

8 位微控制器

KF8F513X

数据手册

产品订购信息

芯片型号	订货号	温度	封装	FLASH	RAM (Byte)	DEE (Byte)	内部 HFOSC (Hz)	外部 HF/LFOSC (Hz)	8位定时器	16位定时器	捕捉/比较	16位 PWM	增强型 PWM	12位 ADC 通道	运放	LED	温度传感器	SPI/I2C	USART	MUL/DIV	工作电压	
KF8F5134	KF8F5134QS	-40~85°C	LQFP-48	16Kx16	2064	256	16M	20M/ 32.768k	1	4	3	2x4	1 (1路全桥 +2路半桥)	34	1	8X16	Y	1/1	2	Y	1.8~5.5V	
	KF8F5134QS-S	-40~125°C																				
	KF8F5134QR	-40~85°C		LQFP-44	16Kx16	2064	256	16M	20M/ 32.768k	1	4	3	2x4	1 (1路全桥 +2路半桥)	31	1	8X14	Y	1/1	2	Y	1.8~5.5V
	KF8F5134QR-S	-40~125°C																				
KF8F5133	KF8F5133SN	-40~85°C	SOIC-28	16Kx16	2064	256	16M	20M/ 32.768k	1	4	3	4+2	1 (1路全桥 +1路半桥)	24	1	8X8	Y	1/1	2	Y	1.8~5.5V	
	KF8F5133SN-S	-40~125°C																				
	KF8F5133TN	-40~85°C	TSSOP-28	16Kx16	2064	256	16M	20M/ 32.768k	1	4	3	4+2	1 (1路全桥 +1路半桥)	24	1	8X8	Y	1/1	2	Y	1.8~5.5V	
	KF8F5133TN-S	-40~125°C																				
KF8F5132	KF8F5132OG	-40~85°C	SSOP-20	16Kx16	2064	256	16M	20M/ 32.768k	1	4	3	3+1	1 (1路全桥 +1路半桥)	17	N	N	Y	1/1	2	Y	1.8~5.5V	
	KF8F5132OG-S	-40~125°C																				

版权所有@

上海芯旺微电子有限公司

本文档为上海芯旺微电子有限公司在现有数据资料基础上慎重且力求准确无误编制而成,确保应用符合技术规范,是您自身应负的责任。上海芯旺微电子有限公司不作任何明示或暗示、书面或口头、法定或其他形式的声明或担保,包括但不限于针对其使用情况、质量、性能、适销性或特定用途的适用性的声明或担保。上海芯旺微电子有限公司对因这些信息及使用这些信息而引起的后果不承担任何责任。如果将芯旺微电子有限公司的芯片用于生命维持和或生命安全应用,一切风险由使用方自负。使用方同意在此引发任何一切伤害、索赔、诉讼或费用时,会维护和保障上海芯旺微电子有限公司免于承担法律责任,并加以赔偿。

本文档中所述的器件应用信息及其他类似内容仅为您提供便利,可能有更新的信息所替代。上海芯旺微电子有限公司会不定期进行更新,恕不另行通知。使用方如需获得最新的产品信息,请及时访问上海芯旺微电子有限公司官网或与上海芯旺微电子有限公司联系。

KF8F513X 芯片使用注意事项

芯片的 ESD 防护措施

KF8F513X 芯片提供 8KV 的 ESD 标准保护电路。建议用户根据芯片存储/应用的环境采取适当静电防护措施。应注意应用环境的湿度；建议避免使用容易产生静电的绝缘体；存放和运输应在抗静电容器、抗静电屏蔽袋或导电材料容器中；包括工作台在内的所有测试和测量工具必须保证接地；操作者应该佩戴静电消除手腕环手套，不能用手直接接触芯片等。

芯片的 EFT 防护措施

KF8F513X 芯片提供 4.2KV 的 EFT 标准的保护电路。当 MCU 芯片应用在 PCB 系统时，需要遵守 PCB 相关设计要求，包括电源线、地线（包括数字/模拟电源分离，单点/多点接地等）、复位管脚保护电路、电源和地之间的去耦电容、高低频电路单独分别处理以及单/多层板选择等。

芯片的 LATCH-UP 防护措施

为有效防护 LATCH-UP 损坏芯片，用户需保证在 VDD 引脚上不出现异常高压或者负压。建议用户在 VDD 和 VSS 之间并接两个 105 和 102 大小的电容，电容尽量靠近芯片的 VDD 引脚。

芯片的焊接

KF8F513X 芯片的焊接应按照工业标准的焊接要求，以免损坏芯片。手工焊接时注意焊接的温度和焊接时间。

芯片的上电/断电

KF8F513X 芯片提供独立电源管脚。当 KF8F513X 芯片应用在多电源供电系统时，应先对 MCU 芯片上电，再对系统其他部件上电；反之，断电时，先对系统其他部件断电，再对 MCU 芯片断电。若操作顺序相反则可能导致芯片内部元件过压或过流，从而导致芯片故障或元件退化。

芯片的复位

KF8F513X 芯片提供内部上电复位。在系统设计时，若使用外部复位电路，建议采用三极管复位电路、RC 复位电路。若不使用外部复位电路，建议采用复位管脚接电阻到电源，或采取必要的电源抖动处理电路或其他保护电路。具体可参照芯片的数据手册说明。

芯片的内部时钟

KF8F513X 芯片提供内部时钟源。内部时钟源会随着温度、电压变化而偏移，可能会影响时钟源精度。具体可参照芯片的数据手册说明。

芯片的初始化

KF8F513X 芯片提供各种内部和外部复位。对于不同的应用系统，有必要对芯片寄存器、内存、功能模块等进行初始化，尤其是 I/O 管脚复用功能进行初始化，避免由于芯片上电以后，I/O 管脚状态的不确定情况发生。

芯片的管脚

KF8F513X 芯片提供宽范围的输入管脚电平，用户输入高电平应大于 VIH 的最小值，低电平应小于 VIL 的最大值，以免波动噪声进入芯片。对于未使用的输入/输出管脚，建议用户设为输入状态，并通过电阻上拉至电源或下拉至地，或设置为输出管脚，输出固定电平并浮空。对未使用的管脚处理因应用系统而异，具体遵循应用系统的相关规定和说明。

VDD 和 VSS 之间需接 104 以上的电容，电容尽量靠近 MCU 芯片的 VDD 引脚。

芯片的低功耗设计

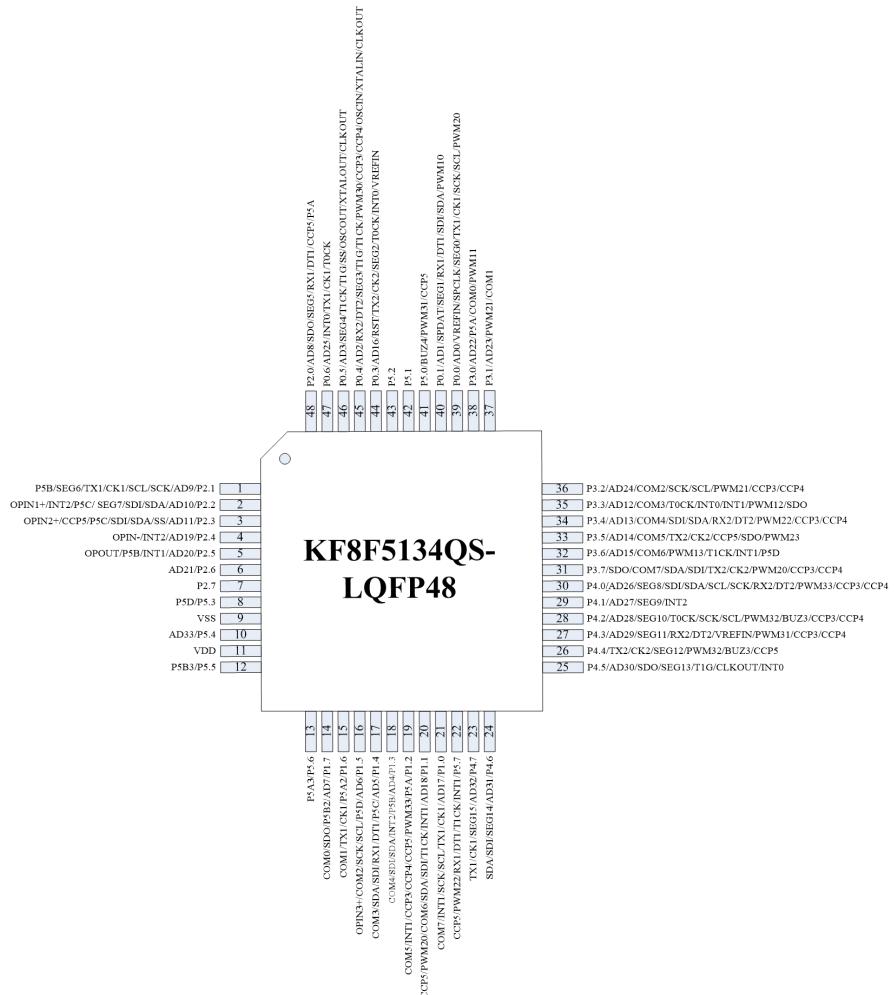
KF8F513X 芯片提供低功耗设计模式，用户在实际应用中可根据应用系统的要求采用各种不同的低功耗模式，包括系统工作时钟的选择和休眠模式的选择等等。

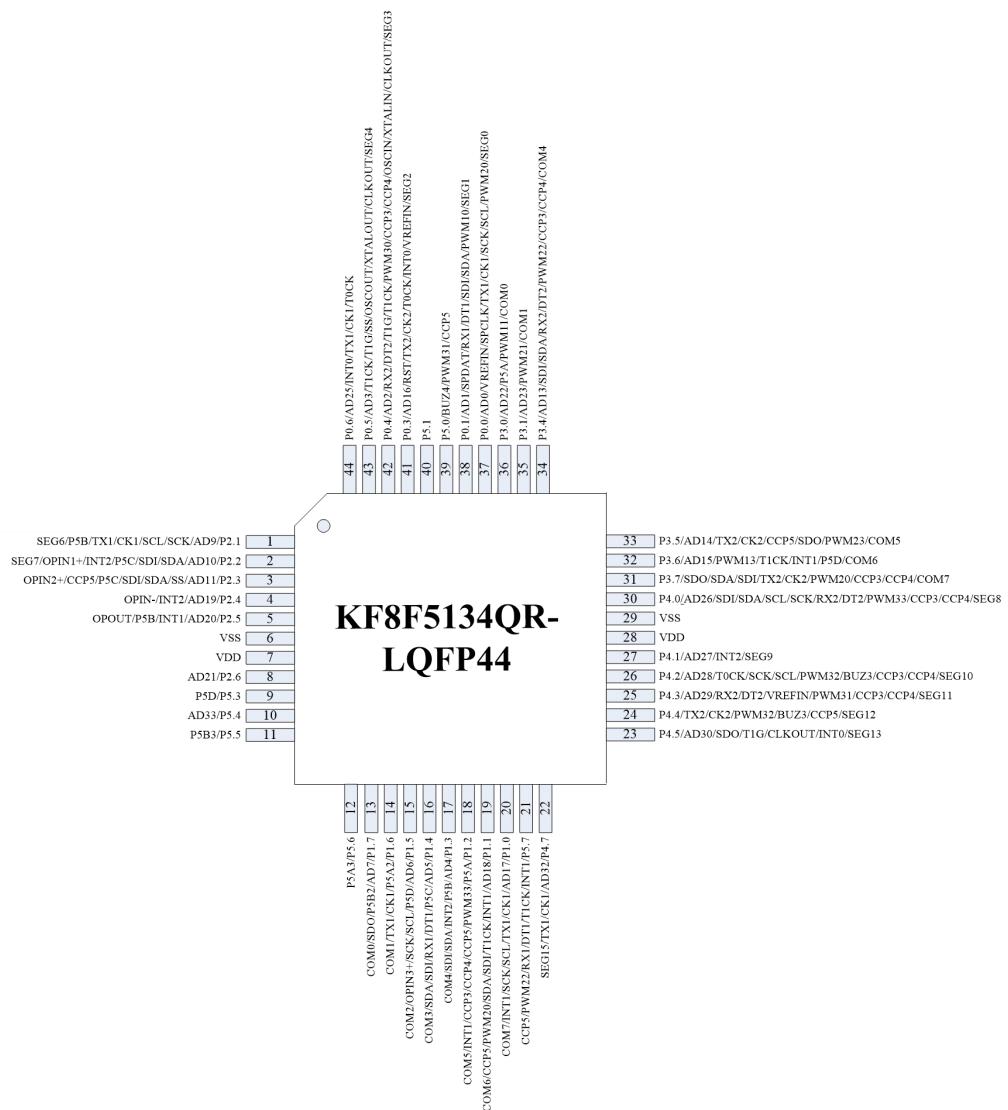
芯片的开发环境

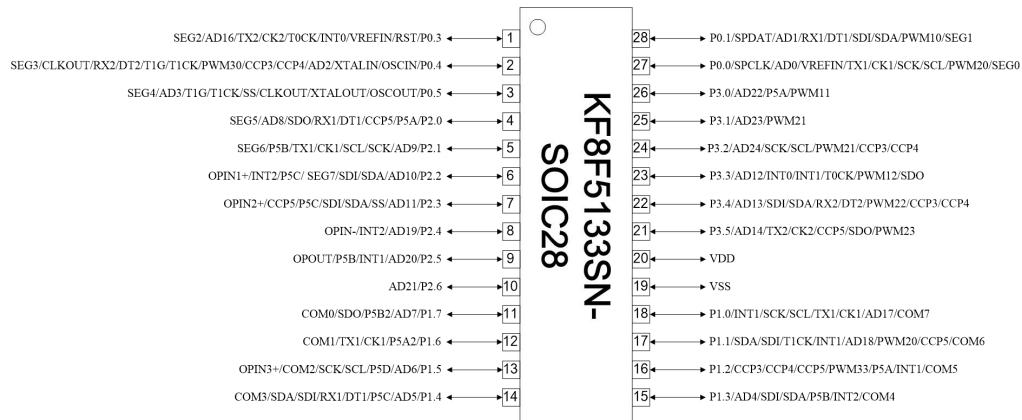
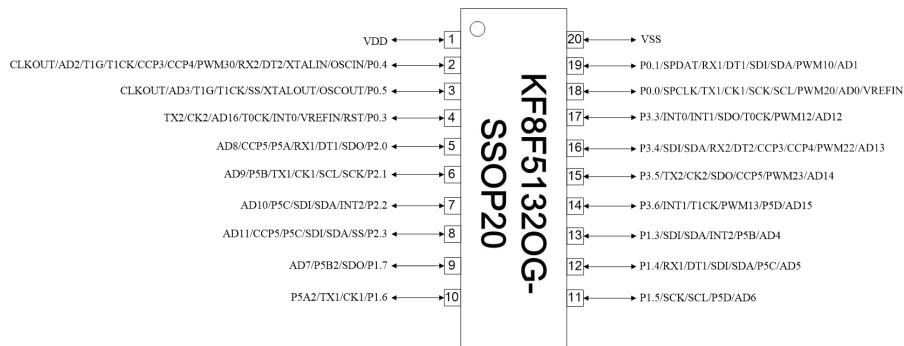
KF8F513X 芯片提供完整的软/硬件开发环境，并受知识产权保护。选择上海芯旺微电子有限公司指定的汇编器、编译器、编程器、硬件仿真器开发环境，必须遵循与芯片相关的规定和说明。

引脚示意图

LQFP-48 管脚图



LQFP-44 管脚图


SOIC-28/TSSOP-28 管脚图

SSOP-20 管脚图


引脚功能对应表

	ADC		PWM5			SSCI(I2C/SPI)			USART		TIMER	LED	OP	INT		EXTHFOSC /EXTLFOSC	PROGRAM	
	ADCH	ADVREF	PWM16	EPWM	CCP				U1	U2				外部中断	电平变化中断			
P0.0	AD0	VREFIN	PWM20			SCL/SCK			TX1/CK1			SEG0			INTP0		SPCLK	
P0.1	AD1		PWM10			SDA/SDI			RX1/DT1			SEG1			INTP0		SPDAT	
P0.3	AD16	VREFIN							TX2/CK2	T0CK	SEG2			INT0	INTP0		RST	
P0.4	AD2		PWM30		CCP3/4				RX2/DT2	T1CK/T1G	SEG3			INTP0	OSCIN	CLKOUT		
P0.5	AD3						SS			T1CK/T1G	SEG4			INTP0	OSCOUT	CLKOUT		
P0.6	AD25							TX1/CK1		T0CK			INT0	INTP0				
P1.0	AD17					SCL/SCK		TX1/CK1			COM7			INT				
P1.1	AD18		PWM20			SDA/SDI					T1CK	COM6		INT1				
P1.2			PWM33	P5A	CCP3/4/5						COM5			INT				
P1.3	AD4			P5B		SDA/SDI					COM4			INT2				
P1.4	AD5			P5C		SDA/SDI		RX1/DT1			COM3							
P1.5	AD6			P5D		SCL/SCK					COM2	OPIN3+						
P1.6				P5A2				TX1/CK1			COM1							
P1.7	AD7			P5B2			SDO				COM0							
P2.0	AD8			P5A	CCP5		SDO	RX1/DT1			SEG5							
P2.1	AD9			P5B		SCL/SCK		TX1/CK1			SEG6							
P2.2	AD10		PWM22	P5C		SDA/SDI					SEG7	OPIN1+	INT2					
P2.3	AD11			P5C	CCP5	SDA/SDI	SS					OPIN2+						
P2.4	AD19											OPIN-	INT2					
P2.5	AD20			P5B								OPOUT	INT1					
P2.6	AD21																	
P2.7																		
P3.0	AD22		PWM11	P5A							COM0							
P3.1	AD23		PWM21								COM1							
P3.2	AD24		PWM21		CCP3/4	SCL/SCK					COM2							
P3.3	AD12		PWM12				SDO			T0CK	COM3		INT0/INT1					
P3.4	AD13		PWM22		CCP3/4	SDA/SDI			RX2/DT2		COM4							
P3.5	AD14		PWM23		CCP5		SDO		TX2/CK2		COM5							
P3.6	AD15		PWM13	P5D						T1CK	COM6		INT1					
P3.7			PWM20		CCP3/4	SDA/SDI	SDO		TX2/CK2		COM7							
P4.0	AD26		PWM33		CCP3/4	SDA/SDI /SCL/SCK			RX2/DT2		SEG8			INTP4				
P4.1	AD27										SEG9		INT2	INTP4				
P4.2	AD28		PWM32		CCP3/4	SCL/SCK				T0CK/BUZ3	SEG10			INTP4				
P4.3	AD29	VREFIN	PWM31		CCP3/4				RX2/DT2		SEG11			INTP4				
P4.4			PWM32						TX2/CK2	BUZ3	SEG12			INTP4				
P4.5	AD30						SDO			T1G	SEG13		INT0	INTP4		CLKOUT		
P4.6	AD31					SDA/SDI					SEG14			INTP4				
P4.7	AD32							TX1/RX1			SEG15			INTP4				
P5.0			PWM31		CCP5					BUZ4								
P5.1																		
P5.2																		
P5.3				P5D														
P5.4	AD33																	
P5.5				P5B3														
P5.6				P5A3														
P5.7			PWM22					RX1/DT1		T1CK			INT1					

引脚功能说明表

引脚脚位				I/O	引脚功能	引脚说明
LQFP48	LQFP44	SOIC28	SSOP20			
1	1	5	6	I/O	P2.1	带上拉功能的双向输入输出端口
					AD9	AD 外部输入通道 9
					SCL/SCK	I2C 时钟脚/SPI 时钟脚
					TX1/CK1	串口 1 异步发送引脚/串口 1 同步时钟引脚
					SEG6	LED 显示 SEG 端 6
					P5B	PWM5 输出通道
2	2	6	7	I/O	P2.2	带上拉功能的双向输入输出端口
					AD10	AD 外部输入通道 10
					SDI/SDA	SPI 数据输入脚/I2C 数据（输入/输出）引脚
					SEG7	LED 显示 SEG 端 7
					P5C	PWM5 输出通道
					INT2	外部中断 2 输入引脚
					OPIN1+	运算放大器正端输入引脚 1
3	3	7	8	I/O	P2.3	带上拉功能的双向输入输出端口
					AD11	AD 外部输入通道 11
					SS	SPI 从动选择引脚
					SDI/SDA	SPI 数据输入脚/I2C 数据（输入/输出）引脚
					P5C	PWM5 输出通道
					CCP5	CCP5 捕捉输入信号/比较输出信号对应引脚
					OPIN2+	运算放大器正端输入引脚 2
4	4	8	-	I/O	P2.4	带上拉功能的双向输入输出端口
					AD19	AD 外部输入通道 19
					INT2	外部中断 2 输入引脚
					OPIN-	运算放大器负端输入引脚
5	5	9	-	I/O	P2.5	带上拉功能的双向输入输出端口
					AD20	AD 外部输入通道 20
					INT1	外部中断 1 输入引脚
					P5B	PWM5 输出通道
					OPOUT	运算放大器输出引脚
-	6	-	-	P	VSS	地
-	7	-	-	P	VDD	电源
6	8	10	-	I/O	P2.6	带上拉功能的双向输入输出端口

引脚脚位				I/O	引脚功能	引脚说明
LQFP48	LQFP44	SOIC28	SSOP20			
					AD21	AD 外部输入通道 21
7	-	-	-	I/O	P2.7	带上拉功能的双向输入输出端口
8	9	-	-	I/O	P5.3	带上拉功能的双向输入输出端口
					P5D	PWM5 输出通道
9	-	-	-	P	VSS	地
10	10	-	-	I/O	P5.4	带上拉功能的双向输入输出端口
					AD33	AD 外部输入通道 33
11	-	-	-	P	VDD	电源
12	11	-	-	I/O	P5.5	带上拉功能的双向输入输出端口
					P5B3	PWM5 输出通道
13	12	-	-	I/O	P5.6	带上拉功能的双向输入输出端口
					P5A3	PWM5 输出通道
14	13	11	9	I/O	P1.7	带上拉功能的双向输入输出端口
					AD7	AD 外部输入通道 7
					P5B2	PWM5 输出通道
					SDO	SPI 数据输出引脚
					COM0	LED 显示 COM 端 0
15	14	12	10	I/O	P1.6	带上拉功能的双向输入输出端口
					P5A2	PWM5 输出通道
					TX1/CK1	串口 1 异步发送引脚/串口 1 同步时钟引脚
					COM1	LED 显示 COM 端 1
16	15	13	11	I/O	P1.5	带上拉功能的双向输入输出端口
					AD6	AD 外部输入通道 6
					P5D	PWM5 输出通道
					SCK/SCL	I2C 时钟脚/SPI 时钟脚
					COM2	LED 显示 COM 端 2
					OPIN3+	运算放大器正端输入引脚 3
17	16	14	12	I/O	P1.4	带上拉功能的双向输入输出端口
					AD5	AD 外部输入通道 5
					P5C	PWM5 输出通道
					RX1/DT1	串口 1 异步接收引脚/串口 1 同步数据引脚
					SDI/SDA	SPI 数据输入脚/I2C 数据（输入/输出）引脚
					COM3	LED 显示 COM 端 3

引脚脚位				I/O	引脚功能	引脚说明
LQFP48	LQFP44	SOIC28	SSOP20			
18	17	15	13	I/O	P1.3	带上拉功能的双向输入输出端口
					AD4	AD 外部输入通道 4
					P5B	PWM5 输出通道
					INT2	外部中断 2 输入引脚
					SDI/SDA	SPI 数据输入脚/I2C 数据（输入/输出）引脚
					COM4	LED 显示 COM 端 4
19	18	16	-	I/O	P1.2	带上拉功能的双向输入输出端口
					P5A	PWM5 输出通道
					PWM33	PWM3 输出引脚 3
					CCP5	CCP5 捕捉输入信号/比较输出信号对应引脚
					CCP4	CCP4 捕捉输入信号/比较输出信号对应引脚
					CCP3	CCP3 捕捉输入信号/比较输出信号对应引脚
					INT1	外部中断 1 输入引脚
					COM5	LED 显示 COM 端 5
20	19	17	-	I/O	P1.1	带上拉功能的双向输入输出端口
					AD18	AD 外部输入通道 18
					INT1	外部中断 1 输入引脚
					T1CK	T1 外部时钟输入引脚
					SDI/SDA	SPI 数据输入脚/I2C 数据（输入/输出）引脚
					COM6	LED 显示 COM 端 6
					PWM20	PWM2 输出引脚 0
					CCP5	CCP5 捕捉输入信号/比较输出信号对应引脚
21	20	18	-	I/O	P1.0	带上拉功能的双向输入输出端口
					AD17	AD 外部输入通道 17
					TX1/CK1	串口 1 异步发送引脚/串口 1 同步时钟引脚
					SCK/SCL	I2C 时钟脚/SPI 时钟脚
					INT1	外部中断 1 输入引脚
					COM7	LED 显示 COM 端 7
22	21	-	-	I/O	P5.7	带上拉功能的双向输入输出端口
					INT1	外部中断 1 输入引脚
					T1CK	T1 外部时钟输入引脚
					RX1/DT1	串口 1 异步接收引脚/串口 1 同步数据引脚
					PWM22	PWM2 输出引脚 2

引脚脚位				I/O	引脚功能	引脚说明
LQFP48	LQFP44	SOIC28	SSOP20			
					CCP5	CCP5 捕捉输入信号/比较输出信号对应引脚
23	22	-	-	I/O	P4.7	带上拉和电平变化中断功能的双向输入输出端口
					AD32	AD 外部输入通道 32
					SEG15	LED 显示 SEG 端 15
					TX1/CK1	串口 1 异步发送引脚/串口 1 同步时钟引脚
24	-	-	-	I/O	P4.6	带上拉和电平变化中断功能的双向输入输出端口
					AD31	AD 外部输入通道 31
					SEG14	LED 显示 SEG 端 14
					SDI/SDA	SPI 数据输入脚/I2C 数据（输入/输出）引脚
25	23	-	-	I/O	P4.5	带上拉和电平变化中断功能的双向输入输出端口
					AD30	AD 外部输入通道 30
					SDO	SPI 数据输出引脚
					SEG13	LED 显示 SEG 端 13
					T1G	T1 外部时钟门控信号输入
					CLKOUT	指令时钟输出引脚
					INT0	外部中断输入 0
26	24	-	-	I/O	P4.4	带上拉和电平变化中断功能的双向输入输出端口
					TX2/CK2	串口 2 异步发送引脚/串口 2 同步时钟引脚
					SEG12	LED 显示 SEG 端 12
					PWM32	PWM3 输出引脚 2
					BUZ3	蜂鸣器 3 输出引脚
					CCP5	CCP5 捕捉输入信号/比较输出信号对应引脚
27	25	-	-	I/O	P4.3	带上拉和电平变化中断功能的双向输入输出端口
					AD29	AD 外部输入通道 29
					SEG11	LED 显示 SEG 端 11
					RX2/DT2	串口 2 异步接收引脚/串口 2 同步数据引脚
					VREFIN	ADC 外部参考电压输入引脚
					PWM31	PWM3 输出引脚 1
					CCP3	CCP3 捕捉输入信号/比较输出信号对应引脚
					CCP4	CCP4 捕捉输入信号/比较输出信号对应引脚
28	26	-	-	I/O	P4.2	带上拉和电平变化中断功能的双向输入输出端口
					AD28	AD 外部输入通道 28
					SEG10	LED 显示 SEG 端 10

引脚脚位				I/O	引脚功能	引脚说明
LQFP48	LQFP44	SOIC28	SSOP20			
				I/O	T0CK	T0 外部时钟输入引脚
					SCK/SCL	I2C 时钟脚/SPI 时钟脚
					PWM32	PWM3 输出引脚 2
					BUZ3	蜂鸣器 3 输出引脚
					CCP3	CCP3 捕捉输入信号/比较输出信号对应引脚
					CCP4	CCP4 捕捉输入信号/比较输出信号对应引脚
29	27	-	-	I/O	P4.1	带上拉和电平变化中断功能的双向输入输出端口
					AD27	AD 外部输入通道 27
					SEG9	LED 显示 SEG 端 9
					INT2	外部中断 2 输入引脚
-	28	20	-	P	VDD	电源
-	29	19	-	P	VSS	地
30	30	-	-	I/O	P4.0	带上拉和电平变化中断功能的双向输入输出端口
					AD26	AD 外部输入通道 26
					SEG8	LED 显示 SEG 端 8
					SDI/SDA	SPI 数据输入脚/I2C 数据（输入/输出）引脚
					SCK/SCL	I2C 时钟脚/SPI 时钟脚
					RX2/DT2	串口 2 异步接收引脚/串口 2 同步数据引脚
					PWM33	PWM3 输出引脚 3
					CCP3	CCP3 捕捉输入信号/比较输出信号对应引脚
					CCP4	CCP4 捕捉输入信号/比较输出信号对应引脚
31	31	-	-	I/O	P3.7	带上拉功能的双向输入输出端口
					SDO	SPI 数据输出引脚
					COM7	LED 显示 COM 端 7
					SDI/SDA	SPI 数据输入脚/I2C 数据（输入/输出）引脚
					TX2/CK2	串口 2 异步发送引脚/串口 2 同步时钟引脚
					PWM20	PWM2 输出引脚 0
					CCP3	CCP3 捕捉输入信号/比较输出信号对应引脚
					CCP4	CCP4 捕捉输入信号/比较输出信号对应引脚
32	32	-	14	I/O	P3.6	带上拉功能的双向输入输出端口
					AD15	AD 外部输入通道 15
					COM6	LED 显示 COM 端 6
					PWM13	PWM1 输出引脚 3

引脚脚位				I/O	引脚功能	引脚说明
LQFP48	LQFP44	SOIC28	SSOP20			
33	33	21	15	I/O	T1CK	T1 外部时钟输入引脚
					INT1	外部中断 1 输入引脚
					P5D	PWM5 输出通道
34	34	22	16	I/O	P3.5	带上拉功能的双向输入输出端口
					AD14	AD 外部输入通道 14
					COM5	LED 显示 COM 端 5
					TX2/CK2	串口 2 异步发送引脚/串口 2 同步时钟引脚
					CCP5	CCP5 捕捉输入信号/比较输出信号对应引脚
					SDO	SPI 数据输出引脚
					PWM23	PWM2 输出引脚 3
					P3.4	带上拉功能的双向输入输出端口
35	-	23	17	I/O	AD13	AD 外部输入通道 13
					COM4	LED 显示 COM 端 4
					SDI/SDA	SPI 数据输入脚/I2C 数据（输入/输出）引脚
					RX2/DT2	串口 2 异步接收引脚/串口 2 同步数据引脚
					PWM22	PWM2 输出引脚 2
					CCP3	CCP3 捕捉输入信号/比较输出信号对应引脚
					CCP4	CCP4 捕捉输入信号/比较输出信号对应引脚
					P3.3	带上拉功能的双向输入输出端口
36	-	24	-	I/O	AD12	AD 外部输入通道 12
					COM3	LED 显示 COM 端 3
					T0CK	T0 外部时钟输入引脚
					INT0	外部中断输入 0
					INT1	外部中断 1 输入引脚
					PWM12	PWM1 输出引脚 2
					SDO	SPI 数据输出引脚
					P3.2	带上拉功能的双向输入输出端口
					AD24	AD 外部输入通道 24
					COM2	LED 显示 COM 端 2
					SCK/SCL	I2C 时钟脚/SPI 时钟脚
					PWM21	PWM2 输出引脚 1
					CCP3	CCP3 捕捉输入信号/比较输出信号对应引脚
					CCP4	CCP4 捕捉输入信号/比较输出信号对应引脚

引脚脚位				I/O	引脚功能	引脚说明
LQFP48	LQFP44	SOIC28	SSOP20			
37	35	25	-	I/O	P3.1	带上拉功能的双向输入输出端口
					AD23	AD 外部输入通道 23
					PWM21	PWM2 输出引脚 1
					COM1	LED 显示 COM 端 1
38	36	26	-	I/O	P3.0	带上拉功能的双向输入输出端口
					AD22	AD 外部输入通道 22
					P5A	PWM5 输出通道
					COM0	LED 显示 COM 端 0
					PWM11	PWM1 输出引脚 1
39	37	27	18	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
					AD0	AD 外部输入通道 0
					VREFIN	ADC 外部参考电压输入引脚
					SPCLK	编程时钟引脚
					SEG0	LED 显示 SEG 端 0
					TX1/CK1	串口 1 异步发送引脚/串口 1 同步时钟引脚
					SCK/SCL	I2C 时钟脚/SPI 时钟脚
					PWM20	PWM2 输出引脚 0
40	38	28	19	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
					AD1	AD 外部输入通道 1
					SPDAT	编程数据引脚
					SEG1	LED 显示 SEG 端 1
					RX1/DT1	串口 1 异步接收引脚/串口 1 同步数据引脚
					SDI/SDA	SPI 数据输入脚/I2C 数据（输入/输出）引脚
					PWM10	PWM1 输出引脚 0
41	39	-	-	I/O	P5.0	带上拉功能的双向输入输出端口
					BUZ4	蜂鸣器 4 输出引脚
					PWM31	PWM3 输出引脚 1
					CCP5	CCP5 捕捉输入信号/比较输出信号对应引脚
42	40	-	-	I/O	P5.1	带上拉功能的双向输入输出端口
43	-	-	-	I/O	P5.2	带上拉功能的双向输入输出端口
44	41	-	4	I/O	P0.3	带上拉和电平变化中断功能的双向输入输出端口
					AD16	AD 外部输入通道 16
					RST	外部复位引脚

引脚脚位				I/O	引脚功能	引脚说明
LQFP48	LQFP44	SOIC28	SSOP20			
				I/O	TX2/CK2	串口 2 异步发送引脚/串口 2 同步时钟引脚
					SEG2	LED 显示 SEG 端 2
					T0CK	T0 外部时钟输入引脚
					INT0	外部中断输入 0
					VREFIN	ADC 外部参考电压输入引脚
45	42	-	2	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
					AD2	AD 外部输入通道 2
					RX2/DT2	串口 2 异步接收引脚/串口 2 同步数据引脚
					SEG3	LED 显示 SEG 端 3
					T1G	T1 外部时钟门控信号输入
					T1CK	T1 外部时钟输入引脚
					PWM30	PWM3 输出引脚 0
					CCP3	CCP3 捕捉输入信号/比较输出信号对应引脚
					CCP4	CCP4 捕捉输入信号/比较输出信号对应引脚
					OSCIN	外部高频晶振输入引脚
					XTALIN	外部低频晶振输入引脚
					CLKOUT	指令时钟输出引脚
46	43	-	3	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
					AD3	AD 外部输入通道 3
					SEG4	LED 显示 SEG 端 4
					T1CK	T1 外部时钟输入引脚
					T1G	T1 外部时钟门控信号输入
					SS	SPI 从动选择引脚
					OSCOUT	外部高频晶振输出引脚
					XTALOUT	外部低频晶振输出引脚
					CLKOUT	指令时钟输出引脚
47	44	-	-	I/O	P0.6	带上拉和电平变化中断功能的双向输入输出端口
					AD25	AD 外部输入通道 25
					INT0	外部中断输入 0
					TX1/CK1	串口 1 异步发送引脚/串口 1 同步时钟引脚
					T0CK	T0 外部时钟输入引脚
48	-	-	-	I/O	P2.0	带上拉功能的双向输入输出端口
					AD8	AD 外部输入通道 8

引脚脚位				I/O	引脚功能	引脚说明
LQFP48	LQFP44	SOIC28	SSOP20			
					SDO	SPI 数据输出引脚
					SEG5	LED 显示 SEG 端 5
					RX1/DT1	串口 1 异步接收引脚/串口 1 同步数据引脚
					CCP5	CCP5 捕捉输入信号/比较输出信号对应引脚
					P5A	PWM5 输出通道

目 录

产品订购信息.....	2
KF8F513X 芯片使用注意事项.....	3
引脚示意图.....	5
引脚功能对应表.....	8
引脚功能说明表.....	9
目 录.....	18
1 系统概述.....	29
1.1 芯片特征.....	30
1.2 系统框图.....	32
1.3 存储器.....	33
1.4 系统时钟.....	34
1.4.1 时钟模块相关寄存器.....	35
1.4.1.1 系统频率控制寄存器 OSCCTL.....	35
1.4.1.2 系统时钟标志寄存器 OSCSTA.....	36
1.4.2 上电延时.....	36
1.4.3 内部高频振荡器.....	36
1.4.3.1 内部高频振荡器校准寄存器.....	37
1.4.4 内部低频振荡器.....	37
1.4.4.1 内部低频振荡器校准相关寄存器.....	38
1.4.5 外部高频振荡器.....	38
1.4.6 外部低频振荡器.....	40
1.4.7 时钟切换和时钟信号同步.....	40
1.4.8 双速启动模式.....	41
1.4.9 双速启动过程.....	41
1.4.10 外部时钟故障检测.....	42
1.4.11 时钟故障保护处理.....	42
1.5 配置位.....	44
1.6 在线串行编程.....	46
2 I/O 端口.....	47
2.1 I/O 端口的读写.....	47
2.2 P0 口.....	48
2.2.1 P0 口相关的寄存器.....	48
2.2.1.1 P0 口状态寄存器 (P0)	48
2.2.1.2 P0 口输出锁存寄存器 (P0LR)	48
2.2.1.3 P0 口方向控制寄存器(TR0).....	49
2.2.1.4 P0 口上拉功能控制寄存器(PUR0).....	49
2.2.1.5 P0 口电平变化中断控制寄存器(IOCL0).....	49
2.2.1.6 P0 口模拟/数字口设置寄存器 (ANS0)	50

2.2.2 P0 口各引脚内部原理功能框图.....	51
2.3 P1 口.....	52
2.3.1 P1 口相关的寄存器.....	52
2.3.1.1 P1 口状态寄存器(P1).....	52
2.3.1.2 P1 口输出锁存寄存器 (P1LR)	52
2.3.1.3 P1 口方向控制寄存器(TR1).....	53
2.3.1.4 P1 口上拉功能控制寄存器(PUR1).....	53
2.3.1.5 P1 口模拟/数字口设置寄存器 (ANS1)	53
2.3.2 P1 口原理功能框图.....	54
2.4 P2 口.....	55
2.4.1 P2 口相关的寄存器.....	55
2.4.1.1 P2 口状态寄存器(P2).....	55
2.4.1.2 P2 口输出锁存寄存器 (P2LR)	55
2.4.1.3 P2 口方向控制寄存器(TR2).....	55
2.4.1.4 P2 口上拉功能控制寄存器(PUR2).....	56
2.4.1.5 P2 口模拟/数字口设置寄存器 (ANS2)	56
2.4.2 P2 口原理功能框图.....	57
2.5 P3 口.....	58
2.5.1 P3 口相关的寄存器.....	58
2.5.1.1 P3 口状态寄存器(P3).....	58
2.5.1.2 P3 口输出锁存寄存器 (P3LR)	58
2.5.1.3 P3 口方向控制寄存器(TR3).....	59
2.5.1.4 P3 口上拉功能控制寄存器(PUR3).....	59
2.5.1.5 P3 口模拟/数字口设置寄存器 (ANS3)	59
2.5.2 P3 口原理功能框图.....	61
2.6 P4 口.....	62
2.6.1 P4 口相关的寄存器.....	62
2.6.1.1 P4 口状态寄存器(P4).....	62
2.6.1.2 P4 口输出锁存寄存器 (P4LR)	62
2.6.1.3 P4 口方向控制寄存器(TR4).....	63
2.6.1.4 P4 口上拉功能控制寄存器(PUR4).....	63
2.6.1.5 P4 口电平变化中断控制寄存器(IOCL4).....	63
2.6.1.6 P4 口模拟/数字口设置寄存器 (ANS4)	64
2.6.2 P4 口原理功能框图.....	64
2.7 P5 口.....	66
2.7.1 P5 口相关的寄存器.....	66
2.7.1.1 P5 口状态寄存器(P5).....	66
2.7.1.2 P5 口输出锁存寄存器 (P5LR)	66
2.7.1.3 P5 口方向控制寄存器(TR5).....	67
2.7.1.4 P5 口上拉功能控制寄存器(PUR5).....	67
2.7.1.5 P5 口模拟/数字口设置寄存器 (ANS5)	67
2.7.2 P5 口原理功能框图.....	68
2.8 引脚复用功能.....	69
2.8.1 引脚复用功能相关寄存器.....	69

2.8.1.1 引脚复用功能控制寄存器 APFCTL0.....	70
2.8.1.2 引脚复用功能控制寄存器 APFCTL1.....	70
2.8.1.3 引脚复用功能控制寄存器 APFCTL2.....	71
2.8.1.4 引脚复用功能控制寄存器 APFCTL3.....	72
2.8.1.5 引脚复用功能控制寄存器 APFCTL4.....	72
2.8.1.6 引脚复用功能控制寄存器 APFCTL5.....	73
2.8.1.7 引脚复用功能控制寄存器 APFCTL6.....	73
2.8.1.8 引脚复用功能控制寄存器 APFCTL7.....	74
2.8.1.9 引脚复用功能控制寄存器 APFCTL8.....	74
2.8.1.10 引脚复用功能控制寄存器 APFCTL9.....	75
2.8.1.11 引脚复用功能控制寄存器 APFCTLA.....	75
2.8.1.12 引脚复用功能控制寄存器 APFCTLB.....	76
3 存储器.....	77
3.1 程序存储器(ROM)区.....	77
3.1.1 程序计数器(PC).....	77
3.1.1.1 PCL/PCH 寄存器的使用.....	78
3.1.1.2 执行 JMP、CALL 指令时的情况.....	78
3.1.2 堆栈.....	79
3.2 数据存储器(RAM)区.....	79
3.2.1 通用寄存器区.....	80
3.2.2 特殊功能寄存器(SFR)区.....	81
3.2.3 状态字寄存器 (PSW)	81
3.3 DATA EEPROM.....	82
3.3.1 寄存器 NVMDATAL.....	82
3.3.2 寄存器 NVMADDR.....	82
3.3.1 寄存器 NVMCTL0/NVMCTL1.....	82
3.3.2 写 DATA EEPROM.....	83
3.3.3 读 DATA EEPROM.....	84
3.4 寄存器组 RN.....	84
3.5 ID 地址单元.....	84
4 汇编指令及寻址方式.....	85
4.1 寻址方式.....	85
4.1.1 寄存器寻址.....	85
4.1.2 直接寻址.....	85
4.1.3 立即数寻址.....	85
4.1.4 寄存器间接寻址.....	86
4.1.5 位寻址.....	86
4.2 汇编指令.....	86
5 中断.....	87
5.1 中断相关的寄存器.....	90
5.1.1 中断控制寄存器 INTCTL.....	90

5.1.2 中断使能寄存器 EIE1.....	92
5.1.3 中断使能寄存器 EIE2.....	93
5.1.4 中断使能寄存器 EIE3.....	94
5.1.5 中断使能寄存器 EIE4.....	94
5.1.6 中断标志寄存器 EIF1.....	95
5.1.7 中断标志寄存器 EIF2.....	96
5.1.8 中断标志寄存器 EIF3.....	97
5.1.9 中断标志寄存器 EIF4.....	97
5.1.10 中断优先级控制寄存器 IP0.....	98
5.1.11 中断优先级控制寄存器 IP1.....	98
5.1.12 中断优先级控制寄存器 IP2.....	99
5.1.13 中断优先级控制寄存器 IP3.....	99
5.1.14 中断优先级控制寄存器 IP4.....	100
5.1.15 电源控制寄存器 PCTL.....	101
5.1.16 INT 中断沿选择寄存器 INTEDGCTL.....	101
5.1.17 中断响应.....	102
5.2 INT 中断.....	102
5.2.1 INT0 中断.....	102
5.2.2 INT1 中断.....	103
5.2.3 INT2 中断.....	103
5.3 定时器中断.....	103
5.4 P0/P4 口中断.....	103
5.5 CCP3/4/5 中断.....	105
5.6 PWM1/2/3/5 中断.....	105
5.7 USART 中断.....	105
5.8 中断现场保护.....	105
6 定时/计数器.....	106
6.1 定时/计数器 T0.....	106
6.1.1 T0 原理框图.....	106
6.1.2 T0 引脚复用.....	106
6.1.3 T0 相关的寄存器.....	107
6.1.3.1 OPTR 选择寄存器.....	107
6.1.4 定时模式.....	108
6.1.5 计数模式.....	108
6.1.6 T0 的使用.....	108
6.2 定时/计数器 T1.....	109
6.2.1 T1 原理框图.....	109
6.2.2 T1 引脚复用.....	109
6.2.3 T1 相关的寄存器.....	110
6.2.3.1 T1 控制寄存器.....	110
6.2.3.2 INT 中断沿选择寄存器.....	111
6.2.4 定时模式.....	111
6.2.5 计数模式.....	111

6.2.6 T1 重载功能.....	112
6.2.7 T1 在休眠模式下的运行.....	112
6.2.8 T1 分配给 PWM1.....	112
6.3 定时器 T2.....	113
6.3.1 T2 的工作原理.....	113
6.3.2 T2 相关的寄存器.....	114
6.3.2.1 T2 控制寄存器 T2CTL0.....	114
6.3.2.2 T2 控制寄存器 T2CTL1.....	115
6.3.2.3 T2CCR1/0H 和 T2CCR1/0L 寄存器.....	115
6.3.3 T2 中断.....	115
6.3.4 T2 在休眠模式.....	116
6.3.5 T2 分配给 PWM5.....	116
6.4 定时/计数器 T3.....	117
6.4.1 T3 原理框图.....	117
6.4.2 T3 相关寄存器.....	117
6.4.2.1 T3 控制寄存器.....	119
6.4.2.2 CCP3 功能寄存器.....	120
6.4.3 T3 预分频器.....	120
6.4.4 T3 计数时钟选择.....	120
6.4.5 T3 重载功能.....	121
6.4.6 T3 蜂鸣器输出功能.....	121
6.4.7 T3 中断.....	121
6.4.8 T3 工作在休眠模式.....	121
6.5 CCP3 (捕捉/比较/PWM 功能)	122
6.5.1 捕捉模式.....	122
6.5.2 比较模式.....	123
6.5.3 PWM3 模式.....	124
6.5.3.1 PWM3 输出端口.....	125
6.5.3.2 PWM3 相关的寄存器.....	125
6.5.3.3 PWM3 启动控制寄存器.....	125
6.5.3.4 PWM3 周期.....	125
6.5.3.5 PWM3 占空比.....	127
6.5.4 PWM3 分辨率.....	127
6.5.5 PWM3 中断.....	127
6.5.6 休眠模式下的操作.....	127
6.5.7 系统时钟频率的改变.....	128
6.5.8 复位的影响.....	128
6.5.9 PWM3 使用方法.....	128
6.6 定时/计数器 T4.....	129
6.6.1 T4 原理框图.....	129
6.6.2 T4 相关寄存器.....	129
6.6.2.1 T4 控制寄存器.....	130
6.6.2.2 CCP4 功能寄存器.....	131
6.6.3 T4 预分频器.....	131

6.6.4 T4 计数时钟选择.....	131
6.6.5 T4 重载功能.....	132
6.6.6 T4 蜂鸣器输出功能.....	132
6.6.7 捕捉比较功能 (CCP4)	132
6.6.7.1 捕捉模式.....	132
6.6.7.2 比较模式.....	133
6.6.8 T4 中断.....	135
6.6.9 T4 工作在休眠模式.....	135
7 ADC12 模数转换模块.....	136
7.1 与 ADC12 相关的寄存器.....	136
7.1.1 ADC12 控制寄存器 0(ADCCTL0).....	137
7.1.2 ADC12 控制寄存器 1(ADCCTL1).....	137
7.1.3 ADC12 控制寄存器 2(ADCCTL2).....	138
7.1.4 ADC12 中断控制寄存器 (ADCINTCTL)	139
7.1.5 温度传感器配置寄存器(TEMPSNR).....	139
7.2 通道的选择.....	139
7.3 ADC12 转换参考电压的选择.....	139
7.4 转换时钟的选择.....	140
7.5 输出格式.....	140
7.6 AD 转换的启动和完成.....	140
7.7 ADC12 工作在休眠模式.....	140
7.8 复位的影响.....	141
7.9 使用 ADC12 转换器的设置.....	141
8 PWM1/2 模块.....	142
8.1 PWM1/2 输出端口.....	142
8.2 工作原理.....	143
8.3 PWM1/2 相关的寄存器.....	144
8.3.1 PWM1/2 启动控制寄存器.....	144
8.4 PWM1/2 周期.....	144
8.5 PWM1/2 占空比.....	145
8.6 PWM1/2 分辨率.....	145
8.7 PWM1/2 中断.....	145
8.8 PWM2 配置为 16 位标准 PWM.....	146
8.9 休眠模式下的操作.....	146
8.10 复位的影响.....	146
8.11 PWM1/2 使用方法.....	146
9 CCP5(捕捉/比较/PWM5)模块.....	147
9.1 CCP5 引脚复用.....	147
9.2 CCP5 相关寄存器.....	147
9.3 捕捉模式.....	149
9.4 比较模式.....	150

9.5	PWM5 模式.....	151
9.5.1	PWM5 相关控制寄存器.....	153
9.5.1.1	PWM5CTL0 寄存器.....	153
9.5.1.2	PWM5CTL1 寄存器.....	154
9.5.1.3	PWM5CTL2 寄存器.....	155
9.5.1.4	PWM5 极性控制寄存器 (PWM5PC)	156
9.5.1.5	PMW5 强制控制寄存器 (PWM5FC)	156
9.5.1.6	PWM5 输出控制寄存器 (PWM5OC)	157
9.5.1.7	PWM5 自动关闭控制寄存器 (P5ASCTL0)	158
9.5.1.8	P5ASCTL1 寄存器.....	158
9.5.1.9	PSTRCTL0 寄存器.....	159
9.5.1.10	PSTRCTL1 寄存器.....	159
9.5.2	PWM5 的周期、占空比及分辨率.....	161
9.5.2.1	PWM5 周期.....	161
9.5.2.2	PWM5 占空比.....	161
9.5.2.3	PWM5 分辨率.....	162
9.5.2.4	PWM5 中断.....	163
9.5.3	边沿对齐 PWM 信号.....	163
9.5.4	中心对齐 PWM 信号.....	163
9.5.5	PWM 信号产生和中断.....	164
9.5.6	单输出模式.....	165
9.5.7	半桥输出模式.....	166
9.5.7.1	死区延时.....	167
9.5.8	全桥输出模式.....	168
9.5.9	脉冲转向控制.....	172
9.5.10	输出控制.....	172
9.5.11	强制输出模式.....	172
9.5.12	PWM 输出极性模式.....	172
9.5.13	PWM 更新锁定.....	173
9.5.14	PWM 复位控制.....	173
9.5.15	自动关断和自动重启模式.....	174
9.5.15.1	自动关断模式.....	174
9.5.15.2	自动重启模式.....	175
10	SSCI 模块.....	176
10.1	概述.....	176
10.2	SSCI 引脚复用.....	176
10.3	SSCI 相关寄存器.....	176
10.3.1	SSCI 控制寄存器 0 (SSCICCTL0)	177
10.3.2	SSCI 控制寄存器 1 (SSCICCTL1)	178
10.3.3	SSCI 状态寄存器 (SSCICISTA)	179
10.3.4	SSCI 屏蔽寄存器 (SSCICIMSK)	180
10.4	I2C 模式.....	182
10.4.1	工作原理.....	182

10.4.2 I2C 从动模式.....	183
10.4.2.1 寻址.....	183
10.4.2.2 接收.....	184
10.4.2.3 发送.....	185
10.4.2.4 广播呼叫地址支持.....	187
10.4.3 I2C 主动模式.....	188
10.4.3.1 主控模式支持.....	188
10.4.3.2 I2C 主模式操作.....	189
10.4.3.3 波特率发生器.....	190
10.4.3.4 I2C 主控模式启动条件时序.....	192
10.4.3.5 I2C 主控模式重复启动条件时序.....	192
10.4.3.6 I2C 主控模式发送.....	194
10.4.3.7 I2C 主控模式接收.....	196
10.4.3.8 应答序列时序.....	197
10.4.3.9 停止条件序列.....	197
10.4.3.10 时钟仲裁.....	199
10.4.4 多主控器模式.....	200
10.4.4.1 多主机通信, 总线冲突与总线仲裁.....	200
10.4.4.2 启动条件期间的总线冲突.....	201
10.4.4.3 重复启动条件期间的总线冲突.....	203
10.4.4.4 停止条件期间的总线冲突.....	205
10.4.4.5 SSCI 屏蔽寄存器.....	206
10.5 SPI 模式.....	207
10.5.1 工作原理.....	208
10.5.2 使能 SPI/IO 与外部链接.....	209
10.5.3 典型连接.....	209
10.5.4 主模式.....	210
10.5.5 从模式.....	211
10.5.5.1 从动模式.....	211
10.5.5.2 从动选择同步.....	211
10.5.6 休眠模式和复位.....	214
10.5.7 SPI 四种工作模式设置.....	214
10.5.7.1 主控发送工作流程.....	214
10.5.7.2 从动接收工作流程.....	214
10.5.7.3 主控接收工作流程.....	215
10.5.7.4 从动发送工作流程.....	215
11 通用全双工/ 半双工收发器 (USART)	216
11.1 系统概述.....	216
11.1.1 USART1/2 引脚复用.....	216
11.1.2 相关寄存器.....	217
11.1.3 原理框图.....	218
11.2 波特率发生器.....	219
11.2.1 USARTx 波特率控制寄存器 BRCTLx.....	220

11.2.2 波特率的选择.....	221
11.2.3 自动波特率检测.....	222
11.2.4 接收间隔字符时自动唤醒.....	224
11.2.5 间隔符时序.....	225
11.2.5.1 间隔和同步发送序列.....	225
11.2.5.2 接收间隔字符.....	226
11.3 USART 全双工模式.....	227
11.3.1 USART 全双工发送操作.....	228
11.3.2 发送和控制状态寄存器 TSCTLx.....	228
11.3.3 发送数据.....	229
11.3.4 全双工发送的设置.....	230
11.3.5 USART 全双工接收操作.....	231
11.3.6 接收状态和控制寄存器 RSCTLx.....	232
11.3.7 接收数据.....	233
11.3.8 接收错误.....	233
11.3.9 地址检测.....	234
11.3.10 全双工接收的设置.....	234
11.3.11 RS-485 发送/接收.....	235
11.3.11.1 RS-485 9 位地址检测模式设置.....	236
11.3.12 全双工操作时钟的精确性.....	236
11.4 USART 半双工模式.....	237
11.4.1 USARTx 半双工主控模式.....	237
11.4.2 半双工主控发送.....	237
11.4.3 半双工主控接收.....	239
11.4.4 USART 半双工从动模式.....	242
11.4.5 USART 半双工从动发送.....	242
11.4.6 USART 半双工从动接收.....	243
11.4.7 USART 半双工 RS-485 模式.....	243
11.5 USART 工作在休眠模式下.....	243
11.5.1 USART 休眠使能寄存器 USLPEN.....	243
11.6 USART 单线通信模式.....	244
11.6.1 UPINSET1 和 UPINSET2 寄存器.....	244
11.7 7816 模式.....	245
11.7.1 7816 模式发送.....	246
11.7.1.1 发送设置.....	246
11.7.2 7816 模式接收.....	247
11.7.3 7816 模式寄存器.....	248
11.7.3.1 7816 发送控制寄存器 U7816TXCTL2.....	248
11.7.3.2 7816 接收控制寄存器 U7816RXCTL2.....	249
11.7.3.3 7816 控制寄存器 U7816CTL2.....	249
11.7.3.4 预分频控制寄存器 CLKDIV2.....	250
11.7.3.5 EGT 控制寄存器 EGTCTL2.....	250
12 运算放大器模块.....	251

12.1	与运放有关的寄存器.....	251
12.1.1	运放控制寄存器.....	251
12.1.2	运放校准寄存器.....	252
12.2	运放使用方法.....	252
12.2.1	校准方法.....	252
12.2.2	使用方法.....	252
13	LED 驱动模块.....	254
13.1	概述.....	254
13.2	工作原理.....	255
13.2.1	扫描时钟产生.....	255
13.2.2	扫描信号波形.....	255
13.3	LED 相关寄存器.....	256
13.3.1	LED 显示驱动控制寄存器(LEDCTL).....	256
13.3.2	LED 显示驱动控制寄存器 2(LEDCTL2).....	257
13.3.3	LED 显示驱动分频寄存器(LEDPRE).....	257
13.3.4	LED 显示驱动数据寄存器(LEDDATAx).....	258
13.3.5	LED 显示驱动辉度调节控制寄存器 (LEDLUM)	259
13.4	LED 通道.....	260
13.5	辉度调节.....	261
13.6	输出模式.....	261
13.7	LED 显示驱动使用方法.....	261
14	硬件乘法器模块.....	262
14.1	概述.....	262
14.2	硬件乘法器相关寄存器.....	263
14.2.1	乘法运算控制寄存器 (MULCTL)	263
14.3	硬件乘法器的使用.....	263
15	硬件除法器模块.....	265
15.1	概述.....	265
15.2	硬件除法器相关寄存器.....	266
15.2.1	除法控制寄存器 (DIVCTL)	266
15.2.2	数据寄存器.....	266
15.2.2.1	被除数寄存器{DIVAH:DIVAL}.....	267
15.2.2.2	除数寄存器 DIVB.....	267
15.2.2.3	结果商寄存器{DIVQH:DIVQL}.....	267
15.2.2.4	运算余数寄存器 DIVR.....	268
15.3	硬件除法器操作.....	268
16	复位.....	269
16.1	电源控制状态寄存器(PCTL).....	270
16.2	上电复位(POR).....	271
16.3	WDT 复位.....	271
16.4	RST 复位.....	271

16.5 欠压检测复位(LVR).....	272
16.6 上电延时定时器.....	272
16.7 不同复位条件下对寄存器的影响.....	273
17 休眠模式.....	278
18 看门狗定时器.....	279
18.1 看门狗相关寄存器.....	279
18.1.1 看门狗预分频选择寄存器 WDTPS.....	279
18.2 看门狗的开启方式.....	280
18.3 看门狗的清狗方式.....	280
18.4 看门狗的周期.....	280
19 电气规范.....	282
19.1 极限参数值.....	282
19.2 INTHFOSC 的频率精度与 VDD 和温度之间的关系.....	283
19.3 静态电流特性.....	283
19.4 休眠电流特性.....	284
19.5 外设电流特性.....	284
19.6 I/O 端口电平和芯片供电电压特性.....	285
19.7 POR.....	286
19.8 LVR.....	286
19.9 内部高频振荡器.....	287
19.10 内部低频振荡器.....	287
19.11 外部低频振荡器.....	287
19.12 外部高频振荡器.....	287
19.13 ADC12.....	288
19.14 运算放大器.....	288
19.15 直流特性图表.....	289
20 封装信息.....	297
附录 1 特殊功能寄存器 (SFR) 功能汇总.....	300
附录 2 汇编指令集.....	304
附录 3 寄存器全称表.....	306
附录 4 全双工异步模式的典型波特率和误差值.....	312
产品标识体系.....	316
版本信息.....	317
ROSH 认证.....	318
声明及销售网络.....	319

1 系统概述

KF8F513X 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，大多数指令能在在一个机器周期内执行完成。一共有 73 条指令，效率高，容易进行指令扩展。

KF8F513X 芯片内集成了多种外设，包括：

- 1 个 8 位定时器/计数器 T0
- 1 个 16 位定时器/计数器 T1
- 1 个 16 位的定时器 T2
- 1 个 16 位定时器/计数器 T3 附带 CCP 功能
- 1 个 16 位定时器/计数器 T4 附带 CCP 功能
- 1 个 12 位 34 路外部通道 ADC 模块
- 1 个 4 路 8 位 PWM (PWM1)
- 2 个 4 路 16 位 PWM (PWM2 和 PWM3)
- 1 个 CCP(捕捉/比较/PWM5)模块
- 1 个 I2C/SPI 模块
- 2 个 USART 模块(USART2 兼容 7816)
- 1 个 8×16 的 LED 驱动模块

温度传感器硬件看门狗（带软件使能）和低电压检测及低电压复位模块等。

芯片内集成了 $(2048+16) \times 8$ 位的数据存储器 RAM、 $16K \times 16$ 位的程序存储器和 256×8 位的 DATA EEPROM。

1.1 芯片特征

● CPU

- 高性能哈佛结构的 RISC CPU
- 73 条精简指令
- 支持中断优先级处理，共 25 个中断源
- 复位向量位于 0000H
- 两级中断可选，用不同的入口地址（高 0004H，低 0014H）
- 支持 16 级硬件堆栈
- 工作频率默认内部高频时为 62.5KHz~16MHz，软件可选时钟源及分频

● 存储器

- 16K×16 位 FLASH 程序存储器
- (2048+16)×8 位的数据存储器
- 256×8 位的 DATA EEPROM
- 工作寄存器组 R0~R7
- FLASH 可经受 100 000 次写操作
- DATA 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
- 内部高频时钟精度 16MHz±1%
- 内部可校正低频 32KHz 时钟
- 提供一个 1.5V/2V/3V/4V 可选的参考电压，精度为±5%
- 支持在线串行编程，低功耗休眠模式

● I/O 口配置

- 输入输出口：均为双向输入输出口
- 内置上拉功能：P0~P5 口均带有弱上拉功能
- 电平变化中断：P0 和 P4 口均有电平变化中断功能
- IO 口数字输入类型：P0~P5 口均为 SMIT 型

● 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：带重载功能、门控和预分频器的 16 位定时器/计数器
- 定时器 2：带 16 位周期寄存器、预分频器和后分频器 16 位定时器
- 定时器 3：带有重载功能、CCP、预分频器及时钟源多选的 16 位定时器/计数器
- 定时器 4：带有重载功能、CCP、预分频器及时钟源多选的 16 位定时器/计数器

● 其它外设

- 1 个 12 位 34 路外部通道 ADC 模块
- 1 个 4 路 8 位脉宽调制 PWM 模块(PWM1)
- 2 个 4 路 16 位脉宽调制 PWM 模块（PWM2 和 PWM3）

1 路 CCP(捕捉/比较/增强型 PWM5)模块

1 个 I2C/SPI 模块

2 个 USART 模块

1 个 8×16 的 LED 驱动模块

1 个温度传感器

● 工作条件

工作电压: 1.8V~5.5V

工作温度范围: -40°C~85°C

-40°C~125°C(*-S)

1.2 系统框图

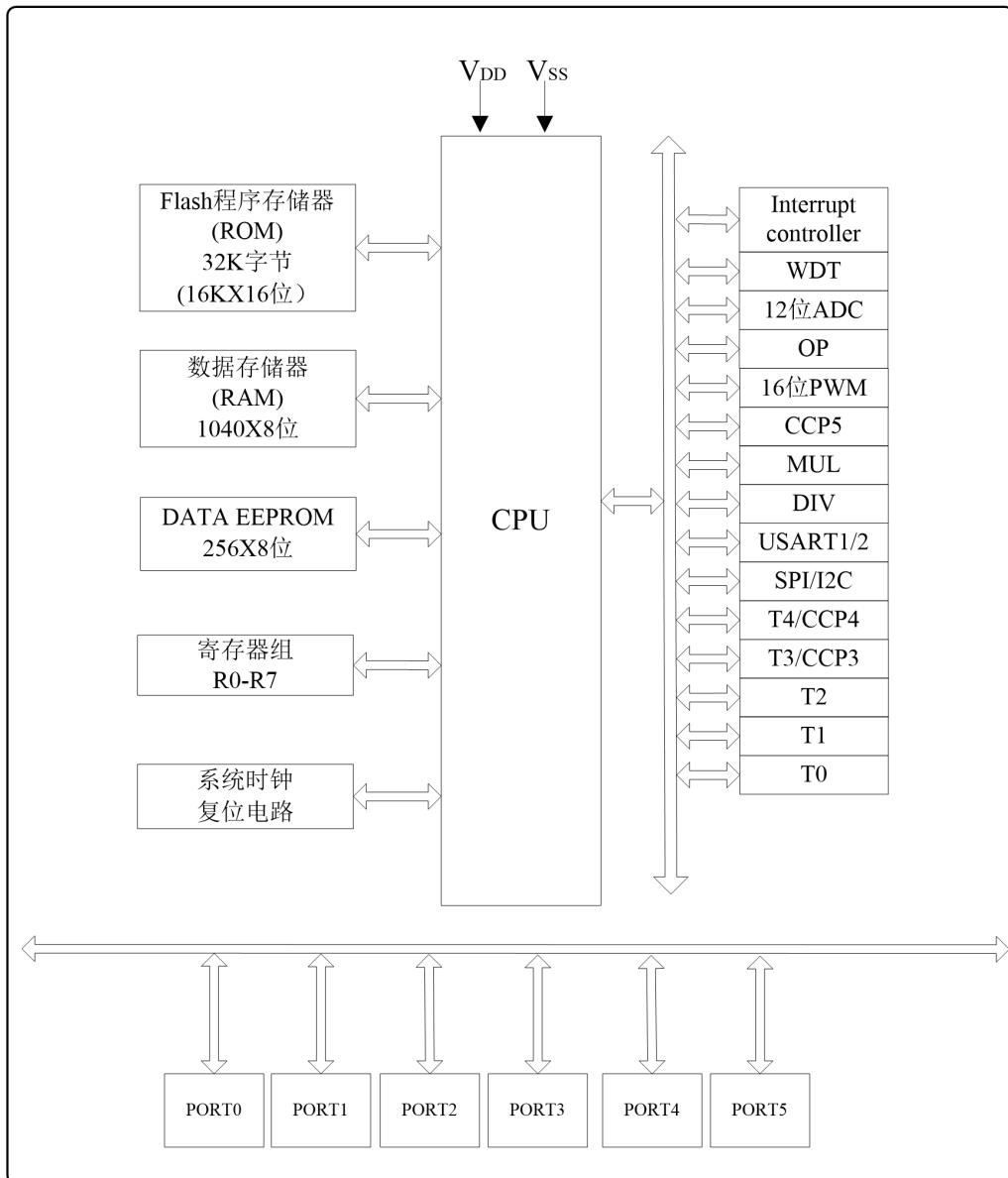


图 1.1 系统框图

1.3 存储器

KF8F513X 单片机的存储器包含:程序存储器(ROM)、数据存储器(RAM)和 DATA EEPROM。

KF8F513X 的程序存储器空间为 32K 字节 (16K×16 位), 寻址范围为 0000H~27FFH, 可擦写次数为 10 万次。数据存储器分为特殊寄存器区(SFR)和通用存储器区, 其中通用存储器区包括通用存储器区 0 至通用存储器区 15。通用存储器区 0 至通用存储器区 15 有 128×16 位的存储单元, 各区的地址请查阅第 3 章。

DATA EEPROM 的地址为独立寻址, 地址为 00H~FFH。有关以上各种存储器的具体介绍请参考第 3 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,系统时钟是由振荡器时钟分频而来。本芯片中一个机器周期等于四个系统周期,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。

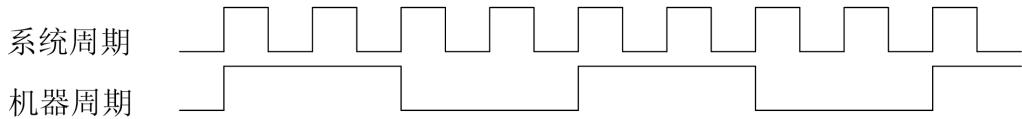


图 1.2 机器周期

KF8F513X 系列单片机提供 4 个可选时钟源:

内部高频振荡器 INTHF:以内部高频振荡器为时钟源;

内部低频振荡器 INTLF:以内部低频振荡器为时钟源;

外部高频振荡器 EXTHF:标准晶振、陶瓷谐振器或外接 20MHz~125KHz 的时钟源工作;

外部低频振荡器 EXTLF:低电流工作并使用外接 32.768KHz 的钟表晶振。

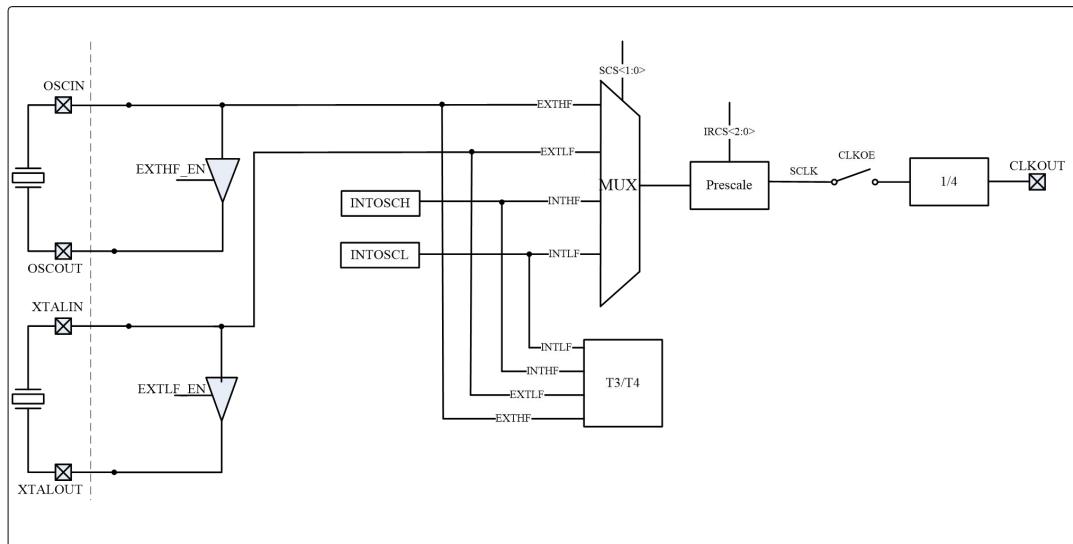


图 1.3 时钟模块原理框图

KF8F513X 系列单片机的系统时钟源可以配置为 EXTHF、EXTLF、INTHF 或者 INTLF。同时 4 个可选时钟源还可以作为部分外设的工作时钟源。外部高频时钟和外部低频时钟共用外部时钟引脚,不能同时使用两个外部时钟源。

表 1-1 与时钟有关的名词表述

名称	定义即描述
SCLK	定义为系统时钟
SCLK/4	定义为机器时钟
T _{sys} 或 T _{sys}	定义为系统时钟周期
T _{mc} 或 T _{mc}	定义为机器周期

INTHF	定义为内部高频振荡器（或时钟源）
INTLF	定义为内部低频振荡器（或时钟源）
EXTHF	定义为外部高频振荡器（或时钟源）
EXTLF	定义为外部低频振荡器（或时钟源）
OSC	定义为 INTHF,INTLF,EXTHF 和 EXTLF 的集合

1.4.1 时钟模块相关寄存器

表 1-2 时钟模块相关寄存器表

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
2FH	OSCCTL	CLKOE	IRCS2	IRCS1	IRCS0	SCS1	SCS0	IESO	FSCM
28H	OSCSTA	-	OSTS	HTS	LTS	-	-	SCF1	SCF0

1.4.1.1 系统频率控制寄存器 OSCCTL

寄存器1.2: OSCCTL系统频率控制寄存器(地址:2FH)

复位值	bit7	CLKOE	IRCS2	IRCS1	IRCS0	SCS1	SCS0	IESO	bit0
0010 0000		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

注: R=可读 W=可写 -=未用 x=不定 后面与此相同不再复述

CLKOE: 系统时钟输出使能位

1=使能系统时钟四分频输出
0=禁止系统时钟四分频输出

IRCS<2:0>: 时钟频率选择位

111=1:1
110=1:2
101=1:4
100=1:8
011=1:16
010=1:32 (默认)
001=1:64
000=1:256

SCS<1:0>: 系统时钟源选择位

00=选择内部高频时钟
01=选择内部低频时钟
10=选择外部低频时钟
11=选择外部高频时钟

IESO: 双速模式使能位

0=禁止双速功能
1=启动双速功能

FSCM: 外部时钟故障检测使能位

0=禁止故障检测功能
1=使能故障检测功能

1.4.1.2 系统时钟标志寄存器 OSCSTA

寄存器1.2: OSCSTA系统时钟标志寄存器(地址:28H)

复位值	bit7	OSTS	HTS	LTS	bit0	SCF1	SCF0
-110--00	-	U	R	R	U	U	R

注: R=可读 W=可写 -=未用 x=不定 后面与此相同不再复述

OSTS: 内外时钟标志位

0=系统时钟为外部时钟

1=系统时钟为内部时钟

HTS: 内部高频时钟稳定位

0=内部高频时钟未稳定

1=内部高频时钟稳定

LTS: 内部低频时钟稳定位

0=内部低频时钟未稳定

1=内部低频时钟稳定

SCF<1:0>: 系统时钟源标志位

00=当前系统时钟源为内部高频时钟

01=当前系统时钟源为内部低频时钟

10=当前系统时钟源为外部低频时钟

11=当前系统时钟源为外部高频时钟

1.4.2 上电延时

KF8F513X 系列单片机的上电延时可以通过配置位 \overline{PWRT} 设置, 上电延时计数时钟 $PWRTCLK$ 为内部低频时钟。

当 $\overline{PWRT} = 1$ 时, 上电延时关闭; 当 $\overline{PWRT} = 0$ 时, 上电延时打开, 延时时间如下:

式 1.1 上电延时时间计算公式

$$T_{pwrt} = \frac{2^{10}}{PWRTCLK}$$

1.4.3 内部高频振荡器

KF8F513X 系列单片机的内部高频时钟由系统内部高频振荡器提供, 时钟频率为 16MHz, 精度为 $\pm 1\%$ 。

1.4.3.1 内部高频振荡器校准寄存器

表 1-3 INTHFOSC 校准相关寄存器表

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
30H	OSCCAL0								内部高频晶振校准寄存器 0
37H	OSCCAL1								内部高频晶振校准寄存器 1
4CH	OSCCAL2								内部高频晶振校准寄存器 2

OSCCAL0、OSCCAL1 和 OSCCAL2 为内部高频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 3FFFH、3FFEH 和 3FFDH 的晶振校准值读出来分别存放到内部高频振荡器的校准寄存器中，否则会导致系统时钟频率不准。程序示例如下：

例 1-1 内部高频振荡器校准样例

```
读内部高频振荡器校准值
MOVP #0X3F ;切换到PAGE2区
CALL 0XFFF
MOV OSCCAL0, R0
CALL 0XFFE
MOV OSCCAL1, R0
CALL 0XFFD
MOV OSCCAL2, R0
MOVP #0X00 ;切换回PAGE0区
```

1.4.4 内部低频振荡器

KF8F513X 模块内置内部低频振荡器为系统内部低频振荡器，其振荡器频率为 32KHz，它不仅可以为 SCLK 时钟提供时钟源，而且还可以作为独立时钟信号供外设模块使用。也可以作为内部上电延时定时器和看门狗 WDT 定时器的时钟。

LPRCCAL 为内部低频振荡器的时钟校准寄存器，用来存放校准值。用户在编程时，需要在程序初始化部分，将存放在程序空间 3FF8H 的晶振校准值存放到 LPRCCAL 中。程序示例如下。

例 1-2 内部低频振荡器校准样例

```
读内部低频晶振校准值
MOVP #0X3F ;切换到PAGE2区
CALL 0XFF8
MOV LPRCCAL, R0
MOVP #0X00 ;切换回PAGE0区
```

1.4.4.1 内部低频振荡器校准相关寄存器

表 1-4 INTLFOSC 校准相关寄存器表

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	
151H	LPRCCTL	-	-	-	-	-	LPRC2	LPRC1	LPRC0	
152H	LPRCCAL	内部低频晶振校准寄存器								

寄存器1.1: LPRCCTL: 内部低频RC控制寄存器(地址:151H)

复位值	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0000 0000	-	-	-	-	-	LPRC2	LPRC1	LPRC0

注: R=可读 W=可写 -=未用 x=不定 后面与此相同不再复述

LPRC<2:0>: 低功耗内部低频振荡器电流偏置选择位

111=max

...

100=默认值

...

001=min

000=关闭

1.4.5 外部高频振荡器

如图 1.4 所示, 引脚 OSCIN 和引脚 OSCOUT 可以接外部标准晶体、陶瓷谐振器或外接 20MHz~125KHz 的时钟为时钟源。它不仅可以为 SCLK 时钟提供时钟源, 而且还可以作为独立时钟信号供外设模块使用。

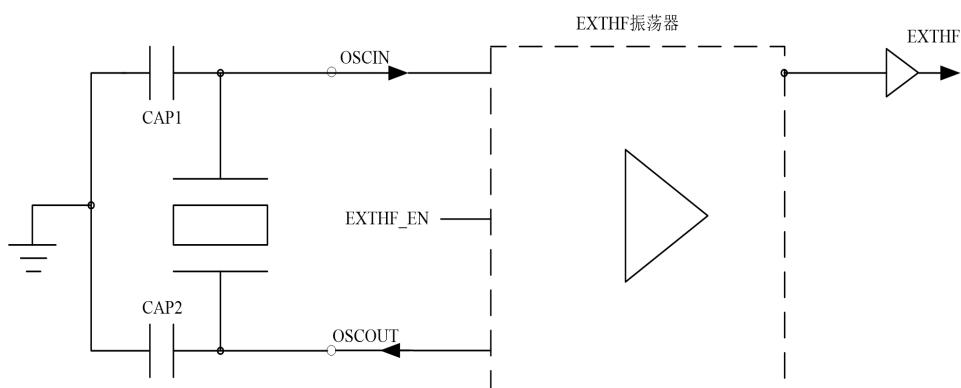


图 1.4 外部高频振荡器原理图

表 1-5 外部高频振荡器相关名称说明表

名称	说明
OSCIN	外部高频晶振输入引脚

OSCOUT	外部高频晶振输出引脚
EXTHF_EN	外部高频晶振软件&硬件使能信号
EXTHF	外部高频时钟信号
CAP1&CAP2	外部高频晶振输入输出引脚外接到地电容, 推荐 10pF

1.4.6 外部低频振荡器

如图 1.5 所示, 引脚 XTALIN 和引脚 XTALOUT 接外部 32.768KHz 石英晶振。EXTLF 可以作为主系统时钟 SCLK 的时钟源, 还可以作为独立时钟信号供外设模块使用。

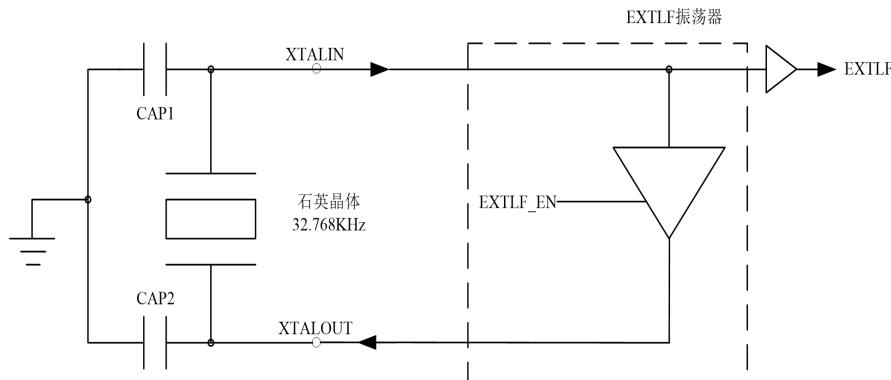


图 1.5 外部低频振荡器原理图

表 1-6 外部低频振荡器相关名称说明表

名称	说明
XTALIN	外部低频晶振输入引脚
XTALOUT	外部低频晶振输出引脚
EXTLF_EN	外部低频晶振软件&硬件使能信号
EXTLF	外部低频时钟信号
CAP1&CAP2	外部低频晶振输入输出引脚外接到地电容, 推荐 10pF

1.4.7 时钟切换和时钟信号同步

通过设置 OSCCTL 寄存器的 SCS<1:0>位可以选择不同的时钟源作为系统时钟。SCS 位复位为 00, 即选择内部高频时钟源作为单片机的系统时钟。

当系统时钟由外部时钟源切换至内部时钟源时, 系统时钟将在 SCS 配置后立即对时钟进行切换。

当配置 SCS 位将系统时钟切换至外部时钟源时 (包括 EXTHF 和 EXTLF), 振荡器起振定时器 OST 将启动, 并以 SCS 位配置的外部时钟为计数时钟开始计数, OST 计数的时间内, 系统仍以原来的时钟源作为系统时钟, 直到 OST 计数器达到 1024 次计数, 系统时钟源切换至 SCS 位配置的外部时钟。



图 1.6 时钟切换流程图

当系统时钟在一个时钟源切换到另一个时钟源时, 切换必须同步以避免发生时间竞争。

当选择一个新的时钟源，会发生以下过程。

- (1) 写 SCS 位改变时钟源；如切换至外部时钟源则须经过 OST 计数器 1024 计数；
- (2) 时钟切换电路等待当前时钟的下降沿；
- (3) 时钟 CLK 保持为低电平，时钟切换电路等待新的时钟的上升沿；
- (4) CLK 与新的时钟连接，完成时钟切换。

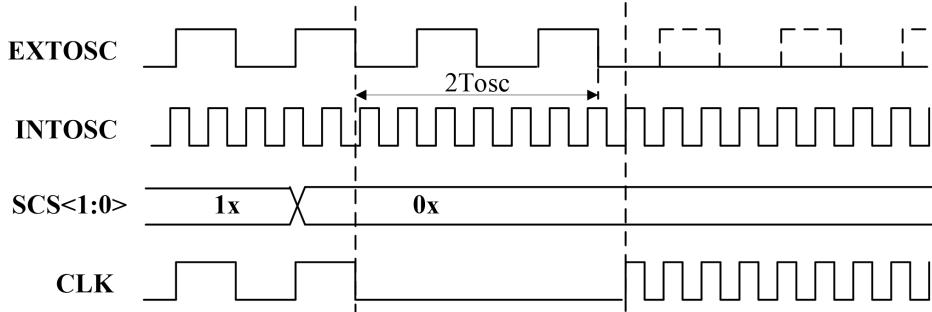


图 1.7 切换至内部时钟源时时钟信号同步时序图

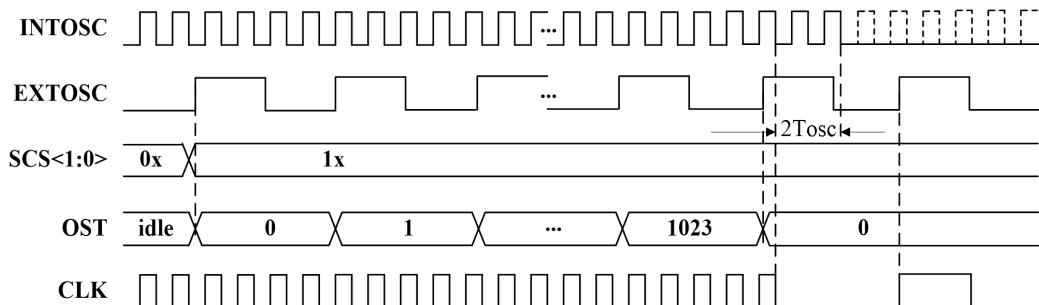


图 1.8 切换至外部时钟源时时钟信号同步时序图

1.4.8 双速启动模式

当系统时钟选择外部时钟进入休眠后，唤醒时，外部时钟需要经过 OST 计数器完成 1024 次计数后才会恢复系统时钟。

双速启动模式通过寄存器 OSCCTL 的 IESO 位设置，当使能双速模式时，MCU 会在唤醒后外部时钟进行 OST 计数期间，通过内部高频时钟作为系统时钟运行，当外部时钟完成 OST 计数后，MCU 会自动将系统时钟从内部高频时钟切换至外部时钟。

1.4.9 双速启动过程

1. 从休眠状态唤醒；
2. 内部高频振荡器作为时钟源开始执行指令；
3. 使能 OST 计数器对外部时钟计数 1024 个时钟周期；
4. OST 超时，等待内部时钟下降沿；
5. 系统时钟保持低电平直到新的时钟的下一个下降沿；
6. 系统时钟切换到外部时钟源。

1.4.10 外部时钟故障检测

故障保护时钟监视 (FSCM) 能使器件在振荡器发生故障时继续运行, 其可以检测出振荡器起振定时器 (OST) 延时结束后的任何时刻发生的振荡器故障。

将 OSCCTL 寄存器的 FSCM 位置 1 使能 FSCM, 适用于外部振荡器模式。

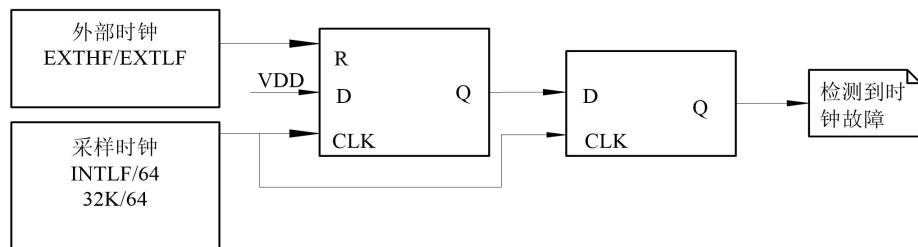


图 1.9 时钟故障检测原理图

FSCM 模块通过比较外部振荡器和采样时钟检测使用的外部振荡器。

FSCM 模块在采样时钟下降沿将第一个寄存器置 1, 在外部时钟下降沿将该寄存器复位为 0, 由于外部时钟频率远大于采样时钟频率, 所以当第一个寄存器刚被采样时钟置 1 不久就被外部时钟复位, 经过第二个寄存器(CLK 为采样时钟)后的 Q 值将保持为 0; 当发生外部时钟故障时, 第一个寄存器由于外部时钟故障而失去复位能力, 当采样时钟下降沿到达置 1 后, 将一直保持输出为 1, 检测到故障。

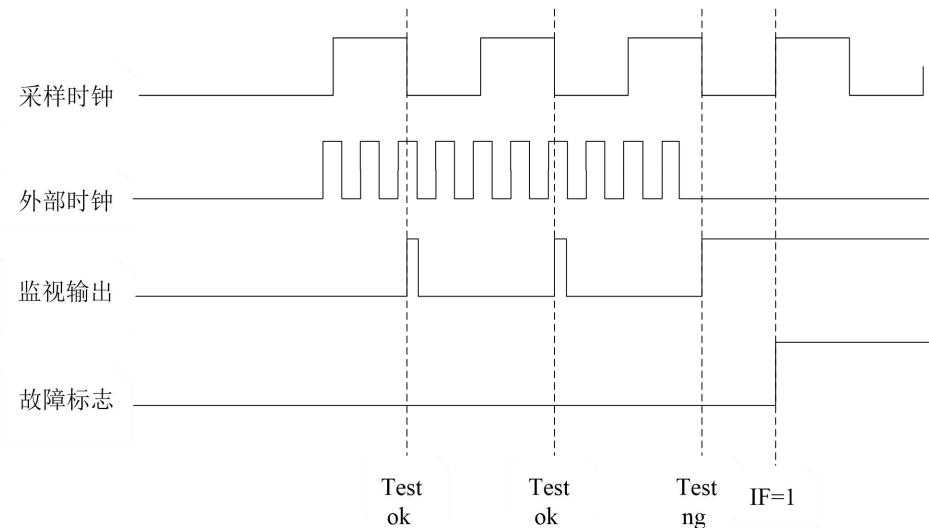


图 1.10 时钟故障检测时序

注: 系统时钟频率实际比采样时钟大很多, 图示为方便分析起见将频率差异减小。

1.4.11 时钟故障保护处理

检测到时钟故障后, 系统时钟将自动切换至内部高频时钟源继续工作, 分频值由 OSCCTL 寄存器的 IRCS<2:0>设置, 直到器件固件成功重启外部振荡器并使时钟重新切换

到外部振荡器为止。

在切换至内部高频时钟源后，时钟故障标志位置 1，如果打开时钟故障中断使能位，程序将进入中断行。

1.5 配置位

如下寄存器所示，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。KF8F513X 的配置位映射在 8007H 和 8008H 地址单元。

CONFIG: 配置字(地址:8007H)

R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P
—	—	PWMPIN	HPOL	LPOL	DEBUG	SWRTE N1	—	CODEP	LVREN	RSTEN	PWRT	WDTEN	DATAP	—	SWRTEN 0

bit15

bit8

bit0

注: R=编程器可读 P=编程时可写

PWMPIN: PWM5 复位控制

PWMPIN = 1 PWM 模块控制复位时的引脚输出状态

PWMPIN = 0 I/O 端口控制复位时的引脚状态

HPOL: PWM高边驱动器极性位 (high side polarity)

HPOL = 0 引脚P5CHxH引脚上的输出信号极性为低电平有效

HPOL = 1 引脚P5CHxH引脚上的输出信号极性为高电平有效

LPOL: PWM低边驱动器极性位 (low side polarity)

LPOL=0 引脚P5CHxL引脚上的输出信号极性位低电平有效

LPOL=1 引脚 P5CHxL 引脚上的输出信号极性位高电平有效

注: 复位默认高阻态

DEBUG: 在线调试使能位

DEBUG=1 禁止在线调试

DEBUG=0 使能在线调试

CODEP: 代码保护使能位

CODEP=1 禁止程序存储器代码保护

CODEP=0 使能程序存储器代码保护

LVREN: 欠压检测功能使能位

LVREN=1 使能欠压检查功能

LVREN=0 禁止欠压检查功能

RSTEN: P0.3/RST 引脚功能选择

RSTEN=1 P0.3/RST 引脚配置为外部复位输入

RSTEN=0 P0.3/RST 引脚功能为数字输入口

PWRT: 上电延时使能位

PWRT=0 使能上电延时

PWRT=1 禁止上电延时

WDTEN: 看门狗定时器(WDT)使能位

WDTEN=1 使能 WDT

WDTEN=0 禁止 WDT

DATAP: 数据存储区加密使能位

CPD=1 禁止数据存储区加密

CPD=0 使能数据存储区加密

SWRTEN<1:0>: 保留位

CONFIG: 校验字(地址:8008H)

R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P								
—	—	—	—	—	—	—	—	POR2	POR1	POR0	BOD4	BOD3	BOD2	BOD1	BOD0	

注： R=编程器可读 P=编程时可写

BOR<4:0>: BOD 电压校验位

POR<2:0>: POR 电压校验位

1.6 在线串行编程

如下图所示，在最终应用电路中可对 KF8F513X 单片机进行在线串行编程。实现编程需要五根线包括：复位线(RST)、时钟线(SPCLK)、数据线(SPDAT)、电源线(VDD)、地线(Vss)。

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。

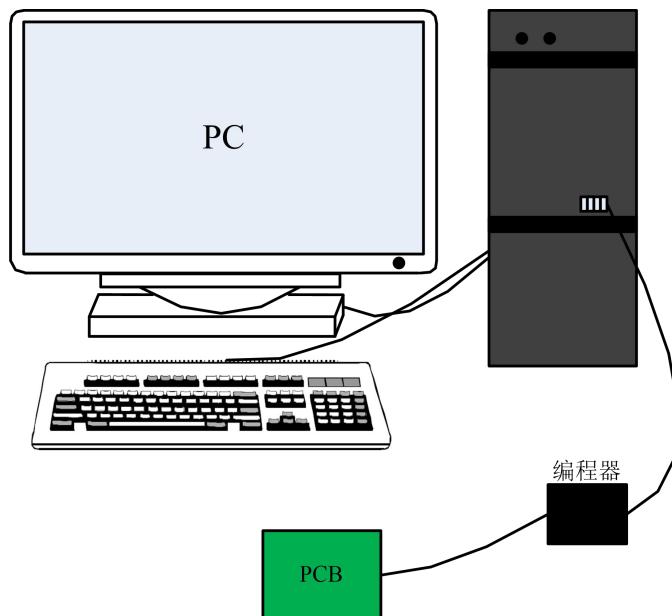


图 1.11 在线调试系统示意图

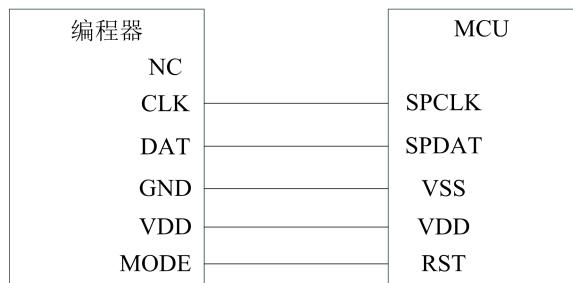


图 1.12 在线串行编程连接图

2 I/O 端口

KF8F513X 系列单片机相关引脚:

