将其设置为全桥反向输出模式时，P3C 被驱动为有效电平，引脚 P3B 为 PWM 调制信号，而 P3A 和 P3D 为无效电平，图 8.9(a)和 8.10(b)为全桥反向输出模式引脚信号示例。图 8.11 给出了全桥输出模式的应用电路示例。



8.9 全桥输出模式引脚信号示例(高电平有效)



8.10 全桥输出模式引脚信号示例(低电平有效)

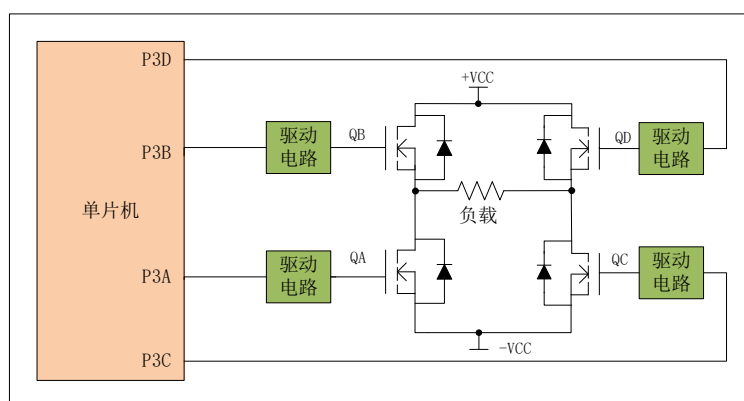


图 8.11 全桥输出模式应用电路示例

在使用全桥输出模式时，需将 P3A、P3B、P3C 和 P3D 引脚对应的方向控制位 TR1x 清 0，设置为输出，将对应 ANSEx 位清 0，设置为数字 I/O 口。

从全桥输出模式引脚信号波形以及应用电路示例可以看出：正向模式时，调制输出仅能控制一组开关管，假设此时流经负载的电流为正，则反向模式控制另一组开关管，使流经负载的电流为负。因此在使用全桥输出模式时，可将正向模式应与反向模式配合使用，使流经负载的电流方向改变。

通过将寄存器 PWM3CTL0 的 P3M<1:0>位设置为 01(正向模式)和 11(反向模式)来改变流经负载的电流。如图 8.12 所示，从一种模式模式切换到另一种模式时，在前一种模式最

后一个周期结束之前，P3B(或 P3D)被切换到无效状态，P3A(或 P3C)被切换到相反的状态。图中时间 $T = 1/F_{osc} \cdot T2$ 预分频值。

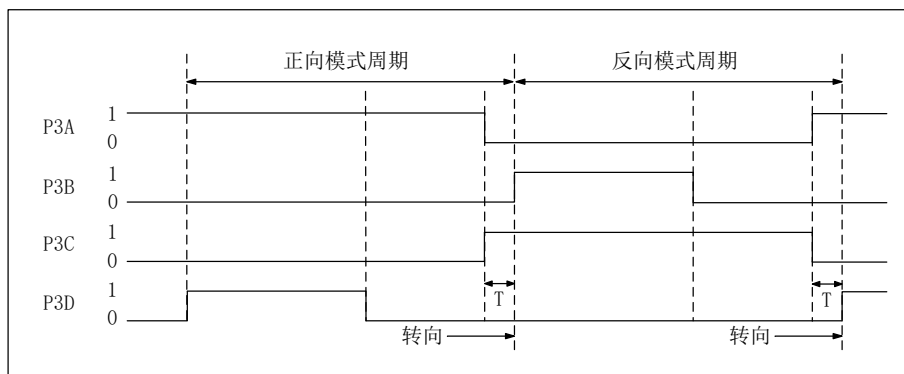


图 8.12 全桥正向、反向输出模式转换示例(高电平有效)

全桥输出模式下没有死区延时功能。通常在此模式中，任何时间只调制一对输出，因此不会导致电路产生直通电流，所以不需要死区延时。然而，当 PWM 的占空比接近百分之一百，且开关管导通时间小于截止时间时，将会导致电路产生直通电流。图 8.13 为此情况下各处信号示例。

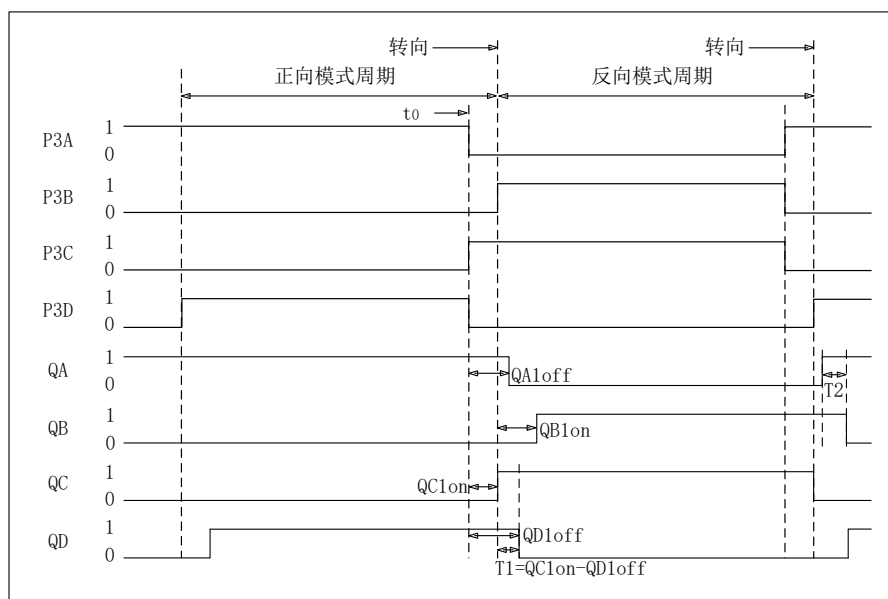


图 8.13 接近满占空比时换向时信号示例

图 8.13 中在 t_0 时刻 P3A 和 P3D 变为无效，P3C 为有效。QA、QB、QC 和 QD 分别为四路 PWM3 输出控制的开关管，QAoff 和 QDoff 为开关管的截止延时时间，QBon 和 QCon 为导通延时时间，由于管子导通和截止时间的差异，在第一次换向时，开关管 QC 和 QD 产生直通电流，持续时间为 T_1 ，在后面一次换向时 QA 和 QB 产生直通电流，持续时间为 T_2 。为了消除这种问题，可将换向前一个 PWM 信号周期的占空比调小，或者选用开关速度快的驱动电路，或者其它方法。

8.2.10 自动关断和自动重启模式

8.2.10.1 自动关断模式

PWM3 模块具有自动关断功能。如图 8.14 所示，为自动关断模式的原理框图。使能自动关断功能后，在外部关断事件发生时，该功能自动禁止 PWM 输出，然后将 P3A、P3B、P3C 和 P3D 四个引脚输出电平置于其预定义的状态。此模式用于防止 PWM 破坏应用电路。

自动关断模式具有 4 个关断源:INT0 引脚的逻辑低电平、比较器 1 输出高电平、比较器 2 输出高电平和在软件中直接将 P3ASE 位置 1。关断源触发关断的信号是高电平或低电平，而不是上升沿或下降沿，只要关断源的关断电平存在，自动关断状态将保持。

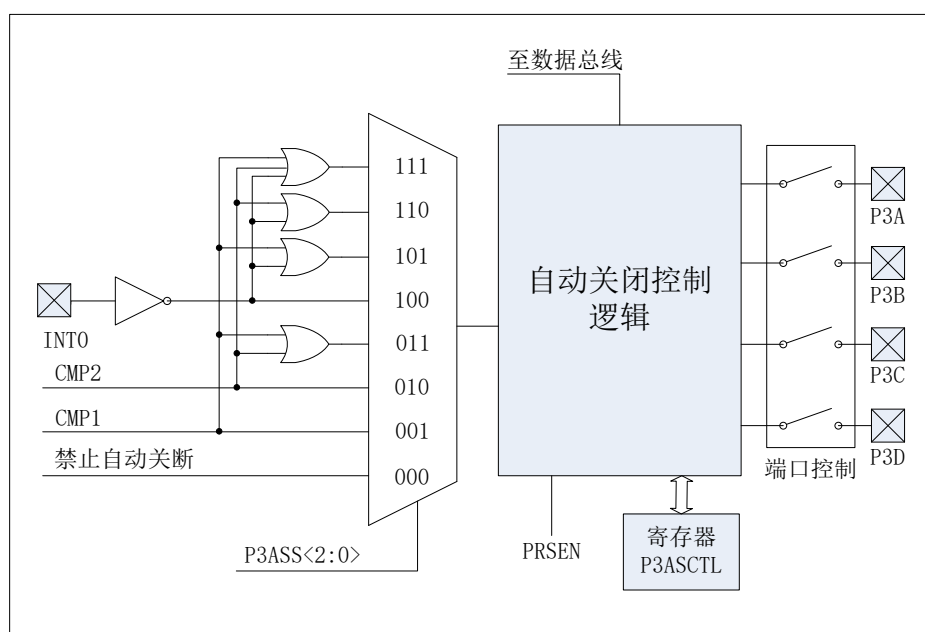


图 8.14 自动关断模式原理框图

通过寄存器 P3ASCTL 的 P3ASS<2:0>位选择自动关断源。将 P3ASS<2:0>位设置为 000 时，关闭自动关断功能。

寄存器 P3ASCTL(寄存器 8.6)中的 P3ASE 位指示关断的状态。如果该位为 0，表示 PWM3 的四个引脚输出正常的 PWM 信号，如果该位为 1，表示 PWM 的四路输出处于关断状态。

发生关断事件时，将会：

- ① P3ASE 位被置 1。直到被软件清 0 或发生自动重启才会将该位清 0。
- ② 使能的四个 PWM 引脚将被置于关断电平状态。

关断时，四路输出电平的状态由寄存器 P3ASCTL 的 P3SSAC<1:0>和 P3SSBD<1:0>位决定。通过设定可将输出引脚置为：三态、高电平和低电平三种状态。其中 P3A 和 P3C 的状态由 P3SSAC<1:0>设置，P3B 和 P3D 的状态由 P3SSBD<1:0>设置。

打开自动关断功能后，如果关断源产生关断事件，则 P3ASE 标志位被硬件置 1，四路输出被驱动为关断模式电平；关断源清除关断事件后，P3ASE 仍然为 1(如果 PRSEN=0)，四路输出仍然为关断模式电平，直到将 P3ASE 位清 0，PWM 重启。

8.2.10.2 自动重启模式

可将 PWM3 配置为一旦清除自动关断条件就自动重启 PWM。通过 将 PWM3CTL1 寄存器中的 PRSEN 位置 1 使能自动重启。

如果使能自动重启，只要自动关断条件有效，P3ASE 位就将保持置 1。当清除自动关闭条件时，将通过硬件将 P3ASE 位清 0，并且将恢复常规操作。

如图 8.15 所示，在图(a)中，PRSEN=0，自动重启模式关闭，通过软件将 P3ASE 标志位清 0，PWM 才会重启。在图(b)中，PRSEN=1，自动重启模式被打开，关断事件被清除后，P3ASE 标志位由硬件自动清 0，然后重启 PWM。

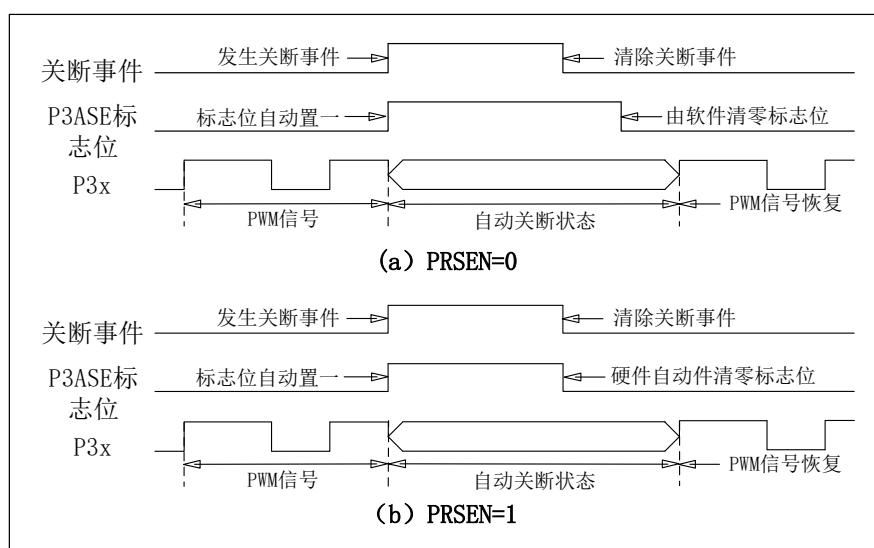


图 8.15 软件重启和自动重启 PWM 示例

9 模拟比较器模块

KF8F312M4 含有 2 路模拟比较器，其中模拟比较器的正端输入为 IO 端口，负端输入可选择 IO 端口或内部的 $1/2VDD$ 。

9.1 模拟比较器原理

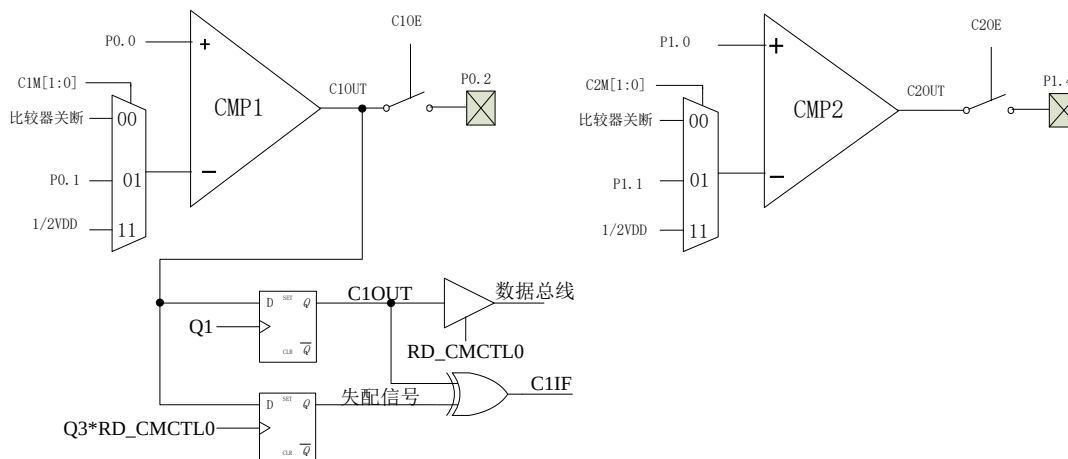


图 9.1 模拟比较器原理框图

9.2 CMCTL0 寄存器

寄存器 9.1: CMCTL0: CMCTL0 控制寄存器 (地址: 19H)

复位值	bit7						bit0
—00 00—	—	—	C2OUT	C1OUT	C2OE	C1OE	—
	u	u	R	R	R/W	R/W	u

C2OUT: 比较器 2 输出

1 = C2IN+ > C2IN-

0 = C2IN+ < C2IN-

C1OUT: 比较器 1 输出

1 = C1IN+ > C1IN-

0 = C1IN+ < C1IN-

C2OE: 比较器 CMP2 输出使能位

1 = CMP2 输出到对应引脚

0 = CMP2 不输出到对应引脚

C1OE: 比较器 CMP1 输出使能位

1 = CMP1 输出到对应引脚

0 = CMP1 不输出到对应引脚

9.3 CMCTL1 寄存器

寄存器9. 2: CMCTL1: CMCTL1控制寄存器(地址: 1AH)

复位值	bit7						bit0
0000	-	-	-	-	C2M1	C2M0	C1M1
	U	U	U	U	R/W	R/W	R/W

C2M[1:0]: 模拟比较器 2 的负输入端选择位

00 = 模拟比较器 2 关断

01 = P1.1 作为 CMP2 的负端输入

11 = 1/2VDD 作为 CMP2 的负端输入

C1M[1:0]: 模拟比较器 1 的负输入端选择位

00 = 模拟比较器 1 关断

01 = P0.1 作为 CMP1 的负端输入

11 = 1/2VDD 作为 CMP1 的负端输入

9.4 比较器中断

如图 9.1, 两个寄存器和一个异或门组成比较器 1 的失配电路。读取 CMCTL0 寄存器时, 失配信号的寄存器被比较器输出信号更新, 并保持到下次读取 CMCTL0 寄存器或复位为止。CMCTL0 的 C1OUT 寄存器在每个 Q1 时钟更新。当比较器输出变化引起寄存器 C1OUT 变化时, 将产生失配条件, C1OUT 和失配信号不相等, 产生比较器 CMP1 的中断标志 C1IF 置 1。失配条件将持续到读 CMCTL0 寄存器或比较器输出返回到先前状态为止。如果 CMP1 的中断使能, 且 AIE(全局中断允许位)和 PUIE(外设中断允许位)置 1, 程序将相应中断。

当比较器 2 输出寄存器 C2OUT 为 1 时, 比较器 CMP2 的中断标志 C2IF 将一直保持为 1。如果 CMP2 的中断使能, 且 AIE(全局中断允许位)和 PUIE(外设中断允许位)置 1, 程序将相应中断。

注: 由于对寄存器的写同样包含读操作, 写 CMCTL0 时同样将清除失配条件。

9.5 比较器的用法

- 1、 配置 I/O 口。需要作为输入的 I/O 端口如果有模拟输入功能则设为模拟输入口, 否则只用设为输入口, 需要作为输出的 I/O 端口只用设为输出口;
- 2、 配置 CMCTL0 寄存器。选择是否输出到对应引脚;
- 3、 配置 CMCTL1 寄存器。选择输入模式。

10 通用全双工/ 半双工收发器

10.1 系统概述

USART 是 Universal Synchronous /Asynchronous Receive & Transmit 的缩写，它的中文名称是通用同步/异步收发器，又称通用全双工/半双工收发器。本文称作通用全双工/半双工收发器。这是一个串口通信的 I/O 外设，也可作为串行通信接口。它可被配置为与个人计算机等外设通信的全双工异步系统。也可以被配置为与外设或其它单片机通信的半双工同步系统，与之通信的单片机通常不具有产生波特率的内部时钟，它需要主控同步器件提供外部时钟信号。

10.1.1 相关寄存器

表 10.1 相关寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
0BH	INTCTL	AIE	PUIE	TOIE	INT0IE	POIE	TOIF	INT0IF	POIF
0DH	EIF2	-	-	RXIF	TXIF	-	-	-	-
2DH	EIE2	-	-	RXIE	TXIE	-	-	-	-
58H	RSCTL	SPEN	RX9	SRXEN	CRXEN	ADREN	FRER	OVFER	RX9D
59H	TXSDR	USART 发送数据寄存器							
5AH	RXSDR	USART 接收数据寄存器							
77H	BRCTL	ABRDOVF	RCIDLF	-	SCKPS	BRG16	-	WUEN	ADRBEN
78H	TSCTL	CSRS	TX9	TXEN	SYNC	SENDER	HBRG	TXSRS	TX9D
79H	EUBRGL	BRG7	BRG6	BRG5	BRG4	BRG3	BRG2	BRG1	BRG0
7AH	EUBRGH	BRG15	BRG14	BRG13	BRG12	BRG11	BRG10	BRG9	BRG8

10.1.2 原理框图

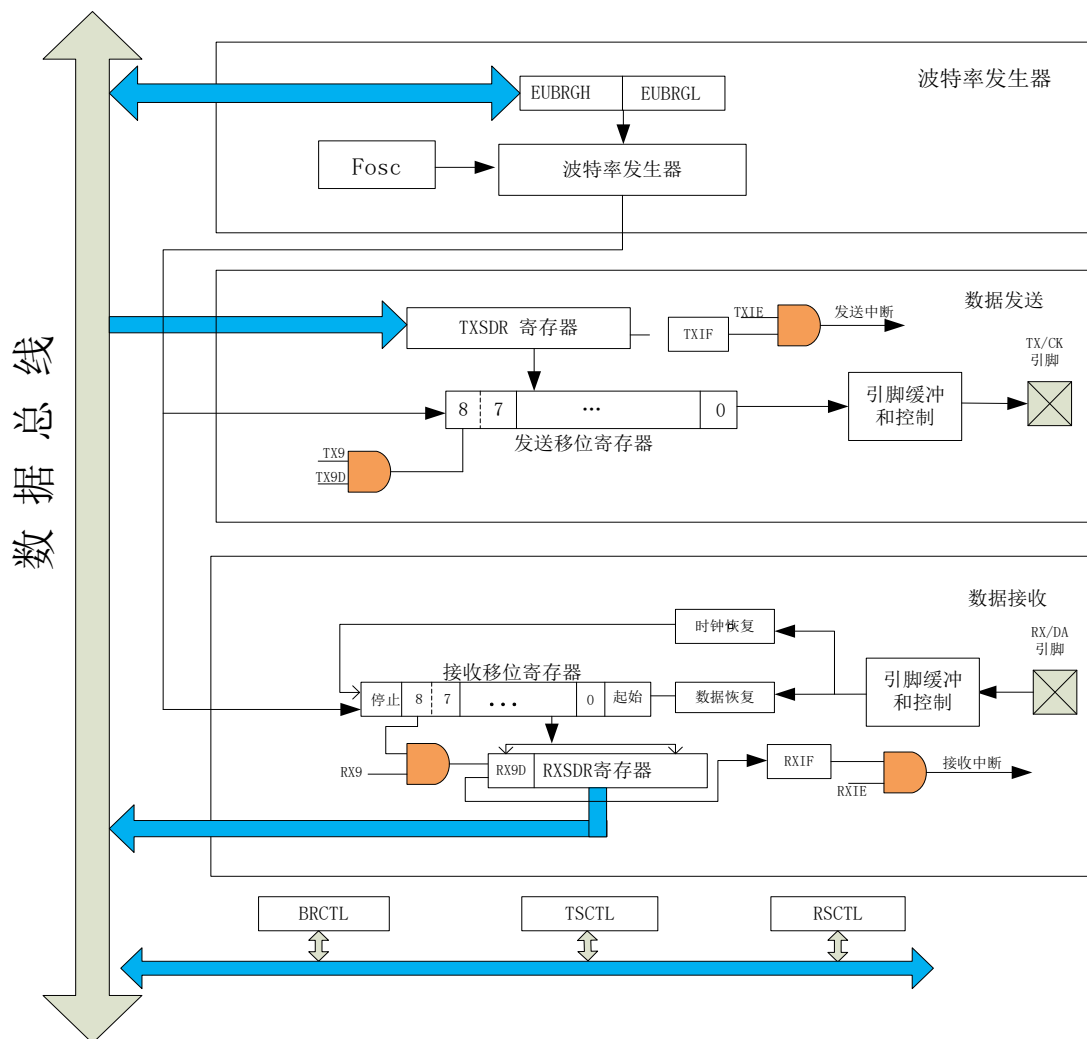


图 10.1 USART 模块框图

从图 10.1 中可以看出，通用全双工/半双工收发器（USART）模块包括波特率发生器（BRG）、数据发送和数据接收这三部分，每个部分都有相应的寄存器设置，主要包括波特率控制寄存器 BRCTL，发送状态和控制寄存器 TSCTL 和接收状态和控制寄存器 RSCTL。