- 一组电源地 VDD/VSS
- 其余管脚均为 I/O 端口, 包括 P0~P5

2.1 I/O 端口的读写

读 P0~P7 口时实际为读端口电平, 即读 Px (x=0~7), 而写 P0~P7 时实际为写 PxLR(x=0~7)寄存器。其原理框图如图 2.1 所示:

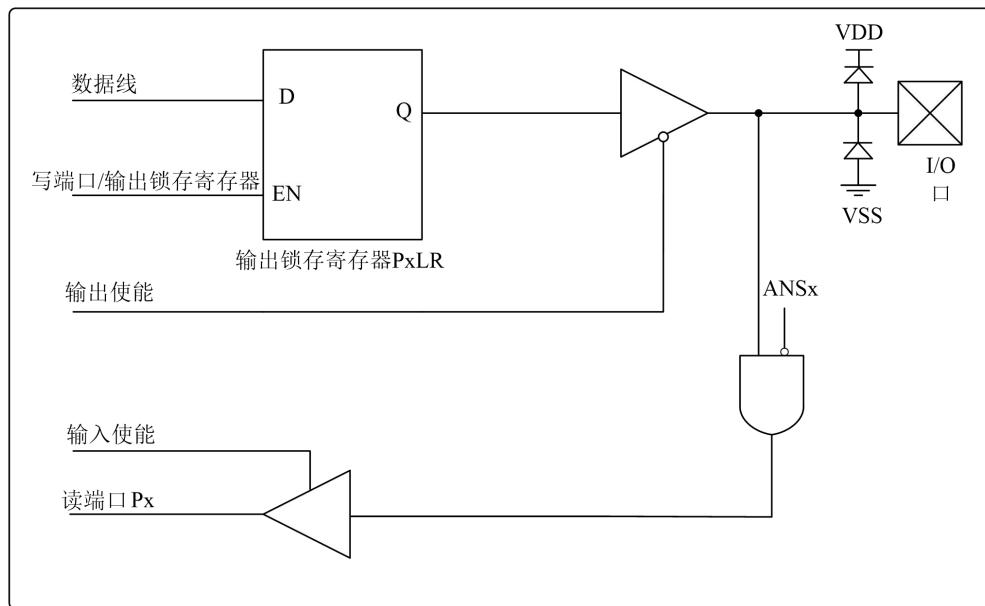


图 2.1 I/O 口读写原理图

IO 端口作为输出时, 可对 PxLR (x=0~7) 进行赋值(寄存器操作或位操作), 以避免原来的 IO 端口的读-修改-写操作引起的错误。

2.2 P0 口

在线编程时 P0 口的 SPCLK、SPDAT 作为编程脚使用。所有 P0 口均可作为普通 I/O 口且带有上拉功能和电平变化中断功能。

2.2.1 P0 口相关的寄存器

表 2-1 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	-	P06	P05	P04	P03	-	P01	P00
45H	P0LR	-	P0LR6	P0LR5	P0LR4	P0LR3	-	P0LR1	P0LR0
25H	TR0	-	TR06	TR05	TR04	TR03	-	TR01	TR00
36H	IOCL0	-	IOCL06	IOCL05	IOCL04	IOCL03	-	IOCL01	IOCL00
35H	PUR0	-	PUR06	PUR05	PUR04	PUR03	-	PUR01	PUR00
31H	ANS0	-	ANS06	ANS05	ANS04	ANS03	-	ANS01	ANS00

2.2.1.1 P0 口状态寄存器 (P0)

寄存器 P0 各位对应 P0 口相应引脚当前的状态。

寄存器2.1: P0: P0口状态寄存器(地址: 05H)

复位值	bit7	P06	P05	P04	P03	bit0	P01	P00
xxxx xxxx	-	R/W	R/W	R/W	R/W	-	R/W	R/W

P0<6:3><1:0>: 读 P0 口各端口电平

P0x=1 对应引脚为逻辑高电平

P0x=0 对应引脚为逻辑低电平

2.2.1.2 P0 口输出锁存寄存器 (P0LR)

寄存器 P0LR 是 P0 口输出锁存寄存器。在 P0 口作为输出时，通过写 P0LR 寄存器来设置输出 P0 口的状态。

寄存器2.2: P0LR: P0口输出锁存寄存器(地址: 45H)

复位值	bit7	P0LR6	P0LR5	P0LR4	P0LR3	bit0	P0LR1	P0LR0
xxxx xxxx	-	R/W	R/W	R/W	R/W	-	R/W	R/W

P0LR<6:3><1:0>: 写 P0 口输出状态

P0LR x=1 对应引脚输出高电平

P0LR x=0 对应引脚输出低电平

2.2.1.3 P0 口方向控制寄存器(TR0)

如寄存器 2.3 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2.3: TR0: P0口方向控制寄存器(地址: 25H)

复位值	-	TR06	TR05	TR04	TR03	-	TR01	TR00	bit0
1111 1111	-	R/W	R/W	R/W	R/W	-	R/W	R/W	

TR0<6:3><1:0>: 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

2.2.1.4 P0 上拉功能控制寄存器(PUR0)

KF8F513X 所有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 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 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 2.4 为上拉功能控制寄存器。

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

寄存器2.4: PUR0: P0口弱上拉控制寄存器(地址: 35H)

复位值	-	PUR06	PUR05	PUR04	PUR03	-	PUR01	PUR00	bit0
1111 1111	-	R/W	R/W	R/W	R/W	-	R/W	R/W	

PUR0<6:3><1:0>: 上拉功能使能位

PUR0x=1 使能对应的端口上拉功能

PUR0x=0 禁止对应的端口上拉功能

2.2.1.5 P0 口电平变化中断控制寄存器(IOCL0)

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 2.5 所示，IOCL0 为电平变化中断控制寄存器，将 IOCL0 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(P0IE)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口时才可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口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2.5: IOCL0: P0口电平变化中断控制寄存器(地址:36H)

复位值	bit7	IOCL06	IOCL05	IOCL04	IOCL03	bit0	IOCL01	IOCL00
0000 0000	-	R/W	R/W	R/W	R/W	-	R/W	R/W

IOCL0<6:3><1:0>: P0 端口引脚电平变化中断使能控制位

- IOCL0 x=1 使能对应引脚的电平变化中断
 IOCL0 x=0 禁止对应引脚的电平变化中断

2.2.1.6 P0 口模拟/数字口设置寄存器 (ANS0)

P0 口模拟/数字口设置寄存器 ANS0 用于将 P0 口设置为模拟口或者数字口，通过将 ANS0 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2.6: ANS0: P0口模拟/数字口设置寄存器(地址:31H)

复位值	bit7	ANS06	ANS05	ANS04	ANS03	bit0	ANS01	ANS00
1111 1111	-	R/W	R/W	R/W	R/W	-	R/W	R/W

ANS0<6:3><1:0>: P0 口各引脚模拟/数字口设置位

- 1 = 将对应引脚配置为模拟口
 0 = 将对应引脚配置为数字口或者特殊功能引脚

2.2.2 P0 口各引脚内部原理功能框图

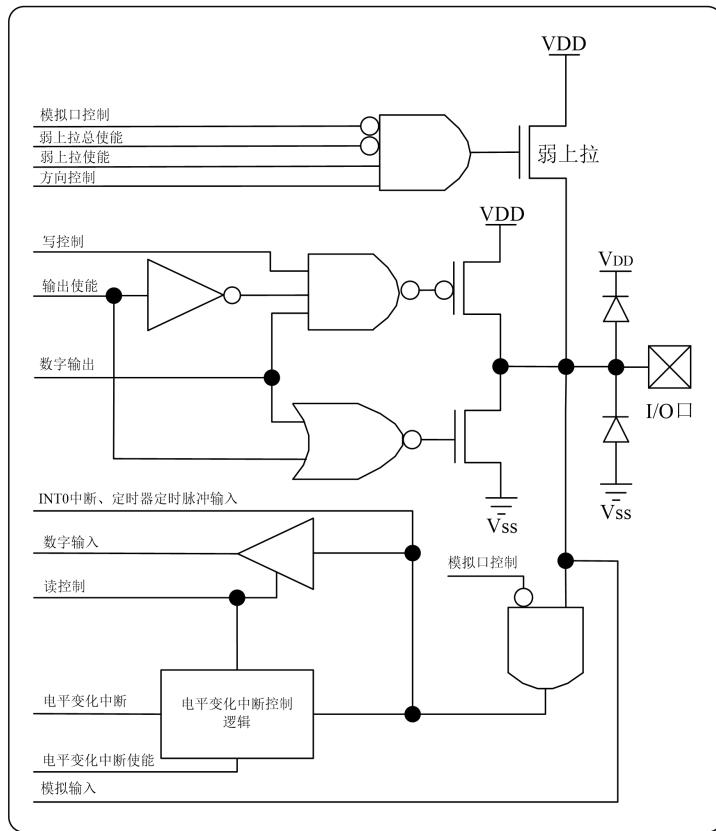


图 2.2 P0 口引脚原理框图

2.3 P1 口

P1 口所有管脚均可作为普通 I/O 口且带有上拉功能。

2.3.1 P1 口相关的寄存器

表 2-2 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
47H	P1LR	P1LR7	P1LR6	P1LR5	P1LR4	P1LR3	P1LR2	P1LR1	P1LR0
27H	TR1	TR17	TR16	TR15	TR14	TR13	TR12	TR11	TR10
60H	PUR1	PUR17	PUR16	PUR15	PUR14	PUR13	PUR12	PUR11	PUR10
1DH	ANS1	ANS17	ANS16	ANS15	ANS14	ANS13	ANS12	ANS11	ANS10

2.3.1.1 P1 口状态寄存器(P1)

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 2.7 所示

寄存器2.7: P1: P1口状态寄存器(地址: 07H)

复位值 XXXX XXXX	bit7								bit0							
	P17	P16	P15	P14	P13	P12	P11	P10	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

2.3.1.2 P1 口输出锁存寄存器 (P1LR)

寄存器 P1LR 是 P1 口输出锁存寄存器。在 P1 口作为输出时，通过写 P1LR 寄存器来设置输出 P1 口的状态。

寄存器2.8: P1LR: P1口输出锁存寄存器(地址: 47H)

复位值 XXXX XXXX	bit7								bit0							
	P1LR7	P1LR6	P1LR5	P1LR4	P1LR3	P1LR2	P1LR1	P1LR0	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

P1LR<7:0>: 写 P1 口输出状态

P1LRx=1 对应引脚输出高电平

P1LRx=0 对应引脚输出低电平

2.3.1.3 P1 口方向控制寄存器(TR1)

如寄存器 2.9 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2.9: TR1: P1口方向控制寄存器(地址: 27H)

复位值	TR17	TR16	TR15	TR14	TR13	TR12	TR11	TR10	bit0
1111 1111	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

2.3.1.4 P1 口上拉功能控制寄存器(PUR1)

KF8F513X 中 P1 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 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 PUPH (上拉功能总使能位)位清 0，允许 P1 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 2.10 为上拉功能控制寄存器。

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

寄存器2.10: PUR1: P1口弱上拉控制寄存器(地址: 60H)

复位值	PUR17	PUR16	PUR15	PUR14	PUR13	PUR12	PUR11	PUR10	bit0
1111 1111	R/W								

PUR1<7:0>: 上拉功能使能位

PUR1x=1 使能对应的端口上拉功能

PUR1x=0 禁止对应的端口上拉功能

2.3.1.5 P1 口模拟/数字口设置寄存器 (ANS1)

P1 口模拟/数字口设置寄存器 ANS1 用于将 P1 口设置为模拟口或者数字口，通过将 ANS1 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2.11: ANS1: P1口模拟/数字口设置寄存器(地址:1DH)

复位值	ANS17	ANS16	ANS15	ANS14	ANS13	ANS12	ANS11	ANS10	bit0
1111 1111	R/W								

ANS1<7:0>: P1 口各引脚模拟/数字口设置位

1 = 将对应引脚配置为模拟口

0 = 将对应引脚配置为数字口或者特殊功能引脚

2.3.2 P1 口原理功能框图

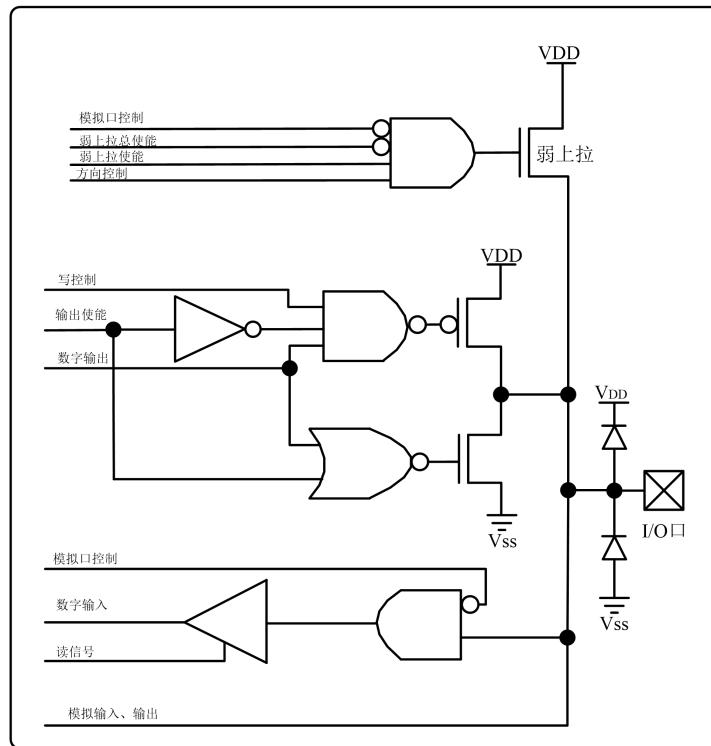


图 2.3 P1 口引脚原理功能框图

2.4 P2 口

P2 口所有管脚均可作为普通 I/O 口且带有上拉功能。

2.4.1 P2 口相关的寄存器

表 2-3 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	P23	P22	P21	P20
46H	P2LR	P2LR7	P2LR6	P2LR5	P2LR4	P2LR3	P2LR2	P2LR1	P2LR0
26H	TR2	TR27	TR26	TR25	TR24	TR23	TR22	TR21	TR20
61H	PUR2	PUR27	PUR26	PUR25	PUR24	PUR23	PUR22	PUR21	PUR20
32H	ANS2	ANS27	ANS26	ANS25	ANS24	ANS23	ANS22	ANS21	ANS20

2.4.1.1 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 2.12 所示：

寄存器2.12: P2: P2口状态寄存器(地址: 06H)

复位值 xxxx xxxx	bit7								bit0
	P27	P26	P25	P24	P23	P22	P21	P20	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

P2<7:0>: 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

2.4.1.2 P2 口输出锁存寄存器 (P2LR)

在 P2 口作为输出时，通过写 P2LR 寄存器来设置输出 P2 口的状态。

寄存器2.13: P2LR: P2口输出锁存寄存器(地址: 46H)

复位值 xxxx xxxx	bit7								bit0
	P2LR7	P2LR6	P2LR5	P2LR4	P2LR3	P2LR2	P2LR1	P2LR0	
	R/W	R/W							

P2LR<7:0>: 写 P2 口输出状态

P2LRx=1 对应引脚输出高电平

P2LRx=0 对应引脚输出低电平

2.4.1.3 P2 口方向控制寄存器(TR2)

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2.14: TR2: P2口方向控制寄存器(地址: 26H)

复位值 1111 1111	bit7 TR27	TR26	TR25	TR24	TR23	TR22	TR21	bit0 TR20
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

TR2<7:0>: 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 口对应引脚被配置为输出端口

2.4.1.4 P2 口上拉功能控制寄存器(PUR2)

KF8F513X 中 P2 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 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 PUPH (上拉功能总使能位)位清 0, 允许 P2 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 2.15 为上拉功能控制寄存器。

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

寄存器2.15: PUR2: P2口弱上拉控制寄存器(地址:61H)

复位值 1111 1111	bit7 PUR27	PUR26	PUR25	PUR24	PUR23	PUR22	PUR21	bit0 PUR20
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

PUR2<7:0>: 上拉功能使能位

PUR2x=1 使能对应的端口上拉功能

PUR2x=0 禁止对应的端口上拉功能

2.4.1.5 P2 口模拟/数字口设置寄存器 (ANS2)

P2 口模拟/数字口设置寄存器 ANS2 用于将 P2 口设置为模拟口或者数字口, 通过将 ANS2 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2.16: ANS2: P2口模拟/数字口设置寄存器(地址:32H)

复位值 1111 1111	bit7 ANS27	ANS26	ANS25	ANS24	ANS23	ANS22	ANS21	bit0 ANS20
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ANS2<7:0>: P2 口各引脚模拟/数字口设置位

1 = 将对应引脚配置为模拟口

0 = 将对应引脚配置为数字口或者特殊功能引脚

2.4.2 P2 口原理功能框图

如引脚示意图所示, P2 口共有 8 个引脚, 根据各引脚的作用不同, P2 口引脚原理功能框图中的模拟输入和输出有微小差别。

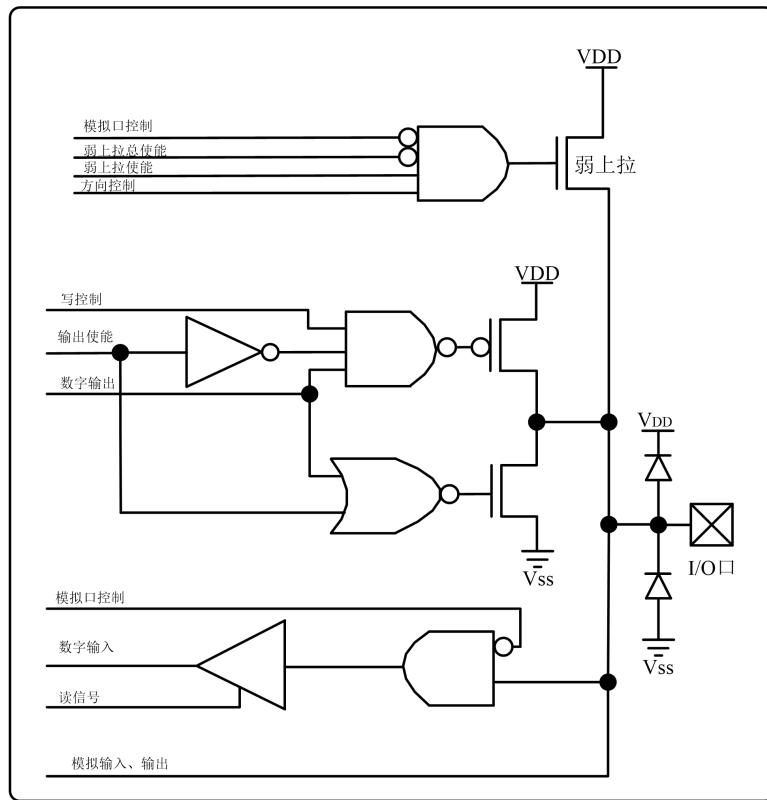


图 2.4 P2 口引脚原理功能框图

2.5 P3 口

P3 口所有管脚均可作为普通 I/O 口且带有上拉功能。

2.5.1 P3 口相关的寄存器

表 2-4 与 P3 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
08H	P3	P37	P36	P35	P34	P33	P32	P31	P30
48H	P3LR	P3LR7	P3LR6	P3LR5	P3LR4	P3LR3	P3LR2	P3LR1	P3LR0
49H	TR3	TR37	TR36	TR35	TR34	TR33	TR32	TR31	TR30
6CH	PUR3	PUR37	PUR36	PUR35	PUR34	PUR33	PUR32	PUR31	PUR30
33H	ANS3	ANS37	ANS36	ANS35	ANS34	ANS33	ANS32	ANS31	ANS30

2.5.1.1 P3 口状态寄存器(P3)

寄存器 P3 各位对应端口 P3 口各引脚作为普通 I/O 口时的状态。如寄存器 2.17 所示。

寄存器2.17: P3: P3口状态寄存器(地址:08H)

复位值 xxxx xxxx	bit7								bit0
	P37	P36	P35	P34	P33	P32	P31	P30	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

P3<7:0>: P3 口各引脚状态位

P3x=1 对应引脚为逻辑高电平

P3x=0 对应引脚为逻辑低电平

2.5.1.2 P3 口输出锁存寄存器 (P3LR)

寄存器 P3LR 是 P3 口输出锁存寄存器。在 P3 口作为输出时，我们是通过写 P3LR 寄存器来设置输出 P3 口的状态。

寄存器2.18: P3LR: P3口输出锁存寄存器(地址:48 H)

复位值 xxxx xxxx	bit7								bit0
	P3LR7	P3LR6	P3LR5	P3LR4	P3LR3	P3LR2	P3LR1	P3LR0	
	R/W	R/W							

P3LR<7:0>: 写 P3 口输出状态

P3LRx=1 对应引脚输出高电平

P3LRx=0 对应引脚输出低电平

2.5.1.3 P3 口方向控制寄存器(TR3)

如寄存器 2.19 所示，通过将寄存器 TR3 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2.19: TR3: P3口方向控制寄存器(地址: 49H)

复位值	bit7	TR37	TR36	TR35	TR34	TR33	TR32	TR31	bit0	TR30
1111 1111		R/W		R/W						

TR3<7:0>: P3 口各引脚方向控制位

TR3x=1 P3 口对应引脚被配置为输入端口

TR3x=0 P3 口对应引脚被配置为输出端口

2.5.1.4 P3 口上拉功能控制寄存器(PUR3)

KF8F513X 中 P3 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{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{PUPH} (上拉功能总使能位)位清 0，允许 P3 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 2.20 为上拉功能控制寄存器。

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

寄存器2.20: PUR3: P3口弱上拉控制寄存器(地址:6CH)

复位值	bit7	PUR37	PUR36	PUR35	PUR234	PUR33	PUR32	PUR31	bit0	PUR30
1111 1111		R/W	R/W	R/W	R/W	R/W	R/W	R/W		R/W

PUR3<7:0>: 上拉功能使能位

PUR3x=1 使能对应的端口上拉功能

PUR3x=0 禁止对应的端口上拉功能

2.5.1.5 P3 口模拟/数字口设置寄存器 (ANS3)

P3 口模拟/数字口设置寄存器 ANS3 用于将 P3 口设置为模拟口或者数字口，通过将 ANS3 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2.22: ANS3: P3口模拟/数字口设置寄存器(地址:33H)

复位值	bit7	ANS37	ANS36	ANS35	ANS34	ANS33	ANS32	ANS31	bit0	ANS30
1111 1111		R/W		R/W						

ANS3<7:0>: P3 口各引脚模拟/数字口设置位

- 1 = 将对应引脚配置为模拟口
- 0 = 将对应引脚配置为数字口或者特殊功能引脚

2.5.2 P3 口原理功能框图

如图 2.5 所示, P3 口共有 8 个引脚, 根据各引脚的作用不同, P3 口引脚原理功能框图中的模拟输入和输出有微小差别。

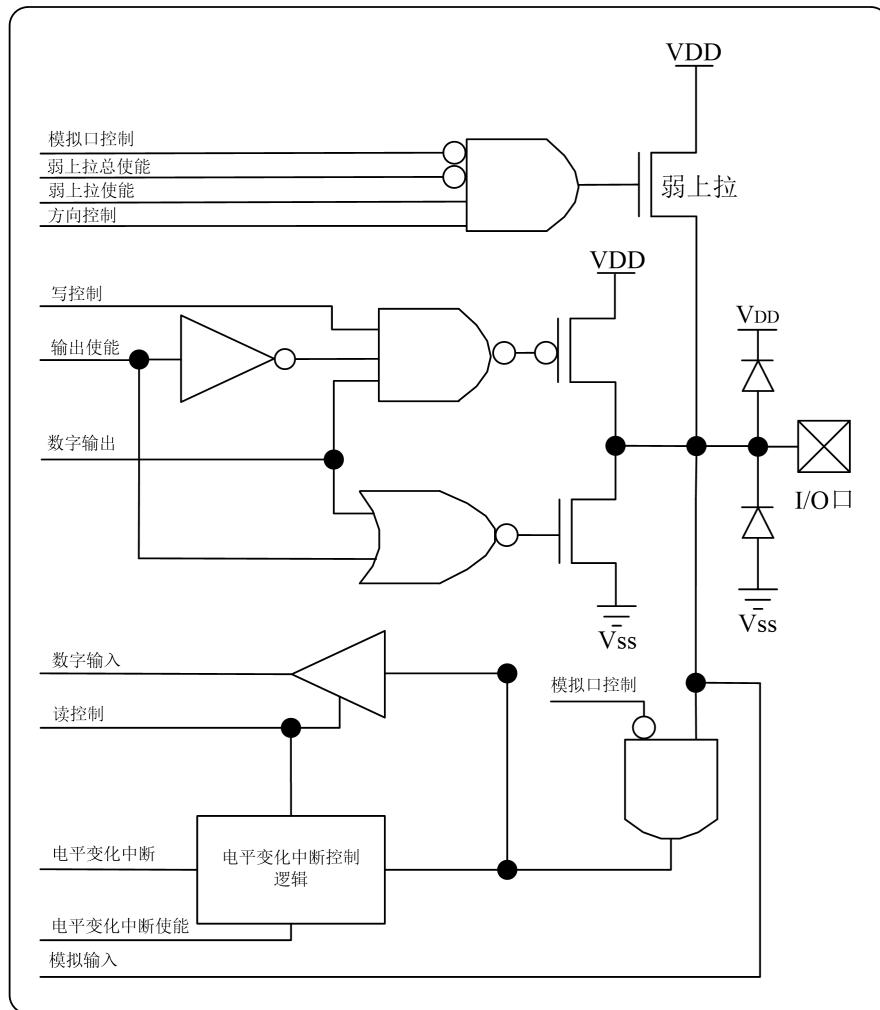


图 2.5 口引脚原理功能框图

2.6 P4 口

P4 口所有管脚均可作为普通 I/O 口且带有上拉功能和 I/O 口电平变化中断功能。

2.6.1 P4 口相关的寄存器

表 2-5 与 P4 口相关寄存器

地址	名称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
20CH	P4	P47	P46	P45	P44	P43	P42	P41	P40
20DH	P4LR	P4LR7	P4LR6	P4LR5	P4LR4	P4LR3	P4LR2	P4LR1	P4LR0
20EH	TR4	TR47	TR46	TR45	TR44	TR43	TR42	TR41	TR40
20FH	PUR4	PUR47	PUR46	PUR45	PUR44	PUR43	PUR42	PUR41	PUR40
206H	ANS4	ANS47	ANS46	ANS45	ANS44	ANS43	ANS42	ANS41	ANS40
6BH	IOCL4	IOCL47	IOCL46	IOCL45	IOCL44	IOCL43	IOCL42	IOCL41	IOCL40

2.6.1.1 P4 口状态寄存器(P4)

寄存器 P4 各位对应端口 P4 口各引脚作为普通 I/O 口时的状态。如寄存器 2.23 所示：

寄存器2.23: P4: P4口状态寄存器(地址: 20CH)

复位值 XXXX XXXX	bit7								bit0
	P47	P46	P45	P44	P43	P42	P41	P40	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

P4<7:0>: P4 口各引脚状态位

P4x=1 对应引脚为逻辑高电平

P4x=0 对应引脚为逻辑低电平

2.6.1.2 P4 口输出锁存寄存器 (P4LR)

寄存器 P4LR 是 P4 口输出锁存寄存器。在 P4 口作为输出时，我们是通过写 P4LR 寄存器来设置输出 P4 口的状态。

寄存器2.24: P4LR: P4口输出锁存寄存器(地址:20DH)

复位值 XXXX XXXX	bit7								bit0
	P4LR7	P4LR6	P4LR5	P4LR4	P4LR3	P4LR2	P4LR1	P4LR0	
	R/W								

P4LR<7:0>: 写 P4 口输出状态

P4LRx=1 对应引脚输出高电平

P4LRx=0 对应引脚输出低电平

2.6.1.3P4 口方向控制寄存器(TR4)

如寄存器 2.25 所示, 通过将寄存器 TR4 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2.25: TR4: P4口方向控制寄存器(地址: 20EH)

复位值	bit7	TR47	TR46	TR45	TR44	TR43	TR42	TR41	TR40	bit0
1111 1111		R/W								

TR4<7:0>: P4 口各引脚方向控制位

TR4x=1 P4 口对应引脚被配置为输入端口

TR4x=0 P4 口对应引脚被配置为输出端口

2.6.1.4P4 口上拉功能控制寄存器(PUR4)

KF8F513X 中 P4 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 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 PUPH (上拉功能总使能位)位清 0, 允许 P4 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 2.26 为上拉功能控制寄存器。

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

寄存器2.26: PUR4: P4口弱上拉控制寄存器(地址:20FH)

复位值	bit7	PUR47	PUR46	PUR45	PUR44	PUR43	PUR42	PUR41	PUR40	bit0
1111 1111		R/W								

PUR4<7:0>: 上拉功能使能位

PUR4x=1 使能对应的端口上拉功能

PUR4x=0 禁止对应的端口上拉功能

2.6.1.5P4 口电平变化中断控制寄存器(IOCL4)

P4 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 P4 寄存器时的电平不匹配时将产生电平变化中断。IOCL4 为 P4 口电平变化中断控制寄存器, 将 IOCL4 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(P4IF)都会置 1, 如果全局中断使能位(AIE)和电平变化中断使能位(P4IE)都已置 1, 则会响应中断进入中断服务子程序。P4 口所有引脚的电平变化中断共用一个标志位 P4IF。

注：1. 只有将引脚设置为数字输入口时才可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口时将会自动禁止该引脚的电平变化中断功能。
 2. P4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2.21: IOCL4: P4口电平变化中断控制寄存器(地址:6BH)

复位值 0000 0000	bit7								bit0
	IOCL47	IOCL46	IOCL45	IOCL44	IOCL43	IOCL42	IOCL41	IOCL40	
	R/W								

IOCL4<7:0>: P4 端口引脚电平变化中断使能控制位

- IOCL4=1 使能对应引脚的电平变化中断
- IOCL4=0 禁止对应引脚的电平变化中断

2.6.1.6 P4 口模拟/数字口设置寄存器 (ANS4)

P4 口模拟/数字口设置寄存器 ANS4 用于将 P4 口设置为模拟口或者数字口，通过将 ANS4 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2.27: ANS4: P4口模拟/数字口设置寄存器(地址:206H)

复位值 1111 1111	bit7								bit0
	ANS47	ANS46	ANS45	ANS44	ANS43	ANS42	ANS41	ANS40	
	R/W								

ANS4<7:0>: P4 口各引脚模拟/数字口设置位

- 1 = 将对应引脚配置为模拟口
- 0 = 将对应引脚配置为数字口或者特殊功能引脚

2.6.2 P4 口原理功能框图

根据各引脚的作用不同，P4 口引脚原理功能框图中的模拟输入和输出有微小差别。

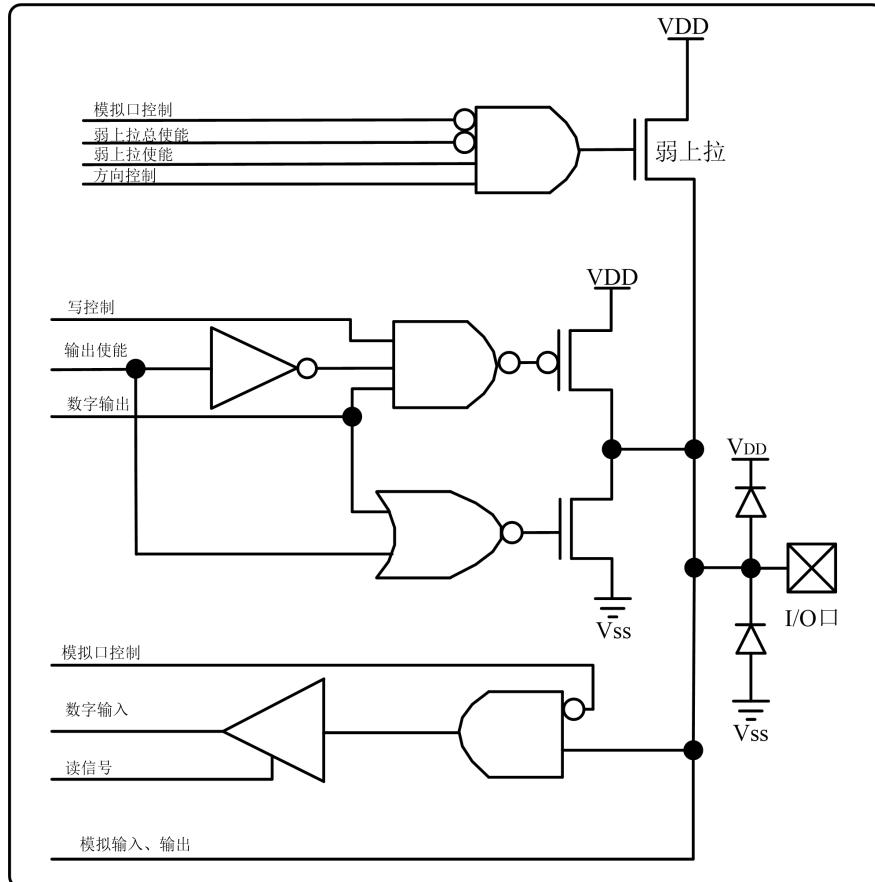


图 2.6 P4 口引脚原理功能框图

2.7 P5 口

P5 口所有管脚均可作为普通 I/O 口且带有上拉功能。

2.7.1 P5 口相关的寄存器

表 2-6 与 P5 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
210H	P5	P57	P56	P55	P54	P53	P52	P51	P50
211H	P5LR	P5LR7	P5LR6	P5LR5	P5LR4	P5LR3	P5LR2	P5LR1	P5LR0
212H	TR5	TR57	TR56	TR55	TR54	TR53	TR52	TR51	TR50
213H	PUR5	PUR57	PUR56	PUR55	PUR54	PUR53	PUR52	PUR51	PUR50
207H	ANS5	ANS57	ANS56	ANS55	ANS54	ANS53	ANS52	ANS51	ANS50

2.7.1.1 P5 口状态寄存器(P5)

寄存器 P5 各位对应端口 P5 口各引脚作为普通 I/O 口时的状态。如寄存器 2.28 所示：

寄存器 2.28: P5: P5 口状态寄存器(地址: 210H)

复位值 xxxx xxxx	bit7								bit0
	P57	P56	P55	P54	P53	P52	P51	P50	-
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	-

P5<7:0>: P5 口各引脚状态位

1 = 对应引脚为逻辑高电平

0 = 对应引脚为逻辑低电平

2.7.1.2 P5 口输出锁存寄存器 (P5LR)

寄存器 P5LR 是 P5 口输出锁存寄存器。在 P5 口作为输出时，我们是通过写 P5LR 寄存器来设置输出 P5 口的状态。

寄存器 2.29: P5LR: P5 口输出锁存寄存器(地址: 211H)

复位值 xxxx xxxx	bit7								bit0
	P5LR7	P5LR6	P5LR5	P5LR4	P5LR3	P5LR2	P5LR1	P5LR0	-
	R/W	-							

P5LR<7:0>: 写 P5 口输出状态

1 = 对应引脚输出高电平

0 = 对应引脚输出低电平

2.7.1.3 P5 口方向控制寄存器(TR5)

如寄存器 2.30 所示，通过将寄存器 TR5 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2.30: TR5: P5口方向控制寄存器(地址: 212H)

	bit7	TR57	TR56	TR55	TR54	TR53	TR52	TR51	TR50	bit0
复位值 1111 1111		R/W								

TR5<7:0>: P5 口各引脚方向控制位

- 1 = P5 口对应引脚被配置为输入端口
0 = P5 口对应引脚被配置为输出端口

2.7.1.4 P5 口上拉功能控制寄存器(PUR5)

KF8F513X 中 P5 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 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 PUPH (上拉功能总使能位)位清 0，允许 P5 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 2.31 为上拉功能控制寄存器。

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

寄存器2.31: PUR5: P5口弱上拉控制寄存器(地址:213H)

	bit7	PUR57	PUR56	PUR55	PUR54	PUR53	PUR52	PUR51	PUR50	bit0
复位值 1111 1111		R/W								

PUR5<7:0>: 上拉功能使能位

- 1 = 使能对应的端口上拉功能
0 = 禁止对应的端口上拉功能

2.7.1.5 P5 口模拟/数字口设置寄存器 (ANS5)

P5 口模拟/数字口设置寄存器 ANS5 用于将 P5 口设置为模拟口或者数字口，通过将 ANS5 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2.32: ANS5: P5口模拟/数字口设置寄存器(地址:207H)

	bit7	ANS57	ANS56	ANS55	ANS54	ANS53	ANS52	ANS51	ANS50	bit0
复位值 1111 1111		R/W								

ANS5<7:0>: P5 口各引脚模拟/数字口设置位

1 = 将对应引脚配置为模拟口

0 = 将对应引脚配置为数字口或者特殊功能引脚

2.7.2 P5 口原理功能框图

根据各引脚的作用不同, P5 口引脚原理功能框图中的模拟输入和输出有微小差别。

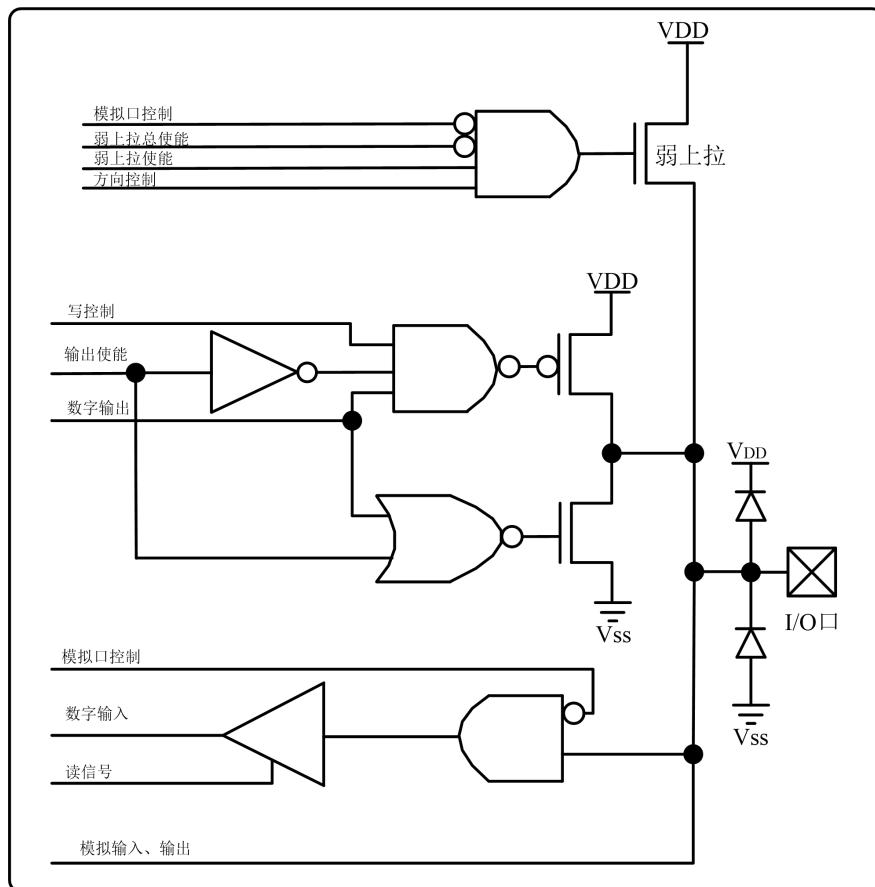


图 2.7 P5 口引脚原理功能框图

2.8 引脚复用功能

KF8F513X 系列单片机的部分外设功能引脚提供丰富的引脚复用功能。如下外设相关引脚带有复用功能，请根据相关寄存器自行配置选择。

- FVR 输出引脚
- ADC 外部参考电压输入引脚 VREFIN
- Power18 输出引脚
- 指令时钟输出引脚 CLKOUT
- CCP5 模块相关引脚 P5A、P5B、P5C、P5D 和捕捉比较 CCP5 引脚
- PWM1/PWM2/PWM3 输出引脚
- USART1/USART2 相关引脚
- SSC1 相关引脚
- 定时器相关引脚 T0CK、T1CK 和 T1G
- 外部中断输入引脚 INT0、INT1 和 INT2
- LED 公共端 COMx 引脚

2.8.1 引脚复用功能相关寄存器

表 2-7 引脚复用功能相关寄存器表

地址	名称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
260H	APFCTL0	FVROE1	FVROE0	VREFINS EL1	VREFINS EL0	P18OE1	P18OE0	CLKOUT SEL1	CLKOUT SEL0
261H	APFCTL1	CCP3SEL 2	CCP3SEL 1	CCP3SEL 0	CCP4SEL 2	CCP4SEL 1	CCP4SEL 0	-	-
262H	APFCTL2	P5DSEL1	P5DSEL0	P5CSEL1	P5CSEL0	P5BSEL1	P5BSEL0	P5ASEL1	P5ASEL0
263H	APFCTL3	PWM33 SEL	PWM32 SEL	PWM31 SEL	-	PWM22 SEL	PWM21 SEL	PWM20 SEL1	PWM20 SEL0
264H	APFCTL4	-	U1TCSE L2	U1TCSE L1	U1TCSE L0	-	-	U1RDSE L1	U1RDSE L0
265H	APFCTL5	-	-	-	-	U2TCSE L1	U2TCSE L0	U2RDSE L1	U2RDSE L0
266H	APFCTL6	-	SDOSEL 2	SDOSEL 1	SDOSEL 0	-	SCKSEL2	SCKSEL1	SCKSEL0
267H	APFCTL7	-	-	-	SSSEL	SDISEL3	SDISEL2	SDISEL1	SDISEL0
268H	APFCTL8	BUZ3SE L	T1CKSE L2	T1CKSE L1	T1CKSE L0	T1GSEL1	T1GSEL0	T0CKSE L1	T0CKSE L0
269H	APFCTL9	-	INT1SEL 2	INT1SEL 1	INT1SEL 0	INT2SEL 1	INT2SEL 0	INT0SEL 1	INT0SEL 0
26AH	APFCTLA	COM7 SEL	COM6 SEL	COM5 SEL	COM4 SEL	COM3 SEL	COM2 SEL	COM1 SEL	COM0 SEL
26BH	APFCTLB	-	-	-	-	-	CCP5 SEL2	CCP5 SEL1	CCP5 SEL0

2.8.1.1 引脚复用功能控制寄存器 APFCTL0

寄存器: APFCTL0: 引脚复用功能控制寄存器0(地址: 260H)

复位值 0000 0000								bit0 R/W
	bit7 FVROE1 R/W	FVROE0 R/W	VREFINSEL1 R/W	VREFINSEL0 R/W	P18OE1 R/W	P18OE0 R/W	CLKOUTSEL1 R/W	CLKOUTSEL0 R/W

FVROE<1:0>: FVR 输入引脚选择位

FVROE0 = 1

FVR 功能位于 P0.4 引脚上

FVROE1 = 1

FVR 功能位于 P0.5 引脚上

注: 不建议将 FVROE0 位和 FVROE1 位同时置 1。

VREFINSEL<1:0>: VREF 输入引脚选择位

00 = VREF 功能位于 P0.0 引脚上

01 = VREF 功能位于 P0.3 引脚上

10 = VREF 功能位于 P4.3 引脚上

P18OE<1:0>: P18 输出引脚选择位

P18OE0 = 1

P18 功能位于 P0.4 引脚上

P18OE1 = 1

P18 功能位于 P2.7 引脚上

CLKOUTSEL<1:0>: SYSCLK 输入引脚选择位

00 = SYSCLK 功能位于 P0.4 引脚上

01 = SYSCLK 功能位于 P0.5 引脚上

10 = SYSCLK 功能位于 P4.5 引脚上

其他 = 保留, 配置后将无法输出

2.8.1.2 引脚复用功能控制寄存器 APFCTL1

寄存器: APFCTL1: 引脚复用功能控制寄存器1(地址: 261H)

复位值 0000 0000								bit0 R/W
	bit7 CCP3SEL2 R/W	CCP3SEL1 R/W	CCP3SEL0 R/W	CCP4SEL2 R/W	CCP4SEL1 R/W	CCP4SEL0 R/W	- R/W	- R/W

CCP3SEL<2:0>: CCP3 功能引脚选择位

000 = CCP3 功能位于 P0.4 引脚上

001 = CCP3 功能位于 P1.2 引脚上

010 = CCP3 功能位于 P3.2 引脚上

011 = CCP3 功能位于 P3.4 引脚上

100 = CCP3 功能位于 P3.7 引脚上

101 = CCP3 功能位于 P4.0 引脚上

110 = CCP3 功能位于 P4.2 引脚上

111 = CCP3 功能位于 P4.3 引脚上

CCP4SEL<2:0>: CCP4 功能引脚选择位

- 000 = CCP4 功能位于 P0.4 引脚上
- 001 = CCP4 功能位于 P1.2 引脚上
- 010 = CCP4 功能位于 P3.2 引脚上
- 011 = CCP4 功能位于 P3.4 引脚上
- 100 = CCP4 功能位于 P3.7 引脚上
- 101 = CCP4 功能位于 P4.0 引脚上
- 110 = CCP4 功能位于 P4.2 引脚上
- 111 = CCP4 功能位于 P4.3 引脚上

2.8.1.3 引脚复用功能控制寄存器 APFCTL2

寄存器: APFCTL2: 引脚复用功能控制寄存器2(地址: 262H)

	bit7								bit0
复位值	P5DSEL1	P5DSEL0	P5CSEL1	P5CSEL0	P5BSEL1	P5BSEL0	P5ASEL1	P5ASEL0	
0000 0000	R/W								

P5DSEL<1:0>: P5D 功能引脚选择位

- 00 = P5D 功能位于 P1.5 引脚上
- 01 = P5D 功能位于 P3.6 引脚上
- 10 = P5D 功能位于 P5.3 引脚上
- 其他 = 保留, 配置后将无法输出

P5CSEL<1:0>: P5C 功能引脚选择位

- 00 = P5C 功能位于 P1.4 引脚上
- 01 = P5C 功能位于 P2.2 引脚上
- 10 = P5C 功能位于 P2.3 引脚上
- 其他 = 保留, 配置后将无法输出

P5BSEL<1:0>: P5B 功能引脚选择位

- 00 = P5B 功能位于 P1.3 引脚上
- 01 = P5B 功能位于 P2.1 引脚上
- 10 = P5B 功能位于 P2.5 引脚上
- 其他 = 保留, 配置后将无法输出

P5ASEL<1:0>: P5A 功能引脚选择位

- 00 = P5A 功能位于 P1.2 引脚上
- 01 = P5A 功能位于 P2.0 引脚上
- 10 = P5A 功能位于 P3.0 引脚上
- 其他 = 保留, 配置后将无法输出

2.8.1.4 引脚复用功能控制寄存器 APFCTL3

寄存器: APFCTL3: 引脚复用功能控制寄存器3(地址: 263H)

复位值 0000 0000	bit7								bit0
	PWM33SE L	PWM32SE L	PWM31SE L	-	PWM22SE L	PWM21SE L	PWM20SE L1	PWM20SE L0	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

PWM33SEL: PWM33 功能引脚选择位

0 = PWM33 功能位于 P1.2 引脚上

1 = PWM33 功能位于 P4.0 引脚

PWM32SEL: PWM32 功能引脚选择位

0 = PWM32 功能位于 P4.2 引脚上

1 = PWM32 功能位于 P4.4 引脚

PWM31SEL: PWM31 功能引脚选择位

0 = PWM31 功能位于 P4.3 引脚上

1 = PWM31 功能位于 P5.0 引脚

PWM22SEL: PWM22 功能引脚选择位

0 = PWM22 功能位于 P3.4 引脚上

1 = PWM22 功能位于 P5.7 引脚上

PWM21SEL: PWM21 功能引脚选择位

0 = PWM21 功能位于 P3.1 引脚上

1 = PWM21 功能位于 P3.2 引脚上

PWM20SEL<1:0>: PWM20 功能引脚选择位

00 = PWM20 功能位于 P0.0 引脚上

01 = PWM20 功能位于 P1.1 引脚上

10 = PWM20 功能位于 P3.7 引脚上

其他 = 保留, 配置后将无法输出

2.8.1.5 引脚复用功能控制寄存器 APFCTL4

寄存器: APFCTL4: 引脚复用功能控制寄存器4(地址: 264H)

复位值 0000 0000	bit7								bit0
	-	U1TCSEL2	U1TCSEL1	U1TCSEL0	-	-	U1RDSEL1	U1RDSEL0	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

U1TCSEL<2:0>: USART1 的 TX1/CK1 功能引脚选择位

000 = TX1/CK1 功能位于 P0.0 引脚上

001 = TX1/CK1 功能位于 P0.6 引脚上

010 = TX1/CK1 功能位于 P1.0 引脚上

011 = TX1/CK1 功能位于 P1.6 引脚上

100 = TX1/CK1 功能位于 P2.1 引脚上

101 = TX1/CK1 功能位于 P4.7 引脚上

其他 = 保留, 配置后将无法输出

U1RDSEL<1:0>:	USART1 的 RX1/DT1 功能引脚选择位
	00 = RX1/DT1 功能位于 P0.1 引脚上
	01 = RX1/DT1 功能位于 P1.4 引脚上
	10 = RX1/DT1 功能位于 P2.0 引脚上
	11 = RX1/DT1 功能位于 P5.7 引脚上

2.8.1.6 引脚复用功能控制寄存器 APFCTL5

寄存器: APFCTL5: 引脚复用功能控制寄存器5(地址: 265H)

复位值	bit7	-	-	-	-	U2TCSEL1	U2TCSEL0	U2RDSEL1	U2RDSEL0	bit0
0000 0000	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

U2TCSEL<1:0>: USART2 的 TX2/CK2 功能引脚选择位

- 00 = TX2/CK2 功能位于 P0.3 引脚上
- 01 = TX2/CK2 功能位于 P3.5 引脚上
- 10 = TX2/CK2 功能位于 P3.7 引脚上
- 11 = TX2/CK2 功能位于 P4.4 引脚上

U2RDSEL<1:0>: USART2 的 RX2/DT2 功能引脚选择位

- 00 = RX2/DT2 功能位于 P0.4 引脚上
- 01 = RX2/DT2 功能位于 P3.4 引脚上
- 10 = RX2/DT2 功能位于 P4.0 引脚上
- 11 = RX2/DT2 功能位于 P4.3 引脚上

2.8.1.7 引脚复用功能控制寄存器 APFCTL6

寄存器: APFCTL6: 引脚复用功能控制寄存器6(地址: 266H)

复位值	bit7	-	SDOSEL2	SDOSEL1	SDOSEL0	-	SCKSEL2	SCKSEL1	SCKSEL0	bit0
0000 0000	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

SDOSEL<2:0>: SDO 功能引脚选择位

- 000 = SDO 功能位于 P1.7 引脚上
- 001 = SDO 功能位于 P2.0 引脚上
- 010 = SDO 功能位于 P3.3 引脚上
- 011 = SDO 功能位于 P3.5 引脚上
- 100 = SDO 功能位于 P3.7 引脚上
- 101 = SDO 功能位于 P4.5 引脚上
- 其他 = 保留, 配置后将无法输出

SCKSEL<2:0>: SCL/SCK 功能引脚选择位

- 000 = SCL/SCK 功能位于 P0.0 引脚上
- 001 = SCL/SCK 功能位于 P1.0 引脚上
- 010 = SCL/SCK 功能位于 P1.5 引脚上
- 011 = SCL/SCK 功能位于 P2.1 引脚上
- 100 = SCL/SCK 功能位于 P3.2 引脚上

101 = SCL/SCK 功能位于 P4.0 引脚上

110 = SCL/SCK 功能位于 P4.2 引脚上

其他 = 保留, 配置后将无法输出

2.8.1.8 引脚复用功能控制寄存器 APFCTL7

寄存器: APFCTL7: 引脚复用功能控制寄存器7(地址: 267H)

复位值 0000 0000	bit7								bit0
	-	-	-	SSSEL	SDISEL3	SDISEL2	SDISEL1	SDISEL0	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

SSSEL: SS 功能引脚选择位

0 = SS 功能位于 P0.5 引脚上

1 = SS 功能位于 P2.3 引脚上

SDISEL<3:0>: SDA/SDI 功能引脚选择位

0000 = SDA/SDI 功能位于 P0.1 引脚上

0001 = SDA/SDI 功能位于 P1.1 引脚上

0010 = SDA/SDI 功能位于 P1.3 引脚上

0011 = SDA/SDI 功能位于 P1.4 引脚上

0100 = SDA/SDI 功能位于 P2.2 引脚上

0101 = SDA/SDI 功能位于 P2.3 引脚上

0110 = SDA/SDI 功能位于 P3.4 引脚上

0111 = SDA/SDI 功能位于 P3.7 引脚上

1000 = SDA/SDI 功能位于 P4.0 引脚上

1001 = SDA/SDI 功能位于 P4.6 引脚上

其他 = 保留, 配置后将无法输出

2.8.1.9 引脚复用功能控制寄存器 APFCTL8

寄存器: APFCTL8: 引脚复用功能控制寄存器8(地址: 268H)

复位值 0000 0000	bit7								bit0
	BUZ3SEL	T1CKSEL2	T1CKSEL1	TICKSEL0	T1GSEL1	T1GSEL0	T0CKSEL1	T0CKSEL0	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

BUZ3SEL: BUZ3 输出功能引脚选择位

0 = BUZ3 输入功能位于 P4.2 引脚上

1 = BUZ3 输入功能位于 P4.4 引脚上

T1CKSEL<2:0>: T1CK 输入功能引脚选择位

000 = T1CK 输入功能位于 P0.4 引脚上

001 = T1CK 输入功能位于 P0.5 引脚上

010 = T1CK 输入功能位于 P1.1 引脚上

011 = T1CK 输入功能位于 P3.6 引脚上

100 = T1CK 输入功能位于 P4.2 引脚上

其他 = 保留, 配置后将无法输出

T1GSEL<1:0>:	T1G 输入功能引脚选择位							
	00 = T1G 输入功能位于 P0.4 引脚上							
	01 = T1G 输入功能位于 P0.5 引脚上							
	10 = T1G 输入功能位于 P4.5 引脚上							
	其他= 保留, 配置后将无法输出							
T0CKSEL<1:0>:	T0CK 输入功能引脚选择位							
	00 = T0CK 输入功能位于 P0.3 引脚上							
	01 = T0CK 输入功能位于 P0.6 引脚上							
	10 = T0CK 输入功能位于 P3.3 引脚上							
	11 = T0CK 输入功能位于 P4.2 引脚上							

2.8.1.10 引脚复用功能控制寄存器 APFCTL9

寄存器: APFCTL9: 引脚复用功能控制寄存器9(地址: 269H)

复位值 0000 0000	bit7	bit0							
	-	INT1SEL2	INT1SEL1	INT1SEL0	INT2SEL1	INT2SEL0	INT0SEL1	INT0SEL0	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

INT1SEL<2:0>:	INT1 输入引脚选择位							
	000 = INT1 功能位于 P1.0 引脚上							
	001 = INT1 功能位于 P1.1 引脚上							
	010 = INT1 功能位于 P1.2 引脚上							
	011 = INT1 功能位于 P2.5 引脚上							
	100 = INT1 功能位于 P3.3 引脚上							
	101 = INT1 功能位于 P3.6 引脚上							
	110 = INT1 功能位于 P5.7 引脚上							
	其他= 保留, 配置后该功能将失效							
INT2SEL<1:0>:	INT2 输入引脚选择位							
	00 = INT2 功能位于 P1.3 引脚上							
	01 = INT2 功能位于 P2.2 引脚上							
	10 = INT2 功能位于 P2.4 引脚上							
	11 = INT2 功能位于 P4.1 引脚上							
INT0SEL<1:0>:	INT0 输入引脚选择位							
	00 = INT0 功能位于 P0.3 引脚上							
	01 = INT0 功能位于 P0.6 引脚上							
	10 = INT0 功能位于 P3.3 引脚上							
	11 = INT0 功能位于 P4.5 引脚上							

2.8.1.11 引脚复用功能控制寄存器 APFCTLA

寄存器: APFCTLA: 引脚复用功能控制寄存器A(地址: 26AH)

复位值 0000 0000	bit7	bit0							
	-	COM7SEL	COM6SEL	COM5SEL	COM4SEL	COM3SEL	COM2SEL	COM1SEL	COM0SEL
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

COM7SEL:	COM7 输引脚选择位
	0 = COM7 功能位于 P3.0 引脚上
	1 = COM7 功能位于 P1.0 引脚上
COM6SEL:	COM6 输引脚选择位
	0 = COM6 功能位于 P3.1 引脚上
	1 = COM6 功能位于 P1.1 引脚上
COM5SEL:	COM5 输引脚选择位
	0 = COM5 功能位于 P3.2 引脚上
	1 = COM5 功能位于 P1.2 引脚上
COM4SEL:	COM4 输引脚选择位
	0 = COM4 功能位于 P3.3 引脚上
	1 = COM4 功能位于 P1.3 引脚上
COM3SEL:	COM3 输引脚选择位
	0 = COM3 功能位于 P3.4 引脚上
	1 = COM3 功能位于 P1.4 引脚上
COM2SEL:	COM2 输引脚选择位
	0 = COM2 功能位于 P3.5 引脚上
	1 = COM2 功能位于 P1.5 引脚上
COM1SEL:	COM1 输引脚选择位
	0 = COM1 功能位于 P3.6 引脚上
	1 = COM1 功能位于 P1.6 引脚上
COM0SEL:	COM0 输引脚选择位
	0 = COM0 功能位于 P3.7 引脚上
	1 = COM0 功能位于 P1.7 引脚上

2.8.1.12 引脚复用功能控制寄存器 APFCTLB

寄存器: APFCTLB: 引脚复用功能控制寄存器1(地址: 26BH)

复位值	bit7					bit0		
	-	-	-	-	-	CCP5SEL2	CCP5SEL1	CCP5SEL0
0000 0000	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

CCP5SEL<2:0>: CCP5 功能引脚选择位

- 000 = CCP5 捕捉比较功能位于 P1.2 引脚上
- 001 = CCP5 捕捉比较功能位于 P2.0 引脚上
- 010 = CCP5 捕捉比较功能位于 P2.3 引脚上
- 011 = CCP5 捕捉比较功能位于 P3.5 引脚上
- 100 = CCP5 捕捉比较功能位于 P1.1 引脚上
- 101 = CCP5 捕捉比较功能位于 P5.7 引脚上
- 110 = CCP5 捕捉比较功能位于 P4.4 引脚上
- 111 = CCP5 捕捉比较功能位于 P5.0 引脚上

3 存储器

KF8F513X 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 32K 字节 (16K×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 512×8 位，通用数据寄存器空间为(2048+16)×8 位。另外 KF8F513X 中还有一些其它存储器，包括：寄存器组 R0~R7、16 级硬件堆栈、ID 地址单元等。

3.1 程序存储器(ROM)区

KF8F513X 有一个 14 位的程序计数器，最大可寻址 16K×16 位的程序存储空间，而在 KF8F513X 中只实现了 16K×16 位的程序存储空间，地址为 0000H~3FFFH，复位向量入口地址为 0000H，中断向量有两级入口地址，高为 0004H，低为 0014H。图 3.1 为程序存储器区的地址映射图。

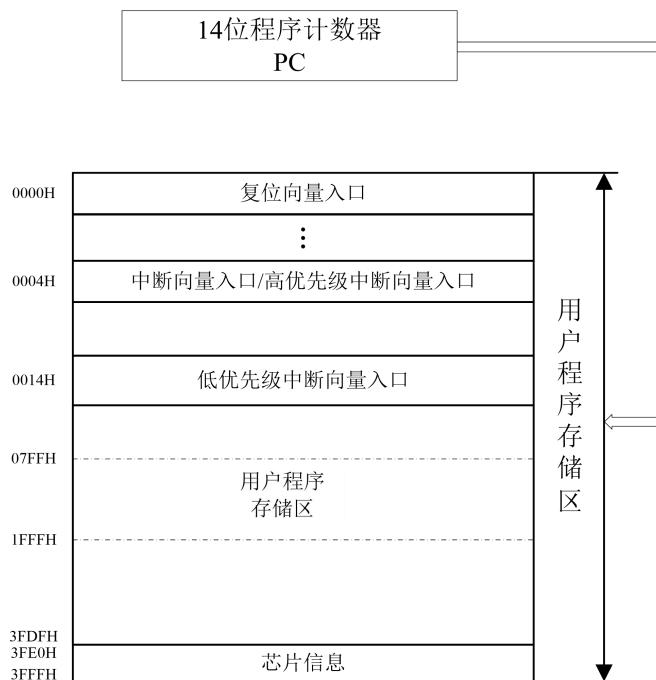


图 3.1 KF8F513X 程序存储器映射

3.1.1 程序计数器(PC)

KF8F513X 的程序计数器(PC)为 14 位宽。程序计数器(PC)的低 8 位(PC<7:0>)来自特殊功能寄存器 PCL，高 6 位(PC<13:8>)来自 PCH 寄存器，PC 不能直接读写，PCH/PCL 寄存器可以读写。当发生复位时，PC 将被清零。在有任何未屏蔽中断发生后 PC 值将指向 0004H 或 0014H 地址。

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。

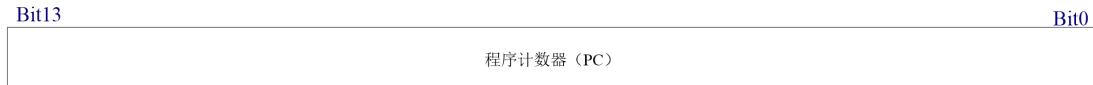


图 3.2 程序计数器 (PC)

3.1.1.1 PCL/PCH 寄存器的使用

当需要改变程序计数器到固定地址（包括跳页）时，可以通过写 PCH 寄存器确定高 6 位地址（写 PCH 寄存器不会改变 PC 的值），当写低 8 位地址数据到 PCL 寄存器时，程序计数器的 14 位地址数据将更新，变为 PCH 寄存器和 PCL 寄存器的数据，如图 3.4 所示。

对 PCL/PCH 寄存器的写操作可以通过 MOV 指令来执行，对 PCH 寄存器的写操作同样可以通过专用写指令 MOVP 来实现（只需 1 条指令即可完成 PCH 的赋值）。

例如当程序计数器从 0000H 开始执行跳转程序到 1F55H 地址，过程如下：

例 3-1 程序计数器地址跳转指令执行示例

PC	指令	备注
0000	MOVP #0X1F	将 1FH 写入 PCH 寄存器
0001	MOV R0, #0X55	将 55H 赋给 R0
0002	MOV PCL, R0	将 R0 内数据写入 PCL, PC 内容变为 PCH/PCL 寄存器的值
1F55

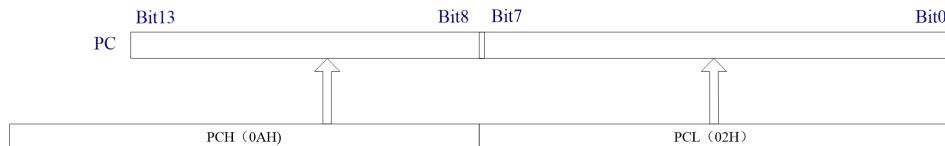


图 3.3 PCL/PCH 寄存器的使用

3.1.1.2 执行 JMP、CALL 指令时的情况

KF8F513X 单片机的 JMP、CALL 指令编码如下：

表 3-1 JMP、CALL 指令编码

JMP #data12	1100_kkkk_kkkk_kkkk
CALL #data12	1101_kkkk_kkkk_kkkk

在执行 JMP 或者 CALL 指令时，程序计数器 (PC) 的值将变为 PCH_BIT5 和 PCH_BIT4 以及指令所带立即数 (#data12)，如图 3.4 所示。

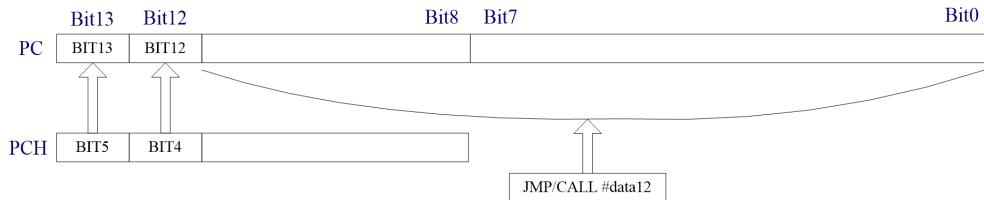


图 3.4 执行 JMP/CALL 指令时 PC 的变化图

执行 JMP 指令时将更新 PC；而执行 CALL 指令在更新 PC 的同时，将 CALL 指令的下一条地址入栈，栈地址加 1；在执行 RETURN 指令（IRET、RRET、CRET）时，将之前入栈的地址数据出栈并更新到 PC，栈地址减 1，PCH 寄存器不受出栈入栈的影响。

3.1.2 堆栈

KF8F513X 单片机具有 16 级的硬件堆栈。堆栈空间单独编址，不占用任何程序存储区和数据存储区的空间，堆栈指针不可读写。当执行 CALL 指令或者中断导致程序跳转时，PC 值将被压入（PUSH）堆栈。在执行返回指令（IRET、RRET、CRET）时，堆栈中的断点地址将从堆栈中弹出（POP）到 PC 中。PCH 不受 PUSH 或者 POP 操作的影响。

KF8F513X 单片机具有 16 级的硬件堆栈，堆栈压栈第 17 次的数值将覆盖第 1 次压栈的所保存的值，堆栈压栈第 18 次的数值将覆盖第 2 次压栈的所保存的值，以此类推。

3.2 数据存储器(RAM)区

如图 3.5 所示，KF8F513X 中的数据存储器由 10 个区组成，每个区的空间都是 128 字节，其中 2 个区用作特殊功能寄存器区(SFR)使用；另外 8 个存储器区为通用寄存器区，由用户支配。SFR 地址空间为 00H~6FH、100H~16FH、200H~26FH 和 300H~36FH；而 70H~7FH 有 16 个字节为 SRAM 共用区，即当用户访问其他 BANK 区 70H~7FH 的存储单元时，均是对 BANK 0 区的 70H~7FH 操作。

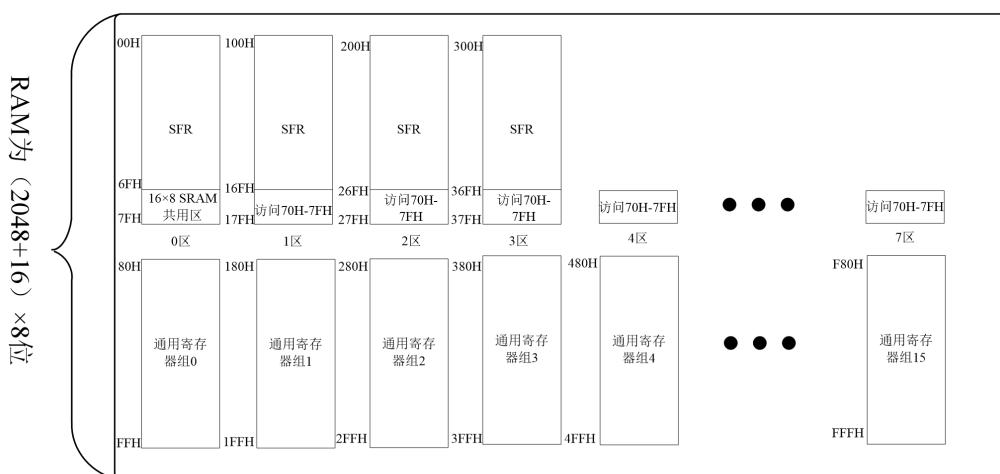


图 3.5 数据存储器地址映射图

3.2.1 通用寄存器区

如图 3.5 所示, 通用寄存器的空间为 2048 字节, 0 区至 15 区通过 BANK 寄存器中的 PR3~PR0 位进行选择, 如表 3-2 所示。

寄存器3.1: BANK: 通用寄存器选区寄存器(地址: 0x17H)

复位值	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
---- 0000	-	-	-	-	PR3	PR2	PR1	PR0

U U U U R/W R/W R/W R/W

表 3-2 通用寄存区地址

PR<3:0>	通用寄存器区	地址
0000	通用寄存器 0 区	80H~FFH
0001	通用寄存器 1 区	180H~1FFH
0010	通用寄存器 2 区	280H~2FFH
0011	通用寄存器 3 区	380H~3FFH
0100	通用寄存器 4 区	480H~4FFH
0101	通用寄存器 5 区	580H~5FFH
0110	通用寄存器 6 区	680H~6FFH
0111	通用寄存器 7 区	780H~7FFH
1000	通用寄存器 8 区	880H~8FFH
1001	通用寄存器 9 区	980H~9FFH
1010	通用寄存器 10 区	A80H~AFFH
1011	通用寄存器 11 区	B80H~BFFH
1100	通用寄存器 12 区	C80H~CFFH
1101	通用寄存器 13 区	D80H~DFFH
1110	通用寄存器 14 区	E80H~EFFH
1111	通用寄存器 15 区	F80H~FFFH

切换通用寄存区的指令例程如下:

例 3-2 通用寄存器区切换样例程序

- 1.单指令切换BANK寄存器存储区


```
MOVB #0X01          ; 切换到存储区1区
      MOVB #0X02          ; 切换到存储区2区
```
- 2.直接对BANK寄存器进行操作


```
MOV R0, #0X03
      MOV BANK, R0          ; 切换到存储区3区
```

3.2.2 特殊功能寄存器(SFR)区

KF8F513X 内部的电源系统、I/O 口控制、定时/计数器、PWM、运放、中断等各种外设的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。

3.2.3 状态字寄存器 (PSW)

状态字寄存器(PSW): 如寄存器 3.2 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)。 $\overline{\text{TO}}$ 和 $\overline{\text{PD}}$ 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。

寄存器3.2: PSW: 状态字寄存器(地址: 03H)

bit7	-	-	-	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	CY	bit0
	U	U	U	R/W	R/W	R/W	R/W		

$\overline{\text{TO}}$: 超时标志位

1 = 在上电复位、CWDT 指令或 IDLE 指令执行之后

0 = WDT 超时被清 0

$\overline{\text{PD}}$: 上电复位标志位

1 = 上电复位或执行 CWDT 指令后

0 = 执行 IDLE 指令后被清 0

Z: 零状态标志位

1 = 算术运算或者逻辑运算的运行结果为 0

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

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)

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

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)

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 DATA EEPROM

KF8F513X 片内的 DATA EEPROM 存储器最大容量为 256×8 位，地址范围 00H~FFH，在 CPU 正常工作期间是可读写的。DATA EEPROM 是单独编址，可以通过特殊功能寄存器寻址。

表 3-3 与 DATA 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
3BH	NVMADDR								NVM 地址指针低 8 位寄存器
3CH	NVMCTL0								NVM 控制寄存器 0
3DH	NVMCTL1								NVM 控制寄存器 1
39H	NVMDATAL								NVM 数据低 8 位寄存器

DATA EEPROM 数据存储器只能以字节为单位进行读写。当对字节写操作时会自动擦除目标存储单元（无论有没有数据）并写入新数据（在写入前擦除）。

当器件被代码保护时，器件编程器将不再能访问 DATA EEPROM。在代码保护时，CPU 仍可读写 DATA EEPROM 存储器。

3.3.1 寄存器 NVMDATAL

使用 DATA EEPROM 时，寄存器 NVMDATAL 用来存放要写入或者读出 DATA EEPROM 的数据。

3.3.2 寄存器 NVMADDR

DATA EEPROM 最大容量为 256×8 位，地址范围 0~255。NVMADDR 寄存器用来存放要写入 DATA EEPROM 的 8 位的地址信息。

寄存器 3.4 NVMADDR: 数据指针低 8 位(地址: 3BH)							
复位值	bit7	bit6	bit5	bit4	bit3	bit2	bit0
0000 0000	R/W						

3.3.1 寄存器 NVMCTL0/NVMCTL1

NVMCTL0/NVMCTL1 为写 DATA 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 DATA EEPROM 时，将 NVMDATAL 中送入要写入的数据，NVMADDR 中送入要写入的地址，然后通过向 NVMCTL0 和 NVMCTL1 送入固定的写命令，将数据写入 DATA EEPROM 对应地址指向的单元中。在读 DATA EEPROM 时，将要读的地址送到 NVMADDR 中，然后向 NVMCTL0 写入固定的读命令，把要读的数据送到 NVMDATAL 中。

3.3.2 写 DATA EEPROM

写 DATA EEPROM 时, 一次最多写入一个地址, 写入操作之前自动附加一个擦除操作, 擦除目标存储单元, 然后写入新的数据。

写 DATA EEPROM 时, 将要写入的地址送到 NVMADDR, 将要写入的数据送到 NVMDATAL。之后通过执行以下操作完成写操作:

例 3-3 DATA EEPROM 写操作样例程序

```
MOV B #0X00
CLR INTCTL, 7
JNB INTCTL, 7
JMP $-2
MOV R0, #0X04          ;使能DATA EEPROM的写操作
MOV NVMCTL0, R0
MOV R0, #0X69
MOV NVMCTL1, R0
MOV R0, #0X96
MOV NVMCTL1, R0
SET NVMCTL0, 1         ;执行写操作
NOP
NOP
MOV R0, #0X00          ;关闭DATA EEPROM的写, 防止意外写
MOV NVMCTL0, R0
SET INTCTL, 7
```

以上指令中的立即数 0X04, 0X00, 0X69, 0X96 是固定不变的。如果未完全按照上述顺序 (先将 0X69 写入 NVMCTL1, 再将 0X96 写入 NVMCTL1, 最后置位 NVMCTL0.1) 执行指令, 将不会启动写操作。写周期完成时, EE 写完成中断标志位 (EEIF) 置 1, 用户可以允许此中断或查询此位。EEIF 必须用软件清零。

注:

- 1.CPU写DATA EEPROM时,不管DATAP设置为何值, 都能写入正确的数据;
- 2.写DATA EEPROM的工作温度范围为-40°C~105°C。

写 DATA EEPROM 的步骤:

1. 将要写入的数据送到 NVMDATAL;
2. 将对应的 DATA EEPROM 地址送到 NVMADDR;
3. 执行上面的写命令, 此时, CPU 发出擦除 DATA EEPROM 目标存储单元的命令, 擦除完毕后, 将 NVMDATAL 中的数据送到对应地址中。DATA EEPROM 需要 6ms 来执行擦除和写入数据的命令, 此期间 CPU 仍可正常工作, 用户可通过中断标志位 EEIF 判断写 DEE 是否完成, 期间指令可正常执行;
4. 重复执行步骤 1、2、3, 可以执行其他地址的写入。

3.3.3 读 DATA EEPROM

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

例 3-4 读 DATA EEPROM 样例程序

```
MOV R0, #0Xxx           ; (将目标地址写入NVMADDR)
MOV NVMADDR, R0
MOV R0, #0X01
MOV NVMCTL0, R0
NOP
MOV R0, NVMDATA          ;R0=NVMDATA
```

上面指令中的立即数 0X01 是固定不变的。此时，该地址的数据被送到 NVMDATA。

读 DATA EEPROM 是逐字读取的。读 DATA EEPROM 时通过向 NVMCTL0 写入 0X01 来执行读命令。NVMDATA 寄存器保存数据直到下一次读命令覆盖当前值。

注：读DATA EEPROM时，不管DATAP设置为何值，都能读出正确的数据

读 DATA EEPROM 的步骤如下：

1. 将要读的数据单元的地址送到 NVMADDR 中；
2. 向 NVMCTL0 写入读命令；
3. 一个指令周期后该单元的数据被送到 NVMDATA。

3.4 寄存器组 Rn

KF8F513X 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；隐含目的操作数的指令中，默认 R0 作为目的操作数(如: RRCR 0X81)；在读晶振校准值和参考电压校准值时，默认将读到的值送到 R0 中。

3.5 ID 地址单元

KF8F513X 的程序存储器空间的最后 32 个地址单元被指定为 ID 地址单元，地址为 3FE0H~3FFFH。用户可在其中存放校准值或其它信息。

4 汇编指令及寻址方式

4.1 寻址方式

KF8F513X 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。KF8F513X 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 NVMADDR。

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

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

KF8F513X 系列单片机汇编指令共有 73 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

KF8F513X 单片机的中断源有：

- INT0 中断
- P0 口引脚电平变化中断
- T0 溢出中断
- INT1/2 中断
- P4 口引脚电平变化中断
- T1/2/3/4 中断
- CCP3/CCP4 中断
- CCP5(捕捉/比较/PWM5)中断
- PWM2 中断
- ADC 中断
- SPI 中断
- I2C 中断
- SSCI 模块 I2C 总线冲突(BCL)中断
- 通用串行通讯接口 USART1/2 的发送和接收中断
- 外部时钟故障(OSCFAIL)中断
- DATA EEPROM 写操作(EE)中断

在本单片机中有多个中断优先级，其中高优先级向量位于 0X0004H，低优先级向量位于 0X0014H。在中断服务程序里可通过 PCTL 的 IPEN(PCTL<3>)进行中断优先级设置。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。

KF8F513X 中的 INT1/2、T1、T2、T3、T4、ADC、PWM1x、CMP、LED、LCD、CT、HLVD、SPI、I2C、BCL、CCP、USART 和 EE 等都属于外设，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部中断。中断逻辑如图 5.2 所示。在中断逻辑框图中，每个中断源有 3 个位用于控制其操作。这些位的功能分别是：

☒ 标志位表明发生了中断事件；

☒ 中断允许位允许程序跳转到中断向量地址处执行；

☒ 中断优先级位用于选择高优先级及还是低优先级；

通过将 IPEN 位 (PCTL<3>) 置 1，可使能中断优先级功能。

当 IPEN 置 1 时，有两个中断允许位，分别是 AIEH 和 AIEL。将 AIEH (INTCTL<7>) 置 1，可允许所有中断优先级位已置 1 的中断，即高优先级的中断。将 AIEL (INTCTL<6>) 置 1，可允许所有中断优先级位已清 0 的中断，即低优先级的中断。当中断标志位、中断允许位和中断优先级位都被置 1 时，中断将根据设置的中断优先级立即跳转到地址 0x0004H 或者 0x0014H。进低优先级中断时清 0 AIEL，退出中断时置 1 AIEL；进高优先级中断时清 0 AIEH，退出中断时置 1 AIEH。(AIEH=0 时禁止所有中断)。高优先级中断事件可以中断正在处理的低优先级中断事件。

其原理框图如图 5.1 下:

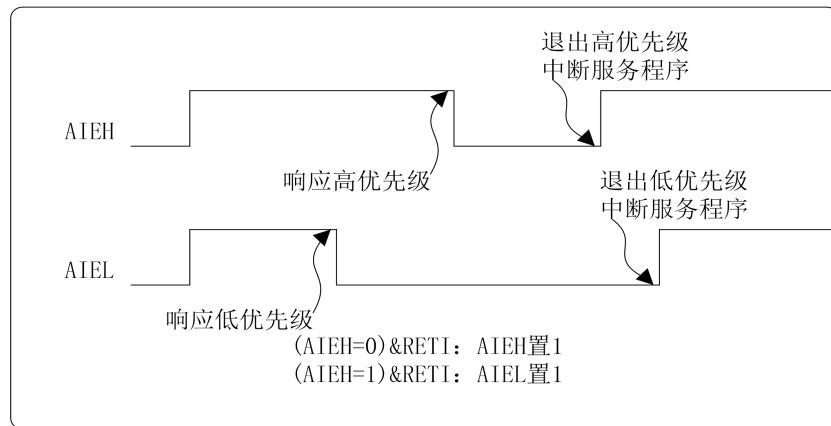


图 5.1 中断优先级工作原理图

当 IPEN 位清 0 时, 就会禁止中断优先级, 即为普通模式。所有中断都跳转到 0x0004H 开始执行。在普通模式下, 没有中断优先级, 各个中断源的中断优先级控制寄存器 IPx 均无效。AIE(INTCTL<7>)为全局中断使能位, PUIE(INTCTL<6>)为外设中断使能位。

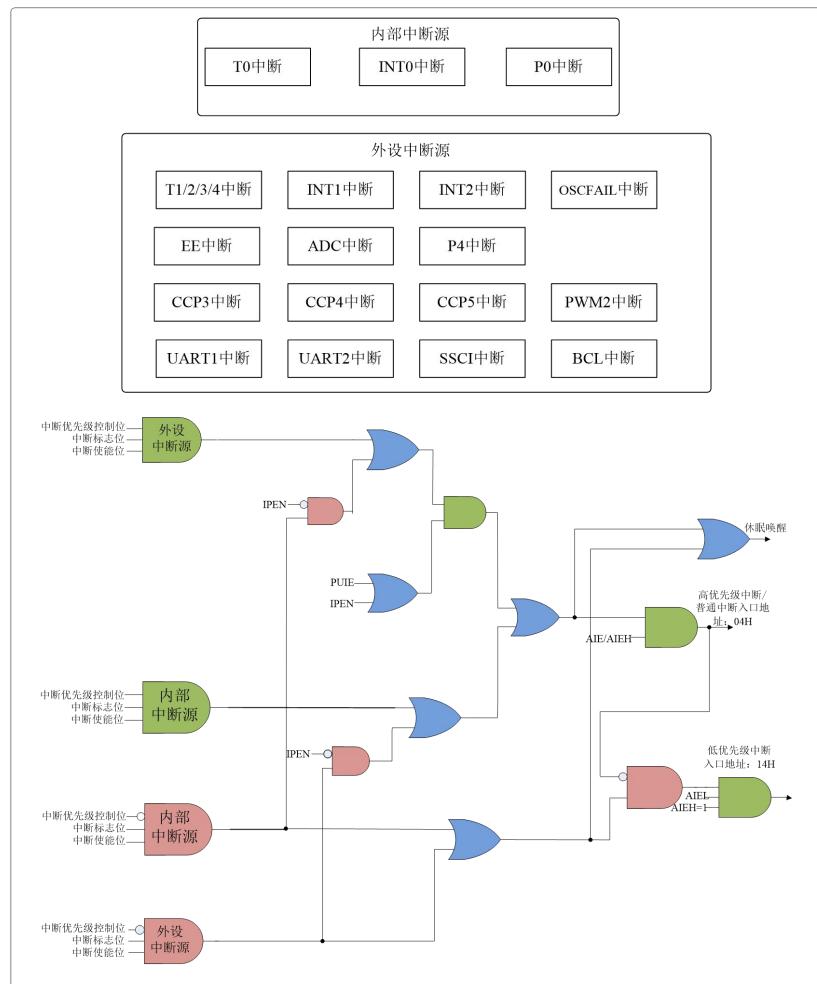


图 5.2 中断逻辑

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/ AIEH	PUIE/ AIEL	T0IE	INT0IE	POIE	T0IF	INT0IF	POIF
2CH	EIE1	EEIE	ADIE	INT2IE	INT1IE	-	PWM2IE	T2IE	PWM1IE/T1 IE
2DH	EIE2	T3IE	-	RC1IE	TX1IE	-	CCP5IE	BCLIE	SSCIIE
4AH	EIE3	T4IE	-	OSCFAILIE	CCP4IE	CCP3IE	-	P4IE	-
251H	EIE4	-	-	-	-	RE2IE	TE2IE	RC2IE	TX2IE
0CH	EIF1	EEIF	ADIF	INT2IF	INT1IF	-	PWM2IF	T2IF	PWM1IF/T1 IF
0DH	EIF2	T3IF	-	RC1IF	TX1IF	-	CCP5IF	BCLIF	SSCIIF
4BH	EIF3	T4IF	-	OSCFAILIF	CCP4IF	CCP3IF	-	P4IF	-
252H	EIF4	-	-	-	-	RE2IF	TE2IF	RC2IF	TX2IF
22H	IP0	-	-	-	-	-	PT0	PINT0	PP0
23H	IP1	PEE	PADC	PINT2	PINT1	-	PPWM2	PT2	PPWM1/ PT1
24H	IP2	PT3	-	PRC1	PTX1	-	PCCP5	PBCL	PSSCI
29H	IP3	PT4	-	POSCFAIL	PCCP4	PCCP3	-	PP4	-
253H	IP4	-	-	-	-	PRE2	PTE2	PRC2	PTX2
2EH	PCTL	-	-	-	SLVREN	IPEN	SWDTEN	<u>POR</u>	<u>LVR</u>
67H	INTEDGCTL	INT2SE	INT1SE	-	-	-	-	-	T1CLK SE

5.1.1 中断控制寄存器 INTCTL

普通模式下，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。

在优先级中断中，AIEH 为全局优先级中断使能位，当其被清 0 时，禁止所有中断。AIEL 为低优先级中断使能位，当其被清 0 时禁止所有低优先级中断。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。
3. AIEL和PUIE是两个地址相同但物理上分开的寄存器，AIEL只有在IPEN=1时才可写，PUIE只有在IPEN=0时才可写；使用时在配置IPEN位之后，再对PUIE（或AIEL）位赋值。

寄存器5.1: INTCTL: 中断控制寄存器(地址: 0BH)

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

AIE/AIEH: 全局中断使能位/优先级中断使能位

当 IPEN=0

1=使能所有未屏蔽的中断

0=禁止所有中断

当 IPEN=1

1=允许所有高优先级的中断

0=禁止所有中断

PUIE/AIEL: 外设中断使能位/低优先级中断使能位

当 IPEN=0

1=使能所有未屏蔽的外设中断

0=禁止所有外设中断

当 IPEN=1

1=允许所有低优先级的外设中断

0=禁止所有低优先级的外设中断

TOIE: T0 溢出中断使能位

1=使能 T0 中断

0=禁止 T0 中断

INT0IE: INT0 中断使能位

1=使能 INT0 中断

0=禁止 INT0 中断

POIE: P0 口电平变化中断使能位

1=使能 P0 口电平变化中断

0=禁止 P0 口电平变化中断

TOIF: T0 溢出中断标志位

1=T0 寄存器溢出

0=T0 寄存器未溢出

INT0IF: INT0 中断标志位

1=INT0/P0.0 产生外部中断

0=INT0/P0.0 未产生外部中断

POIF: P0 口电平变化中断标志位

1=引脚 P0.0~P0.5 至少有一个电平状态发生变化

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

寄存器5.2: EIE1: 中断使能寄存器(地址: 2CH)

复位值 0000 0000	bit7							bit0 PWM1IE/ T1IE
	EEIE	ADIE	INT2IE	INT1IE	-	PWM2IE	T2IE	
	R/W	R/W	R/W	R/W	-	R/W	R/W	R/W

- EEIE: EE 中断使能位
 1 = 使能独立 EE 中断
 0 = 禁止独立 EE 中断
- ADIE: AD 中断使能位
 1 = 使能 AD 中断
 0 = 禁止 AD 中断
- INT2IE: INT2 中断使能位
 1 = 使能 INT2 中断
 0 = 禁止 INT2 中断
- INT1IE: INT1 中断使能位
 1 = 使能 INT1 中断
 0 = 禁止 INT1 中断
- PWM2IE: PWM2 中断使能位
 1 = 使能 PWM2 中断
 0 = 禁止 PWM2 中断
- T2IE: T2 与 PP5 匹配中断允许位
 1 = 允许 T2 与 PP5 匹配中断
 0 = 禁止 T2 与 PP5 匹配中断
- PWM1IE/T1IE: PWM1/T1 中断使能位
 1 = 使能 PWM1/T1 中断
 0 = 禁止 PWM1/T1 中断

5.1.3 中断使能寄存器 EIE2

寄存器5.3: EIE2: 中断使能寄存器(地址: 2DH)

复位值	bit7	T3IE	-	RC1IE	TX1IE	-	CCP5IE	BCLIE	SSCIE	bit0
0000 0000		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

T3IE: T3 中断使能位

1 = 允许 T3 中断
0 = 禁止 T3 中断

RC1IE: USART1 接收中断使能位

1 = 允许 RX1 中断
0 = 禁止 RX1 中断

TX1IE: USART1 发送中断使能位

1 = 允许 TX1 中断
0 = 禁止 TX1 中断

CCP5IE: CCP5 中断使能位

1 = 允许 CCP5 中断
0 = 禁止 CCP5 中断

BCLIE: BCL 中断使能位

1 = 允许 BCL 中断
0 = 禁止 BCL 中断

SSCIE: SSCI 中断使能位

1 = 允许 SSCI 中断
0 = 禁止 SSCI 中断

5.1.4 中断使能寄存器 EIE3

寄存器**5.4:** EIE3: 中断使能寄存器(地址: 4AH)

复位值 0000 0000	bit7							bit0
	T4IE	-	OSCFAILIE	CCP4IE	CCP3IE	-	P4IE	-
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

T4IE: T4 中断使能位

1 = 允许 T4 中断

0 = 禁止 T4 中断

OSCFAILIE: 外部时钟故障中断使能位

1 = 允许外部时钟故障中断

0 = 禁止外部时钟故障中断

CCP4IE: P4 口电平变化中断使能位

1 = 允许 P4 口电平变化中断

0 = 禁止 P4 口电平变化中断

CCP3IE: P3 口电平变化中断使能位

1 = 允许 P3 口电平变化中断

0 = 禁止 P3 口电平变化中断

P4IE: P4 口电平变化中断使能位

1 = 允许 P4 口电平变化中断

0 = 禁止 P4 口电平变化中断

5.1.5 中断使能寄存器 EIE4

寄存器**5.4:** EIE4: 中断使能寄存器4(地址:251H)

复位值 ---- 0000	bit7							bit0
	-	-	-	-	RE2IE	TE2IE	RC2IE	TX2IE
	U	U	U	U	R/W	R/W	R/W	R/W

RE2IE: UART2 接收错误中断使能位

1 = 使能 UART2 接收错误中断

0 = 禁止 UART2 接收错误中断

TE2IE: UART2 发送错误中断使能位

1 = 使能 UART2 发送错误中断

0 = 禁止 UART2 发送错误中断

RC2IE: USART2 接收中断使能位

1 = 允许 RX2 中断

0 = 禁止 RX2 中断

TX2IE: USART2 发送中断使能位

1 = 允许 TX2 中断

0 = 禁止 TX2 中断

5.1.6 中断标志寄存器 EIF1

寄存器5.5: EIF1: 外设中断标志寄存器(地址0CH)

复位值 0000 0000	bit7								bit0
	EEIF	ADIF	INT2IF	INT1IF	-	PWM2IF	T2IF	PWM1IF/ T1IF	
	R/W	R/W	R/W	R/W	-	R/W	R/W	R/W	

- EEIF: EE 中断标志位
 1 = 独立 EE 完成写操作
 0 = 独立 EE 未完成写操作
- ADIF: AD 完成中断标志位
 1 = AD 转换完成
 0 = AD 转换没有完成
- INT2IF: INT2 中断标志位
 1 = INT2 产生外部中断
 0 = INT2 未产生外部中断
- INT1IF: INT1 中断标志位
 1 = INT1 产生外部中断
 0 = INT1 未产生外部中断
- PWM2IF: PWM2 中断标志位
 1 = 发生了 T2H/T2L 与 PP5H/PP5L 匹配
 0 = 未发生 T2H/T2L 与 PP5H/PP5L 匹配
- T2IF: T2 与 PP5 匹配中断标志位
 1 = 发生了 T2H/T2L 与 PP5H/PP5L 匹配
 0 = 未发生了 T2H/T2L 与 PP5H/PP5L 匹配
- PWM1IF/T1IF: PWM1/T1 寄存器溢出标志位
 1 = T1 寄存器溢出
 0 = T1 寄存器未溢出

5.1.7 中断标志寄存器 EIF2

寄存器5.6: EIF2: 外设中断标志寄存器(地址: 0DH)

复位值 0000 0000	bit7				bit0			
	T3IF	-	RC1IF	TX1IF	-	CCP5IF	BCLIF	SSCIIF
	R/W	R/W	R	R	R/W	R/W	R/W	R/W

T3IF: T3 中断标志位

1 = T3 产生了中断

0 = T3 未产生中断

RC1IF: USART1 接收中断标志位

1 = USART1 接收缓冲器满 (通过 RXSDR1 清 0)

0 = USART1 接收缓冲器空

TX1IF: USART1 发送中断标志位

1 = USART1 发送缓冲器满

0 = USART1 发送缓冲器空 (通过 TXSDR1 清 0)

CCP5IF: CCP5 中断标志位

1 = CCP5 产生了中断

0 = CCP5 未产生中断

BCLIF: BCL 中断标志位

1 = BCL 产生了中断

0 = BCL 未产生中断

SSCIIF: SSCI 中断标志位

1 = SSCI 产生了中断

0 = SSCI 未产生中断

5.1.8 中断标志寄存器 EIF3

寄存器**5.7:** EIF3: 中断标志寄存器(地址: 4BH)

复位值 0000 0000	bit7								bit0
	T4IF	-	OSCFAILI F	CCP4IF	CCP3IF	-	P4IF	-	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

T4IF: T4 中断标志位

1 = T4 溢出或发生重载

0 = T4 不溢出或发生重载

OSCFAILIF: 外部时钟故障中断标志位

1 = 外部时钟发生故障

0 = 外部时钟未发生故障

CCP4IF: CCP4 中断标志位

1 = CCP4 发生捕捉或比较事件

0 = CCP4 未发生捕捉和比较事件

CCP3IF: CCP3 中断标志位

1 = CCP3 发生捕捉或比较事件

0 = CCP3 未发生捕捉和比较事件

P4IF: P4 口电平变化中断标志位

1=引脚 P4.0~P4.7 至少有一个电平状态发生变化

0=引脚 P4.0~P4.7 电平状态未发生变化

5.1.9 中断标志寄存器 EIF4

寄存器**5.8:** EIF4: 中断使能寄存器4(地址: 252H)

复位值 ---- 0000	bit7				bit0			
	-	-	-	-	RE2IF	TE2IF	RC2IF	TX2IF
	U	U	U	U	R/W	R/W	R	R

RE2IF: UART2 接收错误中断标志位

1 = 发生 UART2 接收错误中断

0 = 未发生 UART2 接收错误中断

TE2IF: UART2 发送错误中断标志位

1 = 发生 UART2 发送错误中断

0 = 未发生 UART2 发送错误中断

RC2IF: USART2 接收中断标志位

1 = USART2 接收缓冲器满 (通过 RXSDR2 清 0)

0 = USART1 接收缓冲器空

TX2IF: USART2 发送中断标志位

0 = USART2 发送缓冲器满

1 = USART2 发送缓冲器空 (通过 TXSDR2 清 0)

5.1.10 中断优先级控制寄存器 IP0

当 IPEN=1 时，中断优先级控制寄存器 IP0 才有效。

寄存器5.9:IP0: 中断优先级控制寄存器0(地址: 22H)

复位值 ---- -000	bit7						bit0	
	-	-	-	-	-	PT0	PINT0	PP0
	U	U	U	U	U	R/W	R/W	R/W

- PT0: T0 中断优先级控制位
 1 = T0 中断为高优先级
 0 = T0 中断为低优先级
- PINT0: INT0 中断优先级控制位
 1 = INT0 中断为高优先级
 0 = INT0 中断为低优先级
- PP0: P0 电平变化中断优先级控制位
 1 = P0 电平变化中断为高优先级
 0 = P0 电平变化中断为低优先级

5.1.11 中断优先级控制寄存器 IP1

当 IPEN=1 时，中断优先级控制寄存器 IP1 才有效。

寄存器5.10:IP1: 中断优先级控制寄存器1(地址: 23H)

复位值 0000 0000	bit7							
	PEE	PADC	PINT2	PINT1	-	PPWM2	PT2	PPWM1/ PT1
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- PEE: 独立 EE 中断高优先级控制位
 1 = 独立 EE 中断为高优先级
 0 = 独立 EE 中断为低优先级
- PADC: ADC 中断高优先级控制位
 1 = ADC 中断为高优先级
 0 = ADC 中断为低优先级
- PINT2: INT2 中断高优先级控制位
 1 = INT2 中断为高优先级
 0 = INT2 中断为低优先级
- PINT1: INT1 中断高优先级控制位
 1 = INT1 中断为高优先级
 0 = INT1 中断为低优先级
- PPWM2: PWM2 中断高优先级控制位
 1 = PWM2 中断为高优先级
 0 = PWM2 中断为低优先级
- PT2: T2 中断高优先级控制位

1 = T2 中断为高优先级

0 = T2 中断为低优先级

PPWM1/PT1: PWM1/T1 中断高优先级控制位

1 = PWM1/T1 中断为高优先级

0 = PWM1/T1 中断为低优先级

5.1.12 中断优先级控制寄存器 IP2

当 IPEN=1 时, 中断优先级控制寄存器 IP2 才有效。

寄存器5.11: IP2: 中断优先级控制寄存器2(地址: 24H)

复位值	bit7								bit0
	PT3	-	PRC1	PTX1	-	PCCP5	PBCL	PSSCI	
0000 0000	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

PT3: T3 中断高优先级控制位

1 = T3 中断为高优先级

0 = T3 中断为低优先级

PRC1: RX1 中断高优先级控制位

1 = RX1 中断为高优先级

0 = RX1 中断为低优先级

PTX1: TX1 中断高优先级控制位

1 = TX1 中断为高优先级

0 = TX1 中断为低优先级

PCCP5: CCP5 中断高优先级控制位

1 = CCP5 中断为高优先级

0 = CCP5 中断为低优先级

PBCL: BCL 中断高优先级控制位

1 = BCL 中断为高优先级

0 = BCL 中断为低优先级

PSSCI: SSCI 中断高优先级控制位

1 = SSCI 中断为高优先级

0 = SSCI 中断为低优先级

5.1.13 中断优先级控制寄存器 IP3

当 IPEN=1 时, 中断优先级控制寄存器 IP3 才有效。

寄存器5.12: IP3: 中断优先级控制寄存器3(地址: 29H)

复位值	bit7								bit0
	PT4	-	POSCFAI_L	PCCP4	PCCP3	-	PP4	-	
0000 0000	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

PT4: T4 中断高优先级控制位

1 = T4 中断为高优先级

0 = T4 中断为低优先级

POSCFAIL: OSCFAIL 中断高优先级控制位

1 = OSCFAIL 中断为高优先级

0 = OSCFAIL 中断为低优先级

P CCP4: CCP4 中断高优先级控制位

1 = CCP4 中断为高优先级

0 = CCP4 中断为低优先级

P CCP3: CCP3 中断高优先级控制位

1 = CCP3 中断为高优先级

0 = CCP3 中断为低优先级

PP4: P4 口电平变化中断高优先级控制位

1 = P4 电平变化中断为高优先级

0 = P4 电平变化中断为低优先级

5.1.14 中断优先级控制寄存器 IP4

寄存器5.13: IP4: 中断优先级控制寄存器4(地址: 253H)

复位值 ---- 0000	bit7				bit0			
	-	-	-	-	PRE2	PTE2	PRC2	PTX2
	U	U	U	U	R/W	R/W	R/W	R/W

PRE2: UART2 接收错误中断优先级控制位

1 = UART2 接收错误中断为高优先级

0 = UART2 接收错误中断为低优先级

PTE2: UART2 发送错误中断优先级控制位

1 = UART2 发送错误中断为高优先级

0 = UART2 发送错误中断为低优先级

PRC2: RX2 中断高优先级控制位

1 = RX2 中断为高优先级

0 = RX2 中断为低优先级

PTX2: TX2 中断高优先级控制位

1 = TX2 中断为高优先级

0 = TX2 中断为低优先级

5.1.15 电源控制寄存器 PCTL

PCTL 寄存器的 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 LVR =0，则表示发生过欠压检测复位。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)，LVR 状态位是不可预知的。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5.14:PCTL: 电源控制寄存器(地址:2EH)

复位值 --01 00xx	bit7							bit0
	-	-	-	SLVREN	IPEN	SWDTEN	<u>POR</u>	<u>LVR</u>
	U	U	R/W	R/W	R/W	R/W	R/W	R/W

SLVREN: 软件欠压检测使能位

1 = 使能欠压检测

0 = 禁止欠压检测

IPEN: 中断优先级控制位

1 = 使能中断优先级功能

0 = 禁止中断优先级，即为普通模式

SWDTEN: 软件看门狗定时器使能位

当配置字的 WDTEN=0 时

1 = 软件使能看门狗定时器

0 = 软件禁止看门狗定时器

POR : 上电复位状态位

1 = 未发生上电复位

0 = 发生了上电复位

LVR : 欠压复位状态位

1 = 未发生欠压复位

0 = 已发生欠压复位

5.1.16 INT 中断沿选择寄存器 INTEDGCTL

寄存器5.15:INTEDGCTL:INT中断沿选择寄存器 (67H)

复位值 00----1	bit7							bit0
	INT2SE	INT1SE	-	-	-	-	-	T1CLKSE
	R/W	R/W	U	U	U	U	U	-

INT2SE: INT2 触发脉冲边沿选择位

1 = 上升沿触发

0 = 下降沿触发

INT1SE: INT1 触发脉冲边沿选择位

1 = 上升沿触发

0 = 下降沿触发

T1CLKSE: T1 定时器时钟选择位

1 = 内部高频时钟
0 = 系统时钟 SCLK/4

5.1.17 中断响应

当 IPEN=1 时, 中断被响应后:

1. 进入优先级中断, 由软件对 AIEH 或 AIEL 位都置 1。
2. 返回地址压入堆栈;
3. 中断入口地址载入 PC;
4. AIEL 位由硬件自动清 0, 执行低优先级中断服务程序, 【如果进入高优先级中断时, AIEH 由硬件自动清 0。执行高优先级中断服务程序, 执行指令 IRET 退出高优先级中断硬件自动置 1 AIEH, 继续执行低优先级中断服务程序】, 执行指令 IRET 退出低优先级中断硬件自动置 1 AIEL。
5. 只置 1 AIEH, 则直接进入高优先级中断, AIEH 由硬件自动清 0;
6. 继续执行中断服务子程序;
7. 执行指令 IRET 退出中断服务子程序并由硬件自动将 AIEH 置 1, 重新使能未屏蔽的中断;
8. 跳转到中断发生处继续执行下面的程序。

当 IPEN=0 时, 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。

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 设置为下降沿触发, 清零设置为上升沿触发。INTCTL 中的 INT0IF 为 INT0 的中断标志位。如果 IPEN 和 PINT0 位均置 1, 则 INT0 为高优先级中断。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 INTEDGCTL 中的 INT1SE 位设置触发边沿, INT1SE 置 1, 将 INT1 设置为上升沿触发, 清零设置为下降沿触发。EIF1 中的 INT1IF 为 INT1 的中断标志位。如果 IPEN 和 PINT1 位均置 1, 则 INT1 为高优先级中断。

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 INTEDGCTL 中的 INT2SE 位设置触发边沿, INT2SE 置 1, 将 INT2 设置为上升沿触发, 清零设置为下降沿触发。EIF1 中的 INT2IF 为 INT2 的中断标志位。如果 IPEN 和 PINT2 位均置 1, 则 INT2 为高优先级中断。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), 如果为高优先级, 则 IPEN 和 PINTx 均置 1。

注:x=0/1/2

5.3 定时器中断

定时/计数器产生中断的条件如下:

T0: 计数溢出;

T1: 计数溢出或者与 PP2/PP1 寄存器匹配;

T2: 计数值与 PP5H/PP5L 寄存器匹配;

T3: 计数溢出或者与 T3REH/T3REL 寄存器匹配;

T4: 计数溢出或者与 T4REH/T4REL 寄存器匹配;

有关定时/计数器模块中断的操作, 请参考定时/计数器部分。

5.4 P0/P4 口中断

P0/P4 口引脚的输入电平变化将使 P0IF/P4IF 位置 1。通过设置/清除 P0IE/P4IE 位, 可使能/禁止该中断。且该端口各引脚可通过 IOCL0/IOCL4 寄存器来对每个引脚进行配置。

当 IPEN 和 PP0/PP4 均置 1 时, P0/P4 口中断可以配置为高优先级中断。



有关 P0/P4 口的操作，请参考 P0/P4 口部分。

5.5 CCP3/4/5 中断

KF8F513X 共有 3 个 CCP 模块，各自中断相关寄存器位如下表所示：

表 5-2 CCP 中断相关寄存器位对应表

模块	中断标志位	中断使能位	中断高优先级配置位
CCP3	EIF3 CCP3IF	EIF3 CCP3IE	IP3_PCCP3
CCP4	EIF3 CCP4IF	EIF3 CCP4IE	IP3_PCCP4
CCP5	EIF2 CCP5IF	EIF2 CCP5IE	IP2_PCCP5

以 CCP5 为例：

捕捉模式下，当一个捕捉发生时，中断请求标志位 EIF2 寄存器中的 CCP5IF 置 1。

比较模式下，所有比较模式可以产生中断，当 PWM5L1：PWM5L0 与 T1H 和 T1L 匹配时，中断请求标志位 EIF2 寄存器中的 CCP5IF 置 1。

如果使能中断 (AIE=1, PUIE=1, CCP5IE=1)，则会响应中断请求；如果使能中断优先级功能 (IPEN=1)，则可通过 PCCP5 位配置 CCP5 中断的高低优先级。

5.6 PWM1/2/3/5 中断

KF8F513X 共有 4 个 PWM 模块，PWM 中断与对应定时器中断共用，即 PWM1 和 PWM2 使用 T1 中断相关位 (T1IF、T1IE 和 PT1)，PWM3 使用 T3 中断相关位 (T3IF、T3IE 和 PT3)，PWM5 使用 T2 中断相关位 (T2IF、T2IE 和 PT2)；其中 PWM2 还有一组独立的中断相关寄存器位 (PWM2IF, PWM2IE 和 PPWM2)，当使能 PWM2 时，其作用和 T2 中断相关位 (T2IF、T2IE 和 PT2) 一致。

以 PWM2 模块为例：

使能 PWM2 后，对应定时器 T1 分配给 PWM 进行计数，当定时器溢出时，会触发相应的定时器中断标志位 T1IF；如果使能中断 (AIE=1, PUIE=1, T1IF=1)，则会响应中断请求；如果使能中断优先级功能 (IPEN=1)，则可通过 PT1 位配置 T1 中断的高低优先级。详见 PWM 部分。

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

KF8F513X 单片机提供：

- 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：带重载功能、门控和预分频器的 16 位定时器/计数器
- 定时器 2：带 16 位周期寄存器、预分频器和后分频器 16 位定时器
- 定时器 3：带有重载功能、CCP、预分频器及时钟源多选的 16 位定时器/计数器
- 定时器 4：带有重载功能、CCP、预分频器及时钟源多选的 16 位定时器/计数器

6.1 定时/计数器 T0

T0 是一个 8 位的定时器/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。

6.1.1 T0 原理框图

图 6.1 为 T0 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。

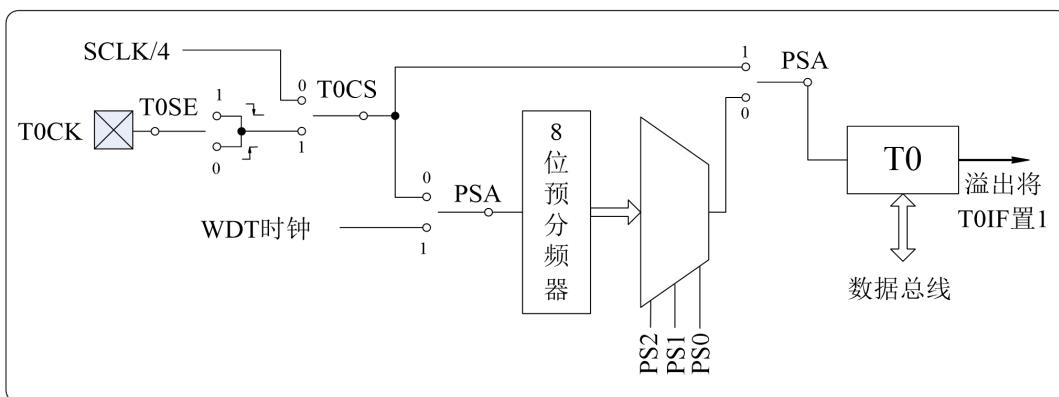


图 6.1 T0 原理框图

6.1.2 T0 引脚复用

T0CK 引脚可以通过 APFCTL8 寄存器的 T0CKSEL<1:0>位进行复用配置；T0CK 输入功能可以位于 P0.3、P0.6、P3.3 或 P4.2 口上。

表 6-1 T0CK 引脚复用功能对照表

T0CKSEL<1:0>	T0CK 所在引脚
--------------	-----------

00	P0.3
01	P0.6
10	P3.3
11	P4.2

6.1.3 T0 相关的寄存器

表 6-2 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
01H	T0					8 位计数器			
21H	OPTR	PUPH	INT0SE	T0CS	T0SE	PSA	PS2	PS1	PS0
268H	APFCTL8	-	TICKSEL ₂	TICKS _{EL1}	TICKSE _{L0}	T1GSE _{L1}	T1GSEL0	T0CKSE _{L1}	T0CKSE _{L0}

6.1.3.1 OPTR 选择寄存器

寄存器6.1: OPTR: 选择寄存器(地址: 21H)

复位值	bit7	PUPH	INT0SE	T0CS	T0SE	PSA	PS2	PS1	bit0 PS0
1111 1111		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 I/O 口上拉功能总使能位

1 = 禁止所有 I/O 口的上拉功能

0 = 允许 I/O 口使用上拉功能

INT0SE INT0 中断触发脉冲边沿选择位

1 = INT0/P0.0 为下降沿触发

0 = INT0/P0.0 为上升沿触发

T0CS: T0 模式选择位

1 = 计数模式, T0 的时钟为外部时钟 T0CK/P0.0

0 = 定时模式, T0 的时钟为内部机器时钟 SCLK/4 (系统时钟 4 分频)

T0SE: T0 计数脉冲信号边沿选择位

1 = 下降沿触发

0 = 上升沿触发

PSA: 预分频器分配控制位

1 = 预分频器用于 WDT

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

6.1.4 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。

6.1.5 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,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,以实现 T0CK 与内部相位时钟的同步。

6.1.6 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 T1

T1 是一个 16 位的定时器/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。

6.2.1 T1 原理框图

T1 的原理框图如图 6.2 所示，T1 是一个带有门控和预分频的 16 位定时器/计数器，计数时钟可选择外部时钟或者内部时钟，当 T1 与内部时钟一起使用时，T1 用作定时器，当 T1 与外部时钟一起使用时，T1 工作在计数器模式，通过对 $\overline{\text{T1SY}}$ 位($\text{T1CTL}<2>$)设置可使 T1 工作在异步计数器模式。

该 T1 模块还带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($\text{T1CTL}<5:4>$)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。

此外，T1 还带有重载功能，重载寄存器利用 PP2/PP1 设置。当使能 T1 重载功能时，T1 计数器计数到 T1 重载寄存器中设置的值时，T1 计数器将清 0 重新开始计数，且将 T1 中断标志位置 1。

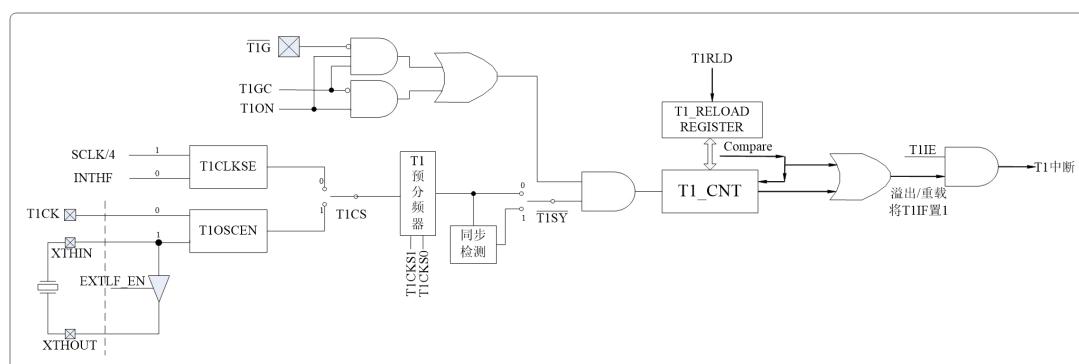


图 6.2 T1 原理框图

6.2.2 T1 引脚复用

T1CK 和 T1G 引脚可以分别通过 APFCTL8 寄存器的 T1CKSEL<2:0>位和 T1GSEL<1:0>位进行复用配置。

表 6-3 T1CK 和 T1G 引脚复用功能对照表

T1CKSEL<2:0>	T1CK 所在引脚	T1GSEL<1:0>	T1G 所在引脚
000	P0.4	00	P0.4
001	P0.5	01	P0.5
010	P1.1	10	P4.5
011	P3.6		
100	P4.2		

6.2.3 T1 相关的寄存器

表 6-4 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
0EH	T1L	定时/计数器 T1 低字节寄存器							
0FH	T1H	定时/计数器 T1 高字节寄存器							
10H	T1CTL	T1RLD	T1GC	T1CKS1	T1CKS0	T1OSCEN	$\overline{T1SY}$	T1CS	T1ON
67H	INTEDGCTL	INT2SE	INTISE	-	-	-	-	-	T1CLKSE
268H	APFCTL8	BUZ3SEL	T1CKSEL2	T1CKSEL1	T1CKSEL0	T1GSEL1	T1GSEL0	T0CKSEL1	T0CKSEL0

6.2.3.1 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)

复位值	bit7								bit0
	T1RLD	T1GC	T1CKS1	T1CKS0	T1OSCEN	$\overline{T1SY}$	T1CS	T1ON	
0000 0000	R/W	R/W	R/W	R/W	U	R/W	R/W	R/W	

- T1RLD: T1 重载功能使能位
0 = 禁止 T1 重载功能
1 = 使能 T1 重载功能
- T1GC: T1 门控使能位
如果 T1ON=0 则该位被忽略
如果 T1ON=1 则:
1 = 使能 $\overline{T1G}$ 引脚控制: $\overline{T1G}$ 引脚为低电平, 启动 T1, 为高电平, 关闭 T1
0 = 禁止 $\overline{T1G}$ 引脚控制
- T1CKS<1:0>: T1 输入时钟预分频比选择位
11 = 1/8 倍预分频比
10 = 1/4 倍预分频比
01 = 1/2 倍预分频比
00 = 1/1 倍预分频比
- T1OSCEN: T1 外部低频时钟使能位
1 = 使能外部低频时钟源作为 T1 计数时钟
0 = 禁止外部低频时钟源作为 T1 计数时钟
- $\overline{T1SY}$: T1 计数模式外部触发脉冲输入同步控制位
- T1CS=1:
1 = 外部触发脉冲输入不与系统时钟同步
0 = 外部触发脉冲输入与系统时钟同步
- T1CS=0: 该位被忽略, T1 使用内部时钟
- T1CS: T1 定时/计数模式选择

- 1 = 计数模式, T1 时钟为外部时钟
 当 T1OSCEN=1 时 T1 时钟为外部低频时钟
 当 T1OSCEN=0 时 T1 时钟为 T1CK
- 0 = 定时模式
 当 T1CLKSE=1 时 T1 时钟为内部高频时钟
 当 T1CLKSE=0 时 T1 时钟为指令时钟 (系统时钟四分频)
- T1ON: T1 启动控制位
 1 = 启动 T1
 0 = 停止 T1

6.2.3.2 INT 中断沿选择寄存器

寄存器6.4: INTEDGCTL:INT中断沿选择寄存器 (67H)

复位值	bit7	INT2SE	INT1SE	-	-	-	-	bit0
00----1		R/W	R/W	U	U	U	U	T1CLKSE R/W -

- INT2SE: INT2 触发脉冲边沿选择位
 1 = 上升沿触发
 0 = 下降沿触发
- INT1SE: INT1 触发脉冲边沿选择位
 1 = 上升沿触发
 0 = 下降沿触发
- T1CLKSE: T1 定时器时钟选择位
 1 = 内部高频时钟
 0 = 系统时钟 SCLK/4

6.2.4 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。

6.2.5 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。

6.2.6 T1 重载功能

置位 T1CTL 寄存器的 T1RLD 位使能 T1 重载功能。通过设置 PP2/PP1 寄存器来设置 T1 的重载点。当设置完 PP2/PP1 寄存器且使能重载功能后，T1 计数器从 0 开始计数，当计数到 PP2/PP1 寄存器设置的值时，T1 计数器清 0 并重新开始计数，T1IF 置 1。

T1RLD 位置 1 时会载入一次 PP2/PP1 寄存器的值，当置位 T1RLD 且在 T1 计数过程中对 PP2/PP1 寄存器写入新值时，T1 模块会在下一次重载时载入新的重载点。

注：T1 使能重载前应该配置好 PP2/PP1，在 T1 计数中使能重载可能会直接进入中断。

6.2.7 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。

6.2.8 T1 分配给 PWM1

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

6.3 定时器 T2

T2 是一个带 16 位周期寄存器、预分频器和后分频器 16 位定时器，由 2 个 8 位寄存器 T2H 和 T2L 组成。没有外部计数时钟输入脚。

6.3.1 T2 的工作原理

如图 6.3 所示，为 T2 的原理框图，T2 的计数时钟为内部高频振荡器时钟 (INTHF)，每个周期 (当分频器 1 分频比为 1:1 时) 寄存器 T2 的值自动加 1。通过 T2CTL0 的 T2ON 启动/禁止 T2。

内部高频振荡器时钟 (INTHF) 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 T2CTL0 中的 T2CKPS<1:0>位，可将 INTHF 进行 1:1、1:4 和 1:16 分频。

当 T2H/L 寄存器与 PP5H/L 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 T2H/L 的值与寄存器 T2CCRxH/L 相比较，如果等于 T2CCRxH/L，且对应的控制位 T2CCRxON 置 1，则会发出信号，使 ADCCTL0 的 START (ADCCTL0<1>) 位置 1。

分频器 2 的分频比可通过寄存器 T2CTL0 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 T2H/L 与 PP5H/L 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 T2H/L 与 PP5H/L 相等累计两次才会使 T2IF 置 1，以此类推。

如果对 T2 寄存器执行写操作、对寄存器 T2CTL0 执行写操作或发生任何器件复位事件 (上电复位、 \overline{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 T2H/L、PP5 和 T2CCRxH/L 均可读写。任何复位时，寄存器 T2H/L 均被设置为 00H，T2CCRxH/L 被设置为 00H/00H，PP5 被设置为 FFH。

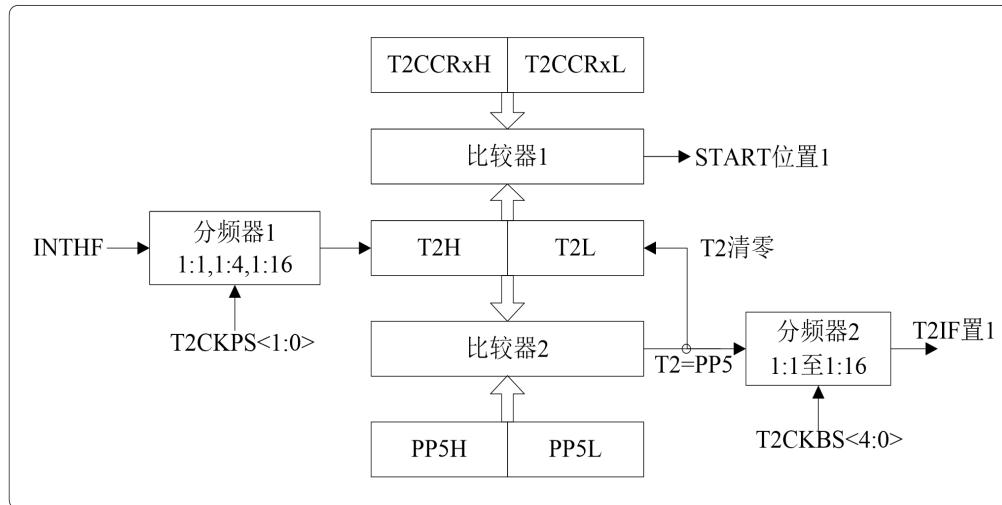


图 6.3 定时器 T2 原理框图

注：需要在 T2ON 位置 1 的前提下对 T2 计数寄存器 T2H/T2L 进行赋值操作，否则无法写入。

6.3.2 T2 相关的寄存器

表 6-5 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
12H	T2CTL0	-	T2CKBS3	T2CKBS2	T2CKBS1	T2CKBS0	T2ON	T2CKPS1	T2CKPS0
10FH	T2CTL1	T2MOD1	T2MOD0	T2OFS1	T2OFS0	T2TRIGEN	-	-	T2DIR
41H	T2H	T2 高字节寄存器							
11H	T2L	T2 低字节寄存器							
42H	PP5H	PWM5 周期寄存器高 8 位							
52H	PP5L	PWM5 周期寄存器低 8 位							
54H	T2CCR0L	T2 触发 ADC 启动设置寄存器 0 低 8 位							
40H	T2CCR0H	T2 触发 ADC 启动设置寄存器 0 高 8 位							
116H	T2CCR1L	T2 触发 ADC 启动设置寄存器 1 低 8 位							
118H	T2CCR1H	T2 触发 ADC 启动设置寄存器 1 高 8 位							

6.3.2.1 T2 控制寄存器 T2CTL0

寄存器6.3: T2CTL0: T2控制寄存器1(地址: 12H)

复位值	bit7								bit0
	-	T2CKBS3	T2CKBS2	T2CKBS1	T2CKBS0	T2ON	T2CKPS1	T2CKPS0	
-000 0000	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

T2CKBS<3:0>: T2 分频器 2 (后分频) 分频比选择位

- 0000 = 分频比为 1:1
- 0001 = 分频比为 1:2
- 0010 = 分频比为 1:3
- 0011 = 分频比为 1:4
- 0100 = 分频比为 1:5
- 0101 = 分频比为 1:6
- 0110 = 分频比为 1:7
- 0111 = 分频比为 1:8
- 1000 = 分频比为 1:9
- 1001 = 分频比为 1:10
- 1010 = 分频比为 1:11
- 1011 = 分频比为 1:12
- 1100 = 分频比为 1:13
- 1101 = 分频比为 1:14
- 1110 = 分频比为 1:15
- 1111 = 分频比为 1:16

T2ON: T2 使能位

1 = 使能 T2

0 = 禁止 T2

T2CKPS<1:0>: T2 分频器 1 (预分频) 分频比选择位

00 = 分频比为 1:1

01 = 分频比为 1:4

1x = 分频比为 1:16

6.3.2.2 T2 控制寄存器 T2CTL1

寄存器6.4: T2CTL1: T2控制寄存器1(地址:10FH)

复位值	bit7	T2MOD1	T2MOD0	T2OFS1	T2OFS0	T2TRIGEN	-	-	bit0	T2DIR
0000 0--0		R/W	R/W	R/W	R/W	R/W	U	U		R

T2MOD<1:0>: T2 计数方式选择位。

00 = 向上计数, 上溢时产生中断标志

01 = 向上-向下计数, 上溢时产生中断标志

10 = 向上-向下计数, 下溢时产生中断标志

11 = 向上-向下计数, 上溢和下溢时都产生中断标志

T2OFS1: T2 溢出中断选择, T2TRIGEN 为 1 时有效

0 = 禁止上溢中断触发 AD

1 = 使能上溢中断触发 AD

T2OFS0: T2 溢出中断选择, T2TRIGEN 为 1 时有效

0 = 禁止下溢中断触发 AD

1 = 使能下溢中断触发 AD

T2TRIGEN: T2 溢出中断自动触发 AD 控制位, 当 AD 转换结束后自动清零。

0 = 禁止 T2 溢出中断自动触发 AD

1 = 使能 T2 溢出中断自动触发 AD

T2DIR: T2 计数方向标志位 (只读)

0 = 当前计数方向为向下计数

1 = 当前计数方向为向上计数

6.3.2.3 T2CCR1/0H 和 T2CCR1/0L 寄存器

T2CCR1/0H 和 T2CCR1/0L 寄存器是 T2 触发 ADC 启动设置寄存器。通过 T2CCR_xON (x=0、1) 位可以控制相应的 T2CCR_xH/L (x=0、1) 是否工作。当 T2CCR_xON (x=0、1) 使能时, 系统不停的将 T2H/L 的值与寄存器 T2CCR_xH/L (x=0、1) 相比较, 如果等于 T2CCR_xH/L (x=0、1), 且对应的控制位置 1(见 7.1.1 和 7.1.4 寄存器), 则会发出信号, 使 ADCCTL0 的 START (ADCCTL0<1>) 位置 1, 使能 ADC 采样转换。

6.3.3 T2 中断

如图 6.3 所示, 预分频器 2 的计算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, 将会响应 T2 中断。

如果 IPEN 和 PT2 位均置 1, 则 T2 为高优先级中断。

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 PWM5

PWM5 在使用时，需要使用 T2 进行定时，PWM5 的详细内容请参考 PWM5 部分。

6.4 定时/计数器 T3

T3 是一个 16 位的定时/计数器，T3 的低 8 位在寄存器 T3L 中，高 8 位在寄存器 T3H 中，当 T3 计数值达到 65535 后，T3 的值再加 1 就会产生溢出，将 T3 中断标志位置 1。

T3 带有重载功能。当使能 T3 重载功能时，T3 计数器计数到 T3 重载寄存器中设置的值时，T3 计数器将清 0 重新开始计数，且将 T3 中断标志位置 1。

T3 属于外部单元，因此在使用 T3 中断时，需将 PUIE 位置 1，使能外设中断。

6.4.1 T3 原理框图

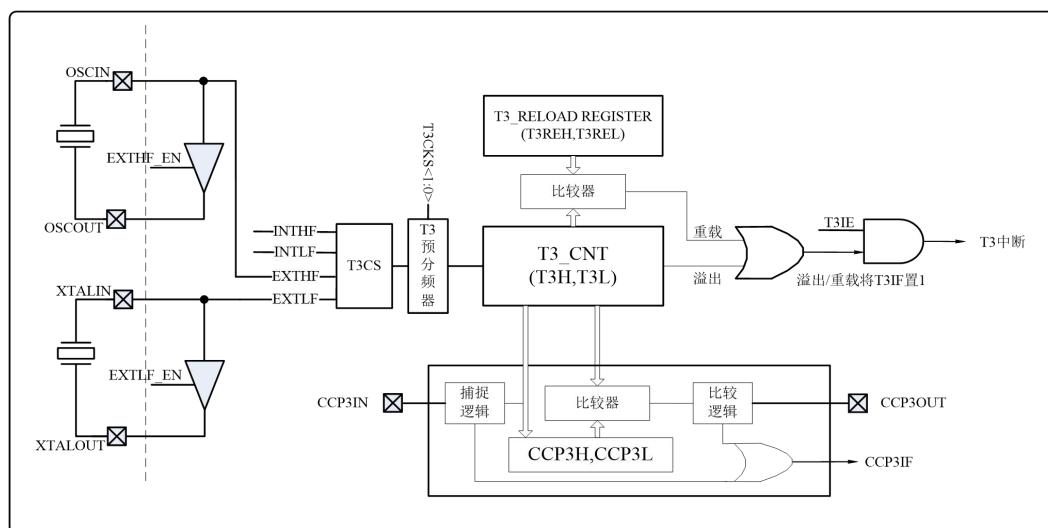


图 6.4 定时器 T3 原理框图

6.4.2 T3 相关寄存器

表 6-6 与 T3 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
4EH	T3CTL	T3REN	T3BUZ OE	T3CKS1	T3CKS0	-	T3CS1	T3CS0	T3ON
13H	CCP3CTL				CCP3OE	CCP3 MOD3	CCP3 MOD2	CCP3 MOD1	CCP3 MOD0
229H	PWM3CTL	-	-	-	-	PWM33 ON	PWM32 ON	PWM31 ON	PWM30 ON
4FH	T3L					T3 低位寄存器			
5FH	T3H					T3 高位寄存器			
15EH	T3REH					T3 重载设置寄存器高 8 位			
15DH	T3REL					T3 重载设置寄存器低 8 位			
14H	CCP3H					CCP3 寄存器高 8 位			
15H	CCP3L					CCP3 寄存器低 8 位			
22AH	PP3H					PWM3 周期设置寄存器高 8 位			
22BH	PP3L					PWM3 周期设置寄存器低 8 位			
22CH	PWM30H					PWM30 占空比高 8 位寄存器			
22DH	PWM31H					PWM31 占空比高 8 位寄存器			
22EH	PWM32H					PWM32 占空比高 8 位寄存器			
22FH	PWM33H					PWM33 占空比高 8 位寄存器			
230H	PWM30L					PWM30 占空比低 8 位寄存器			
231H	PWM31L					PWM31 占空比低 8 位寄存器			
232H	PWM32L					PWM32 占空比低 8 位寄存器			
233H	PWM33L					PWM33 占空比低 8 位寄存器			

6.4.2.1 T3 控制寄存器

如寄存器 6.5 所示, T3 控制寄存器 (T3CTL) 用于启动/禁止 T3 以及选择 T3 模块的不同功能特性。

寄存器6.5: T3CTL: T3控制寄存器(地址: 4EH)

复位值 0000 0000	bit7							bit0
	T3REN	T3BUZOE	T3CKS1	T3CKS0	-	T3CS1	T3CS0	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

T3REN: T3 重载功能使能位

- 0 = 禁止 T3 重载功能
- 1 = 使能 T3 重载功能

T3BUZOE: T3 蜂鸣器控制输出使能位

- 0 = 禁止 T3 蜂鸣器输出
- 1 = 使能 T3 蜂鸣器输出

T3CKS<1:0>: T3 输入时钟预分频比选择位

- 11 = 1/8 倍预分频比
- 10 = 1/4 倍预分频比
- 01 = 1/2 倍预分频比
- 00 = 1/1 倍预分频比

T3CS<1:0>: T3 定时/计数模式选择

- 00 = T3 时钟为内部高频时钟 INTHF
- 01 = T3 时钟为内部低频时钟 INTLF
- 10 = T3 时钟为外部高频时钟 EXTHF
- 11 = T3 时钟为外部低频时钟 EXTLF

T3ON: T3 启动控制位

- 1 = 启动 T3
- 0 = 停止 T3

6.4.2.2 CCP3 功能寄存器

寄存器6.7: CCP3CTL: CCP3控制寄存器(地址: 13H)

				bit7					bit0
复位值	-	-	-	CCP3OE	CCP3 MOD3	CCP3 MOD2	CCP3 MOD1	CCP3 MOD0	-
---0 0000	U	U	U	R/W	R/W	R/W	R/W	R/W	-

CCP3OE: CCP3比较模式输出使能位

1 = 比较模式下使能引脚的CCP3输出

0 = 禁止比较模式下引脚的CCP3输出

CCP3MOD<3:0>: CCP3模式选择位

0000 = 捕捉/比较 关闭 (复位CCP3模块)

0001 = 未使用 (保留)

0010 = 比较模式: 匹配时输出电平翻转 (T3IF置1)

0011 = 未使用 (保留)

0100 = 捕捉模式: 在每个下降沿发生捕捉

0101 = 捕捉模式: 在每个上升沿发生捕捉

0110 = 捕捉模式: 每4 个上升沿发生捕捉

0111 = 捕捉模式: 每16 个上升沿发生捕捉

1000 = 比较模式: 比较匹配时输出高电平 (T3IF置1)

1001 = 比较模式: 比较匹配时输出低电平 (T3IF置1)

1010 = 比较模式: 比较匹配时产生软件中断 (T3IF置1, CCP3OUT不受影响)

1011 = 比较模式: 触发特殊事件 (T3IF位置1, CCP3复位T3, 如果ADC使能
将启动AD转换)

1100 = 保留

1101 = 保留

1110 = 保留

1111 = 保留

6.4.3 T3 预分频器

如寄存器 6.5 所示, T3 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。T3CKS 位(T3CTL<5:4>)对预分频计数器进行控制。T3 预分频计数器不能直接进行读写操作, 可通过写入 T3H 或 T3L 使预分频计数器清 0。

6.4.4 T3 计数时钟选择

用户可以通过 T3CTL 寄存器的 T3CS<1:0>位来选择 T3 的计数时钟, KF8F513X 系列提供 4 个时钟源, 内部系统高频时钟、内部低频时钟、外部高频时钟和外部低频时钟。

6.4.5 T3 重载功能

置位 T3CTL 寄存器的 T3REN 位使能 T3 重载功能。通过设置 T3REH/T3REL 寄存器来设置 T3 的重载点。当设置 T3REH/T3REL 寄存器且使能重载功能后，T3 计数器从 0 开始计数，当计数到 T3REH/T3REL 寄存器设置的值时，T3 计数器清 0 并重新开始计数，T3IF 置 1。

T3REN 位置 1 时会载入一次 T3REH/T3REL 寄存器的值，当置位 T3REN 且在 T3 计数过程中对 T3REH/T3REL 寄存器写入新值时，T3 模块会在下一次重载时载入新的重载点。

6.4.6 T3 蜂鸣器输出功能

当使能重载功能时，可以通过 T3CTL 寄存器的 T3BUZOE 位使能蜂鸣器输出功能。蜂鸣器的输出引脚通过 APFCTL8 寄存器的 BUZ3SEL 位选择。

蜂鸣器的周期等于 2 倍重载设置值（T3REH/T3REL 寄存器设置值），占空比为 1:1。

6.4.7 T3 中断

在两种情况下会使 T3 中断标志位置 1：

1. 当 T3 计数值达到 65535 后，T3 的值再加 1 就会产生溢出，将 T3 中断标志位置 1；
2. 当 T3 重载时，T3 中断标志位置 1。

6.4.8 T3 工作在休眠模式

当选择低频时钟信号作为 T3 的计数时钟时，T3 模块可以工作在休眠模式下。当使能 T3 中断时，系统会在 T3 溢出或者重载时唤醒。

6.5 CCP3 (捕捉/比较/PWM 功能)

定时器 T3 带有 CCP 功能：捕捉模式、比较模式和 PWM 模式。

用户在使用时需注意，捕捉比较模式的中断相关位独立于 T3，捕捉模式和比较模式带有专门的中断标志位 CCP3IF、中断使能位 CCP3IE 和中断优先级配置位 PCCP3。

6.5.1 捕捉模式

在捕捉模式下，当对应的CCP3IN引脚发生事件时，寄存器CCP3H和CCP3L捕捉T3寄存器的16位值，原理框图如下图6.5所示：

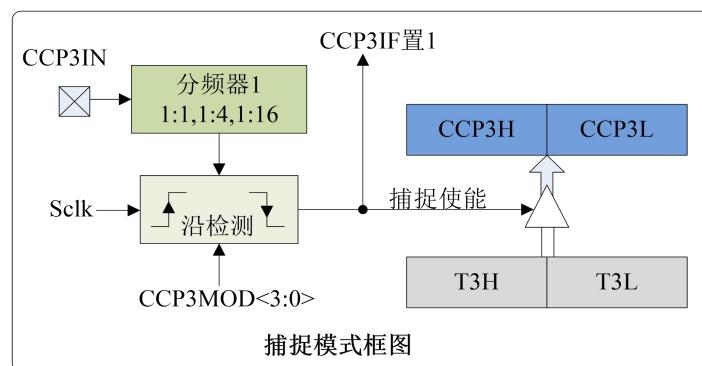


图 6.5 捕捉模式原理框图

触发捕捉的事件可被定义为以下四者之一，并且由 CCP3CTL 寄存器中的模式选择位 CCP3MOD<3:0>选择事件类型位配置：

- ◆ 0100 = 每个下降沿
- ◆ 0101 = 每个上升沿
- ◆ 0110 = 每4 个上升沿
- ◆ 0111 = 每16个上升沿

在捕捉模式下，应通过将相应的 CCP3 引脚配置为数字输入。

当一个捕捉发生时，硬件自动中断请求标志位 CCP3IF 置 1；它必须用软件清零。注意如果在 CCP3H、CCP3L 这对寄存器中的值被读取之前发生另一次捕捉，那么之前捕捉的值将被新捕捉的值覆盖。

当捕捉模式改变时，可能会产生错误的捕捉中断。用户应该在捕捉模式改变之前保持 CCP3IE 中断允许位清零以避免产生误中断。在捕捉模式发生任何改变之后也应清零中断标志位 CCP3IF。

CCP3CTL 寄存器中的 CCP3MOD<3:0>位指定了 4 种预分频器设置。每当关闭 CCP 模块或禁止捕捉模式时，就会清零预分频器计数器。这意味着任何复位都将清零预分频计数器。

虽然从一种捕捉预分频比切换到另一种捕捉预分频比不会将预分频计数器清零，但可能会产生误中断。因此要避免出现这种不期望的操作，应在改变预分频比前通过将 CCP3CTL 寄存器清零关闭该模块。

6.5.2 比较模式

在比较模式下，16位CCP3H、CCP3L寄存器的值将不断与T3寄存器的值相比较。当两者匹配时，CCP模块可能会出现以下几种情况：

- ◆ CCP3OUT引脚的输出电平翻转
- ◆ CCP3OUT引脚输出高电平
- ◆ CCP3OUT引脚输出低电平
- ◆ 产生软件中断触发信号
- ◆ 产生特殊事件触发信号

CCP3OUT引脚的动作取决于CCP3CTL寄存器中CCP3MOD<3:0>控制位的值。

所有比较模式都会产生中断。原理图如图6.6所示：

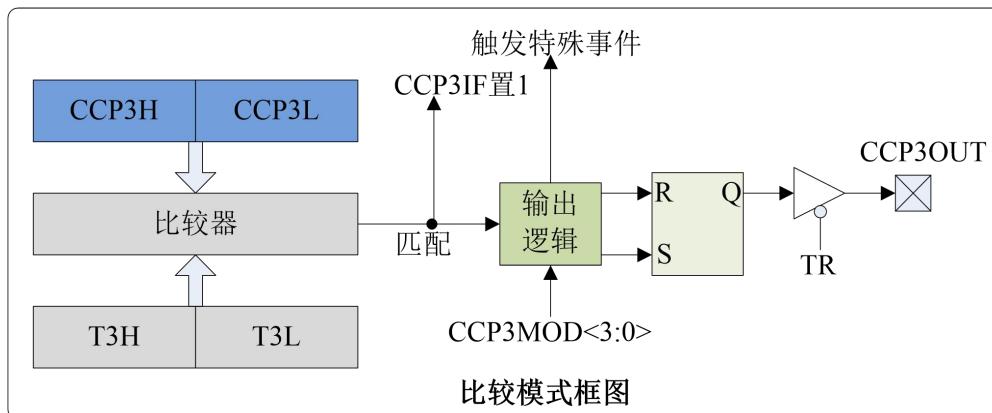


图 6.6 比较模式原理框图

用户必须通过将相应的CCP3OUT引脚配置为数字输出。

- 1) 当选择输出电平翻转模式时，比较匹配时，CCP3OUT引脚的输出电平翻转，并将CCP3IF置1。
- 2) 当选择普通比较模式时（1010或1001）时，比较匹配时，CCP3OUT引脚输出高电平或者低电平，并且将CCP3IF置1。
- 3) 当选择了软件中断触发模式时（1010）时，比较匹配时，将CCP3IF置1，但是CCP模块不会控制CCP3OUT引脚。
- 4) 当选择了特殊事件触发模式（1011）时，比较匹配时，CCP会立即产生特殊事件触发输出，将CCP3IF置1，此时如果ADC已使能，将启动AD转换。但T3计数寄存器不会立即复位，直到T3计数脉冲的下一个上升沿才复位。从而使CCP3H、CCP3L寄存器实际上成为了定时器T3的16位可编程周期寄存器。

6.5.3 PWM3 模式

PWM3 模式下，带有 4 路 16 位标准 PWM：PWM30、PWM31、PWM32 和 PWM33。4 路 PWM3 共用周期配置寄存器 PP3H 和 PP3L；占空比配置寄存器独立。下文以 PWM31 为例进行介绍，其他 PWM3x 输出的工作原理和 PWM31 模块完全一致。

启动 PWM31 后，在对应的 PWM31 引脚输出 PWM 脉冲。PWM 脉冲的频率和占空比通过<PP3H:PP3L>和<PWM31H:PWM31L>设置。

下图显示了 PWM31 逻辑框图。其中<PP3H:PP3L>为 PWM31 模块的周期寄存器，<PWM31H:PWM31L>为 PWM31 模块占空比设置寄存器，使用 PWM31 时需要将定时器 3 配置给 PWM11 做定时用，并且需要使能 T3 重载。

启动 PWM31 后，当<T3H:T3L>计数值和<PP3H:PP3L>相等时，PWM31 引脚被置 1，此时<T3H:T3L>被清 0，重新开始计数；当<T3H:T3L>的计数值和<PWM31H:PWM31L>相等时，PWM31 引脚清 0。

改变<PP3H:PP3L>和<PWM31H:PWM31L>的值可产生不同的 PWM31 周期和 PWM31 占空比。

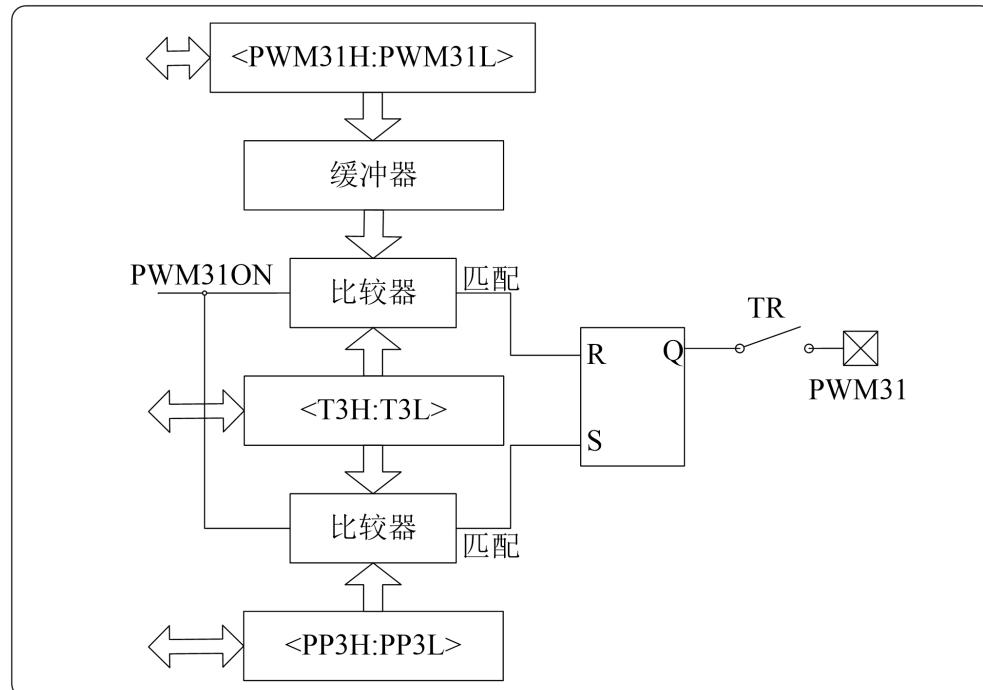


图 6.7 PWM31 逻辑框图

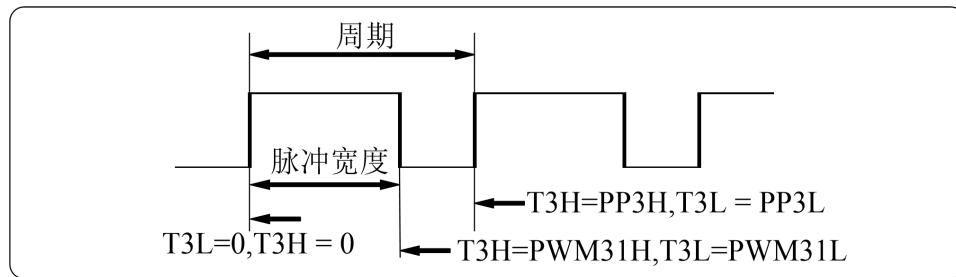


图 6.8 PWM31 输出波形图

注：T3H=PWM31H, T3L=PWM31L 时 PWM31 输出变为低电平；T3H=PP3H, T3L=PP3L 时

PWM31 输出变为高电平，同时将 T3H 和 T3L 清零。

6.5.3.1 PWM3 输出端口

表 6-7 PWM3 对应 IO 口

PWM3	对应 I/O 口	备注
PWM30	P0.4	通过 APFCTL3 寄存器相关位进行配置
PWM31	P5.0 / P4.3	
PWM32	P4.4 / P4.2	
PWM33	P1.2 / P4.0	

6.5.3.2 PWM3 相关的寄存器

表 6-8 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
229H	PWM3CTL	-	-	-	-	PWM33ON	PWM32ON	PWM31ON	PWM30ON
22AH	PP3H					PWM3 周期设置寄存器高 8 位			
22BH	PP3L					PWM3 周期设置寄存器低 8 位			
22CH	PWM30H					PWM30 占空比高 8 位寄存器			
22DH	PWM31H					PWM31 占空比高 8 位寄存器			
22EH	PWM32H					PWM32 占空比高 8 位寄存器			
22FH	PWM33H					PWM33 占空比高 8 位寄存器			
230H	PWM30L					PWM30 占空比低 8 位寄存器			
231H	PWM31L					PWM31 占空比低 8 位寄存器			
232H	PWM32L					PWM32 占空比低 8 位寄存器			
233H	PWM33L					PWM33 占空比低 8 位寄存器			

6.5.3.3 PWM3 启动控制寄存器

寄存器： PWM3CTL: PWM3启动控制寄存器(地址:229H)

复位值	bit7				bit0			
	0000 0000	R/W	R/W	R/W	R/W	R/W	R/W	R/W

PWM3xON: PWM3x 启动控制位

1 = 启动 PWM3x

0 = 禁止 PWM3x

注： x=0~3。

6.5.3.4 PWM3 周期

PWM3 周期通过<PP3H: PP3L>进行设置，其值可设置为 0~65535。PWM 周期通过如下公式进行计算。T3 计数时钟周期由 T3 工作时钟源和预分频比共同决定。

式 6.1 PWM3 周期计算公式

$$\text{PWM3周期} = (\text{PP3H:PP3L} + 1) \cdot \text{Tt3} \text{, 其中} \text{ Tt3} = \text{T3计数时钟周期}$$

6.5.3.5 PWM3 占空比

PWM3x 占空比通过<PWM3xH: PWM3xL>设置。脉冲宽度和占空比通过如下公式计算。T3 计数时钟周期由 T3 工作时钟源和预分频比共同决定。

式 6.2 PWM3 脉冲宽度计算公式

$$\text{脉冲宽度} = <\text{PWM3xH: PWM3xL}> \cdot \text{Tt3}$$

其中: $x=0\sim3$; $\text{Tt3}=T3$ 计数时钟周期

式 6.3 PWM3 占空比计算公式

$$\text{PWM3 占空比} = \frac{\text{脉冲宽度}}{\text{PWM3 周期}} = \frac{<\text{PWM3xH: PWM3xL}>}{<\text{PP3H: PP3L}> + 1} \quad (x=0\sim3)$$

6.5.4 PWM3 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。KF8F513X 中当 PP3H 和 PP3L 均为 255 时, PWM 的最大分辨率为 16 位。分辨率的计算公式如下所示。

式 6.4 PWM3 分辨率计算公式

$$\text{分辨率} = \frac{\log[<\text{PP3H: PP3L}> + 1]}{\log 2} \text{ 位}$$

6.5.5 PWM3 中断

PWM3 和 T3 共用中断相关寄存器位: 中断使能位 T3IE、中断标志位 T3IF 和高优先级控制位 PT3。

在 PWM3 启用后, 当 T3H/T3L 的计数值与 PWM3xH/PWM3xL 的值匹配后, 其对应的输出引脚变为低电平。当 T3H/T3L 的计数值与 PP3H/PP3L 的值匹配后, 其对应的输出引脚变为高电平, 同时将 T3H/T3L 清 0, 将 T3IF 置 1, 如果允许 T3 中断, 将会转入对应的中断子程序中。

6.5.6 休眠模式下的操作

在休眠模式下, PWM3x 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, T3 将从原来的状态继续工作。

6.5.7 系统时钟频率的改变

PWM3x 的输出频率是通过 T3 定时产生的，T3 的时钟源改变、T3 所选时钟源的频率发生任何改变都会使 PWM 频率发生变化。

6.5.8 复位的影响

任何复位都会将所有端口强制为输入模式，强制 PWM3 使用的寄存器进入其复位状态。

6.5.9 PWM3 使用方法

PWM3 工作的设置应按照以下步骤：

- 1、将引脚 PWM3x 所对应的方向控制位 TRxx 置 1，禁止引脚 PWM3x 的输出驱动器。
- 2、赋<PP3H: PP3L>寄存器的初值以设置 PWM3x 的 PWM 周期。
- 3、赋<PWM3xH: PWM3xL>寄存器的初值以设置 PWM3x 的占空比。
- 4、配置并启动定时器/计数器 T3：
 - 配置 T3 的计数时钟，T3CKS 位设置预分频比，T3CS 位选择 T3 工作时钟源；
 - 将 T3H/T3L 清 0；
 - 使能 T3 重载功能；
 - 将 T3ON 位置 1 以启动 T3。
- 5、将 PWM3CTL 寄存器的 PWM3xON 置 1 以启动 PWM3x。
- 6、将引脚 PWM3x 所对应的方向控制位 TRxx 清 0 使能引脚 PWM3x 的输出驱动器。

6.6 定时/计数器 T4

T4 是一个 16 位的定时/计数器，T4 的低 8 位在寄存器 T4L 中，高 8 位在寄存器 T4H 中，当 T4 计数值达到 65535 后，T4 的值再加 1 就会产生溢出，将 T4 中断标志位置 1。

T4 带有重载功能。当使能 T4 重载功能时，T4 计数器计数到 T4 重载寄存器中设置的值时，T4 计数器将清 0 重新开始计数，且将 T4 中断标志位置 1。

T4 属于外部单元，因此在使用 T4 中断时，需将 PUIE 位置 1，使能外设中断。

6.6.1 T4 原理框图

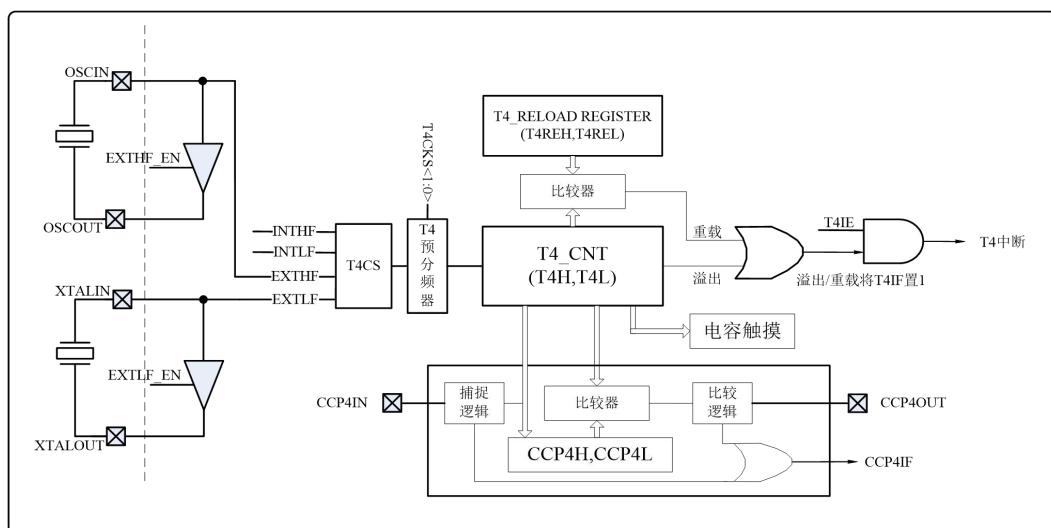


图 6.9 T4 原理框图

6.6.2 T4 相关寄存器

表 6-9 与 T4 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
164H	T4CTL	T4REN	T4BUZOE	T4CKS1	T4CKS0	-	T4CS1	T4CS0	T4ON
161H	T4H	T4 计数器高 8 位							
160H	T4L	T4 计数器低 8 位							
163H	T4REH	T4 重载设置寄存器高 8 位							
162H	T4REL	T4 重载设置寄存器低 8 位							
109H	CCP4CTL	-	-	-	CCP4OE	CCP4MOD3	CCP4MOD2	CCP4MOD1	CCP4MOD0
106H	CCP4H	CCP4 寄存器高 8 位							
105H	CCP4L	CCP4 寄存器低 8 位							

6.6.2.1 T4 控制寄存器

如寄存器 6.6 所示, T4 控制寄存器 (T4CTL) 用于启动/禁止 T4 以及选择 T4 模块的不同功能特性。

寄存器6.6: T4CTL: T4控制寄存器(地址: 164H)

复位值 0000 -000	bit7							bit0
	T4REN	T4BUZOE	T4CKS1	T4CKS0	-	T4CS1	T4CS0	T4ON
	R/W	R/W	R/W	R/W	U	R/W	R/W	R/W

T4REN: T4 重载功能使能位

- 0 = 禁止 T4 重载功能
- 1 = 使能 T4 重载功能

T4BUZOE: T4 蜂鸣器控制输出使能位

- 0 = 禁止 T4 蜂鸣器控制输出到 P5.0 口
- 1 = 使能 T4 蜂鸣器控制输出到 P5.0 口

T4CKS<1:0>: T4 输入时钟预分频比选择位

- 11 = 1/8 倍预分频比
- 10 = 1/4 倍预分频比
- 01 = 1/2 倍预分频比
- 00 = 1/1 倍预分频比

T4CS<1:0>: T4 定时/计数模式选择

- 00 = T4 时钟为内部高频时钟 INTHF
- 01 = T4 时钟为内部低频时钟 INTLF
- 10 = T4 时钟为外部高频时钟 EXTHF
- 11 = T4 时钟为外部低频时钟 EXTLF

T4ON: T4 启动控制位

- 1 = 启动 T4
- 0 = 停止 T4

6.6.2.2 CCP4 功能寄存器

寄存器6.7: CCP4CTL: CCP4控制寄存器(地址: 109H)

复位值 ---0 0000	bit7			bit0				
	-	-	-	CCP4OE	CCP4MO D3	CCP4MO D2	CCP4MO D1	CCP4MO D0
	U	U	U	R/W	R/W	R/W	R/W	R/W

CCP4OE: CCP4比较模式输出使能位

1 = 比较模式下使能引脚的CCP4输出

0 = 禁止比较模式下引脚的CCP4输出

CCP4MOD<3:0>: CCP4模式选择位

0000 = 捕捉/比较 关闭 (复位CCP4模块)

0001 = 未使用 (保留)

0010 = 比较模式: 匹配时输出电平翻转 (T4IF置1)

0011 = 未使用 (保留)

0100 = 捕捉模式: 在每个下降沿发生捕捉

0101 = 捕捉模式: 在每个上升沿发生捕捉

0110 = 捕捉模式: 每4 个上升沿发生捕捉

0111 = 捕捉模式: 每16 个上升沿发生捕捉

1000 = 比较模式: 比较匹配时输出高电平 (T4IF置1)

1001 = 比较模式: 比较匹配时输出低电平 (T4IF置1)

1010 = 比较模式: 比较匹配时产生软件中断 (T4IF置1, CCP4OUT不受影响)

1011 = 比较模式: 触发特殊事件 (T4IF置1, CCP4复位T4, 如果ADC使能将启动AD转换)

1100 = 保留

1101 = 保留

1110 = 保留

1111 = 保留

6.6.3 T4 预分频器

如寄存器 6.6 所示, T4 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。T4CKS 位(T1CTL<5:4>)对预分频计数器进行控制。T4 预分频计数器不能直接进行读写操作, 可通过写入 T4H 或 T4L 使预分频计数器清 0。

6.6.4 T4 计数时钟选择

用户可以通过 T4CTL 寄存器的 T4CS<1:0>位来选择 T4 的计数时钟源, 提供 4 个时钟源, 内部系统高频时钟、内部低频时钟、外部高频时钟和外部低频时钟。

6.6.5 T4 重载功能

置位 T4CTL 寄存器的 T4REN 位使能 T4 重载功能。通过设置 T4REH/T4REL 寄存器来设置 T4 的重载点。当设置 T4REH/T4REL 寄存器且使能重载功能后，T4 计数器从 0 开始计数，当计数到 T4REH/T4REL 寄存器设置的值时，T4 计数器清 0 并重新开始计数，T4IF 置 1。

T4REN 位置 1 时会载入一次 T4REH/T4REL 寄存器的值，当置位 T4REN 且在 T4 计数过程中对 T4REH/T4REL 寄存器写入新值时，T4 模块会在下一次重载时载入新的重载点。

6.6.6 T4 蜂鸣器输出功能

当使能重载功能时，可以通过 T4CTL 寄存器的 T4BUZOE 位使能蜂鸣器输出功能。蜂鸣器的输出引脚为 P5.0。

蜂鸣器的周期等于 2 倍重载设置值（T3REH/T3REL 寄存器设置值），占空比为 1:1。

6.6.7 捕捉比较功能 (CCP4)

定时器 T4 带有 CCP 功能：捕捉模式和比较模式。

6.6.7.1 捕捉模式

在捕捉模式下，当对应的 CCP4IN 引脚发生事件时，寄存器 CCP4H 和 CCP4L 捕捉 T4 寄存器的 16 位值，原理框图如下图 6.10 所示：

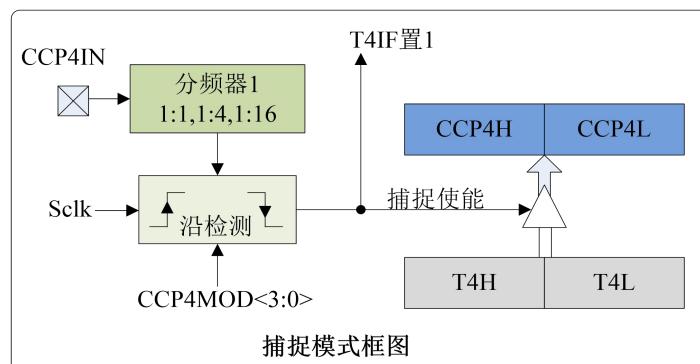


图 6.10 捕捉模式原理框图

触发捕捉的事件可被定义为以下四者之一，并且由 CCP4CTL 寄存器中的模式选择位 CCP4MOD<3:0>选择事件类型位配置：

- ◆ 0100 = 每个下降沿
- ◆ 0101 = 每个上升沿
- ◆ 0110 = 每4 个上升沿
- ◆ 0111 = 每16个上升沿

在捕捉模式下，应通过将相应的 CCP4 引脚配置为数字输入。

当一个捕捉发生时，硬件自动中断请求标志位 T4IF 置 1；它必须用软件清零。注意如果在 CCP4H、CCP4L 这对寄存器中的值被读取之前发生另一次捕捉，那么之前捕捉的值将被新捕捉的值覆盖。

当捕捉模式改变时，可能会产生错误的捕捉中断。用户应该在捕捉模式改变之前保持 T4IE 中断允许位清零以避免产生误中断。在捕捉模式发生任何改变之后也应清零中断标志位 T4IF。

CCP4CTL 寄存器中的 CCP4MOD<3:0>位指定了 4 种预分频器设置。每当关闭 CCP 模块或禁止捕捉模式时，就会清零预分频器计数器。这意味着任何复位都将清零预分频计数器。

虽然从一种捕捉预分频比切换到另一种捕捉预分频比不会将预分频计数器清零，但可能会产生误中断。因此要避免出现这种不期望的操作，应在改变预分频比前通过将 CCP4CTL 寄存器清零关闭该模块。

6.6.7.2 比较模式

在比较模式下，16 位 CCP4H、CCP4L 寄存器的值将不断与 T4 寄存器的值相比较。当两者匹配时，CCP 模块可能会出现以下几种情况：

- ◆ CCP4OUT 引脚的输出电平翻转
- ◆ CCP4OUT 引脚输出高电平
- ◆ CCP4OUT 引脚输出低电平
- ◆ 产生软件中断触发信号
- ◆ 产生特殊事件触发信号

CCP4OUT 引脚的动作取决于 CCP4CTL 寄存器中 CCP4MOD<3:0> 控制位的值。

所有比较模式都会产生中断。原理图如图 6.11 所示：

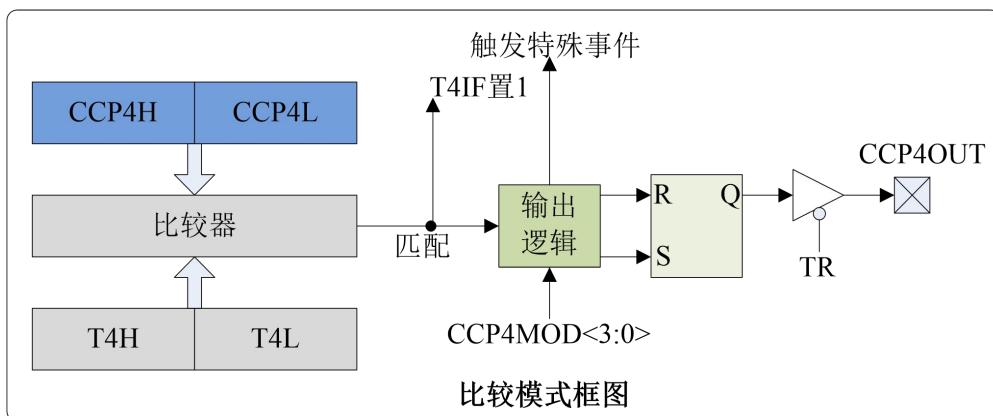


图 6.11 比较模式原理框图

用户必须通过将相应的 CCP4OUT 引脚配置为数字输出。

- 5) 当选择输出电平翻转模式时，比较匹配时，CCP4OUT 引脚的输出电平翻转，并将 T4IF

置1。

- 6) 当选择普通比较模式时（1010或1001）时，比较匹配时，CCP4OUT引脚输出高电平或者低电平，并且将T4IF置1。
- 7) 当选择了软件中断触发模式时（1010）时，比较匹配时，将T4IF置1，但是CCP模块不会控制CCP4OUT引脚。
- 8) 当选择了特殊事件触发模式（1011）时，比较匹配时，CCP会立即产生特殊事件触发输出，将T4IF置1，此时如果ADC已使能，将启动AD转换。但T4计数寄存器不会立即复位，直到T4计数脉冲的下一个上升沿才复位。从而使CCP4H、CCP4L寄存器实际上成为了定时器T4的16位可编程周期寄存器。

6.6.8 T4 中断

在三种情况下会使 T4 中断标志位置 1:

1. 当 T4 计数值达到 65535 后, T4 的值再加 1 就会产生溢出, 将 T4 中断标志位置 1;
2. 当 T4 重载时, T4 中断标志位置 1。
3. 当用于 CCP 功能时产生中断, T4 中断标志位置 1。

6.6.9 T4 工作在休眠模式

当选择低频时钟信号作为 T4 的计数时钟时, T4 模块可以工作在休眠模式下。当使能 T4 中断时, 系统会在 T4 溢出或者重载时唤醒。

7 ADC12 模数转换模块

ADC12 模数转换模块可将模拟输入信号转换为 12 位二进制值。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 12 位寄存器中。可通过软件方式选择 VDD 或施加在 VREFIN 引脚上的电压作为转换使用的参考电压。

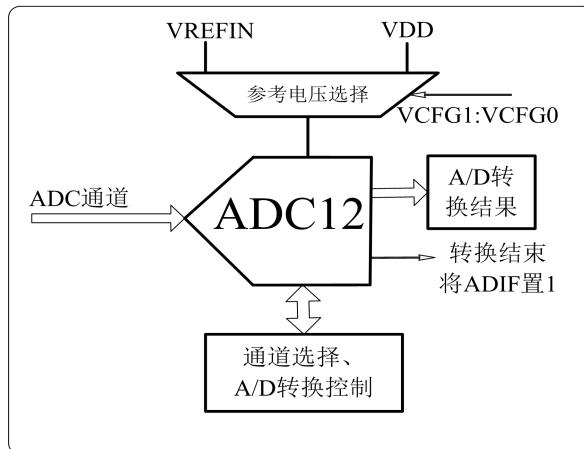


图 7.1 ADC12 模块结构框图

7.1 与 ADC12 相关的寄存器

表 7-1 与 ADC12 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
1FH	ADCCTL0	ADLR	T2CCR0 ON	-	-	-	-	START	ADEN
3FH	ADCCTL1	ADCALEN	ADCS2	ADCS1	ADCS0	VCFG1	VCFG0	-	-
63H	ADCCTL2		ADCHS6	ADCHS5	ADCHS4	ADCHS3	ADCHS2	ADCHS1	ADCHS0
58H	ADCINTCTL	T2CCR1 ON	-	-	-	-	-	INTCTL1	INTCTL0
236H	TEMPSNR	-	-	-	-	-	-	-	TSEN
1EH	ADCDATA0H	ADC 数据寄存器 0 高字节							
3EH	ADCDATA0L	ADC 数据寄存器 0 低字节							
50H	ADCDATA1H	ADC 数据寄存器 1 高字节							
51H	ADCDATA1L	ADC 数据寄存器 1 低字节							
59H	ADCDATA2H	ADC 数据寄存器 2 高字节							
5AH	ADCDATA2L	ADC 数据寄存器 2 低字节							
5EH	ADCDATA3H	ADC 数据寄存器 3 高字节							
6AH	ADCDATA3L	ADC 数据寄存器 3 低字节							
159H	VREFCAL0	内部参考电压校准寄存器 0							
15AH	VREFCAL1	内部参考电压校准寄存器 1							

7.1.1 ADC12 控制寄存器 0(ADCCTL0)

寄存器7.1: ADCCTL0: A/D控制寄存器0(地址: 1FH)

复位值 00----00	bit7							bit0
	ADLR	T2CCR0 ON	-	-	-	-	START	ADEN
	R/W	R/W	U	U	U	U	R/W	R/W

ADLR: ADC12 转换结果输出格式选择位

1 = 结果右对齐

0 = 结果左对齐

T2CCR0ON: T2CRR0H/L 与 T2H/L 匹配时触发 ADC12 启动使能位

1 = 使能 T2 触发 ADC12 启动

ADC12 采样转换结束后, 将采样的数据保存到

ADCDATA0H 和 ADCDATA0L 中, 并且硬件自动将该位清 0

0 = 禁止 T2 触发 ADC12 启动

START: ADC12 转换状态位

1 = ADC12 转换正在进行, 该位置 1 将启动 ADC12 转换, 在转换结束后该位将被硬件自动清 0。

0 = ADC12 转换结束或者未进行

ADEN: ADC12 工作使能位

1 = 使能 ADC12 转换模块工作

0 = ADC12 转换器关闭且不消耗工作电流

注: 如果内部使用参考电压VREFOUT时 (用作ADC12参考电压和比较器参考电压) ,
不需要将寄存器VRECTL1的VREOE位置1。

7.1.2 ADC12 控制寄存器 1(ADCCTL1)

寄存器7.2: ADCCTL1: A/D控制寄存器1(地址: 3FH)

复位值 0000 0000	bit7							bit0
	ADCALE N	ADCS2	ADCS1	ADCS0	VCFG1	VCFG0	保留	保留
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ADCALEN: ADC12 失调校准使能位

0 = 关闭 ADC12 校准

1 = 打开 ADC12 校准

ADCS<2:0>: ADC12 转换时钟选择位

000 = Fad=SCLK/2

001 = Fad=SCLK/8

010 = Fad=SCLK/32

x11 = Fad=内部专用 500KHz 时钟

100 = Fad=SCLK/4

101 = Fad=SCLK/16

110 = Fad=SCLK/64

VCFG<1:0>: ADC12 转换参考电压选择位

00 = 断开

01 = VDD 作为 ADC12 参考电压

10 = VREFIN 作为 ADC12 参考电压

11 = 保留

7.1.3 ADC12 控制寄存器 2(ADCCTL2)

如寄存器 7.3 所示，ADC12 控制寄存器 2 包含 ADC12 通道选择位 CHS<6:0>。

寄存器7.3: ADCCTL2:A/D控制寄存器2 (地址: 63H)

复位值	bit7								bit0
	-	ADCHS6	ADCHS5	ADCHS4	ADCHS3	ADCHS2	ADCHS1	ADCHS0	
-000 0000	U	R/W							

ADCHS<6:5>: 保留位，请勿将该两位置 1，否则将无法正常配置 ADC12 通道

ADCHS<4:0>: 模拟通道选择位

ADx = ADC12 模拟通道 x

表 7-2 ADC 通道配置对应表

ADCHS	通道数	ADCHS	通道数
00 0000	P0.0/AD0	01 0101	P2.6/AD21
00 0001	P0.1/AD1	01 0110	P3.0/AD22
00 0010	P0.4/AD2	01 0111	P3.1/AD23
00 0011	P0.5/AD3	01 1000	P3.2/AD24
00 0100	P1.3/AD4	01 1001	P0.6/AD25
00 0101	P1.4/AD5	01 1010	P4.0/AD26
00 0110	P1.5/AD6	01 1011	P4.1/AD27
00 0111	P1.7/AD7	01 1100	P4.2/AD28
00 1000	P2.0/AD8	01 1101	P4.3/AD29
00 1001	P2.1/AD9	01 1110	P4.5/AD30
00 1010	P2.2/AD10	01 1111	P4.6/AD31
00 1011	P2.3/AD11	10 0000	P4.7/AD32
00 1100	P3.3/AD12	10 0001	P5.4/AD33
00 1101	P3.4/AD13	10 0010	-
00 1110	P3.5/AD14	10 0011	-
00 1111	P3.6/AD15	10 0100	OPOUT
01 0000	P0.3/AD16	10 0101	-
01 0001	P1.0/AD17	10 0110	FVR_OUT
01 0010	P1.1/AD18	10 0111	TS_OUT
01 0011	P2.4/AD19	10 1000	-
01 0100	P2.5/AD20	其他	保留

7.1.4 ADC12 中断控制寄存器 (ADCINTCTL)

寄存器7.4: ADCINTCTL: ADC12中断控制寄存器(地址:58H)

复位值 0---00	bit7 T2CCR10 N							bit0 INTCTL1 INTCTL0
		R/W	U	U	U	U	U	

T2CCR1ON: T2CCR1H/L 寄存器触发 ADC12 启动使能位

1 = 使能 T2 触发 ADC12, ADC12 采样转换结束后, 将采样的数据保存到
ADCDATA4H 和 ADCDATA4L 中, 并且硬件自动将该位清 0

0 = 禁止 T2 触发 ADC12

INTCTL<1:0>: 保留位

7.1.5 温度传感器配置寄存器(TEMPSNR)

寄存器7.5: TEMPSNR: 温度传感器配置寄存器(地址:236H)

复位值 ----00	bit7							bit0 TSEN
		U	U	U	U	U	U	

TSEN: 温度传感器使能位

0 = 未使能温度传感器
1 = 使能温度传感器

7.2 通道的选择

如图 7.1 所示, KF8F513X 中的 ADC12 转换模块的输入通过寄存器 ADCCTL2(如寄存器 7.3 所示)进行通道的选择, 除外部输入通道外, 还提供 2 路内部信号通道。

当选择外部的模拟信号作为 ADC12 转换的输入时, 需要将对应的引脚配置为模拟输入口; 当选择内部信号作为 ADC12 转换的输出时, 无需占用任何 I/O 引脚; ADC12 可选的内部信号输入及配置方式如下 (用户如需使用温度传感器, 通过 TEMPSNR 寄存器的 TSEN 位置 1 来使能该功能):

- 通道 14=OPOUT, 运算放大器的输出
- 通道 15=TSOUT, 温度传感器的输出

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。

7.3 ADC12 转换参考电压的选择

KF8F513X 中 ADC12 模块的参考电压可以选择电源电压(VDD)或外部参考电压(VREFIN)。通过寄存器 ADCCTL1(如寄存器 7.2 所示)的 VCFG<1:0>设置参考电压。

7.4 转换时钟的选择

完成一次 ADC12 转换所需的时间为 $13T_{ad}$ 。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 ADC12 转换时钟周期和频率。

7.5 输出格式

KF8F513X 中 ADC12 转换的结果为 12 位二进制数, ADC12 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。

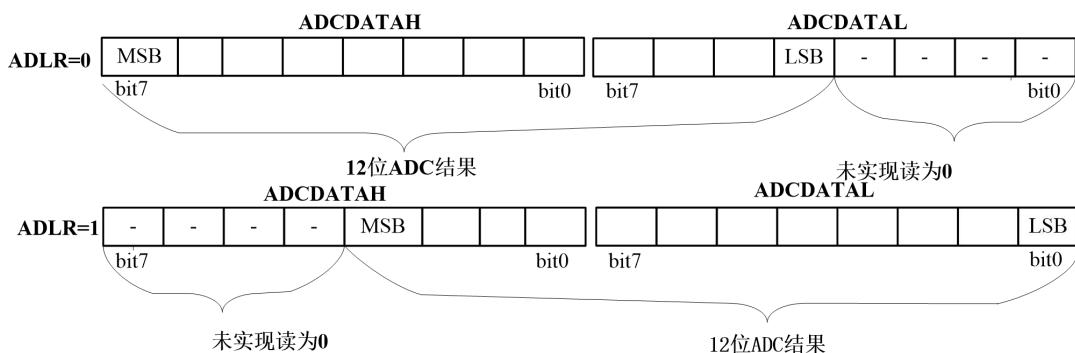


图 7.2 ADC 输出对齐方式

7.6 AD 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 AD 转换。当转换结束时, ADC12 模块将:

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

可采取在程序中将 START 位清 0 的方法中止当前转换操作。在 AD 转换采样全部结束之前, ADCDATA0H:ADCDATA0L 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。AD 转换被中止后, 需至少等待 $2T_{ad}$ 的延时时间后才能开始下一次数据采集。

7.7 ADC12 工作在休眠模式

ADC12 转换器模块可以在休眠模式下工作。这需要把 ADC12 的时钟源设定为 ADC12 专用的内部振荡器。当选择了专用内部时钟源后, ADC12 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 ADCDATA0H:ADCDATA0L 寄存器。如果 ADC12 中断被使能, 器件将从休眠状态唤醒。如果 ADC12 中断被禁止, ADC12 模块在转换完成后被关闭。

如果 ADC12 时钟源不是 ADC12 专用的内部振荡器, 执行 IDLE 指令将导致当前转换操作中止, 并使 ADC12 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，ADC12 模块将被关闭，任何正在进行中的转换操作被中止。ADCDATA0H:ADCDATA0L 寄存器中的值不变。

7.9 使用 ADC12 转换器的设置

启动 ADC12 转换器时的设置：

1. 选择 ADC12 采样输入通道，设置 AD 转换结果对齐方式；
2. 将对应的 ADC12 采样输入通道设置为模拟输入模式；
3. 如果需要使能 ADC12 的失调校准，将 ADCCTL1 寄存器的 ADCALEN 位置 1；
4. 选择参考电压和 ADC12 采样时钟频率，打开 AD 转换；
5. 如果采用中断方式，使能 ADC12 转换中断；
6. 等待 ADC12 所需的采集时间；
7. 启动 AD 转换；
8. 查询 ADC12 是否转换完成(START=0)或进入 AD 中断；
9. 读取 AD 转换结果。

8 PWM1/2 模块

KF8F513X 单片机的 CCP (捕捉/比较/PWM) 资源如下表所示:

表 8-1 CCP 资源列表

模块	捕捉/比较资源	PWM 资源		使用定时器情况
PWM1	-	8 位标准 PWM	4 路	T1 低 8 位
PWM2	-	8 位标准 PWM (可配置为 16 位)	4 路	T1 高 8 位
CCP3	带捕捉/比较模式	16 位标准 PWM	4 路	T3
CCP4	带捕捉/比较模式	-	-	T4
CCP5	带捕捉/比较模式	16 位增强型 PWM	1 路全桥 +2 路半桥	T2

注: 通过使能 T1 的重载功能使能位 T1RLD, PWM1 和 PWM2 可组合成 16 位标准 PWM 模块进行使用, 输出端口同 PWM2。

本章节对标准 PWM 模块 PWM1 和 PWM2 进行介绍。CCP3 模块详见定时/计数器 T3 章节; CCP4 模块的捕捉/比较模式详见定时/计数器 T4 章节; CCP5 详见 CCP5(捕捉/比较/PWM5)模块章节。

8.1 PWM1/2 输出端口

表 8-2 PWM1/2 对应 IO 口

PWM1	对应 I/O 口	备注
PWM10	P0.1	
PWM11	P3.0	
PWM12	P3.3	
PWM13	P3.6	
PWM2	对应 I/O 口	备注
PWM20	P0.0/P1.1/P3.7	通过 APFCTL3 寄存器相关位进行配置
PWM21	P3.1/P3.2	
PWM22	P3.4/P5.7	
PWM23	P3.5	

8.2 工作原理

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 PWM1xL(或 PWM2xH)设置。

图 9.1 显示了 PWM1/2 逻辑框图。其中 PP1 为 PWM1 模块的周期寄存器，PWM1xL 为 PWM1 模块占空比设置寄存器，使用 PWM1/2 时需要将定时器 T1 配置给 PWM1/2 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，PWM1 引脚被置 1，此时 T1L 被清零，重新开始计数，当 T1L 的计数值和 PWM1xL 相等时，PWM1 引脚清零。改变 PP1 和 PWM1xL 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。

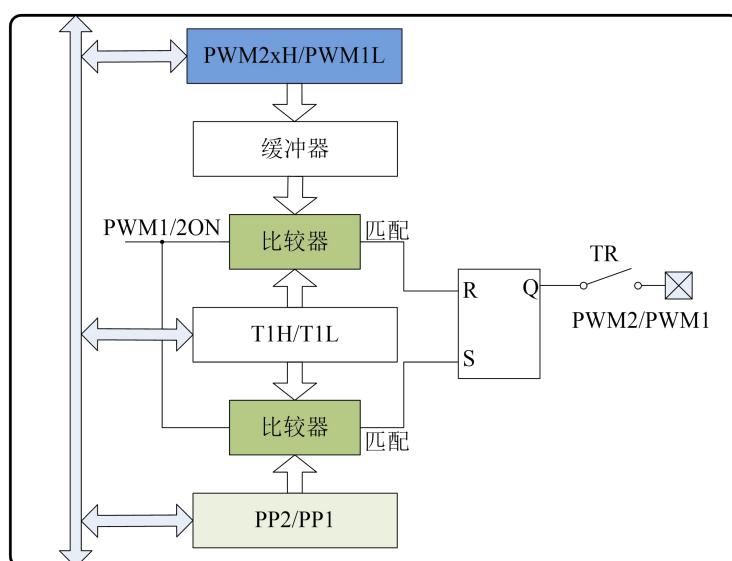


图 8.1 PWM1/2 逻辑框图

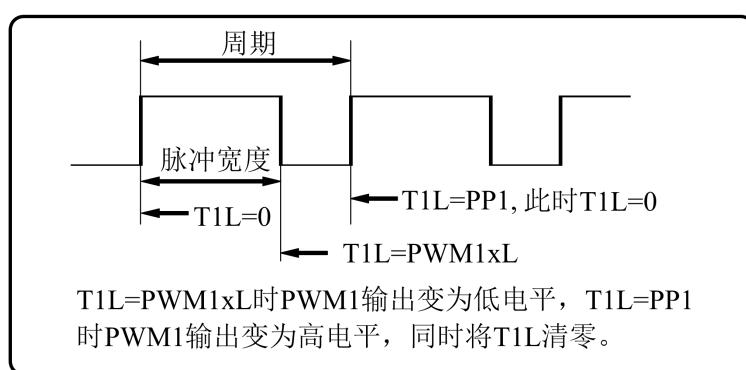


图 8.2 PWM1 输出波形图

8.3 PWM1/2 相关的寄存器

表 8-3 与 PWM1/2 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
21DH	PWMCTL0	-	-	-	-	PWM13O N	PWM12O N	PWM11O N	PWM10O N
21EH	PWMCTL1	-	-	-	-	PWM23O N	PWM22O N	PWM21O N	PWM20O N
21FH	PP1	PWM1 周期设置寄存器							
220H	PP2	PWM2 周期设置寄存器							
221H	PWM10L	PWM10 占空比位寄存器							
222H	PWM20H	PWM20 占空比位寄存器							
223H	PWM11L	PWM11 占空比位寄存器							
224H	PWM21H	PWM12 占空比位寄存器							
225H	PWM12L	PWM13 占空比位寄存器							
226H	PWM22H	PWM14 占空比位寄存器							
227H	PWM13L	PWM15 占空比位寄存器							
228H	PWM23H	PWM16 占空比位寄存器							

8.3.1 PWM1/2 启动控制寄存器

寄存器9.1: PWMCTL0: PWM启动控制寄存器0(地址:21DH)

复位值	bit7				bit0			
	-	-	-	-	PWM13ON	PWM12ON	PWM11ON	PWM10ON
0000 0000	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

寄存器9.2: PWMCTL1: PWM启动控制寄存器1(地址:21EH)

复位值	bit7				bit0			
	-	-	-	-	PWM23ON	PWM22ON	PWM21ON	PWM20ON
0000 0000	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

PWM_{yx}ON: PWM_{yx} 启动控制位

1 = 启动 PWM_{yx}

0 = 禁止 PWM_{yx}

注: x=0~3, y=1~2。

8.4 PWM1/2 周期

PWM 周期通过 PP1/2 (地址: 16H/32H)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 9.1 进行计算。T1 计数时钟周期由 T1 工作时钟源和预分频比共同决定。

式 8.1 PWM1/2 周期计算公式

$$\text{PWM}_x \text{ 周期} = (\text{PP}_x + 1) \cdot \text{Tt1}, \text{ 其中: } x=1 \text{ 或 } 2, \text{ Tt1} = \text{T1} \text{ 计数时钟周期}$$

8.5 PWM1/2 占空比

PWM1x 和 PWM2x 的占空比分别通过 PWM1xL 和 PWM2xH 设置。脉冲宽度和占空比通过式 9.2 和式 9.3 计算：

式 8.2 PWM1/2 脉冲宽度计算公式

$$\begin{aligned}\text{PWM1x脉冲宽度} &= \text{PWM1xL} \cdot Tt1 \\ \text{PWM2x脉冲宽度} &= \text{PWM2xH} \cdot Tt1 \\ \text{其中: } x &= 0 \sim 3, Tt1 = T1 \text{计数时钟周期}\end{aligned}$$