在波特率发生器这部分，通过设置 SYNC、HBRG 和 BRG16 位来进行选择设置 EUBRGH: EUBRGL 寄存器的值来配置波特率。详细见“10.2 波特率发生器”。

在数据发送部分，包括全双工异步发送和半双工同步发送，通过发送状态和控制寄存器 TSCTL 来控制实现相应的功能，详见“10.3 及 10.4”。

在数据接收部分，包括全双工异步接收和半双工同步接收，通过接收状态和控制寄存器 RSCTL 来控制实现相应的功能，详见“10.3 及 10.4”。

USART 有如下功能特点：全双工异步发送和接收、RS485 检测、双字节输入缓冲器、单字节输出缓冲器、可将字符长度编程为 8 位或 9 位、输入缓冲溢出错误检测、接收到字符的帧错误检测、半双工同步主控/从动模式和半双工同步模式下可编程时钟极性。

USART 模块还可实现如下附加功能，从而使其成为局域互联网络总线系统的理想选择：自动波特率检测、校准和 13 位间隔字符发送。

10.2 波特率发生器

波特率发生器（BRG）是 8 位或者 16 位定时器，专用于支持全双工和半双工 USART 操作，默认情况下，BRG 工作在 8 位模式。将 BRCTL 寄存器的 BRG16 位置 1 可选择 16 位模式。

EUBRGH 和 EUBRGL 寄存器决定运行波特率定时器的周期。在全双工模式下，波特率周期的乘数由 TSCTL 寄存器的 HBRG 位和 BRCTL 寄存器的 BRG16 位决定。在半双工同步模式下，HBRG 位被忽略。

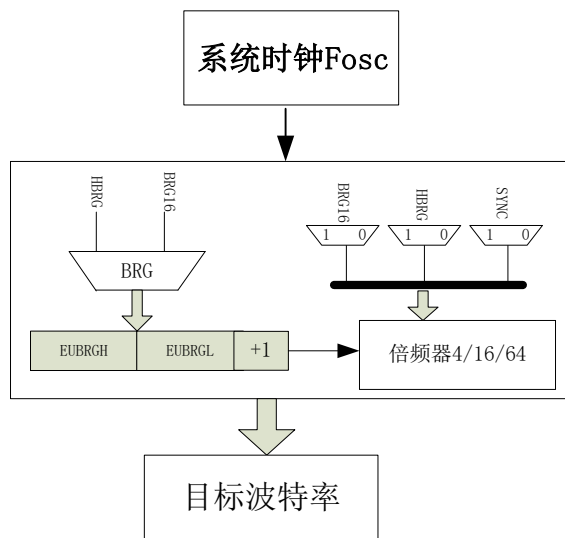


图 10.2 目标波特率产生原理框图

10.2.1 波特率控制寄存器 BRCTL

寄存器 10.1: BRCTL: 波特率控制寄存器 (地址: 77H)

bit7				bit0			
ABRDOVF	RCIDLF	—	SCKPS	BRG16	—	WUEN	ABRDEN
R/W	R/W	U	R/W	R/W	U	R/W	R/W

ABRDOVF: 自动波特率检测溢出位

全双工异步模式:

ABRDOVF=1 自动波特率定时器溢出

ABRDOVF=0 自动波特率定时器没有溢出

半双工同步模式:

无关位

RCIDLF: 接收空闲标志位

全双工异步模式:

RCIDLF=1 接收器空闲

RCIDLF=0 已接收到起始位且接收器正在接收

半双工同步模式:

	无关位
SCKPS:	半双工同步时钟极性选择位
	全双工异步模式:
	SCKPS=1 传送反相数据到 P27/TX/CK 引脚
	SCKPS=0 传送非反相数据到 P27/TX/CK 引脚
	半双工同步模式:
	SCKPS=1 数据在时钟上升沿同步
	SCKPS=0 数据在时钟下降沿同步
BRG16:	16 位波特率发生器位
	BRG16=1 使用 16 位波特率发生器
	BRG16=0 使用 8 位波特率发生器
WUEN:	唤醒使能位
	全双工异步模式:
	WUEN=1 接收器正在等待下降沿。不会接收字符。RXIF 将被置 1。RXIF 置 1 后 WUEN 将被自动清 0。
	WUEN=0 接收器正常工作
	半双工同步模式:
	无关位
ABRDEN:	自动波特率检测使能位
	全双工异步模式:
	ABRDEN=1 使能自动波特率模式(完成自动波特率后清 0)
	ABRDEN=0 禁止自动波特率模式
	半双工同步模式:
	无关位

10.2.2 波特率的选择

波特率的选择与状态位 SYNC、BRG16 和 HBRG 有关，还与工作定时器 8 位或 16 位、半双工同步或全双工都有关系。

波特率的计算公式为：

$$\text{目标波特率} = \frac{F_{\text{osc}}}{m \times ([\text{EUBRGH} : \text{EUBRGL}] + 1)}$$

其中 m 为倍频器选择，分别为 4、16 和 64，它与状态位 SYNC、BRG16 和 HBRG 有关，并且与工作的定时器 8 位还是 16 位，半双工或全双工有一定的关系。表 10.2 列出了 m 选择表。

表 10.2 倍频器 m 选择表

配置位			BRG/USART 模式	倍频器 m
SYNC	BRG16	HBRG		
0	0	0	8 位/异步	64
0	0	1	8 位/异步	16
0	1	0	16 位/异步	
0	1	1	16 位/异步	4
1	0	x	8 位/同步	
1	1	x	16 位/同步	

注：x 为无关位

例 10.1 是确定波特率和波特率误差的计算示例。

例 10.1: 计算波特率误差

器件工作在 $F_{osc} = 4\text{MHz}$ ，目标波特率 = 9600，全双工异步模式，8 位 BRG:

目标波特率 = $\frac{F_{osc}}{m \times ([EUBRGH:EUBRGL] + 1)}$ ，求解 EUBRGH:EUBRGL。

$$X = \frac{\frac{F_{osc}}{\text{目标波特率}} - 1}{64}$$

$$\begin{aligned} &= \frac{\frac{4000000}{9600} - 1}{64} \\ &= 5.5104 \\ &= 5.5 \text{ 十进制数} \\ &= 05\text{H} \text{ 十六进制数} \end{aligned}$$

$$\text{计算波特率} = \frac{4000000}{64(5.5 + 1)}$$

$$= 9615$$

$$\text{误差} = \frac{\text{计算波特率} - \text{目标波特率}}{\text{目标波特率}}$$

$$= \frac{(9615 - 9600)}{9600}$$

$$= 0.16\%$$

各种全双工异步模式的典型波特率和误差值已经计算出来，如表 10.3 所示。使用高波特率（HBRG = 1）或 16 位 BRG（BRG16 = 1）有助于降低波特率误差。16 位 BRG 模式用于在高速振荡器频率下取得较缓慢的波特率。

将新值写入 EUBRGH 和 EUBRGL 寄存器将导致 BRG 定时器复位（或清 0）。这可以确保 BRG 无需等待定时器溢出就可以输出新的波特率。

如果在有效接收操作过程中更改系统时钟，可能会导致接收错误或数据丢失。为避免此问题，应检查 RCIDLIF 位的状态，以确保在更改系统时钟前接收操作处于空闲状态。

表 10.3 全双工异步模式下波特率

SYNC=0,HBRG=0,BRG16=0									
波特率 FOSC		300	1200	2400	9600	10417	19.2K	57.6K	115.2K
	实际波特率	—	1202	2404	9615	10417	19231	—	—
4.000MHz	实际波特率	300	1202	2404	—	10417	—	—	—
	误差(%)	0.16	0.16	0.16	—	0.00	—	—	—
	EUBRGL 值(十六进 制)	6BH	33H	19H	—	05H	—	—	—
2.000MHz	实际波特率	300	1202	2404	—	10417	—	—	—
	误差(%)	0.16	0.16	0.16	—	0.00	—	—	—
	EUBRGL 值(十六进 制)	67H	19H	0CH	—	02H	—	—	—
1.000MHz	实际波特率	300	1202	—	—	—	—	—	—
	误差(%)	0.16	0.16	—	—	—	—	—	—
	EUBRGL 值(十六进 制)	33H	0CH	—	—	—	—	—	—
512 KHz	实际波特率	307	-	-	—	—	—	—	—
	误差(%)	2.33	-	-	—	—	—	—	—
	EUBRGL 值(十六进 制)	19H	-	-	—	—	—	—	—

SYNC=0,HBRG=1,BRG16=0									
波特率 FOSC		300	1200	2400	9600	10417	19.2K	57.6K	115.2K
4.000MHz	实际波特率	—	1202	2404	9615	10417	19.23K	—	—
	误差(%)	—	0.16	0.16	0.16	0. 00	0.16	—	—
	EUBRGL 值(十六进 制)	—	CFH	67H	19H	17H	0CH	—	—
2.000MHz	实际波特率	—	1202	2404	9615	10417	—	—	—
	误差(%)	—	0.16	0.16	0.16	0. 00	—	—	—
	EUBRGL 值(十六进 制)	—	67H	33H	0CH	0BH	—	—	—
1.000MHz	实际波特率	300	1202	2404	—	10417	—	—	—
	误差(%)	0.16	0.16	0.16	—	0.00	—	—	—
	EUBRGL 值(十六进 制)	CFH	33H	19H	—	05H	—	—	—
511.000kHz	实际波特率	301	1231	2462	-	-	—	—	—
	误差(%)	0.63	2.58	2.58	-	-	—	—	—
	EUBRGL 值(十六进 制)	69H	19H	0CH	-	-	—	—	—

SYNC=0,HBRG=0,BRG16=1									
波特率 FOSC		300	1200	2400	9600	10417	19.2K	57.6K	115.2K
4.000MHz	实际波特率	300.1	1202	2404	9615	10417	19.23K	—	—
	误差(%)	0.04	0.16	0.16	0.16	0.00	0.16	—	—
	EUBRGL 值 (十六进制)	0340H	CFH	67H	19H	17H	0CH	—	—
2.000MHz	实际波特率	299.8	1202	2404	9615	10417	—	—	—
	误差(%)	-0.18	0.16	0.16	0.16	0.00	—	—	—
	EUBRGL 值 (十六进制)	01A0H	33H	33H	0BH	0BH	—	—	—
1.000MHz	实际波特率	300	1202	2404	—	10417	—	—	—
	误差(%)	0.16	0.16	0.16	—	0.00	—	—	—
	EUBRGL 值 (十六进制)	CFH	33H	19H	—	05H	—	—	—
511.000kHz	实际波特率	301	1231	2462	-	-	—	—	—
	误差(%)	0.63	2.58	2.58	-	-	—	—	—
	EUBRGL 值 (十六进制)	69H	19H	0CH	-	-	—	—	—

SYNC=0,HBRG=1,BRG16=1 或 SYNC=1, BRG16=1									
波特率 FOSC		300	1200	2400	9600	10417	19.2K	57.6K	115.2K
4.000MHz	实际波特率	300.0	1200	2398	9615	10417	19.23K	56.82K	111.1K
	误差(%)	0.01	0.04	0.08	0.16	0.00	0.16	2.12	-3.55
	EUBRGH: EUBRGL	0D04H	0340H	01A0H	0067H	005FH	0033H	0010H	0008H
2.000MHz	实际波特率	299.9	1199	2404	9615	10417	19.23K	55.56K	—
	误差(%)	-0.02	-0.08	0.16	0.16	0.00	0.16	-3.55	—
	EUBRGH: EUBRGL	0682H	01A0H	00CFH	0033H	002FH	0019H	0008H	—
1.000MHz	实际波特率	300.1	1202	2404	9615	10417	19.23K	—	—
	误差(%)	0.04	0.16	0.16	0.16	0.00	0.16	—	—
	EUBRGH: EUBRGL	0340H	00CFH	0067H	0019H	0017H	000CH	—	—
511.000kHz	实际波特率	300.5	1208	2415	9846	10666	—	—	—
	误差(%)	0.16	0.63	0.63	2.56	2.40	—	—	—
	EUBRGH: EUBRGL	01A9H	0069H	0034H	000CH	000BH	—	—	—

注：以上表格中波特率寄存器的值均是十六进制的数值形式

10.2.3 自动波特率检测

USART 模块支持自动波特率检测和校准。

在自动波特率检测（Auto-Baud Rate Detect, ABRD）模式下，BRG 不为 RX 输入提供时钟信号，而是由 RX 为 BRG 定时。波特率发生器用于为接收的 0X55（“U”的 ASCII 码）定时，0X55 是 LIN 总线的同步字符。此字符的特殊之处在于它具有包括停止位边沿在内的 5 个上升沿。

将 BRCTL 寄存器的 ABRDEN 位置 1 将启动自动波特率校验序列（图 10.3）。当发生 ABRD 序列时，USART 状态机保持在空闲状态。在接收线的第一个上升沿（起始位之后），EUBRGL 使用 BRG 计数器时钟递增计数，如表 10.4 所示。在第 8 位周期的末尾将在 RX 引脚上出现第 5 个上升沿。此时，对正确的 BRG 周期的累计值被留在 EUBRGH 和 EUBRGL 寄存器中，ABRDEN 位被自动清 0 而 RXIF 中断标志被置 1。要清除 RXIF 中断，需要读取 RXSDR 中的值。RXSDR 的内容应该被丢弃。校准不使用 EUBRGH 寄存器的模式时，用户可通过查询 EUBRGH 寄存器中的 0X00 验证 EUBRGL 寄存器是否未溢出。

BRG 自动波特率时钟由 BRG16 和 HBRG 位决定，如表 10.4 所示。在 ABRD 期间，EUBRGH 和 EUBRGL 寄存器共同用作 16 位计数器，这与 BRG16 位的设置无关。在校准波特率周期时，EUBRGH 和 EUBRGL 寄存器的定时频率为 BRG 基时钟频率的 1/8。得到的字节测量结果为全速下的平均位时间。

- 注：1、如果 WUEN 位和 ABRDEN 位都置 1，自动波特率检测将发生在间隔字符之后的字节处
2、需要由用户来判断输入字符的波特率是否处于所选 BRG 时钟源范围内。某些振荡器频率和 EUSART 波特率组合不可能实现。
3、在自动波特率过程中，自动波特率计数器从 1 开始计数。自动波特率序列完成后，为了得到最准确的结果，应从 EUBRGH:EUBRGL 寄存器对的值中减去 1。

表 10.4 BRG 计数器时钟速率

BRG16	HBRG	BRG 基时钟	BRG ABRD 时钟
0	0	FOSC/64	FOSC/512
0	1	FOSC/16	FOSC/128
1	0	FOSC/16	FOSC/128
1	1	FOSC/4	FOSC/32

注：在 ABRD 序列期间，EUBRGL 和 EUBRGH 寄存器都用作 16 位计数器，与 BRG16 的设置无关。

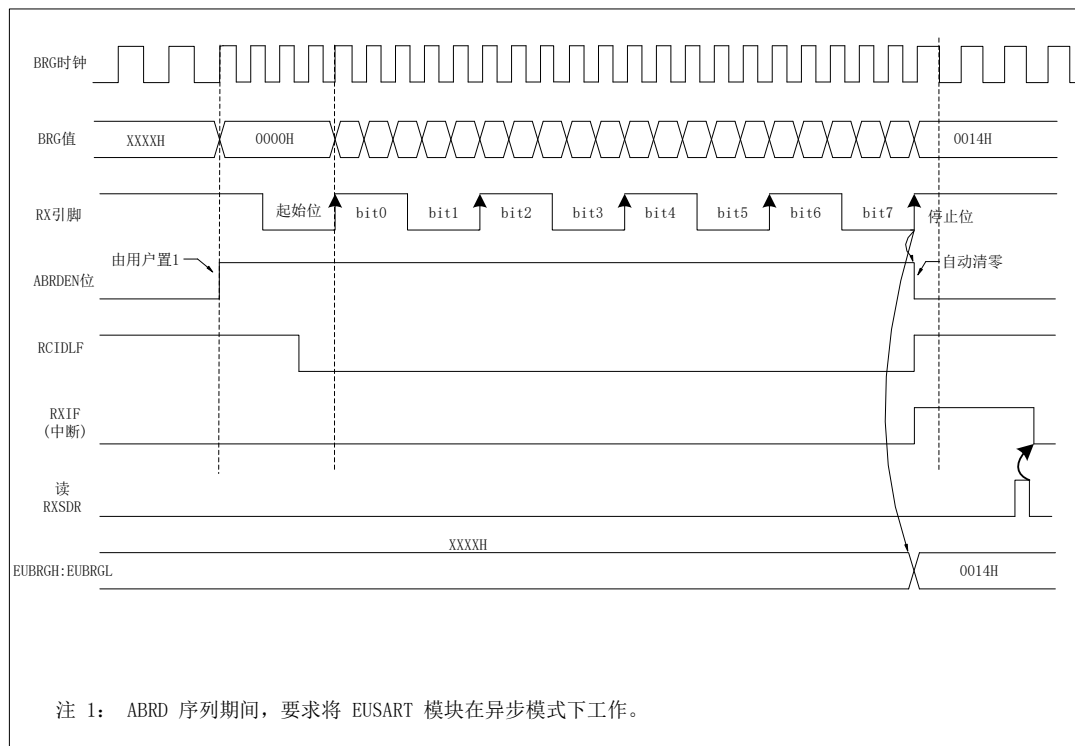


图 10.3 自动波特率检测

10.3 USART全双工模式

在全双工异步通信中，数据是一帧一帧传送的，每一帧的数据格式如图 10.4 所示。

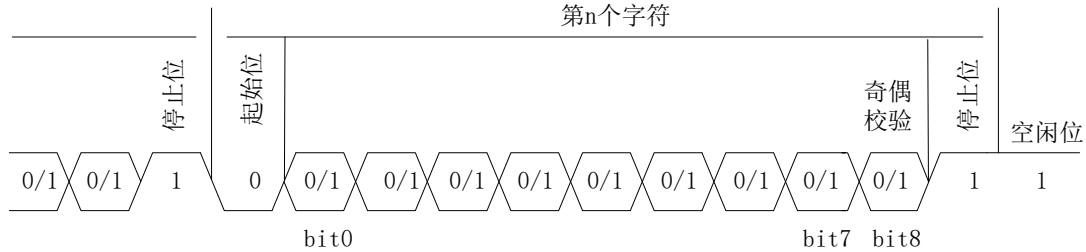


图 10.4 全双工异步通信数据结构图

在帧格式中，一个字符由 4 部分组成：起始位，数据位，奇偶校验位和停止位。

起始位：通常情况下是逻辑 0，占用一位，用来通知接收设备一个等待接收字符的开始。

数据位：8 位。

奇偶校验位：bit8，占用一位，但在字符中可以规定不用奇偶校验位，则这一位可以省去。

停止位：一定为逻辑 1，用来表征字符的结束。停止位可以是 1 位、1.5 位或 2 位。接收端收到停止位后，知道上一字符已经传送完毕，同时，也为接收下一字符作好准备——只要在接收到 0，就是新字符的起始位。若停止位以后不再紧接着传送下一个字符，则使线路电平保持为高电平（逻辑 1），处于空闲状态。这也是全双工异步通信的一大特点。

最常用的数据格式为 8 位。每个发送位的持续时间为 $1/(\text{波特率})$ 。片上专用 8 位/16 位波特率发生器可用于通过系统振荡器产生标准波特率频率。请参见表 10.2 了解波特率配置示例。

USART 首先发送和接收低位。USART 的发送器和接收器在功能上是相互独立的，但采用相同的数据格式和波特率。硬件不支持奇偶校验，但可以用软件实现（奇偶校验位是第 9 个数据位）。