式 8.3 PWM1/2 占空比计算公式

$$\text{PWM1x占空比} = \frac{\text{脉冲宽度}}{\text{PWM1周期}} = \frac{\text{PWM1xL}}{\text{PP1+1}}$$

$$\text{PWM2x占空比} = \frac{\text{脉冲宽度}}{\text{PWM2周期}} = \frac{\text{PWM2xH}}{\text{PP2+1}}$$

其中：x=0~3。

8.6 PWM1/2 分辨率

分辨率决定在给定周期内的占空比数。KF8F513X 中当 PP1/2 为 255 时，PWM 的最大分辨率为 8 位，8 位分辨率产生 256 个离散的占空比。分辨率的计算公式如式 9.4 所示。

式 8.4 PWM1/2 分辨率计算公式

$$\text{分辨率} = \frac{\log[(\text{PP}x+1)]}{\log 2} \text{ 位} \quad (x=1, 2)$$

8.7 PWM1/2 中断

PWM2 有一个专门的中断使能位 PWM2IE 和中断标志位 PWM2IF，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 清零，将 T1IF/PWM2IF 置 1，如果允许 T1/ PWM2 中断，将会转入对应的中断子程序中。

8.8 PWM2 配置为 16 位标准 PWM

用户可通过将 T1CTL 寄存器的 T1RLD 位置 1，将 PWM2 配置为 16 位标准 PWM。此时，PWM1 失效，被用于 16 位标准 PWM 的低 8 位使用；寄存器 PP2 作为周期寄存器高 8 位，PP1 作为周期寄存器低 8 位；寄存器 PWM2xH 作为占空比配置寄存器的高 8 位，PWM1xL 作为占空比配置寄存器的低 8 位。

配置为 16 位标准 PWM 后，PWMCTL0 寄存器失效，通过 PWMCTL1 寄存器的 PWM2xON 位控制 PWM2 的使能。T1 中断相关位(T1IF、T1IE 和 PT1)和 PWM2 中断相关位 (PWM2IF、PWM2IE 和 PPWM2) 功能完全相同，用户可选择任意一组。

8.9 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。

8.10 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。

8.11 PWM1/2 使用方法

PWM1/2 工作的设置应按照以下步骤：

- 1、将 PWM 对应 I/O 口配置为输入口，禁止 PWM1 或 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 PWM1xL 或 PWM2xH 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 T1CS 位和 T1CLKEN 位选择 T1 的计数时钟 (一般选择内部时钟)；
 - 配置 T1CTL 寄存器的 T1CKS1 和 T1CKS0 以选择 T1 的预分频比；
 - 将 T1L/H 清零；
 - 如需使用 16 位 PWM，将 T1RLD 位置 1；
 - 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。

9 CCP5(捕捉/比较/PWM5)模块

在 CCP5 模块中, PWM5 为 16 位模式, T1/T2 计数模式为 16 位。PWM5 的 16 位周期寄存器由{PP5H,PP5L}组成。

在捕捉比较模式下, 寄存器 PWM5L1 和 PWM5L0 分别作为数据寄存器的高 8 位和低 8 位寄存器; 捕捉和比较的 16 位数据寄存器 PWM5L0 和 PWM5L1 与 T1L 和 T1H 进行配合使用。

在 PWM5 模式下, 寄存器 PWM5L0 和 PWM5H0 作为 16 位占空比寄存器使用。

9.1 CCP5 引脚复用

CCP5 的相关功能引脚, 可以通过 APFCTL2 和 APFCTLB 寄存器进行配置, 选择合适的通道。其中, APFCTL2 寄存器用于 P5A、P5B、P5C 和 P5D 的通道选择; APFCTLB 寄存器用于 CCP5 捕捉比较模式的通道选择。

表 9-1 CCP5 引脚复用对应表

CCP5 相关功能引脚	对应 I/O 口	备注
P5A	P1.2 / P2.0 / P3.0	通过 APFCTL3 寄存器相关位进行配置
P5B	P1.3 / P2.1 / P2.5	
P5C	P1.4 / P2.2 / P2.3	
P5D	P1.5 / P3.6 / P5.3	
P5A2	P1.6	
P5B2	P1.7	
P5A3	P5.6	
P5B3	P5.5	
CCP5	P1.2 / P2.0 / P2.3 / P3.5 / P1.1 / P5.7 / P4.4 / P5.0	通过 APFCTLB 寄存器相关位进行配置

9.2 CCP5 相关寄存器

当 CCP 模块配置为捕捉比较模式时, PWM5L1:PWM5L0 作为 CCP 的数据寄存器使用。

在捕捉模式下时,当 CCP5IN 引脚发生事件时, PWM5L1:PWM5L0 这对寄存器捕捉 T1H 和 T1L 寄存器的 16 位值。T1H 和 T1L 寄存器的值自动传递给 PWM5L1:PWM5L0 这对寄存器。

在比较模式下, 16 位 PWM5L1:PWM5L0 寄存器的值将不断与 T1H/L 寄存器的值相比较。当两者匹配时, CCP 就会触发相应的事件。

当 CCP 模块配置为 PWM5 模式时, PWM5H0:PWM5L0 为 CCP 模块的占空比寄存器; PP5H: PP5L 为周期寄存器; PWM5H1: PWM5L1 为 PWM5 占空比缓冲寄存器 (与用户无关)。

表 9-2 CCP 模块的相关寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
52H	PP5L								PWM5 周期寄存器低 8 位
42H	PP5H								PWM5 周期寄存器高 8 位
55H	PWM5L0								捕捉比较模式时, PWM5L0=CCP 数据寄存器低 8 位 PWM5 模式时, PWM5L0= PWM5 通道 1 占空比寄存器低 8 位
43H	PWM5H0								PWM5 通道 1 占空比寄存器高 8 位
56H	PWM5L1								捕捉比较模式时, PWM5L1=CCP 数据寄存器高 8 位 PWM5 模式时, PWM5L1= PWM5 通道 1 低 8 位占空比缓冲寄存器 (与用户无关)
44H	PWM5H1								PWM5 通道 1 高 8 位占空比缓冲寄存器 (与用户无关)
10CH	PWM5L2								PWM5 通道 2 占空比寄存器低 8 位
11EH	PWM5H2								PWM5 通道 2 占空比寄存器高 8 位
10DH	PWM5L3								PWM5 通道 3 占空比寄存器低 8 位
11FH	PWM5H3								PWM5 通道 3 占空比寄存器高 8 位
57H	PWM5CTL0	P5CH1M OD1	P5CH1M OD0	P5CH3 MOD	P5CH2MO D	P5MOD3	P5MOD2	P5MOD1	P5MOD0
119H	PWM5FC	-	-	FCA3	FCB3	FCA2	FCB2	FCA	FCB
11DH	PWM5PC	-	-	PCA3	PCB3	PCA2	PCB2	PCA	PCB
10EH	PWM5OC	-	-	OCA3	OCB3	OCA2	OCB2	OCA	OCB
5BH	PWM5CTL1	P5RSEN	P5DC6	P5DC5	P5DC4	P5DC3	P5DC2	P5DC1	P5DC0
5CH	P5ASCTL0	P5ASE	P5ASS2	P5ASS1	P5ASS0	P5SSAC1	P5SSAC0	P5SSBD1	P5SSBD0
5DH	PSTRCTL0	-	-	-	STRSYNC	STREND	STRENC	STRENB	STRENA
11AH	P5ASCTL1	P5SSA31	P5SSA30	P5SSB31	P5SSB30	P5SSA21	P5SSA20	P5SSB21	P5SSB20
11BH	PSTRCTL1	-	-	-	-	STRENA3	STRENB3	STRENA2	STRENB2
11CH	PWM5CTL2	-	-	PFUSES	-	-	UDEVT1	UDEVT0	UDEN
262H	APFCTL2	P5DSEL1	P5DSEL0	P5CSEL 1	P5CSEL0	P5BSEL1	P5BSEL0	P5ASEL1	P5ASEL0
26BH	APFCTLB	-	-	-	-	-	CCP5 SEL2	CCP5 SEL1	CCP5 SEL0

9.3 捕捉模式

在捕捉模式下, 当对应的CCP5IN引脚发生事件时, PWM5L1:PWM5L0这对寄存器捕捉T1H和T1L 寄存器的16位值, 原理框图如下图所示:

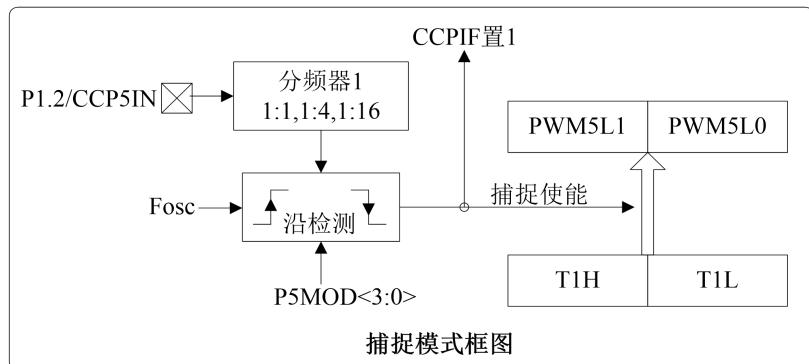


图 9.1 捕捉模式原理框图

触发捕捉的事件可被定义为以下四者之一, 并且由 PWM5CTL0 寄存器中的模式选择位 P5MOD<3:0>选择事件类型位配置:

- ◆ 0100 = 每个下降沿
- ◆ 0101 = 每个上升沿
- ◆ 0110 = 每4 个上升沿
- ◆ 0111 = 每16个上升沿

在捕捉模式下, 将 CCP5IN 引脚配置为数字输入。

当一个捕捉发生时, 硬件自动中断请求标志位EIF2寄存器中的CCP5IF置1; 它必须用软件清零。注意如果在PWM5L1和PWM5L0这对寄存器中的值被读取之前发生另一次捕捉, 那么之前捕捉的值将被新捕捉的值覆盖。

当捕捉模式改变时, 可能会产生错误的捕捉中断。用户应该在捕捉模式改变之前保持 EIE2 寄存器中的 CCP5IE 中断允许位清零以避免产生误中断。在捕捉模式发生任何改变之后也应清零 EIF2 寄存器中的中断标志位 CCP5IF。

PWM5CTL0 寄存器中的 P5MOD<3:0> 位指定了 4 种预分频器设置。每当关闭 CCP 模块或禁止捕捉模式时, 就会清零预分频器计数器。这意味着任何复位都将清零预分频计数器。

虽然从一种捕捉预分频比切换到另一种捕捉预分频比不会将预分频计数器清零, 但可能会产生误中断。因此要避免出现这种不期望的操作, 应在改变预分频比前通过将 PWM5CTL0 寄存器清零关闭该模块。

注: T1必须运行在定时模式或同步计数模式下CCP模块才能使用捕捉功能。
在异步计数模式下无法进行捕捉操作。

9.4 比较模式

在比较模式下，16位的PWM5L1:PWM5L0寄存器的值将不断与T1寄存器的值相比较。当两者匹配时，CCP模块可能会出现以下几种情况：

- ◆ CCP5OUT引脚的输出电平翻转
- ◆ CCP5OUT引脚输出高电平
- ◆ CCP5OUT引脚输出低电平
- ◆ 产生软件中断触发信号
- ◆ 产生特殊事件触发信号

CCP5OUT引脚的动作取决于PWM5CTL0寄存器中P5MOD<3:0>控制位的值。

所有比较模式都会产生中断。原理图如下图所示：

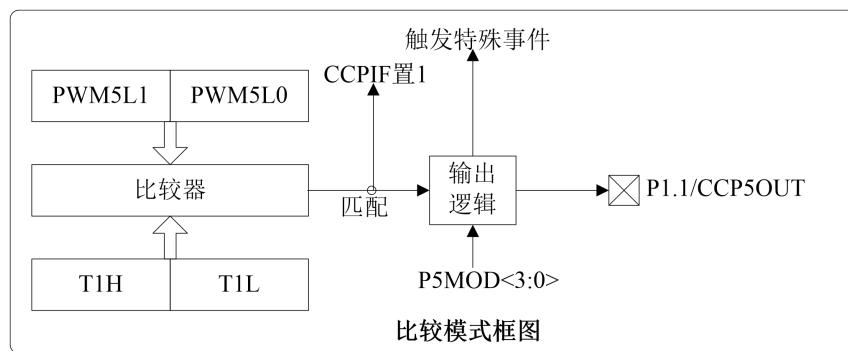


图 9.2 比较模式原理框图

用户必须将CCP5OUT引脚配置为数字输出。

在比较模式下，T1必须运行在定时模式或同步计数模式，计数周期至少大于一个机器周期。在异步计数模式下，可能无法进行比较操作。

- 9) 当选择输出电平翻转模式（P5MOD<3:0> = 0010）时，比较匹配时，CCP5OUT引脚的输出电平翻转，并将CCP5IF置1。
- 10) 当选择普通比较模式时（P5MOD<3:0> = 1010或1001）时，比较匹配时，CCP5OUT引脚输出高电平或者低电平，并且将CCP5IF置1。
- 11) 当选择了软件中断触发模式时（P5MOD<3:0> = 1010）时，比较匹配时，将CCP5IF置1，但是CCP模块不会控制CCP5OUT引脚。
- 12) 当选择了特殊事件触发模式（P5MOD <3:0> = 1011）时，比较匹配时，CCP会立即产生特殊事件触发输出，将CCP5IF置1，此时如果ADC已使能，将启动AD转换。但T1H/T1L寄存器不会立即复位，直到T1计数脉冲的下一个上升沿才复位。从而使PWM5L1/PWM5L0寄存器实际上成为了定时器1(T1)的16位可编程周期寄存器。

9.5 PWM5 模式

PWM5 为带有死区控制功能的增强型 PWM 模块。PWM5 有 3 个通道，每个通道都有一个 16 位的占空比设置寄存器。PWM5 的原理框图如下图所示。

PWM5不同通道所对应的占空比设置寄存器

通道1占空比寄存器	PWM5H0	PWM5L0
通道2占空比寄存器	PWM5H2	PWM5L2
通道3占空比寄存器	PWM5H3	PWM5L3

图 9.3 PWM5 通道和占空比设置寄存器对应图

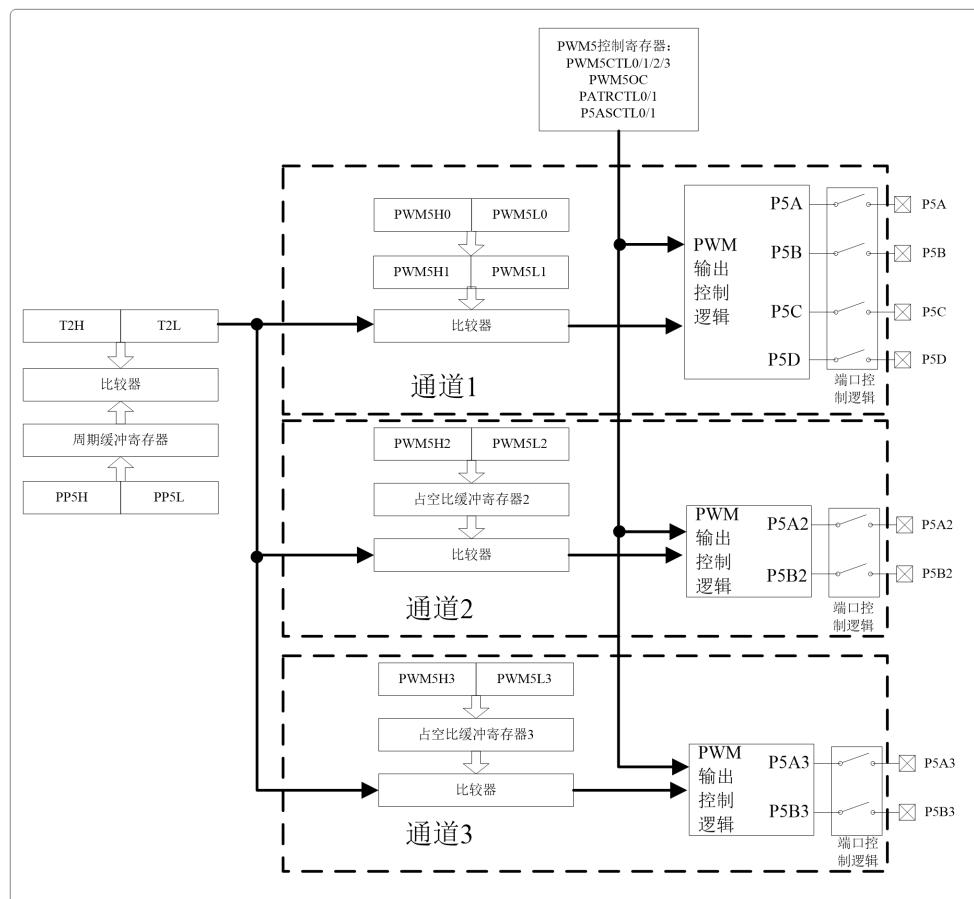


图 9.4 PWM5 原理框图

PWM5 最多可在 8 个不同的引脚输出 PWM 信号，分辨率最高 16 位。其中，通道 1 有 4 个引脚分别为 P5A、P5B、P5C 和 P5D；通道 2 和通道 3 分别有两个输出引脚，分别为 P5A2/P5B3 和 P5A3/P5B3。

PWM5 通道 1 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 PWM5CTL0 中的 P5CH1MOD<1:0>位选择 4 种输出模式之一，通过 P5MOD<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)。

PWM5 通道 2 和通道 3 一致，都有 2 种输出模式：单输出和半桥输出模式，可以通过 PWM5CTL0 中的 P5CH<3:2>MOD 位选择通道 2 和通道 3 的输出模式。在单输出模式和半桥输出模式下，均可通过 PSTRCTL1 寄存器控制引脚作为 PWM 引脚还是通用 IO 引脚。在半桥模式下，通道 1/2/3 通均可以获得一个带死区控制的互补 PWM 输出。

PSTRCTL0/1 寄存器控制引脚作为 PWM 引脚还是通用 IO 引脚；通过寄存器 PWM5OC 中的 OCA3/ OCB3/ OCA2/ OCB2 位选择通道 2 和通道 3 作为 PWM 输出还是强制输出；通过 PWM5CTL3 中的 PCA3/ PCB3/ PCA2/ PCB2 位可以设置通道 2 和通道 3 输出引脚的极性；通过 PWM5CTL3 寄存器的 FCA3/ FCB3/ FCA2/ FCB2 位可以设置通道 2 和通道 3 引脚作为强制输出时的电平。

注：通道 1 通过选择单输出模式可以选择指定引脚输出 PWM 波形或作为 IO 口；通过选择半桥模式可以获得一个带死区控制的互补 PWM 输出；通过选择全桥模式可以实现 PWM 输出和强制输出同时使用，通过正向和反向模式还可以控制输出的极性，但对 PWM 的输出有限制，只能从 P5B 或 P5D 中输出 PWM 波形。如下表所示。

表 9-3 PWM5 通道 1 输出模式控制及有效引脚

PWM5 输出模式	P5CH1MOD<1:0>	有效引脚
单输出模式	00	默认将 P5A 配置为 PWM 输出，P5B、P5C 和 P5D 配置为端口引脚；可通过 PSTRCTL0 寄存器中的 STREN<A:D>各位置 1，分别将 P5A、P5B、P5C 和 P5D 配置为 PWM 输出；PWM5 最多可在 4 个引脚输出 PWM 信号。
半桥输出模式	10	P5A 和 P5B 配置为调制输出；P5C 和 P5D 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	P5D 配置为 PWM 调制输出；P5A 为有效电平；P5B 和 P5C 为无效电平
全桥反向输出模式	11	P5B 配置为 PWM 调制输出；P5C 为有效电平，P5A 和 P5D 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。

表 9-4 PWM5 通道 2 输出模式控制及有效引脚

PWM5 输出模式	P5CH2MOD	有效引脚
单输出模式	0	默认将 P5A2 配置为 PWM 输出，P5B2 配置为端口引脚；可通过 PSTRCTL1 寄存器中的 STRENA2 和 STRENB2 各位置 1，分别将 P5A2、P5B2 配置为 PWM 输出。
半桥输出模式	1	P5A2 和 P5B2 配置为调制输出；半桥输出模式带有死区控制功能

表 9-5 PWM5 通道 3 输出模式控制及有效引脚

PWM5 输出模式	P5CH3MOD	有效引脚
单输出模式	0	默认将 P5A3 配置为 PWM 输出, P5B3 配置为端口引脚; 可通过 PSTRCTL1 寄存器中的 STRENA3 和 STRENB3 各位置 1, 分别将 P5A3、P5B3 配置为 PWM 输出。
半桥输出模式	1	P5A3 和 P5B3 配置为调制输出; 半桥输出模式带有死区控制功能

表 9-6 PWM5 通道 3 输出模式控制及有效引脚

P5MOD<3:0>	PCA/PCB	PWM 输出极性
高有效	高有效	高有效
高有效	低有效	低有效
低有效	高有效	低有效
低有效	低有效	高有效

注: 上述满足同或结果, 注意这里的同或与数值 1 和 0 的同或无联系, 这里仅从逻辑关系上去考虑, 例如当通过 P5MOD<3:0>设置输出极性为高有效和通过 PCA/PCB 设置输出为高有效时, 两者同或得实际 PWM 输出极性高有效, 当通过 P5MOD<3:0>设置输出极性为低有效和通过 PCA/PCB 设置输出为高有效时, 两者同或得实际 PWM 输出极性为低有效。

9.5.1 PWM5 相关控制寄存器

9.5.1.1 PWM5CTL0 寄存器

寄存器: PWM5CTL0: PWM5控制寄存器0(地址: 57H)

复位值	bit7	P5CH1MO D1	P5CH1MO D0	P5CH3MO D	P5CH2MO D	P5MOD3	P5MOD2	P5MOD1	bit0 P5MOD0
0000 0000		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

P5CH1MOD<1:0>: PWM5 通道 1 输出配置位

00 = 单输出模式; 默认 P5A 配置为 PWM 输出, P5B、P5C 和 P5D 为端口引脚; 可通过 PSTRCTL 寄存器中的 STREN<A:D>各位置 1, 分别将 P5A、P5B、P5C 和 P5D 配置为 PWM 输出; 通道 1 最多可提供 4 路 PWM 输出

01 = 全桥正向输出模式; P5D 配置为 PWM 调制输出, P5A 为有效电平, P5B 和 P5C 为无效电平

10 = 半桥输出模式; P5A 和 P5B 配置为调制输出; P5C 和 P5D 被分配为端口引脚, 此模式带有死区控制功能

11 = 全桥反向输出模式; P5B 配置为调制输出; P5C 为有效电平; P5A 和 P5D 为无效电平

P5CH3MOD: PWM5 通道 3 输出配置位

0 = 单输出模式, 默认为 P5A3 配置为 PWM 输出, P5B3 为端口引脚; 通过 PSTRCTL1 寄存器中的 STRENA3 和 STRBEN3 两位可以分别设置作为 PWM 引脚或端口引脚。

1 = 半桥输出模式, P5A3 和 P5B3 配置为调制输出; 此模式带死区控制。

P5CH2MOD: PWM5 通道 2 输出配置位

0 = 单输出模式, 默认为 P5A2 配置为 PWM 输出, P5B2 为端口引脚; 通过

PSTRCTL1 寄存器中的 STRAEN2 和 STRBEN2 两位可以分别设置作为 PWM 引脚或端口引脚。

1 = 半桥输出模式, P5A2 和 P5B2 配置为调制输出; 此模式带死区控制。

P5MOD<3:0>: PWM5模块的模式选择位

0000 = 捕捉/ 比较/PWM 关闭 (复位 CCP 模块)

0001 = 未使用 (保留)

0010 = 比较模式, 匹配时输出电平翻转 (CCP5IF 置 1)

0011 = 未使用 (保留)

0100 = 捕捉模式, 在每个下降沿发生捕捉

0101 = 捕捉模式, 在每个上升沿发生捕捉

0110 = 捕捉模式, 每 4 个上升沿发生捕捉

0111 = 捕捉模式, 每 16 个上升沿发生捕捉

1000 = 比较模式, 比较匹配时输出高电平 (CCP5IF 置 1)

1001 = 比较模式, 比较匹配时输出低电平 (CCP5IF 置 1)

1010 = 比较模式, 比较匹配时产生软件中断 (CCP5IF 置 1, P5A 引脚不受影响)

1011 = 比较模式, 触发特殊事件 (CCP5IF 位置 1, CCP 复位 T1, 如果 ADC 使能将启动 AD 转换。)

1100 = PWM 模式; P5A 和 P5C 为高电平有效; P5B 和 P5D 也为高电平有效

1101 = PWM 模式; P5A 和 P5C 为高电平有效; P5B 和 P5D 为低电平有效

1110 = PWM 模式; P5A 和 P5C 为低电平有效; P5B 和 P5D 为高电平有效

1111 = PWM 模式; P5A 和 P5C 为低电平有效; P5B 和 P5D 也为低电平有效

注: 通道 1 中 P5A/P5B 极性控制由 PCA/PCB 和 P5MOD<3:0>同或影响 (见表:10-5 所示), P5C/P5D 由 P5MOD<3:0>控制。

9.5.1.2 PWM5CTL1 寄存器

寄存器: PWM5CTL1: PWM5控制寄存器1(地址: 5BH)

复位值	bit7	P5RSEN	P5DC6	P5DC5	P5DC5	P5DC3	P5DC2	P5DC1	P5DC0	bit0
0000 0000		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

P5RSEN: PWM5 重启使能位

1 = 自动关闭时, 一旦关闭事件消失, P5ASE 位自动清零, PWM5 自动重启

0 = 自动关闭时, P5ASE 由软件清零, 以重启 PWM5

P5DC<6:0>: PWM5 死区延时时间设置位, 用于设置死区延时的时间, 见下式:

式 9.1 PWM5 死区延时时间计算公式

$$\text{延时时间} = \text{P5DC}<6:0> \cdot \text{TINTHF}$$

9.5.1.3 PWM5CTL2 寄存器

寄存器: PWM5CTL2: PWM5更新控制寄存器(地址:11CH)

复位值 --1--000	bit7				bit0			
	-	-	PFUSES	-	-	UDEVT1	UDEVT0	UDEN
	U	U	R/W	U	U	R/W	R/W	R/W

PFUSES: 由 PFUSES 位控制 PWM 输出

0 = 不由配置字控制 PWM 输出所对应的 IO 口

1 = 由配置字中的 PWMPIN、HPOL 和 LPOL 位控制输出所对应的 IO 口

UDEVT1: 更新事件控制 1 (在 UDEN=1 时有效)

0 = 当定时器 T2 为 0 时更新占空比、周期寄存器到缓冲器中

1 = 产生更新事件, 更新占空比、周期、输出控制、极性控制、强制控制寄存器到缓冲器中, 并将定时器和死区定时器清零

UDEVT0: 更新事件控制 0 (在 UDEN=1 时有效)

0 = 当定时器 T2 为 0 时更新输出控制、极性控制、强制控制寄存器

1 = 立即更新输出控制、极性控制、强制控制寄存器到缓冲器中

UDEN: 更新使能

1 = 允许占空比、周期、输出控制、极性控制、强制控制寄存器更新

0 = 禁止占空比、周期、输出控制、极性控制、强制控制寄存器更新

9.5.1.4 PWM5 极性控制寄存器 (PWM5PC)

寄存器: PWM5PC: PWM5极性控制寄存器(地址:11DH)

复位值	bit7	-	-	PCA3	PCB3	PCA2	PCB2	PCA	PCB	bit0
--00 0000		U	U	R/W	R/W	R/W	R/W	R/W	R/W	

PCA3: 当OCA3为0时, 通道3中P5A3端口输出极性控制

- 0 = 输出为高电平有效
1 = 输出为低电平有效

PCB3: 当OCB3为0时, 通道3中P5B3端口输出极性控制

- 0 = 输出为高电平有效
1 = 输出为低电平有效

PCA2: 当OCA2为0时, 通道2中P5A2端口输出极性控制

- 0 = 输出为高电平有效
1 = 输出为低电平有效

PCB2: 当OCB2为0时, 通道2中P5B2端口输出极性控制

- 0 = 输出为高电平有效
1 = 输出为低电平有效

PCA: 当OCA为0时, 通道1中P5A端口输出极性控制

- 0 = 输出为高电平有效
1 = 输出为低电平有效

PCB: 当OCB为0时, 通道1中P5B端口输出极性控制

- 0 = 输出为高电平有效
1 = 输出为低电平有效

注: 通道 1 中 P5A/P5B 极性控制由 PCA/PCB 和 P5MOD<3:0>同或影响 (见表 10-5 所示),
P5C/P5D 由 P5MOD<3:0>控制。

9.5.1.5 PMW5 强制控制寄存器 (PWM5FC)

寄存器: PWM5FC: PWM5强制控制寄存器(地址:119H)

复位值	bit7	-	-	FCA3	FCB3	FCA2	FCB2	FCA	FCB	bit0
--00 0000		U	U	R/W	R/W	R/W	R/W	R/W	R/W	

FCA3: 当OCA3为1时, 通道3中P5A3端口强制输出控制

- 0 = 强制输出为低电平
1 = 强制输出为高电平

FCB3: 当OCB3为1时, 通道3中P5B3端口强制输出控制

- 0 = 强制输出为低电平
1 = 强制输出为高电平

FCA2: 当OCA2为1时, 通道2中P5A2端口强制输出控制

0 = 强制输出为低电平

1 = 强制输出为高电平

FCB2: 当OCB2为1时, 通道2中P5B2端口强制输出控制

0 = 强制输出为低电平

1 = 强制输出为高电平

FCA: 当OCA为1时, 通道1中P5A端口强制输出控制

0 = 强制输出为低电平

1 = 强制输出为高电平

FCB: 当OCB为1时, 通道1中P5B端口强制输出控制

0 = 强制输出为低电平

1 = 强制输出为高电平

OCA3: 通道3中P5A3端口输出控制

0 = PWM输出, 输出极性由PCA3控制

1 = 强制输出, 输出电平由FCA3控制

OCB3: 通道3中P5B2端口输出控制

0 = PWM输出, 输出极性由PCB3控制

1 = 强制输出, 输出电平由FCB3控制

OCA2: 通道2中P5A2端口输出控制

0 = PWM输出, 输出极性由PCA2控制

1 = 强制输出, 输出电平由FCA2控制

OCB2: 通道2中P5B2端口输出控制

0 = PWM输出, 输出极性由PCB2控制

1 = 强制输出, 输出电平由FCB2控制

OCA: 通道1中P5A端口输出控制

0 = PWM输出, 输出极性由PCA控制

1 = 强制输出, 输出电平由FCA控制

OCB: 通道1中P5B端口输出控制

0 = PWM输出, 输出极性由PCB控制

1 = 强制输出, 输出电平由FCB控制

寄存器: PWM5OC: PWM5输出控制寄存器(地址:10EH)

复位值	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
--00 0000	-	-	OCA3	OCB3	OCA2	OCB2	OCA	OCB

U

U

R/W

R/W

R/W

R/W

R/W

R/W

R/W

R/W

OCA3: 通道3中P5A3端口输出控制

0 = PWM输出, 输出极性由PCA3控制

1 = 强制输出, 输出电平由FCA3控制

OCB3: 通道3中P5B2端口输出控制

0 = PWM输出, 输出极性由PCB3控制

1 = 强制输出, 输出电平由FCB3控制

OCA2: 通道2中P5A2端口输出控制

0 = PWM输出, 输出极性由PCA2控制

1 = 强制输出, 输出电平由FCA2控制

OCB2: 通道2中P5B2端口输出控制

0 = PWM输出, 输出极性由PCB2控制

1 = 强制输出, 输出电平由FCB2控制

OCA: 通道1中P5A端口输出控制

0 = PWM输出, 输出极性由PCA控制

1 = 强制输出, 输出电平由FCA控制

OCB: 通道1中P5B端口输出控制

0 = PWM输出, 输出极性由PCB控制

1 = 强制输出, 输出电平由FCB控制

9.5.1.7 PWM5 自动关闭控制寄存器 (P5ASCTL0)

寄存器: P5ASCTL0: PWM5自动关闭控制寄存器0(地址: 5CH)

复位值 0000 0000	bit7								bit0
	P5ASE	P5ASS2	P5ASS1	P5ASS0	P5SSAC1	P5SSAC0	P5SSBD1	P5SSBD0	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

P5ASE: 自动关闭事件状态位

1 = 发生了关闭事件; 四路输出为关闭状态

0 = 四路输出正常工作

P5ASS<2:0>: 自动关闭源选择位

000 = 禁止自动关断

001 = 比较器 C1 输出高电平

100 = INT0 引脚上的逻辑高电平

101 = INT0 引脚上的逻辑高电平或比较器 C1 输出高电平

其它 = 未使用

P5SSAC<1:0>: 引脚 P5A 和 P5C 关闭状态控制位

00 = 驱动引脚 P5A 和 P5C 为 0

01 = 驱动引脚 P5A 和 P5C 为 1

1x = 引脚 P5A 和 P5C 为三态

P5SSBD<1:0>: 引脚 P5B 和 P5D 关闭状态控制位

00 = 驱动引脚 P5B 和 P5D 为 0

01 = 驱动引脚 P5B 和 P5D 为 1

1x = 引脚 P5B 和 P5D 为三态

9.5.1.8 P5ASCTL1 寄存器

寄存器: P5ASCTL1: PWM5自动关闭控制寄存器1(地址: 11AH)

复位值 0000 0000	bit7								bit0
	P5SSA31	P5SSA30	P5SSB31	P5SSB30	P5SSA21	P5SSA20	P5SSB21	P5SSB20	
	R/W								

P5SSA3<1:0>: 引脚 P5A3 关闭状态控制位

00 = 驱动引脚 P5A3 为 0

01 = 驱动引脚 P5A3 为 1

1x = 引脚 P5A3 为三态

P5SSB3<1:0>: 引脚 P5B3 关闭状态控制位

00 = 驱动引脚 P5B3 为 0

01 = 驱动引脚 P5B3 为 1

1x = 引脚 P5B3 为三态

P5SSA2<1:0>: 引脚 P5A2 关闭状态控制位

00 = 驱动引脚 P5A2 为 0

	01 = 驱动引脚 P5A2 为 1
	1x = 引脚 P5A2 为三态
P5SSB2<1:0>:	引脚 P5B2 关闭状态控制位
	00 = 驱动引脚 P5B2 为 0
	01 = 驱动引脚 P5B2 为 1
	1x = 引脚 P5B2 为三态

9.5.1.9 PSTRCTL0 寄存器

寄存器: PSTRCTL0: 脉冲转向控制寄存器(地址: 5DH)

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

STRSYNC:	转向同步位
	1 = 同步控制输出转向更新, 由 UDEN/UDEVT0 控制更新时机
	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
	1 = P5D 引脚输出 PWM 波形, 其极性由 P5MOD<1:0>控制
	0 = P5D 引脚被分配为端口引脚
STRENC:	转向使能位 C
	1 = P5C 引脚输出 PWM 波形, 其极性由 P5MOD<1:0>控制
	0 = P5C 引脚被分配为端口引脚
STRENB:	转向使能位 B
	1 = P5B 引脚输出 PWM 波形, 其极性由 P5MOD<1:0>控制
	0 = P5B 引脚被分配为端口引脚
STRENA:	转向使能位 A
	1 = P5A 引脚输出 PWM 波形, 其极性由 P5MOD<1:0>控制
	0 = P5A 引脚被分配为端口引脚

9.5.1.10 PSTRCTL1 寄存器

寄存器: PSTRCTL1: 脉冲转向控制寄存器2(地址: 11BH)

复位值 ---- 0000	bit7								bit0
	-	-	-	-	STRENA3	STRENB3	STRENA2	STRENB2	
	U	U	U	U	R/W	R/W	R/W	R/W	

STRENA3:	转向使能位 A3
	1 = P5A3 引脚输出 PWM 波形, 其极性由 P5MOD<1:0>控制
	0 = P5A3 引脚被分配为端口引脚
STRENB3:	转向使能位 B3
	1 = P5B3 引脚输出 PWM 波形, 其极性由 P5MOD<1:0>控制
	0 = P5B3 引脚被分配为端口引脚
STRENA2:	转向使能位 A2
	1 = P5A2 引脚输出 PWM 波形, 其极性由 P5MOD<1:0>控制

0 = P5A2 引脚被分配为端口引脚

STRENB2: 转向使能位 B2

1 = P5B2 引脚输出 PWM 波形, 其极性由 P5MOD<1:0>控制

0 = P5B2 引脚被分配为端口引脚

9.5.2 PWM5 的周期、占空比及分辨率

9.5.2.1 PWM5 周期

PWM5 的周期通过 16 位的寄存器<PP5H: PP5L> 进行设置,其值可设置为 0~65535,在边沿对齐模式下和中心对齐模式下,周期的计算方式不同,中心对齐模式下得到的周期是边沿对齐模式下周期寄存器设置值的两倍。PWM5 边沿对齐和中心对齐的周期分别通过式 10.2 和式 10.3 进行计算。

式 9.2 PWM5 边沿对齐模式周期计算公式

$$\text{PWM周期} = (\text{PP5H:PP5L} + 1) \cdot T_{\text{INTHF}} \cdot (\text{T2预分频比})$$

式 9.3 PWM5 中心对齐模式周期计算公式

$$\text{PWM周期} = (\text{PP5H:PP5L}) \cdot T_{\text{INTHF}} \cdot (\text{T2预分频比}) \cdot 2$$

注: T_{INTHF} = 内部高频振荡器时钟周期;

T_2 预分频比通过 T_2 分频器 1 的配置位 $T2CKPS<1:0>$ 位配置。

周期寄存器为带缓冲模式,当 UDEN 位为 0 时,禁止更新占空比寄存器和周期寄存器。当 UDEN 为 1 时,允许更新占空比寄存器和周期寄存器,此时若 UDEVT1 为 0,则周期可以在 T_2 为 0 时更新到周期缓冲器中;若 UDEVT1 为 1 时,则立即更新周期到周期缓冲器中, T_2 被清零。下一次立即更新前,需要软件清零 UDEVT1,再置 1 才能开启新一次的立即更新。

注: (1) 当 T_2 启动时周期寄存器立即更新;

(2) 当周期寄存器的值更新到周期缓冲器后,才真正完成了周期寄存器的设置。

9.5.2.2 PWM5 占空比

PWM5 有 3 个占空比寄存器,占空比设置寄存器全为 16 位,通过寄存器<PWM5H0:PWM5L0>、<PWM5H2:PWM5L2>和<PWM5H3:PWM5L3>(为后面描述方便,统称为<PWM5Hx:PWM5Lx>)进行设置, PWM5Hx 为占空比的高 8 位, PWM5Lx 为低 8 位。在不同模式下占空比计算方式不一致。在边沿对齐模式下,脉冲宽度和占空比通过式 10.4 和式 10.5 计算;在中心对齐模式下,脉冲宽度和占空比通过式 10.6 和式 10.7 计算。

式 9.4 PWM5 边沿对齐模式脉冲宽度计算公式

$$\text{脉冲宽度} = (\text{PWM5Hx:PWM5Lx}) \cdot T_{\text{INTHF}} \cdot (\text{T2预分频比})$$

式 9.5 PWM5 边沿对齐模式占空比计算公式

$$\text{占空比} = \frac{\text{脉冲宽度}}{\text{PWM周期}} = \frac{\text{PWM5Hx:PWM5Lx}}{\text{PP5+1}}$$

式 9.6 PWM5 中心对齐模式脉冲宽度计算公式

$$\text{脉冲宽度} = (\text{PWM5Hx:PWM5Lx}) \cdot T_{INTHF} \cdot (T2\text{预分频比}) \cdot 2$$

式 9.7 PWM5 中心对齐模式占空比计算公式

$$\text{占空比} = \frac{\text{脉冲宽度}}{\text{PWM周期}} = \frac{\text{PWM5Hx:PWM5Lx}}{\text{PP5}}$$

由上述公式可知，占空比为 0% 和 100% 满足条件如下：

- 边沿对齐模式：
 - 0%：占空比寄存器为 0
 - 100%：占空比寄存器为大于等于 (PP5+1) (其中 PP5 为周期寄存器)
- 中心对齐模式：
 - 0%：占空比寄存器为 0
 - 100%：占空比寄存器为大于等于 PP5 (其中 PP5 为周期寄存器)
- 在半桥模式下，占空比为 0% 或者 100% 时，只有在边界 PWM 变化时插入死区，否则不插入死区。

占空比寄存器为带缓冲模式，通过 UDEN 位可以使能或禁止占空比和周期寄存器的更新。当 UDEN 位为 0 时，禁止更新占空比寄存器和周期寄存器。当 UDEN 位置 1 时，若 UDEVT1 为 0，则占空比可以在 T2 为 0 时更新到缓冲器中；若 UDEVT1 为 1 时，则立即更新占空比到缓冲器中，T2 被清零。下一次立即更新前，需要软件清零 UDEVT1，再置 1 才能开启新一次的立即更新。

注： (1) 当 T2 启动时占空比寄存器立即更新；
(2) 占空比寄存器的值更新到占空比缓冲器后，才真正完成了占空比寄存器的设置。

9.5.2.3 PWM5 分辨率

当 PP5 为 65535 时，PWM5 的最大分辨率为 16 位。分辨率的计算公式如式 10.8 所示。

式 9.8 PWM5 分辨率计算公式

$$\text{分辨率} = \frac{\text{Log}[4(<\text{PP5H:PP5L}>+1)]}{\log 2} \text{ 位}$$

9.5.2.4 PWM5 中断

当 PWM5 调制输出满一个周期时, 将中断标志位 T2IF 置 1, 如果 PWM5 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。如果 IPEN 和 PT2 位均置 1, 则为高优先级中断。

9.5.3 边沿对齐 PWM 信号

当 T2CTL1 控制寄存器中的 T2MOD<1:0>计数模式选择位设置成 00 时, 产生的 PWM 信号为边沿对齐的 PWM 信号。在该模式下, PWM 信号的周期由<PP5H:PP5L>周期寄存器决定, 占空比由 16 位的寄存器<PWM5Hx:PWM5Lx>决定 (其中 x=0,2,3)。在周期开始时 (即计数器 T2 等于 0 时) PWM 驱动为高电平, 当计数器 T2 与<PP5Hx:PP5Lx>占空比寄存器发生匹配时, PWM 输出低电平。

如果占空比寄存器设置成 0, 那么在整个 PWM 周期都输出低电平。如果占空比寄存器设置大于<PP5H:PP5L>周期寄存器的值, 那么在整个 PWM 周期都输出高电平。

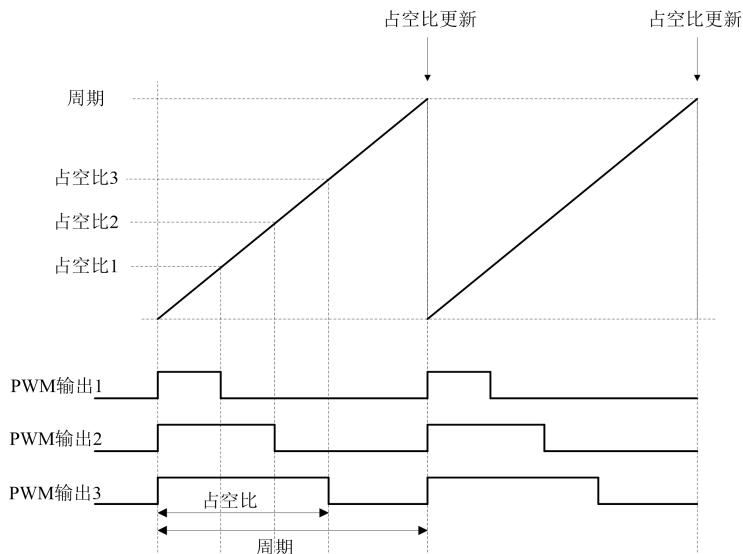


图 9.5 边沿对齐 PWM

9.5.4 中心对齐 PWM 信号

当 T2CTL1 控制寄存器中的 T2MOD<1:0>计数模式选择位设置成 01,10,11 时, 产生的 PWM 信号为中心对齐的 PWM 信号。在该模式下, PWM 信号的周期由<PP5H:PP5L>周期

寄存器决定, 此时, 周期寄存器中的值表示的是周期的一半。占空比由 16 位的寄存器 <PWM5Hx:PWM5Lx> 决定 (其中 $x=0,2,3$)。

在周期开始时, 计数器 T2 向上计数, 当占空比与 T2 寄存器匹配时, PWM 输出为低电平; 当计数器 T2 的值与周期寄存器值的一半匹配时, (应该是周期的一半, 或者与周期寄存器的值匹配时) 计数器 T2 开始向下计数, 而后当占空比与 T2 寄存器匹配时, PWM 输出为高电平。

如果占空比值为 0, 则在整个 PWM 输出均为低电平, 若占空比值比周期寄存器的值大或者等于周期, 则在整个 PWM 输出均为高电平。

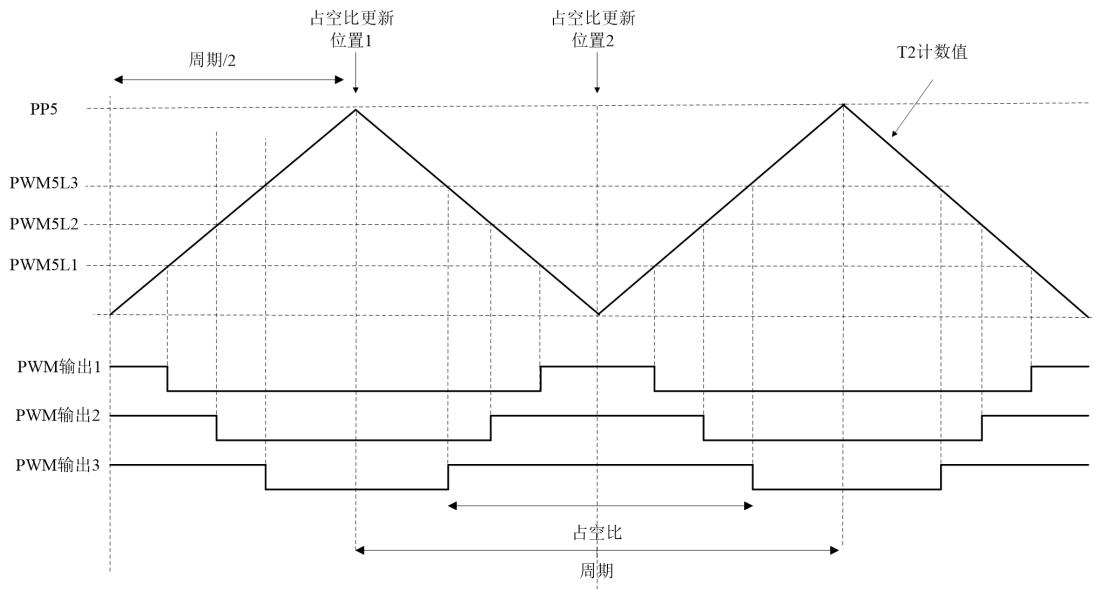


图 9.6 中心对齐 PWM

9.5.5 PWM 信号产生和中断

通过设置 T2CTL1 控制寄存器中的 T2MOD<1:0>可以选择 T2 的计数方式, 从而产生边沿对齐和中心对齐的 PWM 信号。当 T2 向上计数时, 产生边沿对齐的 PWM 信号; 当 T2 向上向下计数时, 产生中心对齐的 PWM 信号。

通过设置 T2CTL1 寄存器中的 T2MOD<1:0>位控制选择溢出中断, 则可以选择在不同的时机触发 PWM 中断操作。

当 T2MOD<1:0>=00 或者 01 时, 是上溢时产生中断标志, 就是在计数器 T2 向上计数到周期值的时候产生一个中断信号。

当 T2MOD<1:0>=10 时, 是下溢产生中断标志, 也就是计数器 T2 向下递减为 0 的时候产生一个中断信号。

当 T2MOD<1:0>=11 时, 是上溢-下溢中断, 在这种状态下在<PP5H:PP5L>和计数器 T2 相等以及计数器 T2 向下计数到 0 时都产生中断信号。

在向上计数模式时, 使用上溢中断, 在向上-向下计数模式时可以使用 3 种中断产生方式。

9.5.6 单输出模式

通过将寄存器 PWM5CTL0 中的 P5CH1MOD<1:0>位设置为 00, 选择通道 1 为单输出模式, 在此模式下, 默认从 P5A 引脚输出 PWM 信号, P5B、P5C 和 P5D 引脚为通用端口引脚。将 PWM5CTL0 中的 P5CH3MOD 和 P5CH3MOD 位均设置为 0, 选择通道 2 和通道 3 为单输出模式, 在此模式下, 默认从 P5A2 和 P5A3 引脚输出 PWM 信号, P5B2 和 P5B3 引脚为通用端口引脚。在使能相应的 PWM5 引脚时, 应将对应的 TRxx 位清零, 以将此引脚设置为输出模式。

可通过设置寄存器 PSTRCTL0/1, 使能或禁止 PWM5 的 8 路输出, PWM5 使用单输出模式时, 最多可同时在 8 个引脚输出 PWM 信号。当 8 个引脚同时输出 PWM 信号时, 通道 1 的四个引脚 P5A,P5B,P5C,P5D 输出的 PWM 信号完全相同; 通道 2 的两个引脚 P5A2,P5B2 输出的 PWM 信号完全相同; 通道 3 的两个引脚 P5A3, P5B3 输出的 PWM 信号完全相同。

通道 1,2,3 有各自独立的占空比寄存器故可输出互不相同的 PWM 信号。如下图所示, 在单输出模式下, 只将 P5A 一路设置为 PWM 输出和将 P5A 和 P5B 两路同时设置为 PWM 输出的示例, 其它设置情况与此类似。

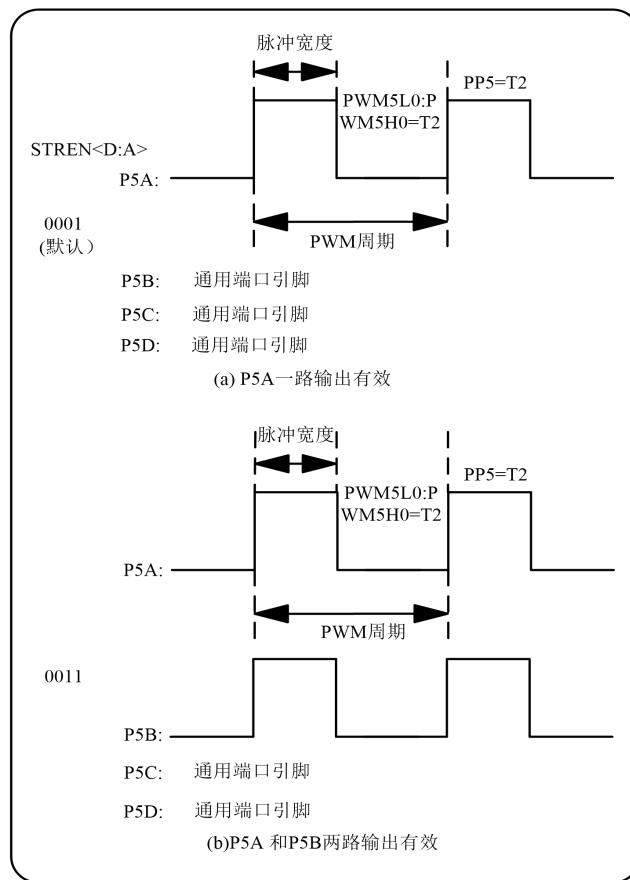


图 9.7 PWM5 的输出示例

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 PSTRCTL0 中的 STRSYNC 位进行设置引脚输出切换时是否与指令同步。如下图所示, STRSYNC=1 时, 脉冲转向控制为同步更新模式。在该模式下, 对应引脚的脉冲转向控制受 UDEN/UDEVT0 控制, 可以实现同步更新, 只有在

UDEN 使能时才能使能所配置的脉冲转向设置，而 UDEVT0 则选择是脉冲转向控制是与周期同步还是立即有效，T2 使能时会产生一次更新操作。STRSYNC=0 时，对应引脚 P5x 输出的 PWM 信号在 STRENx 置一后立即输出；

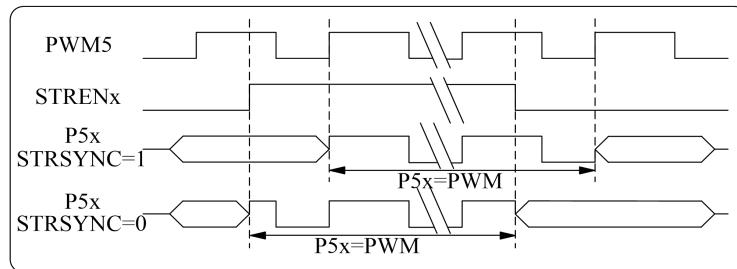


图 9.8 单输出模式 PWM 输出引脚切换

9.5.7 半桥输出模式

通过将寄存器 PWM5CTL0 的 P5CH1MOD<1:0>位设置为 10，可将通道 1 设置为半桥输出模式；将 P5CH3MOD 和 P5CH2MOD 置 1，可将通道 2 和 3 设置为半桥输出模式。在此模式下，通道 1 的 P5A 和 P5B 被配置为调制输出，来驱动推挽式负载，P5C 和 P5D 被配置为通用端口。PWM 输出信号在 P5A 引脚上输出，而互补的 PWM 输出信号在 P5B 引脚上输出，如下图所示，通道 2、3 与此类似。

通道 2 和 3 的在半桥模式下与通道 1 相似，P5A2/P5B2，P5A3/P5B3 被配置为调制输出，PWM 输出信号在 P5A2/3 引脚上输出，而互补的 PWM 输出信号在 P5B2/3 引脚上输出。

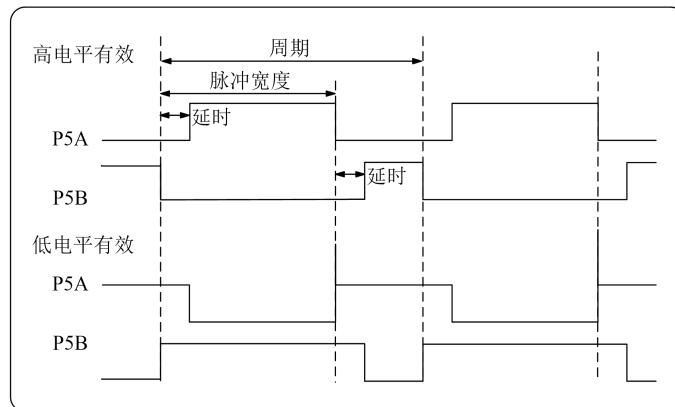


图 9.9 半桥输出模式输出信号示例

半桥输出模式可用于控制半桥和全桥控制电路，如下图所示，为半桥输出模式应用于半桥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。

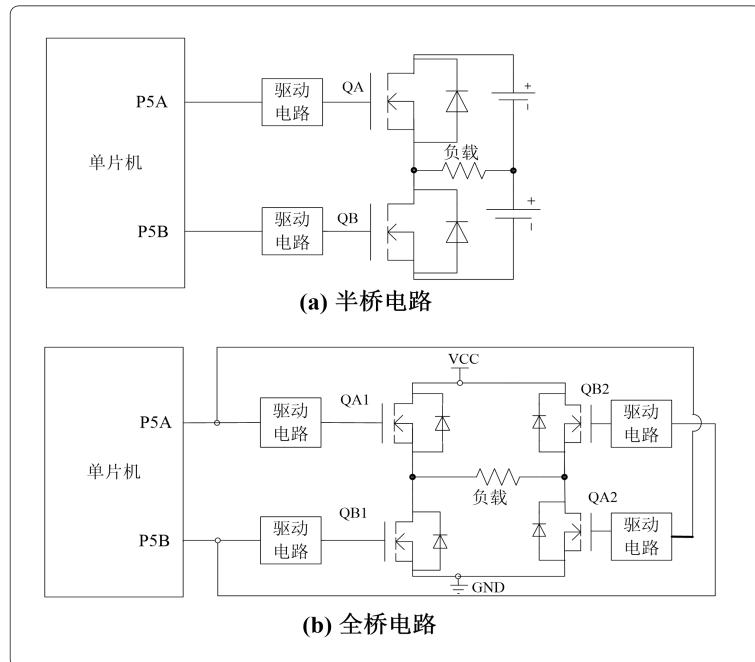


图 9.10 半桥输出模式应用举例

在使用半桥输出模式时，需将 P5A 和 P5B 引脚对应的方向控制位 TRxx 清零，设置为输出。

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。PWM5CTL1 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。

9.5.7.1死区延时

如图 10.10(a)、10.11 所示，在半桥输出模式应用中，P5A 和 P5B 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 两个管子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 10.11(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。3 个通道共用一个死区设置。如图 10.11 (b) 所示，该延时在 PWM5 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 PWM5CTL1 的低 7 位进行设置。延时时间计数公式如式 10.9 所示。

式 9.9 PWM5 死区延时时间计算公式 (同式 10.1)

$$\text{延时时间} = \text{P5DC}<6:0> \cdot T_{INTHF}$$

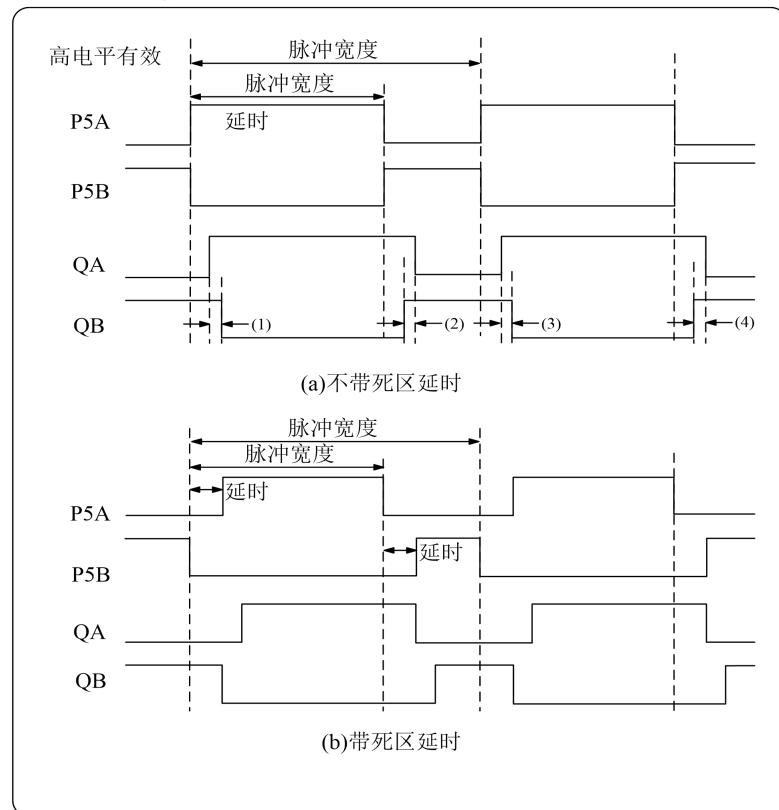


图 9.11 带死区控制和不带死区控制时的信号示例

9.5.8全桥输出模式

通道 1 还可以实现全桥输出模式。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 PWM5CTL0 的 P5CH1MOD<1:0>设置为 01，把 PWM5 设置为全桥正向输出模式；将其设置为 11，则把 PWM5 设置为全桥反向输出模式。

在全桥输出模式下，P5A、P5B、P5C 和 P5D 四个引脚都用作输出。将其设置为全桥正向模式时，引脚 P5A 被设置为有效电平，引脚 P5D 为 PWM 调制信号，P5B 和 P5C 为无效电平，图 10.12(a)和 10.13(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，P5C 被驱动为有效电平，引脚 P5B 为 PWM 调制信号，而 P5A 和 P5D 为无效电平，图 10.12(b)和 10.13(b)为全桥反向输出模式引脚信号示例。图 10.14 给出了全桥输出模式的应用电路示例。

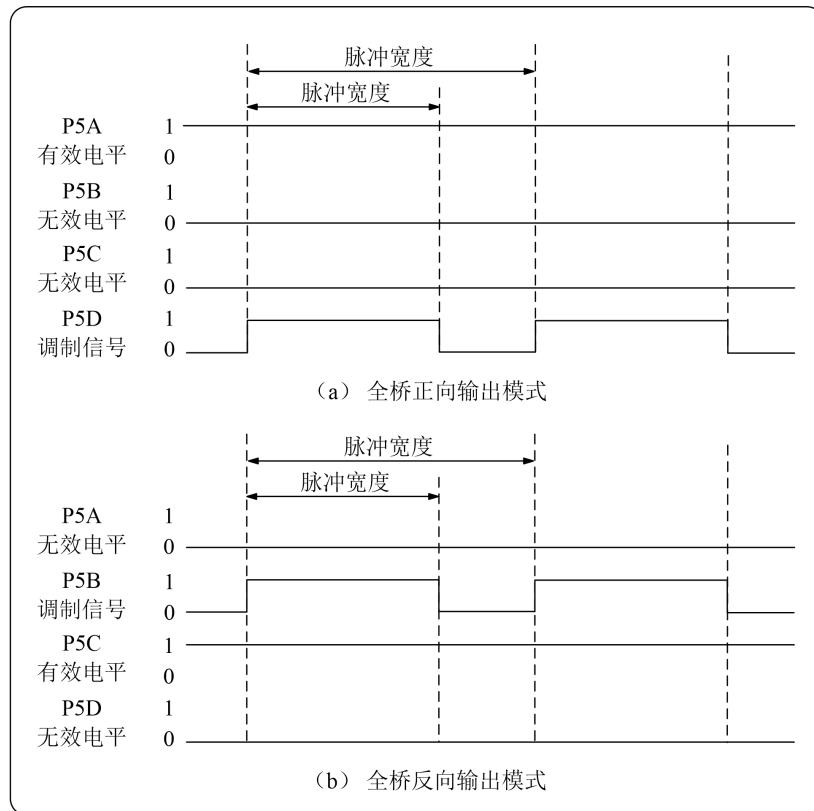


图 9.12 全桥输出模式引脚信号示例(高电平有效)

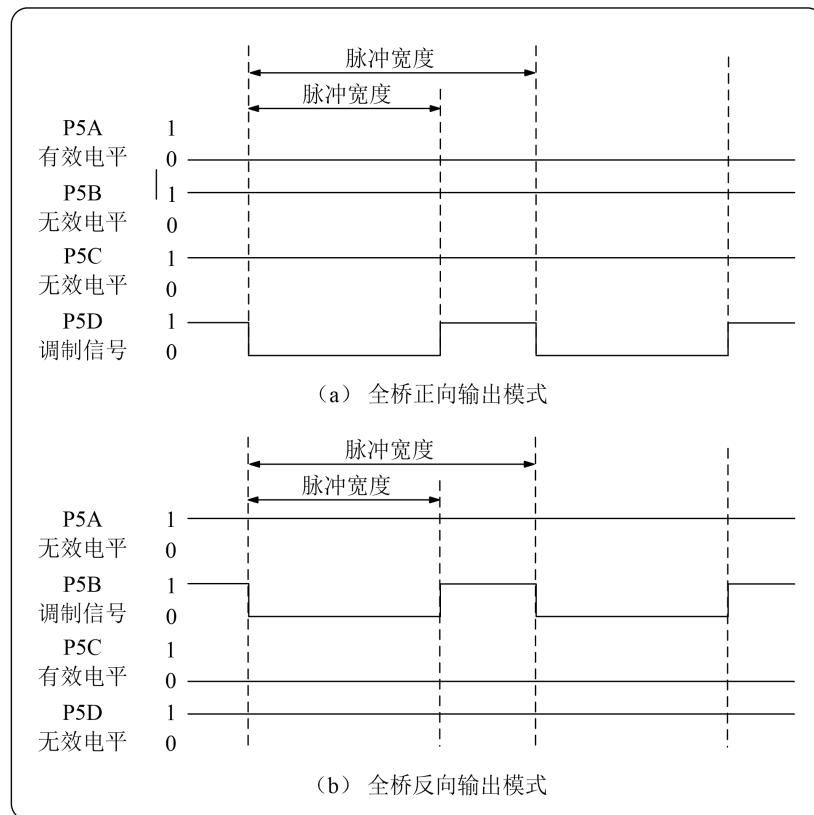


图 9.13 全桥输出模式引脚信号示例(低电平有效)

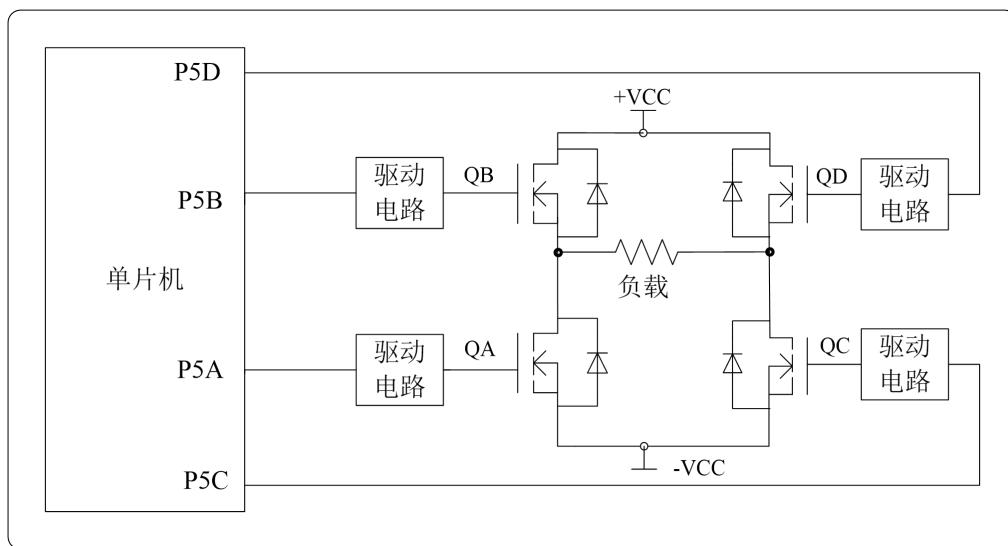


图 9.14 全桥输出模式应用电路示例

在使用全桥输出模式时，需将 P5A、P5B、P5C 和 P5D 引脚对应的方向控制位 TRxx 清零，设置为输出。

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 PWM5CTL0 的 P5CHMOD<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 10.15 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，P5B(或 P5D)被切换到无效状态，P5A(或 P5C)被切换到相反的状态。图中时间 $T = 1/Fosc \bullet T2$ 预分频值。

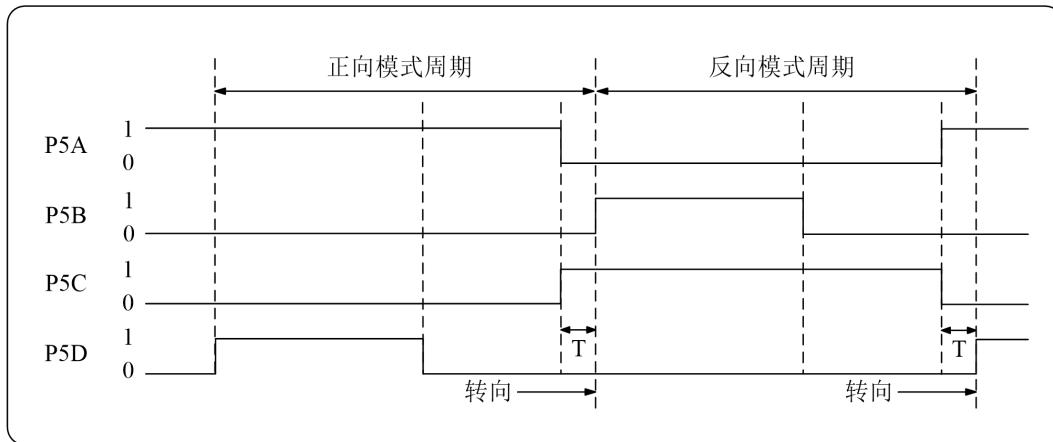


图 9.15 全桥正向、反向输出模式转换示例(高电平有效)

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 10.16 为此情况下各处信号示例。

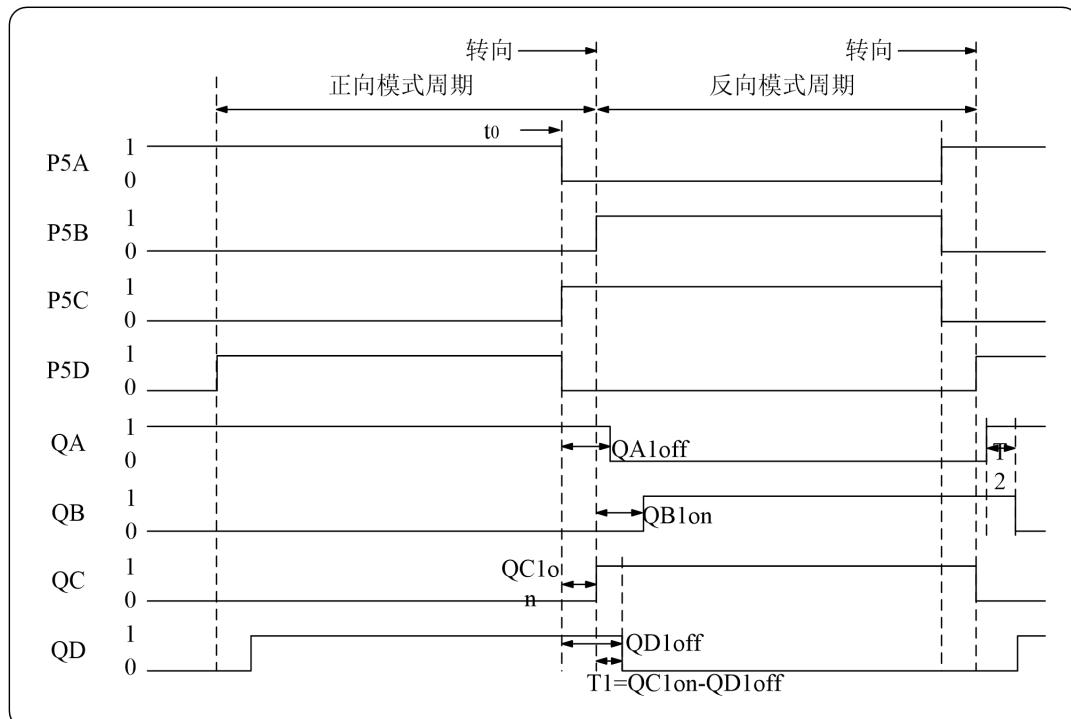


图 9.16 接近满占空比时换向时信号示例

图 10.16 中在 t_0 时刻 P5A 和 P5D 变为无效，P5C 为有效。QA、QB、QC 和 QD 分别为四路 PWM5 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。

9.5.9 脉冲转向控制

通过寄存器脉冲转向寄存器 PSTRCTL0/PSTRCTL1 可将相应的引脚设置成 PWM5 输出引脚或者通用 IO 引脚。通过寄存器 PSTRCTL0 中的 STRSYNC 位可以控制脉冲转向功能切换是与同步模式还是立即有效（与指令周期同步）。通过 STRENA/ STRENB/ STRENC/ STREND/ STRENA2/ STRENB2/ STRENA3/ STRENB3 位可以设置相应端口引脚功能切换。默认情况下，通道 1/2/3 的 Ax 输出为 PWM5 引脚输出，Bx(C/D) 为 IO 口引脚。

注意，通道 1 的脉冲转向控制只在单输出模式下有效，而通道 2/3 的在任何模式下都有效，当使用 PWM5 输出若不需要太多端口时，可自由屏蔽通道 2/3 的输出而不影响 IO 口功能。

当 STRSYNC 为 0 时，对应引脚的输出在 STREN_x 置 1 后立即更新；当 STRSYNC 为 1 时，对应引脚的输出由 UDEN/UDEVT0 控制。只有当 UDEN 为 1 时，对应引脚的 STREN_x 的设置才能生效。当 UDEN 为 1 时，通过 UDEVT0 可以控制同步更新的时机，当 UDEVT0 为 0 时，对应引脚的输出与 PWM 周期同步更新；当 UDEVT0 为 1 时，对应引脚的输出立即更新。通过 UDEN/UDEVT0 可以实现脉冲转向的同步更新操作。

9.5.10 输出控制

在半桥模式下，可以通过 PWM5OC 输出控制寄存器控制引脚的输出。当 PWM5OC 寄存器中的 OC_x 设置为 0 时，相应的引脚选择 PWM 输出，输出极性由 PWM5PC 极性控制寄存器控制；当 OC_x 设置为 1 时，相应的引脚选择强制信号输出，输出电平由 PWM5FC 强制控制寄存器控制。

输出控制为带缓冲模式，通过 PWM5CTL2 更新控制寄存器可以控制输出控制寄存器缓冲的更新。详见 PWM 更新锁定。

9.5.11 强制输出模式

通过将 PWM5OC 输出控制寄存器中相应输出引脚设置成 1 可以选择 P5A/ P5B/ P5A2/ P5B2/ P5A3/ P5B3 输出强制信号。在该模式下，通过 PWM5FC 强制控制寄存器可以设置 P5A/ P5B/ P5A2/ P5B2/ P5A3/ P5B3 端口输出高电平或低电平，强制输出不受极性控制的影响。

强制输出控制为带缓冲模式，通过 PWM5CTL2 更新控制寄存器可以控制强制输出的更新。详见 PWM 更新锁定。

9.5.12 PWM 输出极性模式

通过将 PWM5OC 输出控制寄存器中相应输出引脚设置成 0 可以选择 P5A/ P5B/ P5A2/ P5B2/ P5A3/ P5B3 输出 PWM 信号。在该模式下，通过 PWM5PC 极性控制寄存器可以设置 P5A/ P5B/ P5A2/ P5B2/ P5A3/ P5B3 端口输出的极性。

当 PWM 输出极性选择高有效时，P5A/ P5A2/ P5A3 输出在周期开始输出为高电平，当

占空比与定时器匹配时，输出翻转；而互补的输出 P5B/ P5B2/ P5B3 则在周期开始输出为低电平，当占空比与定时器匹配时，输出翻转。

需要注意的是，通道 1 的 P5A/P5B 有两个极性控制使能，一个是 PWM5CTL0 中的 P5MOD<3:0>，一个是 PWM5PC 中的 PCA/PCB。这两个寄存器相应的控制位共同影响产生最后的极性控制结果。如前述表 9-5 所示。而 P5C/P5D 则只由 P5MOD<3:0>控制。P5A/ P5B/ P5C/ P5B 的极性控制需要在 UDEN 为 1 时才允许更新。

PWM 输出极性控制为带缓冲模式，通过 PWM5CTL2 更新控制寄存器可以控制强制输出的更新。PWM 极性更新有两种形式，一种是立即更新，一种是定时器为 0 时更新。详见 PWM 更新锁定。

9.5.13 PWM 更新锁定

通过设置 PWM5CTL2 更新控制寄存器的 UDEN 可以使能或禁止占空比、周期、极性控制、强制控制、输出控制和脉冲转向控制的更新。当 PWM5CTL2 寄存器的 UDEN 为 0 时，占空比、周期、极性控制、强制控制和输出控制不能更新，这使得用户能够在新值生效前将所需要的值写入到相应的寄存器中。当 UDEN 为 1 时，可以通过 UDEVT1/0 来控制占空比、周期、极性控制、强制控制和输出控制的更新到缓冲中。UDEVT1 控制周期和占空比的更新，UDEVT0 控制输出控制、极性控制和强制控制寄存器的更新。若 UDEVT1 为 0，则周期和占空比可以在 T2 为 0 时更新到缓冲中；若 UDEVT1 为 1 时，则立即更新周期和占空比到缓冲中，T2 被清。若 UDEVT0 为 0，则输出控制、极性控制和强制控制寄存器可以在定时器为 0 时更新到缓冲中；若 UDEVT0 为 1 时，则立即更新输出控制、极性控制和强制控制寄存器到缓冲中，此时 T2 继续计数，不影响占空比的完整性。

脉冲转向控制中，当 STRSYNC=1 时，脉冲转向更新与极性控制、强制控制、输出控制一致，都是由 UDEVT0 控制，且 UDEN 为更新总使能。当 STRSYNC=0 时，脉冲转向与 UDEN/UDEVT0 无关，当脉冲转向控制 STRENx 设置后输出立即改变。

注意，当 UDEVT1 为 1 时，会清零定时器，并清零死区定时器，此时若 UDEVT0 为 0，也会引起极性控制、强制控制、输出控制和脉冲转向控制的更新。UDEVT0/1 置 1 后必须软件清零再置 1 才能使能下一次的立即更新控制。

9.5.14 PWM 复位控制

PWM5 的复位控制使得在复位状态下，可以通过将配置位中的 PWMPIN 设置成 I/O 口控制或是选择引脚的有效极性状态。引脚的极性状态可以通过控制配置位中的 HPOL 和 LPOL 输出所需要的极性。配置 HPOL 可以设置 P5A、P5C、P5A2/3 的极性，配置 LPOL 可以设置 P5B、P5D、P5B2/3 的极性。

9.5.15 自动关断和自动重启模式

9.5.15.1 自动关断模式

PWM5 模块具有自动关断功能。如图 10.17 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 P5A、P5B、P5C、P5D、P5A2、P5B2、P5A3、P5B3 八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 3 个关断源：

- INT0 引脚的逻辑高电平；
- 比较器 1 输出高电平；
- 在软件中直接将 P5ASE 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。

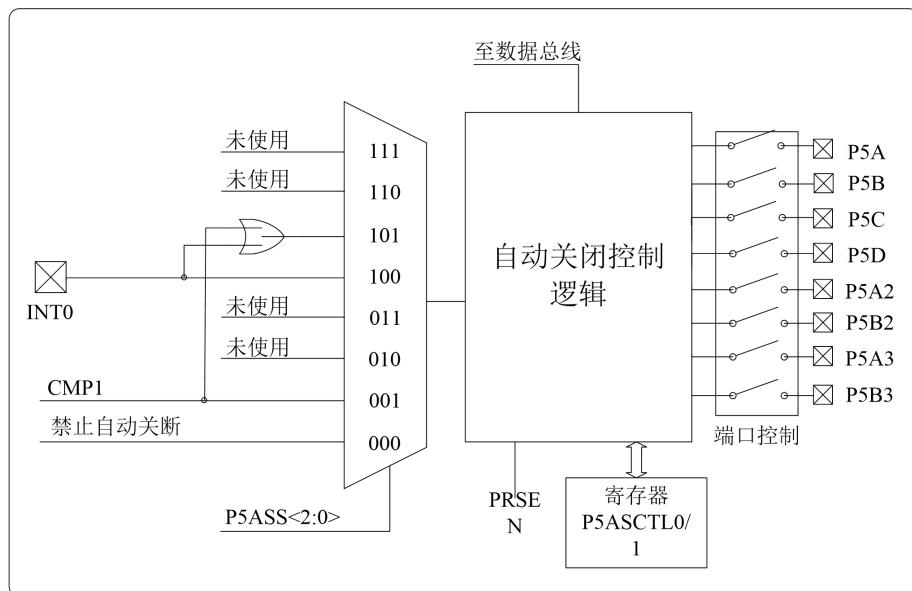


图 9.17 自动关断模式原理框图

通过寄存器 P5ASCTL0 的 P5ASS<2:0>位选择自动关断源。将 P5ASS<2:0>位设置为 000 时，关闭自动关断功能。

寄存器 P5ASCTL0 中的 P5ASE 位指示关断的状态。如果该位为 0，表示 PWM5 的八个引脚输出正常的 PWM 信号，如果该位为 1，表示 PWM 的八路输出处于关断状态。

发生关断事件时，将会：

- ① P5ASE 位被置 1。直到被软件清零或发生自动重启才会将该位清零。
- ② 使能的八个 PWM 引脚将被置于关断电平状态。

关断时，通道 1 四路输出电平的状态由寄存器 P5ASCTL0 的 P5SSAC<1:0>和 P5SSBD<1:0>位决定。通道 2 和通道 3 输出电平状态由寄存器 P5ASCTL1 的 P5SSA2<1:0>、P5SSB2<1:0>、P5SSA3<1:0>和 P5SSB3<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 P5A 和 P5C 的状态由 P5SSAC<1:0>设置，P5B 和 P5D 的状

态由 P5SSBD<1:0>设置, P5A2 状态由 P5SSA2<1:0>设置, P5B2 状态由 P5SSB2<1:0>设置, P5A3 状态由 P5SSA3<1:0>设置, P5B3 状态由 P5SSB3<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 P5ASE 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, P5ASE 仍然为 1(如果 P5RSEN=0), 四路输出仍然为关断模式电平, 直到将 P5ASE 位清零, PWM 重启。

9.5.15.2 自动重启模式

可将 PWM5 配置为一旦清除自动关断条件就自动重启 PWM。通过将 PWM5CTL1 寄存器中的 P5RSEN 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, P5ASE 位就将保持置 1。当清除自动关断条件时, 将通过硬件将 P5ASE 位清零, 并且将恢复常规操作。

如图 10.18 所示, 在图(a)中, P5RSEN=0, 自动重启模式关闭, 通过软件将 P5ASE 标志位清零, PWM 才会重启。在图(b)中, P5RSEN=1, 自动重启模式被打开, 关断事件被清除后, P5ASE 标志位由硬件自动清零, 然后重启 PWM。

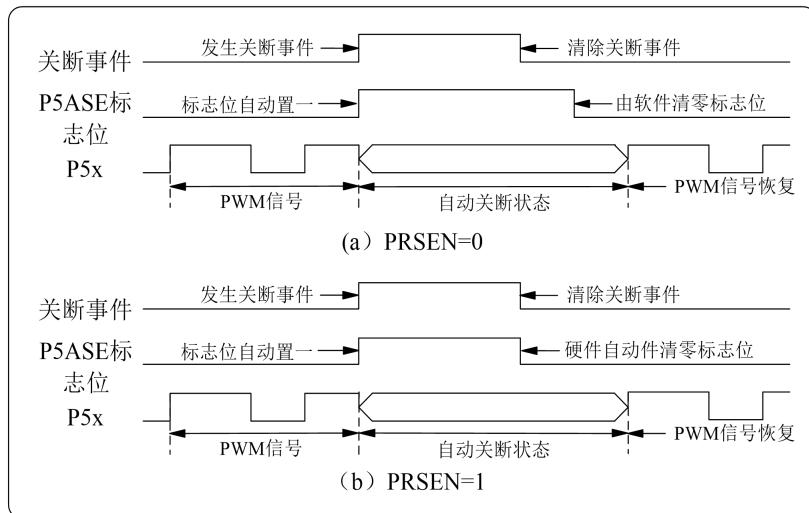


图 9.18 自动重启和软件重启 PWM 示例

10 SSCI 模块

10.1 概述

KF8F513X 包含一个 SSCI (Synchronous Serial Communication interface) 同步串行端口。它是用于其他外设或单片机进行通信的串行接口。SSCI 包含两种工作模式：

- ◆ I2C (Inter Integrated Circuit) 接口模式。
- ◆ 串行外设接口 (Serial Peripheral Interface, SPI) 模式

10.2 SSCI 引脚复用

SSCI 模块相关功能引脚 SDI/SDA、SCL/SCK、SDO 和 SS，提供多个 I/O 口可选，可以通过 APFCTL6 和 APFCTL7 进行配置。

表 10-1 SSCI 引脚复用对应表

SSCI 相关功能引脚	对应 I/O 口	备注
SDA/SDI	P0.1 / P1.1 / P1.3 / P1.4 / P2.2 / P2.3 / P3.4 / P3.7 / P4.0 / P4.6	通过 APFCTL7 寄存器的 SDISEL<3:0>位进行配置
SCL/SCK	P0.0 / P1.0 / P1.5 / P2.1 / P3.2 / P4.0 / P4.2	通过 APFCTL6 寄存器的 SCKSEL<2:0>位进行配置
SDO	P1.7 / P2.0 / P3.3 / P3.5 / P3.7 / P4.5	通过 APFCTL6 寄存器的 SDOSEL<2:0>位进行配置
SS	P0.5 / P2.3	通过 APFCTL7 寄存器的 SSSEL<3:0>位进行配置

10.3 SSCI 相关寄存器

表 10-2 与 SSCI 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
128H	SSCICL0	SSCIWCFL	SSCIOV	SSCIEN	SSCICKP	SSCI MOD3	SSCI MOD2	SSCI MOD1	SSCI MOD0
12AH	SSCICL1	SSCI CALLEN	SSCI ACKSTA	SSCI ACKDAT	SSCI ACKEN	SSCI RCEN	STOPEN	RESTART EN	STARTEN
12BH	SSCISTA	SAMPLE	CKEGE	SSCIDA	SSCISTOP	SSCISTART	SSCIRW	SSCIUA	SSCIBUF
12CH	SSCIBUFR	SSCI 数据接收缓冲/发送寄存器							
12EH	SSCIADD	SSCI 的 I2C 地址寄存器							
	SSCIMSK	SSCIMSK7	SSCIMSK6	SSCIMSK5	SSCIMSK4	SSCIMSK3	SSCIMSK2	SSCIMSK1	SSCIMSK0
266H	APFCTL6	-	SDOSEL2	SDOSEL1	SDOSEL0	-	SCKSEL2	SCKSEL1	SCKSEL0
267H	APFCTL7	-	-	-	SSSEL	SDISEL3	SDISEL2	SDISEL1	SDISEL0

10.3.1 SSCI 控制寄存器 0 (SSCICCTL0)

寄存器: SSCICCTL0: SSCI控制寄存器0(地址:128H)

复位值 0000 0000	bit7								bit0 0
	SSCI WCFL	SSCIOV	SSCIEN	SSCICKP	SSCIMOD 3	SSCIMOD 2	SSCIMOD 1	SSCIMOD 0	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

SSCIWCFL: 写冲突检测位

1 = 正在发送前一个字时, 又有数据写入SSCIBUFR寄存器 (必须用软件清零)

0 = 无冲突

SSCIOV: 接收溢出指示位

在SPI 模式下:

1 = 当SSCIBUFR 中仍保存前一数据时, 又接收到一个新的字节。如果溢出, SSCISR 中的数据会丢失。溢出只会在从动模式下发生。即使只是发送数据, 用户也必须读SSCIBUFR, 以避免将溢出标志位置1。在主控模式下, 溢出位不会被置1, 因为每次接收 (和发送) 新数据都是通过写入SSCIBUFR寄存器启动。

0 = 无溢出

在I2C模式下:

1 = SSCIBUFR中仍保存前一数据时, 又接收到一个新的字节。SSCIOV 在发送模式下被忽略。两种模式下都必须用软件将SSCIOV 清零。

0 = 无溢出

SSCIEN: 同步串行端口使能位

在SPI模式下:

1 = 使能串行端口并将SCK、SDO 和SDI 配置为串行端口引脚

0 = 禁止串行端口并将这些引脚配置为I/O 端口引脚

在I2C模式下:

1 = 使能串行端口并将SDA 和SCL 引脚配置为串行端口引脚

0 = 禁止串行端口并将这些引脚配置为I/O 端口引脚

在两种模式下, 当使能时, 这些引脚必须被正确配置为输入或输出。

SSCICKP: 时钟极性选择位

在SPI 模式下:

1 = 空闲状态时, 时钟为高电平

0 = 空闲状态时, 时钟为低电平

在I2C模式下:SCK 释放控制

1 = 使能时钟

0 = 保持时钟为低电平 (时钟低电平时间延长)。(用于确保数据建立时间。)

SSCIMOD<3:0>: 同步串行端口模式选择位

0000 = SPI 主控模式, 时钟 = 工作时钟/4

0001 = SPI 主控模式, 时钟 = 工作时钟/16

0010 = SPI 主控模式, 时钟 = 工作时钟/64

0011 = SPI 主控模式, 时钟 = T2输出/2

0100 = SPI 从动模式, 时钟 = SCK 引脚。使能 SS 引脚控制。

0101 = SPI 从动模式, 时钟 = SCK 引脚。禁止 SS 引脚控制。SS 可作为 I/O 引脚使用。

0110 = I2C 从动模式, 7 位地址

0111 = I2C 从动模式, 10 位地址

1000 = I2C 主控模式, 时钟 = $F_{OSC}/(4*(SSCIADD+1))$

1001 = 允许 SSCIMSK 寄存器读写操作

1010 = 保留

1011 = I2C 固件控制主控模式 (从动空闲模式)

1100 = 保留

1101 = 保留

1110 = I2C 从动模式, 7 位地址, 并允许启动位和停止位中断

1111 = I2C 从动模式, 10 位地址, 并允许启动位和停止位中断

10.3.2 SSCI 控制寄存器 1 (SSCICCTL1)

寄存器: SSCICCTL1: SSCI 控制寄存器 1 (地址: 12AH)

	bit7	bit0						
复位值	0000 0000							
	SSCICAL LEN	SSCIACK STA	SSCIACK DAT	SSCIACK EN	SSCIRCE N	STOPEN	RESTART EN	STARTEN
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

SSCICALLEN: 广播呼叫使能位 (仅限 I2C 从动模式)

1 = 允许在 SSCISR 中接收到广播呼叫地址 (0000h) 时产生中断

0 = 禁止广播呼叫地址

SSCIACKSTA: 应答状态位 (仅限于 I2C 主控模式)

在主控发送模式下:

1 = 未接收到来自从动器件的应答。

0 = 已接收到来自从动器件的应答

SSCIACKDAT: 应答数据位 (仅限于 I2C 主控模式)

在主控接收模式下: 用户在接收完成后发送的应答序列的值

1 = 不应答

0 = 应答

SSCIACKEN: 应答序列使能位 (仅限 I2C 主控模式)

在主控接收模式下:

1 = 在 SDA 和 SCL 引脚启动应答序列, 发送 ACKDT 数据位。由硬件自动清零。

0 = 应答序列空闲

SSCIRCEN: 接收使能位 (仅限 I2C 主控模式)

1 = 使能 I2C 接收模式

0 = 接收空闲

STOPEN: 停止条件使能位 (仅限 I2C 主控模式)

SCK 释放控制:

1 = 在 SDA 和 SCL 引脚启动停止条件。由硬件自动清零。

0 = 停止条件空闲

RESTARTEN: 重复启动条件使能位 (仅限I2C主控模式)

1 = 在SDA 和SCL 引脚启动重复启动条件。由硬件自动清零。

0 = 重复启动条件空闲

STARTEN: 启动条件使能位 (仅限I2C主控模式)

在主控模式下:

1=在SDA 和SCL引脚启动条件。由硬件自动清零。

0=启动条件空闲

在从动模式下:

1=从发送和接收都会使能时钟延长 (使能时钟延长)

0=禁止时钟延长

注:

对于SSCIACKEN、SSCIRCEN、STOPEN、RESTARTEN 和STARTEN 位:如果I2C模块不处在空闲模式, 此位可能无法被置1 (没有假脱机 (spooling)) 且可能无法对SSCIBUFR 进行写操作 (禁止写SSCIBUFR)。

10.3.3SSCI 状态寄存器 (SSCISTA)

寄存器: SSCISTA: SSCI状态寄存器(地址:12BH)

复位值 0000 0000	bit7								bit0
	SAMPLE	CKEGE	SSCIDA	SSCISTOP	SSCISTA RT	SSCIRW	SSCIUA	SSCIBUF	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