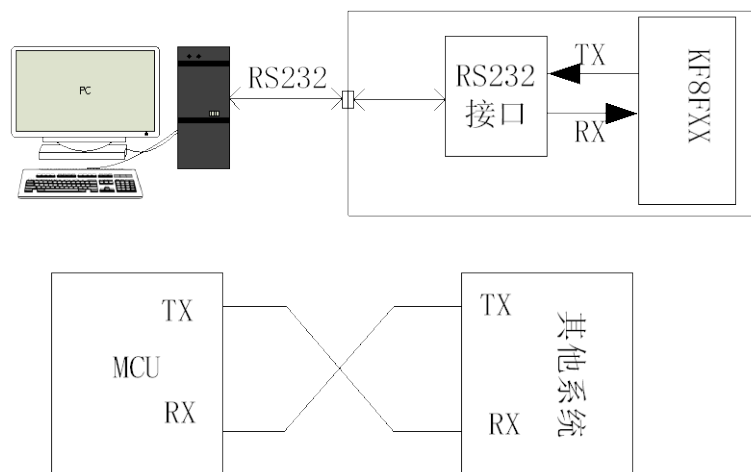


图 10.5 串口通信原理框图

10.3.1 USART全双工发送操作

USART 全双工异步发送操作通过 USART 发送器完成。发送器的核心是串行发送移位寄存器（发送移位），该寄存器不能由软件直接访问。发送移位寄存器从 TXSDR 发送缓冲寄存器获取数据。

通过配置如下三个控制位使能 USART 发送器，以用于全双工异步操作：

- 1) TXEN = 1
- 2) SYNC = 0
- 3) SPEN = 1

假设所有其它 USART 控制位处于其默认状态。将 TSCTL 寄存器的 TXEN 位置 1，使能 USART 发送器电路。将 TSCTL 寄存器的 SYNC 位清 0，将 USART 配置用于全双工异步操作。将 RSCTL 寄存器的 SPEN 位置 1，使能 USART 并自动将 TX/CK 的 I/O 引脚配置为输出引脚。如果与模拟外设共用 TX/CK 引脚，必须清 0 相应的 ANSEL 位禁止模拟 I/O 功能。

注：1、 将SPEN 位置1 会自动将RX/DT I/O 引脚配置为输入引脚时，无论相关的TR位的状态如何以及EUSART 接收器使能与否。可以通过普通端口读RX/DT 引脚数据，但却无法使用该端口锁存输出数据。

2、 如果TXEN 使能位置1，TXIF 发送器中断标志位会置1。

10.3.1.1 发送状态和控制寄存器TSCTL

寄存器11. 2: TSCTL: 发送状态和控制寄存器(地址:78H)

bit7				bit0			
复位值 0000 0010	CSRS	TX9	TXEN	SYNC	SENDER	HBRG	TRSR
	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- CSRS: 时钟源选择位
 全双工异步模式:
 无关位
 半双工同步模式:
 CSRS=1 主模式(由 BRG 内部产生时钟)
 CSRS=0 从模式(时钟源来自外部)
- TX9: RS485 发送使能位
 TX9=1 选择 RS485 发送
 TX9=0 选择 8 位发送
- TXEN: 发送使能位(1)
 TXEN=1 使能发送
 TXEN=0 禁止发送
- SYNC: 串行通信模式选择位

	SYNC=1 半双工同步模式
	SYNC=0 全双工异步模式
SENDB:	发送间隔字符位
	全双工异步模式:
	SENDB=1 在下次发送时发送同步间隔字符(完成后由硬件清 0)
	SENDB=0 同步间隔字符发送完成
	半双工同步模式:
	无关位
HBRG:	高波特率选择位
	全双工异步模式:
	HBRG=1 高速
	HBRG=0 低速
	半双工同步模式:
	在此模式下不使用
TXSRS:	发送移位寄存器状态位
	TXSRS =1 发送移位空
	TXSRS =0 发送移位满
TX9D:	发送数据的第 9 位
	可以是地址/数据位或奇偶校验位。

注： 在同步模式下， SRCEN/CRCEN 可改写 TREN。

10.3.1.2 发送数据

向 TXSDR 寄存器写入一个字符，以启动发送。如果这是第一个字符，或者前一个字符已经完全从发送移位寄存器中移出，TXSDR 中的数据会立即发送给发送移位寄存器。如果发送移位寄存器中仍保存全部或部分前一字符，新的字符数据将保存在 TXSDR 中，直到发送完前一字符的停止位为止。然后，在停止位发送完毕后经过一个发送周期，TXSDR 中待处理的数据将被传输到发送移位寄存器。当数据从 TXSDR 传输至发送移位寄存器后，立即开始进行起始位、数据位和停止位序列的发送。

只要使能 USART 发送器且 TXSDR 中没有待发送数据，就将 EIF2 寄存器的 TXIF 中断标志位置 1。换句话说，只有当发送移位寄存器忙于处理字符和 TXSDR 中有排队等待发送的新字符时，TXIF 位才处于清 0 状态。写 TXSDR 时，不立即清 0 TXIF 标志位。TXIF 在写指令后的第 2 个指令周期清 0。在写 TXSDR 后立即查询 TXIF 会返回无效结果。TXIF 为只读位，不能由软件置 1 或清 0。

可通过将 EIE2 寄存器的 TXIE 中断允许位置 1 允许 TXIF 中断。然而，只要 TXSDR 为空，不管 TXIE 允许位的状态如何都会将 TXIF 标志位置 1。

如果要在发送数据时使用中断，只有有待发送数据时，才将 TXIE 位置 1。当将待发送的最后一个字符写入 TXSDR 后，将 TXIE 中断允许位清 0。

TSCTL 寄存器的 TXSRS 位指示发送移位寄存器的状态。TXSRS 位为只读位。当发送移位寄存器为空时，TXSRS 位被置 1，当有字符从 TXSDR 传输到发送移位寄存器时，TXSRS 被清 0。TXSRS 位保持清 0 状态，直到所有位从发送移位寄存器移出为止。没有任何中断

逻辑与该位有关，所以用户必须查询该位来确定发送移位寄存器的状态。

10.3.1.3 全双工发送的设置

1. 初始化EUBRGH和EUBRGL这对寄存器以及HBRG和BRG16位，以获得所需的波特率（见第10.2节“波特率发生器（BRG）”）。
2. 通过将SYNC位清0并将SPEN位置1使能全双工异步串口。
3. 如果需要9位发送，将TX9控制位置1。当接收器被设置为进行地址检测时，将数据位的第9位置1，指示8个最低数据位为地址。
4. 将TXEN控制位置1，使能发送；这将导致TXIF中断标志位置1。
5. 如果需要中断，将EIE2寄存器中的TXIE中断允许位置1；如果INTCTL寄存器的AIE和PUIE位也置1将立即产生中断。
6. 若选择发送9位数据，第9位应该被装入TX9D数据位。
7. 将8位数据装入TXSDR寄存器开始发送数据。

例11.2 发送数据

```
SET TSCTL, TXEN
MOV R1, UART_TEMP
MOV TXSDR, R1
NOP
NOP
NOP
JB TSCTL, TXSRS
;判断是否发送完毕
JMP $-3
```

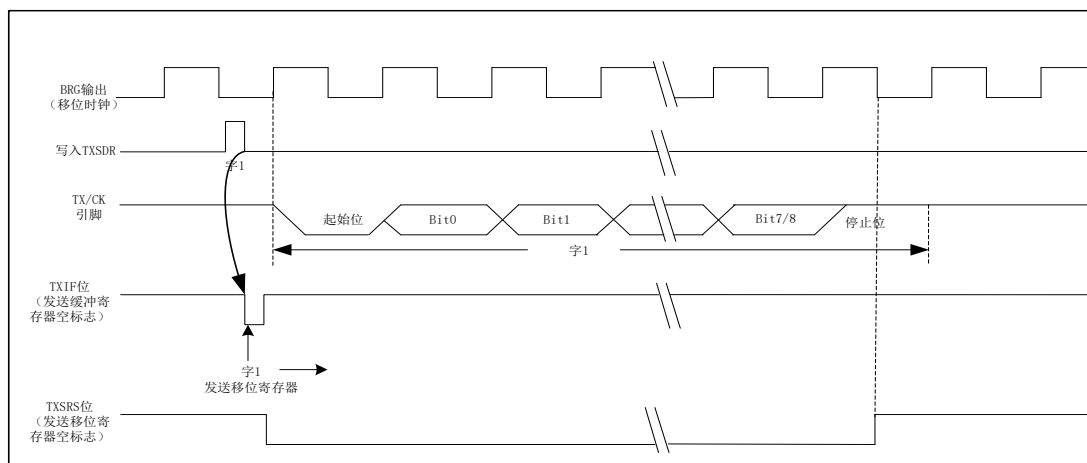


图 10.6 全双工异步发送（两字符间有空闲位）

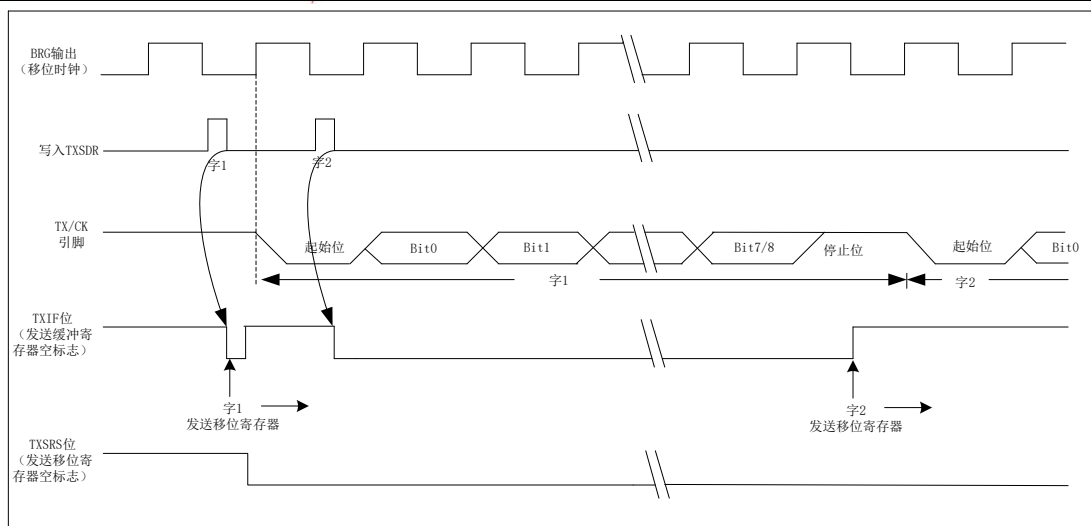


图 10.7 全双工异步发送（一字符接一字符）

10.3.2 USART全双工接收操作

全双工异步接收模式通常用于 RS-232 系统。图 10.1 给出了接收器的框图。在 RX/DT 引脚上接收数据和驱动数据恢复电路。数据恢复电路实际上是一个以 16 倍波特率为工作频率的高速移位器，而串行接收移位寄存器（ReceiveShift Register, RSR）则以波特率工作。当字符的全部 8 位或 9 位数据位被移入后，立即将它们传输到一个 2 字符的先入先出（FIFO 缓冲器）缓冲器。FIFO 缓冲器允许接收 2 个完整的字符和第 3 个字符的起始位，然后必须由软件将接收到的数据提供给 USART 接收器。FIFO 缓冲器和 RSR 寄存器不能直接由软件访问。通过 RXSDR 寄存器访问接收到的数据。

通过配置如下三个控制位使能 USART 接收器，以用于全双工异步接收操作。

◆ CRXEN = 1

◆ SYNC = 0

◆ SPEN = 1

假设所有其它 USART 控制位都处于默认状态。将 RSCTL 寄存器的 CRXEN 位置 1，使能 USART 接收器电路。将 TSCTL 寄存器的 SYNC 位清 0，配置 USART 以用于全双工异步操作。将 RSCTL 寄存器的 SPEN 位置 1，使能 USART 并自动将 RX/DT 引脚配置为输入引脚。如果 RX/DT 引脚与模拟外设共用，必须清 0 相应的 ANSEL 位禁止模拟 I/O 功能。

注： 当将SPEN 位置1，TX/CK I/O 引脚被自动配置为输出引脚时，无需考虑相应TR 位的状态以及EUSART发送器使能与否。端口锁存器与输出驱动器是断开的，从而不能将TX/CK 引脚用作通用输出引脚。

10.3.2.1 接收状态和控制寄存器RSCTL

寄存器11. 3: RSCTL: 接收状态和控制寄存器(地址:58H)

复位值 0000 000x	bit7						bit0	
	SPEN	RX9	SRXEN	CRXEN	ADREN	FRER	OVFER	RX9D
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- SPEN: 串行口使能位
 SPEN=1 使能串行口(将 RX/DT 和 TX/CK 引脚配置为串行口引脚)
 SPEN=0 禁止串行口(保持为复位状态)
- RX9: 9 位接收使能位
 RX9=1 选择 9 位接收
 RX9=0 选择 8 位接收
- SRXEN: 单字节接收使能位
 全双工异步模式:
 无关位
 半双工同步模式——主:
 SRXEN=1 使能单字节接收

	SRXEN=0 禁止单字节接收 此位在接收完成后清 0。 半双工同步模式——从： 无关位
CRXEN:	连续接收使能位 全双工异步模式： CRXEN=1 使能接收器 CRXEN=0 禁止接收器 半双工同步模式： CRXEN=1 使能连续接收，直到使能位 CRXEN 清 0(CRXEN 改写 SRXEN) CRXEN=0 禁止连续接收
ADREN:	地址检测使能位 RS-485 全双工异步模式(RX9 = 1): ADREN=1 使能地址检测、允许中断，当 RSR<8>置 1 时装入接收缓冲区 ADREN=0 禁止地址检测，接收所有字节并且第 9 位可作为奇偶校验位 8 位异步模式(RX9=0): 无关位
FRER:	帧错误位 FRER=1 帧错误(可通过读 RXSDR 寄存器刷新该位，并接收下一个有效字节) FRER=0 无帧错误
OVFER:	溢出错误位 OVFER=1 溢出错误(可通过清 0 CRXEN 位来清 0 该位) OVFER=0 无溢出错误
RX9D:	接收数据的第 9 位 该位可以是地址/数据位或奇偶校验位，并且必须由用户固件计算得到。

10.3.2.2 接收数据

接收器数据恢复电路在第一个位的下降沿开始接收字符。第一个位，通常称为起始位，始终为 0。由数据恢复电路计数半个位时间，到起始位的中心位置，校验该位是否仍为零。如果该位不为零，数据恢复电路放弃接收该字符，而不会产生错误，并且继续查找起始位的下降沿。如果起始位零校验通过，则数据恢复电路计数一个完整的位时间，到达下一位的中心位置。由数据检测电路对该位进行采样，将相应的采样结果 0 或 1 移入 RSR。重复该过程，直到完成所有数据位的采样并将其全部移入 RSR 寄存器。测量最后一个位的时间并采样其电平。此位为停止位，总是为 1。如果数据恢复电路在停止位的位置采样到 0，则该字符的帧错误标志将置 1，反之，该字符的帧错误标志会清 0。详见第 10.3.2.5 节“接收帧错误”获得有关帧错误描述。

当接收到所有数据位和停止位后，RSR 中的字符会被立即传输到 USART 的接收 FIFO 缓冲器并将 EIF2 寄存器的 RXIF 中断标志位置 1。通过读 RXSDR 寄存器将 FIFO 缓冲器最顶端的字符移出 FIFO 缓冲器。

注：如果接收FIFO溢出，则不能再继续接收其他字符，直到溢出条件被清除。请参见下一节“接收错误”获得有关溢出错误的更多相关信息。

只要使能 USART 接收器且在接收 FIFO 缓冲器中没有未读数据，EIF2 寄存器中的 RXIF 中断标志位就会清 0。RXIF 中断标志位为只读，不能由软件置 1 或清 0。通过将下列所有位均置 1 来允许 RXIF 中断：

- ◆ EIE2 寄存器的 RXIE 中断允许位
- ◆ INTCTL 寄存器的 PUIE 外设中断允许位
- ◆ INTCTL 寄存器的 AIE 全局中断允许位

如果 FIFO 缓冲器中有未读数据，无论中断允许位的状态如何，都会将 RXIF 中断标志位置 1。

10.3.2.3 接收错误

接收 FIFO 缓冲器中的每个字符都有一个相应的帧错误状态位。帧错误指示未在预期的时间内接收到停止位。由 RSCTL 寄存器的 FRER 位获取帧错误状态。FRER 位代表接收 FIFO 缓冲器最顶端未读字符的状态。因此，必须在读 RXSDR 寄存器之前读 FRER 位。

FRER 位为只读位，且只能用于接收 FIFO 缓冲器的最顶端未读字符。帧错误（FRER = 1）并不会阻止接收更多的字符。无需清 0 FRER 位。从 FIFO 缓冲器读下一字符会使 FIFO 缓冲器指针前进至下一字符和下一个相应的帧错误。

清 0 RSCTL 寄存器的 SPEN 位会复位 USART，并强制清 0 FRER 位。清 0 RSCTL 寄存器的 CRXEN 位不影响 FRER 位。帧错误本身不会产生中断。

注：如果接收FIFO缓冲器中所有接收到的字符都有帧错误，重复读RCSR 不会清零FRER 位。

接收 FIFO 缓冲器可以保存 2 个字符。但如果在访问 FIFO 缓冲器之前，接收到完整的第 3 个字符，则会产生溢出错误。此时，RSCTL 寄存器的 OVFER 位会置 1。可以读取 FIFO 缓冲器内的字符，但是在错误清除之前，不能再接收其它字符。可以通过清 0 RSCTL 寄存器的 CRXEN 位或通过清 0 RSCTL 寄存器的 SPEN 位使 USART 复位来清除错误。

10.3.2.4 地址检测

当多个接收器共享同一传输线时（如在 RS-485 系统中），可使用特殊地址检测模式。将 RSCTL 寄存器的 ADREN 位置 1，使能地址检测模式。地址检测要求接收 9 位字符。使能地址检测后，只有第 9 位数据位被置 1 的字符可以被传输到接收 FIFO 缓冲器，从而使 RXIF 中断标志位置 1。所有其它字符将被忽略。

由用户软件判断接收到的地址字符是否与其匹配。如果匹配，用户软件必须在下一个停止位产生之前，清 0 ADREN 位以禁止地址检测。当用户软件检测到信息的末尾（由所使用的信息协议判断）时，由软件将 ADREN 位置 1，从而使接收器返回地址检测模式。

10.3.2.5 全双工接收的设置

1. 初始化 EUBRGH 和 EUBRGL 这对寄存器以及 HBRG 和 BRG16 位，以获得所需的波特

率（见第 10.2 节“波特率发生器（BRG）”）。

2. 将 SPEN 位置 1，使能串行端口。必须清 0 SYNC 位以执行全双工异步操作。
3. 如果需要中断，将 EIE2 寄存器中的 RXIE 位和 INTCTL 寄存器的 AIE 和 PUIE 位置 1。
4. 如果需要接收 9 位数据，将 RX9 位置 1。
5. 将 CRXEN 位置 1 使能接收。
6. 当一个字符从 RSR 传输到接收缓冲器时，将 RXIF 中断标志位置 1。如果 RXIE 中断允许位也置 1 还将产生中断。
7. 读 RSCTL 寄存器获取错误标志位和第 9 位数据位（如果使能 9 位数据接收）。
8. 读 RXSDR 寄存器，从接收缓冲器获取接收到的 8 个低数据位。
9. 如果发生溢出，通过清 0 CRXEN 接收器使能位清 0 OVFER 标志。

例11.3 接收

LOOP_USART

```

NOP
NOP
SET    RSCTL, CRXEN           ;使能接收器
JNB    RSCTL, OVFER           ;检测溢出错误
CLR    RSCTL, CRXEN
JB     EIF2, RXIF              ;接收数据的标志
JMP    LOOP_USART
CLR    EIF2, RXIF
MOV    R0, RXSDR
MOV    UART_TEMP, R0          ;存储接收到的数据
    
```

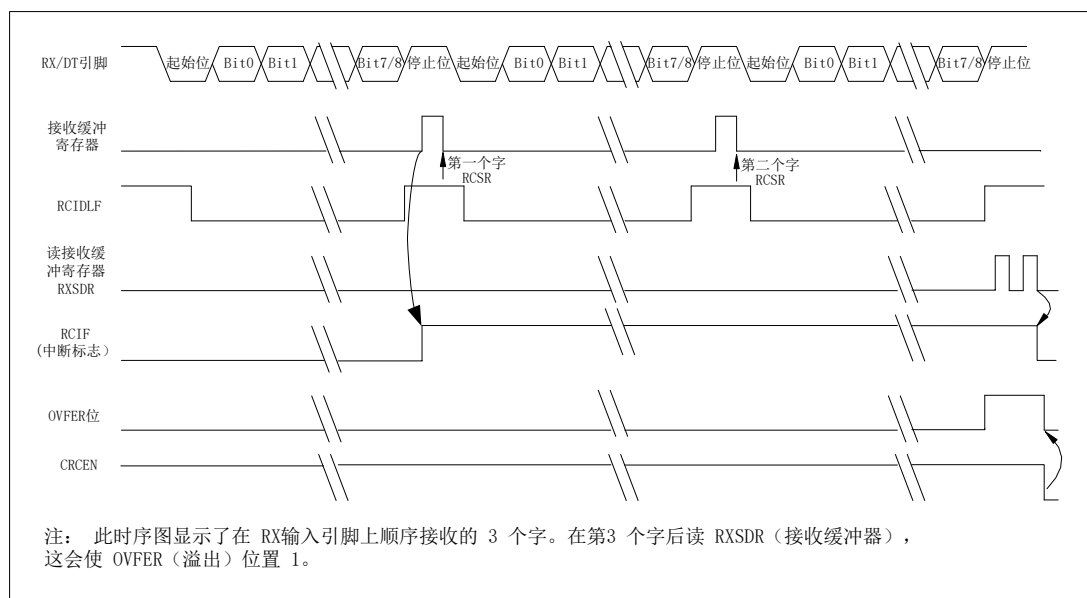


图 10.8 全双工异步接收

10.3.3 RS-485 发送/接收

USART 支持 RS-485 模式 9 位发送。当 TSCTL 寄存器的 TX9 位置 1 时，USART 将移出每个待发送字符的 9 位。TSCTL 寄存器的 TX9D 位为发送数据第 9 位，即最高数据位。当发送 9 位数据时，必须在将 8 个最低位写入 TXSDR 之前，写 TX9D 数据位。在写入 TXSDR 寄存器后会立即将 9 个数据位传输到发送移位寄存器。