SAMPLE: SPI数据输入采样相位

SPI 主控模式:

1 = 在数据输出时间结束时采样输入数据

0 = 在数据输出时间中间采样输入数据

SPI 从动模式:

当SPI 用于从动模式时, 必须将SAMPLE 清零

I2C模式:

此位必须保持清零

CKEGE: SPI 时钟边沿选择位

SPI模式, SSCICKP = 0:

1 = 在SCK 的下降沿发送数据

0 = 在SCK 的上升沿发送数据

SPI 模式, SSCICKP = 1:

1 = 在SCK 的上升沿发送数据

0 = 在SCK 的下降沿发送数据

I2C模式:

此位必须保持清零

SSCIDA: 数据/ 地址位 (I2C模式)

1 = 表示上次接收或发送的字节是数据

0 = 表示上次接收或发送的字节是地址

SSCISTOP: 停止位 (仅I2C模式)

当禁止SSCI 模块或上次检测到启动位时，该位被清零。

SSCIEN 被清零。

1 = 表示上次检测到了停止位（此位在复位时为0）

0 = 表示上次没有检测到停止位

SSCISTART: 启动位（仅I2C模式）

当禁止SSCI 模块或上次检测到停止位时，该位被清零。

SSCIEN 被清零。

1 = 表示上次检测到了启动位（此位在复位时为0）

0 = 表示上次没有检测到启动位

SSCIRW: 读/写信息位（仅I2C模式）

该位用来保存在上次地址匹配后的SSCIRW 位信息。此位仅在地址匹配与遇到下一个启动位、停止位或SSCIACK 位之间有效。

1 = 读

0 = 写

SSCIUA: 更新地址位（仅10位I2C模式）

1 = 表示用户需要更新SSCIADD 寄存器中的地址

0 = 不需要更新地址

SSCIBUF: 缓冲器满状态位

接收（SPI和I2C模式）：

1 = 接收完成，SSCIBUFR满

0 = 接收未完成，SSCIBUFR空

发送（仅I2C模式）：

1 = 正在发送，SSCIBUFR满

0 = 发送完成，SSCIBUFR空

10.3.4 SSCI 屏蔽寄存器（SSCIMSK）

寄存器: SSCIMSK: SSCI屏蔽寄存器(地址:12EH)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
复位值	SSCIMSK 1111 1111	SSCIMSK 7	SSCIMSK 6	SSCIMSK 5	SSCIMSK 4	SSCIMSK 3	SSCIMSK 2	SSCIMSK 1
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

SSCIMSK<7:1>:屏蔽位

1 = 接收到的地址的bit n 与SSCIADD<n> 比较以检测I²C的地址匹配情况

0 = 接收到的地址的bit n 不用于检测I²C的地址匹配情况

SSCIMSK<0>: 在I²C从动模式下，10位地址的屏蔽位

在I²C 从动模式，10位地址（SSCIMOD<3:0> = 0111或1111）条件下：

1 = 将接收到的地址的bit 0位与SSCIADD<0> 相比较以检测I²C的地址匹配情况

0 = 接收到的地址的bit 0位不用于检测I²C的地址匹配情况

在I²C从动模式，7位地址条件下，该位为无关位

注：SSCIADD与SSCIMSK共用一个地址，当SSCICL0位SSCIMOD<3:0>=1001时，SFR地址对应SSCIMSK寄存器；当SSCICL0位SSCIMOD<3:0>不为1001时，SFR地址对应SSCIADD寄

存器。

10.4 I2C 模式

- ◆ 多主机模式:可用作主设备或者从设备
- ◆ I2C 主设备产生时钟, 起始和停止信号
- ◆ 检测 7 位和 10 位地址

10.4.1 工作原理

I2C模式下的SSCI 能实现全部从动功能（除广播呼叫支持外），且硬件支持启动位和停止位中断，以便于固件实现主控功能。SSCI模式实现标准模式规范以及7位和10位寻址。有两个引脚用于数据传输: SCK/SCL引脚作为时钟线（SCL），而SDI/SDA引脚作为数据线（SDA）。通过将SSCI使能位SSCIEN（SSCICCTL0<5>）置1以使能SSCI模块的功能。

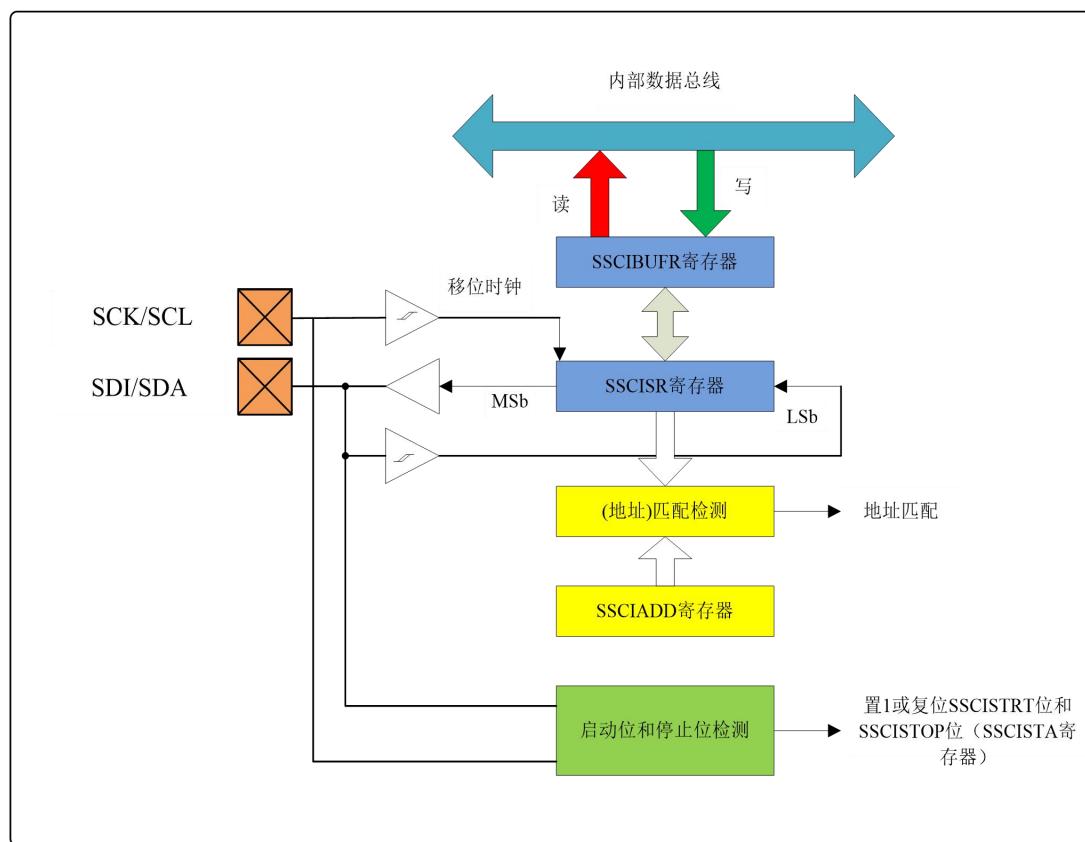


图 10.1 I2C 模式方框图

SSCI模块有7个寄存器用于I2C操作，这7个寄存器是:

- SSCI控制寄存器（SSCICCTL0）
- SSCI控制寄存器1（SSCICCTL1）
- SSCI状态寄存器（SSCISTA）
- 串行接收/发送缓冲器（SSCIBUFR）
- SSCI 移位寄存器（SSCISR）——不可直接访问
- SSCI 地址寄存器（SSCIADD）

- SSCI 屏蔽寄存器 (SSCIMSK)

SCICL0 寄存器用于控制I2C 的工作。可通过设置四个模式选择位 (SSCICL0<3:0>) 选择以下I2C 模式之一：

- I2C 从动模式 (7 位地址)
- I2C 从动模式 (10 位地址)
- I2C 从动模式 (7 位地址)，允许启动位和停止位中断以支持固件主控模式
- I2C 从动模式 (10 位地址)，允许启动位和停止位中断以支持固件主控模式
- 允许I2C 启动位和停止位中断以支持固件主控模式而从动模式空闲

任何I2C 模式的选择，在SSCIEN置1后都会强制SCL和SDA引脚为漏极开路（假定通过编程将相应的IO口方向寄存器TRx位置1，使这些引脚成为输入引脚）。必须在SCL和SDA引脚上外接上拉电阻，才能使I2C模块正常工作。

10.4.2 I2C 从动模式

在从动模式下，SCL 引脚和SDA 引脚必须被配置为输入。必要时SSCI 模块将用输出数据改写输入状态（从发送器）。

当地址匹配或在地址匹配后发送的数据被接收时，硬件会自动产生一个应答（ACK）脉冲，并把当时SSCISR寄存器中接收到的值装入SSCIBUFR寄存器。

某些条件会使SSCI 模块不发出此ACK（低电平有效）脉冲。这些条件包括（之一或全部）：

- 1) 在接收到数据前，缓冲器满标志位SSCIBUF (SSCISTA<0>) 置1。
- 2) 在接收到数据前，溢出标志位SSCIOV (SSCICL0<6>) 置1。

在这些情况下，SSCISR寄存器的值不会载入SSCIBUFR，但是SSCIIF位会置1。下表显示了当已知SSCIBUF位和SSCIOV位的状态时，接收到数据发送字节时产生的结果。阴影单元显示了当用户软件没有正确将溢出状态清零时的情况。当SSCIOV位通过软件清零时，通过读SSCIBUFR寄存器可以将标志位SSCIBUF清零。

表 10-3 接收数据后的动作

接收到传输数据时的状态位		SSCISR 数据存入 SSCIBUFR	产生 ACK 脉冲	SSCIIF 位置 1 (如果允许 SSCI 中断，还将产生 SSCI 中断)
SSCIBUF	SSCIOV			
0	0	有	有	有
1	0	无	无	有
1	1	无	无	有
0	1	无	无	有

注:阴影单元显示了当用户软件没有正确将溢出状态清零时的情况。

10.4.2.1 寻址

一旦SSCI模块被使能，它就会等待启动条件发生。在7位地址模式下，当启动条件发生后，8位数据被移入SSCISR寄存器。在时钟 (SCL) 线的上升沿采样所有的输入位。在第8个时钟 (SCL) 脉冲的下降沿寄存器SSCISR<7:1>的值会和SSCIADD地址寄存器的值比较。如果地址匹配，并且SSCIBUF和SSCIOV都被清零，会发生下列事件：

- 1) SSCISR寄存器的值被装入SSCIBUFR寄存器。
- 2) 缓冲器满标志位SSCIBUF被置1。
- 3) 产生ACK脉冲。
- 4) 在第9个SCL脉冲的下降沿, SSCI中断标志位SSCIIF被置1 (如果允许中断, 则产生中断)。

在10位地址模式下, 从控制器需要收到两个地址字节 (图12.3)。第一个地址字节的高5位将指定这是否是一个10位地址。SSCIRW位 (SSCISTA<2>) 必须指定写操作, 这样从控制器才能接收到第二个地址字节。对于10位地址, 第一个字节等于“1111 0 A9 A8 0”, 其中A9和A8是该地址的两个最高有效位。

10位地址的工作步骤如下, 其中7-9步是针对从动发送器而言的:

- 1) 接收地址的第一个 (高) 字节 (SSCIIF位、SSCIBUF位和SSCIUA位置1)。
- 2) 用地址的第二个 (低) 字节更新SSCIADD寄存器 (SSCIUA位清零并释放SCL线)。
- 3) 读SSCIBUFR寄存器 (SSCIBUF位清零), 并将标志位SSCIIF清零。
- 4) 接收地址的第二个 (低) 字节 (SSCIIF位、SSCIBUF位和SSCIUA位置1)。
- 5) 用地址的第一个 (高) 字节更新SSCIADD寄存器; 如果匹配, 则释放SCL线, 此时将会清零SSCIUA位。
- 6) 读SSCIBUFR寄存器 (SSCIBUF位清零) 并将标志位SSCIIF清零。
- 7) 接收重复启动条件。
- 8) 接收地址的第一个 (高) 字节 (SSCIIF位和SSCIBUF位置1)。
- 9) 读SSCIBUFR寄存器 (SSCIBUF位清零) 并将标志位SSCIIF清零。

10.4.2.2接收

当地址字节的SSCIRW状态位清零并发生地址匹配时, SSCISTA寄存器中的SSCIRW位清零。接收到的地址被装入SSCIBUFR寄存器。

当发生地址字节溢出时, 则不会产生应答脉冲 (ACK)。溢出条件是指SSCIBUF位置1, 或者SSCIOV位 (SSCICL0<6>) 置1。这是一个由于用户固件导致的错误状态。

每个数据传输字节都会产生SSCI中断。标志位SSCIIF必须用软件清零。通过SSCISTA寄存器可以确定该字节的状态。

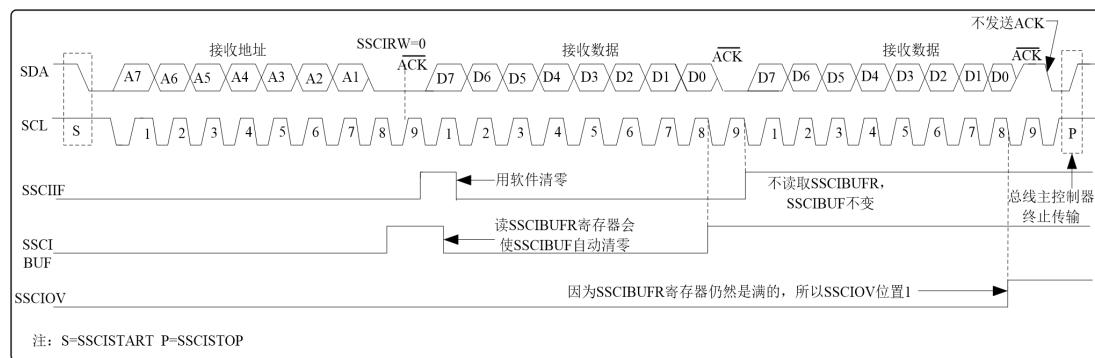


图 10.2 从动模式时序 (接收, 7 位地址)

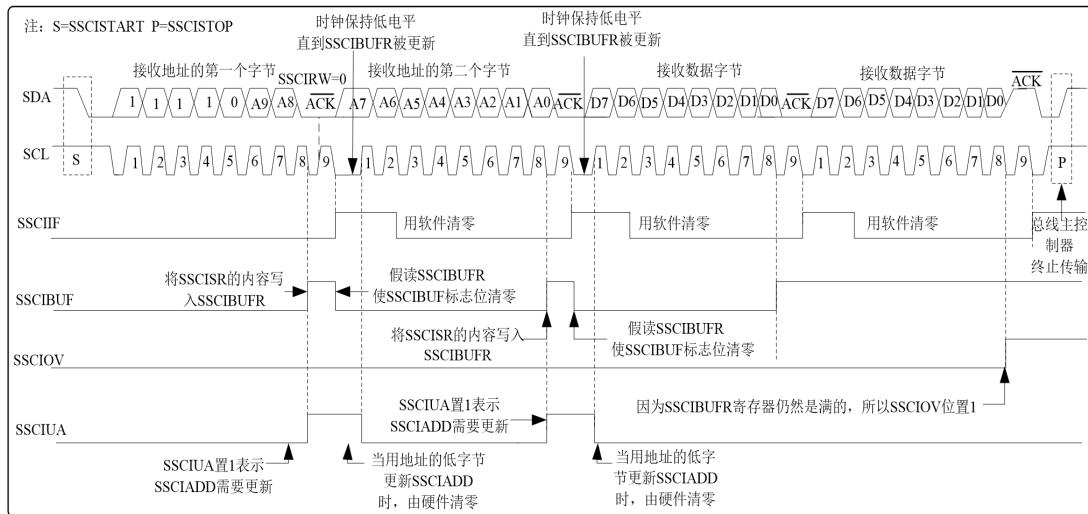


图 10.3 从动模式时序 (接收, 10 位地址)

从动接收设置:

- 1、通过 SSCIMOD<3:0>位选择 I2C 工作模式
 - 0110 = I2C 从动模式, 7 位地址
 - 0111 = I2C 从动模式, 10 位地址
 - 1011 = I2C 固件控制主控模式 (从动空闲模式)
 - 1110 = I2C 从动模式, 7 位地址, 并允许启动位和停止位中断
 - 1111 = I2C 从动模式, 10 位地址, 并允许启动位和停止位中断
- 2、设置SSCIADD寄存器, 设置从机地址, 仅高七位有效;
- 3、清零SSCISTA寄存器的各标志, 包括SSCIDA、SSCIRW、SSCIBUF等。
- 4、设置SDA引脚为输入, SCL为输入;
- 5、清零SSCIIF标志, 如果需要中断打开各终端使能位;
- 6、使能SSCIEN, 开始接收数据, 等待地址匹配; 如果地址匹配, 则SSCISTA寄存器的SSCIRW位清零。SSCISR寄存器的值被装入SSCIBUFR寄存器;
- 7、缓冲器满标志位SSCIBUF被置1; 产生 $\overline{\text{ACK}}$ 脉冲信号; 在第9个SCL脉冲的下降沿, SSCI中断标志位SSCIIF被置1, 软件清零。

10.4.2.3发送

当输入地址字节的SSCIRW位置1 并发生地址匹配时, SSCISTA寄存器的SSCIRW位被置1。接收到的地址被装入SSCIBUFR寄存器。ACK脉冲在第9位上发送, SCL引脚保持低电平。发送数据必须被装入SSCIBUFR寄存器, 同时也装入SSCISR寄存器。然后, 应该通过将SSCICKP位 (SSCICL0<4>) 置1来使能SCL引脚。主控制器必须在发出另一个时钟脉冲前监视SCL引脚。从控制器可以通过延长时钟低电平时间不与主控制器同步。8个数据位在SCL输入的下降沿被移出。这可以确保在SCL为高电平期间SDA信号是有效的。

每个数据传输字节都会产生SSCI中断。标志位SSCIIF必须用软件清零, SSCISTA寄存器用于确定字节的状态。标志位SSCIIF在第9个时钟脉冲的下降沿被置1。对于从发送器, 来自主接收器的ACK脉冲将在第9个SCL输入脉冲的上升沿被锁存。若SDA线为高电平 (无ACK应答信号), 则表示数据传输已完成。在这种情况下, 如果从控制器锁存了ACK, 将

复位从动逻辑（复位SSCISTA寄存器），同时从控制器监视下一个启动位的出现。如果SDA线为低电平（ACK），则必须将下一个要发送的数据装入SSCIBUFR寄存器。然后，通过将SSCICKP位（SSCICL0<4>）置1使能SCL引脚。

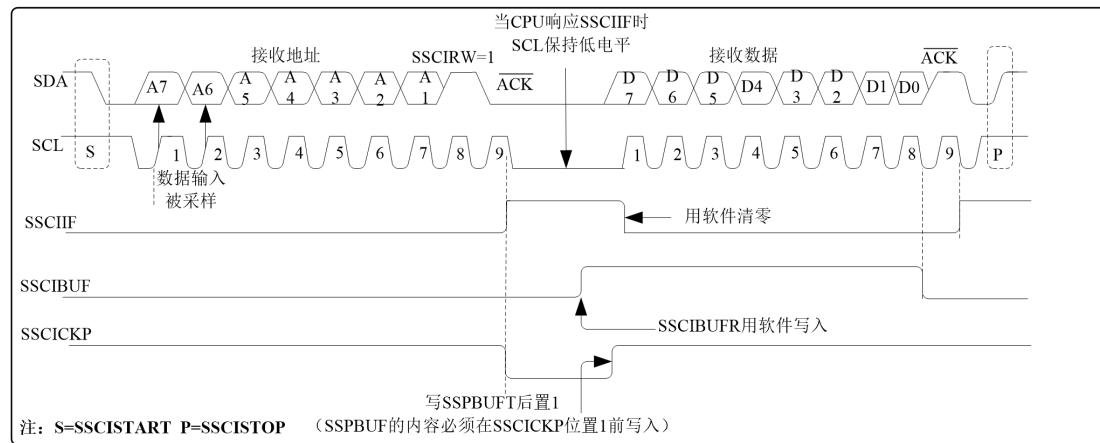


图 10.4 从动模式时序 (发送, 7 位地址)

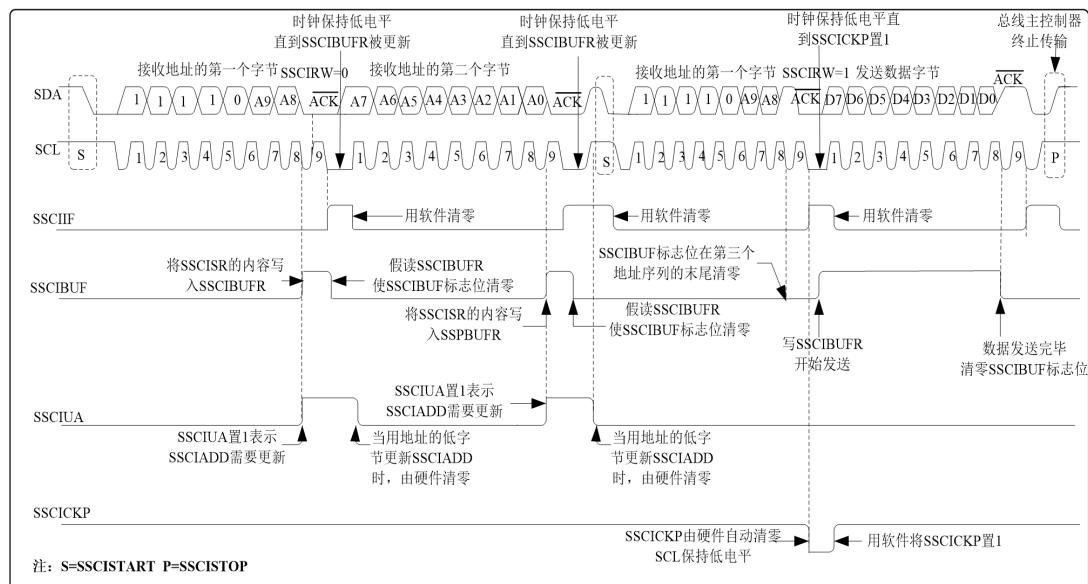


图 10.5 从动模式时序 (发送, 10 位地址)

从动发送设置：

- 1、通过 SSCIMOD<3:0>位选择 I2C 工作模式
 - 0110 = I2C 从动模式，7 位地址
 - 0111 = I2C 从动模式，10 位地址
 - 1011 = I2C 固件控制主控模式（从动空闲模式）
 - 1110 = I2C 从动模式，7 位地址，并允许启动位和停止位中断
 - 1111 = I2C 从动模式，10 位地址，并允许启动位和停止位中断
- 2、设置SSCIADD寄存器，设置从机地址，仅高七位有效；
- 3、清零SSCISTA寄存器的各标志，包括SSCIDA、SSCIRW、SSCIBUF等。
- 4、设置SDA引脚为输出，SCL为输入；

- 5、清零SSCIIF标志，如果需要中断打开各终端使能位；
- 6、使能SSCIEN，当输入地址字节的SSCIRW位置1并发生地址匹配时，SSCISTA寄存器的SSCIRW位被置1。接收到的地址被装入SSCIBUFR寄存器。
- 7、ACK脉冲在第9位上发送，SCL引脚保持低电平。发送的数据装载到SSCIBUFR寄存器。
- 8、置1SSCICKP位使能SCL引脚。主控制器必须再发送另一个时钟脉冲前件事SCL引脚。从控制器可以通过延长时钟低电平时间不与主控制器同步。
- 9、标志位SSCIIF在第9个时钟脉冲的下降沿被置1。软件清零
- 10、对于从发送器，来自主接收器的ACK脉冲将在第9个SCL输入脉冲的上升沿被锁存。若SDA线为高电平（无ACK应答信号），则表示数据传输已完成。在这种情况下，如果从控制器锁存了ACK，将复位从动逻辑（复位SSCISTA寄存器），同时从控制器监视下一个启动位的出现。如果SDA线为低电平（ACK），则必须将下一个要发送的数据装入SSCIBUFR寄存器。然后，通过将SSCICKP位（SSCICL0<4>）置1使能SCL引脚。

10.4.2.4广播呼叫地址支持

在I2C总线的寻址过程中，通常由启动条件后的第一个字节决定主器件将寻址哪个从器件。但广播呼叫地址例外，它能寻址所有器件。当使用这个地址时，理论上所有的器件都应该发送一个应答响应。

广播呼叫地址是根据I2C协议为特定目的保留的八个地址之一。它由全0组成，且SSCIRW=0。广播呼叫使能位SSCICALLEN（SSCICL1<7>寄存器使能时，即可识别广播呼叫地址。检测到起始位后，8位数据会移入SSCISR，同时将该地址与SSCIADD进行比较。它还会与广播呼叫地址进行比较并用硬件设定。

如果与广播呼叫地址匹配，SSCISR的值将传输到SSCIBUFR，SSCIBUF标志位（第8位）置1，并且SSCIIF中断标志位在第9位（ACK位）的下降沿置1。

当响应中断时，可以通过读取SSCIBUFR的内容来判断中断源。该值可以用于判断地址是特定器件的还是一个广播呼叫地址。

在10位模式下，需要更新SSCIADD以使地址的后半部分匹配，同时SSCIUA位（SSCISTA寄存器）置1。如果SSCICALLEN位置1时采样到广播呼叫地址，同时从器件被配置为10位地址模式，则不再需要地址的后半部分，也不会将SSCIUA位置1，从器件将在应答后开始接收数据如下图13.6所示。

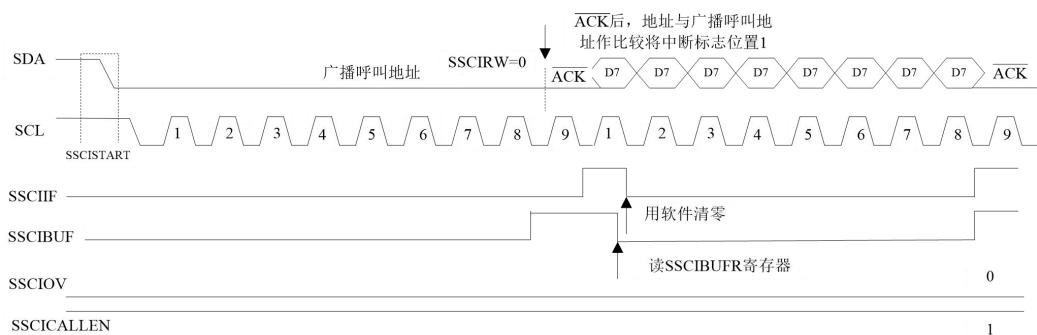


图 10.6 从动模式广播呼叫地址时序（7或10位地址模式）

10.4.3 I2C 主动模式

主控模式通过固件在检测到启动条件和停止条件时产生中断来工作。停止（SSCISTOP）位和启动（SSCISTART）位在复位时或禁止SSCI模块时清零。停止（SSCISTOP）位和启动（SSCISTART）位会根据启动和停止条件翻转。当SSCISTOP位置1时，可以获得I2C总线的控制权；否则，停止（SSCISTOP）位和启动（SSCISTART）位都清零，总线处于空闲状态。

在主控模式下，SCL和SDA线通过清零相应的I/O口方向控制位（TRx位）来控制。输出电平始终为低电平。因此当发送数据时，SDA和SCL对应I/O口都需要配置为输出。同时SCL和SDA引脚上必须外接上拉电阻，才能使I2C模块正常工作。

下列事件会使SSCI中断标志位SSCIIF置1（如果允许SSCI中断，则产生中断）：

启动条件

- 停止条件
- 发送/接收到数据传输字节
- 应答发送
- 重复启动条件

可用从动模式空闲（SSCIMOD<3:0> = 1011）或从动模式活动完成主控模式操作。当同时使能主控模式和从动模式时，需要使用软件区分中断源。

10.4.3.1 主控模式支持

通过设置SSCICL0中的SSCIMOD<3:0>并将SSCIEN位置1可使能主控模式。一旦使能主控模式，

用户即可选择以下6项操作：

- 1) 在SDA 和SCL 上发出一个启动条件。
- 2) 在SDA 和SCL 上发出一个重复启动条件。
- 3) 写入SSCIBUFR寄存器，开始数据/地址的发送。
- 4) 在SDA 和SCL 上产生停止条件
- 5) 将I2C 端口配置为接收数据。
- 6) 在接收到数据字节后产生应答条件。

注：

当配置为I2C主控模式时，SSCI模块不允许事件排队。例如，在启动条件结束前，不允许用户发出另一个启动条件并立即写SSCIBUFR寄存器以发起传输。这种情况下，将不会写入SSCIBUFR，SSCIWCFL 位将被置1，这表明没有发生对SSCIBUFR的写操作。下图为I2C主模式框图。

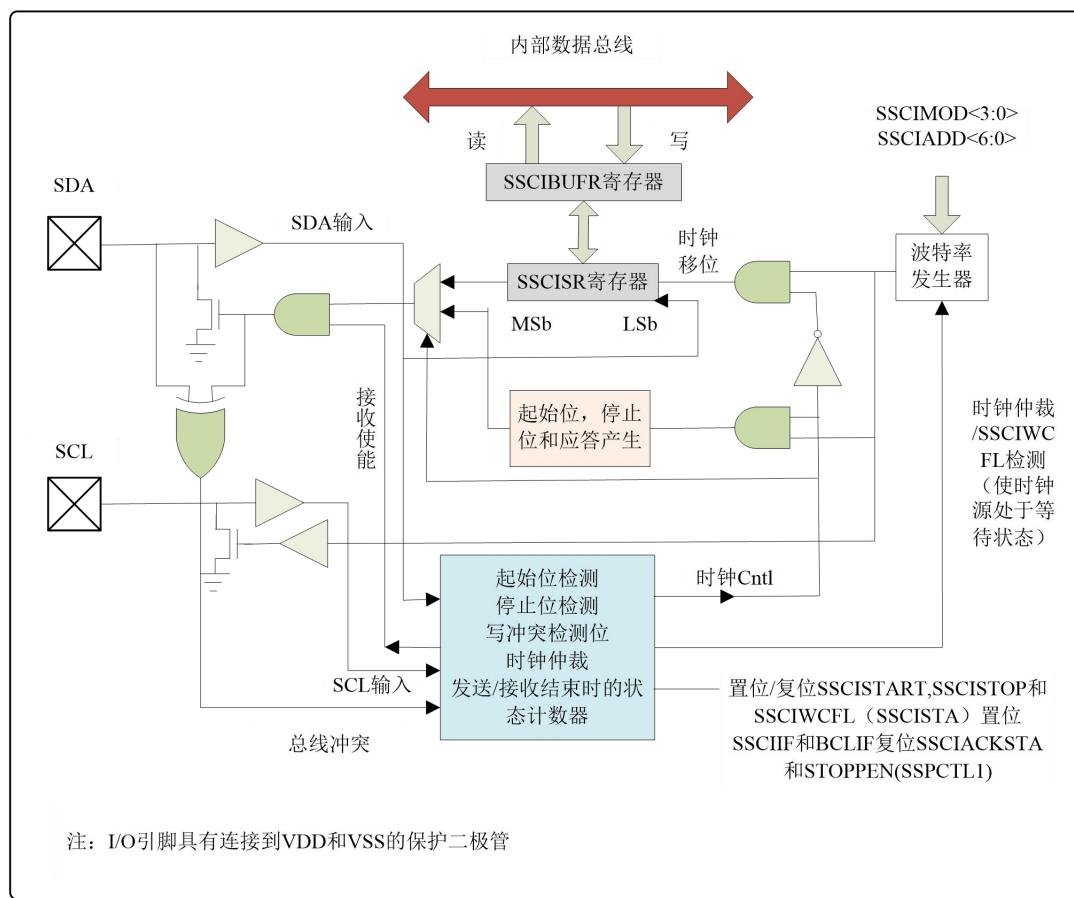


图 10.7 I2C 主模式框图

10.4.3.2 I2C 主模式操作

所有串行时钟脉冲和启动/停止条件均由主器件产生。停止条件或重复启动条件能结束传输。因为重复启动条件也是下一次串行传输的开始，因此不会释放I2C总线。在主控发送器模式下，串行数据通过SDA输出，而串行时钟由SCL输出。发送的第一个字节包括接收器件的地址（7位）和读/写（SSCIRW）位。在这种情况下，SSCIRW位将是逻辑0。串行数据每次发送8位。每发送一个字节，会收到一个应答位。启动和停止条件的输出表明串行传输的开始和结束。

在主控接收模式下，发送的第一个字节包括发送器件的地址（7位）和SSCIRW位。在这种情况下，SSCIRW位将是逻辑1。因此，发送的第一个字节是一个7位从器件地址，后面跟1表示接收。串行数据通过SDA接收，而串行时钟由SCL输出。每次接收8位串行数据。每接收到一个字节，都会发送一个应答位。启动和停止条件分别表明发送的开始和结束。

在I2C模式下，在SPI模式中使用的波特率发生器被用于将SCL时钟频率设置为100 kHz、400 kHz或1 MHz。波特率发生器的重载值位于SSCIADD寄存器的低7位。当发生对SSCIBUFR的写操作时，波特率发生器将自动开始计数。如果指定操作完成（即，发送的最后一个数据位后面跟着ACK），内部时钟将自动停止计数，SCL引脚将保持在其最后的状态。

下面是一个典型的发送事件序列：

- 1) 用户通过将启动使能位STARTEN（SSCICL1寄存器）置1产生启动条件。

- 2) SSCIIF 位置1。在进行任何其他操作前，SSCI模块将等待所需的启动时间。
- 3) 用户将从器件地址装入SSCIBUFR进行发送。
- 4) 地址从SDA 引脚移出，直到发送完所有8 位为止。
- 5) SSCI模块移入来自从器件的ACK位，并将它的值写入SSCICCTL1 寄存器的SSCIACKSTA位。
- 6) SSCI模块在第9 个时钟周期的末尾将SSCIIF位置1，产生一个中断。
- 7) 用户将8 位数据装入SSCIBUFR。
- 8) 数据从SDA 引脚移出，直到发送完所有8 位为止。
- 9) SSCI模块移入来自从器件的ACK位，并将它的值写入SSCICCTL1 寄存器的SSCIACKSTA 位。
- 10) SSCI 模块在第9 个时钟的末尾将SSCIIF 位置1，产生一个中断。
- 11) 用户通过将停止使能位 (STOPEN) 位 (SSCICCTL1寄存器) 置1产生停止。
- 12) 一旦停止条件完成，将产生一个中断。

10.4.3.3 波特率发生器

在I2C主控模式下，波特率发生器的重载值位于SSCIADD 寄存器的低7位。当装载了该值后，波特率发生器将自动开始计数并递减至0，然后停止直到下次重载为止。BRG 会在每个指令周期 (TCY) 中的Q2 和Q4 时钟周期上进行两次减计数。在I2C 主控模式下，会自动重载BRG。例如，在发生时钟仲裁时，BRG 将在SCL 引脚采样到高电平时重载。如图11.8和图11.9 所示。

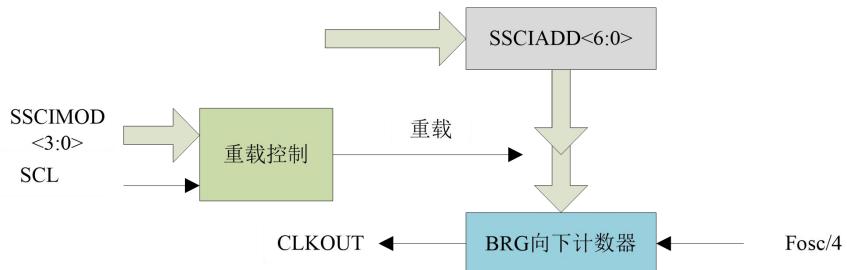


图 10.8 波特率发生器框图

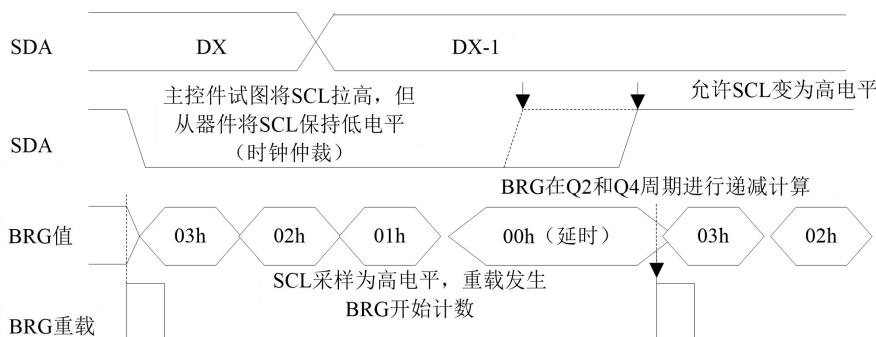


图 10.9 带有时钟仲裁的波特率发生器时序

10.4.3.4 I2C 主控模式启动条件时序

要发起启动条件，用户应将SSCICCTL1寄存器的启动条件使能位STARTEN置1。当SDA和SCL引脚都采样为高电平时，波特率发生器重新装入SSCIADD<6:0>的内容并开始计数。当波特率发生器发生超时（TBRG）时，如果SCL和SDA都采样为高电平，则SDA引脚被驱动为低电平。当SCL为高电平时，将SDA驱动为低电平就是启动条件，将使SSCISTART位（SSCISTART寄存器）置1。随后波特率发生器重新装入SSCIADD<6:0>的内容并恢复计数。当波特率发生器超时（TBRG）时，SSCICCTL1寄存器的STARTEN位将自动被硬件清零。波特率发生器暂停工作，SDA线保持低电平，启动条件结束。

注意：

如果在启动条件开始时，SDA和SCL引脚已经采样为低电平，或者在启动条件期间，SCL在SDA线被驱动为低电平之前已经采样为低电平，则会发生总线冲突。总线冲突中断标志位BCLIF置1，启动条件中止，I2C模块复位到空闲状态。

SSCIWCFL状态标志

当启动序列进行时，如果用户写SSCIBUFR，则SSCIWCFL被置1，同时缓冲器内容不变（未发生写操作）。

注：

由于不允许事件排队，在启动条件结束之前，不能对SSCICCTL1的低5位进行写操作。

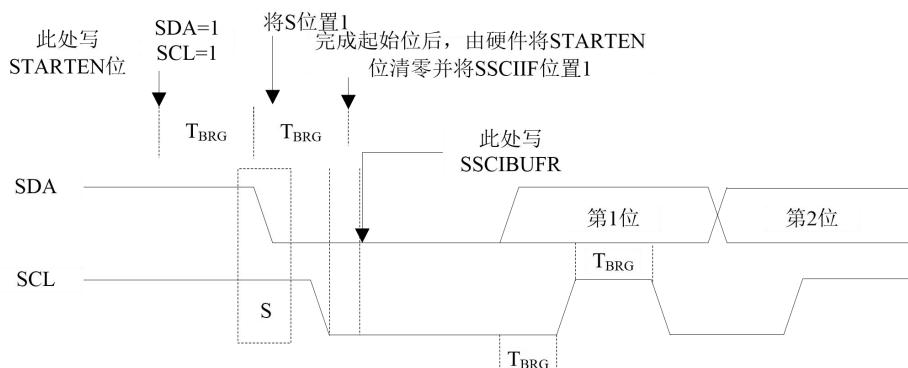


图 10.10 第一个启动位时序

10.4.3.5 I2C 主控模式重复启动条件时序

将RESTARTEN位（SSCICCTL1寄存器）编程为高电平，并且I2C逻辑模块处于空闲状态时，就会产生重复启动条件。当RESTARTEN置1时，SCL引脚被拉为低电平。当SCL引脚采样为低电平时，波特率发生器装入SSCIADD<6:0>的内容，并开始计数。在一个波特率发生器计数周期（TBRG）内SDA引脚被释放（其引脚电平被拉高）。当波特率发生器超时时，如果SDA采样为高电平，SCL引脚将被拉高。当SCL引脚采样为高电平时，波特率发生器将被重新装入SSCIADD<6:0>的内容并开始计数。SDA和SCL必须在一个计数周期TBRG内采样为高电平。随后将SDA引脚拉为低电平（SDA = 0）并保持一个计数周期TBRG，同时SCL为高电平。然后RESTARTEN位（SSCICCTL1寄存器）将自动清零，波特率发生器不会重载，SDA引脚保持低电平。一旦在SDA和SCL引脚上检测到启动条件，SSCISTART位

(SSCISTA寄存器) 将被置1。直到波特率发生器超时后, SSCIIIF位才会置1。

注:

- 1) 有任何其他事件进行时, 对RESTARTEN的编程无效。
- 2) 在重复启动条件期间, 下列事件将会导致总线冲突:
 - 当SCL 由低电平变为高电平时, SDA 采样为低电平。
 - 在SDA 被拉低之前, SCL 变为低电平。这表示可能有另一个主器件正尝试发送数据1。

一旦SSCIIIF 位被置1, 用户便可以在7 位地址模式下将7 位地址写入SSCIBUFR, 或者在10 位地址模式下写入默认的第一个地址字节。当发送完第一个8 位并接收到一个ACK 后, 用户可以发送另外8 位地址(10 位地址模式下)或8 位数据(7 位地址模式下)。

SSCIWCFL 状态标志

当重复启动序列进行时, 如果用户写SSCIBUFR, 则SSCIWCFL 被置1, 同时缓冲器内容不变(未发生写操作)。

注:由于不允许事件排队, 在重复启动条件结束之前, 不能对SSCICCTL1 的低5位进行写操作。

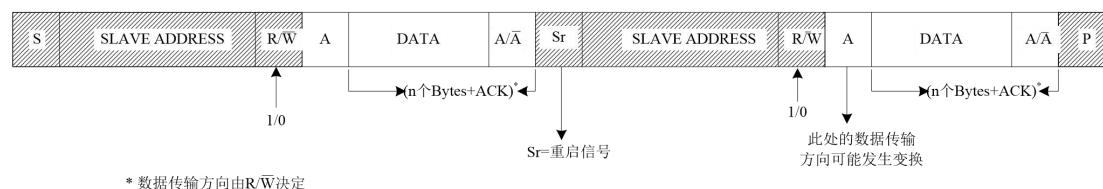


图 10.11 I2C 协议复合数据帧格式

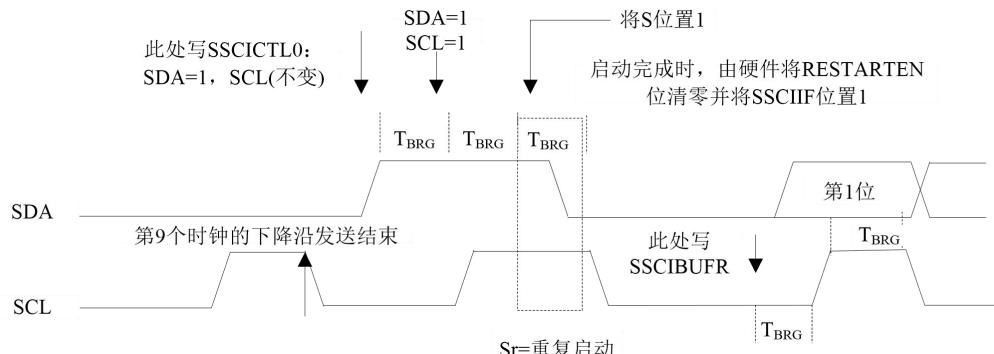


图 10.12 重复启动条件时序波形

10.4.3.6 I2C 主控模式发送

I2C 协议中 SDA 引脚上的数据不仅来源于 SSCIBUFR 寄存器，还需要按 I2C 协议的规定发送起始位、停止位、应答位等信号。下图所示是 I2C 协议典型写数据帧格式。

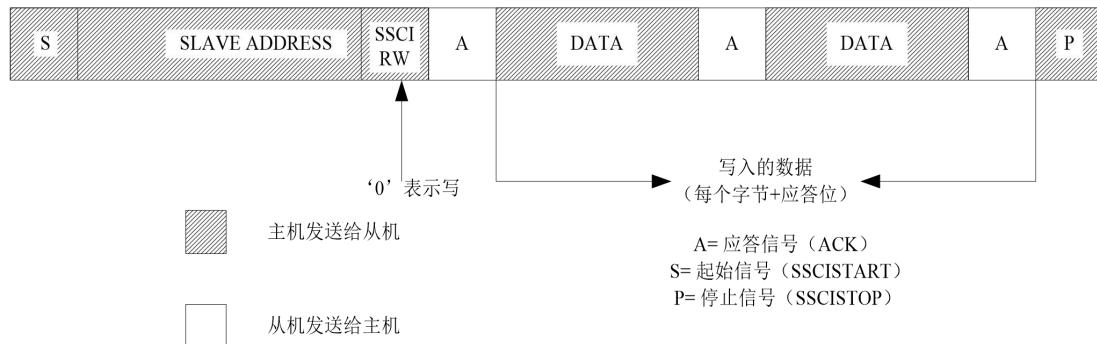


图 10.13 I2C 协议典型写数据帧格式

发送一个数据字节、一 7 位地址或一 10 位地址的另一半，都可以直接通过写一个值到 SSCIBUFR 寄存器来实现。该操作将使缓冲器满标志位 SSCIBUF 置 1，并且波特率发生器开始计数，同时启动下一次发送。

在 SCL 的下降沿有效后，地址/数据的每一位将被移出至 SDA 引脚。在一个波特率发生器计满返回计数周期 (TBRG) 内，SCL 保持低电平。数据应该在 SCL 释放为高电平前保持有效。当 SCL 引脚被释放为高电平时，它将在整个 TBRG 中保持高电平状态。在此期间以及下一个 SCL 下降沿之后的一段时间内，SDA 引脚上的数据必须保持稳定。在第 8 位被移出（第 8 个时钟周期的下降沿）之后，SSCIBUF 标志位清零，同时主器件释放 SDA。此时如果发生地址匹配或是数据被正确接收，被寻址的从器件将在第 9 位的时间以一个 ACK 位响应。ACK 的状态在第 9 个时钟周期的下降沿写入 SSCIACKDAT 位。主器件接收到应答之后，应答状态位 SSCIACKSTA 会被清零；如果未收到应答，则该位被置 1。第 9 个时钟之后，SSCIIF 位会置 1，主控时钟（波特率发生器）暂停，直到下一个数据字节装入 SSCIBUFR 为止，SCL 引脚保持低电平，SDA 保持不变。

在写 SSCIBUFR 之后，地址的每一位在 SCL 的下降沿被移出，直至地址的所有 7 位和 SSCIRW 位都被移出为止。在第 8 个时钟的下降沿，主器件将 SDA 引脚拉为高电平以允许从器件发出应答响应。在第 9 个时钟的下降沿，主器件通过采样 SDA 引脚来判断地址是否被从器件识别。ACK 位的状态被装入 SSCIACKSTA 状态位 (SSCICONTL1 寄存器)。在发送地址的第 9 个时钟下降沿之后，SSCIIF 置 1，SSCIBUF 标志位清零，波特率发生器关闭直到下一次写 SSCIBUFR，且 SCL 引脚保持低电平，允许 SDA 引脚悬空。

SSCIBUF 状态标志

在发送模式下，SSCIBUF 位 (SSCICONTA 寄存器) 在 CPU 写 SSCIBUFR 时置 1，在所有 8 位数据移出后清零。

SSCIWCFL 状态标志位

如果用户在发送过程中（即，SSCISR 仍在移出数据字节时）写 SSCIBUFR，则

SSCIWCFL 置 1 且缓冲器的内容保持不变（未发生写操作）SSCIWCFL 必须由软件清零。

SSCIACKSTA 状态标志

在发送模式下，当从器件发送应答响应（ACK = 0）时，SSCIACKSTA 位（SSCICTL1 寄存器）清零；当从器件没有应答（ACK = 1）时，该位置 1。从器件在识别出其地址（包括广播呼叫地址）或正确接收数据后，会发送一个应答。

注：若主机发送完一个字节后收到的应答标志位 SSCIACKSTA=1，则应及时停止传输（通过发送停止信号 STOPEN 位来实现）。

下图所示为 I2C 协议的典型写数据帧格式在单片机硬件 I2C 上的实现过程时序图。

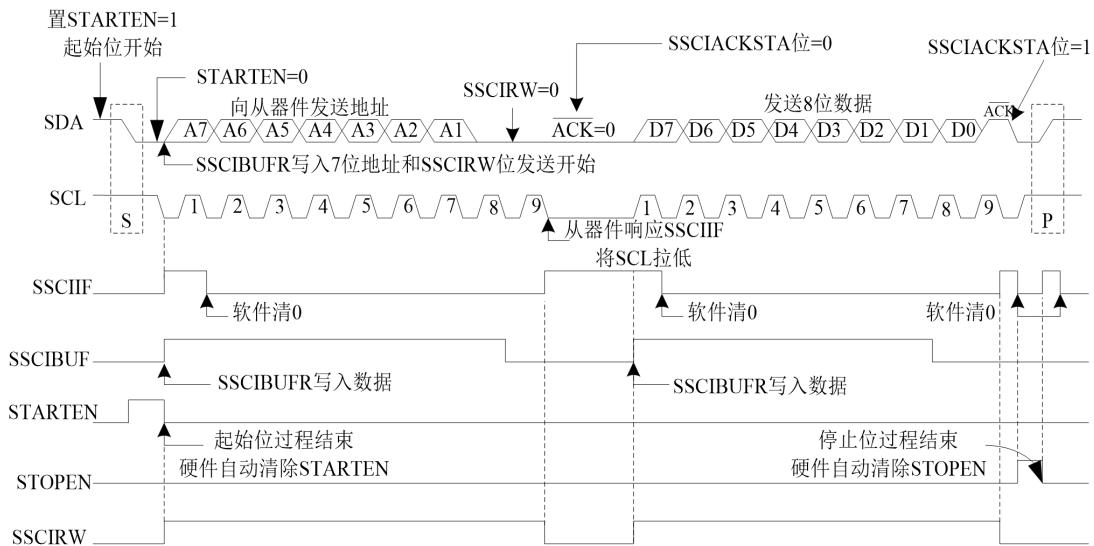


图 10.14 硬件 I2C 主模式发送数据时序图 (7 位地址)

10.4.3.7 I2C 主控模式接收

I2C主模式数据接受的系统结构与主模式数据发送系统结构相同，但是数据接收流程与数据发送流程不同。如下图为I2C协议典型读数据帧格式。

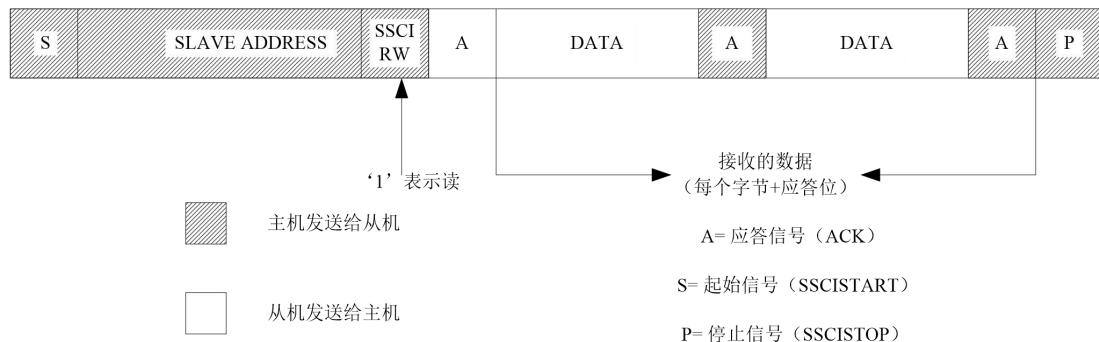


图 10.15 I2C 协议典型读数据帧格式

通过编程接收使能位SSCIRCEN（SSCICL1寄存器）使能主控模式接收。

注：SSCIRCEN位被置1前，SSCI模块必须处于空闲状态，否则SSCIRCEN位将被忽略。

波特率发生器开始计数，每次计满返回时，SCL引脚的状态都发生改变（由高变低或由低变高），且数据被移入SSCISR。第8个时钟的下降沿之后，接收使能标志位自动清零，SSCISR的内容装入SSCIBUFR，SSCIBUF标志位置1，SSCIIF标志位置1，波特率发生器暂停计数，SCL保持为低电平。此时SSCI处于空闲状态，等待下一条命令。当CPU读缓冲器时，SSCIBUF标志位将自动清零。通过将应答序列使能位SSCIACKEN（SSCICL1寄存器）置1，用户可以在接收结束后发送应答位。

SSCIBUF 状态标志

接收时，当将地址或数据字节从SSCISR装入SSCIBUFR时，SSCIBUF位置1；在读SSCIBUFR寄存器时SSCIBUF位清零

注：当读操作完成时若SSCIBUF还是1（说明SSCIBUFR上次读到的数据未被读走），会使SSCIIF接收溢出信号SSCIOV自动置位。SSCIOV必须软件清零。

SSCIOV 状态标志

接收时，当SSCISR 接收到8位数据时，SSCIOV位置1，SSCIBUF标志位已经在上一次接收时置1。

SSCIWCFL 状态标志

如果用户在接收过程中（即，SSCISR仍在移入数据字节时）写SSCIBUFR，则SSCIWCFL位置1，缓冲器内容不变（未发生写操作）

下图为典型读数据帧格式在单片机硬件I2C上的实现接收过程时序图。

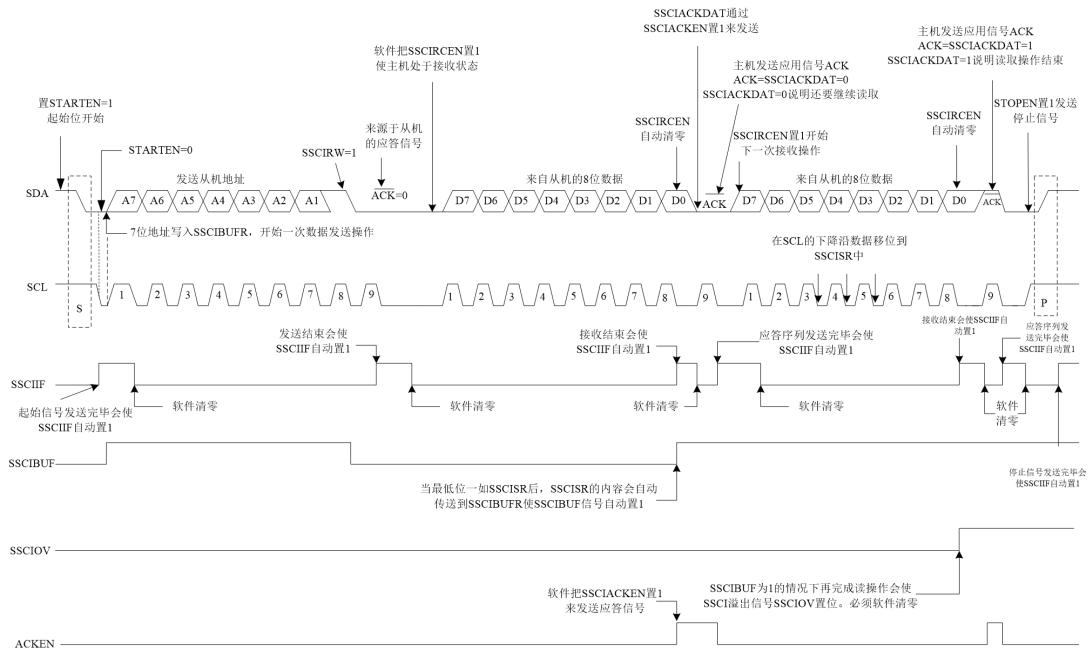


图 10.16 I2C 主模式接收数据时序图 (7 位地址)

10.4.3.8 应答序列时序

将应答序列使能位SSCIACKEN（SSCICCTL1寄存器）置1即可使能应答序列。当该位被置1时，SCL引脚被拉低，应答数据位的内容出现在SDA引脚上。如果用户希望产生一个应答，则应该将SSCIACKDAT位清零；否则，用户应该在应答序列开始前将SSCIACKDAT位置1。然后波特率发生器进行一个计满返回周期（TBRG）的计数，随后SCL引脚电平被拉高。当SCL引脚采样为高电平时（时钟仲裁），波特率发生器再进行一个TBRG周期的计数。然后SCL引脚被拉低。在这之后，SSCIACKEN位自动清零，波特率发生器关闭，SSCI模块进入空闲模式。

SSCIWCFL 状态标志位

如果用户在应答序列正在进行时写SSCIBUFR, SSCIWCFL 将被置1 且缓冲器的内容保持不变（未发生写操作）。

10.4.3.9 停止条件序列

在接收/发送结束时，通过置停止序列的使能位，STOPEN（SSCICCTL1寄存器），SDA引脚将产生一个停止位。在接收/发送结束时，SCL引脚在第9个时钟的下降沿后保持低电平。当STOPEN位置1时，主控器件将SDA置为低电平。当SDA线采样为低电平时，波特率发生器被重新装入值并递减计数至0。波特率发生器发生超时时，SCL引脚被拉到高电平，且一个TBRG（波特率发生器计满回零）后，SDA引脚被重新拉到高电平。当SDA引脚采样为高电平且SCL也是高电平时，SSCISTOP位（SSCISTATA寄存器）置1。一个TBRG周期后，STOPEN位清零且SSCIIF位置1。

SSCIWCFL 状态标志

如果用户在停止序列进行过程中试图写SSCIBUFR，则SSCIWCFL 位将置1，缓冲器的

内容不会改变 (未发生写操作)。

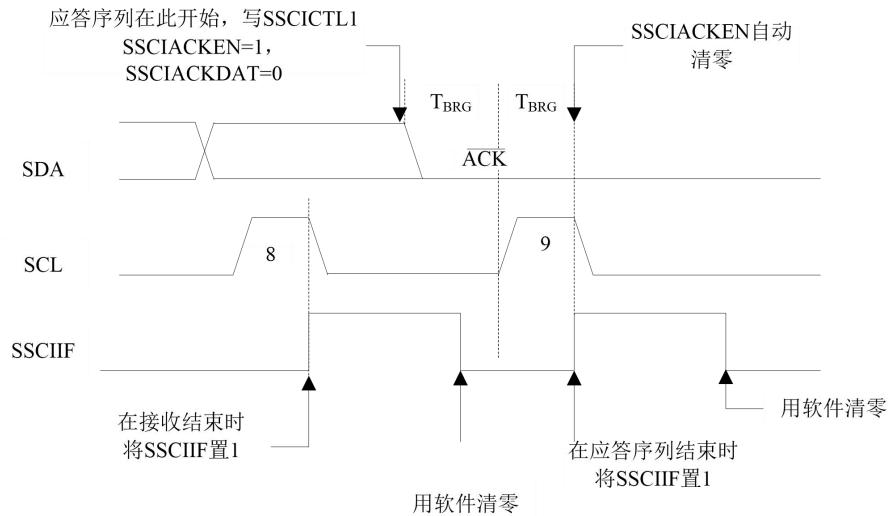


图 10.17 应答序列时序波形

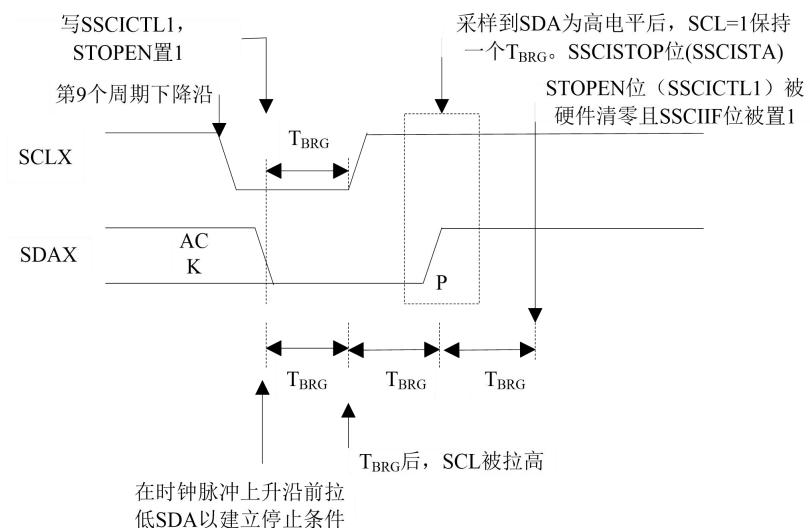


图 10.18 停止条件接收或发送模式

10.4.3.10时钟仲裁

如果在任何接收、发送或重复启动/停止条件期间，主器件拉高了 SCL 引脚（允许 SCL 引脚悬空为高电平），就会发生时钟仲裁。如果允许 SCL 引脚悬空为高电平，波特率发生器（BRG）将暂停计数，直到实际采样到 SCL 引脚为高电平为止。当 SCL 引脚采样为高电平时，波特率发生器中将被重新装入 $SSCIADD<6:0>$ 的内容并开始计数。这可以保证当外部器件将时钟拉低时，SCL 始终保持至少一个 BRG 计满返回周期的高电平。

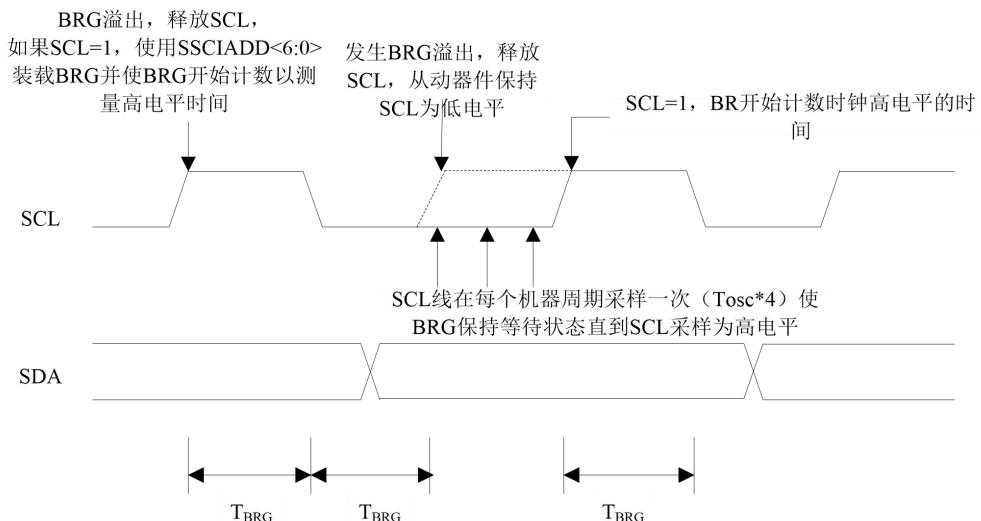


图 10.19 主控发送模式下的时钟仲裁时序

10.4.4 多主控器模式

在多主控制器模式下,在检测到启动条件和停止条件时产生的中断可用于判断总线是否空闲。停止 (SSCISTOP) 位和启动 (SSCISTART) 位在复位时或禁止SSCI模块时被清零。停止 (SSCISTOP) 位和启动 (SSCISTART) 位会根据启动和停止条件翻转。当SSCISTOP 位 (SSCISTA<4>) 置1时, 可以获得I2C总线的控制权; 否则, SSCISTOP位和SSCISTART位都清零, 总线处于空闲状态。当总线处于忙状态且允许SSCI中断时, 一旦发生停止条件便产生中断。

在多主控制器操作中, 必须监视SDA线以确定信号电平是否为所需的输出电平。此检查仅需在输出为高电平时进行。如果期望输出高电平, 但检测到的是低电平, 器件就需要释放SDA和SCL线 (将对应的I/O口方向控制位TRx置1)。此仲裁在以下状态可能会失败:

- 地址传输
- 数据传输
- 启动条件
- 重复启动条件
- 应答条件

当使能从动逻辑电路时, 从控制器将继续接收数据。如果在地址传输阶段仲裁失败, 可能表示与器件的通信正在进行中。如果寻址到器件, 则将会产生一个ACK脉冲。如果在数据传输阶段仲裁失败, 则器件需要在以后重新传输数据。

10.4.4.1 多主机通信, 总线冲突与总线仲裁

多主机模式是通过总线仲裁来支持的。

当主器件将地址/ 数据位输出到 SDA 引脚时, 如果一个主器件通过将 SDA 引脚悬空为高电平以在 SDA 上输出 1, 而另一个主器件输出 0, 就会发生总线仲裁。

如果 SDA 引脚上期望的数据是 1, 而实际在 SDA 引脚上采样到的数据是 0, 则发生了总线冲突。主器件将把总线冲突中断标志位 BCLIF 置 1, 并将 I2C 端口复位到空闲状态。如果在发送过程中发生总线冲突, 则发送停止, SSCIBUF 标志位清零, SDA 和 SCL 线被拉高, 并且允许对 SSCIBUFR 进行写操作。当执行完总线冲突中断服务程序后, 如果 I2C 总线空闲, 用户可通过发出启动条件恢复通信。

如果在启动、重复启动、停止或应答条件的进行过程中发生总线冲突, 则该条件被中止, SDA 和 SCL 线被拉高, SSCICTL1 寄存器中的对应控制位清零。当执行完总线冲突中断服务程序后, 如果 I2C 总线空闲, 用户可通过发出启动条件恢复通信。主器件将继续监视 SDA 和 SCL 引脚。如果出现停止条件, SSCIIF 位将被置 1。无论发生总线冲突时发送的进度如何, 写 SSCIBUFR 都会从第一个数据位开始发送数据。在多主机模式下, 通过在检测到启动和停止条件时产生中断可以确定总线何时空闲。SSCISTOP 位置 1 时, 可以获取 I2C 总线的控制权, 否则总线空闲且 SSCISTART 和 SSCISTOP 位清零。

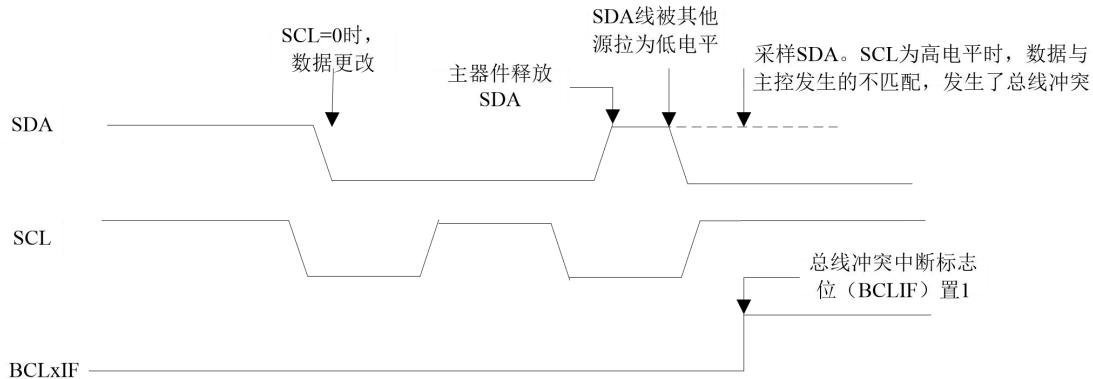


图 10.20 发送和应答时的总线冲突时序

10.4.4.2 启动条件期间的总线冲突

启动条件期间, 以下事件将导致总线冲突:

- 1) 在启动条件开始时, SDA 或 SCL 被采样为低电平。
- 2) SDA 被拉低之前, SCL 采样为低电平。

在启动条件期间, SDA 和 SCL 引脚都会被监视。如果 SDA 引脚已经是低电平, 或 SCL 引脚已经是低电平, 则:

- 中止启动条件,
- BCLIF 标志位置 1,
- 并将 SSCI 模块复位为空闲状态。

启动条件从 SDA 和 SCL 引脚被拉高开始。当 SDA 引脚采样为高电平时, 波特率发生器装入 $\text{SSCIADD}<6:0>$ 的值并递减计数到 0。如果在 SDA 为高电平时, SCL 引脚采样为低电平, 则发生总线冲突, 因为这表示另一个主器件在启动条件期间试图发送一个数据 1。

如果 SDA 引脚在该计数周期内采样为低电平, 则 BRG 复位, 同时 SDA 线保持原值。但是, 如果 SDA 引脚采样为 1。如果 SDA 引脚将在 BRG 计数结束时被置为低电平。随后波特率发生器被重新装入值并递减计数至 0。在此期间, 如果 SCL 引脚采样到 0, 则没有发生总线冲突。在 BRG 计数结束时, SCL 引脚被拉为低电平。

注:

在启动条件期间不会发生总线冲突是因为两个总线主器件不可能精确地在同一时刻发出启动条件。因此总是有一个主器件先于另一个主器件将 SDA 拉低。但是这一情况不会引起总线冲突, 因为允许两个主器件对启动条件后的第一个地址进行仲裁。如果地址是相同的, 将继续对数据部分、重复启动条件或停止条件进行仲裁。

在STARTEN位置1之前SDA变为低电平。BCLIF置1，因为
SDA=0, SCL=1, 所以SSCISTART位和SSCHIF位被置1

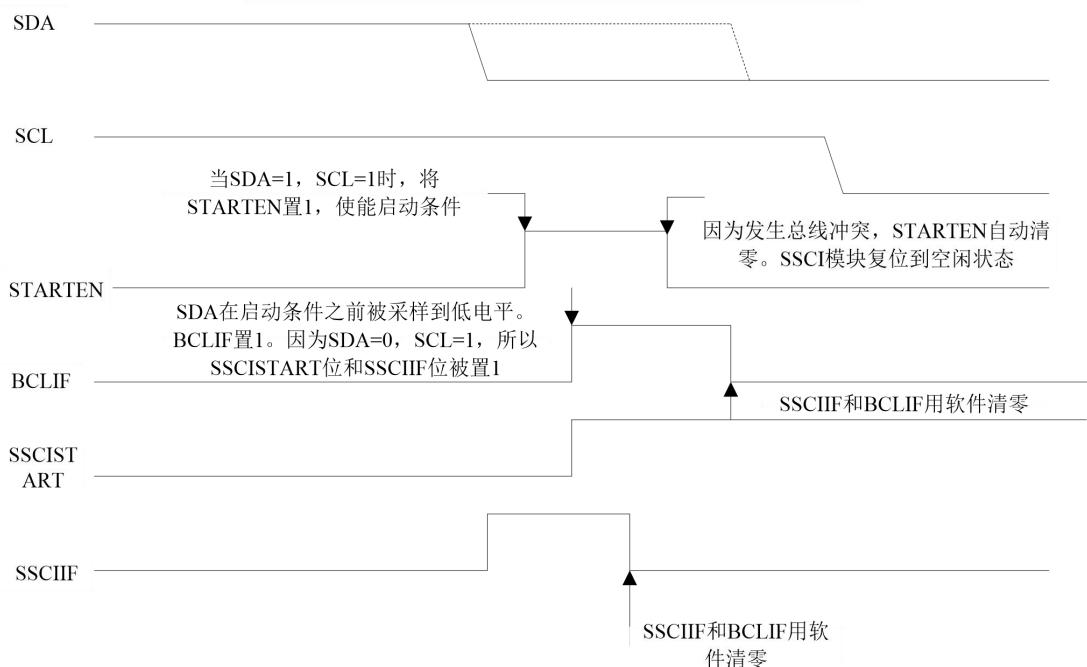


图 10.21 启动条件期间的总线冲突（仅 SDA）

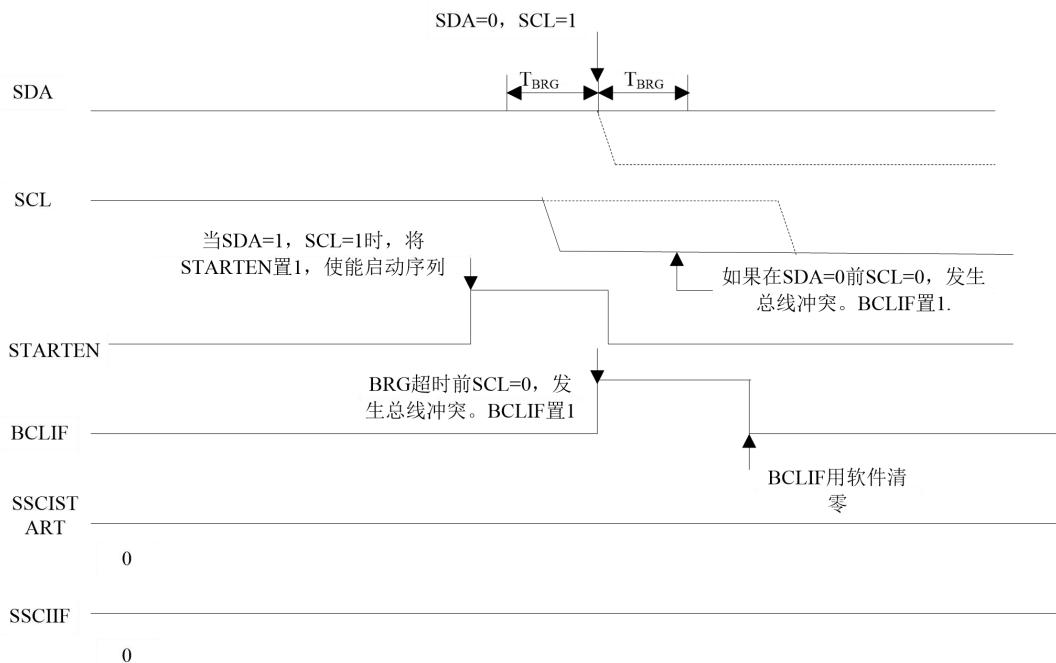


图 10.22 启动条件期间的总线冲突（SCL=0）

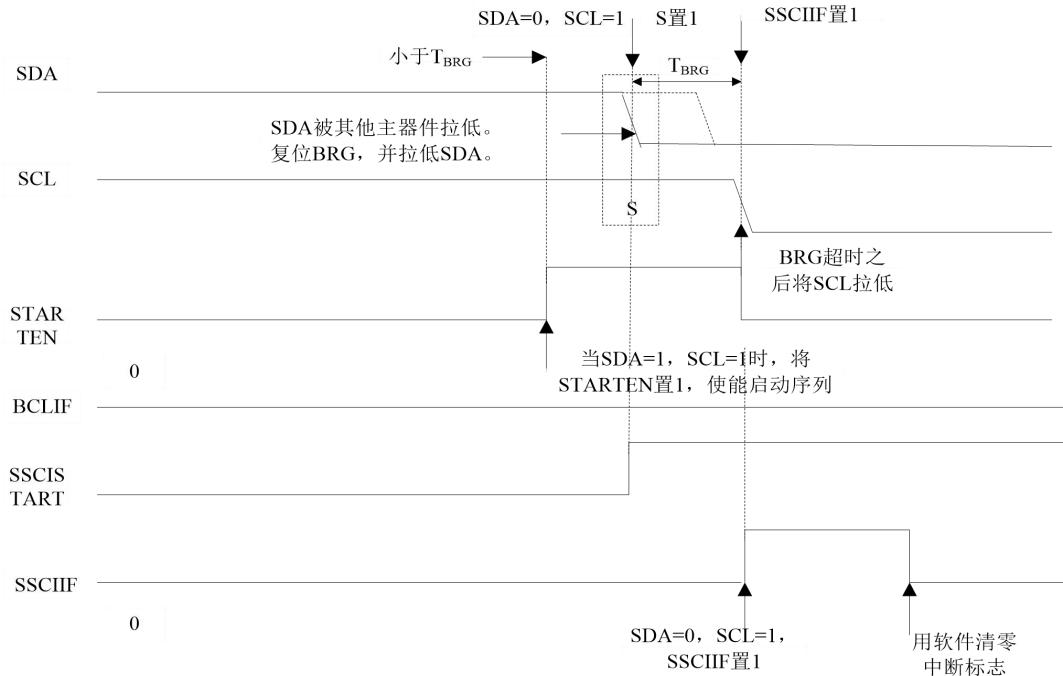


图 10.23 启动条件期间由 SDA 仲裁引起的 BRG 复位

10.4.4.3 重复启动条件期间的总线冲突

在下列情况下，重复启动条件期间会发生总线冲突：

- 1) 在SCL 由低电平变为高电平的过程中， SDA 采样到低电平。
- 2) 在SDA 被拉为低电平之前， SCL 变为低电平，表示另一个主器件正试图发送一个数据1。

当用户拉高SDA 并允许该引脚悬空时， BRG 中装入SSCIADD<6:0> 中的值并递减计数至0。接着SCL 引脚被置为高电平，当SCL 采样到高电平时，对SDA 引脚进行采样。

如果 SDA 为低电平，则已发生了总线冲突（即，另一个主器件正试图发送一个数据0）。如果 SDA 采样为高电平，则 BRG 被重新装入值并开始计数。如果 SDA 在 BRG 超时之前从高电平变为低电平，则没有发生总线冲突，因为两个主器件不可能精确地在同一时刻将 SDA 拉低。

如果 SCL 在 BRG 超时之前从高电平变为低电平，且 SDA 尚未变为低电平，表示发生了总线冲突。在此情况下，在重复启动条件期间另一个主器件正试图发送一个数据1。

如果在 BRG 超时结束时 SCL 和 SDA 都仍然是高电平，则 SDA 引脚被拉低，BRG 重新装入值并开始计数。在计数结束时，无论 SCL 引脚的状态如何， SCL 引脚都被拉低，重复启动条件结束。

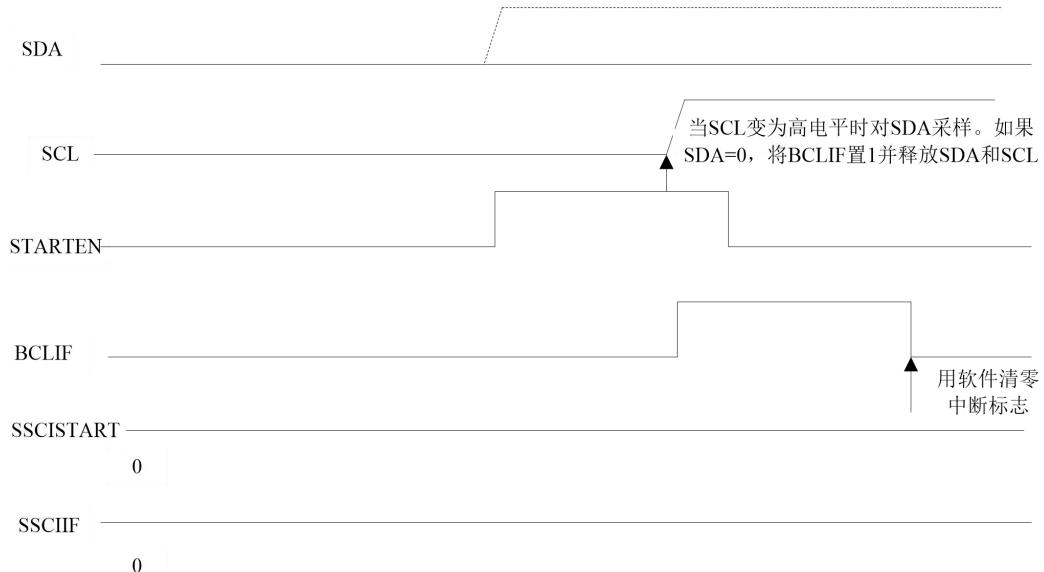


图 10.24 重复启动条件期间的总线冲突（情形 1）

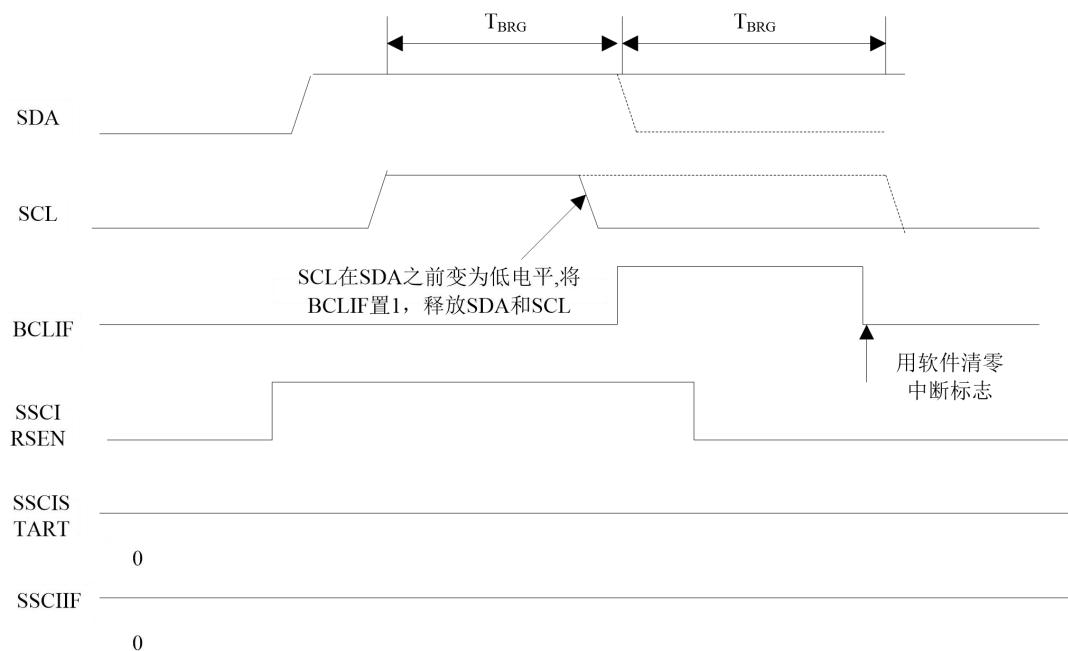


图 10.25 重复启动条件期间的总线冲突（情形 2）

10.4.4.4 停止条件期间的总线冲突

以下事件会导致停止条件期间的总线冲突：

1) SDA已被拉高并允许悬空为高电平之后，SDA在BRG 超时后被采样到低电平。

2) SCL 引脚被拉高之后，SCL 在SDA 变成高电平之前被采样到低电平。

停止条件从SDA被拉低开始。当SDA采样为低电平时，SCL 引脚就可以悬空为高电平。当引脚被采样到高电平时（时钟仲裁），波特率发生器中装入SSCIADD<6:0>的内容并递减计数到0。BRG 超时后，采样SDA。如果SDA 采样到低电平，则已发生总线冲突。这是因为另一个主器件正试图发送一个数据0。如果SCL 引脚在允许SDA 悬空为高电平前被采样到低电平，也会发生总线冲突。这是另一个主器件正试图发送一个数据0 的又一种情况。

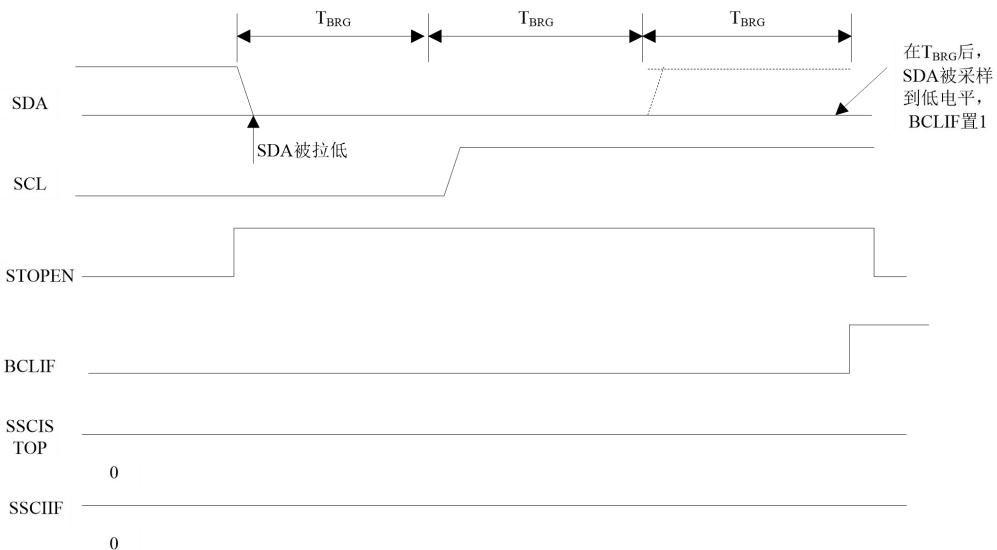


图 10.26 停止条件期间的总线冲突（情形 1）

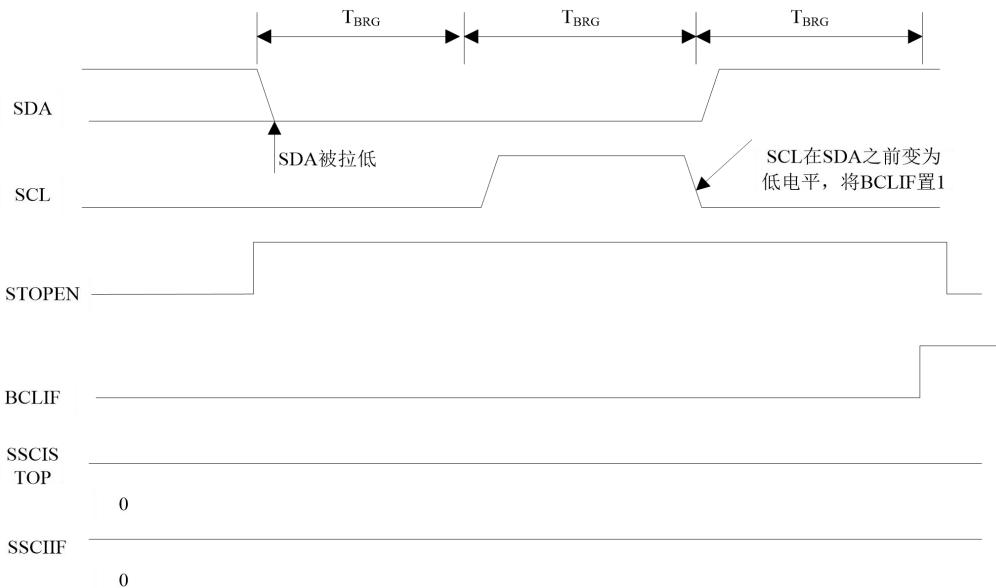


图 10.27 停止条件期间的总线冲突（情形 2）

10.4.4.5SSCI 屏蔽寄存器

在I2C 从动模式下, SSCI 屏蔽(SSCIMSK)寄存器用于在地址比较操作下屏蔽SSCISR 寄存器中的值。SSCIMSK 寄存器中某位为0 会使SSCISR寄存器中相应的位成为“无关位”。

此寄存器在任何复位条件发生时均复位为全1, 因此, 在写入屏蔽值前, 它对标准SSCI 操作没有影响。

必须在通过设置SSCIMOD<3:0> 位以选择I2C 从动模式(7位或10 位地址)之前对此寄存器进行初始化。只有通过SSCICTL0 的SSCIMOD<3:0> 位选择了适当的模式后才可访问此寄存器。SSCI 屏蔽寄存器在以下情况下有效:

7 位地址模式:与A<7:1> 进行地址比较。

10 位地址模式: 仅与 A<7:0> 进行地址比较。SSCI 屏蔽在接收到地址的第一个(高)字节期间无效。

10.5 SPI 模式

SPI 是一种应用很广泛的串口总线技术，其特点主要表现在：

- ◆ 3 线或者 4 线数据传输
- ◆ 8 位传输帧格式
- ◆ 主从模式
- ◆ 时钟频率可设
- ◆ 可编程的时钟极性和相位
- ◆ 可触发中断的发送和接收标志

SPI 的原理框图如下图所示：

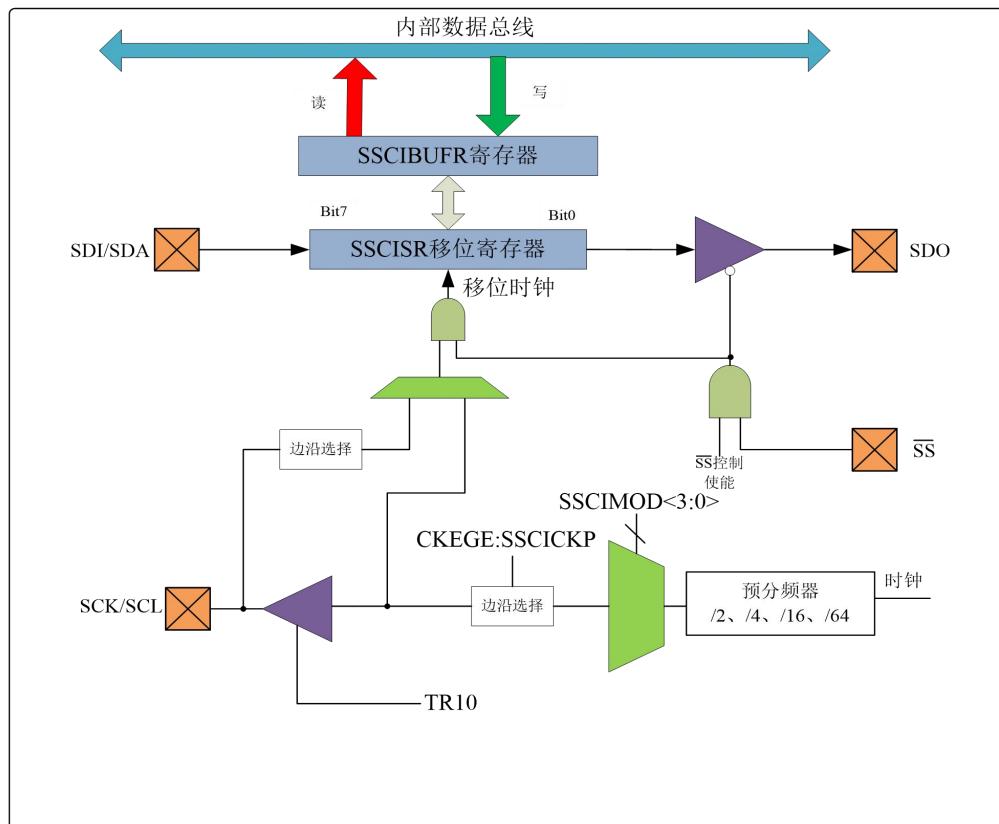


图 10.28 SPI 原理框图

在 SPI 模式下允许同时同步发送和接收 8 位数据。通常使用以下三个引脚来完成通信：

- ◆ 串行数据输出 (Serial Data Out , SDO)
- ◆ 串行数据输入 (Serial Data In, SDI)
- ◆ 串行时钟 (Serial Clock, SCK)

此外，当工作在从动模式下时可以使用第 4 个引脚：

- ◆ 从动选择 (\overline{SS})

注意：

- 1) 如果使用 SPI 从动模式，且 $CKEGE = 1$ ，则必须使能 \overline{SS} 引脚控制 ($SSCICL0<3:0> = 0100$)。
- 2) 当 SPI 处于从动模式时，如果 \overline{SS} 引脚设为 VDD，那么 SPI 模块将复位。

10.5.1 工作原理

在用户初始化SPI时,需要先通过对相应的控制位(`SSCICCTL0<5:0>` 和 `SSCISTA<7:6>`)编程来指定几个选项。这些控制位用于设置以下选项:

- ◆ 主控模式 (SCK 作为时钟输出)
- ◆ 从动模式 (SCK 作为时钟输入)
- ◆ 时钟极性 (SCK 的空闲状态)
- ◆ 输入数据的采样相位 (数据输出时间的中间或末端)
- ◆ 时钟边沿 (在 SCK 的上升沿/ 下降沿输出数据)
- ◆ 从动选择模式 (仅用于从动模式)