使用多个接收器时可使用特殊的 9 位地址模式。请参见第 10.3.2.7 节“地址检测”获得有关地址模式的更多信息。

USART 支持 RS-485 模式 9 位接收。将 RSCTL 寄存器的 RX9 位置 1 时，USART 将接收到的每个字符的 9 位移入 RSR。RSCTL 寄存器的 RX9D 位是接收 FIFO 缓冲器顶端未读字符的第 9 位，同时也是最高数据位。当从接收 FIFO 缓冲器读取 9 位数据时，必须在读 RXSDR 中的低 8 位之前，读取 RX9D 数据位。

10.3.3.1 RS-485 9 位地址检测模式设置

要设置使能地址检测的全双工异步接收：

1. 初始化 EUBRGH 和 EUBRGL 这对寄存器以及 HBRG 和 BRG16 位，以获得所需的波特率（见第 10.2 节“波特率发生器（BRG）”）。
2. 将 SPEN 位置 1，使能串行端口。必须清 0 SYNC 位以执行全双工异步操作。
3. 如果需要中断，将 EIE2 寄存器中的 RXIE 位和 INTCTL 寄存器的 AIE 和 PUIE 位置 1。
4. 将 RX9 位置 1，使能 9 位数据接收。
5. 将 ADREN 位置 1，使能地址检测。
6. 将 CRXEN 位置 1 使能接收。
7. 当一个第 9 位置 1 的字符从 RSR 传输到接收缓冲器时，将 RXIF 中断标志位置 1。如果 RXIE 中断允许位也置 1 还将产生中断。
8. 读 RSCTL 寄存器获取错误标志位。第 9 个数据位始终置 1。
9. 读 RXSDR 寄存器，从接收缓冲器获取接收到的 8 个低数据位。由软件判断此地址是否为本地器件的地址。
10. 如果发生溢出，通过清 0 CRXEN 接收器使能位清 0 OVFER 标志。
11. 如果是对当前器件寻址，将 ADREN 位清 0 以允许所有接收到的数据进入接收缓冲器并产生中断。

10.3.4 全双工操作时钟的精确性

内部振荡器模块输出(OSCCTL)在出厂时做了校准。但是，VDD 或温度变化时 OSCCTL 频率有可能漂移，这将直接影响全双工异步波特率。下面的方法可用来调整波特率时钟，但要某种参考时钟源。

这种方法是调整波特率发生器的值。自动波特率检测可自动完成这种调整（见第 10.2.3 节“自动波特率检测”）。调整波特率发生器以补偿外设时钟频率的逐渐变化时，可能分辨率精度不够。

10.4 USART半双工模式

半双工同步串行通信通常用在具有一个主控制器件和一个或多个从动器件的系统中。主控制器件包含产生波特率时钟所必需的电路，并为系统中的所有器件提供时钟。从动器件可以使用主控时钟，因此无需内部时钟发生电路。

在半双工同步模式下，有 2 条信号线：双向数据线和时钟线。从动器件使用主控制器件提供的时钟，将数据串行移入或移出相应的接收和发送移位寄存器。半双工是指：主控制器件和从动器件都可以接收和发送数据，但是不能同时进行接收或发送。USART 既可以作为主控制器件，也可以作为从动器件。

半双工同步模式发送无需使用起始位和停止位。

10.4.1 USART半双工主控模式

下列位用来将 USART 配置为半双工同步主控操作：

- SYNC = 1
- CSRS = 1
- SRXEN = 0（用于发送）；SRXEN = 1（用于接收）
- CRXEN = 0（用于发送）；CRXEN = 1（用于接收）
- SPEN = 1

将 TSCTL 寄存器的 SYNC 位置 1，可将 USART 配置用于半双工同步操作。将 TSCTL 寄存器的 CSRS 位置 1，将器件配置为主控制器件。将 RSCTL 寄存器的 SRXEN 和 CRXEN 位清 0，以确保器件处于发送模式，否则器件配置为接收模式。将 RSCTL 寄存器的 SPEN 位置 1，使能 USART。如果 RX/DT 或 TX/CK 引脚与模拟外设共用，必须清 0 相应的 ANSEL 位禁止模拟 I/O 功能。

半双工同步数据传输使用独立的时钟线半双工同步传输数据。配置为主控制器件的器件在 TX/CK 引脚发送时钟信号。当 USART 被配置为半双工同步发送或接收操作时，TX/CK 输出驱动器自动使能。串行数据位在每个时钟的上升沿发生改变，以确保它们在下降沿有效。每个数据位的时间为一个时钟周期，有多少数据位就只能产生多少个时钟周期。

由 BRCTL 寄存器的 SCKPS 位选择时钟极性。将 SCKPS 位置 1 将时钟空闲状态设置为高电平。当 SCKPS 位置 1 时，数据在每个时钟的下降沿发生改变。清 0 SCKPS 位，将时钟空闲状态设置为低电平。当清 0 SCKPS 位时，数据在每个时钟的上升沿发生改变。

10.4.1.1 半双工主控发送

由器件的 RX/DT 引脚输出数据。当 USART 配置为半双工同步主控发送操作时，器件的 RX/DT 和 TX/CK 输出引脚自动使能。

向 TXSDR 寄存器写入一个字符开始发送。如果发送移位寄存器中仍保存全部或部分前一字符，新的字符数据保存在 TXSDR 中，直到发送完前一字符的停止位为止。如果这是第一个字符，或者前一个字符已经完全从发送移位中移出，则 TXSDR 中的数据会被立即传输到发送移位寄存器。当字符从 TXSDR 传输到发送移位后会立即开始发送数据。

每个数据位在主控时钟的上升沿发生改变，并保持有效，直至下一个时钟的上升沿为止。

注：发送移位寄存器并未映射到数据存储寄存器中，因此用户不能直接访问它。

半双工主控发送设置：

1. 初始化 EUBRGH 和 EUBRGL 这对寄存器以及 HBRG 和 BRG16 位，以获得所需的波特率（见第 10.2 节“波特率发生器”）。
2. 将 SYNC、SPEN 和 CSRS 位置 1，使能半双工同步主控串行端口。
3. 将 SRXEN 和 CRXEN 位清 0，禁止接收模式。
4. 将 TXEN 位置 1 使能发送模式。
5. 如果需要发送 9 位字符，将 TX9 置 1。
6. 若需要中断，将 EIE2 寄存器中的 TXIE 位，以及 INTCTL 寄存器中的 AIE 和 PUIE 位置 1。
7. 如果选择发送 9 位字符，应该将第 9 位数据装入 TX9D 位。
8. 通过将数据装入 TXSDR 寄存器启动发送。（请参考第 10.3.1.3 节例 10.2）

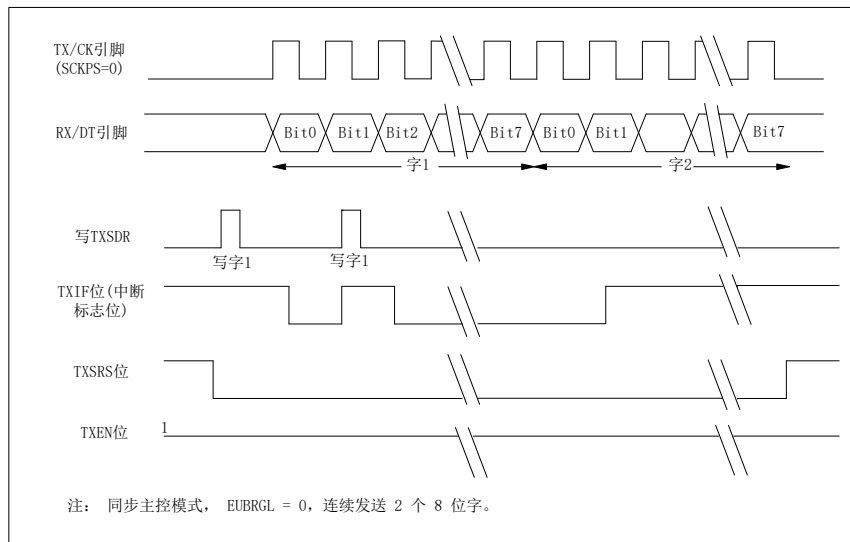


图 10.9 半双工同步发送（SCKPS=0）

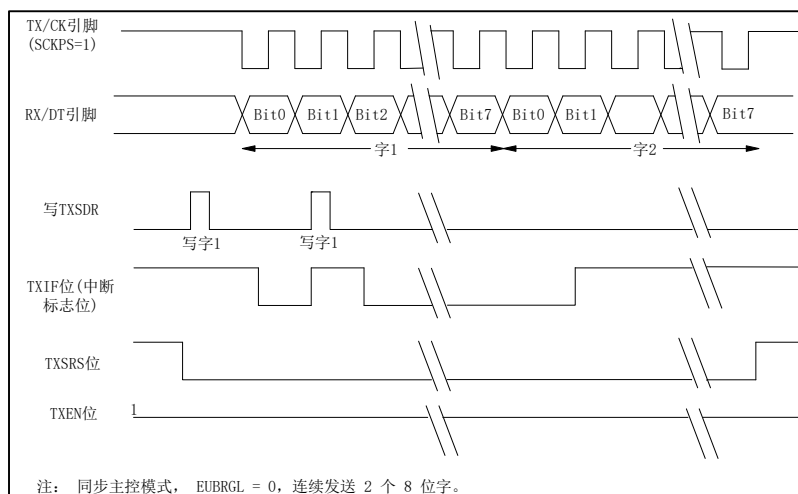


图 10.10 半双工同步发送(SCKPS=1)

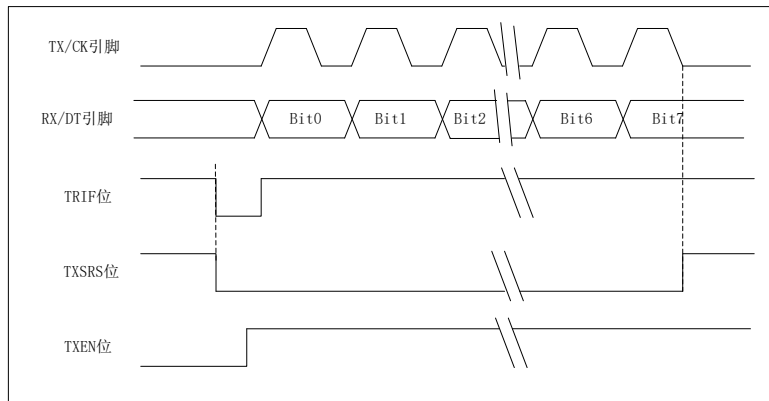


图 10.11 半双工同步发送（通过 TXEN）

10.4.1.2 半双工主控接收

在 RX/DT 引脚接收数据。当 USART 配置为半双工同步主控接收时，自动禁止器件的 RX/DT 引脚的输出驱动器。在半双工同步模式下，将单字接收使能位（RSCTL 寄存器的 SRXEN 位）或连续接收使能位（RSCTL 寄存器的 CRXEN 位）置 1 使能接收。

当将 SRXEN 置 1，CRXEN 位清 0 时，一个单字符中有多少数据位就只能产生多少时钟周期。一个字符传输结束后，自动清 0 SRXEN 位。当 CRXEN 置 1 时，将产生连续时钟，直到清 0 CRXEN 为止。如果 CRXEN 在一个字符的传输过程中清 0，则 CK 时钟立即停止，并丢弃该不完整的字符。如果 SRXEN 和 CRXEN 都置 1，则当第一个字符传输完成时，SRXEN 位被清 0，CRXEN 保持。

将 SRXEN 或 CRXEN 位置 1，启动接收。在 TX/CK 时钟引脚信号的下降沿采样 RX/DT 引脚上的数据，并将采样到的数据移入接收移位寄存器（RSR）。当 RSR 接收到一个完整字符时，将 RXIF 位置 1，字符自动移入 2 字节接收 FIFO 缓冲器。接收 FIFO 缓冲器中最顶端字符的低 8 位可通过 RXSDR 读取。只要接收 FIFO 缓冲器中仍有未读字符，则 RXIF 位就保持置 1 状态。

半双工同步数据传输使用与数据线通读的独立时钟线。配置为从器件的器件接收 TX/CK 线上的时钟信号。当器件被配置为半双工同步从发送或接收操作时，TX/CK 引脚的输出驱动器自动被禁止。串行数据位在时钟信号的前沿改变，以确保其在每个时钟的后沿有效。每个时钟周期只能传输一位数据，因此有多少数据位要传输就必须接收多少个时钟。

接收 FIFO 缓冲器可以保存 2 个字符。在读 RXSDR 以访问 FIFO 缓冲器之前，若完整地接收到第 3 个字符，则产生溢出错误。此时，RSCTL 寄存器的 OVFER 位会置 1。FIFO 缓冲器中先前的数据不会被改写。可以读取 FIFO 缓冲器内的 2 个字符，但是在错误被清除前，不能再接收其它字符。只能通过清除溢出条件，将 OVFER 位清 0。如果发生溢出时，SRXEN 位为置 1 状态，CRXEN 位为清 0 状态，则通过读 RXSDR 寄存器清除错误。如果溢出时，CRXEN 为置 1 状态，则可以清 0 RSCTL 寄存器的 CRXEN 位或清 0 SPEN 位以复位 USART，从而清除错误。

半双工主控接收设置：

1. 用正确的波特率初始化 EUBRGH:EUBRGL 寄存器。按需要将 HBRG 和 BRG16 位置 1

或清 0，以获得所需的波特率。

2. 将 SYNC、SPEN 和 CSRS 位置 1 使能半双工同步主控串行端口。
3. 确保将 CRXEN 和 SRXEN 位清 0。
4. 如果使用中断，将 INTCTL 寄存器的 AIE 和 PUIE 位置 1，并将 EIE2 寄存器的 RXIE 位也置 1。
5. 如果需要接收 9 位字符，将 RX9 位置 1。
6. 将 SRXEN 位置 1，启动接收，或将 CRXEN 位置 1 使能连续接收。
7. 当字符接收完毕后，将 RXIF 中断标志位置 1。如果允许位 RXIE 置 1，还会产生一个中断。
8. 读 RSCTL 寄存器以获取第 9 个数据位（使能 9 位接收时），并判断接收过程中是否产生错误。
9. 读 RXSDR 寄存器获取接收到的 8 位数据。
10. 如果产生溢出错误，清 0 RSCTL 寄存器的 CRXEN 位或清 0 SPEN 以复位 USART 来清除错误。（请参考第 10.3.2.5 节例 10.3）

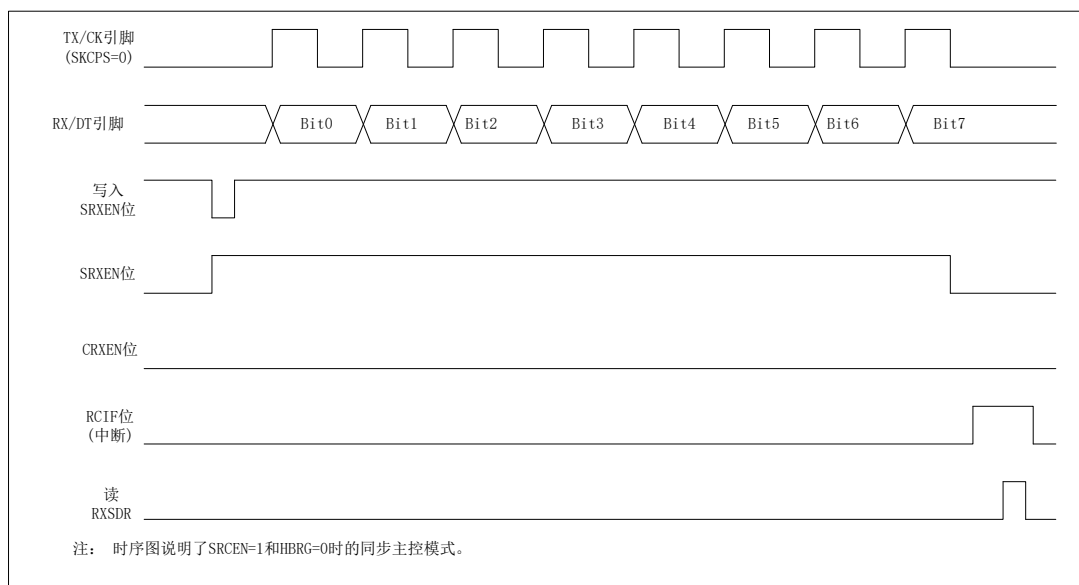


图 10.12 半双工同步接收（主控模式，SRXEN=1，SCKPS=0）

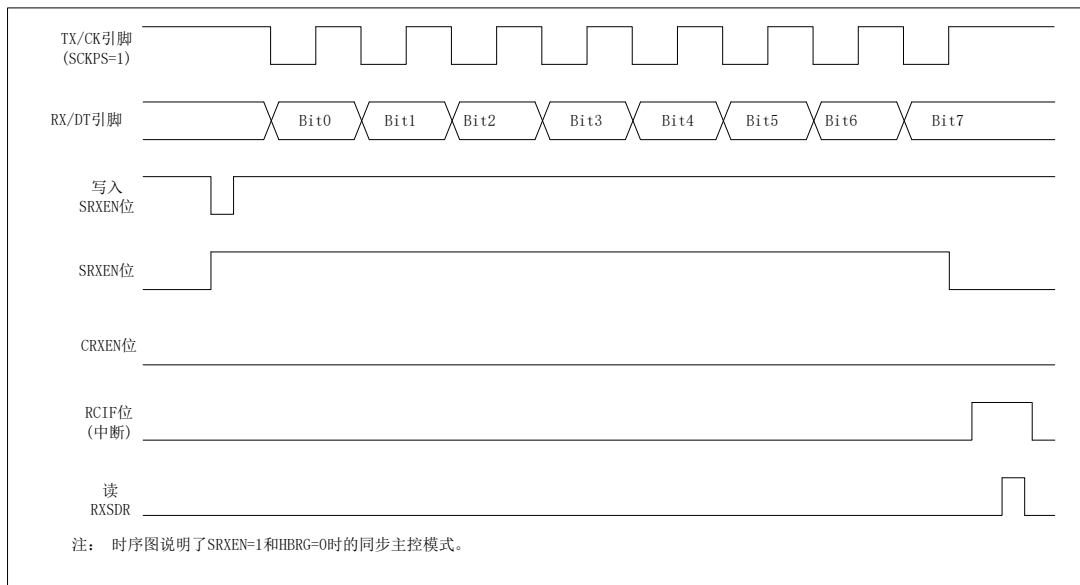


图 10.13 半双工同步接收（主控模式，SRXEN=1，SCKPS=1

10.4.2 USART半双工从动模式

下列位用来将 USART 配置为半双工同步从动操作：

- SYNC=1
- CSRS=0
- SRXEN=0（用于发送）；SRXEN = 1（用于接收）
- CRXEN=0（用于发送）；CRXEN = 1（用于接收）
- SPEN=1

将 TSCTL 寄存器的 SYNC 位置 1，可将器件配置用于半双工同步操作。将 TSCTL 寄存器的 CSRS 位置 1，将器件配置为从动器件。将 RSCTL 寄存器的 SRXEN 和 CRXEN 位清 0，以确保器件处于发送模式，否则器件将被配置为接收模式。将 RSCTL 寄存器的 SPEN 位置 1，使能 USART。如果 RX/DT 或 TX/CK 引脚与模拟外设共用，必须清 0 相应的 ANSEL 位禁止模拟 I/O 功能。

10.4.2.1 USART 半双工从动发送

除休眠模式外，半双工主控和从动模式的工作原理是相同的（见第 10.4.1.2 节“半双工主控发送”）。

如果向 TXSDR 写入 2 个字，然后执行 IDLE 指令，则会出现下列情况：

1. 第一个字立即传输到发送移位寄存器并进行发送。
2. 第二个字留在 TXSDR 寄存器中。
3. TXIF 中断标志位不会置 1。
4. 当第一个字符移出发送移位时，TXSDR 寄存器将把第二个字符传输到发送移位，然后标志位 TXIF 置 1。
5. 如果 PUIE 和 TXIE 位都置 1，则由中断将器件从休眠模式唤醒，然后执行下一条指令。如果 AIE 位也置 1，程序将调用中断服务程序。

USART 半双工从动发送设置：

1. 将 SYNC 和 SPEN 位置 1 并将 CSRS 位清 0。
2. 将 CRXEN 和 SRXEN 位清 0。
3. 如果使用中断，将 INTCTL 寄存器的 AIE 和 PUIE 位置 1，并将 EIE2 寄存器的 TXIE 位也置 1。
4. 如果需要发送 9 位数据，将 TX9 位置 1。
5. 将 TXEN 位置 1 使能发送。
6. 若选择发送 9 位数据，将最高位写入 TX9D 位。
7. 将低 8 位数据写入 TXSDR 寄存器开始传输。（请参考第 10.3.1.3 节例 10.2）

10.4.2.2 USART 半双工从动接收

除了以下不同外，半双工主控和从动模式的工作原理相同。（见第 10.4.1.2 节“半双工主控接收”）。

- 休眠模式
- CRXEN 位总是置 1，因此接收器不能进入空闲状态。
- SRXEN 位，在从动模式可为“任意值”。

如果在进入休眠模式之前，已经将 CRXEN 位置 1，则在休眠模式仍可接收字符。RSR 寄存器接收到字后，就会立即将接收到的数据传输到 RXSDR 寄存器。如果将 RXIE 允许位置 1，则产生的中断将使器件从休眠模式唤醒，然后执行下一条指令。如果 AIE 位也置 1，则程序将跳转到中断向量处执行。

半双工从动接收设置：

1. 将 SYNC 和 SPEN 位置 1 并将 CSRS 位清 0。
2. 如果使用中断，将 INTCTL 寄存器的 AIE 和 PUIE 位置 1，并将 EIE2 寄存器的 RXIE 位也置 1。
3. 如果需要接收 9 位字符，将 RX9 位置 1。
4. 将 CRXEN 位置 1，使能接收。
5. 当接收完成后，将 RXIF 位置 1。如果 RXIE 已置 1，还会产生一个中断。
6. 如果使能 9 位模式，从 RSCTL 寄存器的 RX9D 位获取最高位。
7. 读 RXSDR 寄存器，从接收 FIFO 缓冲器获取接收到的 8 个低数据位。
8. 如果产生溢出错误，清 0 RSCTL 寄存器的 CRXEN 位或清 0 SPEN 位以复位 USART 来清除错误。（请参考第 10.3.2.5 节例 10.3）

10.4.3 USART 半双工 RS-485 模式

请参考第 10.3.3 节 RS-485 发送/接收

10.4.4 USART 用作 SPI 模式

KF8F312M4 的半双工同步 USART 也能用作 SPI（Serial Peripheral Interface）通信，可以配置为主（Master）/从（Slave）模式，只能用作 SPI 单线传输。具体设置和半双工同步 USART 相同。配置为 Master 模式时请参考 10.4.1.1 半双工主控发送或 10.4.1.2 半双工主控接收；配置为 Slave 模式时，请参考 10.4.2.1 半双工从动发送或 10.4.2.2 半双工从动接收。通信如下图所示：

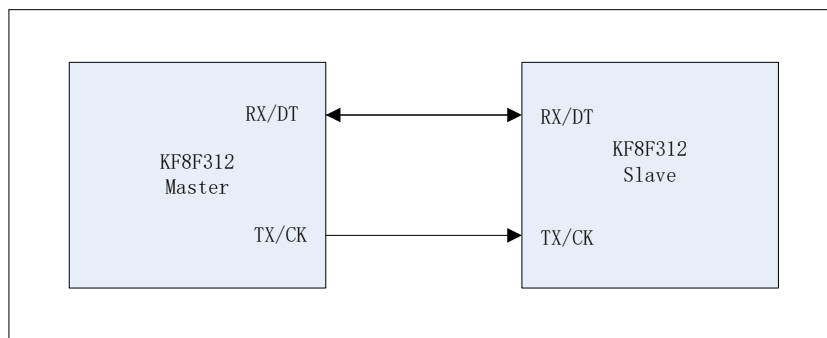


图 10.14 SPI 单线传输图

当和其它具有 SPI 接口的单片机或设备进行通信时，只能单线传输，以下是 KF8F312M4 作为 Master 的两种方式，作为 Slave 的两种方式原理相同。

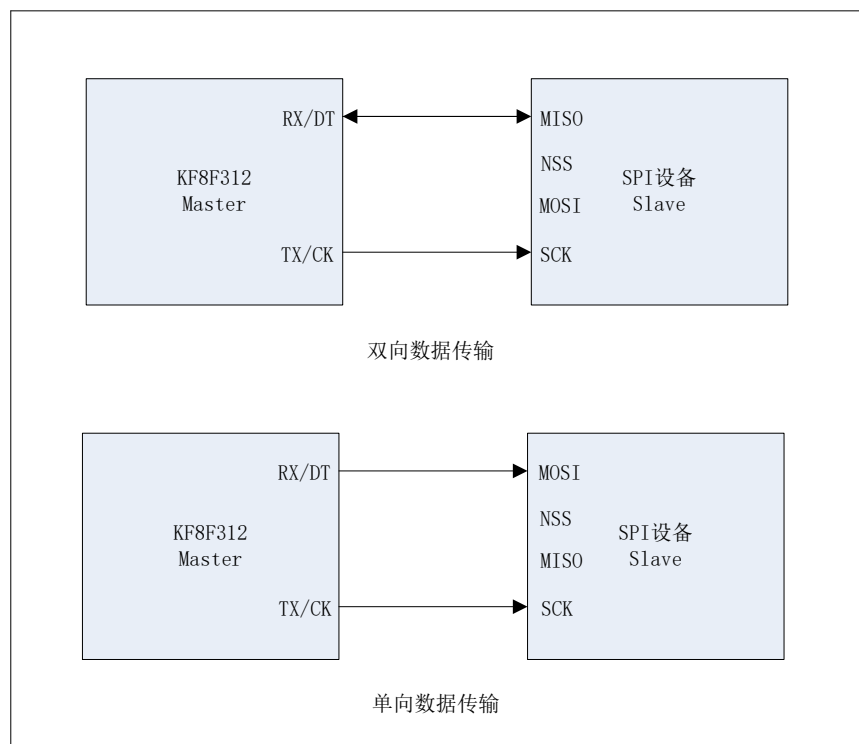


图 10.15 SPI 单线传输图

11 复位

KF8F312M4 具有:上电复位(POR)、WDT 复位、RST 复位和欠压检测复位(LVR)四种复位方式。

有些寄存器的状态在任何复位条件下都不会受到影响,而在其它复位发生时其状态将保持不变。其它大多数寄存器在复位事件发生时将被复位成“复位状态”。图 11.1 给出了片内复位电路的简化结构方框图。

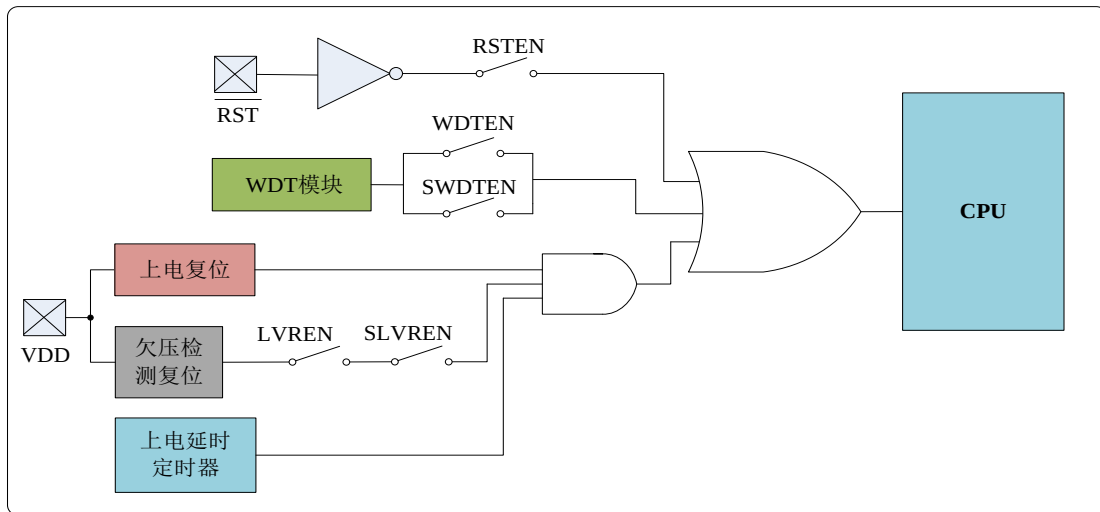


图 11.1 片内复位电路简化框图

注：上电延时定时器只对上电复位(POR)和欠压复位(LVR)有效

11.1 电源控制状态寄存器(PCTL)

如寄存器 11.1 所示， $\overline{\text{LVR}}$ 位的状态在单片机上电复位时是不确定的。如果用户在使用中要用到该位，在程序初始化部分需将其置 1，随后如果有复位发生且 $\overline{\text{LVR}}=0$ ，则表示发生过欠压检测复位。 $\overline{\text{LVR}}$ 状态位是“无关”位，如果欠压检测电路被关闭(通过设定配置字中的 LVREN 位和 PCTL 中的 SLVREN 位)， $\overline{\text{LVR}}$ 状态位是不可预知的。 $\overline{\text{POR}}$ 是上电复位状态位，该位在上电复位时被清 0，在其它情况下不受影响。

寄存器11.1: PCTL: 电源控制寄存器(地址:2EH)

	bit7						bit0	
复位值 00-1 --xx	-	-	-	SLVREN	-	-	$\overline{\text{POR}}$	$\overline{\text{LVR}}$
	0	0	U	R/W	U	U	R/W	R/W

SLVREN: 软件欠压检测使能位

SLVREN =1 使能欠压检测

SLVREN =0 禁止欠压检测

$\overline{\text{POR}}$: 上电复位状态位

$\overline{\text{POR}}=1$ 未发生上电复位

$\overline{\text{POR}}=0$ 发生了上电复位

$\overline{\text{LVR}}$: 欠压复位状态位

$\overline{\text{LVR}}=1$ 未发生欠压复位

$\overline{\text{LVR}}=0$ 已发生欠压复位

注: bit7 和 bit6 位必须强制设置成 0

11.2 上电复位(POR)

在 VDD 达到适合单片机正常工作的电平之前，片内上电复位电路使单片机保持在复位状态，直到 VDD 达到正常工作电平之后单片机才开始正常工作。KF8F312M4 的上电复位时间为 70ms 左右。

11.3 WDT复位

看门狗定时器有一个独立的时钟源，因此单片机在正常工作和休眠模式下都可以正常工作。在单片机正常工作且打开看门狗后，当看门狗计数器计满后产生溢出，将使单片机复位。

在休眠模式下，WDT 也可以正常工作，当 WDT 定时器计满溢出后，将会使单片机从休眠模式唤醒转入正常工作模式，在休眠模式不会对各寄存器复位。

11.4 RST复位

使能外部 RST 复位（配置位 RSTEN=1）后，当引脚 P0.3/ $\overline{\text{RST}}$ 输入复位信号，不管单片机工作在正常模式还是休眠模式，均会使单片机复位。通过在编程时将 P0.3 引脚配置为 $\overline{\text{RST}}$ 复位引脚，即可打开 $\overline{\text{RST}}$ 复位。

在 $\overline{\text{RST}}$ 复位时，KF8F312M4 器件有一个噪声滤波器用于检测和滤除小脉冲，图 11.2 是建议 $\overline{\text{RST}}$ 复位电路。

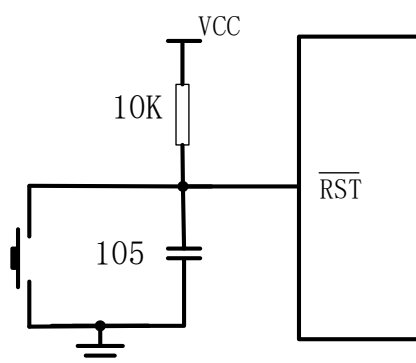


图 11.2 建议 $\overline{\text{RST}}$ 复位电路

11.5 欠压检测复位(LVR)

KF8F312M4 系列中的单片机具有片内欠压检测复位电路。通过编程时设定配置位中的 LVREN 位可以禁止/使能(清 0/置 1)欠压检测复位电路，当配置位中的 LVREN 位被使能后，用户还要在软件中设置 PCTL 中的 SLVREN 位来禁止/使能(清 0/置 1)欠压检测复位电路。

如果 VDD 跌落至 V_{LVR} 以下且持续时间大于 TLVR (TLVR 大于 10us)，欠压检测电路将使单片机复位，单片机保持复位状态直到 VDD 上升到 V_{LVR} 以上，此时上电延时定时器启动，使器件在随后 70ms 左右的延时时间处于延时复位状态，过了 70ms 以后单片机开始正常工作。

如果 VDD 跌落至 V_{LVR} 以下的时间小于规定参数(V_{LVR})，将不保证可产生复位。

如果在上电延时定时器运行过程中发生 VDD 跌落至 V_{LVR} 以下的情况，器件将返回欠压检测复位状态且上电延时定时器被重新初始化。直到 VDD 上升至 V_{LVR} 以上时，上电延时定时器启动一个 70ms 的复位延时，如在延时器件没有欠压发生，单片机会退出复位状态开始正常工作。

11.6 上电延时定时器

上电延时定时器仅在器件上电复位或欠压检测复位发生后提供一个长度为 70ms(标称值)的固定延时时间。上电延时定时器的定时时钟为系统内部低频振荡器。只要单片机产生上电复位或欠压检测复位，单片机就会在上电复位或欠压检测复位发生后保持复位状态 70ms。上电延时定时器使单片机在 VDD 上升到适当电平后才投入正常运行。由于 VDD、温度、制造工艺、内部震荡器频率等的变化，不同单片机的上电延时时间有所差异。

11.7 不同复位条件下对寄存器的影响

表 11.1 寄存器在各种复位发生后的状态

寄存器	地址	上电复位	RST 复位 欠压检测	中断唤醒 WDT 超时唤醒
T0	01H	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	02H	0000 0000	0000 0000	PC+1
PSW	03H	0001 1xxx	000q quuu	uuuq quuu
P0	05H	--xx xxxx	--xx xxxx	--uu uuuu
P2	06H	xxxx ----	xxxx ----	uuuu ----
P1	07H	xxxx xxxx	xxxx xxxx	uuuu uuuu
PCH	0AH	---0 0000	---0 0000	---u uuuu
INTCTL	0BH	0000 0000	0000 0000	uuuu uuuu
EIF1	0CH	0000 0000	0000 0000	uuuu uuuu
EIF2	0DH	-000 ----	-000 ----	-uuu ----
T1L	0EH	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1H	0FH	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1CTL	10H	-000 0000	-uuu uuuu	-uuu uuuu
T2	11H	0000 0000	0000 0000	uuuu uuuu
T2CTL	12H	-000 0000	-000 0000	-000 0000
PWM1L	13H	xxxx xxxx	uuuu uuuu	uuuu uuuu
PWM1H	14H	xxxx xxxx	uuuu uuuu	uuuu uuuu
PWMCTL	15H	11-- --00	11-- --00	uu-- --uu
PP1	16H	1111 1111	1111 1111	uuuu uuuu
CMCTL0	19H	--00 00--	--00 00--	--uu uu--
CMCTL1	1AH	0000 0000	0000 0000	uuuu uuuu
ANSEH	1DH	---- 0000	---- 0000	---- uuuu
ADCDATAH	1EH	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCCTL0	1FH	0000 0000	0000 0000	uuuu uuuu
OPTR	21H	1111 1111	1111 1111	uuuu uuuu
TR0	25H	1111 1111	1111 1111	uuuu uuuu
TR2	26H	1111 1111	1111 1111	uuuu uuuu
TR1	27H	1111 1111	1111 1111	uuuu uuuu
EIE1	2CH	0000 0000	0000 0000	uuuu uuuu
EIE2	2DH	-000 ----	-000 ----	-uuu ----
PCTL	2EH	0001 000x	0001 000x	uuuu uuuu
OSCCTL	2FH	0011 ----	0011 ----	uuuu ----
OSCCAL0	30H	1000 0000	1000 0000	uuuu uuuu
ANSEL	31H	0000 0000	0000 0000	uuuu uuuu
PP2	32H	1111 1111	1111 1111	uuuu uuuu
PWM2L	33H	xxxx xxxx	uuuu uuuu	uuuu uuuu
PWM2H	34H	xxxx xxxx	uuuu uuuu	uuuu uuuu
PUR	35H	1111 -111	1111 -111	uuuu -uuu
IOCL	36H	0000 0000	0000 0000	uuuu uuuu
OSCCAL1	37H	0000 -100	0000 -100	uuuu -uuu
BADDRH	3AH	---0 0000	---0 0000	---u uuuu
BADDRL	3BH	0000 0000	0000 0000	uuuu uuuu
EECTL1	3CH	---- x000	---- x000	---- uuuu
EECTL2	3DH	---- ----	---- ----	---- ----
ADCDATAL	3EH	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCCTL1	3FH	-000 00--	-000 00--	-uuu uu--

寄存器	地址	上电复位	RST 复位 欠压检测	中断唤醒 WDT 超时唤醒
PP3	52H	1111 1111	1111 1111	uuuu uuuu
T2CCR	54H	0000 0000	0000 0000	uuuu uuuu
PWM3L	55H	xxxx xxxx	uuuu uuuu	uuuu uuuu
PWM3H	56H	xxxx xxxx	uuuu uuuu	uuuu uuuu
PWM3CTL0	57H	0000 0000	0000 0000	uuuu uuuu
RSCTL	58H	0000 0000	0000 0000	uuuu uuuu
TXSDR	59H	0000 0000	0000 0000	uuuu uuuu
RXSDR	5AH	0000 0000	0000 0000	uuuu uuuu
PWM3CTL1	5BH	0000 0000	0000 0000	uuuu uuuu
P3ASCTL	5CH	0000 0000	0000 0000	uuuu uuuu
PATRCTL	5DH	---0 0001	---0 0000	---u uuuu
BRCTL	77H	00-0 0-00	00-0 0-00	uu-u u-uu
TSCTL	78H	0000 0000	0000 0000	uuuu uuuu
EUBRGL	79H	0000 0000	0000 0000	uuuu uuuu
EUBRGH	7AH	0000 0000	0000 0000	uuuu uuuu

注：u=不变； x=未知值； -=未用，读作 0； q=该位取决于当前条件。

表 11.2 不同复位条件下对标志位的影响

$\overline{\text{POR}}$	$\overline{\text{LVR}}$	$\overline{\text{TO}}$	$\overline{\text{PD}}$	复位条件
0	u	1	1	上电复位
1	0	1	1	欠压检测复位
u	u	0	u	WDT 复位
u	u	0	0	WDT 唤醒
u	u	u	u	正常操作中的 $\overline{\text{RST}}$ 复位
u	u	1	0	休眠模式中的 $\overline{\text{RST}}$ 复位

图注：u=未发生变化

12 休眠模式

当单片机空闲的时候，为使其功耗降到最低，可以将其转入休眠模式。通过执行一条 IDLE 指令即可进入休眠模式。

为使这种方式下的电流消耗降至最低，应使所有 I/O 口状态确定，如果有的端口没有使用，最好设置为输入，接到 VDD 或 VSS 上，如果没用的端口悬空，应设置为输出，以确保 I/O 引脚没有耗散电流产生，其他在休眠时不用的外设都要关闭。

注：在单片机正常工作时，通常有些引脚用不到，有的用户可能会直接将其悬空。为了减小单片机的功耗，应该将不用的引脚设置为数字输出。如果是 P0 口的引脚则可打开上拉电阻或者设置为数字输出皆可。

单片机进入休眠模式一段时间后由于工作的需要，要将单片机从休眠模式唤醒，在 KF8F312M4 中可通过以下方式将单片机从休眠模式唤醒：

1. RST 引脚上输入的外部复位
2. 看门狗定时器唤醒(如果 WDT 已被使能)
3. INT0/INT1/INT2 外部中断
4. T1 异步计数中断
5. P0 口电平变化中断/外设中断。

RST 引脚输入的复位信号在唤醒单片机的同时也将导致单片机复位。其它唤醒时将单片机从休眠模式唤醒，并不会导致复位。可通过状态寄存器中的 $\overline{TO}$ 和 $\overline{PD}$ 位来确定单片机唤醒的原因。上电时 $\overline{PD}$ 位将被置 1，而当器件从休眠模式唤醒时，该位将被清 0。 $\overline{TO}$ 位则在 WDT 唤醒发生时被清 0。

在使用后三种方式唤醒时，必须使能相应的中断使能位，唤醒与 AIE 位的状态无关。如果 AIE 位被清 0，单片机被唤醒后将继续执行 IDLE 指令后面的指令。如果 AIE 位被置 1，单片机执行 IDLE 指令后面一条指令后进入中断子程序。如果不希望执行 IDLE 指令后面的那条指令直接进入中断子程序，在 IDLE 指令加一条 NOP 指令即可。

13 看门狗定时器

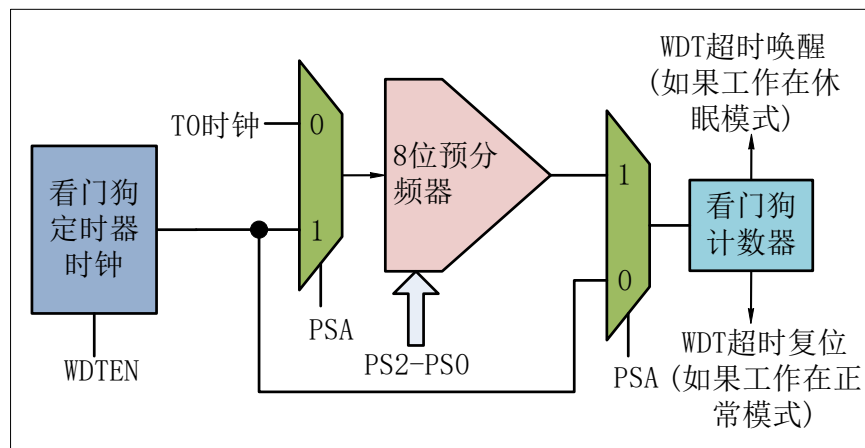
为了防止单片机在正常工作时程序跑飞，KF8F312M4 提供一个看门狗定时器，单片机正常工作时，当看门狗定时器定时时间达到超时时间后，会使单片机产生复位。

看门狗定时器使用片内看门狗专用 RC 振荡器，因此它无需外接任何器件，在休眠模式仍能正常运行。在正常运行时，WDT 超时事件将使单片机产生一次复位。如果单片机处于休眠模式，WDT 超时事件将唤醒单片机并使其继续执行 IDLE 后面的指令。通过将配置位 WDTE 清 0/置 1，可关闭/打开 WDT。