SSCI 模块由一个发送/ 接收移位寄存器 `SSCISR` (`SSCISR` 是内部寄存器, 程序无法直接访问) 和数据缓冲寄存器 (`SSCIBUFR`) 组成。`SSCISR` 对要发送和接收的数据进行移位, 最高有效位在前。

1) 当在接收数据时, 一旦 8 位数据接收完毕, 该字节就被移入 `SSCIBUFR` 寄存器。缓冲器满检测位 `SSCIBUF` (`SSCISTA<0>`) 和中断标志位 `SSCIIF` 将会被置 1, 而在新数据接收完毕前, `SSCIBUFR` 寄存器保存的是上次写入 `SSCISR` 的数据。

这是一种双重缓冲数据接收方式 (`SSCIBUFR`), 它允许在 CPU 读取刚接收的数据之前, 就开始接收下一个字节。但在数据发送/接收期间, 任何试图写 `SSCIBUFR` 寄存器的操作都会被忽略, 并且写冲突检测位 `SSCIWCFL` (`SSCICCTL0<7>`) 会置 1。此时用户必须用软件将 `SSCIWCFL` 位清零, 否则无法判别以后对 `SSCIBUFR` 的写操作是否成功。

2) 当在发送数据时, 为了确保应用软件能有效地接收数据, 应该在要发送的下一数据字节被写入 `SSCIBUFR` 之前, 读取 `SSCIBUFR` 中现有的数据。

缓冲器满标志位 `SSCIBUF` (`SSCISTA<0>`) 指出将接收到的数据装入 `SSCIBUFR` (发送完成) 的时间。当 `SSCIBUFR` 中的数据被读取后, `SSCIBUF` 位即被清零。如果 SPI 仅仅作为一个发送器, 则不必理会接收的数据。通常, 可用 SSCI 中断来判断发送和接收完成的时间。必须读取/ 写入 `SSCIBUFR`。如果不打算使用中断, 用软件查询的方法同样可确保不会发生写冲突。

注: 不能直接读写 `SSCISR` 寄存器, 只能通过寻址 `SSCIBUFR` 寄存器来访问。此外, SSCI 状态寄存器 (`SSCISTA`) 指示各种状态条件。

10.5.2 使能 SPI/IO 与外部链接

要使能串行端口，SSCI 使能位 **SSCIEN**（**SSCICL0<5>**）必须置 1。如果要复位或重新配置 SPI 模式，要先将 **SSCIEN** 位清零，重新初始化 **SSCICL0** 寄存器，然后将 **SSCIEN** 位置 1。这将把 **SDI**、**SDO**、**SCK** 和 **SS** 引脚配置为串行端口引脚。要让上述引脚用于串行端口功能，必须正确设置它们的 I/O 口方向控制位。即：

- ◆ **SDI**，必须将对应的 I/O 口方向控制位（**TRxx**）置 1（由 SPI 模块自动控制）
- ◆ 对于 **SDO**，必须将对应的 I/O 口方向控制位（**TRxx**）清零
- ◆ 对于 **SCK**（主控模式），必须将对应的 I/O 口方向控制位（**TRxx**）清零
- ◆ 对于 **SCK**（从动模式），必须将对应的 I/O 口方向控制位（**TRxx**）置 1
- ◆ 对于 **SS**，必须将对应的 I/O 口方向控制位（**TRxx**）置 1。

对于不需要的任何串行端口功能，可通过将对应的数据方向 **TRx** 寄存器设置为相反值来屏蔽。

10.5.3 典型连接

下图给出了两个单片机之间的典型连接。主控制器（处理器1）通过发送 **SCK** 信号来启动数据传输。在两个处理器的移位寄存器之间，数据在编程设定的时钟边沿被传送，并在相反的时钟边沿被锁存。必须将两个处理器的时钟极性（**SSCICKP**）设置为相同，这样两个处理器就可以同时收发数据。数据是否有效，取决于应用软件。这就导致以下三种数据传输情形：

- ◆ 主控制器发送数据 — 从控制器发送无效数据
- ◆ 主控制器发送数据 — 从控制器发送数据
- ◆ 主控制器发送无效数据 — 从控制器发送数据

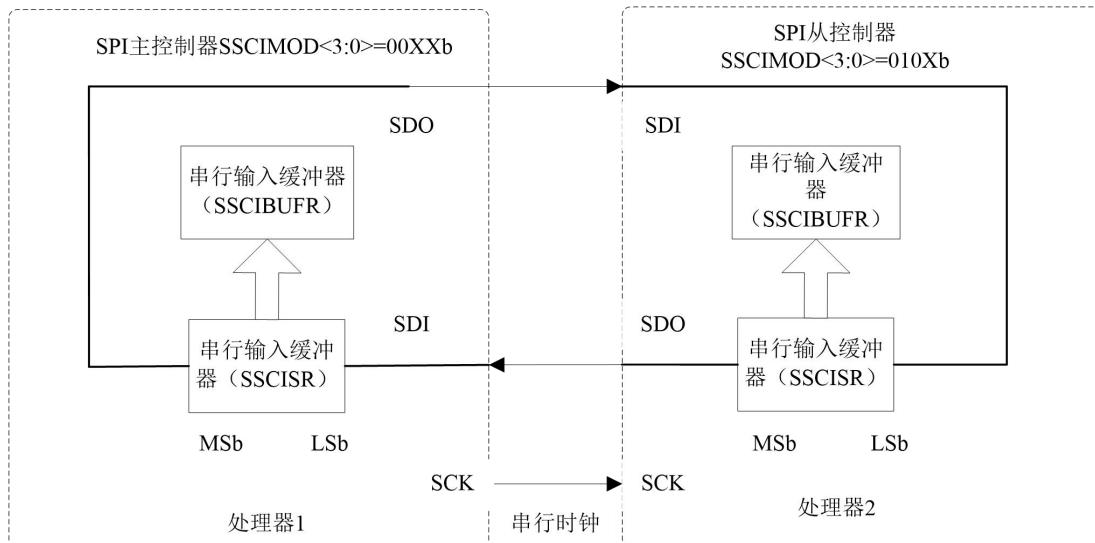


图 10.29 SPI 主从控制器连接

10.5.4 主模式

因为由主控制器控制 SCK 信号，所以它可以在任意时刻启动数据传输。主控制器根据软件协议确定从控制器（图 13.29 中的处理器 2）应在何时广播数据。在主控模式下，数据一旦写入 SSCIBUFR 寄存器就开始发送或接收。如果只打算将 SPI 作为接收器，则可以禁止 SDO 输出（将其编程设置为输入）。SSCISR 寄存器按设置的时钟速率，对 SDI 引脚上的信号进行连续移位输入。每收到一个字节，就将其装入 SSCIBUFR 寄存器，就像接收到普通字节一样（中断和状态位相应置 1）。

可通过对 SSCICKP 位 (SSCICL0<4>) 进行适当的编程来选择时钟极性。图 11.30、图 11.31、图 11.32 和图 11.33 将给出 SPI 通信的时序图，其中首先发送的是最高有效位。在主控模式下，SPI 时钟速率（波特率）可由用户编程设定为下面几种方式之一：

- SCLK/2
- SCLK/4
- SCLK/16
- SCLK/64

下图 11.30 给出了主控模式的波形图。当 CKEGE 位置 1 时，SDO 数据在 SCK 出现时钟边沿前一直有效。图中所示的输入采样的变化由 SAMPLE (SSCISTA<7>) 状态位反映。图中给出了将接收到的数据装入 SSCIBUFR 的时间。

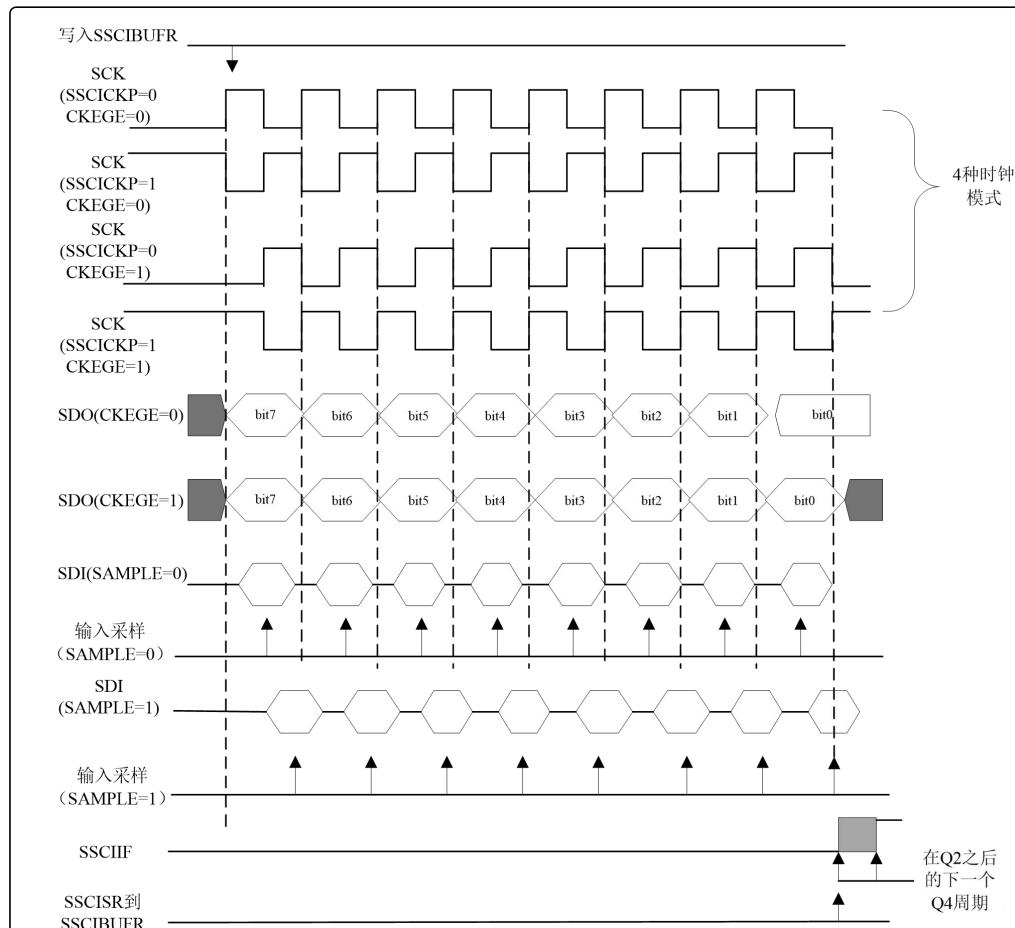


图 10.30 SPI 主控模式时序图

10.5.5 从模式

10.5.5.1 从动模式

在从动模式下，当SCK引脚上出现外部时钟脉冲时发送和接收数据。当最后一位数据被锁存后，中断标志位SSCIIF置1。在从动模式下，外部时钟由SCK引脚上的外部时钟源提供。外部时钟必须满足电气规范中规定的高电平和低电平的最短时间要求。在休眠状态下，从控制器仍可发送/接收数据。当接收到一个字节时，器件从休眠状态唤醒。当有接收数据送到SSCIBUFR后，SSCIBUF信号也会置1。当读取SSCIBUFR后SSCIBUF会自动清零。但有SSCIIF信号，一般情况下就不使用SSCIBUF来判断SSCIBUFR是否有数据了。当SSCIIF置1且SSCIBUF置1时，应立即读取SSCIBUFR中的数据，否则会出现数据丢失。

10.5.5.2 从动选择同步

\overline{SS} 引脚允许器件工作于同步从动模式。SPI必须处于从动模式，并使能 \overline{SS} 引脚控制($SSCICL0<3:0> = 0100$)。要让 \overline{SS} 引脚充当输入端，则不能将此引脚驱动为低电平。数据锁存器必须为高电平。当 \overline{SS} 引脚为低电平时，使能数据的发送和接收，同时驱动SDO引脚。当 \overline{SS} 引脚变为高电平时，即使是在字节的发送过程中，SDO引脚也不再被驱动，而是变成悬空输出状态。根据应用的需要，可在SDO引脚上外接上拉/下拉电阻。

注意：当SPI处于从动模式，并且 \overline{SS} 引脚控制使能($SSCICL0<3:0> = 0100$)时，如果 \overline{SS} 引脚置为VDD电平将使SPI模块复位。如果SPI工作在从动模式下并且CKEGE置1，则必须使能 \overline{SS} 引脚控制。

当SPI模块复位时，位计数器被强制为0。这可以通过强制将 \overline{SS} 引脚拉为高电平或将SSCIEN位清零实现。将SDO引脚和SDI引脚相连，可以仿真二线制通信。当SPI需要作为接收器工作时，SDO引脚可以被配置为输入端。这样就禁止了从SDO发送数据。因为SDI不会引起总线冲突，因而总是可以将其保留为输入(SDI功能)。

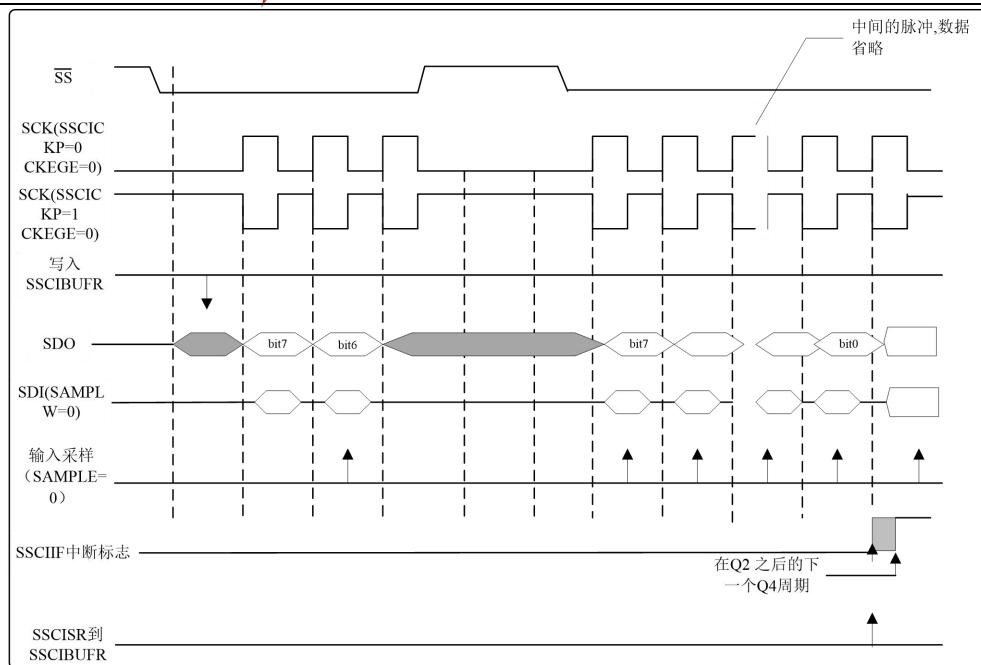


图 10.31 从动时序图

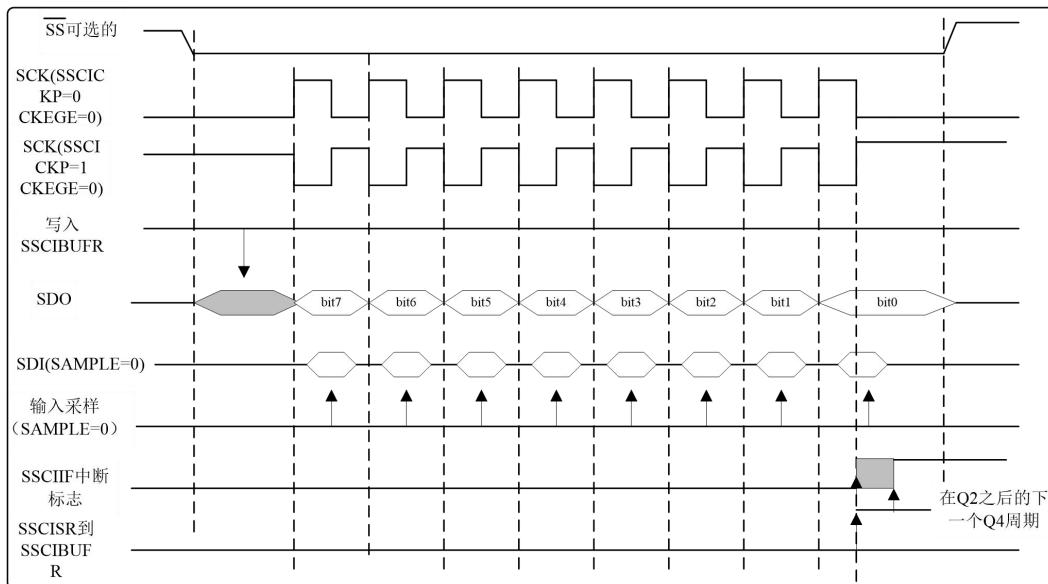


图 10.32 SPI 模式时序图 (从动模式且 CKEGE=0)

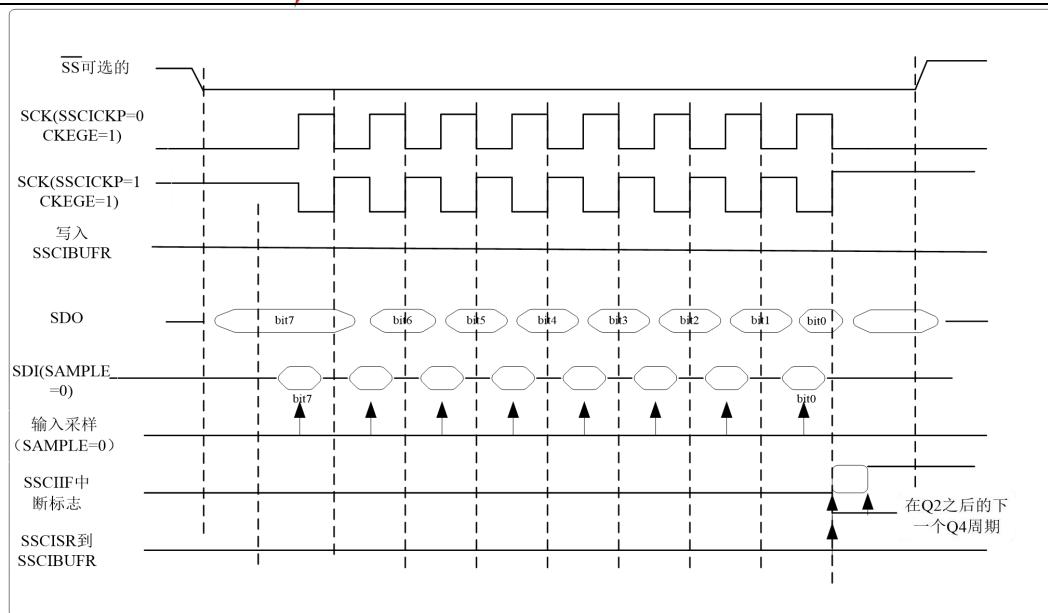


图 10.33 SPI 模式时序图 (从动模式且 CKEGE=1)

10.5.6 休眠模式和复位

休眠模式下的工作

在主控模式下，进入休眠模式后所有模块的时钟都停振，在器件被唤醒前，发送/接收也将保持原先的状态。在器件恢复正常工作模式后，模块将继续发送/接收数据。

在从动模式下，SPI发送/接收移位寄存器与器件异步工作。这可以使器件在休眠状态时，仍可使数据被移入SPI发送/接收移位寄存器。当接收完8位数据后，SSCI中断标志位将置1，如果此时该中断是允许的，还将唤醒器件。

复位的影响

复位会禁止SSCI模块并终止当前的数据传输。

10.5.7 SPI 四种工作模式设置

10.5.7.1 主控发送工作流程

主要步骤：

- 1、通过 SSCIMOD<3:0>确定 SPI 主控方式
 - 0000:SPI 主控方式，时钟=工作时钟/4
 - 0001:SPI 主控方式，时钟=工作时钟/16
 - 0010:SPI 主控方式，时钟=工作时钟/64
 - 0011:SPI 主控方式，时钟= T2 输出/2
- 2、通过寄存器 SSCICL0 的 SSCICKP 和寄存器 SSCISTA 的 CKEGE 选择时钟与数据的时序关系；
- 3、将相应的引脚 SDO 引脚设置为输出，SCK 引脚设置为输出；
- 4、置 1 SSCIEN 位，使能 SSCI 模块；
- 5、此时 SSCIIIF 清零，将要发送的数据写到 SSCIBUFR 寄存器；
- 6、当一个字节发送完毕 SSCIIIF 自动置 1（软件清零），如果需要中断，则使能相应中断使能位；

10.5.7.2 从动接收工作流程

主要步骤：

- 1、通过 SSCIMOD<3:0>确定 SPI 从动方式；
 - 0100:SPI 从动模式，时钟由 SCK 引脚输入，使能 \overline{SS} 引脚功能，此时要设置 \overline{SS} 引脚为输入口
 - 0101:SPI 从动模式，时钟由 SCK 引脚输入，关闭 \overline{SS} 引脚功能， \overline{SS} 被用作普通 I/O 引脚
- 2、通过寄存器 SSCICL0 的 SSCICKP 和寄存器 SSCISTA 的 CKEGE 选择时钟与数据的时序关系；与主控方的时钟时序要一致；
注意：当 CKEGE 为 1 时，只有信号 \overline{SS} 为低电平，从动端才会接收数据，否则从动端不

工作; \overline{SS} 信号都来源于主控方。

将相应的引脚 SDI 引脚设置输入, SCK 引脚设置为输入; 使能 \overline{SS} 引脚功能, 此时要设置 \overline{SS} 引脚为输入口;

- 3、置 1 SSCIEN 位, 使能 SSCI 模块;
- 4、当接受到一个字节的数据后, SSCIIIF 自动置 1 (软件清零), SSCIBUF 置 1, 应立即读取 SSCIBUFR 的值;
- 5、如果需要中断, 则使能相应的中断使能位;

10.5.7.3 主控接收工作流程

主要步骤:

- 1、通过 SSCIMOD<3:0>确定 SPI 主控方式
 - 0000:SPI 主控方式, 时钟=工作时钟/4
 - 0001:SPI 主控方式, 时钟=工作时钟/16
 - 0010:SPI 主控方式, 时钟=工作时钟/64
 - 0011:SPI 主控方式, 时钟=T2 输出/2
- 2、通过寄存器 SSCICL0 的 SSCICKP 和寄存器 SSCISTA 的 CKEGE 选择时钟与数据的时序关系;
- 3、将相应的引脚 SDI 引脚设置输入, SCK 引脚设置为输出;
- 4、置 1 SSCIEN 位, 使能 SSCI 模块;
- 5、当接受到一个字节的数据后, SSCIIIF 自动置 1 (软件清零), SSCIBUF 置 1, 应立即读取 SSCIBUFR 的值;
- 6、如果需要中断, 则使能相应的中断使能位;

10.5.7.4 从动发送工作流程

主要步骤:

- 1、通过 SSCIMOD<3:0>确定 SPI 主控方式;
 - 0100:SPI 从动模式, 时钟由 SCK 引脚输入, 使能 \overline{SS} 引脚功能, 此时要设置 \overline{SS} 引脚为输入口
 - 0101:SPI 从动模式, 时钟由 SCK 引脚输入, 关闭 \overline{SS} 引脚功能, \overline{SS} 被用作普通 I/O 引脚
- 2、通过寄存器 SSCICL0 的 SSCICKP 和寄存器 SSCISTA 的 CKEGE 选择时钟与数据的时序关系;
- 3、将相应的引脚 SDO 引脚设置为输出, SCK 引脚设置为输入; 如果使能 \overline{SS} 引脚功能, 此时要设置 \overline{SS} 引脚为输入口;
- 4、置 1 SSCIEN 位, 使能 SSCI 模块;
- 5、此时 SSCIIIF 清零, 将要发送的数据写到 SSCIBUFR 寄存器;
- 6、当一个字节发送完毕 SSCIIIF 自动置 1 (软件清零), 如果需要中断, 则使能相应中断使能位;

11 通用全双工/ 半双工收发器 (USART)

11.1 系统概述

KF8F513X 包含 2 个通用全双工/半双工收发器 USART1 和 USART2。USART1 与 USART2 是两个完全独立的模块。USART2 与 USART1 工作原理完全相同，区别在于 UART2 兼容 7816(在 12.7 章节介绍)。本章将以 USARTx (x=1、2) 模块来讲述。

USARTx 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。

USARTx 模块可以实现如下功能：自动波特率检测和校准、接收单个字符时唤醒和发送 13 位间隔字符，从而保证该模块可以用于居于互联网络 (Local Interconnect Network, LIN) 总线系统。

11.1.1 USART1/2 引脚复用

USART1 和 USART2 的相关功能引脚 RX1/DT1、TX1/CK1、RX2/DT2 和 TX2/CK2，可以通过 APFCTL4 和 APFCTL5 寄存器的相关位进行配置，选择合适的通道。

表 11-1 USART 引脚复用对应表

USART 相关功能引脚	对应 I/O 口	备注
TX1/CK1	P0.0 / P0.6 / P1.0 / P1.6 / P2.1 / P4.7	通过 APFCTL4 寄存器的 U1TCSEL<2:0>位进行配置
RX1/DT1	P0.1 / P1.4 / P2.0 / P5.7	通过 APFCTL4 寄存器的 U1RDSEL<1:0>位进行配置
TX2/CK2	P0.3 / P3.5 / P3.7 / P4.4	通过 APFCTL5 寄存器的 U2TCSEL<1:0>位进行配置
RX2/DT2	P0.4 / P3.4 / P4.0 / P4.3	通过 APFCTL5 寄存器的 U2RDSEL<1:0>位进行配置

11.1.2 相关寄存器

表 11-2 USART 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/ AIEH	PUIE/ AIEL	T0IE	INT0IE	P0IE	T0IF	INT0IF	P0IF
0DH	EIF2	T3IF	-	RC1IF	TX1IF	-	CCP5IF	BCLIF	SSCIIF
2DH	EIE2	T3IE	-	RC1IE	TX1IE	-	CCP5IE	BCLIE	SSCIIE
24H	IP2	PT3	-	PRC1	PTX1	-	PCCP5	PBCL	PSSCI
264H	APPCTL4	-	U1TCSEL2	U1TCSEL1	U1TCSEL0	-	-	U1RDSEL1	U1RDSEL0
265H	APPCTL5	-	-	-	-	U2TCSEL1	U2TCSEL0	U2RDSEL1	U2RDSEL0
120H	RSCTL1	SPEN1	R1X9	SRXEN1	CRXEN1	ADREN1	FRER1	OVFER1	RX9D1
24AH	RSCTL2	SPEN2	R2X9	SRXEN2	CRXEN2	ADREN2	FRER2	OVFER2	RX9D2
121H	TXSDR1	USART 发送数据寄存器 1							
24EH	TXSDR2	USART 发送数据寄存器 2							
122H	RXSDR1	USART 接收数据寄存器 1							
24BH	RXSDR2	USART 接收数据寄存器 2							
123H	BRCTL1	ABRDO VF1	RCIDL1	SCKPS1	B1RG16	BR1CKS1	BR1CKS0	WUEN1	ABRDEN1
249H	BRCTL2	ABRDO VF2	RCIDL2	SCKPS2	B2RG16	BR2CKS1	BR2CKS0	WUEN2	ABRDEN2
124H	TSCTL1	CSRS1	T1X9	TXEN1	SYNC1	SENDB1	HBRG1	TXSRS1	TX9D1
24FH	TSCTL2	CSRS2	T2X9	TXEN2	SYNC2	SENDB2	HBRG2	TXSRS2	TX9D2
150H	UPINSET1	USLM1	UPSEL1	-	-	-	-	-	-
23FH	UPINSET2	USLM2	UPSEL2	-	-	-	-	-	-
125H	EUBRGL1	USART1 波特率数据寄存器低字节							
24DH	EUBRGL2	USART2 波特率数据寄存器低字节							
126H	EUBRGH1	USART1 波特率数据寄存器高字节							
24CH	EUBRGH2	USART2 波特率数据寄存器高字节							
242H	U7816TXCTL2	U7816 EN2	CLKOUT2	TX9DSEL2	STOP2	TPAR2	TINV2	TCONV2	BGTEN2
241H	U7816RXCTL2	ERSW21	ERSW20	RPAR2	RINV2	RCONV2	-	-	PAREF2
240H	U7816CTL2	PSEL2	-	TREPEN2	TREP21	TREP20	RREPEN2	RREP21	RREP20
243H	CLKDIV2	CLKDIV 27	CLKDIV26	CLKDIV25	CLKDIV24	CLKDIV23	CLKDIV22	CLKDIV21	CLKDIV2 0
244H	EGTCTL2	EGT27	EGT26	EGT25	EGT24	EGT23	EGT22	EGT21	EGT20
127H	USLPEN1	SLPEN1	-	-	-	-	-	-	-
248H	USLPEN2	SLPEN2	-	-	-	-	-	-	-
12DH	UARTMTH1	UART1 地址匹配设置寄存器							
247H	UARTMTH2	UART2 地址匹配设置寄存器							

11.1.3 原理框图

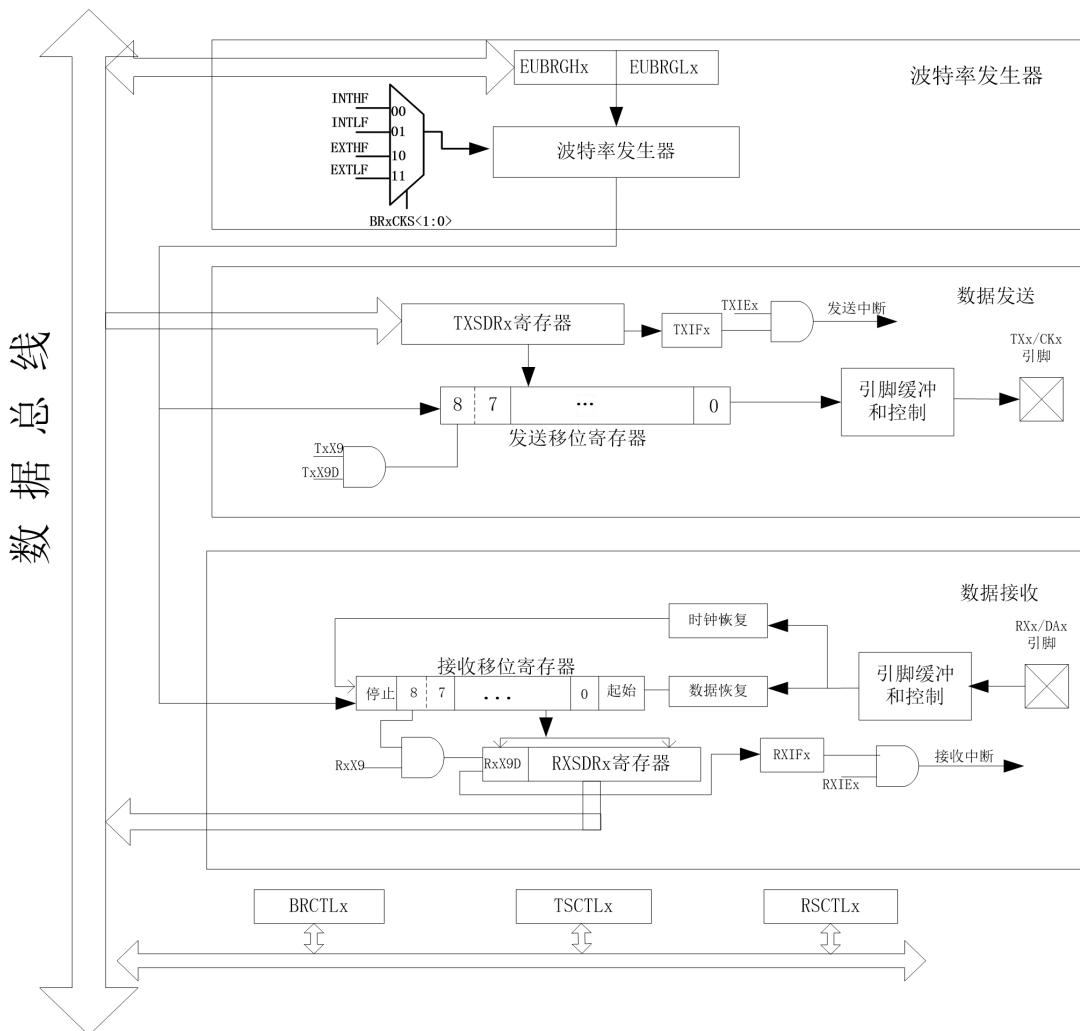


图 11.1 USARTx 模块框图

从上图中可以看出，通用全双工/半双工收发器（USARTx）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 BRCTLx、发送状态和控制寄存器 TSCTLx、接收状态和控制寄存器 RSCTLx。

在波特率发生器这部分，通过设置 SYNCx、HBRGx 和 BxRG16 位来进行选择设置 {EUBRGHx: EUBRGLx} 寄存器的值来配置波特率。详细见“[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 TSCTLx 来控制实现相应的功能。

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 RSCTLx 来控制实现相应的功能。

USARTx 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。

USARTx 模块还可实现如下附加功能，从而使其成为 LIN 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。

11.2 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 USART1 操作，默认情况下，BRG 工作在 8 位模式。BRxCKS<1:0>位决定波特率发生器选用哪一种工作时钟，其中包括 INTLF、INTLF、EXTHF 和 EXTLF 四种时钟信号。将 BRCTLx 寄存器的 BxRG16 位置 1 可选择 16 位模式。

{EUBRGHx: EUBRGLx}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 TSCTLx 寄存器的 HBRGx 位和 BRCTLx 寄存器的 BxRG16 位决定。在半双工同步模式下，HBRGx 位被忽略。

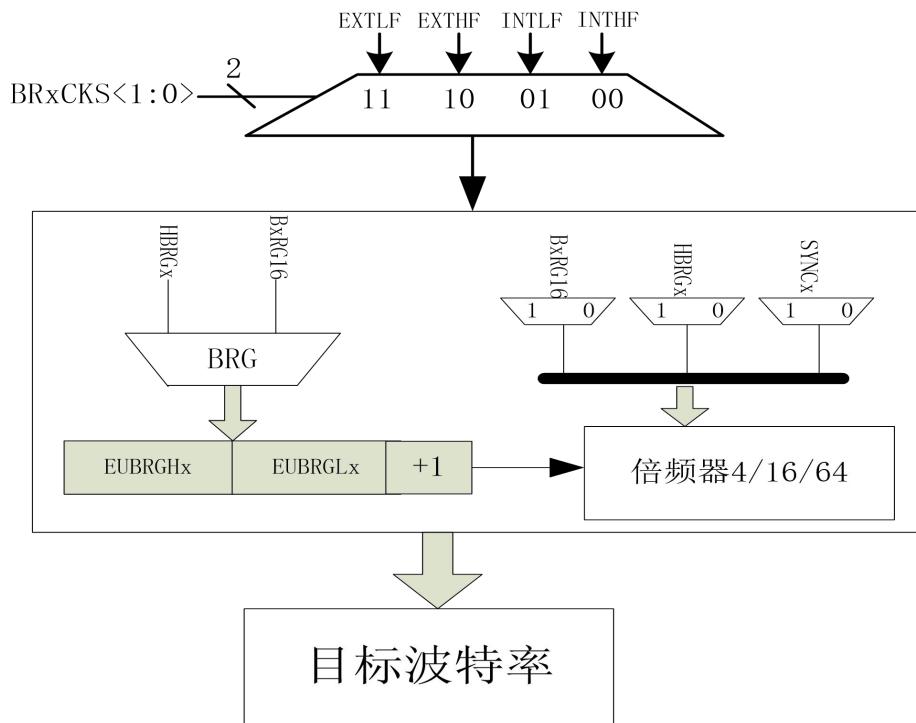


图 11.2 USARTx 目标波特率产生原理框图

11.2.1 USARTx 波特率控制寄存器 BRCTLx

寄存器: BRCTLx: 波特率控制寄存器x

复位值	bit7 ABRDOVF _x	bit6 RCIDL _{Fx}	bit5 SCKPS _x	bit4 BxRG16	bit3 BRxCKS1	bit2 BRxCKS0	bit1 WUEN _x	bit0 ABRDEN _x
0100 0000	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ABRDOVF_x: 自动波特率检测溢出位

全双工异步模式:

1 = 自动波特率定时器溢出

0 = 自动波特率定时器没有溢出

半双工同步模式:无关位

RCIDL_{Fx}: 接收空闲标志位

全双工异步模式:

1 = 接收器空闲

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_x: 半双工同步时钟极性选择位

全双工异步模式:必须为 0

半双工同步模式:

1 = 数据在时钟上升沿同步

0 = 数据在时钟下降沿同步

BxRG16: 16 位波特率发生器位

1 = 使用 16 位波特率发生器

0 = 使用 8 位波特率发生器

BRxCKS<1:0>:USART 波特率发生器时钟选择位

00 = 选择INTHF时钟作为USARTx波特率发生器时钟

01 = 选择INTLF时钟作为USARTx波特率发生器时钟

10 = 选择EXTHF时钟作为USARTx波特率发生器时钟

11 = 选择EXTLF 时钟作为 USARTx 波特率发生器时钟

WUEN_x: 唤醒使能位

全双工异步模式:

1 = 接收器正在等待下降沿, 不会接收字符, RCxIF 将被置 1, 当 RCxIF 置 1 后 WUEN_x 将被自动清 0。

0 = 接收器正常工作

半双工同步模式:无关位

ABRDEN_x: 自动波特率检测使能位

全双工异步模式:

1 = 使能自动波特率模式(完成自动波特率后清 0)

0 = 禁止自动波特率模式

半双工同步模式:无关位

11.2.2 波特率的选择

波特率的选择与状态位 SYNCx、BxRG16 和 HBRGx 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。

波特率发生器有三种时钟选择，分别是系统主时钟 SCLK、高频时钟 HFCLK 和低频时钟 LFCLK，通过比特率寄存器 BRCTLx 的 BRxCKS<1:0>选用。

式 11.1 USARTx 波特率计算公式

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

其中 m 为倍频器选择，分别为 4、16 和 64，它与状态位 SYNCx、BxRG16 和 HBRGx 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。下表列出了 m 选择表。

表 11-3 倍频器 m 选择表

配置位			BRG/USARTx 模式	倍频器 m
SYNCx	BxRG16	HBRGx		
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	
1	0	x	8 位/同步	
1	1	x	16 位/同步	4

注：x 为无关位

如下例是确定波特率和波特率误差的计算示例。

例 11-1 计算波特率误差

器件工作在时钟频率 = 16MHz，目标波特率 = 9600，全双工异步模式，8 位 BRG:

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

$$X = \frac{\text{目标波特率}}{64} - 1$$

$$= \frac{16000000}{64} - 1$$

$$= 25.042$$

$$= 25 \text{ 十进制数}$$

$$= 19H \text{ 十六进制数}$$

$$\text{计算波特率} = \frac{16000000}{64(25 + 1)}$$

$$\begin{aligned} &= 9615 \\ \text{误差} &= \frac{\text{计算波特率} - \text{目标波特率}}{\text{目标波特率}} \\ &= \frac{(9615 - 9600)}{9600} \\ &= 0.16\% \end{aligned}$$

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，见附录 5 的表格所示。

使用高波特率 (HBRGx = 1) 或 16 位 BRG (BxRG16 = 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 {EUBRGHx:EUBRGLx} 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工作时钟，可能会导致接收错误或数据丢失。为避免此问题，应检查 RCIDL 位的状态，以确保在更改工作时钟前接收操作处于空闲状态。

11.2.3 自动波特率检测

USARTx 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 RXx 输入提供时钟信号，而是由 RXx 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 BRCTLx 寄存器的 ABRDENx 位置 1 将启动自动波特率校验序列 (图 13.3)。当发生 ABRDx 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)，EUBRGLx 使用 BRG 计数器时钟递增计数，如图 13.3 所示。在第 8 位周期的末尾将在 RXx 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 EUBRGHx 和 EUBRGLx 寄存器中，ABRDENx 位被自动清 0 而 RCIF 中断标志被置 1。要清除 RCIFx 中断，需要读取 RXSDRx 中的值。RXSDRx 的内容应该被丢弃。校准不使用 EUBRGHx 寄存器的模式时，用户可通过查询 EUBRGHx 寄存器中的 0X00 验证 EUBRGLx 寄存器是否未溢出。

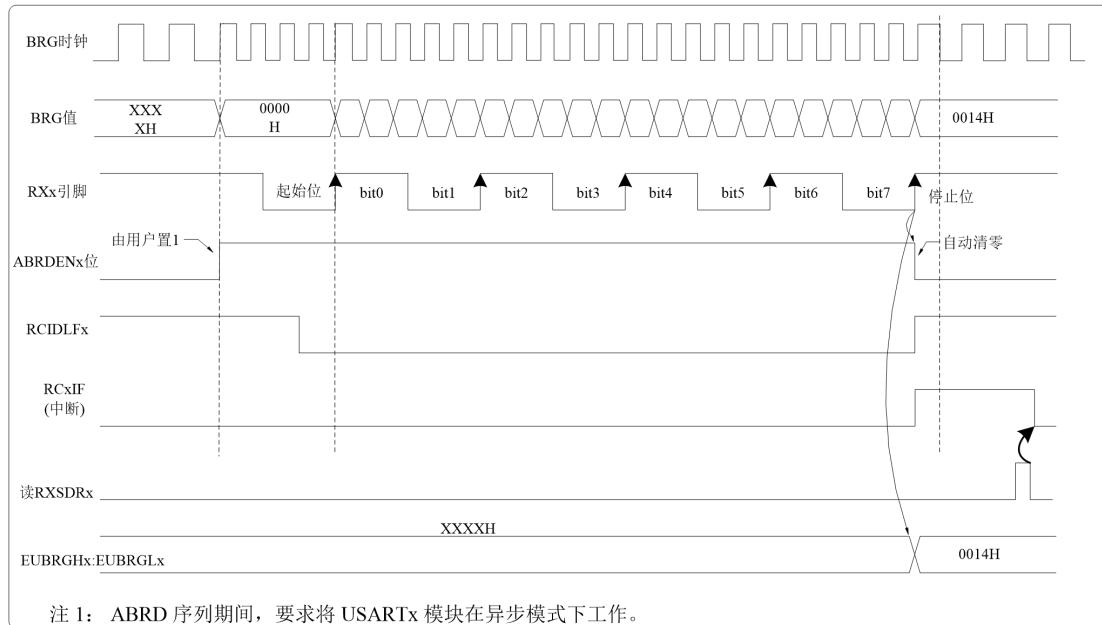


图 11.3 自动波特率检测

BRG 自动波特率时钟由 BxRG16 和 HBRGx 位决定, 如表 13-3 所示。在 ABRD 期间, EUBRGHx 和 EUBRGLx 寄存器共同用作 16 位计数器, 这与 BxRG16 位的设置无关。在校准波特率周期时, EUBRGHx 和 EUBRGLx 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WUENx位和ABRDENx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BRGx时钟源范围内。某些振荡器频率和USART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EUBRGHx:EUBRGLx 寄存器对的值中减去1。

表 11-4 BRG 计数器时钟速率

BxRG16	HBRGx	BRG 基时钟	BRG ABRD 时钟
0	0	工作时钟/64	工作时钟/512
0	1	工作时钟/16	工作时钟/128
1	0	工作时钟/16	工作时钟/128
1	1	工作时钟/4	工作时钟/32

注: 在ABRD序列期间, EUBRGLx和EUBRGHx寄存器都用作16位计数器, 与BRG16x的设置无关。

11.2.4 接收间隔字符时自动唤醒

在休眠模式下, USARTx 的所有时钟都会停止, 此时波特率发生器处于无效状态, 无法进行正确的字符接收。自动唤醒功能允许在 RX/DT 线上活动时唤醒控制器。只有 USART 工作在异步模式下时才可以使用该功能。

通过将 BRCTLx 寄存器的 WUE 位置 1, 使能自动唤醒功能, 置 1 后, 将禁止 RX/DT 上的正常序列的接收, 并且 USARTx 保持在空闲状态, 监视唤醒事件, 监视行为与 CPU 的工作模式无关。唤醒事件是指 RX/DT 线上从高电平到低电平的转换, 这与同步间隔字符或与 LIN 协议唤醒信号字符的启动条件一致。

在检测到唤醒事件时, 将会产生一个 RCIF 中断。在 CPU 正常工作的模式下, 中断会与 Q 时钟同步产生; 如果器件处于休眠模式, 则两者不同步。可通过读 RXSDRx 寄存器清除中断条件。

当 RX 线从低电平向高电平转换时, WUE 会被清零, 这就意味着“同步间隔”事件结束。此时, USART 会处于空闲模式, 等待接收下一字符。

为避免错误, 需要注意以下事项。

(1) 间隔字符

为了避免唤醒事件期间的字符错误或字符分段, 唤醒字符必须为全零。

如果使能唤醒功能, 无论数据的低电平持续时间的长短, 该功能都可以起作用。如果将 WUE 位置 1, 且收到一个有效的非零字符, 则从起始位到第一个上升沿之间的低电平时间被解释为唤醒事件, 而后续的字符将会被接收, 导致帧错误或溢出错误。

因此, 使用唤醒功能时, 发送的初始字符必须全为 0, 且持续时间必须为 10 个或更多数据位的发送时间。针对 LIN 总线建议为 13 个位时间, 而对于标准的 RS-232 器件可为任意数量的位时间。

(2) 振荡器的起振时间

在使用自动唤醒时还需要考虑振荡器的起振时间。“同步间隔”(或唤醒信号)字符必须足够长, 并且跟有足够长的时间间隔, 使选定的振荡器有足够的起振时间并使 USART 正确初始化。

(3) WUE 位

唤醒事件会产生接收中断, 将 RCIF 位置 1。在 RX/DT 的上升沿, 由硬件清零 WUE 位。通过在软件中读 RXSDRx 寄存器清除中断条件。

要保证没有丢失数据, 在将 WUE 位置 1 前, 应检查 RCIDL 位, 以验证是否正在进行接收。如果不是在进行接收, 则可将 WUE 置 1, 时器件立即进入休眠模式。

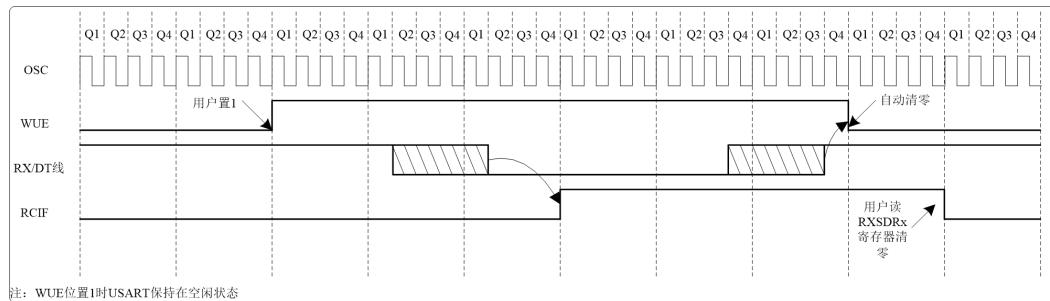


图 11.4 正常工作下的自动唤醒时序

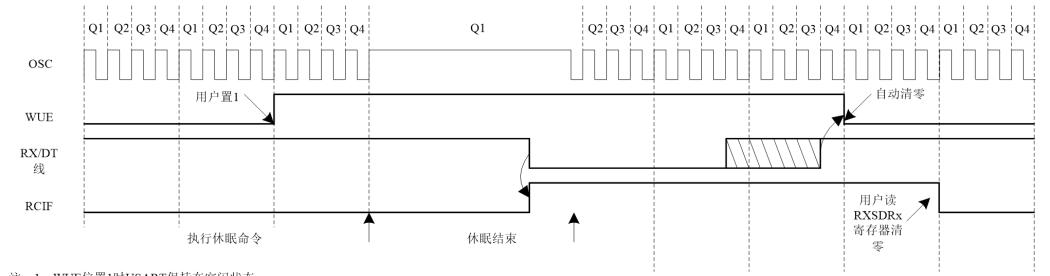


图 11.5 休眠模式下的自动唤醒时序

11.2.5 间隔符时序

USART 模块能够发送符合 LIN 总线标准的特殊间隔字符序列。发送的间隔字符包括 1 个起始位，后面跟着 12 个 0 位和一个停止位。

要发送间隔字符，要先将 TXEN 值 1，然后再将 SENDB 位置 1 (SENDB 位需要在 TXEN 位为 1 的时候才能写入)，之后对 TXSDRx 寄存器的写操作将会启动间隔字符的发送。此时写入 TXSDRx 的数据会被忽略，并会发送全 0。

在发送了相应的停止位之后，硬件会自动将 SENDB 位清零。这样用户可以在发送完间隔字符（在 LIN 规范中通常时同步字符）后将下一个要发送的字节预先装入发送 FIFO 中。TXCTLx 寄存器中的 TXSRSx 位则表明发送正在进行还是处于空闲状态。

11.2.5.1 间隔和同步发送序列

以下序列会发送一个报文帧头，包含一个间隔字符和其后的自动波特率同步字节。此序列适用于典型的 LIN 总线器件。

- 1、将 USART 配置为所需的模式
 - 2、将 TXEN 置位 1
 - 3、将 SENDB 置位 1，设置间隔字符（SENDB 需要在 TXEN 为 1 的时候才能写入）
 - 4、将无效字符装入 TXSDRx 寄存器，启动发送（该值会被忽略）
 - 5、将 55H 写入 TXSDRx 中，以便把同步字符装入发送 FIFO 缓冲中
 - 6、间隔字符发送后，硬件会将 SENDB 位复位。开始发送同步字符
- 当 TXxIF 指示 TXSDRx 为空时，下一个数据会写入 TXSDRx 中。

11.2.5.2 接收间隔字符

USART 接收间隔字符有两种方法。

第一种检测间隔字符的方法是使用 RSCTL 寄存器的 FRER 位和 RXSDR 指示接收的数据。假设已将波特率发生器初始化成预期的波特率。当

- RCIF 位置 1
- FRER 位置 1
- RXSDRx 为 00H 时

表明接收到间隔字符

第二种方法是使用自动唤醒功能。通过使能此功能，USART 将采样 RX/DT 上电平的下两个跳变，产生一个 RCxIF 中断，接收下一个数据字节，之后产生另一个中断。

注意，在间隔字符后，用户通常希望使能自动波特率检测功能。无论使用哪种方法，用户都可以在 USART 进入休眠模式之前，将 BRCTLx 寄存器的 ABDEN 位置 1.

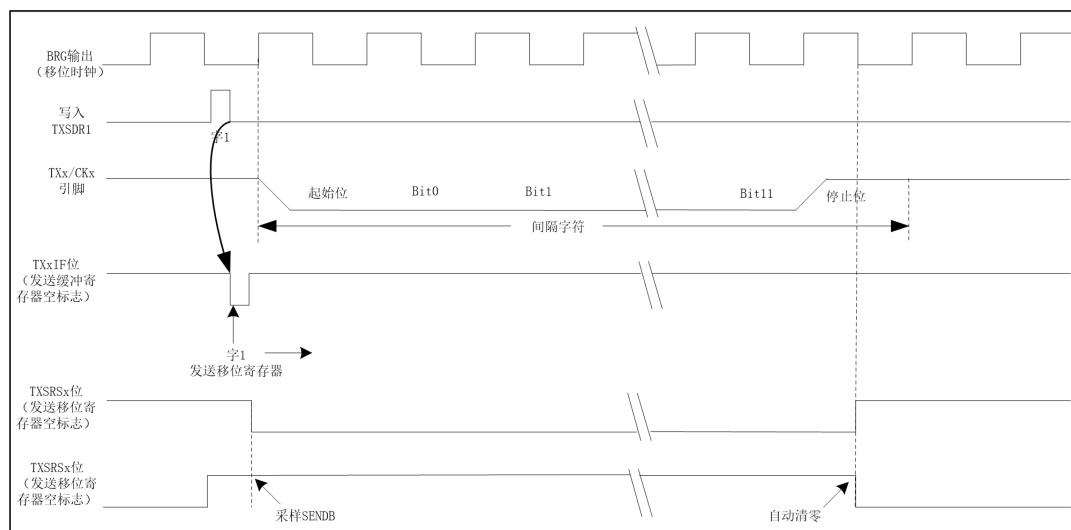


图 11.6 发送间隔字符时序

11.3 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下图所示。

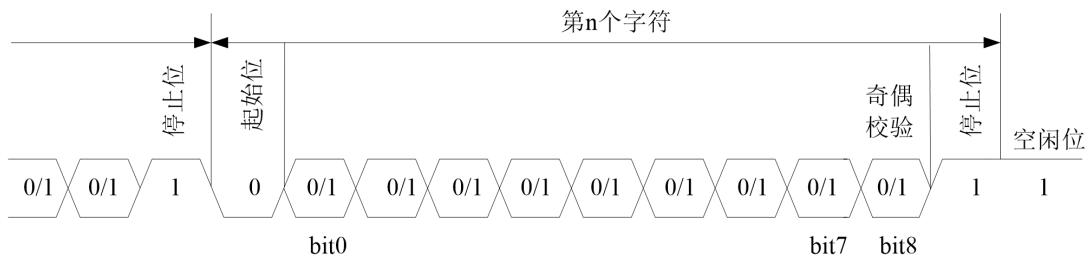


图 11.7 全双工异步通信数据结构图

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/(波特率)。片上专用 8 位/16 位波特率发生器可用于三种时钟信号和四种时钟源。请参见表 12-2 了解波特率配置示例。

USARTx 首先发送和接收低位。USARTx 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）。

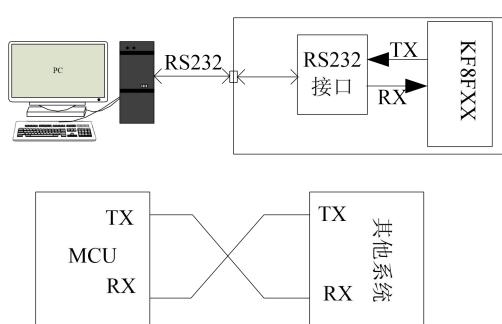


图 11.8 串口通信原理框图

11.3.1 USART 全双工发送操作

USARTx 全双工异步发送操作通过 USARTx 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 TXSDRx 发送缓冲寄存器获取数据。

通过配置如下三个控制位使能 USARTx 发送器，以用于全双工异步操作：

- 1) TXENx= 1
- 2) SYNCx = 0
- 3) SPENx= 1

假设所有其它 USARTx 控制位处于其默认状态。将 TSCTLx 寄存器的 TXENx 位置 1，使能 USARTx 发送器电路。将 TSCTLx 寄存器的 SYNCx 位清 0，将 USARTx 配置用于全双工异步操作。将 RSCTLx 寄存器的 SPENx 位置 1，使能 USARTx 并自动将 TXx/CKx 的 I/O 引脚配置为输出引脚。如果与模拟外设共用 TXx/CKx 引脚，必须清 0 相应的 ANSLx 位禁止模拟 I/O 功能。

注：1、将SPENx 位置1 会自动将RXx/DTx 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USARTx
接收器使能与否。可以通过普通端口读RXx/DTx
引脚数据，但却无法使用该端口锁存输出数据。
2、如果TXENx 使能位置1，TXIFx 发送器中断标志位会置1。

11.3.2 发送和控制状态寄存器 TSCTLx

寄存器： TSCTLx: 发送状态和控制寄存器

	bit0							
复位值	CSRSx	TxX9	TXENx	SYNCx	SENDDBx	HBRGx	TXSRSx	TX9Dx
0000 0010	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

CSRSx: 时钟源选择位
全双工异步模式：
无关位
半双工同步模式：
1 = 主模式(由 BRG 内部产生时钟)
0 = 从模式(时钟源来自外部)

TxX9: RS485 发送使能位
1 = 选择 RS485 发送
0 = 选择 8 位发送

TXENx: 发送使能位
1 = 使能发送
0 = 禁止发送

SYNCx: 串行通信模式选择位
1 = 半双工同步模式
0 = 全双工异步模式

SENDBx:	发送间隔字符位 全双工异步模式: 1 = 在下一次发送时发送同步间隔字符(完成后由硬件清 0) 0 = 同步间隔字符发送完成
	半双工同步模式: 无关位
HBRGx:	高波特率选择位 全双工异步模式: 1 = 高速 0 = 低速
	半双工同步模式: 在此模式下不使用
TXSRSx:	发送移位寄存器状态位 1 = 发送移位空 0 = 发送移位满
TX9Dx:	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。

11.3.3 发送数据

向 TXSDRx 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，TXSDRx 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 TXSDRx 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，TXSDRx 中待处理的数据将被传输到发送移位寄存器。当数据从 TXSDRx 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 USARTx 发送器且 TXSDRx 中没有待发送数据，就将 EIF2 寄存器的 TXxIF 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 TXSDRx 中有排队等待发送的新字符时，TXxIF 位才处于清 0 状态。写 TXSDRx 时，不立即清 0 TXxIF 标志位。TXxIF 在写指令后的第 2 个指令周期清 0。在写 TXSDRx 后立即查询 TXxIF 会返回无效结果。TXxIF 为只读位，不能由软件置 1 或清 0。

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

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

TSCTLx 寄存器的 TXSRSx 位指示发送移位寄存器的状态。TXSRSx 位为只读位。当发送移位寄存器为空时，TXSRSx 位被置 1，当有字符从 TXSDRx 传输到发送移位寄存器时，TXSRSx 被清 0。TXSRSx 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。

11.3.4 全双工发送的设置

1. 设置BRCTLx寄存器的BRxCKS<1:0>位选择波特率发生器的工作时钟，再根据相应的时钟信号选择相应的时钟源。
2. 初始化EUBRGHx和EUBRGLx这对寄存器以及HBRGx和BxRG16位，以获得所需的波特率（见第14.2节“波特率发生器（BRG）”）。
3. 通过将SYNCx位清0并将SPENx位置1使能全双工异步串口。
4. 如果需要9位发送，将TxX9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。
5. 将TXENx控制位置1，使能发送；这将导致TXxIF中断标志位置1。
6. 如果需要中断，将TXxIE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。
7. 若选择发送9位数据，第9位应该被装入TX9Dx数据位。
8. 将8位数据装入TXSDRx寄存器开始发送数据。

例 11-2 USART 发送数据样例程序

```

发送数据
SET  TSCTL,TXEN
MOV  R1, UART_TEMP
MOV  TXSDR, R1
NOP
NOP
NOP
JB   TSCTL,TXSRS;判断是否发生完毕?
JMP $-3

```

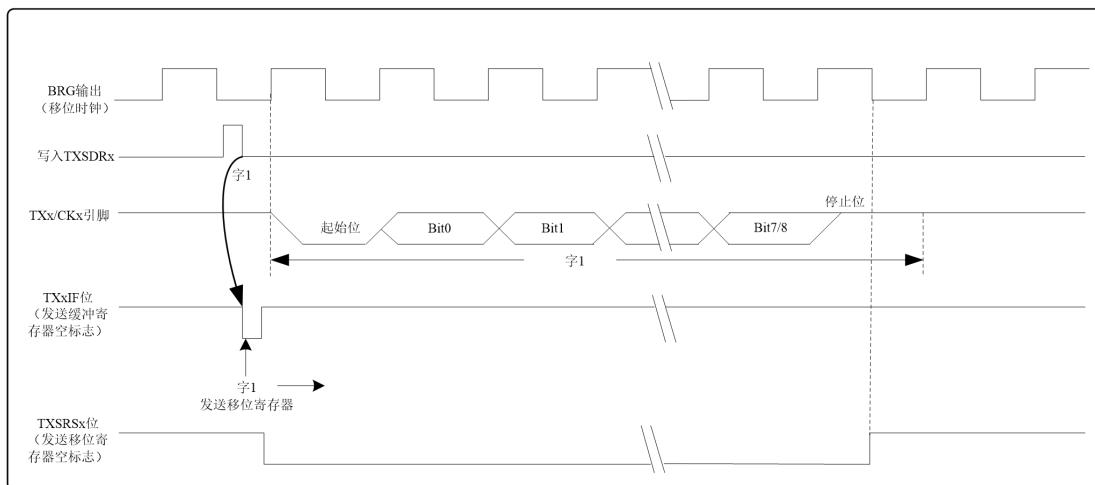


图 11.9 全双工异步发送（两字符间有空闲位）

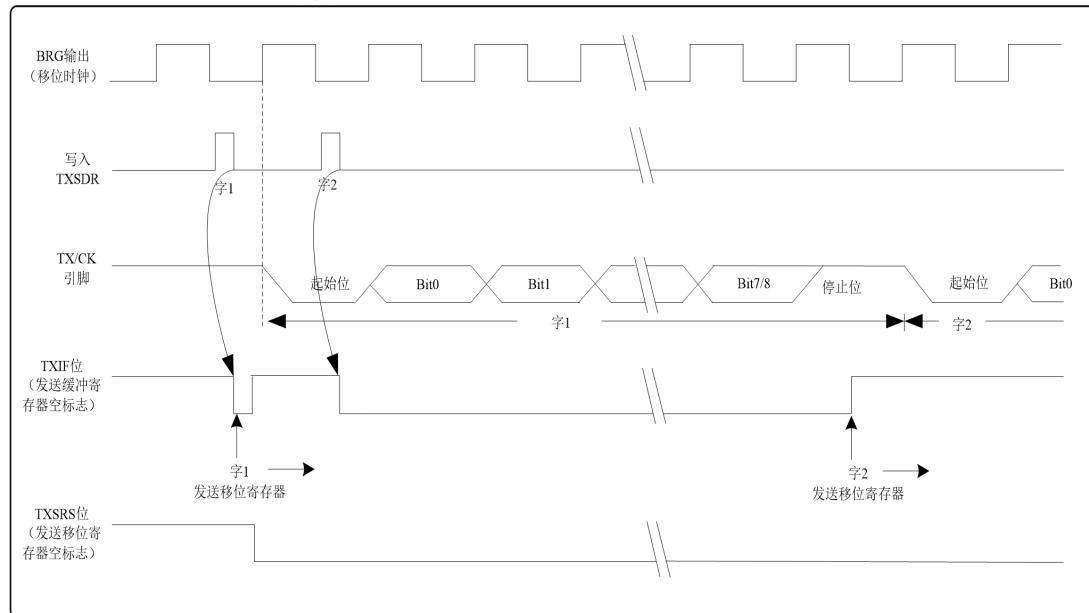


图 11.10 全双工异步发送（一字符接一字符）

11.3.5 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 14.1 给出了接收器的框图。在 RXx/DTx 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 USARTx 接收器。FIFO 缓冲器和 RSR 寄存器不能直接由软件访问。通过 RXSDRx 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。

- ◆ CRXENx= 1
- ◆ SYNCx= 0
- ◆ SPENx= 1

假设所有其它 USARTx 控制位都处于默认状态。将 RSCTLx 寄存器的 CRXENx 位置 1，使能 USARTx 接收器电路。将 TSCTLx 寄存器的 SYNCx 位清 0，配置 USARTx 以用于全双工异步操作。将 RSCTLx 寄存器的 SPENx 位置 1，使能 USARTx 并自动将 RXx/DTx 引脚配置为输入引脚。如果 RXx/DTx 引脚与模拟外设共用，必须清 0 相应的 ANSL 位禁止模拟 I/O 功能。

注：当将 SPENx 位置 1，TXx/CKx 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 USART 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 TXx/CKx 引脚用作通用输出引脚。

11.3.6 接收状态和控制寄存器 RSCTLx

寄存器: RSCTLx: 接收状态和控制寄存器x

								bit0
								bit7
复位值	SPENx	RxX9	SRXENx	CRXENx	ADRENx	FRERx	OVFERx	RX9Dx
0000 000x	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

SPENx: 串行口使能位

- 1 = 使能串行口(将 RXx/DTx 和 TXx/CKx 引脚配置为串行口引脚)
0 = 禁止串行口(保持为复位状态)

RxX9: 9 位接收使能位

- 1 = 选择 9 位接收
0 = 选择 8 位接收

SRXENx: 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:

- 1 = 使能单字节接收
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

CRXENx: 连续接收使能位

全双工异步模式:

- 1 = 使能接收器
0 = 禁止接收器
半双工同步模式:

1 = 使能连续接收, 直到使能位 CRXENx 清 0(CRXENx 改写 SRXENx)

0 = 禁止连续接收

ADRENx: 地址检测使能位

RS-485 全双工异步模式(RxX9 = 1):

- 1 = 使能地址检测、允许中断, 当 RSR<8>置 1 时装入接收缓冲区
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(RxX9=0):

无关位

FRERx: 帧错误位

- 1 = 帧错误(可通过读 RXSDRx 寄存器刷新该位, 并接收下一个有效字节)
0 = 无帧错误

OVFERx: 溢出错误位

- 1 = 溢出错误(可通过清 0 CRXENx 位来清 0 该位)
0 = 无溢出错误

RX9Dx: 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。

11.3.7 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 13.3.2.3 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 USARTx 的接收 FIFO 缓冲器并将 EIE2 寄存器的 RCxIF 中断标志位置 1。通过读 RXSDRx 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。请参见第 11.3.2.3 节“接收错误”获得有关溢出错误的更多相关信息。

只要使能 USARTx 接收器且在接收 FIFO 缓冲器中没有未读数据，EIF2 寄存器中的 RCxIF 中断标志位就会清 0。RCIF 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 RCxIF 中断：

- ◆ EIE2 寄存器的 RCxIE 中断允许位
- ◆ INTCTL 寄存器的 PUIE 外设中断允许位或低优先级中断允许位 AIEL
- ◆ INTCTL 寄存器的 AIE 全局中断允许位或高优先级中断允许位 AIEH

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 RCxIF 中断标志位置 1。

11.3.8 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。帧错误是指未在预期的时间内接收到停止位。由 RSCTLx 寄存器的 FRERx 位获取帧错误状态。FRERx 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 RXSDRx 寄存器之前读 FRERx 位。FRERx 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 (FRERx=1) 并不会阻止接收更多的字符。无需清 0 FRERx 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 RSCTLx 寄存器的 SPENx 位会复位 USARTx，并强制清 0 FRERx 位。清 0 RSCTLx 寄存器的 CRXENx 位不影响 FRERx 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 RCSDRx 不会清零 FRERx 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，RSCTLx 寄存器的 OVFERx 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 RSCTLx

寄存器的 CRXENx 位或通过清 0 RSCTLx 寄存器的 SPENx 位使 USARTx 复位来清除错误。

11.3.9 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 RSCTLx 寄存器的 ADRENx 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，当接收到的地址数据与 UARMTMThx 寄存器内预设的值匹配时，RCxIF 中断标志位将置 1；所有其它字符将被忽略。ADRENx 位不会自动清零，当检测到地址匹配后，如果要进行数据传输，需要将 ADRENx 位手动清零。

寄存器: UARMTMThx: UART 地址匹配设置寄存器 x

复位值 0000 0000	bit7								bit0
	ADMx7	ADMx6	ADMx5	ADMx4	ADMx3	ADMx2	ADMx1	ADMx0	
	R/W								

ADMx<7:0>: USARTx 地址匹配功能设置位

注: UARMTMTH1 寄存器地址为 12DH; UARMTMTH2 寄存器地址为 247H。

11.3.10 全双工接收的设置

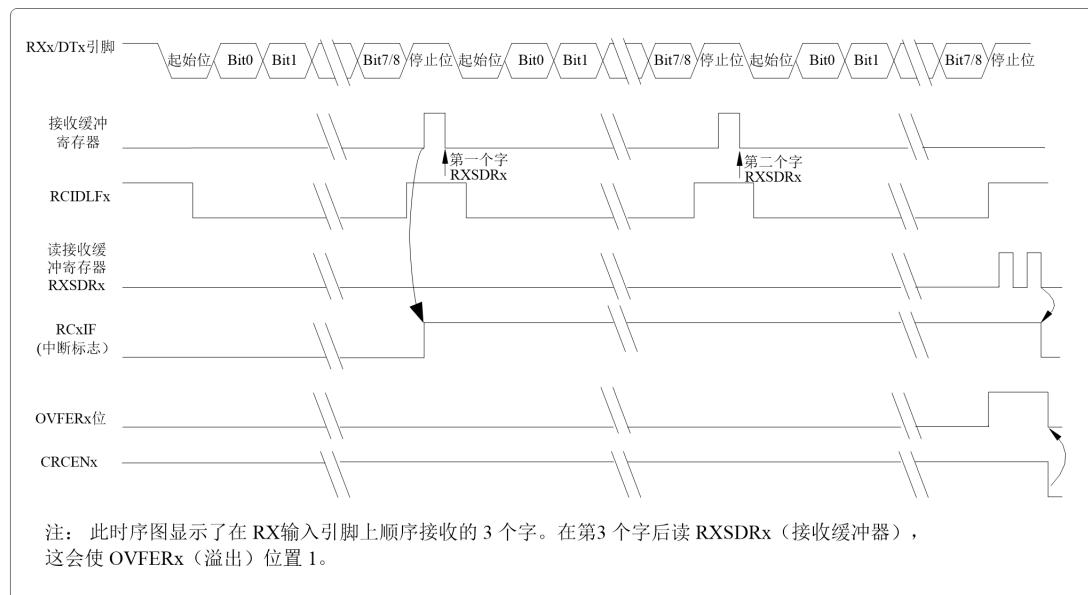
1. 设置 BRCTLx 寄存器的 BRxCKS<1:0>位选择波特率发生器的工作时钟，再根据相应的时钟信号选择相应的时钟源。
2. 初始化 {EUBRGHx:EUBRGLx} 这对寄存器以及 HBRGx 和 BxRG16 位，以获得所需的波特率（见第 14.2 节“波特率发生器（BRG）”）。
3. 将 SPENx 位置 1，使能串行端口。必须清 0 SYNCx 位以执行全双工异步操作。
4. 如果需要中断，将 EIE2 寄存器中的 RCxIE 位和 INTCTLx 寄存器的 AIE 和 PUIE 位置 1。如果使用中断优先级则将 IPEN 和 PTX 均置 1
5. 如果需要接收 9 位数据，将 RxX9 位置 1。
6. 将 CRXENx 位置 1 使能接收。
7. 当一个字符从 RSR 传输到接收缓冲器时，将 RCxIF 中断标志位置 1。如果 RCxIE 中断允许位也置 1 还将产生中断。
8. 读 RSCTLx 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）。
9. 读 RXSDRx 寄存器，从接收缓冲器获取接收到的 8 个低数据位。
10. 如果发生溢出，通过清 0 CRXENx 接收器使能位清 0 OVFERx 标志。

例 11-3 USART1 数据接收样例程序

```

接收(USART1为例)
LOOP_USART
    NOP
    NOP
    MOVB #0X01          ;切换至1区
    SET    RSCTL1,CRXEN1 ;使能接收器
    JNB    RSCTL1,OVFER1 ;检测溢出错误
    CLR    RSCTL1, CRXEN1
    MOVB #0X00          ;切换至0区
    JB     EIF2, RXIF1   ;接收数据的标志
    JMP    LOOP_USART
    CLR EIF2, RXIF1
    MOVB #0X01          ;切换至1区
    MOV R0, RXSDR1
    MOV UART_TEMP, R0    ;存储接收到的数据

```


图 11.11 全双工异步接收
11.3.11 RS-485 发送/接收

USART_x 支持 RS-485 模式 9 位发送。当 TSCTL_x 寄存器的 TxX9 位置 1 时，USART_x 将移出每个待发送字符的 9 位。TSCTL_x 寄存器的 TX9D_x 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_x 之前，写 TX9D_x 数据位。在写入 TXSDR_x 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“地址检测”获得有关地址模式的更多信息。

USART_x 支持 RS-485 模式 9 位接收。将 RSCTL_x 寄存器的 RxX9 位置 1 时，USART_x

将接收到的每个字符的 9 位移入 RSR。RSCTLx 寄存器的 RX9Dx 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 RXSDRx 中的低 8 位之前，读取 RX9Dx 数据位。

11.3.11.1 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 EUBRGHx 和 EUBRGLx 这对寄存器以及 HBRGx 和 BxRG16 位，以获得所需的波特率（见“波特率发生器（BRG）”）。
2. 将 SPENx 位置 1，使能串行端口。必须清 0 SYNCx 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 RCxIE 位和 INTCTL 寄存器的 AIE 和 PUIE 位置 1。如果使用优先级中断，则将 IPEN 和 PRX 均置 1。
4. 将 RxX9 位置 1，使能 9 位数据接收。
5. 将 ADRENx 位置 1，使能地址检测。
6. 将 CRXENx 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 RCxIF 中断标志位置 1。如果 RCxIE 中断允许位也置 1 还将产生中断。
8. 读 RSCTLx 寄存器获取错误标志位。第 9 个数据位始终置 1。
9. 读 RXSDRx 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。
11. 如果发生溢出，通过清 0 CRXENx 接收器使能位清 0 OVFERx 标志。
11. 如果是对当前器件寻址，将 ADRENx 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。

11.3.12 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“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。

11.4 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器件和一个或多个从动器件的系统中。主控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。USARTx 既可以作为主控器件，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。

11.4.1 USARTx 半双工主控模式

下列位用来将 USARTx 配置为半双工同步主控操作：

- SYNCx = 1
- CSRSx = 1
- SRXENx = 0 (用于发送); SRXENx = 1 (用于接收)
- CRXENx = 0 (用于发送); CRXENx = 1 (用于接收)
- SPENx = 1

将 TSCTLx 寄存器的 SYNCx 位置 1，可将 USARTx 配置用于半双工同步操作。将 TSCTLx 寄存器的 CSRSx 位置 1，将器件配置为主控器件。将 RSCTLx 寄存器的 SRXENx 和 CRXENx 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 RSCTLx 寄存器的 SPENx 位置 1，使能 USARTx。如果 RXx/DTx 或 TXx/CKx 引脚与模拟外设共用，必须清 0 相应的 ANSL 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。配置为主控器件的器件在 TXx/CKx 引脚发送时钟信号。当 USARTx 被配置为半双工同步发送或接收操作时，TXx/CKx 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 BRCTLx 寄存器的 SCKPSx 位选择时钟极性。将 SCKPSx 位置 1 将时钟空闲状态设置为高电平。当 SCKPSx 位置 1 时，数据在每个时钟的下降沿发生改变。清 0 SCKPSx 位，将时钟空闲状态设置为低电平。当清 0 SCKPSx 位时，数据在每个时钟的上升沿发生改变。

11.4.2 半双工主控发送

由器件的 RXx/DTx 引脚输出数据。当 USARTx 配置为半双工同步主控发送操作时，器件的 RXx/DTx 和 TXx/CKx 输出引脚自动使能。

向 TXSDRx 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 TXSDRx 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 TXSDRx 中的数据会被立即传输到发送移位寄存器。当字符从 TXSDRx 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 EUBRGHx 和 EUBRGLx 这对寄存器以及 HBRGx 和 BxRG16 位，以获得所需的波特率（见“波特率发生器”）。
2. 将 SYNCx、SPENx 和 CSRSx 位置 1，使能半双工同步主控串行端口。
3. 将 SRXENx 和 CRXENx 位清 0，禁止接收模式。
4. 将 TXENx 位置 1 使能发送模式。
5. 如果需要发送 9 位字符，将 TxX9 置 1。
6. 若需要中断，将 EIE2 寄存器中的 TXxIE 位，以及 INTCTL 寄存器中的 AIE 和 PUIE 位置 1。如果使用优先级中断，则将 IPENx 和 PTXx 均置 1。
7. 如果选择发送 9 位字符，应该将第 9 位数据装入 TX9Dx 位。
8. 通过将数据装入 TXSDRx 寄存器启动发送。（请参考例 14.2）

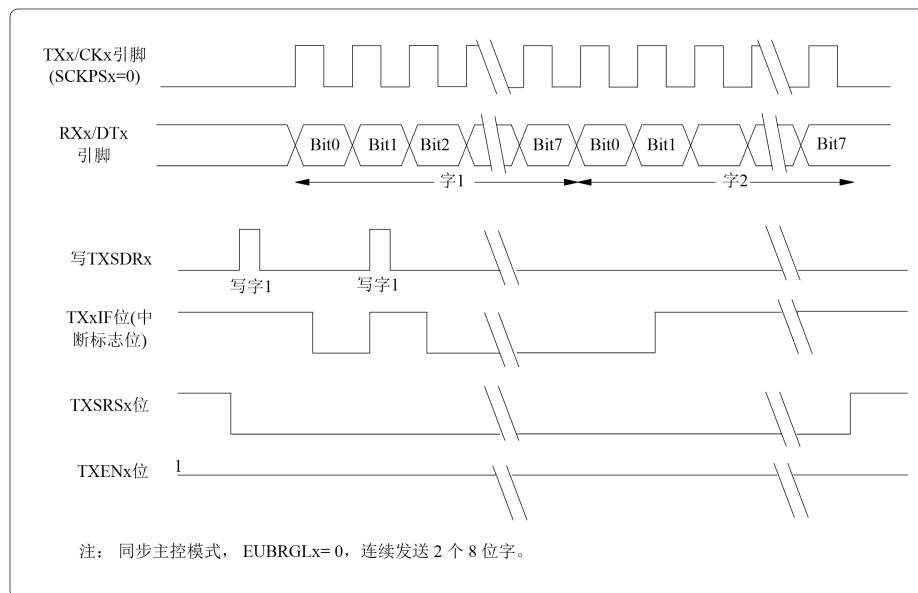


图 11.12 半双工同步发送 (SCKPSx=0)

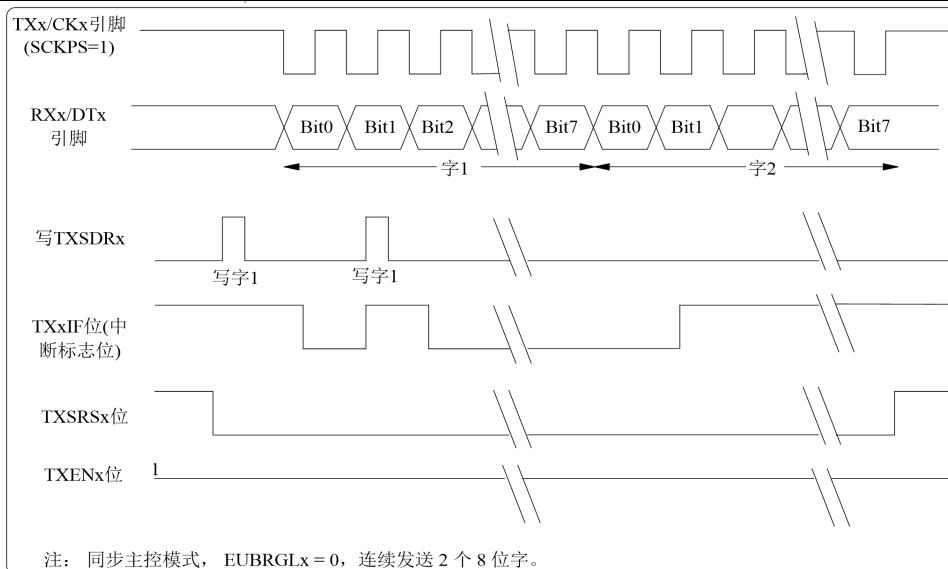


图 11.13 半双工同步发送(SCKPSx=1)

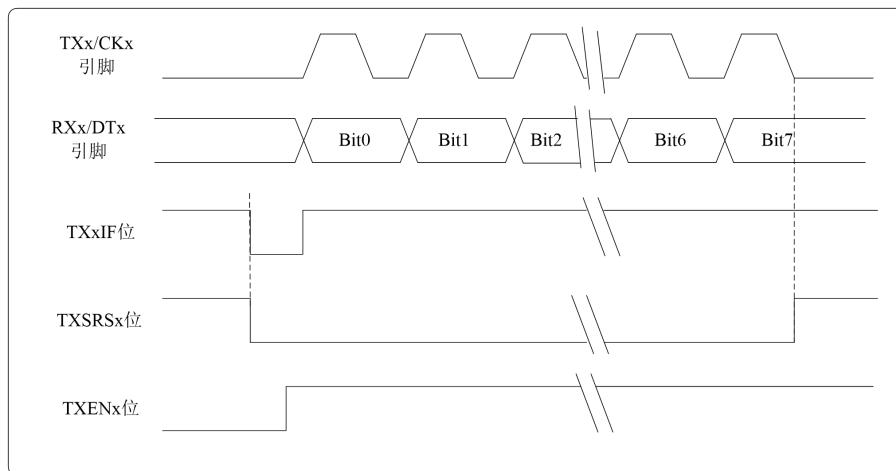


图 11.14 半双工同步发送（通过 TXENx）

11.4.3 半双工主控接收

在 RXx/DTx 引脚接收数据。当 USARTx 配置为半双工同步主控接收时，自动禁止器件的 RXx/DTx 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（RSCTLx 寄存器的 SRXENx 位）或连续接收使能位（RSCTLx 寄存器的 CRXENx 位）置 1 使能接收。

当将 SRXENx 置 1，CRXENx 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 SRXENx 位。当 CRXENx 置 1 时，将产生连续时钟，直到清 0 CRXENx 为止。如果 CRXENx 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 SRXENx 和 CRXENx 都置 1，则当第一个字符传输完成时，SRXENx 位被清 0，CRXENx 保持。

将 SRXENx 或 CRXENx 置 1，启动接收。在 TXx/CKx 时钟引脚信号的下降沿采样 RXx/DTx 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 RCxIF 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 RXSDRx 读取。只要接收 FIFO 缓冲器中仍有未读字符，则

RCxFIF 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 TXx/CKx 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，TXx/CKx 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 RXSDRx 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，RSCTLx 寄存器的 OVFERx 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 OVFERx 位清 0。如果发生溢出时，SRXENx 位为置 1 状态，CRXENx 位为清 0 状态，则通过读 RXSDRx 寄存器清除错误。如果溢出时，CRXENx 为置 1 状态，则可以清 0 RSCTLx 寄存器的 CRXENx 位或清 0 SPENx 位以复位 USARTx，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 EUBRGHx:EUBRGLx 寄存器。按需要将 HBRGx 和 BRG16x 位置 1 或清 0，以获得所需的波特率。
2. 将 SYNCx、SPENx 和 CSRSx 位置 1 使能半双工同步主控串行端口。
3. 确保将 CRXENx 和 SRXENx 位清 0。
4. 如果使用中断，将 INTCTL 寄存器的 AIE 和 PUIE 位置 1，并将中断使能寄存器的 RCxIE 位也置 1。如果使用优先级中断，则将 IPEN 和 PRXx 均置 1。
5. 如果需要接收 9 位字符，将 RxX9 位置 1。
6. 将 SRXENx 位置 1，启动接收，或将 CRXENx 位置 1 使能连续接收。
7. 当字符接收完毕后，将 RCxIF 中断标志位置 1。如果允许位 RCxIE 置 1，还会产生一个中断。
8. 读 RSCTLx 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 RXSDRx 寄存器获取接收到的 8 位数据。

如果产生溢出错误，清 0 RSCTLx 寄存器的 CRXENx 位或清 0 SPENx 以复位 USARTx 来清除错误。（请参考第 12.3.2.5 节例 12.3）

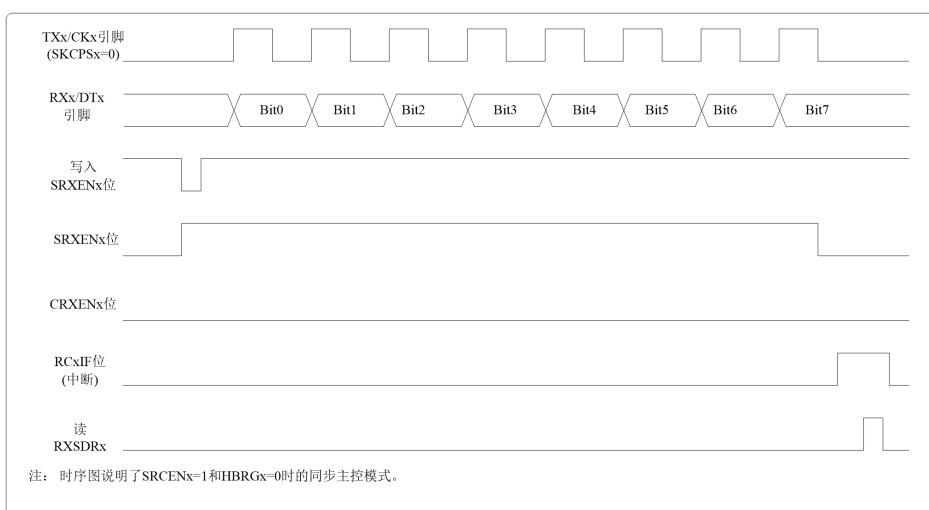


图 11.15 半双工同步接收（主控模式，SRXENx=1，SCKPSx=0）

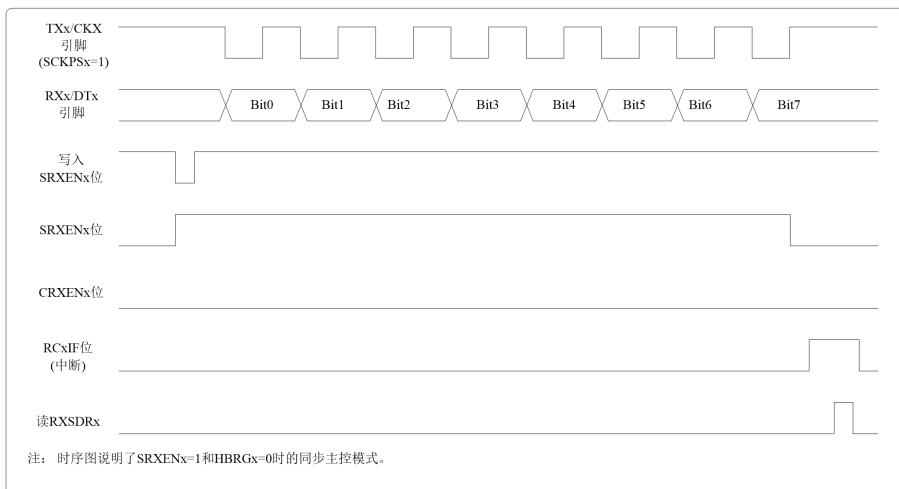


图 11.16 半双工同步接收（主控模式，SRXENx=1，SCKPSx=1）

11.4.4 USART 半双工从动模式

下列位用来将 USARTx 配置为半双工同步从动操作：

- SYNCx=1
- CSRSx=0
- SRXENx=0 (用于发送); SRXENx = 1 (用于接收)
- CRXENx=0 (用于发送); CRXENx = 1 (用于接收)
- SPENx=1

将 TSCTLx 寄存器的 SYNCx 位置 1，可将器件配置用于半双工同步操作。将 TSCTLx 寄存器的 CSRSx 位置 1，将器件配置为从动器件。将 RSCTLx 寄存器的 SRXENx 和 CRXENx 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 RSCTLx 寄存器的 SPENx 位置 1，使能 USARTx。如果 RXx/DTx 或 TXx/CKx 引脚与模拟外设共用，必须清 0 相应的 ANSL 位禁止模拟 I/O 功能。

11.4.5 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“半双工主控发送”）。

如果向 TXSDRx 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 TXSDRx 寄存器中。
3. TXxIF 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 TXxIF 置 1。
5. 如果 PUIE 和 TXxIE 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 SYNCx 和 SPENx 位置 1 并将 CSRSx 位清 0。
2. 将 CRXENx 和 SRXENx 位清 0。
3. 如果使用中断，将 INTCTL 寄存器的 AIE 和 PUIE 位置 1，并将中断使能寄存器的 TXxIE 位也置 1。如果使用优先级中断，则将 IPEN 和 PTXx 均置 1。
4. 如果需要发送 9 位数据，将 TxX9 位置 1。
5. 将 TXENx 位置 1 使能发送。
6. 若选择发送 9 位数据，将最高位写入 TX9Dx 位。

将低 8 位数据写入 TXSDRx 寄存器开始传输。（请参考例 12.2）

11.4.6 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“半双工主控接收”）。

- 休眠模式
- CRXENx 位总是置 1，因此接收器不能进入空闲状态。
- SRXENx 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 CRXENx 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 RXSDRx 寄存器。如果将 RCxIE 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 SYNCx 和 SPENx 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 RCxIE 位也置 1。如果使用优先级中断，则将 IPENx 和 PRXx 均置 1。
3. 如果需要接收 9 位字符，将 RxX9 位置 1。
4. 将 CRXENx 位置 1，使能接收。
5. 当接收完成后，将 RCxIF 位置 1。如果 RCxIE 已置 1，还会产生一个中断。
6. 如果使能 9 位模式，从 RSCTLx 寄存器的 RX9Dx 位获取最高位。
7. 读 RXSDRx 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 RSCTLx 寄存器的 CRXENx 位或清 0 SPENx 位以复位 USARTx 来清除错误。（请参考例 12.3）

11.4.7 USART 半双工 RS-485 模式

请参考第 12.3.11 节 RS-485 发送/接收。

11.5 USART 工作在休眠模式下

当 USART 的工作时钟选择为低频时钟且打开 USART 的休眠使能位时，将允许 USART 工作在休眠模式下。

11.5.1 USART 休眠使能寄存器 USLPEN

寄存器： USLPENx:USARTx休眠时能寄存器x

		bit7										bit0							
		SLPENx	-	-	-	-	-	-	-			-	-	-	-	-	-		
复位值	0----	R/W	U	U	U	U	U	U	U			U	U	U	U	U	U		

SLPENx： USARTx 休眠使能控制位

SLPENx=1 休眠使能，允许 USARTx 在休眠模式下继续收发数据

SLPENx=0 禁止 USARTx 在休眠模式下收发数据

注: x=1 或 2, USLPEN1 寄存器地址为 127H, USLPEN2 寄存器地址为 248H。

11.6 USART 单线通信模式

USART 单线通信模式 (半双工异步模式) 通过 UPINSETx 寄存器的 USLMx 位置 1 使能; 在使能 USART 单线通信模式之前, 须确保 TSCTLx 寄存器的 SYNCx 位保持清零状态。

USART 单线通信模式下, USART 相关引脚特性如下:

- RX/DT 引脚不再使用, 总是被释放, 即可作为通用 I/O 口使用;
- TX/CK 引脚作为数据通信脚;
- 当 USART 配置为单线通信接收时, TX/CK 引脚一直处于接收状态;
- 当 USART 配置为单线通信发送时:
 - 当有数据发送时, TX/CK 引脚处于数据发送状态;
 - 当无数据发送时 (空闲状态), TX/CK 引脚被释放, 表现为通用 I/O 口, 因此在使用前需要将 TX/CK 引脚配置为数字输出口, 并输出高电平 (根据 USART 通信协议, 当 USART 处于空闲状态时, TX/CK 被拉高)。

用户可通过软件来管理线上冲突。

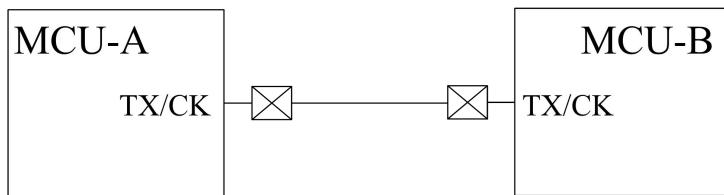


图 11.17 USART 单线通信外接连线示意图

11.6.1 UPINSET1 和 UPINSET2 寄存器

UPINSETx 寄存器中的 USLMx 位用于使能 USARTx 的单线通信模式。

寄存器: UPINSET1:USART1引脚配置寄存器(地址: 150H)

复位值	bit7								bit0
	USLM1	UPSEL1	-	-	-	-	-	-	
0000 0000	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