WDT 周期:

WDT 不使用预分频器时超时时间为 18ms。由于温度、电源电压和工艺等的差异，不同器件之间的超时周期稍有不同。通过软件将 OPTR 寄存器的 PSA 位置 1，可将预分频器分配给 WDT。设置 PS<2:0>选择预分频器的分频比,分频比可选择为:1/1、1/2、1/4、1/8、1/16、1/32、1/64、1/128。使用预分频器时，最长超时时间可达 2.3 秒。

为了防止在正常工作时看门狗超时复位，要在固定的时间内对看门狗定时器清 0。执行 CWDT 和 IDLE 指令后会使看门狗从 0 开始重新计数，当看门狗定时器出现超时，状态寄存器中的 $\overline{TO}$ 位将被清 0。



13.1 看门狗定时器框图

14 电气规范

14.1 极限参数值

极限参数值		
序号	参数说明	参数范围
1	偏置电压下的环境温度	-40℃~85℃
2	储存温度	-60℃~150℃
3	VDD 相对于VSS 的电压	5.5V
4	MODE 相对于Vss 的电压	12.5V
5	其它引脚相对于VSS 的电压	5.5V
6	VSS 引脚的最大输出电流	85mA
7	VDD 引脚的最大输入电流	85mA
8	任一I/O 引脚的最大输出灌电流	15mA
9	任一I/O 引脚的最大输出拉电流	15mA
10	I/O口 的最大灌电流	80mA
11	I/O口 的最大拉电流	80mA

备注：如果器件的工作条件超过“最大值”，可能会对器件造成永久性损坏。上述值仅为运行条件极大值，建议不要使器件在该规范规定的范围以外运行。器件长时间工作在最大值条件下，其稳定性会受到影响。

14.2 静态电流特性

表14.1芯片静态电流（IDD）特性

测试条件: 25°C						
序号	测试条件		最小值	典型值	最大值	单位
	振荡频率	VDD (V)				
1	Fosc=4MHz	2.5	—	255	359	uA
		3.0	—	337	464	
		4.0	—	550	737	
		5.0	—	819	1102	
2	Fosc =2MHz	2.5	—	154	223	
		3.0	—	198	282	
		4.0	—	314	430	
		5.0	—	461	648	
3	Fosc =1MHz	2.5	—	104	159	
		3.0	—	129	193	
		4.0	—	196	279	
		5.0	—	282	421	
4	Fosc =500kHz	2.5	—	78	116	
		3.0	—	94	144	
		4.0	—	136	201	
		5.0	—	192	305	
5	Fosc =250kHz	2.5	—	67	101	
		3.0	—	77	122	
		4.0	—	107	166	
		5.0	—	147	250	
6	Fosc =125kHz	2.5	—	60	91	
		3.0	—	69	114	
		4.0	—	92	144	
		5.0	—	125	219	
7	Fosc =62.5kHz	2.5	—	57	88	
		3.0	—	64	106	
		4.0	—	85	136	
		5.0	—	114	206	
8	Fosc =15.625kHz	2.5	—	55	83	
		3.0	—	61	102	
		4.0	—	79	130	
		5.0	—	106	195	

注 1: 在正常的工作模式下, IDD 测量的条件为: 所有I/O 引脚均设置为输出低, RST = Vss , 禁止 WDT, 关闭时钟输出。

2: 供电电流主要随工作电压和频率而变化。其它因素, 如I/O 引脚负载和开关速率、内部代码执行模式和温度也会影响电流消耗。

14.3 外设电流特性

表14.2芯片外设电流特性

测试条件: 25°C							
序号	测试参数	测试条件		最小值	典型值	最大值	单位
			VDD(V)				
1	休眠电流 (IPD)	WDT、BOR、比较器等 外设被禁止	2.5	—	0.1	0.4	μA
			3.0	—	0.22	0.6	
			4.0	—	0.53	1.1	
			5.0	—	0.94	1.7	
2	WDT 电流 (IWDT)		2.5	—	1.2	1.6	
			3.0	—	2.5	3.4	
			4.0	—	7.2	9.6	
			5.0	—	14.6	19.4	
3	欠电压复位电 流 (ILVR)		2.5	—	9.9	14.3	
			3.0	—	12.6	17.3	
			4.0	—	18.2	24.8	
			5.0	—	24.3	32.6	
4	比较器电流 (ICMP)	都使能单个比较器	2.5	—	17	23	
			3.0	—	21	29	
			4.0	—	31	42	
			5.0	—	41	54	
5	弱上拉电流 (IPUR)	单个 IO 口	2.5	—	64.96	85	
			3.0	—	103.74	136	
			4.0	—	198.42	260	
			5.0	—	312.9	410	

- 注 1: 外设电流是基本IDD 或IPD 电流以及相应外设使能时消耗的额外电流的总和。外设电流可以从此电流中减去基本IDD 或IPD 电流得出。
- 2: 休眠电流与振荡器类型无关。掉电电流是在器件休眠时, 所有I/O 引脚设置为输出低, RST = Vss ; 禁止WDT, 关闭时钟输出时测得的。
- 3: 外设电流还可能受到温度的影响。

14.4 I/O端口电平和芯片供电电压特性

表 14.3 芯片 IO 端口电平特性

工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (工业级)						
符号	参数说明	测试条件	最小值	典型值	最大值	单位
V _{IL}	输入低电平 I/O引脚 采用TTL缓冲器	$4.5\text{V} \leq \text{VDD} \leq 5.5\text{V}$	V _{SS}	—	0.7	V
		$2.5\text{V} \leq \text{VDD} \leq 4.5\text{V}$	V _{SS}	—	0.15VDD	V
	采用施密特缓冲触发器	$2.5\text{V} \leq \text{VDD} \leq 5.5\text{V}$	V _{SS}	—	0.2VDD	V
V _{IH}	输入高电平 I/O端口 采用TTL缓冲器		VDD - 0.7	—	VDD	V
	采用施密特缓冲触发器		0.6VDD	—	VDD	V

表 14.4 芯片供电电压特性

工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (工业级)						
符号	参数说明	测试条件	最小值	典型值	最大值	单位
VDD	电源电压	FOSC $\leq 4\text{ MHz}$	2.3	—	5.5	V
VLVR	VDD 起始电压确保能够产生欠压复位信号		2.3	—	2.65	V
VPOR	VDD 起始电压确保能够产生内部上电复位信号		—	2.3	—	V

14.5 比较器模块规范

表14.5 比较器模块规范

测试条件（特别声明除外）： VDD=5.0V, VSS=0V, T=25℃ 工作温度 -40℃≤TA≤+85℃（工业级）							
参数编号	符号	特性	最小值	典型值	最大值	单位	备注
1	V _{OS}	输入失调电压	-	±5	-	mV	
2	V _{ICM} *	输入共模电压	0	-	V _{DD} -1.5	V	
3	C _{MRR} *	共模抑制比	+55	-	-	db	
4	T _{RESP} *	响应时间上升沿	-	150	-	ns	
5		响应时间下降沿	-	200	-	ns	

*该数据为设计值

14.6 A/D 转换器（ADC）特性

表14.6 A/D 转换器（ADC）特性

工作温度 -40℃≤TA≤+85℃（工业级）						
符号	参数说明	测试条件	最小值	典型值	最大值	单位
N _R *	分辨率	—	—	—	10	位
E _{IL} *	积分误差	V _{REF} = 5V, V _{DD} =5.0V, T=25℃	—	—	±1	LSB
E _{DL} *	微分误差	V _{REF} = 5V, V _{DD} =5.0V, T=25℃	—	—	±1	LSB
E _{OFF} *	失调误差	V _{REF} = 5V, V _{DD} =5.0V, T=25℃	—	—	±1	LSB
E _{GN} *	增益误差	V _{REF} = 5V, V _{DD} =5.0V, T=25℃	—	—	±1	LSB
V _{REF} *	参考电压	保证1个LSb的精度	2	—	V _{DD}	V
V _{AIN} *	满量程范围	—	V _{SS}	—	V _{REF}	V
TCNV*	AD转换时间	A/D控制寄存器ADCCTL0中的START位被清零	—	11	—	T _{AD}

*该数据为设计值

15 直流特性图表

备注：某些图表中的数据超出了规定的工作范围（即超出了规定的VDD 范围），这些图表仅供参考，器件只有在规定的范围下工作才可以确保正常运行。

图15-1： 不同VDD 时典型IDD — FOSC 关系曲线图

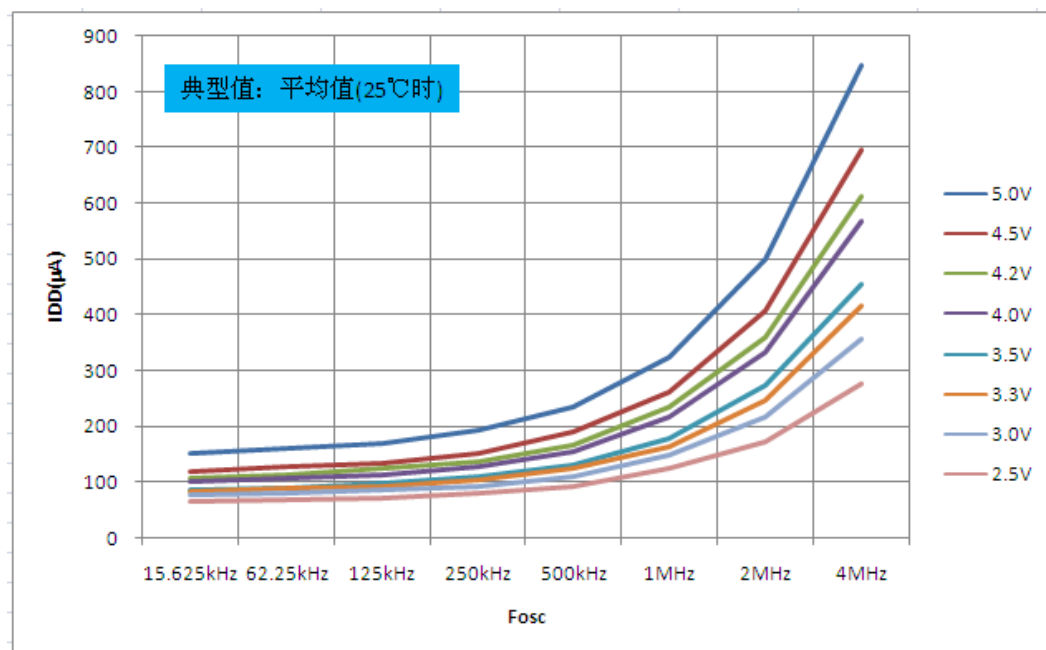


图15-2： 不同VDD 时典型IPD — VDD 关系曲线图

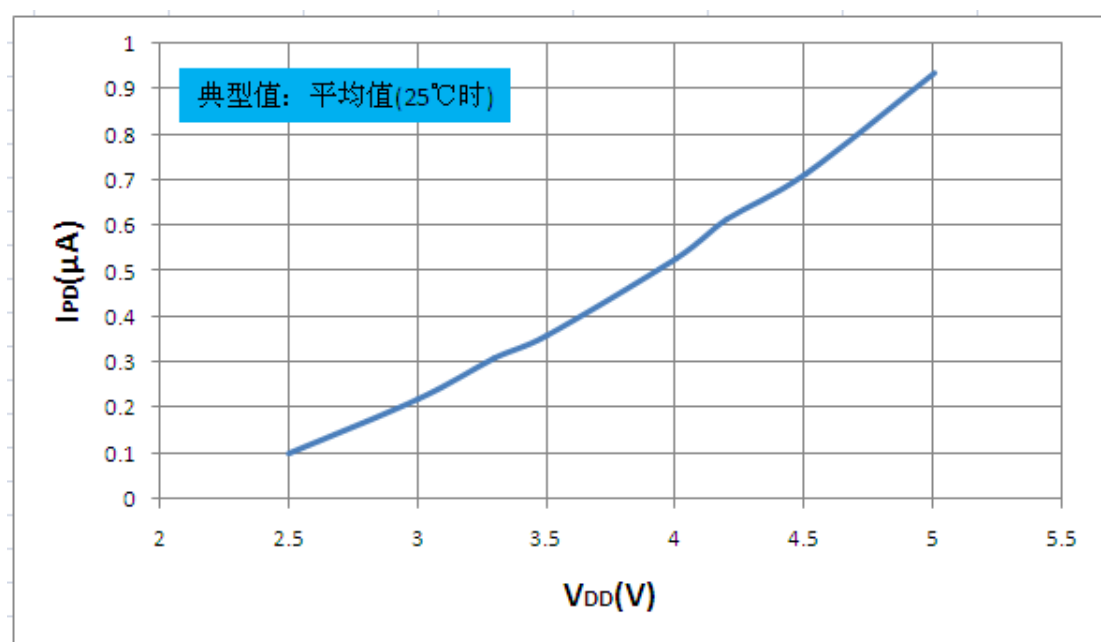


图15-3: 弱上拉电流 I_{PUR} — V_{DD} 关系曲线图

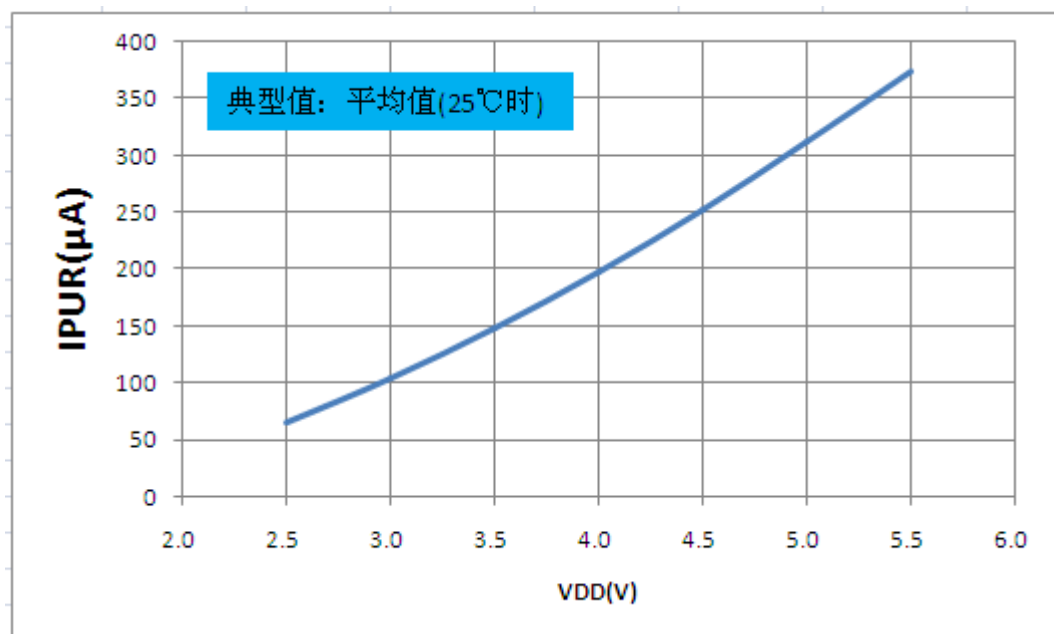
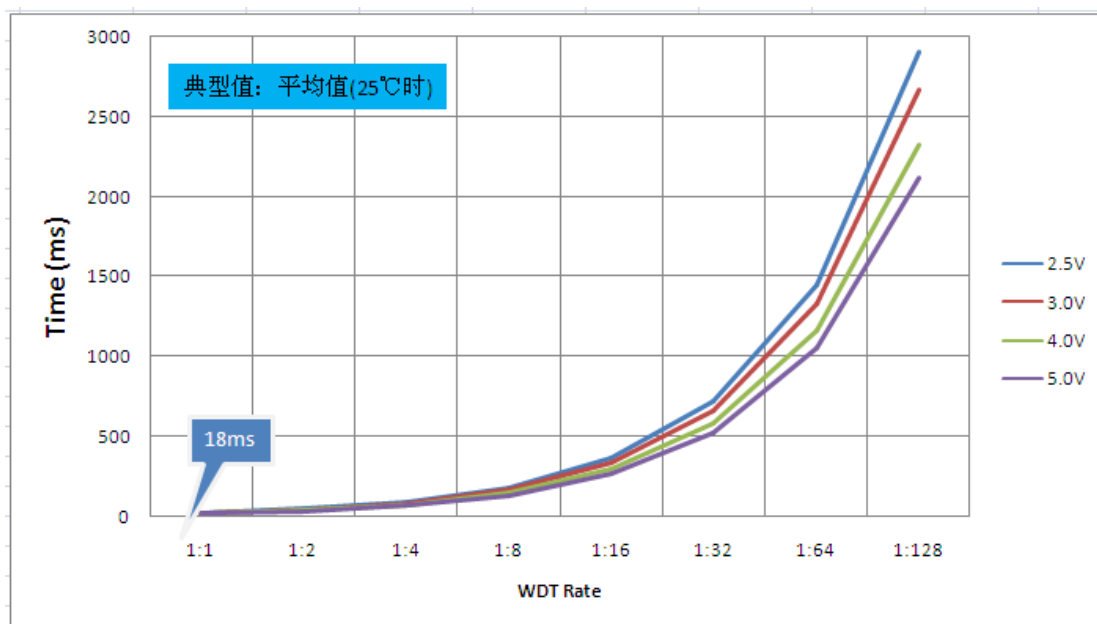


图15-4: 不同 V_{DD} 时看门狗预分频比 — 周期关系曲线图



备注：1：1分频时看门狗周期为18ms。

图15-5: 看门狗电流 — VDD关系曲线图

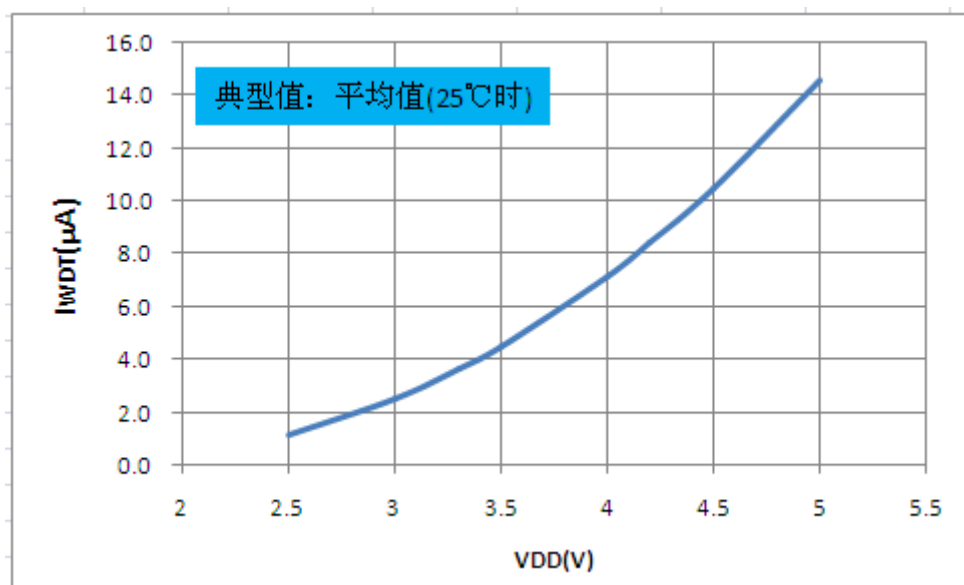


图15-6: 比较器电流 — VDD关系曲线图（使能一路比较器）

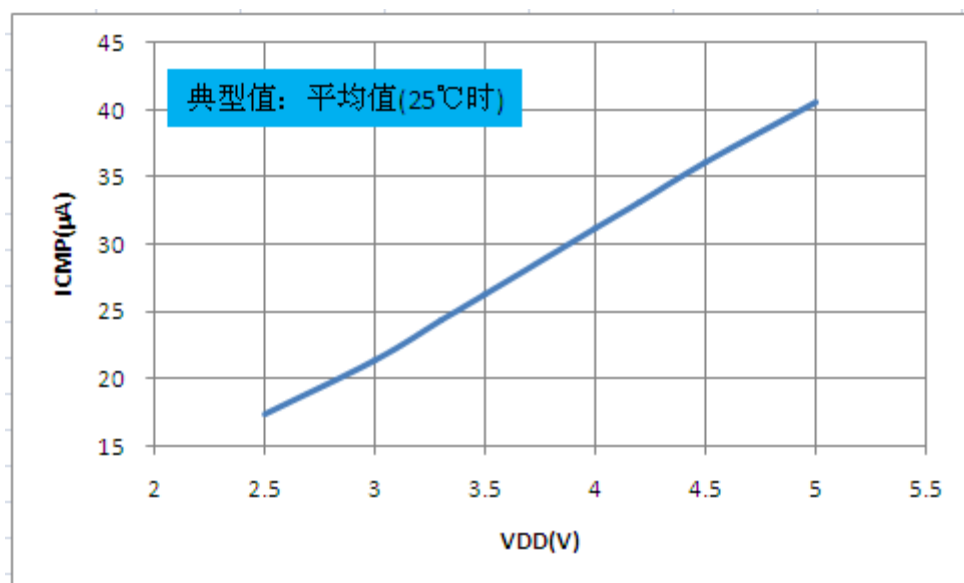


图 15-7: 欠压复位电流 I_{LVR} — V_{DD} 关系曲线图

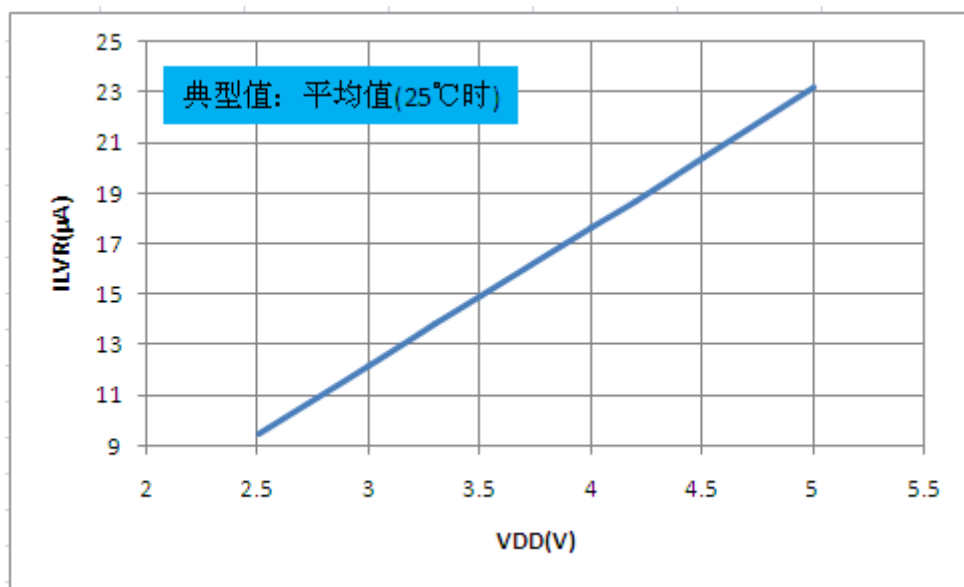


图15-8: 不同温度时 V_{OH} — I_{OH} 关系曲线图 ($V_{DD} = 5.0V$)