USLM1: USART1 单线通信模式使能位

USLM1 =1 使能串口单线通信模式

USLM1 =0 禁止串口单线通信模式

UPSEL1: 保留位

寄存器: UPINSET2:USART2引脚配置寄存器(地址: 23FH)

复位值	bit7								bit0
	USLM2	UPSEL2	-	-	-	-	-	-	
0000 0000	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

USLM2: USART2 单线通信模式使能位

USLM2 =1 使能串口单线通信模式

USLM2 =0 禁止串口单线通信模式

UPSEL2: 保留位

11.7 7816 模式

7816 模式基于 ISO/IEC 7816-3 标准，利用此功能可与其他 7816 设备进行通信。按照 7816 协议标准，7816 基本接口时序如图所示：

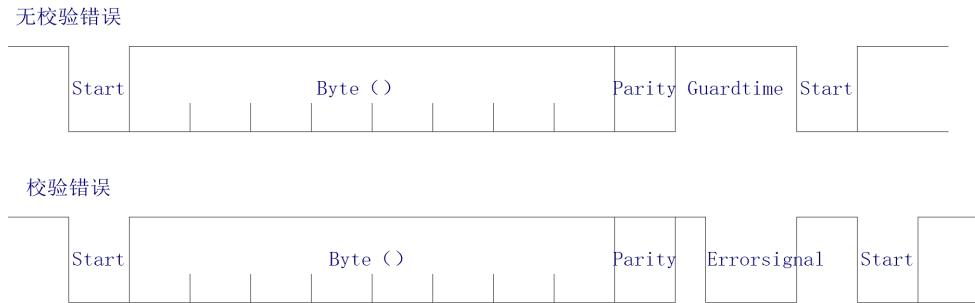


图 11.18 7816 接口时序

- 1, 一个起始位后跟 8 个数据位以及一个校验位，以 2etu 的 guard time 结束。
- 2, 第 10.5 个 etu 接收电路校验接收数据，如果正确，插入 2etu 的 guard time，确保数据长度为 12etu，完成数据发送；若校验错误，则在第 10.5etu 拉低 IO，产生 error signal。error signal 长度可从 1etu、2etu、1.5etu 中选择。
- 3, 第 11 个 etu 时发送电路未采样到 error signal，则说明发送数据正确，数据发送完成。
- 4, 若第 11 个 etu 发送电路采样到 error signal，则说明发送数据错误，等待 2 个 etu 后重发数据。

11.7.1 7816 模式发送

11.7.1.1 发送设置

通过配置如下控制位使能 USART2 发送器，以用于 7816 模式发送操作：

- 1) U7816EN2 = 1 使能 7816 模式
- 2) SPEN2 = 1 使能串口
- 3) SYNC2 = 0 选择异步模式
- 4) CLKOUT2 = 1 使能 7816 时钟输出
- 5) TxX9 = 1 选择 9 位数据发送
- 6) TX9DSEL2 = 1 选择自动生成奇偶校验码
- 7) STOP2 = 0 选择 2 位停止位
- 8) TXEN2 = 1 使能发送端
- 9) TPAR2、TINV2、TCONV2、BGTEN2、PSEL2、TREPEN2、TREP2、EGT2 等控制寄存器视情况进行选择
- 10) 其他设置请参考 12.3.1.3

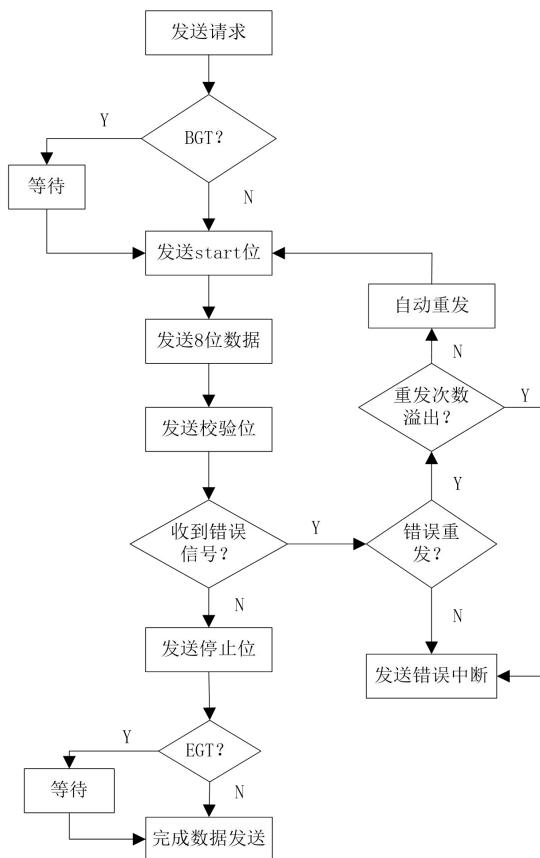


图 11.19 数据发送流程

11.7.2 7816 模式接收

通过配置如下控制位使能 USART2 接收器，以用于 7816 接收操作：

- 1) CRXEN2 = 1 使能接收端
- 2) U7816EN2 = 1 使能 7816 模式
- 3) SPEN2 = 1 使能串口
- 4) SYNC2 = 0 选择异步模式
- 5) R2X9 = 1 选择 9 位数据接收
- 6) CLKOUT2 = 1 使能 7816 时钟输出

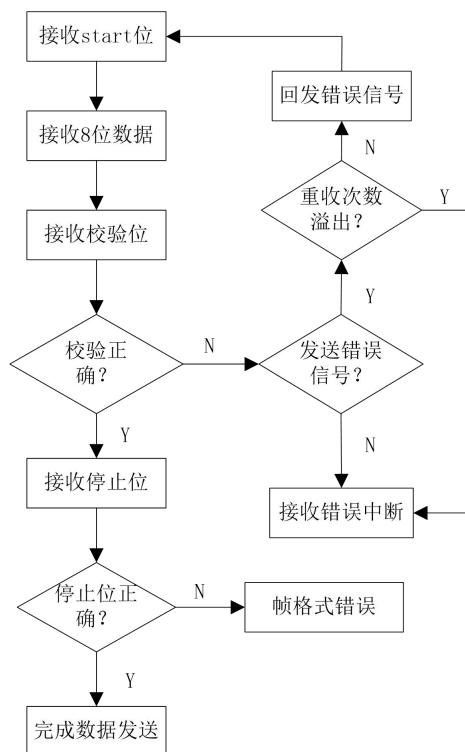


图 11.20 数据接收流程

11.7.3 7816 模式寄存器

11.7.3.1 7816 发送控制寄存器 U7816TXCTL2

寄存器: U7816TXCTL2: 7816发送控制寄存器2(地址:242H)

复位值	bit7	U7816EN2	CLKOUT2	TX9DSEL2	STOP2	TPAR2	TINV2	TCONV2	bit0	BGTEN2
0001 0000		R/W	R/W	R/W	R/W	R/W	R/W	R/W		R/W

U7816EN2: 7816 模式使能控制

- 1 = 使能 7816 模式
- 0 = 禁止 7816 模式

CLKOUT2: 7816 时钟输出使能控制

- 1 = 使能时钟输出
- 0 = 禁止时钟输出

TX9DSEL2: TX9D 位数据选择

- 1 = 选择自动生成奇偶校验
- 0 = 选择软件输入

STOP2: 停止位宽度选择

- 1 = 停止位宽度为 1etu
- 0 = 停止位宽度为 2etu

TPAR2: 发送时奇偶校验选择

- 1 = 奇校验
- 0 = 偶校验

TINV2: 7816 数据发送编码方式选择

- 1 = 发送反相电平
- 0 = 发送正相电平

TCONV2: 发送次序选择

- 1 = 先发送 MSB
- 0 = 先发送 LSB

BGTEN2: BGT(block guard time)(块保护时间, 为两个相对方向发送的连续字符的起始沿之间的最短时间, 因此一个已接收块的最后一个字符与一个被传输块的第一个字符之间的延迟至少应为 BGT)控制位, 控制接收到发送之间是否插入 BGT

- 1 = 插入 BGT, 宽度为 22etu
- 0 = 不插入 BGT

注: (1) TX9DSEL2、STOP2、TPAR2、TCONV2 在 7816 模式禁止时也可使用;

(2) 第九位需要软件输入时, 必须先把 TX9DSEL2 置 0, 再输入 TX9 的值, 否则可能会出错;

(3) 要使用 BGTEN2 功能时, 必须保证 TXEN2 同时也使能, 否则 BGT 不能正常工作。

11.7.3.2 7816 接收控制寄存器 U7816RXCTL2

寄存器: U7816RXCTL2: 7816接收控制寄存器2(地址:241H)

复位值	bit7	ERSW21	ERSW20	RPAR2	RINV2	RCONV2	-	-	bit0	PAREF2
0000 0000		R/W	R/W	R/W	R/W	R/W	U	U		R

ERSW2[1:0]: error signal 宽度选择

01 =1etu

00 =2etu

1X =1.5etu

RPAR2: 接收时奇偶校验选择

1 = 奇校验

0 = 偶校验

RINV2: 数据接收编码方式选择

1 = 接收反相电平

0 = 接收正相电平

RCONV2: 接收次序选择

1 = 先接收 MSB

0 = 先接收 LSB

PAREF2: 奇偶校验错误标志位

1 = 发送或接收上以字节数据奇偶校验错误

0 = 发送或接收上以字节数据奇偶校验正确

注: RINV2、RCONV2 在 7816 模式禁止时也可使用。

11.7.3.3 7816 控制寄存器 U7816CTL2

寄存器: U7816CTL2: 7816控制寄存器2(地址:240H)

复位值	bit7	PSEL2	-	TREPEN2	TREP21	TREP20	RREPEN2	RREP21	bit0	RREP20
0000 0000		R/W	U	R/W	R/W	R/W	R/W	R/W		R/W

PSEL2: 数据通道选择 (保留位)

TREPEN2: 重发使能控制

1 = 收到校验错误信号后重发, 达到最大次数后进入中断

0 = 收到校验错误信号后禁止重发, 直接中断

TREP2[1:0]: 重发的最大次数

TREP2=1-3

RREPEN2: 重收使能控制

1= 奇偶校验错误后重收数据, 达到最大次数后进入中断

0= 奇偶校验错误后直接进入中断

RREP2[1:0]: 重接收的最大次数

RREP2=1-3

11.7.3.4 预分频控制寄存器 CLKDIV2

寄存器: CLKDIV2: 预分频控制寄存器2(地址:243H)

复位值	CLKDIV2 7	CLKDIV2 6	CLKDIV2 5	CLKDIV2 4	CLKDIV2 3	CLKDIV2 2	CLKDIV2 1	CLKDIV2 0	bit0
0000 0000	R/W	R/W							

CLKDIV2[7:0]:7816 工作时钟和引脚输出时钟控制

当 CLKDIV1 的值不为 0 时, 7816 工作时钟 F_{7816} 和系统时钟 SCLK 之间的关系为:

$$F_{7816} = SCLK/[2 \times (CLKDIV2+1)]$$

当 CLKDIV2 的值为 0 时, F_{7816} 和 SCLK 关系为: $F_{7816} = SCLK$

注: 7816 协议规定 7816 工作时钟频率范围为 1-5M。

11.7.3.5 EGT 控制寄存器 EGTCTL2

寄存器: EGTCTL2: EGT控制寄存器2(地址:244H)

复位值	EGT27	EGT26	EGT25	EGT24	EGT23	EGT22	EGT21	EGT20	bit0
0000 0000	R/W	R/W							

EGT2[7:0]: 发送时插入的 EGT(extra guard time)宽度 (单位 etu)

0 = 发送时不插入 EGT

1-255 = 发送时插入 EGT, 宽度为 EGT2 的值

注: EGTCTL2 在发送无校验错误时正常工作, 若有校验错误时无论为何值均不插入 EGT。

12 运算放大器模块

KF8F513X 具有 1 个 4 路正向输入端可选的带校准的运算放大器，运放模块通过与外部元件连接实现其放大等作用，内部框图如图 14.1 所示。

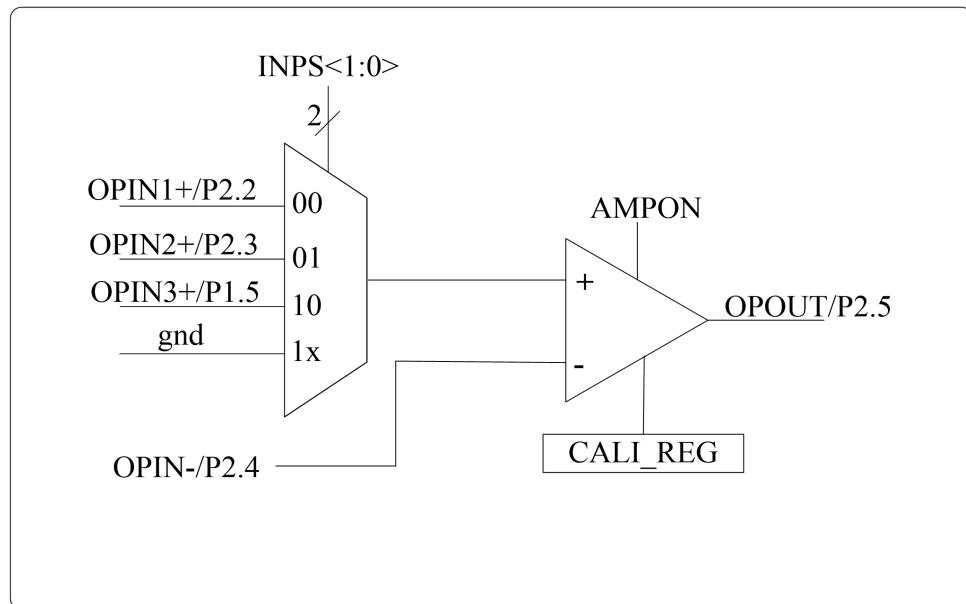


图 12.1 运放内部框图

12.1 与运放有关的寄存器

表 12-1 与运放相关的寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
1BH	AMPCTL	AMPCA LD	-	INPS1	INPS0	-	-	AMPCA LEN	AMPON
20H	AMPDT	AMPDT 7	AMPDT 6	AMPDT 5	AMPDT 4	AMPDT 3	AMPDT 2	AMPDT 1	AMPDT 0

12.1.1 运放控制寄存器

寄存器: AMPCTL: 运放控制寄存器(1BH)

bit7								bit0
复位值 1000 0000	AMPCAL D	-	INPS1	INPS0	-	-	AMPCAL EN	AMPON

AMPCALD: 运放校准输出结果

1 = 运放校准输出为1

0 = 运放校准输出为0

INPS<1:0>: 运放正输入端选择位

00 = P2.2 为运放正输入端

01 = P2.3 为运放正输入端

10 = P1.5为运放正输入端

11 = 运放正输入端接地

AMPCALEN: AMP校准使能位

1 = 使能校准

0 = 禁止校准

AMPON: 运放使能位

1 = 运放使能

0 = 运放禁止

12.1.2 运放校准寄存器

寄存器: **AMPDT: 运放校准寄存器(20H)**

复位值	AMPDT7	AMPDT6	AMPDT5	AMPDT4	AMPDT3	AMPDT2	AMPDT1	AMPDT0
1000 0000	R/W							

AMPDT<7:0>: 校准数据位

AMPDT = 运放校准数据

12.2 运放使用方法

12.2.1 校准方法

运放失调校准步骤:

1. 设置运放的 IO 端口为模拟端口;
2. AMPON 置 1 打开运放;
3. 延迟 100us, 将 AMPCALEN 置 1 使能运放失调校准功能;
4. 延迟 100us, 读出运放输出结果 AMPCALD (AMPCTL<7>), 并写入 AMPDT<7>;
5. 将 AMPDT<6>改写为 1, 延迟 100us, 读出运放输出结果 AMPCALD;
6. 判断 AMPCALD 和 AMPDT7 是否相等, 相等则 AMPDT<6>保持为 1, 不相等则将 AMPDT<6>改写为 0;
7. 同样对 AMPDT<5:3>的每位做步骤 5 和 6 的操作;
8. AMPDT<2:0>从 000 开始加 1, 延迟 100us, 读出运放输出结果 AMPCALD;
9. 判断 AMPCALD 与 AMPDT7 是否相等, 相等返回步骤 8; 不相等, AMPCALEN 清 0 结束运放失调校准。如果 AMPDT<2:0>加到 111 时还是相等, 也结束运放失调校准, 开始正常运放工作。

12.2.2 使用方法

使用运放时软件的设置:

1. 将对应的引脚设置为模拟输入;
2. 选一路为正向输入端
3. 打开运放(AMPON =1)。

注: 1. 运放输出端也要设为模拟输入口

2. 休眠模式下运放被强制关闭不能使用

13 LED 驱动模块

13.1 概述

KF8F513x 内含一个 8×16 的 LED 驱动模块，最多可驱动 16 位 LED 数码管，共阴共阳可调，且带有辉度调节功能。

模块框图如下：

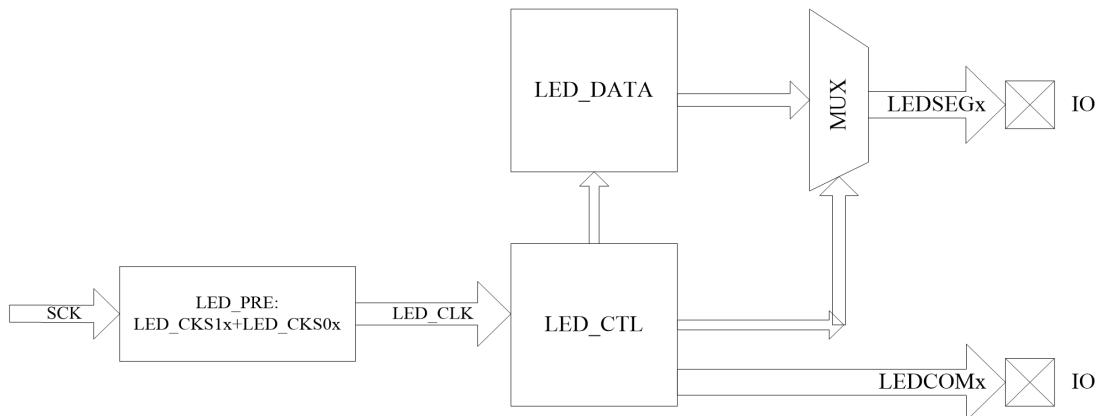


图 13.1 LED 驱动模块框图

当用户配置好 LED 驱动模块时，会在 LEDCOMx 引脚产生一个扫描位选的信号，当 LEDCOMx 为有效电平时，LEDSEGx 引脚将相对应的数据寄存器 LEDDATAx 的内容对应输出，得以产生一个段选信号。下图为 LED 驱动模块与数码管连接示意图。

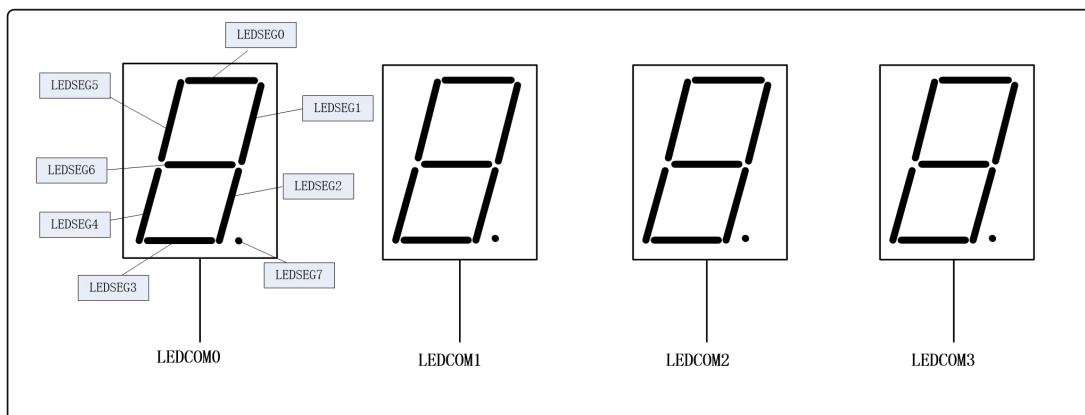


图 13.2 LED 驱动模块与数码管连接示意图

13.2 工作原理

13.2.1 扫描时钟产生

LED 显示电路扫描频率可调，时钟为系统时钟。LED 驱动模块提供两个可调分频，LEDPRE 寄存器的 LEDCKS1<3:0>和 LEDCKS0<3:0>，用户可通过 LEDPRE 寄存器对时钟进行分频得到所需的扫描频率。LED 扫描频率计算公式如下：

公式 13.1 LED 扫描频率计算公式

$$CLK_{LED} = \frac{SCLK}{32 \times 2^{LEDCKS1<3:0>} \times (LEDCKS0<3:0> + 1)} \text{。}$$

13.2.2 扫描信号波形

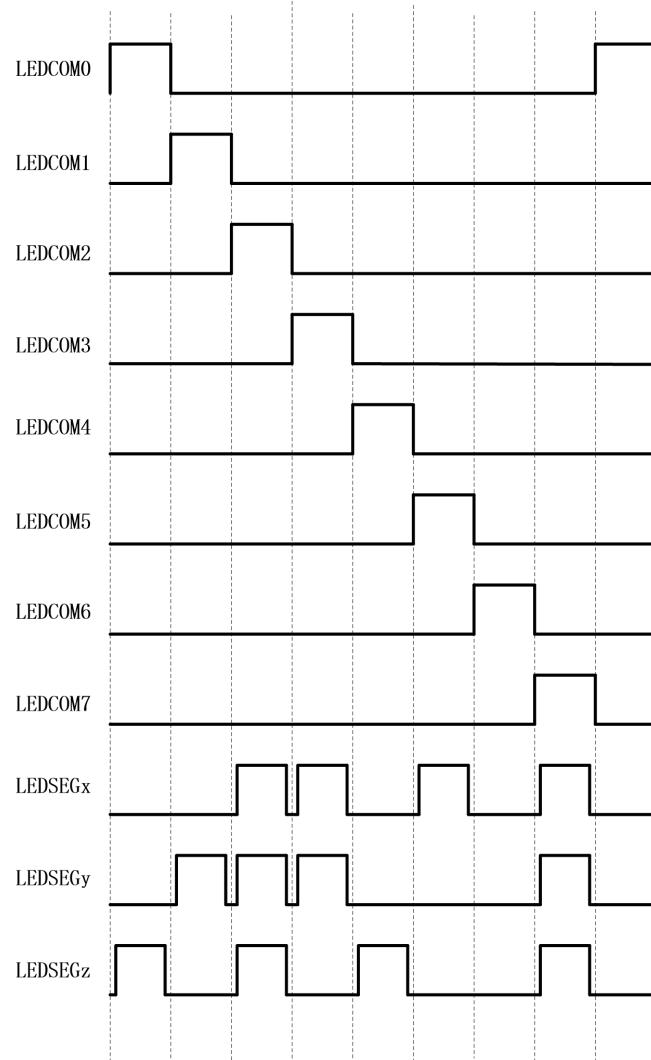


图 13.3 LED 扫描波形例图(使能 8 位)

13.3 LED 相关寄存器

表 13-1 LED 相关寄存器

地址	名称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
140H	LEDCTL	LEDEN	-	-	COMEN2	COMEN1	COMEN0	PNS1	PNS0
157H	LEDCTL2	-	-	-	-	-	LED SEGEN16	LED SEGEN12	LED SEGEN8
141H	LEDPRE	LEDCKS13	LEDCKS12	LEDCKS11	LEDCKS10	LEDCKS03	LEDCKS02	LEDCKS01	LEDCKS00
142H	LEDDATA0	DAT07	DAT06	DAT05	DAT04	DAT03	DAT02	DAT01	DAT00
143H	LEDDATA1	DAT17	DAT16	DAT15	DAT14	DAT13	DAT12	DAT11	DAT10
144H	LEDDATA2	DAT27	DAT26	DAT25	DAT24	DAT23	DAT22	DAT21	DAT20
145H	LEDDATA3	DAT37	DAT36	DAT35	DAT34	DAT33	DAT32	DAT31	DAT30
146H	LEDDATA4	DAT47	DAT46	DAT45	DAT44	DAT43	DAT42	DAT41	DAT40
147H	LEDDATA5	DAT57	DAT56	DAT55	DAT54	DAT53	DAT52	DAT51	DAT50
148H	LEDDATA6	DAT67	DAT66	DAT65	DAT64	DAT63	DAT62	DAT61	DAT60
149H	LEDDATA7	DAT77	DAT76	DAT75	DAT74	DAT73	DAT72	DAT71	DAT70
14AH	LEDDATA8	DAT87	DAT86	DAT85	DAT84	DAT83	DAT82	DAT81	DAT80
14BH	LEDDATA9	DAT97	DAT96	DAT95	DAT94	DAT93	DAT92	DAT91	DAT90
14CH	LEDDATAA	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0
14DH	LEDDATAB	DATB7	DATB6	DATB5	DATB4	DATB3	DATB2	DATB1	DATB0
14EH	LEDDATAC	DATC7	DATC6	DATC5	DATC4	DATC3	DATC2	DATC1	DATC0
14FH	LEDDATAD	DATD7	DATD6	DATD5	DATD4	DATD3	DATD2	DATD1	DATD0
155H	LEDDATAE	DATE7	DATE6	DATE5	DATE4	DATE3	DATE2	DATE1	DATE0
156H	LEDDATAF	DATF7	DATF6	DATF5	DATF4	DATF3	DATF2	DATF1	DATF0
26AH	APFCTLA	COM7 SEL	COM6 SEL	COM5 SEL	COM4 SEL	COM3 SEL	COM2 SEL	COM1 SEL	COM0 SEL
15CH	LEDLUM	LEDBR	LEDIOS	ODEN	ODSET	FRAMEF	LUM2	LUM1	LUM0

13.3.1 LED 显示驱动控制寄存器(LEDCTL)

寄存器: LEDCTL: LED显示驱动控制寄存器 (140H)

复位值 0000 0000	bit7								bit0	
	LEDEN	-	-	COMEN2	COMEN1	COMEN0	PNS1	PNS0	R/W	R/W
	R/W	U	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W

LEDEN: LED 显示驱动使能位

1 = 使能 LED 模块

0 = 禁止 LED 模块

COMEN<2:0>:LED 位使能数量选择位

000 = 使能 1 位, LEDDATA0、8 使能, LEDDATA1-7、9-F 禁止

001 = 使能 2 位, LEDDATA0-1、8-9 使能, LEDDATA2-7、A-F 禁止

010 = 使能 3 位, LEDDATA0-2、8-A 使能, LEDDATA3-7、B-F 禁止

011 = 使能 4 位, LEDDATA0-3、8-B 使能, LEDDATA4-7、C-F 禁止

100 = 使能 5 位, LEDDATA0-4、8-C 使能, LEDDATA5-7、D-F 禁止

101 = 使能 6 位, LEDDATA0-5、8-D 使能, LEDDATA6-7、E-F 禁止

110 = 使能 7 位, LEDDATA0-6、8-E 使能, LEDDATA7、F 禁止

111 = 使能 8 位, LEDDATA0-F 使能

PNS<1:0>: 共阴共阳选择位

00 = 工作在共阴模式, 直接驱动 LED

01 = 工作在共阴模式, 阴级公共端通过控制三级管驱动 LED

10 = 工作在开漏输出模式, 直接驱动 LED

11 = 工作在共阳模式, 直接驱动 LED

注:

1. 当 LED 模块选择在开漏输出模式时, 须配置 PNS<1:0>位为 10;
2. 当外围线路通过控制共阴级三极管驱动 LED 时, 须配置 PNS<1:0>位为 01。

13.3.2 LED 显示驱动控制寄存器 2(LEDCTL2)

寄存器: LEDCTL2: LED 显示驱动控制寄存器 2 (157H)

	bit7					bit0		
复位值	-	-	-	-	-	LED SEGEN16	LED SEGEN12	LED SEGEN8
0000 0000	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

LEDSEGEN16: LED 显示驱动 SEG12-15 使能位

1 = 使能 LEDSEG12-15

0 = 禁止 LEDSEG12-15

LEDSEGEN12: LED 显示驱动 SEG8-11 使能位

1 = 使能 LEDSEG8-11

0 = 禁止 LEDSEG8-11

LEDSEGEN8: LED 显示驱动 SEG0-7 使能位

1 = 使能 LEDSEG0-7

0 = 禁止 LEDSEG0-7

13.3.3 LED 显示驱动分频寄存器(LEDPRE)

寄存器: LEDPRE: LED 显示驱动分频寄存器 (141H)

	bit7								bit0
复位值	LEDCKS1	LEDCKS1	LEDCKS1	LEDCKS1	LEDCKS0	LEDCKS0	LEDCKS0	LEDCKS0	
0000 0000	3	2	1	0	3	2	1	0	

LEDCKS1<3:0>: LED 时钟源分频选择位

000x=1:2

0010=1:4

0011=1:8

0100=1:16

0101=1:32

0110=1:64

0111=1:128

1000=1:256

1001=1:512

1010=1:1024

1011=1:2048

1100=1:4096

1101=1:8192

1110=1:16384

1111=1:32768

LEDCKS0<3:0>:LED 预分频比选择位

1111 = 1:16

1110 = 1:15

1101 = 1:14

1100 = 1:13

1011 = 1:12

1010 = 1:11

1001 = 1:10

1000 = 1:9

0111 = 1:8

0110 = 1:7

0101 = 1:6

0100 = 1:5

0011 = 1:4

0010 = 1:3

0001 = 1:2

0000 = 1:1

13.3.4 LED 显示驱动数据寄存器(LEDDATAx)

寄存器: LEDDATAx: LED显示驱动数字寄存器

复位值 0000 0000	bit7								bit0
	DATx7	DATx6	DATx5	DATx4	DATx3	DATx2	DATx1	DATx0	
	R/W								

DATx<7:0>: LED 显示驱动数据位 x=0,1,2,3,4,5,6,7

1=LED熄灭

0=LED 点亮

LED 共有 16 个 LED 数据寄存器, 均位于 BANK1, 地址如下表所示:

表 13-2 LED 数据寄存器地址

LED 数据寄存器	地址	LED 数据寄存器	地址
LEDDATA0	142H	LEDDATA8	14AH
LEDDATA1	143H	LEDDATA9	14BH
LEDDATA2	144H	LEDDATAA	14CH
LEDDATA3	145H	LEDDATAB	14DH
LEDDATA4	146H	LEDDATAC	14EH
LEDDATA5	147H	LEDDATAD	14FH
LEDDATA6	148H	LEDDATAE	155H
LEDDATA7	149H	LEDDATAF	156H

13.3.5 LED 显示驱动辉度调节控制寄存器 (LEDLUM)

寄存器: LEDLUM: LED显示驱动辉度调节设置寄存器 (15CH)

复位值	LEDDBR	LEDIOS	ODEN	ODSET	FRAMEF	LUM2	LUM1	LUM0	bit0
0000 0000	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	

LEDDBR: LED 波形输出控制位

1 = LED 驱动只输出 LEDCOM 端波形, LEDSEG 端相应 IO 口为通用 IO 口
0 = LED 驱动波形正常输出

LEDIOS: 保留位

ODEN: 开漏输出使能位

1=使能开漏输出
0=禁止开漏输出

ODSET: 开漏输出方式选择位

1=LEDCOM 端 N 管开漏, LEDSEG 端 P 管开漏
0=LEDCOM 端 P 管开漏, LEDSEG 端 N 管开漏

FRAMEF: LED 显示标志位

1 = LED 完成 1 次周期扫描
0 = LED 未完成周期扫描

LUM<2:0>: LED 辉度调节设置位

000 = 15:16
001 = 14:16
010 = 12:16
011 = 10:16
100 = 8:16
101 = 6:16
110 = 4:16
111 = 2:16

13.4 LED 通道

LED 显示模块的 COM 端配备两组端口供用户选择，通过 APFCTLA 寄存器配置。COM 端对应引脚如下表所示。

表 13-2 LED-COM 端对应引脚索引表

APFCTLA	对应引脚	LEDCOM<7:0>	对应引脚
COM7SEL=0	COM0=P3.0	COM7SEL=1	COM0=P1.7
COM6SEL=0	COM1=P3.1	COM6SEL=1	COM1=P1.6
COM5SEL=0	COM2=P3.2	COM5SEL=1	COM2=P1.5
COM4SEL=0	COM3=P3.3	COM4SEL=1	COM3=P1.4
COM3SEL=0	COM4=P3.4	COM3SEL=1	COM4=P1.3
COM2SEL=0	COM5=P3.5	COM2SEL=1	COM5=P1.2
COM1SEL=0	COM6=P3.6	COM1SEL=1	COM6=P1.1
COM0SEL=0	COM7=P3.7	COM0SEL=1	COM7=P1.0

LED 显示模块的 SEG 端对应引脚固定，无复用功能，对应引脚如下表所示。SEG 端的使能通过 LEDCTL2 寄存器的 LEDSEGEN16、LEDSEGEN12 和 LEDSEGEN8 位控制。当 LEDSEGEN8=1 时，LEDSEG0-7 被使能；当 LEDSEGEN12=1 时，LEDSEG8-11 被使能；当 LEDSEGEN16=1 时，LEDSEG12-15 被使能。3 个 LEDSEGENx 位独立，

如果需要使用全部 16 个 LEDSEG 段，可将 LEDSEGEN16、LEDSEGEN12 和 LEDSEGEN8 位都置 1。

如果 LEDSEGEN16=1，LEDSEGEN12=0，LEDSEGEN8=1，则 LEDSEG12-15 和 LEDSEG0-7 被使能，LEDSEG8-11 被禁止。

表 13-2 LED-SEG 端对应引脚索引表

SEGx	对应引脚	SEGx	对应引脚
SEG0	P0.0	SEG8	P4.0
SEG1	P0.1	SEG9	P4.1
SEG2	P0.3	SEG10	P4.2
SEG3	P0.4	SEG11	P4.3
SEG4	P0.5	SEG12	P4.4
SEG5	P2.0	SEG13	P4.5
SEG6	P2.1	SEG14	P4.6
SEG7	P2.2	SEG15	P4.7

13.5 辉度调节

LED 驱动模块带有辉度调节功能。LEDLUM 寄存器的 LUM<2:0>位提供 8 档不同辉度供用户选择。

13.6 输出模式

LED 模块提供 2 种驱动信号输出方式：推挽输出和开漏输出。在选择开漏输出模式同时还可以设置 P 管开漏或者 N 管开漏。

LEDLUM 寄存器的 ODEN 位用于选择 LED 的输出模式。当 ODEN=0 时，LED 驱动波形推挽输出；当 ODEN=1 时，LED 驱动波形开漏输出。LEDLUM 寄存器的 ODSET 位用于设置开漏输出时 COM 端和 SEG 端的开漏方式；ODSET=1 时，LEDCOM 端 N 管开漏，LEDSEG 端 P 管开漏；ODSET=0 时，LEDCOM 端 P 管开漏，LEDSEG 端 N 管开漏。

开漏输出模式可以节省 LED 显示驱动工作时的功耗；COM 端和 SEG 端的开漏方式根据用户实际接线方式选择。

13.7 LED 显示驱动使用方法

1. 通过 LEDPRE 寄存器选择时钟分频得到所需的扫描频率；
2. 通过 APFCTLA 寄存器配置 LEDECOM 端引脚；
3. 设置 LEDCTL 寄存器来设置 LEDECOM 数量；
4. 向相应的 LEDDATAx 寄存器中写入显示数据；
5. 设置 PNS<1:0>位来选择共阴或共阳的输出方式；
6. 如果选择开漏输出模式，设置 ODSET 来选择 P 管开漏或者 N 管开漏；
7. 使能启动 LED 驱动模块。

14 硬件乘法器模块

14.1 概述

KF8F513X 包含一个单独的 16×16 的硬件乘法器模块，它是单片机的一个外设。CPU 通过相关的指令操作来对硬件乘法器的寄存器进行读写与运算操作。

乘法器的特性包含：

- ◆ 无符号 16 位二进制乘法；
- ◆ 乘法通过异步运算过程实现，运算时间=8 个指令周期

工作原理框图如下图所示：

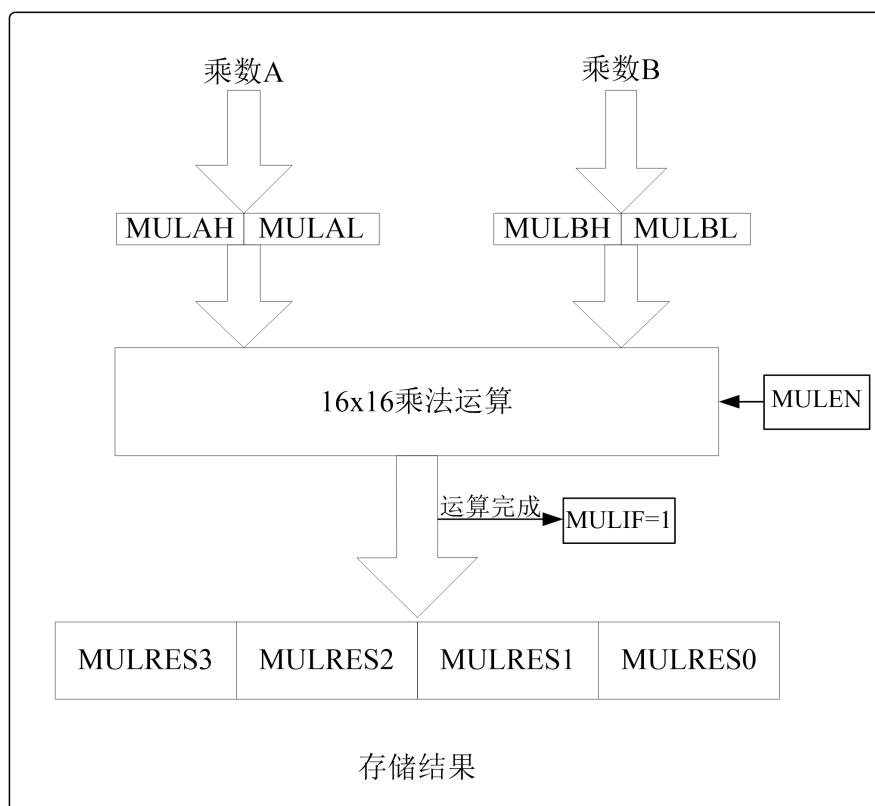


图 14.1 16×16 硬件乘法器原理框图

注：结果存入32位结果寄存器MULRES中，即运算结果为：[MULRES3:MULRES2:MULRES1:MULRES0]，MULRES3是最高8位，MULRES0是最低8位。

14.2 硬件乘法器相关寄存器

表 14-1 硬件乘法器相关寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
130H	MULAH	16 位乘法器乘数 A 高 8 位							
131H	MULAL	16 位乘法器乘数 A 低 8 位							
132H	MULBH	16 位乘法器乘数 B 高 8 位							
133H	MULBL	16 位乘法器乘数 B 低 8 位							
134H	MULCTL	-	-	-	-	-	-	MULEN	MULIF
135H	MULRES ₃	乘法运算结果寄存器 3							
136H	MULRES ₂	乘法运算结果寄存器 2							
137H	MULRES ₁	乘法运算结果寄存器 1							
138H	MULRES ₀	乘法运算结果寄存器 0							

14.2.1 乘法运算控制寄存器 (MULCTL)

寄存器: MULCTL:乘法运算控制寄存器 (地址:134H)

复位值 ----00	bit7							bit0	
	-	-	-	-	-	-	-	MULEN	MULIF
	U	U	U	U	U	U	U	R/W	R

MULEN: 运算使能位

1 = 使能乘法运算

0 = 禁止乘法运算

MULIF: 运算状态标志位

1 = 运算操作已经完成

0 = 没有完成运算

注: MULIF 不能直接由软件清零。需要清零 MULIF 时, 对乘数寄存器 (MULAH/L 或 MULBH/L) 进行操作即可清零。如: CLR MULAH。

14.3 硬件乘法器的使用

运算操作的使用设置如下:

- 1) 向乘数 A 寄存器和乘数 B 寄存器分别写入一个无符号的 16 位乘数
- 2) 乘法运算使能位 MULEN 置 1, 控制开始乘法运算

示例程序如下例所示：

例：

```
MOVB #0X01
MOV R0, #0XxxH
MOV MULAH , R0
MOV R0, #0XxxH
MOV MULAL , R0 ;装载乘数A
MOV R0, #0XxxH
MOV MULBH , R0
MOV R0, #0XxxH
MOV MULBL , R0 ;装载乘数B
SET MULCTL ,MULEN ;使能乘法运算
```

运算结束后，MUCTL 寄存器的 MULIF 置位，控制结果存入寄存器 MULRESx (x=0~3) 中，结果为:[MULRES3:MULRES2:MULRES1:MULRES0]。

15 硬件除法器模块

15.1 概述

KF8F513X 包含一个硬件除法器，它是单片机的一个外设。CPU 通过相关的指令操作来对硬件除法器的寄存器进行读写与运算操作。

硬件除法器的特性包含：

- ◆ 无符号除法
- ◆ $16 \div 16$ 位
- ◆ 运算时间=9 个系统时钟周期

具体的原理框图如下图所示：

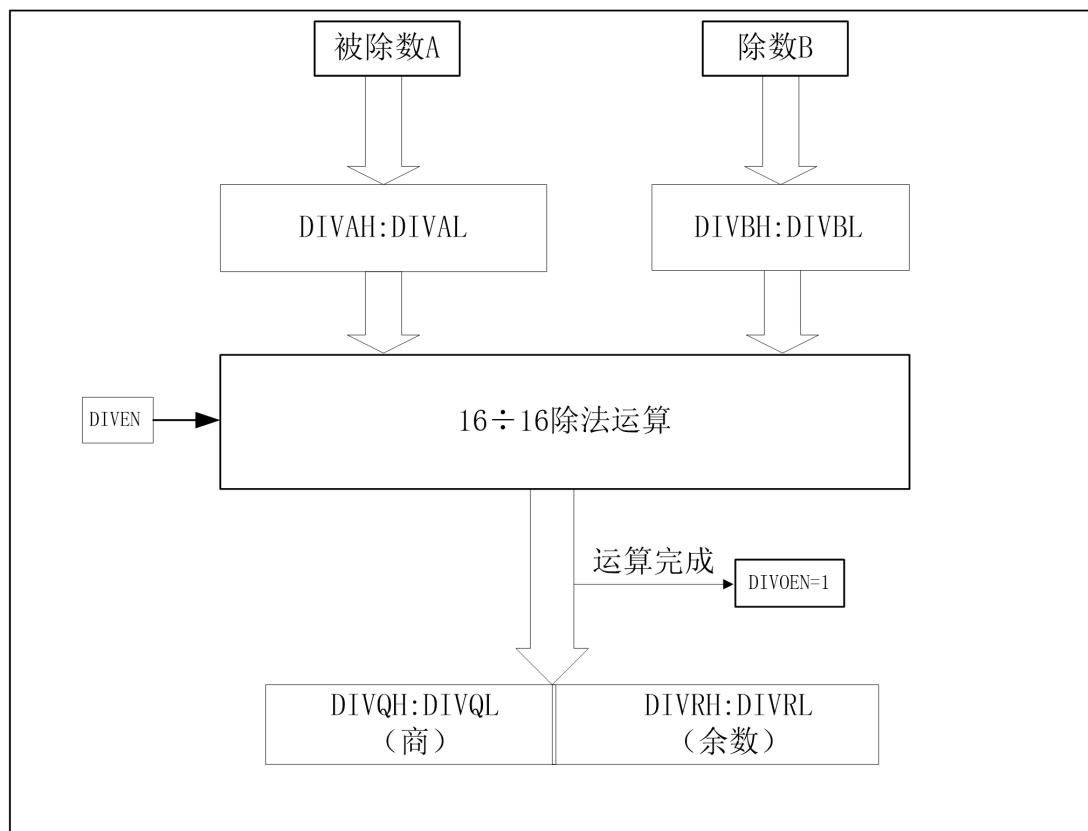


图 15.1 硬件除法器原理框图

15.2 硬件除法器相关寄存器

表 15-1 硬件除法器相关寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
139H	DIVCTL	-	-	-	-	-	-	DIVOEN	DIVEN
13AH	DIVAH	被除数高 8 位寄存器							
13BH	DIVAL	被除数低 8 位寄存器							
115H	DIVBH	除数高 8 位寄存器							
13CH	DIVBL	除数低 8 位寄存器							
13DH	DIVQH	商高 8 位寄存器							
13EH	DIVQL	商低 8 位寄存器							
13FH	DIVRL	余数低 8 位寄存器							
108H	DIVRH	余数高 8 位寄存器							

15.2.1 除法控制寄存器 (DIVCTL)

在除法的过程中，需要对除法模块的使能进行控制，同时需要有一个可以通知 CPU 或者其他模块，除法已经完成的信号。

除法控制寄存器 DIVCTL 包含：

- ◆ 除法运算使能位 DIVEN
- ◆ 除法运算完成标志位 DIVOEN

寄存器： DIVCTL: 除法控制寄存器(地址:139H)

复位值	bit7							bit0	
	-	-	-	-	-	-	DIVOEN	DIVEN	
-----00	U	U	U	U	U	U	R	W	

DIVOEN: 除法运算完成标志位

1 = 除法完成，商和余数可读

0 = 除法未完成或者未开始，商和余数不可读

DIVEN: 除法运算使能位

1 = 使能除法运算

0 = 禁止除法运算

15.2.2 数据寄存器

对于除法，需要有操作数进行操作，在开始除法的时候，需要被除数和除数，当除法结束后，需要商和余数。

这四种分别使用{DIVAH:DIVAL}, {DIVBH:DIVBL}, {DIVQH:DIVQL}, {DIVRH:DIVRL}进行表示。

15.2.2.1被除数寄存器{DIVAH:DIVAL}

寄存器: DIVAH: 被除数高8位寄存器(地址:13AH)

复位值	bit7	DIVAH7	DIVAH6	DIVAH5	DIVAH4	DIVAH3	DIVAH2	DIVAH1	DIVAH0	bit0
00000000		R/W								

寄存器: DIVAL: 被除数低8位寄存器(地址:13BH)

复位值	bit7	DIVAL7	DIVAL6	DIVAL5	DIVAL4	DIVAL3	DIVAL2	DIVAL1	DIVAL0	bit0
00000000		R/W								

DIVAH: 被除数高 8 位寄存器, 用于存放被除数高 8 位。

DIVAL: 被除数低 8 位寄存器, 用于存放被除数低 8 位。

15.2.2.2除数寄存器 DIVB

寄存器: DIVBH: 除数高8位寄存器(地址:115H)

复位值	bit7	DIVBH7	DIVBH6	DIVBH5	DIVBH4	DIVBH3	DIVBH2	DIVBH1	DIVBH0	bit0
00000000		R/W								

寄存器: DIVBL: 除数低8位寄存器(地址:13CH)

复位值	bit7	DIVBL7	DIVBL6	DIVBL5	DIVBL4	DIVBL3	DIVBL2	DIVBL1	DIVBL0	bit0
00000000		R/W								

DIVBH: 用于存放运算过程中的除数高 8 位

DIVBL: 用于存放运算过程中的除数低 8 位

15.2.2.3结果商寄存器{DIVQH:DIVQL}

寄存器: DIVQH: 结果商高8位寄存器(地址:13DH)

复位值	bit7	DIVQH7	DIVQH6	DIVQH5	DIVQH4	DIVQH3	DIVQH2	DIVQH1	DIVQH0	bit0
00000000		R	R	R	R	R	R	R	R	

寄存器: DIVQL: 结果商低8位寄存器(地址:13EH)

复位值	bit7	DIVQL7	DIVQL6	DIVQL5	DIVQL4	DIVQL3	DIVQL2	DIVQL1	DIVQL0	bit0
00000000		R	R	R	R	R	R	R	R	

DIVQH: 存放运算结果商的高 8 位; (只读)

DIVQL: 存放运算结果商的低 8 位; (只读)

15.2.2.4 运算余数寄存器 DIVR

寄存器: DIVRH: 余数高8位寄存器(地址:108H)

复位值	DIVRH7	DIVRH6	DIVRH5	DIVRH4	DIVRH3	DIVRH2	DIVRH1	DIVRH0	bit7	bit0
00000000	R	R	R	R	R	R	R	R		

寄存器: DIVRL: 余数低8位寄存器(地址:13FH)

复位值	DIVRL7	DIVRL6	DIVRL5	DIVRL4	DIVRL3	DIVRL2	DIVRL1	DIVRL0	bit7	bit0
00000000	R	R	R	R	R	R	R	R		

DIVRH: 除法完成后存储余数高 8 位 (只读)

DIVRL: 除法完成后存储余数低 8 位 (只读)

15.3 硬件除法器操作

除法通过使能信号 DIVEN 进行使能, 当 DIVEN=1 的时候, 除法器开始工作, 直到除法完成。

当除法完成之后, 除法器模块会自动清除 DIVEN 信号, 并将完成标志位 DIVOEN 设为 1。

注意, 在除法器使能之前需要将数据输入到 DIVA 和 DIVB 寄存器中, 否则的话, 除法过程就会出现数据错误。

除法运算器的设置:

- (1) 分别向被除数寄存器{DIVAH:DIVAL}写入被除数
- (2) 分别向除数寄存器{DIVBH:DIVBL}写入除数
- (3) 设置 DIVCTL 的 DIVEN (<DIVCTL.0>) 使能除法运算器

示例程序如下例所示:

```

MOV R0, #0xXX
MOV DIVAH, R0
MOV R0, #0xYY
MOV DIVAL, R0
MOV R0, #0xMM
MOV DIVBH, R0
MOV R0, #0xNN
MOV DIVBL, R0
MOV R0, #0x01
MOV DIVCTL, R0

```

这样就可以开始计算 $xxyy \div mmnn$, 当 DIVOEN=1 的时候, 可以从{DIVQH:DIVQL}和{DIVRH:DIVRL}中分别读出商和余数。

16 复位

KF8F513X 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，上电复位时它们的状态不定，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”。下图给出了片内复位电路的简化结构方框图。

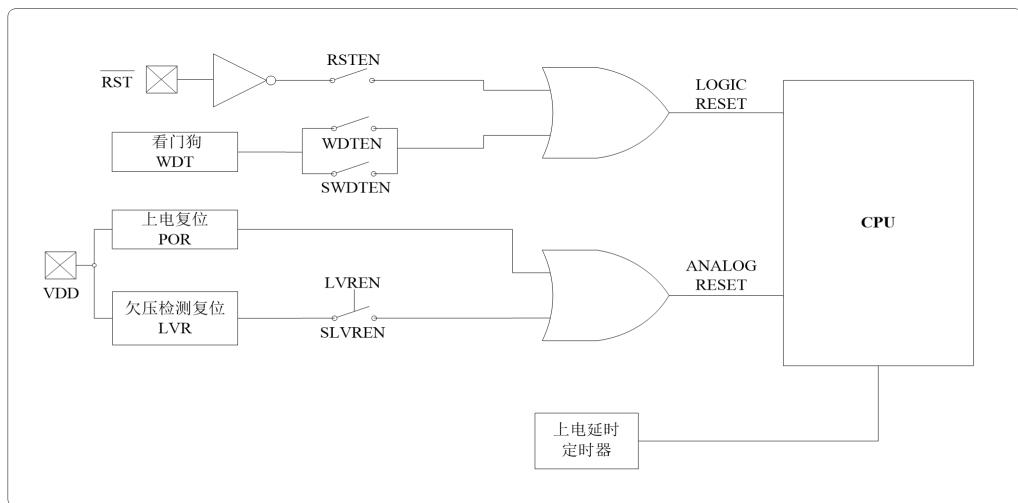


图 16.1 片内复位电路简化框图

注:上电延时定时器只对上电复位(POR)和欠压复位(LVR)有效。

16.1 电源控制状态寄存器(PCTL)

如寄存器 PCTL 所示, \overline{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 $\overline{LVR}=0$, 则表示发生过欠压检测复位。 \overline{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{LVR} 状态位是不可预知的。 \overline{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: **PCTL:** 电源控制寄存器(地址:2EH)

复位值 --01 00xx	bit7			bit0				
	-	-	-	SLVREN	IPEN	SWDTEN	\overline{POR}	\overline{LVR}
	U	U	R/W	R/W	R/W	R/W	R/W	R/W

SLVREN: 软件欠压检测使能位

1 = 软件使能欠压检测

0 = 软件禁止欠压检测

IPEN: 中断优先级控制位

1 = 使能中断优先级功能

0 = 禁止中断优先级, 即为普通模式

SWDTEN: 软件看门狗定时器使能位

当配置字的 WDTEN=0 时

1 = 软件使能看门狗定时器

0 = 软件禁止看门狗定时器

\overline{POR} : 上电复位状态位

1 = 未发生上电复位

0 = 发生了上电复位

\overline{LVR} : 欠压复位状态位

1 = 未发生欠压复位

0 = 已发生欠压复位

注: SLVREN 位的配置方法详见欠压检测复位 (LVR) 章节。

16.2 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。KF8F513X 的上电复位时间为 70ms 左右。

16.3 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。

16.4 RST 复位

使能外部 RST 复位（配置位 RSTEN=1）后，当引脚 P0.3/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 RST 复位引脚，即可打开 RST 复位。

在 RST 复位时，KF8F513X 器件有一个噪声滤波器用于检测和滤除小脉冲，下图是建议 RST 复位电路。

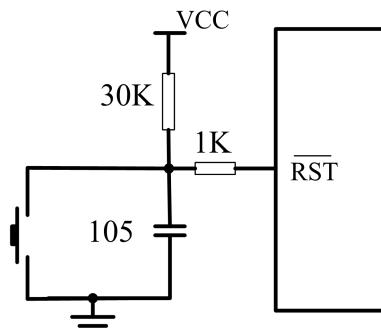


图 16.2 建议 RST 复位电路

16.5 欠压检测复位(LVR)

KF8F513X 系列中的单片机片内配备一个欠压检测复位电路 (LVR)。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, 单片机会按照配置位中的 LVREN 位的配置, 选择 LVR 是否使能。LVR 还提供一个软件位 PCTL_SLVREN, 当 FUSE_LVREN=0 时, PCTL_SLVREN 位失效; 当 FUSE_LVREN=1 时, PCTL_SLVREN 位有效, 可以通过软件清零 SLVREN 位关闭 LVR, 同样可以通过置位该位以重新使能 LVR。上电时, 如果 FUSE_LVREN=1, LVR 将默认工作。

表 16-1 欠压检测复位 LVR 的配置

配置位 LVREN	PCTL_SLVREN	LVR 是否被使能
0	0/1	禁止
1	清零	禁止
1	置 1	使能

如果 VDD 跌落至 VLVR(VLVR=2.1V)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 VLVR 以上, 再经过 2ms 内置延时后单片机开始正常工作; 如果使能上电延时定时器, 则在随后 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 VLVR 以下的时间小于规定参数(TLVR), 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 VLVR 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 VLVR 以上时, 上电延时定时器启动一个 32ms 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。

16.6 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 32ms(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 32ms。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。

16.7 不同复位条件下对寄存器的影响

表 16-2 寄存器在各种复位发生后的状态

地址	名称	上电复位	RST 复位 欠压检测	中断唤醒 WDT 超时唤醒
01H	T0	xxxx xxxx	uuuu uuuu	uuuu uuuu
02H	PCL	0000 0000	0000 0000	PC+1
03H	PSW	---1 1xxx	--0q quuu	--uq quuu
05H	P0	xxxx xxxx	xxxx xxxx	uuuu uuuu
06H	P2	xxxx xxxx	xxxx xxxx	uuuu uuuu
07H	P1	xxxx xxxx	xxxx xxxx	uuuu uuuu
08H	P3	xxxx xxxx	xxxx xxxx	uuuu uuuu
0AH	PCH	---0 0000	---0 0000	---u uuuu
0BH	INTCTL	0000 0000	0000 0000	uuuu uuuu
0CH	EIF1	0000 0000	0000 0000	uuuu uuuu
0DH	EIF2	0-00 -000	0-00 -000	u-uu -uuu
0EH	T1L	xxxx xxxx	xxxx xxxx	uuuu uuuu
0FH	T1H	xxxx xxxx	xxxx xxxx	uuuu uuuu
10H	T1CTL	0000 0000	0000 0000	uuuu uuuu
11H	T2L	0000 0000	0000 0000	uuuu uuuu
12H	T2CTL0	-000 0000	-000 0000	-uuu uuuu
13H	CCP3CTL	0000 0000	0000 0000	uuuu uuuu
14H	CCP3H	0000 0000	0000 0000	uuuu uuuu
15H	CCP3L	0000 0000	0000 0000	uuuu uuuu
17H	BANK	---- 0000	---- 0000	---- uuuu
18H	ADSCANCTL	0000 0000	0000 0000	uuuu uuuu
1BH	AMPCTL	1000 0000	1000 0000	uuuu uuuu
1DH	ANS1	0000 0000	0000 0000	uuuu uuuu
1EH	ADCDATA0H	xxxx xxxx	xxxx xxxx	uuuu uuuu
1FH	ADCCTL0	00-- --00	00-- --00	uu-- --uu
20H	AMPDT	1000 0000	1000 0000	uuuu uuuu
21H	OPTR	1111 1111	1111 1111	uuuu uuuu
22H	IP0	---- -000	---- -000	---- -uuu
23H	IP1	0000 0000	0000 0000	uuuu uuuu
24H	IP2	0-00 -000	0-00 -000	u-uu -uuu
25H	TR0	1111 1111	1111 1111	uuuu uuuu
26H	TR2	1111 1111	1111 1111	uuuu uuuu
27H	TR1	1111 1111	1111 1111	uuuu uuuu
28H	OSCSTA	-110 --00	-110 --00	-uuu --uu
29H	IP3	0000 0000	0000 0000	uuuu uuuu
2BH	VREFCTL	0000 0000	0000 0000	uuuu uuuu
2CH	EIE1	0000 0000	0000 0000	uuuu uuuu
2DH	EIE2	0000 0000	0000 0000	uuuu uuuu
2EH	PCTL	--01 00xx	--01 00xx	--00 00uu
2FH	OSCCTL	0010 0000	0010 0000	uuuu uuuu
30H	OSCCAL2	1000 0000	1000 0000	uuuu uuuu
31H	ANS0	0000 0000	0000 0000	uuuu uuuu
32H	ANS2	0000 0000	0000 0000	uuuu uuuu
33H	ANS3	0000 0000	0000 0000	uuuu uuuu
34H	OSC0CAL2	0000 0101	0000 0101	uuuu uuuu
35H	PUR0	1111 1111	1111 1111	uuuu uuuu
36H	IOCL0	0000 0000	0000 0000	uuuu uuuu
37H	OSCCAL1	---- -000	---- -000	---- -uuu
38H	NVMDATAH	0000 0000	0000 0000	uuuu uuuu
39H	NVMDATAL	0000 0000	0000 0000	uuuu uuuu
3AH	NVMADDRH	0000 0000	0000 0000	uuuu uuuu
3BH	NVMADDRL	0000 0000	0000 0000	uuuu uuuu
3CH	NVMCTL0	---- ----	---- ----	---- ----
3DH	NVMCTL1	---- ----	---- ----	---- ----
3EH	ADCDATA0L	xxxx xxxx	xxxx xxxx	uuuu uuuu

地址	名称	上电复位	RST 复位 欠压检测	中断唤醒 WDT 超时唤醒
3FH	ADCCTL1	0000 00-0	0000 00-0	uuuu uuu-u
40H	T2CCR0H	0000 0000	0000 0000	uuuu uuuuu
41H	T2H	xxxx xxxx	xxxx xxxx	uuuu uuuuu
42H	PP5H	xxxx xxxx	xxxx xxxx	uuuu uuuuu
43H	PWM5H0	xxxx xxxx	xxxx xxxx	uuuu uuuuu
44H	PWM5H1	xxxx xxxx	xxxx xxxx	uuuu uuuuu
45H	P0LR	xxxx xxxx	xxxx xxxx	uuuu uuuuu
46H	P2LR	xxxx xxxx	xxxx xxxx	uuuu uuuuu
47H	P1LR	xxxx xxxx	xxxx xxxx	uuuu uuuuu
48H	P3LR	xxxx xxxx	xxxx xxxx	uuuu uuuuu
49H	TR3	1111 1111	1111 1111	uuuu uuuuu
4AH	EIE3	0000 0000	0000 0000	uuuu uuuuu
4BH	EIF3	0000 0000	0000 0000	uuuu uuuuu
4CH	OSCCAL3	1000 1000	1000 1000	uuuu uuuuu
4DH	OSCCAL0	0001 0000	0001 0000	uuuu uuuuu
4EH	T3CTL	0000 0000	0000 0000	uuuu uuuuu
4FH	T3L	0000 0000	0000 0000	uuuu uuuuu
50H	ADCDATA1H	0000 0000	0000 0000	uuuu uuuuu
51H	ADCDATA1L	0000 0000	0000 0000	uuuu uuuuu
52H	PP5L	1111 1111	1111 1111	uuuu uuuuu
54H	T2CCR0L	0000 0000	0000 0000	uuuu uuuuu
55H	PWM5L0	xxxx xxxx	xxxx xxxx	uuuu uuuuu
56H	PWM5L1	xxxx xxxx	xxxx xxxx	uuuu uuuuu
57H	PWM5CTL0	0000 0000	0000 0000	uuuu uuuuu
58H	ADCINTCTL	0--- --00	0--- --00	u--- --uu
59H	ADCDATA2H	0000 0000	0000 0000	uuuu uuuuu
5AH	ADCDATA2L	0000 0000	0000 0000	uuuu uuuuu
5BH	PWM5CTL1	0000 0000	0000 0000	uuuu uuuuu
5CH	P5ASCTL0	0000 0000	0000 0000	uuuu uuuuu
5DH	PSTRCTL0	---0 0001	---0 0001	---u uuuuu
5EH	ADCDATA3H	0000 0000	0000 0000	uuuu uuuuu
5FH	T3H	0000 0000	0000 0000	uuuu uuuuu
60H	PUR1	0000 0000	0000 0000	uuuu uuuuu
61H	PUR2	0000 0000	0000 0000	uuuu uuuuu
62H	TMRBUZ	---- --00	---- --00	---- --uu
63H	ADCCTL2	---- 0000	---- 0000	---- uuuuu
67H	INTEDGCTL	00-- ---1	00-- ---1	uu-- ---u
6AH	ADCDATA3L	0000 0000	0000 0000	uuuu uuuuu
6BH	IOCL4	0000 0000	0000 0000	uuuu uuuuu
6CH	PUR3	0000 0000	0000 0000	uuuu uuuuu
105H	CCP4L	0000 0000	0000 0000	uuuu uuuuu
106H	CCP4H	0000 0000	0000 0000	uuuu uuuuu
108H	DIVRH	0000 0000	0000 0000	uuuu uuuuu
109H	CCP4CTL	---0 0000	---0 0000	---u uuuuu
10CH	PWM5L2	0000 0000	0000 0000	uuuu uuuuu
10DH	PWM5L3	0000 0000	0000 0000	uuuu uuuuu
10EH	PWM5OC	--00 0000	--00 0000	--uu uuuuu
10FH	T2CTL1	0000 0--0	0000 0--0	uuuu u--u
115H	DIVBH	0000 0000	0000 0000	uuuu uuuuu
116H	T2CCR1L	0000 0000	0000 0000	uuuu uuuuu
118H	T2CCR1H	0000 0000	0000 0000	uuuu uuuuu
119H	PWM5FC	--00 0000	--00 0000	--uu uuuuu
11AH	P5ASCTL1	0000 0000	0000 0000	uuuu uuuuu
11BH	PSTRCTL1	---- 1010	---- 1010	---- uuuuu
11CH	PWM5CTL2	--0- -000	--0- -000	--u- -uuuu
11DH	PWM5PC	--00 0000	--00 0000	--uu uuuuu
11EH	PWM5H2	0000 0000	0000 0000	uuuu uuuuu
11FH	PWM5H3	0000 0000	0000 0000	uuuu uuuuu
120H	RSCTL1	0000 000x	0000 000x	uuuu uuuuu
121H	TXSDR1	0000 0000	0000 0000	uuuu uuuuu

地址	名称	上电复位	RST 复位 欠压检测	中断唤醒 WDT 超时唤醒
122H	RXSDR1	0000 0000	0000 0000	uuuu uuuu
123H	BRCTL1	0100 0000	0100 0000	uuuu uuuu
124H	TSCTL1	0000 0010	0000 0010	uuuu uuuu
125H	EUBRGL1	0000 0000	0000 0000	uuuu uuuu
126H	EUBRGH1	0000 0000	0000 0000	uuuu uuuu
127H	USLPEN1	0----	0----	u----
128H	SSCICL0	0000 0000	0000 0000	uuuu uuuu
12AH	SSCICL1	0000 0000	0000 0000	uuuu uuuu
12BH	SSCISTA	0000 0000	0000 0000	uuuu uuuu
12CH	SSCIBUFR	0000 0000	0000 0000	uuuu uuuu
12DH	UARTMTH1	0000	0000	uuuu
12EH	SSCIADD	1111 1111	1111 1111	uuuu uuuu
	SSCIMSK	1111 1111	1111 1111	uuuu uuuu
12FH	WDTPS	---- 0100	---- 0100	---- uuuu
130H	MULAH	0000 0000	0000 0000	uuuu uuuu
131H	MULAL	0000 0000	0000 0000	uuuu uuuu
132H	MULBH	0000 0000	0000 0000	uuuu uuuu
133H	MULBL	0000 0000	0000 0000	uuuu uuuu
134H	MULCTL	---- --00	---- --00	---- --uu
135H	MULRES3	0000 0000	0000 0000	uuuu uuuu
136H	MULRES2	0000 0000	0000 0000	uuuu uuuu
137H	MULRES1	0000 0000	0000 0000	uuuu uuuu
138H	MULRES0	0000 0000	0000 0000	uuuu uuuu
139H	DIVCTL	0000 0000	0000 0000	uuuu uuuu
13AH	DIVAH	0000 0000	0000 0000	uuuu uuuu
13BH	DIVAL	0000 0000	0000 0000	uuuu uuuu
13CH	DIVBL	0000 0000	0000 0000	uuuu uuuu
13DH	DIVQH	0000 0000	0000 0000	uuuu uuuu
13EH	DIVQL	0000 0000	0000 0000	uuuu uuuu
13FH	DIVRL	0000 0000	0000 0000	uuuu uuuu
140H	LEDCTL	0--0 0000	0--0 0000	u--u uuuu
141H	LEDPRE	0000 0000	0000 0000	uuuu uuuu
142H	LEDDATA0	0000 0000	0000 0000	uuuu uuuu
143H	LEDDATA1	0000 0000	0000 0000	uuuu uuuu
144H	LEDDATA2	0000 0000	0000 0000	uuuu uuuu
145H	LEDDATA3	0000 0000	0000 0000	uuuu uuuu
146H	LEDDATA4	0000 0000	0000 0000	uuuu uuuu
147H	LEDDATA5	0000 0000	0000 0000	uuuu uuuu
148H	LEDDATA6	0000 0000	0000 0000	uuuu uuuu
149H	LEDDATA7	0000 0000	0000 0000	uuuu uuuu
14AH	LEDDATA8	0000 0000	0000 0000	uuuu uuuu
14BH	LEDDATA9	0000 0000	0000 0000	uuuu uuuu
14CH	LEDDATAA	0000 0000	0000 0000	uuuu uuuu
14DH	LEDDATAB	0000 0000	0000 0000	uuuu uuuu
14EH	LEDDATAC	0000 0000	0000 0000	uuuu uuuu
14FH	LEDDATAD	0000 0000	0000 0000	uuuu uuuu
151H	LPRCCTL	0000 0100	0000 0100	uuuu uuuu
152H	LPRCCAL	0000 0000	0000 0000	uuuu uuuu
153H	XTALCAL	0101 0000	0101 0000	uuuu uuuu
154H	PCAL	0000 0000	0000 0000	uuuu uuuu
155H	LEDDATAE	0000 0000	0000 0000	uuuu uuuu
156H	LEDDATAF	0000 0000	0000 0000	uuuu uuuu
157H	LEDCTL2	0000 0000	0000 0000	uuuu uuuu
159H	VREFCAL0	0000 0000	0000 0000	uuuu uuuu
15AH	VREFCAL1	1010 0000	1010 0000	uuuu uuuu
15CH	LEDLUM	0--- 0 000	0--- 0 000	u--- u uuu
15DH	T3REL	0000 0000	0000 0000	uuuu uuuu
15EH	T3REH	0000 0000	0000 0000	uuuu uuuu
160H	T4L	0000 0000	0000 0000	uuuu uuuu
161H	T4H	0000 0000	0000 0000	uuuu uuuu
162H	T4REL	0000 0000	0000 0000	uuuu uuuu

地址	名称	上电复位	RST 复位 欠压检测	中断唤醒 WDT 超时唤醒
163H	T4REH	0000 0000	0000 0000	uuuu uuuu
164H	T4CTL	0000 0000	0000 0000	uuuu uuuu
165H	RC32KCAL	0000 0000	0000 0000	uuuu uuuu
206H	ANS4	1111 1111	1111 1111	uuuu uuuu
207H	ANS5	1111 1111	1111 1111	uuuu uuuu
20CH	P4	XXXX XXXX	XXXX XXXX	uuuu uuuu
20DH	P4LR	XXXX XXXX	XXXX XXXX	uuuu uuuu
20EH	TR4	1111 1111	1111 1111	uuuu uuuu
20FH	PUR4	1111 1111	1111 1111	uuuu uuuu
210H	P5	-XXX XXXX	-XXX XXXX	-uuu uuuu
211H	P5LR	-XXX XXXX	-XXX XXXX	-uuu uuuu
212H	TR5	-111 1111	-111 1111	-uuu uuuu
213H	PUR5	1111 1111	1111 1111	uuuu uuuu
21DH	PWMCTL0	0000 0000	0000 0000	uuuu uuuu
21EH	PWMCTL1	0000 0000	0000 0000	uuuu uuuu
21FH	PP1	1111 1111	1111 1111	uuuu uuuu
220H	PP2	1111 1111	1111 1111	uuuu uuuu
221H	PWM10L	XXXX XXXX	XXXX XXXX	uuuu uuuu
222H	PWM20H	XXXX XXXX	XXXX XXXX	uuuu uuuu
223H	PWM11L	XXXX XXXX	XXXX XXXX	uuuu uuuu
224H	PWM21H	XXXX XXXX	XXXX XXXX	uuuu uuuu
225H	PWM12L	XXXX XXXX	XXXX XXXX	uuuu uuuu
226H	PWM22H	XXXX XXXX	XXXX XXXX	uuuu uuuu
227H	PWM13L	XXXX XXXX	XXXX XXXX	uuuu uuuu
228H	PWM23H	XXXX XXXX	XXXX XXXX	uuuu uuuu
229H	PWM3CTL	0000 0000	0000 0000	uuuu uuuu
22AH	PP3H	1111 1111	1111 1111	uuuu uuuu
22BH	PP3L	1111 1111	1111 1111	uuuu uuuu
22CH	PWM30H	XXXX XXXX	XXXX XXXX	uuuu uuuu
22DH	PWM31H	XXXX XXXX	XXXX XXXX	uuuu uuuu
22EH	PWM32H	XXXX XXXX	XXXX XXXX	uuuu uuuu
22FH	PWM33H	XXXX XXXX	XXXX XXXX	uuuu uuuu
230H	PWM30L	XXXX XXXX	XXXX XXXX	uuuu uuuu
231H	PWM31L	XXXX XXXX	XXXX XXXX	uuuu uuuu
232H	PWM32L	XXXX XXXX	XXXX XXXX	uuuu uuuu
233H	PWM33L	XXXX XXXX	XXXX XXXX	uuuu uuuu
236H	TEMPSNR	---- --00	---- --00	---- --uu
23FH	UPINSET2	0000 0000	0000 0000	uuuu uuuu
240H	U7816CTL2	0-00 0000	0-00 0000	u-uu uuuu
241H	U7816RXCTL2	0000 0000	0000 0000	uuuu uuuu
242H	U7816TXCTL2	0001 0000	0001 0000	uuuu uuuu
243H	CLKDIV2	0000 0000	0000 0000	uuuu uuuu
244H	EGTCTL2	0000 0000	0000 0000	uuuu uuuu
245H	TZBRGH2	0000 0000	0000 0000	uuuu uuuu
246H	TZBRGL2	0000 0000	0000 0000	uuuu uuuu
247H	UARTMTH2	0000 0000	0000 0000	uuuu uuuu
248H	USLPEN2	0---- ----	0---- ----	u---- ----
249H	BRCTL2	0100 0000	0100 0000	uuuu uuuu
24AH	RSCTL2	0000 0000	0000 0000	uuuu uuuu
24BH	RXSDR2	XXXX XXXX	XXXX XXXX	uuuu uuuu
24CH	EUBRGH2	0000 0000	0000 0000	uuuu uuuu
24DH	EUBRGL2	0000 0000	0000 0000	uuuu uuuu
24EH	TXSDR2	XXXX XXXX	XXXX XXXX	uuuu uuuu
24FH	TSCTL2	0000 0000	0000 0000	uuuu uuuu
250H	BKPCTL	0000 0000	0000 0000	uuuu uuuu
251H	EIE4	---- 0000	---- 0000	---- uuuu
252H	EIF4	---- 0000	---- 0000	---- uuuu
253H	IP4	---- 0000	---- 0000	---- uuuu
254H	CTCHCOMB0	0000 0000	0000 0000	uuuu uuuu
255H	CTCHCOMB1	0000 0000	0000 0000	uuuu uuuu
256H	CTCHCOMB2	0000 0000	0000 0000	uuuu uuuu

地址	名称	上电复位	RST 复位 欠压检测	中断唤醒 WDT 超时唤醒
260H	APFCTL0	0000 0000	0000 0000	uuuu uuuu
261H	APFCTL1	0000 0000	0000 0000	uuuu uuuu
262H	APFCTL2	0000 0000	0000 0000	uuuu uuuu
263H	APFCTL3	0000 0000	0000 0000	uuuu uuuu
264H	APFCTL4	0000 0000	0000 0000	uuuu uuuu
265H	APFCTL5	0000 0000	0000 0000	uuuu uuuu
266H	APFCTL6	0000 0000	0000 0000	uuuu uuuu
267H	APFCTL7	0000 0000	0000 0000	uuuu uuuu
268H	APFCTL8	0000 0000	0000 0000	uuuu uuuu
269H	APFCTL9	0000 0000	0000 0000	uuuu uuuu
26AH	APFCTLA	0000 0000	0000 0000	uuuu uuuu
26BH	APFCTLB	0000 0000	0000 0000	uuuu uuuu
305H	CTCTL2	0000 0000	0000 0000	uuuu uuuu
306H	CTCTL3	0001 0000	0001 0000	uuuu uuuu
307H	CTCTL4	0100 0000	0100 0000	uuuu uuuu
308H	CTCH1VALVEH	0000 0000	0000 0000	uuuu uuuu
309H	CTCH1VALVEL	0000 0000	0000 0000	uuuu uuuu
30AH	CTCH2VALVEH	0000 0000	0000 0000	uuuu uuuu
30BH	CTCH2VALVEL	0000 0000	0000 0000	uuuu uuuu
30CH	CTCH3VALVEH	0000 0000	0000 0000	uuuu uuuu
30DH	CTCH3VALVEL	0000 0000	0000 0000	uuuu uuuu
30EH	CTCH4VALVEH	0000 0000	0000 0000	uuuu uuuu
30FH	CTCH4VALVEL	0000 0000	0000 0000	uuuu uuuu
310H	CTCTL0	0000 0000	0000 0000	uuuu uuuu
311H	CTCTL1	0000 0000	0000 0000	uuuu uuuu
312H	SSCTL	---- 0000	---- 0000	---- uuuu
313H	CTSCAN	0000 0000	0000 0000	uuuu uuuu
314H	CTCHSEL	0000 0000	0000 0000	uuuu uuuu

表 16-3 不同复位条件下对标志位的影响

POR	LVR	TO	PD	复位方式
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 RST 复位
u	u	1	0	休眠模式中的 RST 复位

图注: u=未发生变化

17 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，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. 外设中断
5. I/O 口电平变化中断

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 $\overline{\text{TO}}$ 和 $\overline{\text{PD}}$ 位来确定单片机唤醒的原因。上电时 $\overline{\text{PD}}$ 位将被置 1，而当器件从休眠模式唤醒时，该位将被清 0。 $\overline{\text{TO}}$ 位则在 WDT 唤醒发生时被清 0。

在使用中断方式唤醒时，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 指令之后加一条 NOP 指令即可。

18 看门狗定时器

为了防止单片机在正常工作时程序跑飞，KF8F513X 提供一个看门狗定时器。看门狗定时器使用内部低频振荡器作为工作时钟源，因此它无需外接任何器件。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。在休眠模式看门狗仍能正常运行，超时将唤醒单片机并使其继续执行 IDLE 后面的指令。

18.1 看门狗相关寄存器

表 18-1 看门狗相关的寄存器

地址	名称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
2EH	PCTL	-	-	-	SLVREN	IPEN	SWDTEN	<u>POR</u>	<u>LVR</u>
12FH	WDTPS	-	-	-	-	WDTPS3	WDTPS2	WDTPS1	WDTPS0

18.1.1 看门狗预分频选择寄存器 WDTPS

寄存器： WDTPS： WDT预分频选择寄存器(地址:12FH)

	bit7									bit0
复位值 ---- 0100	-	-	-	-	-	WDTPS3	WDTPS2	WDTPS1	WDTPS0	

WDTPS<3:0>:看门狗定时器预分频比选择位

- 0000 = 1: 32
- 0001 = 1: 64
- 0010 = 1: 128
- 0011 = 1: 256
- 0100 = 1: 512 (默认) 16ms
- 0101 = 1: 1024
- 0110 = 1: 2048
- 0111 = 1: 4096
- 1000 = 1: 8192
- 1001 = 1: 16384
- 1010 = 1: 32768
- 1011 = 1: 65536
- 其他 = 保留

18.2 看门狗的开启方式

看门狗的开启/关闭方式：

- 通过配置位 WDTEN，打开/关闭看门狗；
- 通过寄存器 PCTL 的 SWDTEN 位，打开/关闭看门狗。

上述两种方式任意一种都可启动看门狗；配置位 WDTEN 一旦使能，看门狗将一直开启，软件配置位 SWDTEN 无效；配置位 WDTEN 未使能时，软件配置位 SWDTEN 允许用户在软件上根据实际需求打开/关闭看门狗。

18.3 看门狗的清狗方式

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进行清狗操作。执行 CWDT 指令进行清狗操作或者执行 IDLE 指令进入休眠模式后，将清零整个看门狗定时器（包括看门狗预分频器和后分频器）。当看门狗定时器出现超时时，状态字寄存器 PSW 中的 \overline{TO} 位将被清 0。

看门狗定时器使用内部低频振荡器作为工作时钟源，因此它无需外接任何器件，在休眠模式仍能正常运行。WDT 超时事件对单片机的动作：

- 在正常运行时，WDT 超时事件将使单片机产生一次复位；
- 在休眠模式下，WDT 超时事件将唤醒单片机并使其继续执行 IDLE 后面的指令。

18.4 看门狗的周期

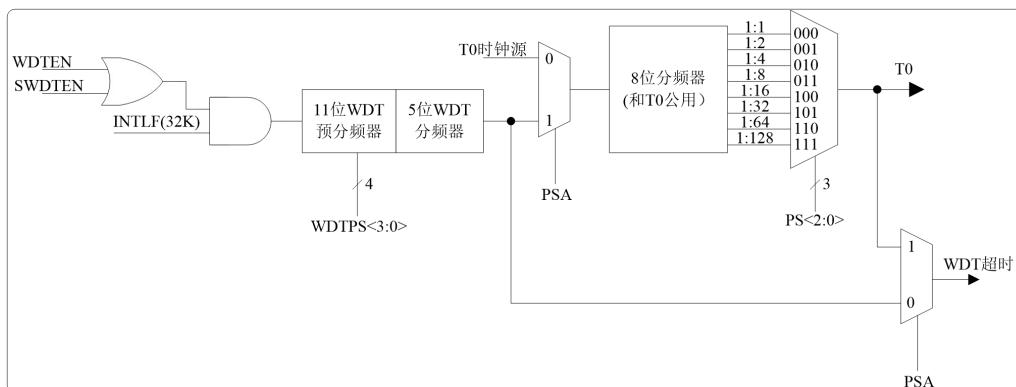


图 18.1 看门狗定时器框图

式 18.1 看门狗周期计算公式

$$T_{WDT} = \frac{2^{WDTPS<3:0>+5} \times 2^{PS<2:0>}}{F_{INTLF}}$$

注：FINTLF 为内部低频振荡器频率，约 31.25KHZ。

如上框图所示：看门狗定时器的时钟源为内部低频时钟（带校正功能的 31.25KHZ），相

相关内容详见内部低频振荡器章节。

看门狗定时器的周期由两个分频器的配置决定，最短约 1ms，最长约 268s，默认配置时为 16ms。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。

看门狗定时器带有两个分频器：

- 一个 16 位（11 位可编程）预分频器；
- 一个 8 位可编程后分频器（与 T0 共用）。

16 位预分频器中有 11 位可编程，由 WDTCTL 寄存器的 WDTPS<3:0>位选择预分频比（1:32 到 1:65536），共 12 档。

后分频器为看门狗定时器和定时/计数器 T0 共用，通过 OPTR 寄存器的 PSA 位将后分频器分配给 WDT 或者 T0；，OPTR 寄存器的 PS<2:0>位选择后分频器的分频比（1/1 到 1/128）。

注：任何对OPTR寄存器的PSA位或者PS<2:0>位操作前，须对看门狗定时器进行清狗操作（执行CWDT指令）。否则，可能引起芯片的异常复位。

19 电气规范

19.1 极限参数值

表 19-1 芯片极限参数值

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°C~125°C
2	储存温度	-40°C~150°C
3	VDD 相对于VSS 的电压	5.5V
4	VPP 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	80mA
7	VDD 引脚的最大输入电流	80mA
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。

19.2 INTHFOSC 的频率精度与 VDD 和温度之间的关系

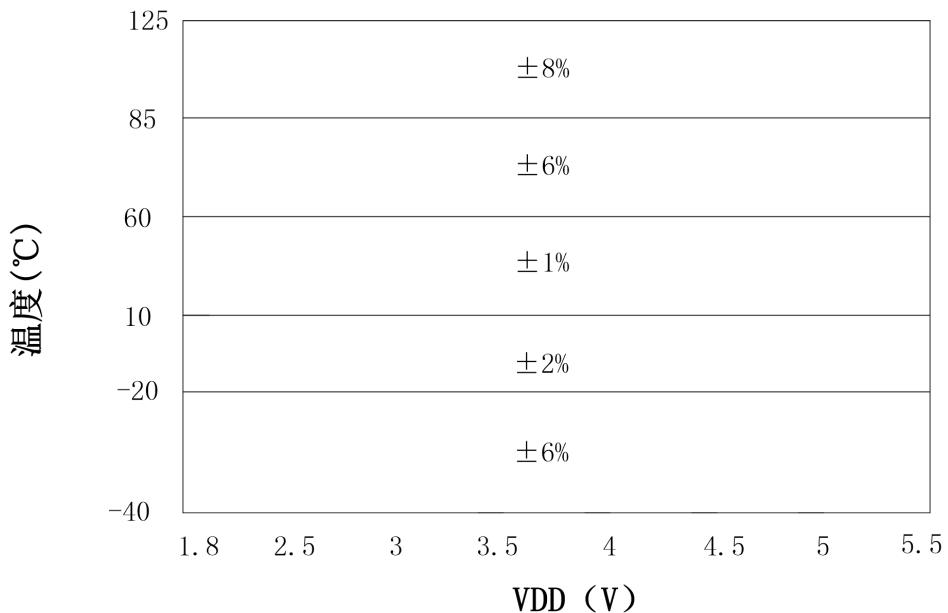


图 19.1 内部高频振荡器频率精度与VDD和温度关系图

19.3 静态电流特性

表 19-2 芯片静态电流 (IDD) 特性

序号	测试条件		最小值	典型值	最大值	单位
	振荡频率	VDD(V)				
1	16MHz	5.0	-	1100	-	μA
		3.3	-	1100	-	
2	8MHz	5.0	-	680	-	μA
		3.3	-	680	-	
3	4MHz	5.0	-	460	-	μA
		3.3	-	460	-	
4	2MHz	5.0	-	350	-	μA
		3.3	-	350	-	
5	1MHz	5.0	-	290	-	μA
		3.3	-	290	-	
6	500kHz	5.0	-	260	-	μA
		3.3	-	260	-	
7	250kHz	5.0	-	250	-	μA
		3.3	-	250	-	
8	125kHz	5.0	-	240	-	μA
		3.3	-	240	-	

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。

19.4 休眠电流特性

表 19-3 芯片休眠电流 (IDD) 特性

测试条件:25°C							
序号	休眠模式	测试条件		最小值	典型值	最大值	uA
		外设	VDD(V)				
1	深度休眠模式	WDT、BOR、比较器等外设被禁止	5.0 3.3	- -	2.5 1.8	5 5	

注:

1:该型号只提供一种休眠模式:深度休眠模式。

19.5 外设电流特性

表 19-4 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	欠电压复位电流 (ILVR)	WDT、RTC 等外设被禁止	5.0	-	9.5	14.0	uA
			3.3	-	6.0	8.5	
			2.5	-	4.0	6.0	
2	WDT	在超低功耗模式下	5.0	-	0.9	1.3	uA
			3.3	-	0.8	1.1	
			2.5	-	0.7	1.0	
6	运放	使能运放, 无负载	5.0	-	48	88	uA
			3.3	-	45	83	
			2.5	-	43	80	
7	ADC	使能 ADC, 未转换	5.0	-	895	1350	uA
			3.3	-	765	1155	
			2.5	-	515	760	
			3.3	-	70	78	
			2.5	-	68	78	

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。

19.6 I/O 端口电平和芯片供电电压特性

表 19-5 芯片 IO 端口电平特性

测试条件 (特别声明除外):						
工作温度 -40°C~85°C -40°C~125°C(*-S)						
符号	参数说明	测试条件	最小值	典型值	最大值	单位
VIL	输入低电平 I/O引脚 采用TTL缓冲器		V _{SS}	-	0.15VDD	V
	采用施密特缓冲触发器		V _{SS}	-	0.2VDD	V
VIH	输入高电平 I/O端口 采用TTL缓冲器		VDD-0.6	-	V _{DD}	V
	采用施密特缓冲触发器		0.8VDD	-	V _{DD}	V
VOL	输出低电压		-	-	0.6	V
VOH	输出高电压		VDD-0.6	-	-	V
IIL	输入漏电流	V _{SS} <Vi<VDD	-1	-	1	μA

表 19-6 芯片供电电压特性

测试条件 (特别声明除外):						
工作温度 -40°C~85°C -40°C~125°C(*-S)						
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		1.8	-	5.5	V

19.7 POR

表 19-7 芯片 POR 特性

测试条件 (特别声明除外):						
工作温度 -40°C~85°C						
-40°C~125°C(*-S)						
符号	参数说明	测试条件	最小值	典型值	最大值	单位
VPOR	上电复位电压	校准	1.6	1.7	1.8	V
Tempo	复位延时		-	2.5	-	ms
Idd	静态电流	VDD=3.3V	-	147	-	nA

19.8 LVR

表 19-8 芯片 LVR 特性

测试条件 (特别声明除外):						
工作温度 -40°C~85°C						
-40°C~125°C(*-S)						
符号	参数说明	测试条件	最小值	典型值	最大值	单位
VLVR	掉电复位电压		1.8	-	2.1	V
Hysteresis	迟滞		-	40	-	mV
Idd	静态电流	VDD=3.3V	-	5	-	uA

19.9 内部高频振荡器

表 19-9 芯片 INTHF 特性

测试条件 (特别声明除外): 工作温度 -40°C~85°C/-40°C~125°C(*-S)						
符号	参数说明	测试条件	最小值	典型值	最大值	单位
f _{CLKOUT}	HOSC frequency	temp=25°C	15.84	16	16.16	MHz
TRIM	User trimming step	temp=25°C	0.1	0.2	0.35	%
DuCy	Duty Cycle	—	45	—	55	%
t _{su}	Oscillator start-up time	—	—	5	—	μs
t _{stab}	Oscillator stabilization time	—	—	17	—	μs
I _{POWER_ISS}	Oscillator power consumption	temp=25°C	—	133	—	μA

19.10 内部低频振荡器

表 19-10 INTLF 特性

测试条件 (特别声明除外): 工作温度 -40°C~85°C/-40°C~125°C(*-S)						
符号	参数说明	测试条件	最小值	典型值	最大值	单位
Freq*	振荡频率	校准后	—	32	—	KHz
Iq*	静态电流	—	—	330	—	nA

*该数据为设计值

19.11 外部低频振荡器

表 19-11 EXTLF 特性

测试条件 (特别声明除外): 工作温度 -40°C~85°C/-40°C~125°C(*-S)						
符号	参数说明	测试条件	最小值	典型值	最大值	单位
Freq	振荡频率	—	—	32.768	—	KHz
Iq*	静态电流	—	—	340	—	nA

*该数据为设计值

19.12 外部高频振荡器

表 19-12 EXTLF 特性

测试条件 (特别声明除外): 工作温度 -40°C~85°C/-40°C~125°C(*-S)						
符号	参数说明	测试条件	最小值	典型值	最大值	单位
Freq	振荡频率	—	—	20	—	MHz
Iq*	静态电流	—	—	4	—	mA

*该数据为设计值

19.13 ADC12

表 19-13 A/D 转换器 (ADC12) 特性

测试条件 (特别声明除外): 工作温度 -40°C~85°C/-40°C~125°C(*-S)						
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		-	-	12	位
E _{INL} *	积分误差		-	±2	-	LSB
E _{DNL} *	微分误差		-	±1	-	LSB
E _{OFF} *	失调误差 ⁽¹⁾		-	±2	-	LSB
E _{GN} *	增益误差		-	±2	-	LSB
V _{A1N} *	满量程范围		V _{SS}	-	V _{REF}	V
T _{CNV} *	AD转换时间		-	11	-	TAD
T _{sOP} A _{le} *	AD转换速率		-	-	400	KSPS
I _q *	AD工作电流			550		uA

注: (1)*表示该数据为设计值。

(2) 失调误差固定为 5mV 正失调, 针对不同的参考电压对应为不同值, 使用时可以直接减去绝对值。

19.14 运算放大器

表 19-14 运算放大器模块直流特性

测试条件 (特别声明除外): 工作温度 -40°C~85°C/-40°C~125°C(*-S)							
类型	符号	特性	最小值	典型值	最大值	单位	备注
直流	V _{os} *	输入失调电压	-	±3	-	mV	校准后
	ICMR*	输入共模电压范围	GND	-	VDD		
	V _{OT} *	输出电压摆幅	GND	-	VDD		
	I _q *	静态电流	-	47		uA	
	I _{source} *	源电流	-	5		mA	
	I _{sink} *	灌电流	-	8.3		mA	
交流	GBW*	单位增益带宽	-	1		Mhz	Cload=100pF
	A _{OL} *	差模开环直流电压增益	-	115		dB	
	CMRR*	共模抑制比	-	120		dB	
	PSRR*	电源抑制比	-	109	-	dB	
瞬态	S _R *	摆率	-1.53		+1.24	V/us	Cload=100pF
	Setting time*		-	4	-	us	Cload=100pF