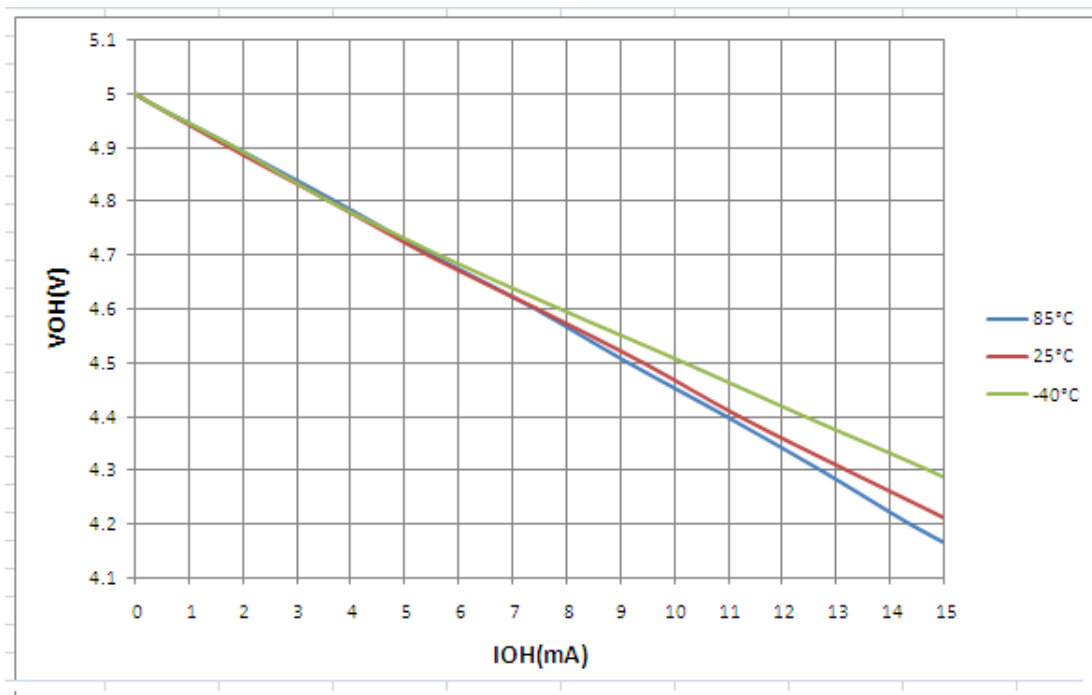


图15-9: 不同温度时 V_{OH} — I_{OH} 关系曲线图 ($V_{DD} = 3.0V$)

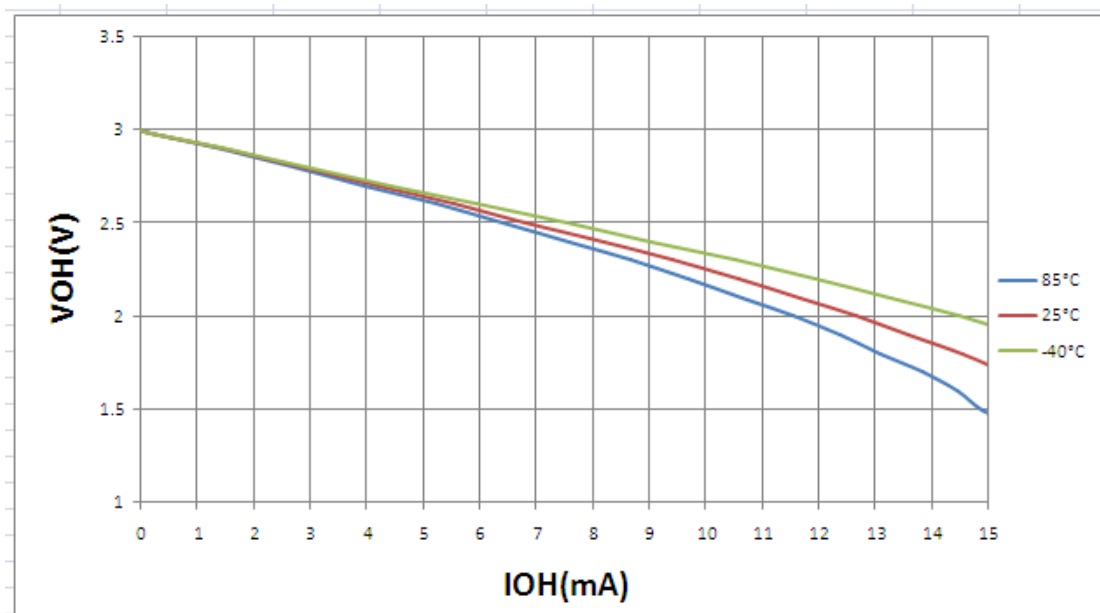


图15-10: 不同温度时 V_{OH} — I_{OL} 关系曲线图 ($V_{DD} = 5.0V$)

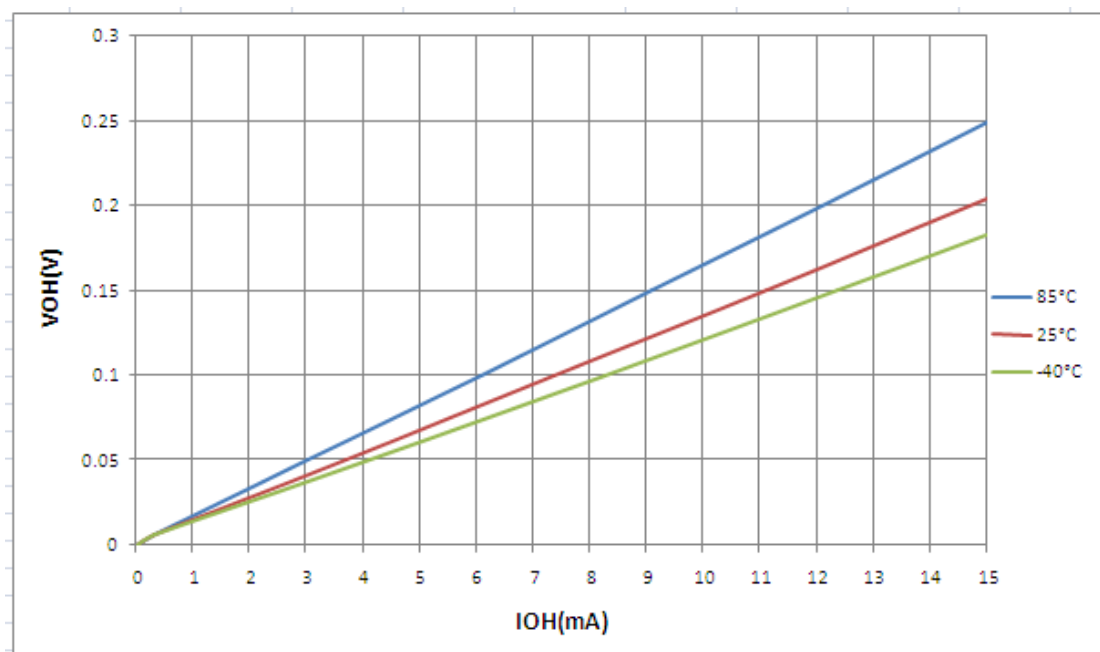


图15-11: 不同温度时 V_{OH} — I_{OL} 关系曲线图 ($V_{DD} = 3.0V$)

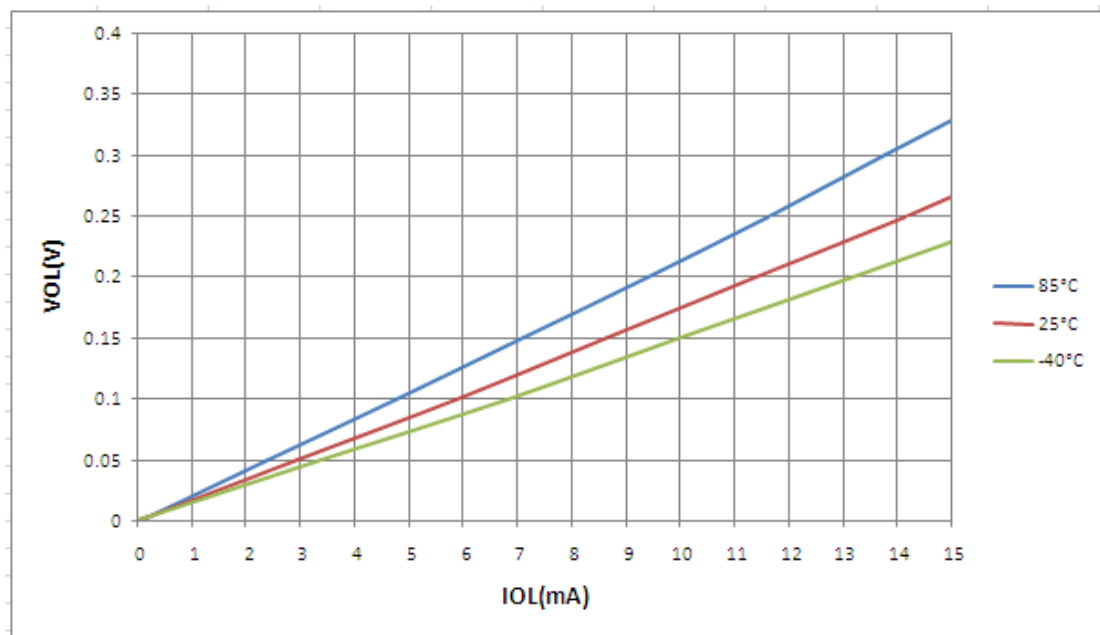


图15-12: 上拉功能打开时端口电压 V_{OP} — 电流 I_{OP} 关系曲线图 ($V_{DD} = 5.0V$)

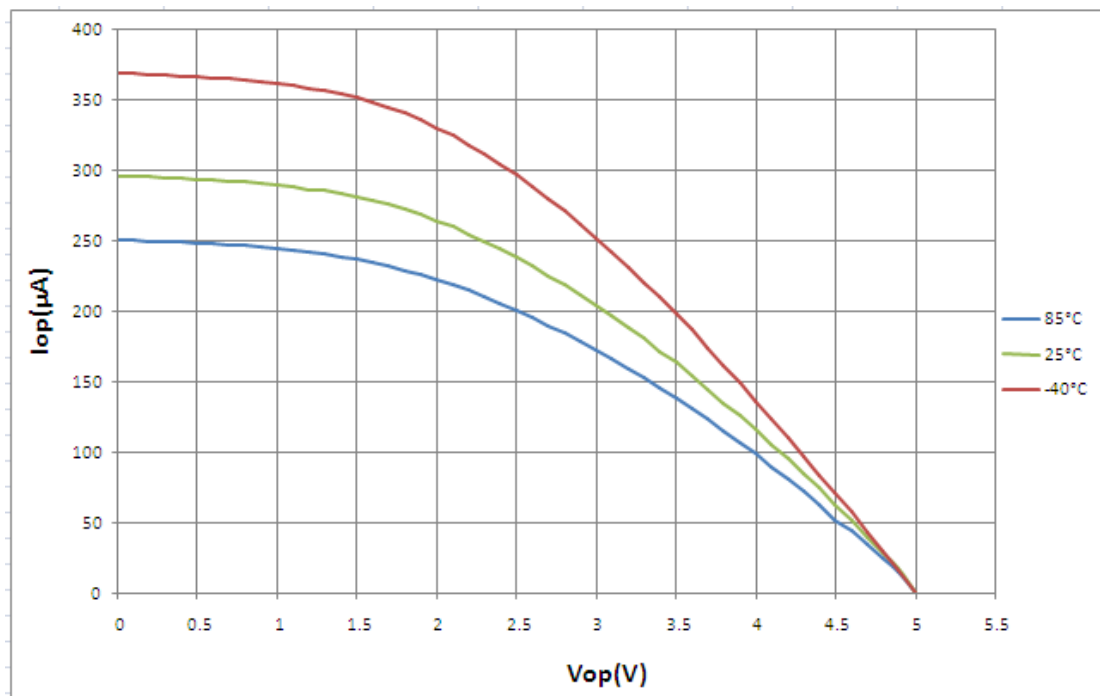
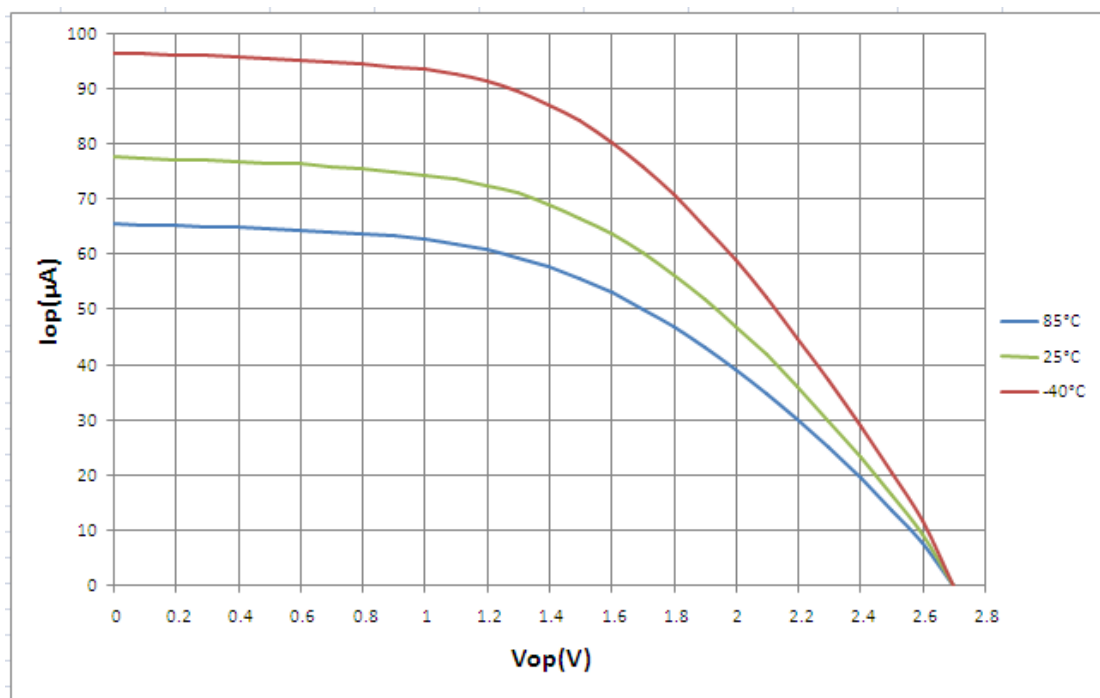
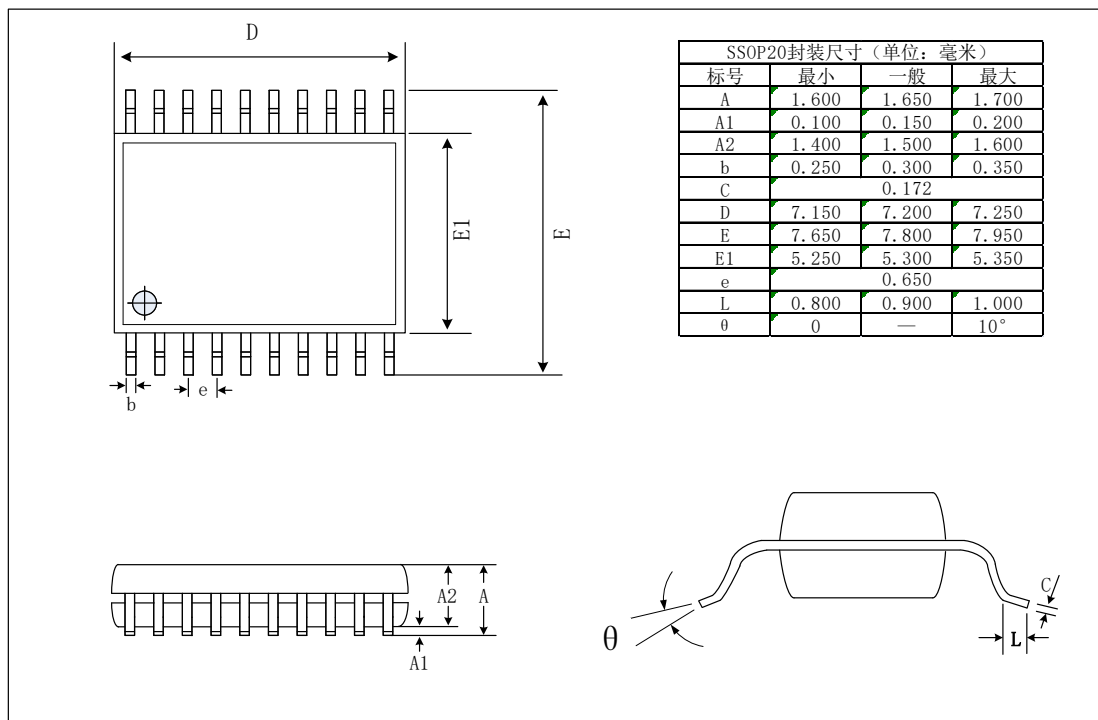


图15-13: 上拉功能打开时端口电压 V_{OP} — 电流 I_{OP} 关系曲线图 ($V_{DD} = 2.7V$)

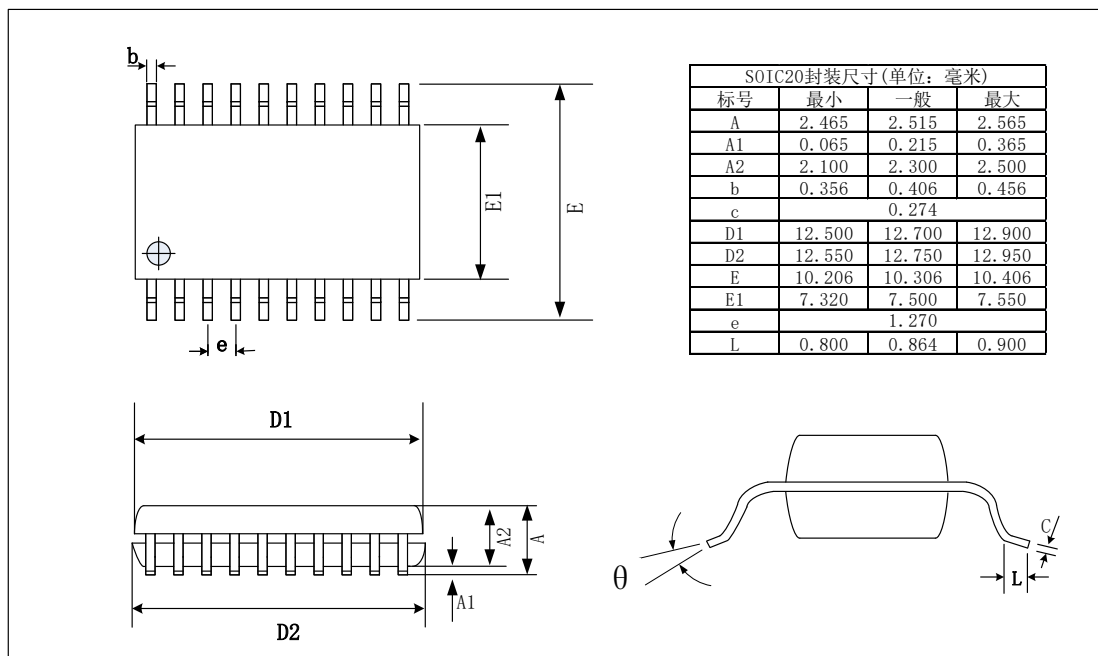


16 封装信息

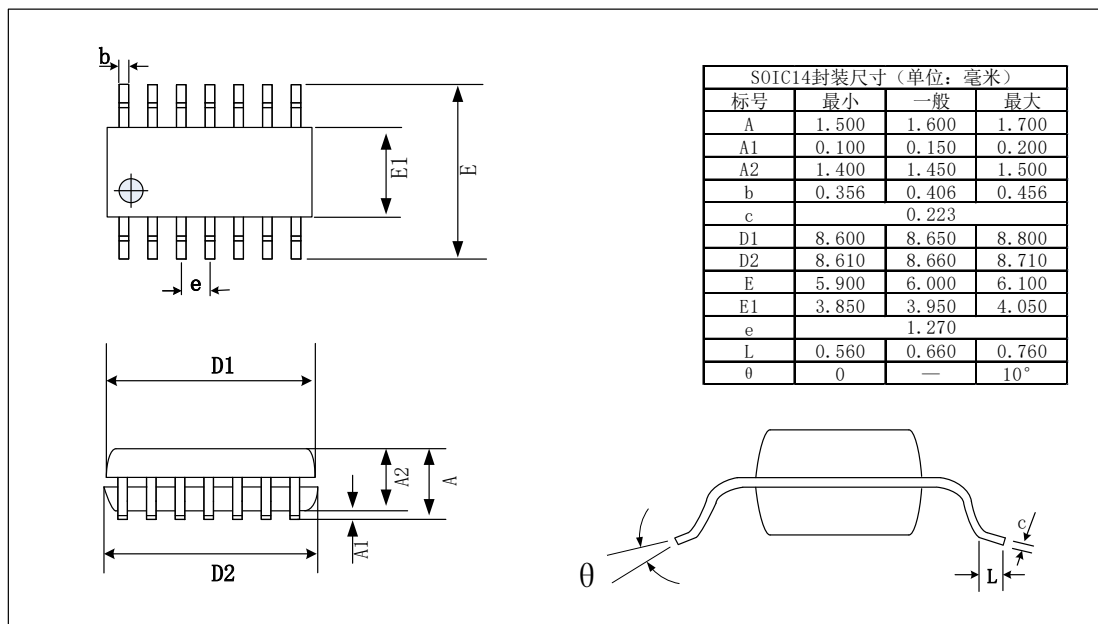
20 脚 SSOP 封装



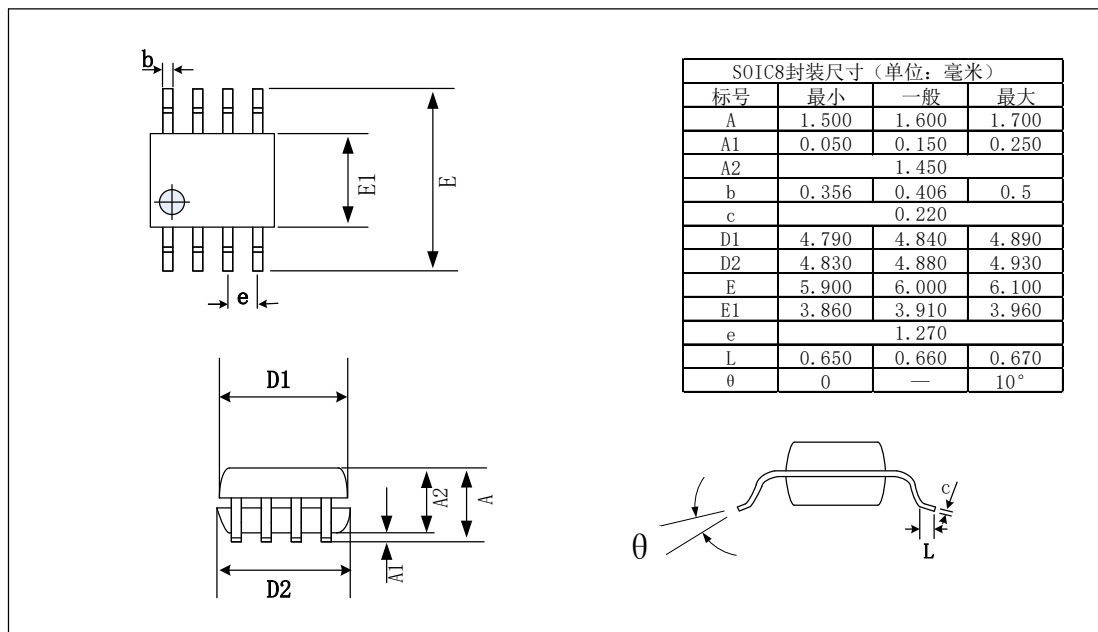
20 脚 SOIC 封装



14 脚 SOIC 封装



8 脚 SOIC 封装



附录 1 KF8F312M4 SFR地址映射及功能汇总

地址	名称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	复位初值
01H	T0	定时/计数器 0(T0) 寄存器								xxxx xxxx
02H	PCL	程序计数器(PC) 低字节								0000 0000
03H	PSW	-	-	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	CY	0001 1xxx
05H	P0	-	-	P05	P04	P03	P02	P01	P00	-- xx xxxx
06H	P2	P27	P26	P25	P24	-	-	-	-	xxxx ----
07H	P1	P17	P16	P15	P14	P13	P12	P11	P10	xxxx xxxx
0AH	PCH	-	-	-	程序计数器(PC) 高字节					---0 0000
0BH	INTCTL	AIE	PUIE	TOIE	INT0IE	POIE	TOIF	INT0IF	POIF	0000 0000
0CH	EIF1	-	ADIF	INT2IF	INT1IF	C1IF	PWM2IF	T2IF	T1IF	0000 0000
0DH	EIF2	-	C2IF	RXIF	TXIF	-	-	-	-	-000 ----
0EH	T1L	定时/计数器 T1 低字节寄存器								xxxx xxxx
0FH	T1H	定时/计数器 T1 高字节寄存器								xxxx xxxx
10H	T1CTL	-	T1GC	T1CKS1	T1CKS0	-	$\overline{T1SY}$	T1CS	T1ON	-000 0000
11H	T2	定时/计数器 2(T2) 寄存器								0000 0000
12H	T2CTL	-	T2CKBS3	T2CKBS2	T2CKBS1	T2CKBS0	T2ON	T2CKPS1	T2CKPS0	-000 0000
13H	PWM1L	PWM1 占空比设置寄存器								xxxx xxxx
14H	PWM1H	PWM1 寄存器								xxxx xxxx
15H	PWMCTL	INT2SE	INT1SE	-	-	-	-	PWM2ON	PWM1ON	11-- --00
16H	PP1	PWM1 周期寄存器								1111_1111
19H	CMCTL0	-	-	C2OUT	C1OUT	C2OE	C1OE	-	-	--00 00--
1AH	CMCTL1	-	-	-	-	C2M1	C2M0	C1M1	C1M0	0000 0000
1DH	ANSEH	-	-	-	-	ANS11	ANS10	ANS9	ANS8	---- 0000
1EH	ADCDATAH	ADC 数据寄存器高字节								xxxx xxxx
1FH	ADCCTL0	ADLR	T2CCRON	CHS3	CHS2	CHS1	CHS0	START	ADEN	0000 0000
21H	OPTR	$\overline{PUPH}$	INT0SE	TOCS	TOSE	PSA	PS2	PS1	PS0	1111 1111
25H	TR0	-	-	TR05	TR04	TR03	TR02	TR01	TR00	1111 1111
26H	TR2	TR27	TR26	TR25	TR24					1111 ----
27H	TR1	TR17	TR16	TR15	TR14	TR13	TR12	TR11	TR10	1111 1111
2CH	EIE1	-	ADIE	INT2IE	INT1IE	C1IE	PWM2IE	T2IE	T1IE	0000 0000
2DH	EIE2	-	C2IE	RXIE	TXIE	-	-	-	-	-000 ----
2EH	PCTL	-	-	-	SLVREN	-	-	$\overline{POR}$	$\overline{LVR}$	0001 000x
2FH	OSCCTL	CKOEN	IRCS2	IRCS1	IRCS0	-	-	-	-	0011 ----
30H	OSCCAL0	晶振校准值寄存器 0								1000 0000
31H	ANSEL	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	0000 0000
32H	PP2	PWM2 周期寄存器								1111 1111
33H	PWM2L	PWM2 占空比设置寄存器								xxxx xxxx
34H	PWM2H	PWM2 寄存器								xxxx xxxx
35H	PUR	-	-	PUR5	PUR4	-	PUR2	PUR1	PUR0	1111 -111
36H	IOCL	-	-	IOCL5	IOCL4	IOCL3	IOCL2	IOCL1	IOCL0	0000 0000
37H	OSCCAL1	晶振校准值寄存器 1								0000 -100

地址	名称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	复位初值
3AH	BADDRH	—	—	—	BLOCK EEPROM 地址指针高 5 位					---0 0000
3BH	BADDRL	BLOCK EEPROM 地址指针低 8 位								0000 0000
3CH	EECTL1	BLOCK EEPROM 控制寄存器 1								---- x000
3DH	EECTL2	BLOCK EEPROM 控制寄存器 2								---- ----
3EH	ADCDATA L	ADC 数据寄存器低字节								xxxx xxxx
3FH	ADCCTL1	—	ADCS2	ADCS1	ADCS0	VCFG1	VCFG0	—	—	-000 00--
52H	PP3	PWM3 周期寄存器								1111 1111
54H	T2CCR	T2 触发 AD 启动寄存器								0000 0000
55H	PWM3L	PWM3 占空比寄存器								xxxx xxxx
56H	PWM3H	PWM3 寄存器								xxxx xxxx
57H	PWM3CTL 0	P3M1	P3M0	PDT1	PDT0	P3ON1	P3ON0	PWM3M1	PWM3M0	0000 0000
58H	RSCTL	SPEN	RX9	SRXEN	CRXEN	ADREN	FRER	OVFER	RX9D	0000 0000
59H	TXSDR	USART 发送数据寄存器								0000 0000
5AH	RXSDR	USART 接收数据寄存器								0000 0000
5BH	PWM3CTL 1	PRSEN	PDC6	PDC5	PDC4	PDC3	PDC2	PDC1	PDC0	0000 0000
5CH	P3ASCTL	P3ASE	P3ASS2	P3ASS1	P3ASS0	P3SSAC 1	P3SSAC 0	P3SSBD 1	P3SSBD 0	0000 0000
5DH	PATRCTL	—	—	—	STRSYN	STREND	STRENC	STRENB	STRENA	---0 0001
77H	BRCTL	ABRDOVF	RCIDLF	-	SCKPS	BRG16	-	WUEN	ADRBEN	00-0 0-00
78H	TSCTL	CSRS	TX9	TXEN	SYNC	SENDB	HBRG	TXSRS	TX9D	0000 0000
79H	EUBRGL	BRG7	BRG6	BRG5	BRG4	BRG3	BRG2	BRG1	BRG0	0000 0000
7AH	EUBRGH	BRG15	BRG14	BRG13	BRG12	BRG11	BRG10	BRG9	BRG8	0000 0000

注：“-”表示未用的存储单元 “x”表示不定

附录 2 汇编指令集

助记符、操作数	指令说明	周期	影响标志
NOP	空操作指令	1	
CRET	子程序返回指令	2	
RRET Rn,#data	立即数送到 Rn 中返回	2	
IRET	中断返回指令	2	
CWDT	WDT 清 0	1	
IDLE	进入休眠模式	1	
数据传送指令			
MOV dir	$dir \leftarrow (dir)$	1	Z
MOV Rn,dir	$Rn \leftarrow (dir)$	1	
MOV dir,Rn	$dir \leftarrow (Rn)$	1	
MOV Rn,#data	$Rn \leftarrow data$	1	
MOV Rn,Rs	$Rn \leftarrow (Rs)$	1	
LD Rn,[Rs]	$Rn \leftarrow ((Rs))$	1	
ST [Rn],Rs	$(Rn) \leftarrow (Rs)$	1	
SWAPR Rn,dir	$Rn <7:4> = dir <3:0> \quad Rn <3:0> = dir <7:4>$	1	
SWAP dir	$dir <7:4> = dir <3:0> \quad dir <3:0> = dir <7:4>$	1	
算术运算指令			
ADD Rm,dir	$Rm \leftarrow (Rm) + (dir)$	1	CY、DC、Z
ADD dir,Rm	$dir \leftarrow (Rm) + (dir)$	1	CY、DC、Z
ADD Rn,#data	$Rn \leftarrow (Rn) + data$	1	CY、DC、Z
ADD Rn,Rs	$Rn \leftarrow (Rn) + (Rs)$	1	CY、DC、Z
SUB Rm,dir	$Rm \leftarrow (dir) - (Rm)$	1	CY、DC、Z
SUB dir,Rm	$dir \leftarrow (dir) - (Rm)$	1	CY、DC、Z
SUB Rn,#data	$Rn \leftarrow data - (Rn)$	1	CY、DC、Z
SUB Rn,Rs	$Rn \leftarrow (Rs) - (Rn)$	1	CY、DC、Z
INC dir	$dir \leftarrow (dir) + 1$	1	Z
INCR dir	$R0 \leftarrow (dir) + 1$	1	Z
INC Rn	$Rn \leftarrow (Rn) + 1$	1	Z
DEC dir	$dir \leftarrow (dir) - 1$	1	Z
DECR dir	$R0 \leftarrow (dir) - 1$	1	Z
DEC Rn	$Rn \leftarrow (Rn) - 1$	1	Z
逻辑运算指令			
AND Rm,dir	$Rm \leftarrow (Rm) \wedge (dir)$	1	Z
AND dir,Rm	$dir \leftarrow (dir) \wedge (Rm)$	1	Z
AND Rn,#data	$Rn \leftarrow (Rn) \wedge data$	1	Z
AND Rn,Rs	$Rn \leftarrow (Rn) \wedge (Rs)$	1	Z
ORL Rm,dir	$Rm \leftarrow (Rm) \vee (dir)$	1	Z
ORL dir,Rm	$dir \leftarrow (dir) \vee (Rm)$	1	Z
ORL Rn,#data	$Rn \leftarrow (Rn) \vee data$	1	Z
ORL Rn,Rs	$Rn \leftarrow (Rn) \vee (Rs)$	1	Z
XOR Rm,dir	$Rm \leftarrow (Rm) \oplus (dir)$	1	Z
XOR dir,Rm	$dir \leftarrow (dir) \oplus (Rm)$	1	Z
XOR Rn,#data	$Rn \leftarrow (Rn) \oplus data$	1	Z
XOR Rn,Rs	$Rn \leftarrow (Rn) \oplus (Rs)$	1	Z

助记符、操作数	指令说明	周期	影响标志
CLR Rn	$Rn=0$	1	Z
CLR dir	$dir=0$	1	Z
CPLR dir	$R0 \leftarrow \neg(dir)$	1	Z
CPL dir	$dir \leftarrow \neg(dir)$	1	Z
CPL Rn	$Rn \leftarrow \neg(Rn)$	1	CY
RRCR dir	$R0 \leftarrow (dir)$ 带进位 C 循环右移 1 位	1	CY
RRC dir	$dir \leftarrow (dir)$ 带进位 C 循环右移 1 位	1	CY
RRC Rn	$Rn \leftarrow (Rn)$ 带进位 C 循环右移 1 位	1	CY
RLCR dir	$R0 \leftarrow (dir)$ 带进位 C 循环左移 1 位	1	CY
RLC dir	$dir \leftarrow (dir)$ 带进位 C 循环左移 1 位	1	CY
RLC Rn	$Rn \leftarrow (Rn)$ 带进位 C 循环左移 1 位	1	CY
位操作指令			
CLR dir,b	将 dir 的 b 位清 0	1	
SET dir,b	将 dir 的 b 位置 1	1	
CLR Rn,b	将 Rn 的 b 位清 0	1	
SET Rn,b	将 Rn 的 b 位置 1	1	
转移指令			
DECRJZ dir	$R0 \leftarrow (dir)-1$, 为 0 跳过下一条指令	1/2	
DECJZ dir	$dir \leftarrow (dir)-1$, 为 0 跳过下一条指令	1/2	
DECJZ Rn	$Rn \leftarrow (Rn)-1$, 为 0 跳过下一条指令	1/2	
INCRJZ dir	$R0 \leftarrow (dir)+1$, 为 0 跳过下一条指令	1/2	
INCJZ dir	$dir \leftarrow (dir)+1$, 为 0 跳过下一条指令	1/2	
INCJZ Rn	$Rn \leftarrow (Rn)+1$, 为 0 跳过下一条指令	1/2	
JNB dir,b	dir 的 b 位为 0 跳过下一条指令	1/2	
JB dir,b	dir 的 b 位为 1 跳过下一条指令	1/2	
JNB Rn,b	Rn 的 b 位为 0 跳过下一条指令	1/2	
JB Rn,b	Rn 的 b 位为 1 跳过下一条指令	1/2	
JMP #data12	无条件转移指令	2	
CALL #data12	子程序调用指令	2	

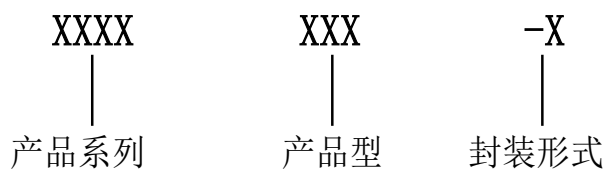
注： dir 为通用寄存器或特殊功能寄存器； Rn、Rs 表示 R0~R7； Rm 表示 R0~R3； #data 表示 8 位立即数； #data12 表示 12 位立即数； b 表示寄存器的第 b 位； [Rn] 表示 Rn 中的数值指向的地址中数据； () 表示特殊功能寄存器、通用数据寄存器或寄存器组中的数据。

附录 3 寄存器全称表

地址	名称	全称
01H	T0	Timer 0
02H	PCL	Program Counter Low
03H	PSW	Program Status Word
05H	P0	Port 0
06H	P2	Port2
07H	P1	Port 1
0AH	PCH	Program Counter High
0BH	INTCTL	Interrupt control
0CH	EIF1	Enable Interrupt Flag 1
0DH	EIF2	Enable Interrupt Flag 2
0EH	T1L	Timer 1 Low
0FH	T1H	Timer 1 High
10H	T1CTL	Timer 1 Control
11H	T2	Timer 2
12H	T2CTL	Timer 2 Control
13H	PWM1L	Pulse-Width Modulation 1 Low
14H	PWM1H	Pulse-Width Modulation 1 High
15H	PWMCTL	Pulse-Width Modulation Control
16H	PP1	Pulse-Width Modulation Periods 1
19H	CMCTL0	Compare Control 0
1AH	CMCTL1	Compare Control 1
1DH	ANSEH	Analog Numbers Select High
1EH	ADCDATAH	Analog Digital Convert Data High
1FH	ADCCTL0	Analog Digital Convert Control 0
21H	OPTR	Opt Register
25H	TR0	Trend Register 0
26H	TR2	Trend Register 2
27H	TR1	Trend Register 1
2CH	EIE1	Enable Interrupt Enable 1
2DH	EIE2	Enable Interrupt Enable 2
2EH	PCTL	Power Control
2FH	OSCCTL	Operation System Crystal Control
30H	OSCCAL0	Operation System Crystal Calibration 0
31H	ANSEL	Analog Numbers Select Low
32H	PP2	Pulse-Width Modulation Periods 2
33H	PWM2L	Pulse-Width Modulation 2 Low
34H	PWM2H	Pulse-Width Modulation 2 High
35H	PUR	Push Register
36H	IOCL	Input Output Control
37H	OSCCAL1	Operation System Crystal Calibration 1
3AH	BADDRH	Buffer Address High
3BH	BADDRL	Buffer Address Low
3CH	EECTL1	EEPROM Control 1
3DH	EECTL2	EEPROM Control 2
3EH	ADCDATAL	Analog Digital Convert Data Low
3FH	ADCCTL1	Analog Digital Convert Control 1
52H	PP3	Pulse-Width Modulation Periods 3
55H	PWM3L	Pulse-Width Modulation 3 Low

56H	PWM3H	Pulse-Width Modulation 3 High
57H	PWM3CTL0	Pulse-Width Modulation 3 Control 0
58H	RSCTL	Receive Status Control
59H	TXSDR	Transmit Data Register
5AH	RXSDR	Receive Data Register
5BH	PWM3CTL1	Pulse-Width Modulation 3 Control 1
5CH	P3ASCTL	Pulse-Width Modulation 3 Auto Shut Control
5DH	PATRCTL	Pulse Auto Turn Control
5EH	OSCLDOCAL	OSC and LDO Calibration
77H	BRCTL	Baud Rate Control
78H	TSCTL	Transmit Control
79H	EUBRGL	Enhance Universal Baud Rate Generator Low
7AH	EUBRGH	Enhance Universal Baud Rate Generator High

产品标识体系



产品系列: KF8F = KF 系列闪存 8 位单片机

产品型号: 312 = 312 型

封装形式: P = DIP
 S = SOIC

示例:

- 1) KF8F312M4-S 表示 KF 系列 312 型贴片式单片机

版本信息

版本	更新描述	更新页码
V1.0	新的数据手册	
V1.1	增加了交流和直流特性参数; 修正了文档中部分叙述	
V1.2	修正了文档中部分叙述和表达	
V1.3	修改了版本订购信息和文档中部分叙述	
V1.4	修改了相关寄存器与地址的匹配性, 增加了模拟比较器用法说明	
V1.5	去掉电容触摸模块章节, 修正了 INTCTL 寄存器的相关描述	
V1.6	修改合并了增加 KF8F312M4 型单片机后的数据手册内容	
V1.7	修正了文档中相关陈述 (第 57/151 页中将“T1IF”修改为“T1IF”)	
V1.8	修改了产品订货号	
V1.9	修改了封装图	
V2.0	修改了封装信息图	
V2.1	修改了 SSOP20 的封装信息图	
V2.2	新增了 SOIC20、SOIC14、SOIC8 的封装形式	
V2.3	去掉内部 VREOUT 和相关内容	

RoHS认证

本产品已通过 RoHS 检测。

声明及销售网络

销售及服务网点

上海 TEL:021-50275927

地址 上海浦东张江科苑路 201 号