*该数据为设计值

19.15 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内工作才可以确保正常运行。

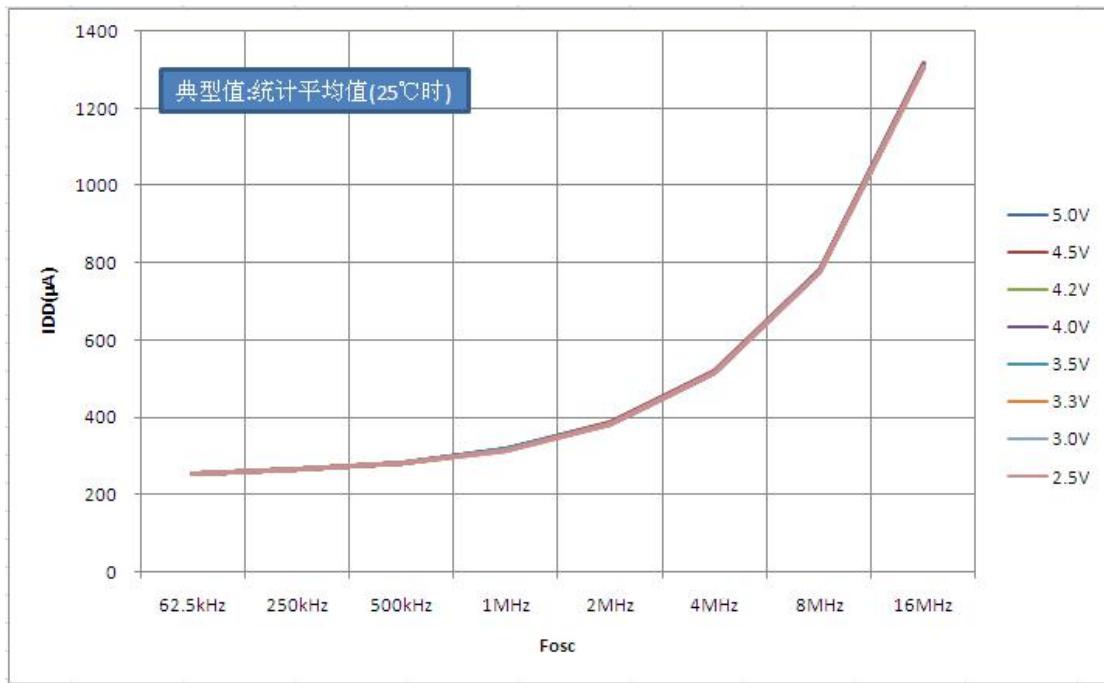


图 19.2 不同 VDD 时典型 IDD – Fosc 关系曲线图(HFINTOSC 模式)

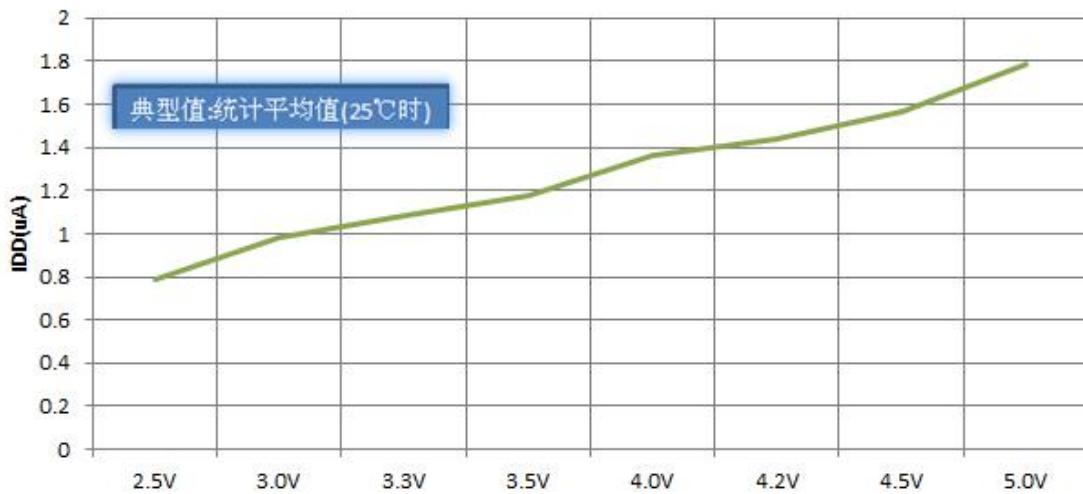


图 19.3 不同 VDD 时典型 IDSLP – VDD 关系曲线图(深度休眠模式, 禁止所有外设)

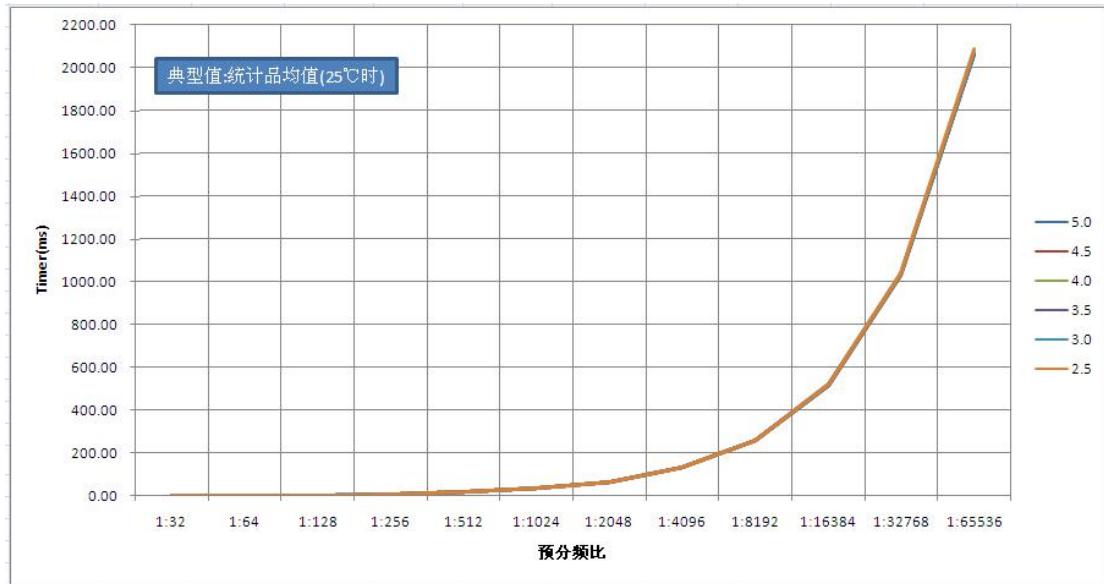


图 19.4 不同VDD时看门狗预分频比WDTPS<3:0> 一周期关系曲线图

注:1/32 分频时看门狗周期为 1ms。

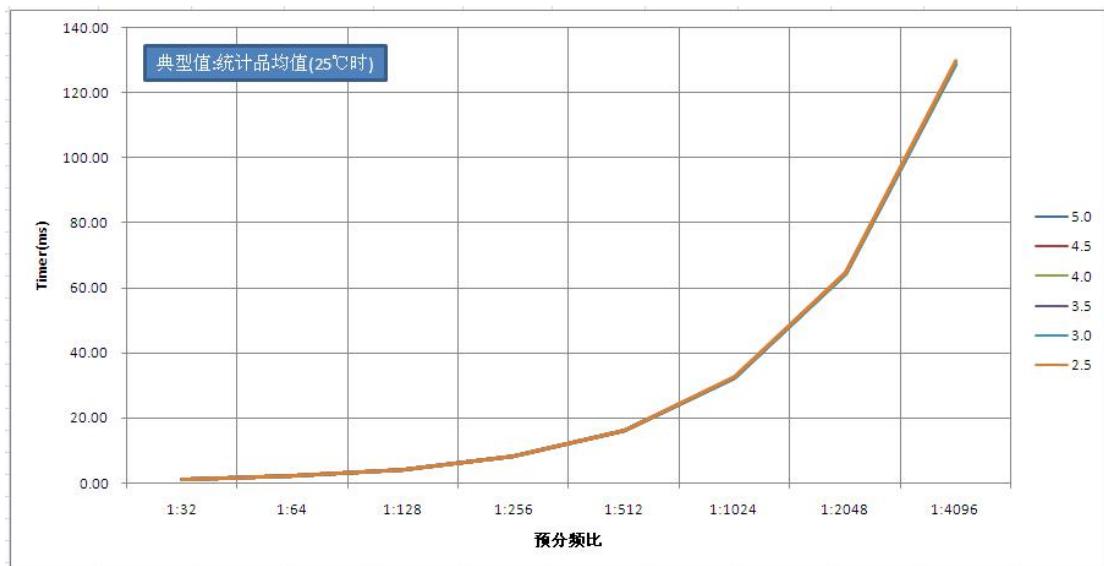


图 19.5 不同 VDD 时看门狗预分频比 PS<2:0> 一周期关系曲线图

备注:1/32 分频时看门狗周期为 1ms。

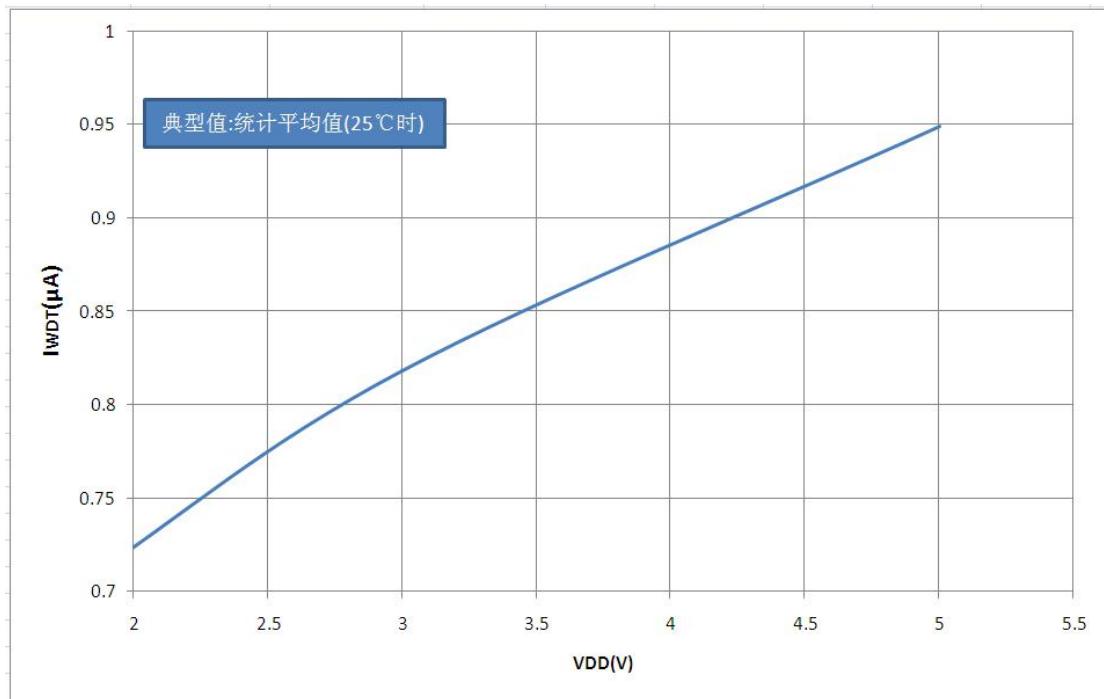


图 19.6 看门狗电流 I_{WDT} — V_{DD} 关系曲线图

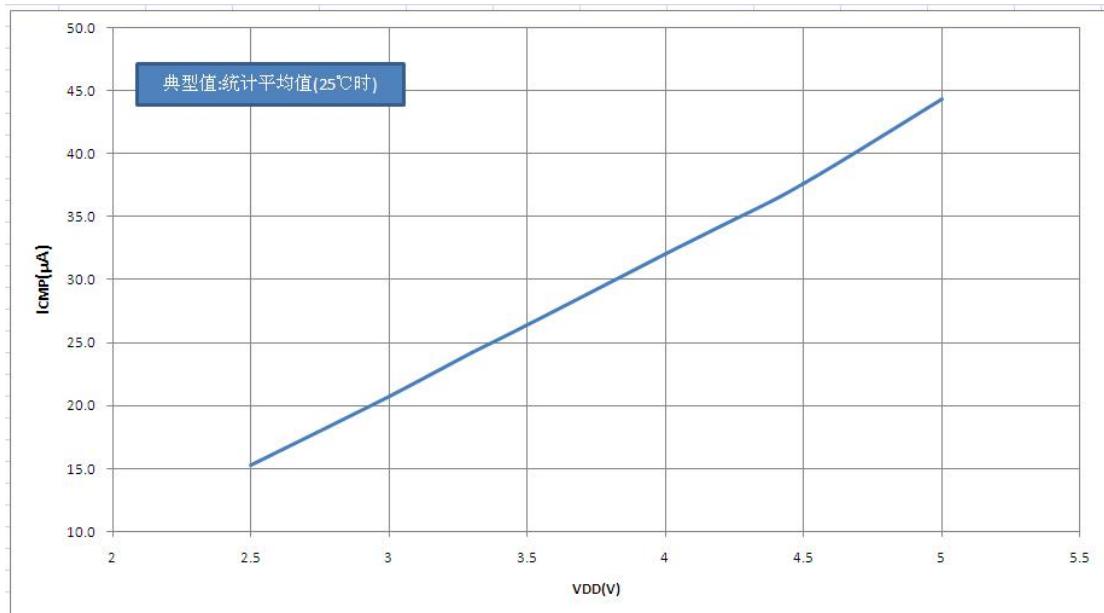


图 19.7 比较器电流 I_{CMP} — V_{DD} 关系曲线图 (使能一路比较器)

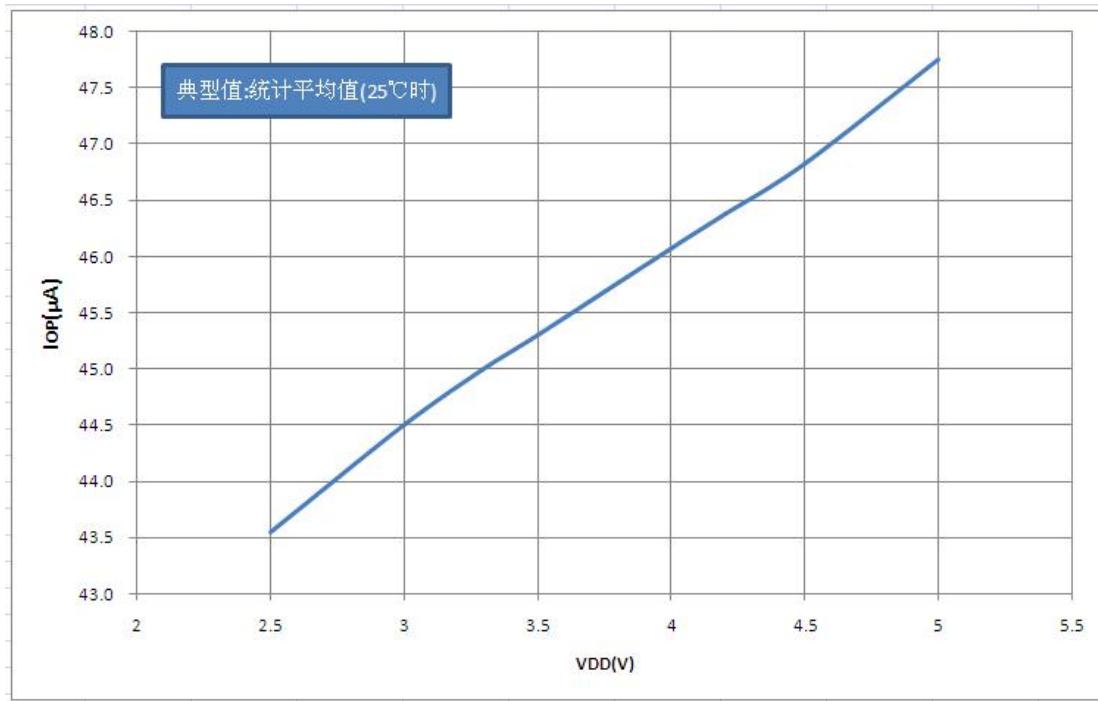


图 19.8 运算放大器电流 I_{OP} — V_{DD} 关系曲线图 (使能一路比较器)

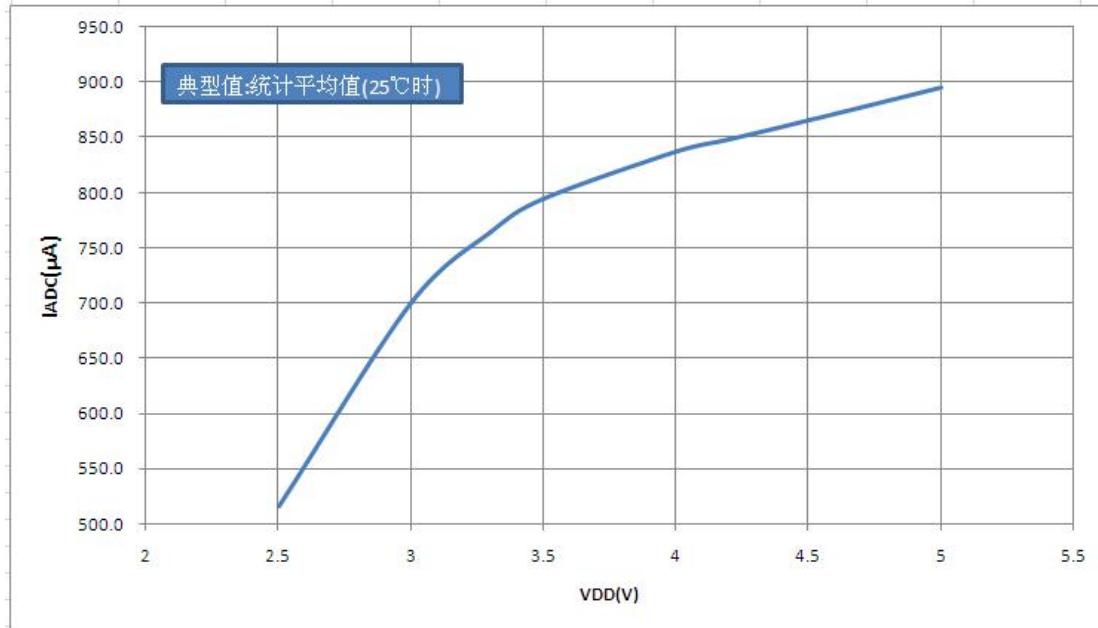


图 19.9 欠压复位电流 I_{ADC} — V_{DD} 关系曲线图

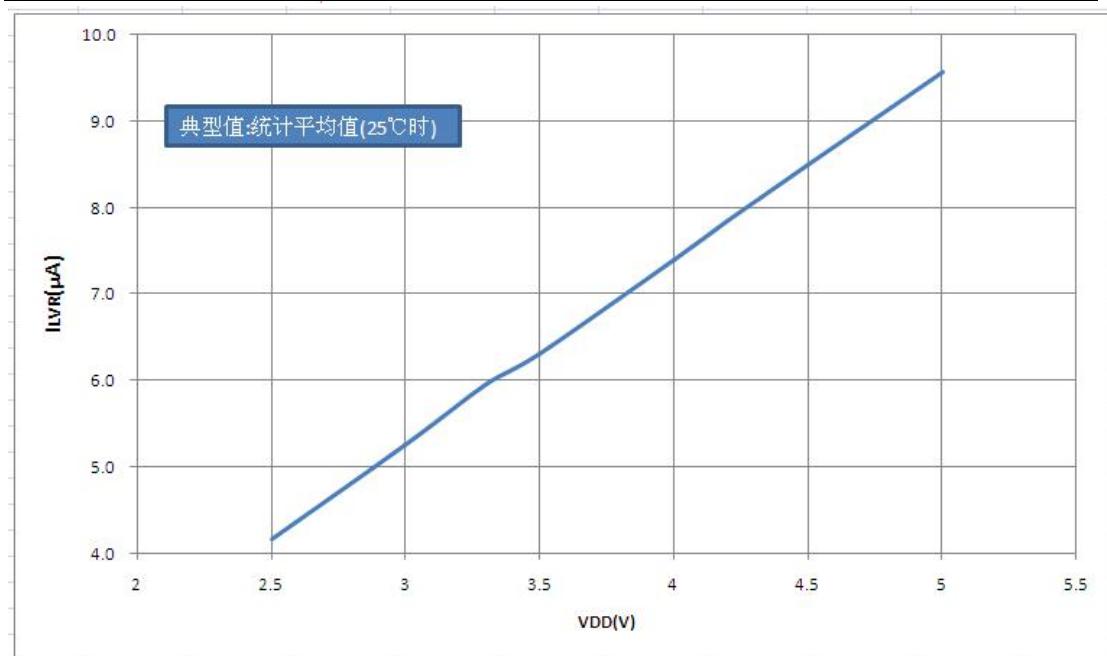
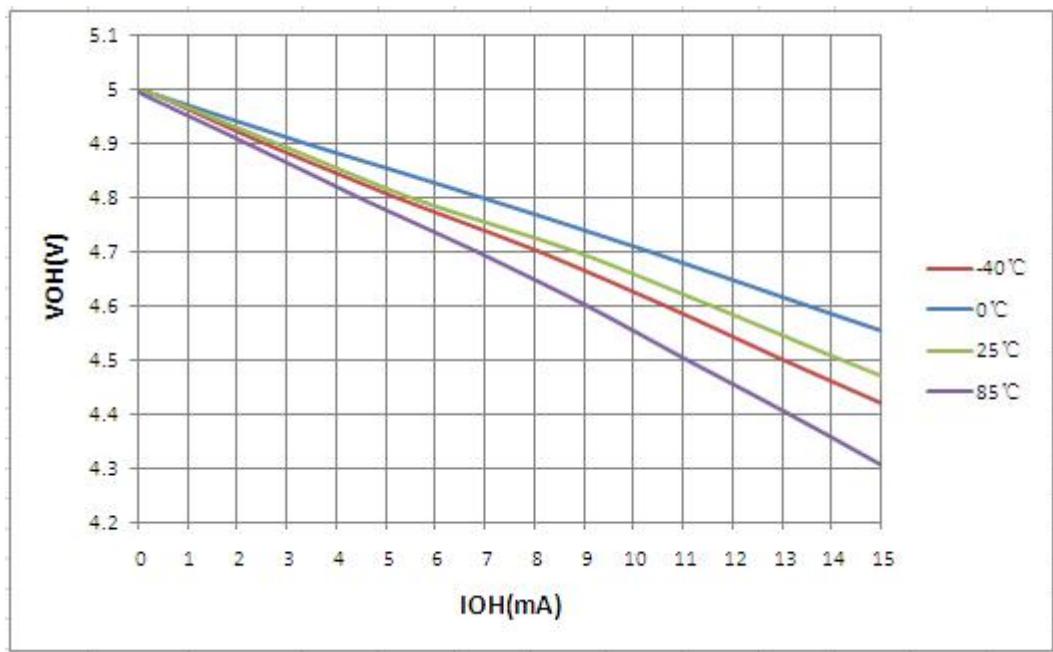
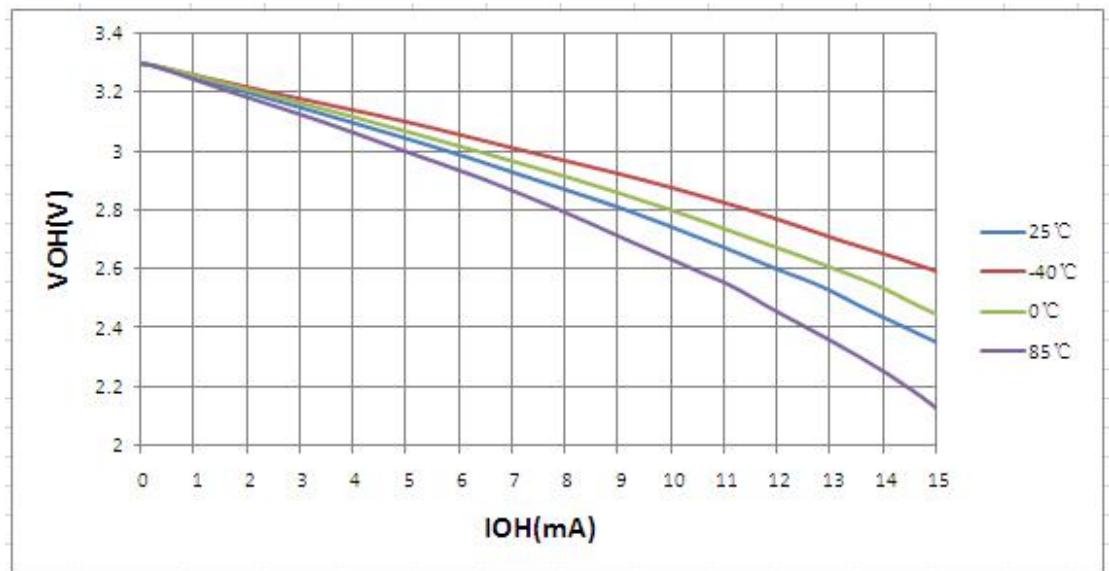


图 19.10 欠压复位电流 ILVR — VDD 关系曲线图

图 19.11 不同温度时 VOH — IOH 关系曲线图 ($VDD = 5.0V$)图 19.12 不同温度时 VOH — IOH 关系曲线图 ($VDD = 3.0V$)

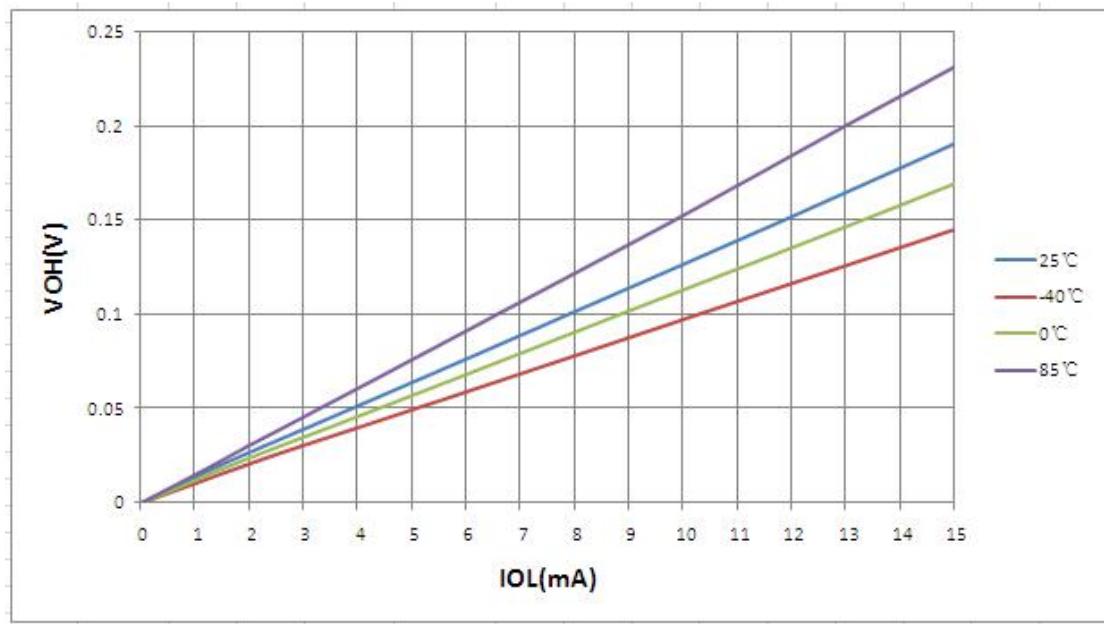


图 19.13 不同温度时 $VOH - IOL$ 关系曲线图 ($VDD = 5.0V$)

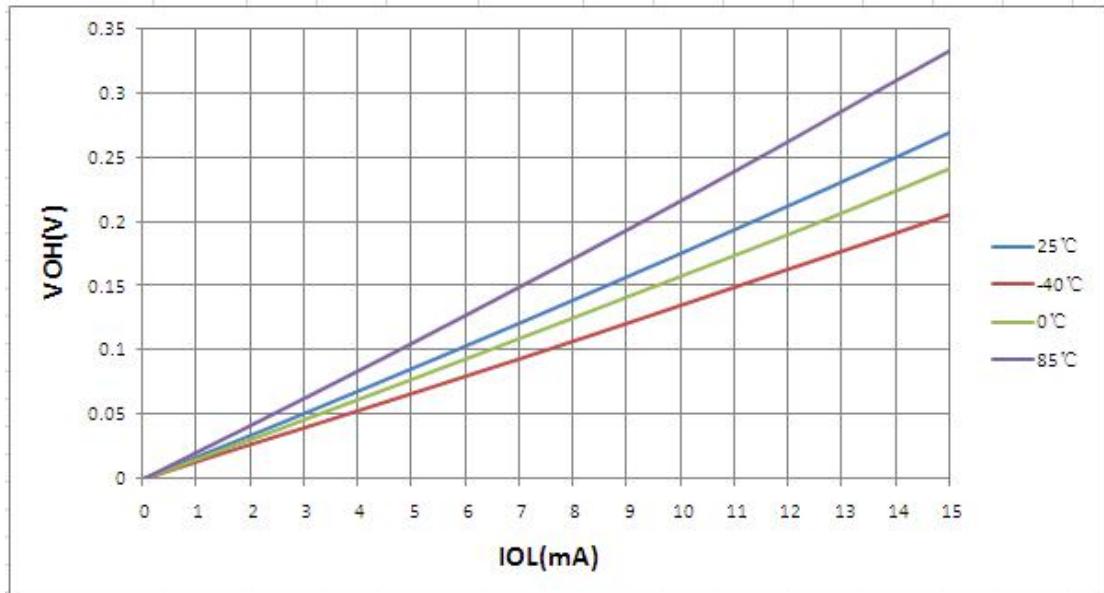


图 19.14 不同温度时 $VOH - IOL$ 关系曲线图 ($VDD = 3.0V$)

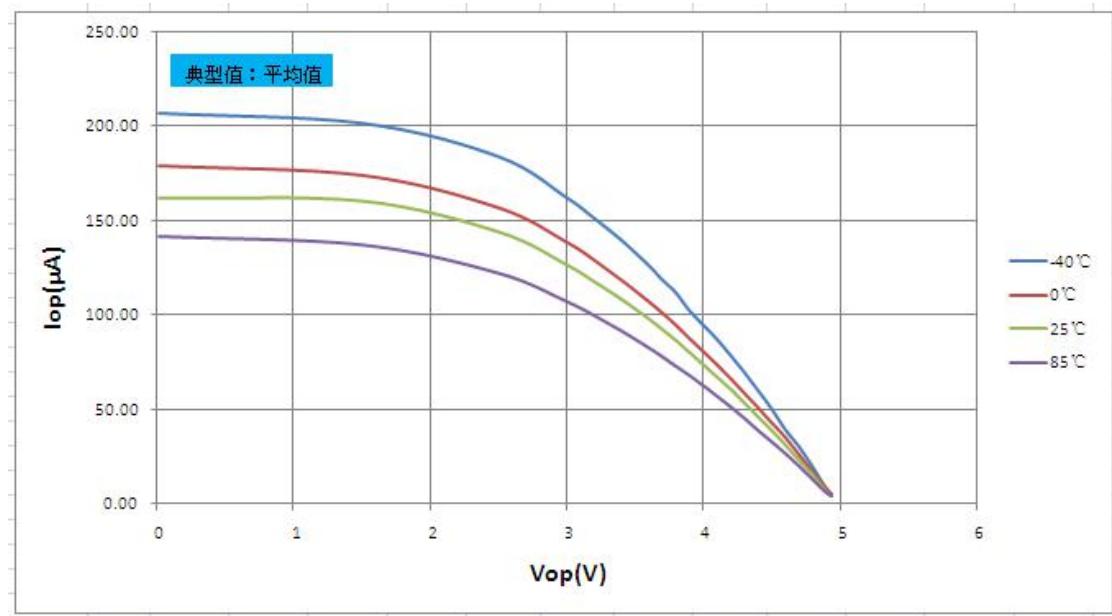


图 19.15 上拉功能打开时端口电压 V_{OP} – 电流 I_{OP} 关系曲线图 ($V_{DD} = 5.0V$)

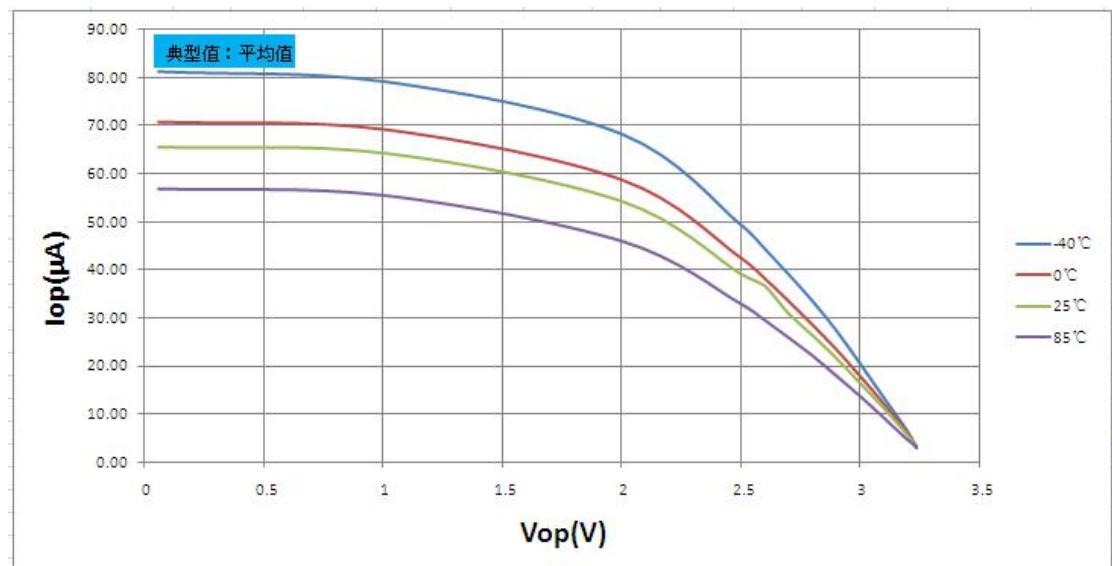
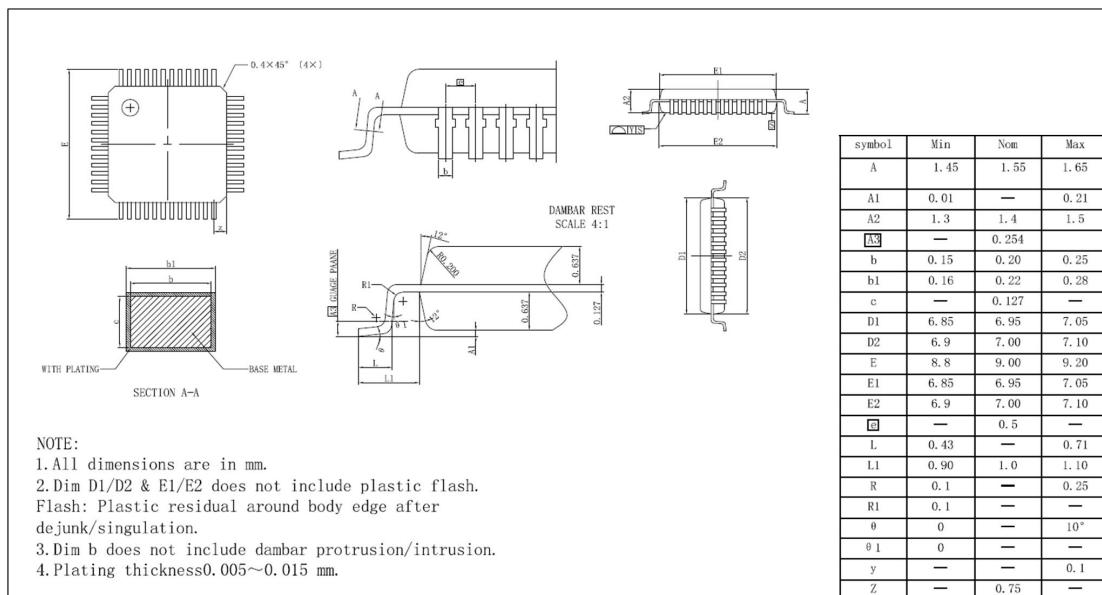


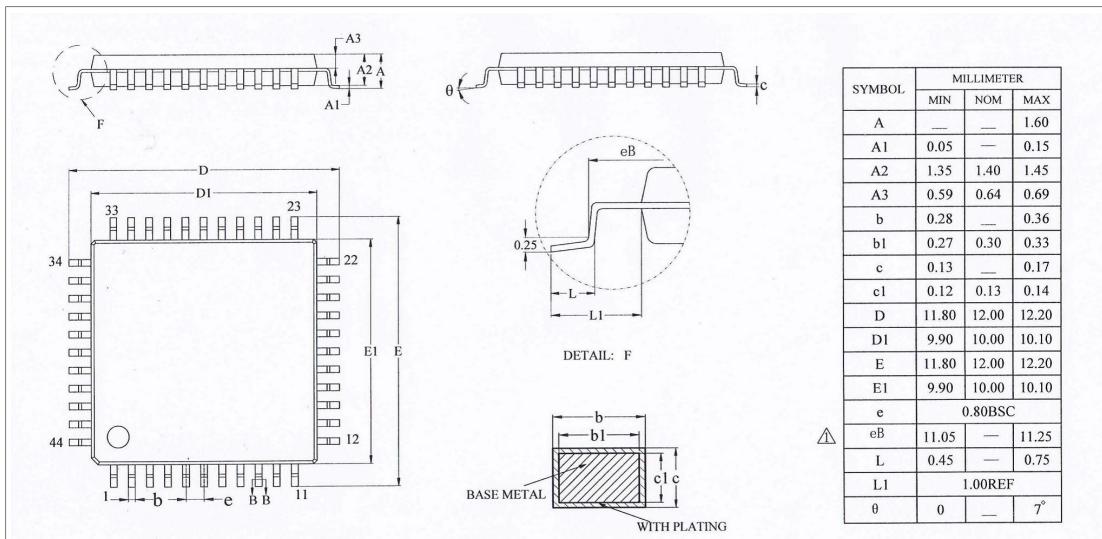
图 19.16 上拉功能打开时端口电压 V_{OP} – 电流 I_{OP} 关系曲线图 ($V_{DD} = 3.3V$)

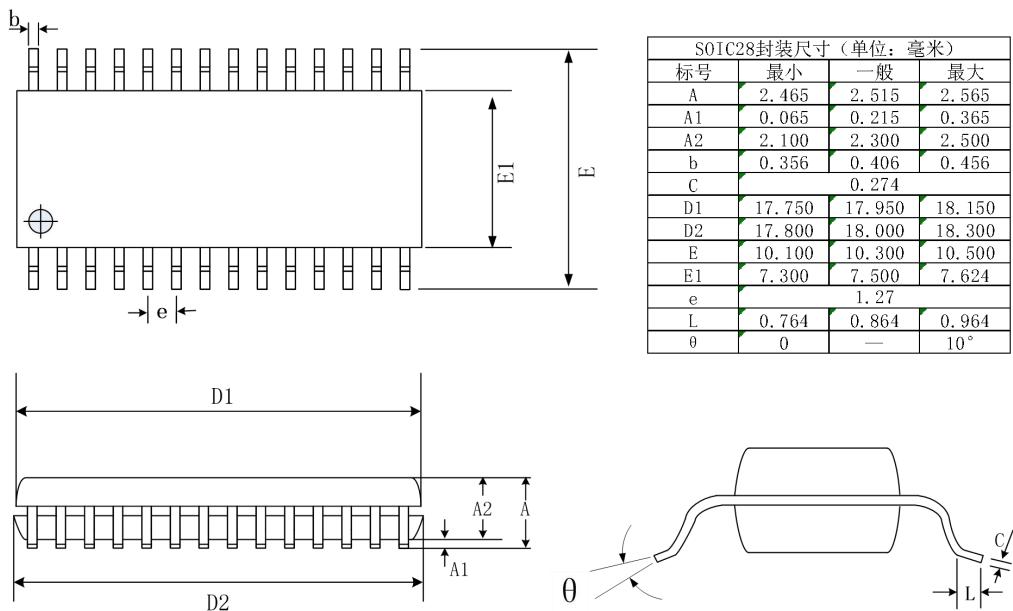
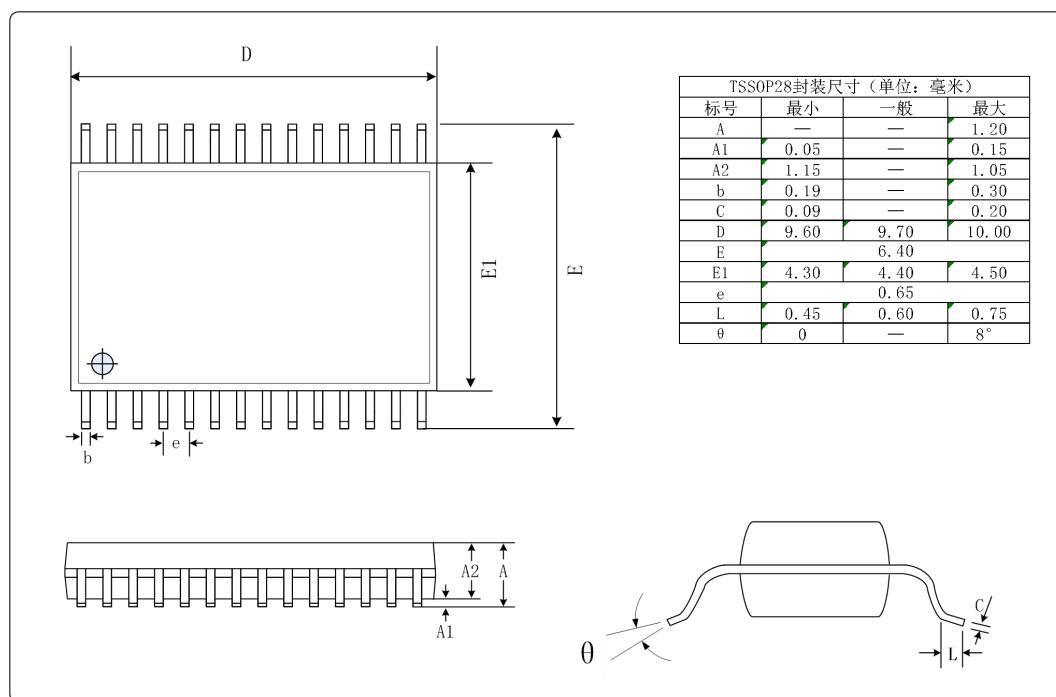
20 封装信息

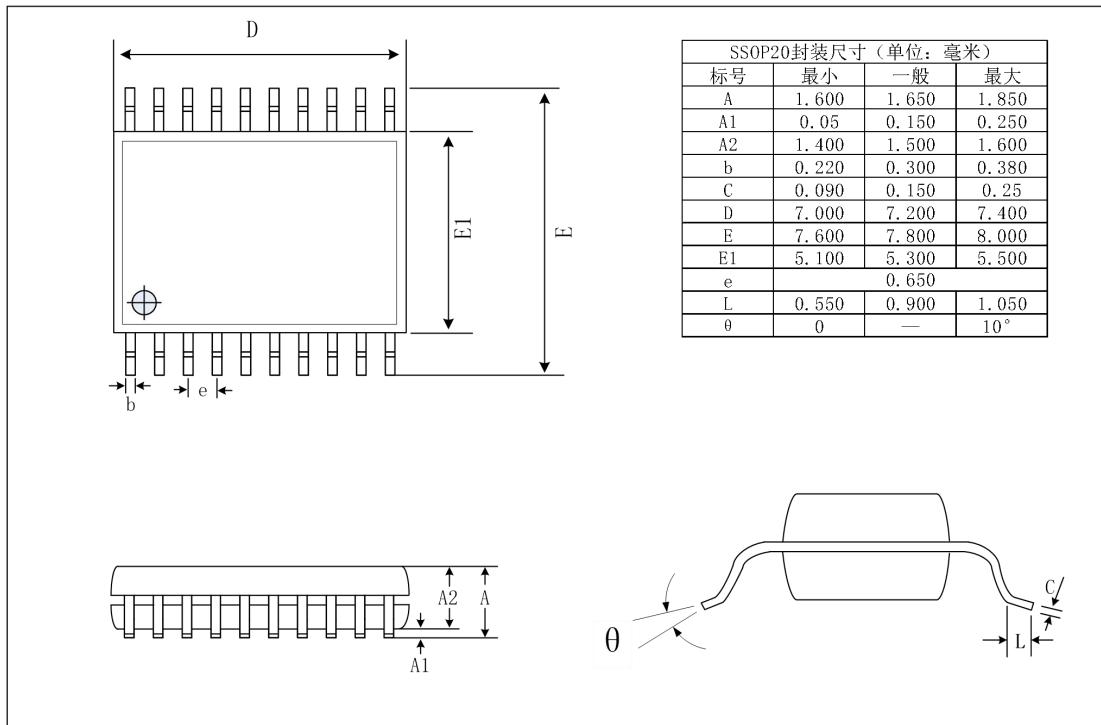
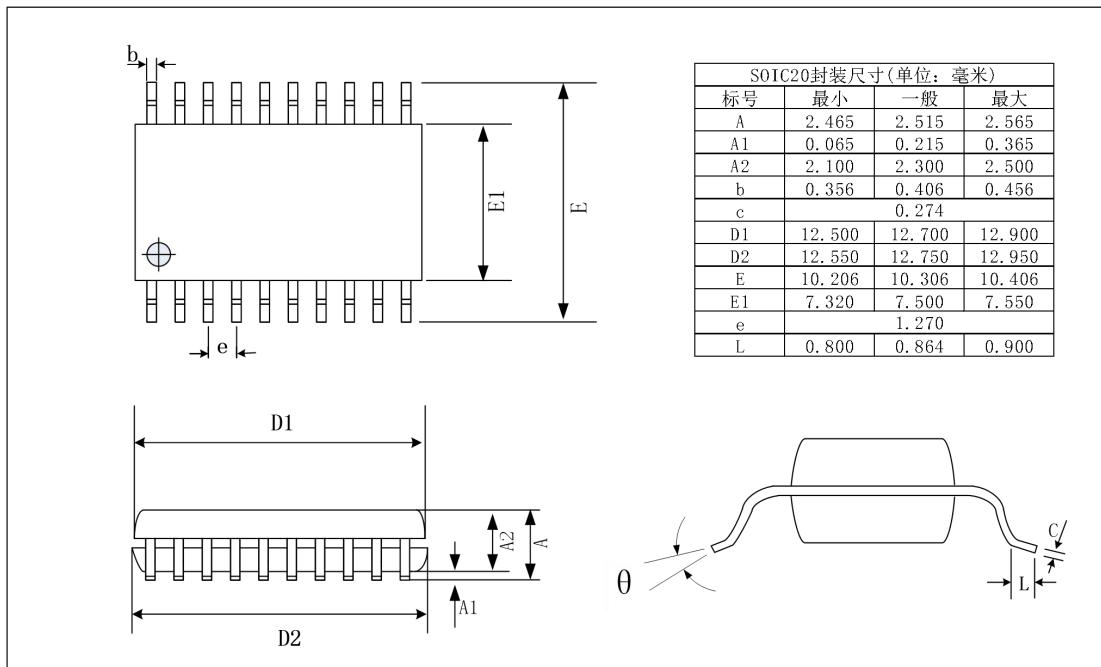
LQFP-48 封装:



LQFP-44 封装:



28 脚 SOIC 封装

28 脚 TSSOP 封装


20 脚 SSOP 封装

20 脚 SOIC 封装


附录 1 特殊功能寄存器 (SFR) 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									XXXX XXXX	
02H	PCL									0000 0000	
03H	PSW	-	-	-	TO	PD	Z	DC	CY	---1 1xxx	
05H	P0	P07	P06	P05	P04	P03	P02	P01	P00	XXXX XXXX	
06H	P2	P27	P26	P25	P24	P23	P22	P21	P20	XXXX XXXX	
07H	P1	P17	P16	P15	P14	P13	P12	P11	P10	XXXX XXXX	
08H	P3	P37	P36	P35	P34	P33	P32	P31	P30	XXXX XXXX	
0AH	PCH	-	-	-						---0 0000	
0BH	INTCTL	AIE/ AIEH	PUIE /AIEL	TOIE	INT0IE	POIE	TOIF	INT0IF	POIF	0000 0000	
0CH	EIF1	EEIF	ADIF	INT2IF	INT1IF	-	PWM2IF	T2IF	PWM1IF/T1IF	0000 0000	
0DH	EIF2	T3IF	-	RC1IF	TX1IF	-	CCP5IF	BCLIF	SSCIIF	0-00-000	
0EH	T1L									XXXX XXXX	
0FH	T1H									XXXX XXXX	
10H	T1CTL	T1RLD	T1GC	T1CKS1	T1CKS0	T1OSCEN	T1SY	T1CS	T1ON	0000 0000	
11H	T2L									0000 0000	
12H	T2CTL0	-	T2CKBS3	T2CKBS2	T2CKBS1	T2CKBS0	T2ON	T2CKPS1	T2CKPS0	-000 0000	
13H	CCP3CTL				CCP3OE	CCP3 MOD3	CCP3 MOD2	CCP3 MOD1	CCP3 MOD0	0000 0000	
14H	CCP3H				CCP3 寄存器高 8 位					0000 0000	
15H	CCP3L				CCP3 寄存器低 8 位					0000 0000	
16H											
17H	BANK	-	-	-	-	PR3	PR2	PR1	PR0	---- 0000	
18H	ADSCANCTL	ADSCANEN	SMOD1	SMOD0	STIM4	STIM3	STIM2	STIM1	STIM0	0000 0000	
1BH	AMPCTL	AMPCALD	-	INPS1	INPS0	-	-	AMPCALEN	AMPON	1000 0000	
1DH	ANS1	ANS17	ANS16	ANS15	ANS14	ANS13	ANS12	ANS11	ANS10	0000 0000	
1EH	ADCDATA0H				ADC 数据寄存器 0 高字节					XXXX XXXX	
1FH	ADCCTL0	ADLR	T2CCR0ON	-	-	-	-	START	ADEN	00--00	
20H	AMPDT	AMPDT7	AMPDT6	AMPDT5	AMPDT4	AMPDT3	AMPDT2	AMPDT1	AMPDT0	1000 0000	
21H	OPTR	PUPH	INT0SE	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	
22H	IP0	-	-	-	-	-	PT0	PINT0	PP0	---- 0000	
23H	IP1	PEE	PADC	PINT2	PINT1	-	PPWM2	PT2	PPWM1/PT1	0000 0000	
24H	IP2	PT3	-	PRC1	PTX1	-	PCCP5	PBCL	PSSCI	0-00-000	
25H	TR0	TR07	TR06	TR05	TR04	TR03	TR02	TR01	TR00	1111 1111	
26H	TR2	TR27	TR26	TR25	TR24	TR23	TR22	TR21	TR20	1111 1111	
27H	TR1	TR17	TR16	TR15	TR14	TR13	TR12	TR11	TR10	1111 1111	
28H	OSCSTA	-	OSTS	HTS	LTS	-	-	SCF1	SCF0	-10--00	
29H	IP3	PT4	-	POSCFAIL	PCCP4	PCCP3	-	PP4	-	0000 0000	
2BH	VREFCTL	VREFSEL1	VREFSEL0	VREF CALEN	CALEN	CLKEN	-	-	VREFEN	-	0000 0000
2CH	EIE1	EEIE	ADIE	INT2IE	INT1IE	-	PWM2IE	T2IE	PWM1IE/T1IE	0000 0000	
2DH	EIE2	T3IE	-	RC1IE	TX1IE	-	CCP5IE	BCLIE	SSCIIE	0000 0000	
2EH	PCTL	-	-	-	SLVREN	IPEN	SWDTEN	POR	LVR	--01 00xx	
2FH	QSCCTL	CLKOE	IRCS2	IRCS1	IRCS0	SCS1	SCS0	IESO	FSCM	0010 0000	
30H	OSCCAL0				内部高晶振校准值寄存器 0					1000 0000	
31H	ANS0	ANS07	ANS06	ANS05	ANS04	ANS03	ANS02	ANS01	ANS00	0000 0000	
32H	ANS2	ANS27	ANS26	ANS25	ANS24	ANS23	ANS22	ANS21	ANS20	0000 0000	
33H	ANS3	ANS37	ANS36	ANS35	ANS34	ANS33	ANS32	ANS31	ANS30	0000 0000	
34H	OSCCAL2	-	-	-	-	-				0000 0101	
35H	PUR0	PUR07	PUR06	PUR05	PUR04	PUR03	PUR02	PUR01	PUR00	1111 1111	
36H	IOCL0	IOCL07	IOCL06	IOCL05	IOCL04	IOCL03	IOCL02	IOCL01	IOCL00	0000 0000	
37H	OSCCAL1				内部高晶振校准值寄存器 1					---- -100	
38H	NVMDATAH				NVM 数据高 8 位寄存器					0000 0000	
39H	NVMDATA0				NVM 数据低 8 位寄存器					0000 0000	
3AH	NVMADDRH				NVM 地址指针高 8 位寄存器					0000 0000	
3BH	NVMADDRL				NVM 地址指针低 8 位寄存器					0000 0000	
3CH	NVMCTL0				NVM 控制寄存器 0					---- ----	
3DH	NVMCTL1				NVM 控制寄存器 1					---- ----	
3EH	ADCDATA0L				ADC 数据寄存器 0 低字节					XXXX XXXX	
3FH	ADCCTL1	ADCALEN	ADCS2	ADCS1	ADCS0	VCFG1	VCFG0	-	-	0000 00-0	
40H	T2CCR0H				T2 触发 ADC 启动设置寄存器 0 高 8 位					0000 0000	
41H	T2H				定时器 2(T2)高字节寄存器					XXXX XXXX	
42H	PP5H				PWM5 周期寄存器高 8 位					XXXX XXXX	
43H	PWM5H0				PWM5 通道 1 占空比寄存器高 8 位					XXXX XXXX	
44H	PWM5H1				PWM5 通道 1 占空比缓冲寄存器高 8 位 (与用户无关)					XXXX XXXX	
45H	P0LR	P0LR7	P0LR6	P0LR5	P0LR4	P0LR3	P0LR2	P0LR1	P0LR0	XXXX XXXX	
46H	P2LR	P2LR7	P2LR6	P2LR5	P2LR4	P2LR3	P2LR2	P2LR1	P2LR0	XXXX XXXX	
47H	P1LR	P1LR7	P1LR6	P1LR5	P1LR4	P1LR3	P1LR2	P1LR1	P1LR0	XXXX XXXX	

地址	名称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	复位初值
48H	P3LR	P3LR7	P3LR6	P3LR5	P3LR4	P3LR3	P3LR2	P3LR1	P3LR0	XXXX XXXX
49H	TR3	TR37	TR36	TR35	TR34	TR33	TR32	TR31	TR30	1111 1111
4AH	EIE3	T4IE	-	OSCFAILIE	CCP4IE	CCP3IE	-	P4IE	-	0000 0000
4BH	EIF3	T4IF	-	OSCFAILIF	CCP4IF	CCP3IF	-	P4IF	-	0000 0000
4CH	OSCCAL2				内部高晶振校准寄存器 2					1000 1000
4DH	OSCCAL3				内部高晶振校准寄存器 3					0001 0000
4EH	T3CTL	T3REN	T3BUZOE	T3CKS1	T3CKS0	-	T3CS1	T3CS0	T3ON	0000 0000
4FH	T3L				T3 低位寄存器					0000 0000
50H	ADCDATA1H				ADC 数据寄存器 1 高字节					0000 0000
51H	ADCDATA1L				ADC 数据寄存器 1 低字节					0000 0000
52H	PP5L				PWM5 周期寄存器低 8 位					1111 1111
54H	T2CCR0L				T2 触发 ADC 启动设置寄存器 0 低 8 位					0000 0000
55H	PWM5L0				捕捉比较模式时, PWM5L0=CCP 数据寄存器低 8 位 PWM5 模式时, PWM5L0= PWM5 通道 1 占空比设置寄存器低 8 位					XXXX XXXX
56H	PWM5L1				捕捉比较模式时, PWM5L1=CCP 数据寄存器高 8 位 PWM5 模式时, PWM5L1= PWM5 通道 1 低 8 位占空比缓冲寄存器 (与用户无关)					XXXX XXXX
57H	PWM5CTL0	P5CH1MOD1	P5CH1MOD0	P5CH3MOD	P5CH2MOD	P5MOD3	P5MOD2	P5MOD1	P5MOD0	0000 0000
58H	ADCINTCTL	T2CCR1ON	-	-	-	-	-	INTCTL1	INTCTL0	0-- --00
59H	ADCDATA2H				ADC 数据寄存器 2 高字节					0000 0000
5AH	ADCDATA2L				ADC 数据寄存器 2 低字节					0000 0000
5BH	PWM5CTL1	P5RSEN	P5DC6	P5DC5	P5DC4	P5DC3	P5DC2	P5DC1	P5DC0	0000 0000
5CH	P5ASCTL0	P5ASE	P5ASS2	P5ASS1	P5ASS0	P5SSAC1	P5SSAC0	P5SSBD1	P5SSBD0	0000 0000
5DH	PSTRCTL0	-	-	-	STRSYNC	STREND	STRENC	STRENB	STRENA	--0 0001
5EH	ADCDATA3H				ADC 数据寄存器 3 高字节					0000 0000
5FH	T3H				T3 高位寄存器					0000 0000
60H	PUR1	PUR17	PUR16	PUR15	PUR14	PUR13	PUR12	PUR11	PUR10	0000 0000
61H	PUR2	PUR27	PUR26	PUR25	PUR24	PUR23	PUR22	PUR21	PUR20	0000 0000
62H	TMRBUZ	-	-	-	-	-	-	T1H_BUZEN	T1L_BUZEN	----00
63H	ADCCCTL2	-	ADCHS6	ADCHS5	ADCHS4	ADCHS3	ADCHS2	ADCHS1	ADCHS0	----0000
67H	INTEDGCTL	INT2SE	INT1SE	-	-	-	-	-	T1CLKSE	00--1
6AH	ADCDATA3L				ADC 数据寄存器 3 低字节					0000 0000
6BH	IOCL4	IOCL47	IOCL46	IOCL45	IOCL44	IOCL43	IOCL42	IOCL41	IOCL40	0000 0000
6CH	PUR3	PUR37	PUR36	PUR35	PUR34	PUR33	PUR32	PUR31	PUR30	0000 0000
105H	CCP4L				CCP4 寄存器低 8 位					0000 0000
106H	CCP4H				CCP4 寄存器高 8 位					0000 0000
108H	DIVRH				余数高 8 位寄存器					0000 0000
109H	CCP4CTL	-	-	-	CCP4OE	CCP4MOD3	CCP4MOD2	CCP4MOD1	CCP4MOD0	--0 0000
10CH	PWM5L2				PWMS 通道 2 占空比寄存器低 8 位					0000 0000
10DH	PWM5L3				PWMS 通道 3 占空比寄存器低 8 位					0000 0000
10EH	PWM5OC	-	-	OCA3	OCB3	OCA2	OCB2	OCA	OCB	--00 0000
10FH	T2CTL1	T2MOD1	T2MOD0	T2OFS1	T2OFS0	T2TRIGEN	-	-	T2DIR	0000 0--0
115H	DIVBH				除数高 8 位寄存器					0000 0000
116H	T2CCR1L				T2 触发 ADC 启动设置寄存器 1 低 8 位					0000 0000
118H	T2CCR1H				T2 触发 ADC 启动设置寄存器 1 高 8 位					0000 0000
119H	PWM5FC	-	-	FCA3	FCB3	FCA2	FCB2	FCA	FCB	--00 0000
11AH	P5ASCTL1	P5SSA31	P5SSA30	P5SSB31	P5SSB30	P5SSA21	P5SSA20	P5SSB21	P5SSB20	0000 0000
11BH	PSTRCTL1	-	-	-	-	STRENA3	STRENB3	STRENA2	STRENB2	----0000
11CH	PWM5CTL2	-	-	PFUSES	-	-	UDEVT1	UDEVT0	UDEN	--0 -000
11DH	PWM5PC	-	-	PCA3	PCB3	PCA2	PCB2	PCA	PCB	--00 0000
11EH	PWM5H2				PWMS 通道 2 占空比寄存器高 8 位					0000 0000
11FH	PWM5H3				PWMS 通道 3 占空比寄存器高 8 位					0000 0000
120H	RSCTL1	SPEN1	R1X9	SRXEN1	CRXEN1	ADREN1	FRER1	OVFER1	RX9D1	0000 000x
121H	TXSDR1				USART 发送数据寄存器 1					0000 0000
122H	RXSDR1				USART 接收数据寄存器 1					0000 0000
123H	BRCTL1	ABRDOVF1	RCIDL1	SCKPS1	B1RG16	BR1CKS1	BR1CKS0	WUEN1	ABRDEN1	0100 0000
124H	TSCTL1	CSRS1	T1X9	TXEN1	SYNC1	SENDB1	HBRG1	TXSRS1	TX9D1	0000 0010
125H	EUBRGL1				USART1 波特率数据寄存器低字节					0000 0000
126H	EUBRGH1				USART1 波特率数据寄存器高字节					0000 0000
127H	USLPEN1	SLPEN1			-	-	-	-	-	0-- --
128H	SSCICLTL0	SSCIWCFL	SSCIOV	SSCIEN	SSCICKP	SSCIMOD3	SSCIMOD2	SSCIMOD1	SSCIMOD0	0000 0000
12AH	SSCICLTL1	SSCICALLEN	SSCIACKSTA	SSCIACKDAT	SSCIACKEN	SSCIRCN	STOPEN	RESTARTEN	STARTEN	0000 0000
12BH	SSCICSTA	SAMPLE	CKEGE	SSCIDA	SSCISTOP	SSCISTART	SSCIRW	SSCIUA	SSCIBUF	0000 0000
12CH	SSCIBUFR				SSCI 数据接收缓冲/发送寄存器					0000 0000
12DH	UARTMTH1				UART1 地址匹配设置寄存器					0000 0000
12EH	SSCIADD				SSCI 的 I2C 地址寄存器					1111 1111
12FH	SSCIMSK	SSCIMSK7	SSCIMSK6	SSCIMSK5	SSCIMSK4	SSCIMSK3	SSCIMSK2	SSCIMSK1	SSCIMSK0	1111 1111
130H	MULAH				16 位乘法器乘数 A 高 8 位					0000 0000
131H	MULAL				16 位乘法器乘数 A 低 8 位					0000 0000
132H	MULBH				16 位乘法器乘数 B 高 8 位					0000 0000
133H	MULBL				16 位乘法器乘数 B 低 8 位					0000 0000

地址	名称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	复位初值
134H	MULCTL	-	-	-	-	-	-	MULEN	MULIF	---- --00
135H	MULRES3									0000 0000
136H	MULRES2									0000 0000
137H	MULRES1									0000 0000
138H	MULRES0									0000 0000
139H	DIVCTL	-	-	-	-	-	-	DIVOEN	DIVEN	0000 0000
13AH	DIVAH									0000 0000
13BH	DIVAL									0000 0000
13CH	DIVBL									0000 0000
13DH	DIVQH									0000 0000
13EH	DIVQL									0000 0000
13FH	DIVRL									0000 0000
140H	LEDCTL	LEDEN	-	-	COMEN2	COMEN1	COMEN0	PNS1	PNS0	0--0 0000
141H	LEDPRE	LEDCKS13	LEDCKS12	LEDCKS11	LEDCKS10	LEDCKS03	LEDCKS02	LEDCKS01	LEDCKS00	0000 0000
142H	LEDDATA0	DAT07	DAT06	DAT05	DAT04	DAT03	DAT02	DAT01	DAT00	0000 0000
143H	LEDDATA1	DAT17	DAT16	DAT15	DAT14	DAT13	DAT12	DAT11	DAT10	0000 0000
144H	LEDDATA2	DAT27	DAT26	DAT25	DAT24	DAT23	DAT22	DAT21	DAT20	0000 0000
145H	LEDDATA3	DAT37	DAT36	DAT35	DAT34	DAT33	DAT32	DAT31	DAT30	0000 0000
146H	LEDDATA4	DAT47	DAT46	DAT45	DAT44	DAT43	DAT42	DAT41	DAT40	0000 0000
147H	LEDDATA5	DAT57	DAT56	DAT55	DAT54	DAT53	DAT52	DAT51	DAT50	0000 0000
148H	LEDDATA6	DAT67	DAT66	DAT65	DAT64	DAT63	DAT62	DAT61	DAT60	0000 0000
149H	LEDDATA7	DAT77	DAT76	DAT75	DAT74	DAT73	DAT72	DAT71	DAT70	0000 0000
14AH	LEDDATA8	DAT87	DAT86	DAT85	DAT84	DAT83	DAT82	DAT81	DAT80	0000 0000
14BH	LEDDATA9	DAT97	DAT96	DAT95	DAT94	DAT93	DAT92	DAT91	DAT90	0000 0000
14CH	LEDDATAA	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0	0000 0000
14DH	LEDDATAB	DATB7	DATB6	DATB5	DATB4	DATB3	DATB2	DATB1	DATB0	0000 0000
14EH	LEDDATAC	DATC7	DATC6	DATC5	DATC4	DATC3	DATC2	DATC1	DATC0	0000 0000
14FH	LEDDATAD	DATD7	DATD6	DATD5	DATD4	DATD3	DATD2	DATD1	DATD0	0000 0000
150H	UPINSET1	USLM1	UPSEL1	-	-	-	-	-	-	0000 0000
151H	LPRCCCTL	-	-	-	-	-	-	LPRC2	LPRC1	LPRC0
152H	LPRCCAL									0000 0000
153H	XTALCAL									0101 0000
154H	PCAL	-	-	PCAL5	PCAL4	PCAL3	PCAL2	PCAL1	PCAL0	0000 0000
155H	LEDDATAE	DATE7	DATE6	DATE5	DATE4	DATE3	DATE2	DATE1	DATE0	0000 0000
156H	LEDDATAF	DATE7	DATE6	DATE5	DATE4	DATE3	DATE2	DATE1	DATE0	0000 0000
157H	LEDCTL2	-	-	-	-	-	LEDSEGEN16	LEDSEGEN12	LEDSEGEN8	0000 0000
158H	LEDOMS	LEDOM7	LEDOM6	LEDOM5	LEDOM4	LEDOM3	LEDOM2	LEDOM1	LEDOM0	0000 0000
159H	VREFCAL0									0000 0000
15AH	VREFCAL1									1010 0000
15CH	LEDLUM	LEDBR	LEDIOS	ODEN	ODSET	FRAMEF	LUM2	LUM1	LUM0	0---0 000
15DH	T3REL									0000 0000
15EH	T3REH									0000 0000
160H	T4L									0000 0000
161H	T4H									0000 0000
162H	T4REL									0000 0000
163H	T4REH									0000 0000
164H	T4CTL	T4REN	T4BUZOE	T4CKS1	T4CKS0	-	T4CS1	T4CS0	T4ON	0000 0000
165H	RC32KCAL									0000 0000
206H	ANS4	ANS47	ANS46	ANS45	ANS44	ANS43	ANS42	ANS41	ANS40	1111 1111
207H	ANS5	ANS57	ANS56	ANS55	ANS54	ANS53	ANS52	ANS51	ANS50	1111 1111
20CH	P4	P47	P46	P45	P44	P43	P42	P41	P40	XXXX XXXX
20DH	P4LR	P4LR7	P4LR6	P4LR5	P4LR4	P4LR3	P4LR2	P4LR1	P4LR0	XXXX XXXX
20EH	TR4	TR47	TR46	TR45	TR44	TR43	TR42	TR41	TR40	1111 1111
20FH	PUR4	PUR47	PUR46	PUR45	PUR44	PUR43	PUR42	PUR41	PUR40	1111 1111
210H	P5	P57	P56	P55	P54	P53	P52	P51	P50	XXXX XXXX
211H	P5LR	P5LR7	P5LR6	P5LR5	P5LR4	P5LR3	P5LR2	P5LR1	P5LR0	XXXX XXXX
212H	TR5	TR57	TR56	TR55	TR54	TR53	TR52	TR51	TR50	1111 1111
213H	PUR5	PUR57	PUR56	PUR55	PUR54	PUR53	PUR52	PUR51	PUR50	1111 1111
21DH	PWMCTL0	-	-	-	-	PWM13ON	PWM12ON	PWM11ON	PWM10ON	0000 0000
21EH	PWMCTL1	-	-	-	-	PWM23ON	PWM22ON	PWM21ON	PWM20ON	0000 0000
21FH	PP1									1111 1111
220H	PP2									1111 1111
221H	PWM10L									XXXX XXXX
222H	PWM20H									XXXX XXXX
223H	PWM11L									XXXX XXXX
224H	PWM21H									XXXX XXXX
225H	PWM12L									XXXX XXXX
226H	PWM22H									XXXX XXXX
227H	PWM13L									XXXX XXXX
228H	PWM23H									XXXX XXXX
229H	PWM3CTL	-	-	-	-	PWM33ON	PWM32ON	PWM31ON	PWM30ON	0000 0000
22AH	PP3H									1111 1111
22BH	PP3L									1111 1111
22CH	PWM30H									XXXX XXXX
22DH	PWM31H									XXXX XXXX
22EH	PWM32H									XXXX XXXX

地址	名称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	复位初值
22FH	PWM33H				PWM33 占空比高 8 位寄存器					XXXX XXXX
230H	PWM30L				PWM30 占空比低 8 位寄存器					XXXX XXXX
231H	PWM31L				PWM31 占空比低 8 位寄存器					XXXX XXXX
232H	PWM32L				PWM32 占空比低 8 位寄存器					XXXX XXXX
233H	PWM33L				PWM33 占空比低 8 位寄存器					XXXX XXXX
236H	TEMPSNR	-	-	-	-	-	-	-	TSEN	---- --00
23FH	UPINSET2	USLM2	UPSEL2	-	-	-	-	-	-	0000 0000
240H	U7816CTL2	PSEL2	-	TREPEN2	TREP21	TREP20	RREPEN2	RREP21	RREP20	0-00 0000
241H	U7816 RXCTL2	ERSW21	ERSW20	RPAR2	RINV2	RCONV2	-	-	PAREF2	0000 0000
242H	U7816 TXCTL2	U7816EN2	CLKOUT2	TX9DSEL2	STOP2	TPAR2	TINV2	TCONV2	BGTEN2	0001 0000
243H	CLKDIV2	CLKDIV27	CLKDIV26	CLKDIV25	CLKDIV24	CLKDIV23	CLKDIV22	CLKDIV21	CLKDIV20	0000 0000
244H	EGTCTL2	EGT27	EGT26	EGT25	EGT24	EGT23	EGT22	EGT21	EGT20	0000 0000
245H	TZBRGH2	-	-	-	-	-	-	-	-	0000 0000
246H	TZBRLG2	-	-	-	-	-	-	-	-	0000 0000
247H	UARTMTH2				UART2 地址匹配设置寄存器					0000 0000
248H	USLPEN2	SLPEN2	-	-	-	-	-	-	-	0- - -
249H	BRCTL2	ABRDOVF2	RCIDL2	SCKPS2	B2RG16	BR2CKS1	BR2CKS0	WUEN2	ABRDEN2	0100 0000
24AH	RSCTL2	SPEN2	R2X9	SRXEN2	CRXEN2	ADREN2	FRER2	OVFER2	RX9D2	0000 0000
24BH	RXSDR2				USART 接收数据寄存器 2					XXXX XXXX
24CH	EUBRGH2				USART2 波特率数据寄存器高字节					0000 0000
24DH	EUBRGL2				USART2 波特率数据寄存器低字节					0000 0000
24EH	TXSDR2				USART 发送数据寄存器 2					XXXX XXXX
24FH	TSCTL2	CSRS2	T2X9	TXEN2	SYNC2	SEND2	HBRG2	TXSRS2	TX9D2	0000 0000
250H	BKPCTL	BKP7	BKP6	BKP5	BKP4	BKP3	BKP2	BKP1	BKP0	0000 0000
251H	EIE4	-	-	-	-	RE2IE	TE2IE	RC2IE	TX2IE	---- 0000
252H	EIF4	-	-	-	-	RE2IF	TE2IF	RC2IF	TX2IF	---- 0000
253H	IP4	-	-	-	-	PRE2	PTE2	PRC2	PTX2	---- 0000
254H	CTCHCOMB0	COMB7	COMB6	COMB5	COMB4	COMB3	COMB2	COMB1	COMB0	0000 0000
255H	CTCHCOMB1	COMB15	COMB14	COMB13	COMB12	COMB11	COMB10	COMB9	COMB8	0000 0000
256H	CTCHCOMB2	COMB23	COMB22	COMB21	COMB20	COMB19	COMB18	COMB17	COMB16	0000 0000
260H	APFCTL0	FVROE1	FVROE0	VREFINSEL1	VREFINSEL0	P18OE1	P18OE0	CLKOUTSEL1	CLKOUTSEL0	0000 0000
261H	APFCTL1	CCP3SEL2	CCP3SEL1	CCP3SEL0	CCP4SEL2	CCP4SEL1	CCP4SEL0	-	-	0000 0000
262H	APFCTL2	P5DSEL1	P5DSEL0	P5CSEL1	P5CSEL0	P5BSEL1	P5BSEL0	P5ASEL1	P5ASEL0	0000 0000
263H	APFCTL3	PWM33 SEL	PWM32 SEL	PWM31 SEL	-	PWM22 SEL	PWM21 SEL	PWM20 SEL1	PWM20 SEL0	0000 0000
264H	APFCTL4	-	U1TCSEL2	U1TCSEL1	U1TCSEL0	-	-	U1RDSEL1	U1RDSEL0	0000 0000
265H	APFCTL5	-	-	-	-	U2TCSEL1	U2TCSEL0	U2RDSEL1	U2RDSEL0	0000 0000
266H	APFCTL6	-	SDOSEL2	SDOSEL1	SDOSEL0	-	SCKSEL2	SCKSEL1	SCKSEL0	0000 0000
267H	APFCTL7	-	-	-	SSSEL	SDISEL3	SDISEL2	SDISEL1	SDISEL0	0000 0000
268H	APFCTL8	BUZ3SEL	TICKSEL2	TICKSEL1	TICKSEL0	T1GSEL1	T1GSEL0	T0CKSEL1	T0CKSEL0	0000 0000
269H	APFCTL9	-	INT1SEL2	INT1SEL1	INT1SEL0	INT2SEL1	INT2SEL0	INT0SEL1	INT0SEL0	0000 0000
26AH	APFCTLA	COM7 SEL	COM6 SEL	COM5 SEL	COM4 SEL	COM3 SEL	COM2 SEL	COM1 SEL	COM0 SEL	0000 0000
26BH	APFCTLB	-	-	-	-	-	CCP5 SEL2	CCP5 SEL1	CCP5 SEL0	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	指令格式	指令说明	周期	影响标志
NOP	0000_0000_0000_0000	空操作指令	1	
NOPZ	1111_1111_1111_1111	空操作指令	1	
CRET	0000_0000_0000_1000	子程序返回指令	2	
RRET Rn,#data	1011_0rrr_kkkk_kkkk	立即数送到 Rn 中返回	2	
IRET	0000_0000_0000_1001	中断返回指令	2	
CWDT	0000_0000_0110_0100	WDT 清 0	1	
IDLE	0000_0000_0110_0011	进入休眠模式	1	
数据传送指令				
MOV dir	0000_1111_ffff_ffff	dir←(dir)	1	Z
MOV Rn,dir	0101_rrr0_ffff_ffff	Rn←(dir)	1	
MOV dir,Rn	0101_rrr1_ffff_ffff	dir←(Rn)	1	
MOV Rn,#data	1001_1rrr_kkkk_kkkk	Rn←data	1	
MOV Rn,Rs	1111_1000_11ss_srrr	Rn←(Rs)	1	
LD Rn,[Rs]	1111_0111_00ss_srrr	Rn←((Rs))	1	
ST [Rn],Rs	1111_0111_01ss_srrr	(Rn)←(Rs)	1	
SWAPR Rn,dir	0100_rrr0_ffff_ffff	Rn<7:4>=dir<3:0> Rn<3:0>=dir<7:4>	1	
SWAP dir	0100_rrr1_ffff_ffff	dir<7:4>=dir<3:0> dir<3:0>=dir<7:4>	1	
MOVB #data	1110_0001_kkkk_kkkk	BANK←data	1	
MOVP #data	1110_0000_kkkk_kkkk	PCH←data	1	
算术运算指令				
ADD Rm,dir	0010_0rr0_ffff_ffff	Rm←(Rm)+(dir)	1	CY、DC、Z
ADD dir,Rm	0010_0rr1_ffff_ffff	dir←(Rm)+(dir)	1	CY、DC、Z
ADD Rn,#data	1000_0rrr_kkkk_kkkk	Rn←(Rn)+data	1	CY、DC、Z
ADD Rn,Rs	1111_1000_00ss_srrr	Rn←(Rn)+(Rs)	1	CY、DC、Z
SUB Rm,dir	0011_1rr0_ffff_ffff	Rm←(dir)-(Rm)	1	CY、DC、Z
SUB dir,Rm	0011_1rr1_ffff_ffff	dir←(dir)-(Rm)	1	CY、DC、Z
SUB Rn,#data	1010_0rrr_kkkk_kkkk	Rn←data-(Rn)	1	CY、DC、Z
SUB Rn,Rs	1111_1000_01ss_srrr	Rn←(Rs)-(Rn)	1	CY、DC、Z
CMP Rn,#data	1111_0010_1kkk_krrr	-	1	CY、DC、Z
CMP Rn,Rs	1111_0001_10ss_srrr	-	1	CY、DC、Z
INC dir	0000_1011_ffff_ffff	dir←(dir)+1	1	Z
INCR dir	0000_1010_ffff_ffff	R0←(dir)+1	1	Z
INC Rn	1111_1111_0001_0rrr	Rn←(Rn)+1	1	Z
DEC dir	0000_0111_ffff_ffff	dir←(dir)-1	1	Z
DECR dir	0000_0110_ffff_ffff	R0←(dir)-1	1	Z
DEC Rn	1111_1111_0000_1rrr	Rn←(Rn)-1	1	Z
逻辑运算指令				
AND Rm,dir	0010_1rr0_ffff_ffff	Rm←(Rm) ∧ (dir)	1	Z
AND dir,Rm	0010_1rr1_ffff_ffff	dir←(dir) ∧ (Rm)	1	Z
AND Rn,#data	1000_1rrr_kkkk_kkkk	Rn←(Rn) ∧ data	1	Z
AND Rn,Rs	1111_1000_10ss_srrr	Rn←(Rn) ∧ (Rs)	1	Z
ORL Rm,dir	0011_0rr0_ffff_ffff	Rm←(Rm) ∨ (dir)	1	Z
ORL dir,Rm	0011_0rr1_ffff_ffff	dir←(dir) ∨ (Rm)	1	Z
ORL Rn,#data	1001_0rrr_kkkk_kkkk	Rn←(Rn) ∨ data	1	Z
ORL Rn,Rs	1111_1001_00ss_srrr	Rn←(Rn) ∨ (Rs)	1	Z

助记符、操作数	指令格式	指令说明	周期	影响标志
XOR Rm,dir	0001_1rr0_ffff_ffff	$Rm \leftarrow (Rm) \oplus (dir)$	1	Z
XOR dir,Rm	0001_1rr1_ffff_ffff	$dir \leftarrow (dir) \oplus (Rm)$	1	Z
XOR Rn,#data	1010_1rrr_kkkk_kkkk	$Rn \leftarrow (Rn) \oplus data$	1	Z
XOR Rn,Rs	1111_1001_01ss_srrr	$Rn \leftarrow (Rn) \oplus (Rs)$	1	Z
CLR Rn	0000_0010_xxxx_1rrr	$Rn = 0$	1	Z
CLR dir	0000_0011_ffff_ffff	$dir = 0$	1	Z
CPLR dir	0000_0100_ffff_ffff	$R0 \leftarrow \neg(dir)$	1	Z
CPL dir	0000_0101_ffff_ffff	$dir \leftarrow \neg(dir)$	1	Z
CPL Rn	1111_1111_0000_0rrr	$Rn \leftarrow \neg(Rn)$	1	Z
RRCR dir	0001_0000_ffff_ffff	$R0 \leftarrow (dir)$ 带进位 C 循环右移 1 位	1	CY
RRC dir	0001_0001_ffff_ffff	$dir \leftarrow (dir)$ 带进位 C 循环右移 1 位	1	CY
RRC Rn	1111_1111_0010_0rrr	$Rn \leftarrow (Rn)$ 带进位 C 循环右移 1 位	1	CY
RLCR dir	0001_0010_ffff_ffff	$R0 \leftarrow (dir)$ 带进位 C 循环左移 1 位	1	CY
RLC dir	0001_0011_ffff_ffff	$dir \leftarrow (dir)$ 带进位 C 循环左移 1 位	1	CY
RLC Rn	1111_1111_0001_1rrr	$Rn \leftarrow (Rn)$ 带进位 C 循环左移 1 位	1	CY
位操作指令				
CLR dir,b	0110_0bbb_ffff_ffff	将 dir 的 b 位清 0	1	
SET dir,b	0110_1bbb_ffff_ffff	将 dir 的 b 位置 1	1	
CLR Rn,b	1111_1110_00bb_brrr	将 Rn 的 b 位清 0	1	
SET Rn,b	1111_1110_01bb_brrr	将 Rn 的 b 位置 1	1	
转移指令				
DECRJZ dir	0000_1000_ffff_ffff	$R0 \leftarrow (dir)-1$, 为 0 跳过下一条指令	1/2	
DECJZ dir	0000_1001_ffff_ffff	$dir \leftarrow (dir)-1$, 为 0 跳过下一条指令	1/2	
DECJZ Rn	1111_1111_0101_1rrr	$Rn \leftarrow (Rn)-1$, 为 0 跳过下一条指令	1/2	
INCRJZ dir	0000_1100_ffff_ffff	$R0 \leftarrow (dir)+1$, 为 0 跳过下一条指令	1/2	
INCJZ dir	0000_1101_ffff_ffff	$dir \leftarrow (dir)+1$, 为 0 跳过下一条指令	1/2	
INCJZ Rn	1111_1111_0101_0rrr	$Rn \leftarrow (Rn)+1$, 为 0 跳过下一条指令	1/2	
JNB dir,b	0111_0bbb_ffff_ffff	dir 的 b 位为 0 跳过下一条指令	1/2	
JB dir,b	0111_1bbb_ffff_ffff	dir 的 b 位为 1 跳过下一条指令	1/2	
JNB Rn,b	1111_0111_10bb_brrr	Rn 的 b 位为 0 跳过下一条指令	1/2	
JB Rn,b	1111_0111_11bb_brrr	Rn 的 b 位为 1 跳过下一条指令	1/2	
JMP #data12	1100_kkkk_kkkk_kkkk	无条件转移指令	2	
CALL #data12	1101_kkkk_kkkk_kkkk	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

BANK0		
地址	名称	全称
01H	T0	Timer 0 register
02H	PCL	Program Counter Low register
03H	PSW	Program Status Word register
05H	P0	Port 0
06H	P2	Port 2
07H	P1	Port 1
08H	P3	Port 3
0AH	PCH	Program Counter High register
0BH	INTCTL	Interrupt control register
0CH	EIF1	Enable Interrupt Flag register 1
0DH	EIF2	Enable Interrupt Flag register 2
0EH	T1L	Timer 1 register Low
0FH	T1H	Timer 1 register High
10H	T1CTL	Timer 1 Control register
11H	T2L	Timer 2 register Low
12H	T2CTL0	Timer 2 register Control 0
17H	BANK	BANK
18H	ADSCANCTL	Analog Digital Convert Scan control register
19H	C1CTL	Comparer 1 Control register
1AH	C1OUT	Comparer 1 Output register
1BH	AMPCTL	Amplifier Control register
1DH	ANSH	Analog channel Selection register High
1EH	ADCDATA0H	Analog Digital Convert Data 0 High register
1FH	ADCCTL0	Analog Digital Convert Control register 0
20H	AMPDT	Amplifier Data register
21H	OPTR	Option Register
22H	IP0	Interrupt Priority 0 register
23H	IP1	Interrupt Priority 1 register
24H	IP2	Interrupt Priority 2 register
25H	TR0	Tri Register 0
26H	TR2	Tri Register 2
27H	TR1	Tri Register 1
28H	OSCSTA	Oscillator Status register
29H	IP3	Interrupt Priority3 register
2AH		Power voltage Calibration register
2BH	VREFCTL	Reference Voltage Control register
2CH	EIE1	Enable Interrupt Enable register 1
2DH	EIE2	Enable Interrupt Enable register 2
2EH	PCTL	Power Control register
2FH	OSCCTL	Oscillator Control register
30H	OSCCAL2	Oscillator Calibration register 2
31H	ANS0	Analog channel Selection register 0
32H	ANS1	Analog channel Selection register 1
33H	ANS2	Analog channel Selection register 2
34H	ANS3	Analog channel Selection register 3
35H	PUR0	Pull-Up Register 0
36H	IOCL0	P0 Interrupt on change register
37H	OSCCAL1	Oscillator Calibration register 1

38H	NVMDATAH	NVW Buffer Data register High
39H	NVMDATAL	NVM Buffer Data register Low
3AH	NVMADDRH	NVM Buffer Address register High
3BH	NVMADDRL	NVM Buffer Address register Low
3CH	NVMCTL0	Nonvolatile Memory control register 1
3DH	NVMCTL1	Nonvolatile Memory control register 2
3EH	ADCDATA0L	Analog Digital Convert Data 0 register Low
3FH	ADCCTL1	Analog Digital Convert Control register 1
40H	T2CCR0H	Timer2 Compare Capture Register 0 High
41H	T2H	Timer2 High
42H	PP5H	Pulse-Width Modulation Periods register High
43H	PWM5H0	Pulse-Width Modulation 5 duty cycle register High 0
44H	PWM5H1	Pulse-Width Modulation 5 duty cycle register High 1
45H	P0LR	Port 0 Latch Register
46H	P2LR	Port 2 Latch Register
47H	P1LR	Port 1 Latch Register
48H	P3LR	Port 3 Latch Register
49H	TR3	Tri Register 3
4AH	EIE3	Enable Interrupt register 3
4BH	EIF3	Enable Interrupt Flag register 3
4CH	OSCCAL3	Oscillator Calibration register 3
4DH	OSCCAL0	Oscillator Calibration register 0
4EH	T3CTL	Timer 3 Control register
4FH	T3L	Timer 3 register low
50H	ADC DATA1H	Analog Digital Convert Data register 1 High
51H	ADC DATA1L	Analog Digital Convert Data register 1 Low
52H	PP5L	Pulse-Width Modulation Periods 5 register Low
53H	CTCTL0	Capacitance Touch Control register 0
54H	T2CCR0L	Timer 2 Compare Capture Register 0 Low
55H	PWM5L0	Pulse-Width Modulation 5 duty cycle register Low 0
56H	PWM5L1	Pulse-Width Modulation 5 duty cycle register Low 1
57H	PWM5CTL0	Pulse-Width Modulation 5 Control register 0
58H	ADCINTCTL	Analog Digital Convert Interrupt control register
59H	ADC DATA2H	Analog Digital Convert Data register 2 High
5AH	ADC DATA2L	Analog Digital Convert Data register 2 Low
5BH	PWM5CTL1	Pulse-Width Modulation 5 Control register 1
5CH	P5ASCTL0	Pulse-Width Modulation 5 Auto Shutdown Control register 0
5DH	PSTRCTL0	Pulse Auto Steer Control register 0
5EH	ADC DATA3H	Analog Digital Convert Data register 3 High
5FH	T3H	Timer 3 High register
62H	BUZCTL	BUZER Control Register
63H	ADCCTL2	Analog Digital Convert Control register 2
64H	CTCTL1	Capacitance Touch Control register 1
65H	MCTCTL	Matrix Capacitance Touch Control register
66H	MCTI	Matrix Capacitance Touch Current selection
67H	INTEDGCTL	Interrupt Edge Control register
6EH	PUR1	Pull-Up Resistor 1
6FH	PUR2	Pull-Up Resistor 2
6AH	ADC DATA3L	Analog Digital Convert Data register 3 Low
6BH	IOCL3	P3 Interrupt on change register
6CH	PUR3	Pull-up control Register 3

BANK1		
108H	DIVRH	Divider Remainder register High
109H		
10CH	PWM5L2	Pulse-Width Modulation 5 duty cycle register 2 low
10DH	PWM5L3	Pulse-Width Modulation 5 duty cycle register 3 low
10EH	PWM5OC	Pulse-Width Modulation 5 Output Control register
10FH	T2CTL1	Timer 2 Control register1
110H	C4FILCTL	Comparer 4 Filter Control register
111H	C4FILTPRE	Comparer 4 Filter Per-scale register
112H	C2CTL	Comparer 2 Control register
113H	C3CTL	Comparer 3 Control register
114H	C4CTL	Comparer 4 Control register
115H	DIVBH	Divider B register High
116H	T2CCR1L	
118H	T2CCR1H	
119H	PWM5FC	Pulse-Width Modulation 5 Force Control
11AH	P5ASCTL1	Pulse-Width Modulation 5Auto Shutdown Control register 1
11BH	PSTRCTL1	Pulse Auto Steer Control register 1
11CH	PWM5CTL2	Pulse-Width Modulation 5 Control register 2
11DH	PWM5PC	Pulse-Width Modulation 5 Polarity Control register
11EH	PWM5H2	Pulse-Width Modulation 5 duty cycle register High 2
11FH	PWM5H3	Pulse-Width Modulation 5 duty cycle register High 2
120H	RSCTL1	Receive Status Control register
121H	TXSDR1	Transmit Date Register
122H	RXSDR1	Receive Date Reigster
123H	BRCTL1	Baud Rate Control register
124H	TSCTL1	Transmit Control register
125H	EUBRGL1	Enhance Universal Baud Rate Generator register Low
126H	EUBRGH1	Enhance Universal Baud Rate Generator register High
127H	USLPEN1	
128H	SSCICCTL0	Synchronous Serial Communication Interface Control register 0
12AH	SSCICCTL1	Synchronous Serial Communication Interface Control register 0
12BH	SSCISTA	Synchronous Serial Communication Interface Status register
12CH	SSCIBUFR	Synchronous Serial Communication Interface Buffer Register
12EH	SSCIMSK	Synchronous Serial Communication Interface Mask register
12FH	WDTPS	Watchdog Pre-divider Selection register
130H	MULAH	Multiplier A register High
131H	MULAL	Multiplier A register Low
132H	MULBH	Multiplier B register High
133H	MULBL	Multiplier B register Low
134H	MULCTL	Multiplier Control register
135H	MULRES3	Multiplier Result register 3
136H	MULRES2	Multiplier Result register 2
137H	MULRES1	Multiplier Result register 1
138H	MULRES0	Multiplier Result register 0
139H	DIVCTL	Divider Control register
13AH	DIVAH	Divider A register High
13BH	DIVAL	Divider A register Low
13CH	DIVBL	Divider B register Low
13DH	DIVQH	Divider Quotient register High
13EH	DIVQL	Divider Quotient register Low
13FH	DIVRL	Divider Remainder register Low

140H	LEDCTL	Light Emitting Diode Control register
141H	LEDPRE	Light Emitting Diode Control Pre-divider register
142H	LEDDATA0	Light Emitting Diode Data register 0
143H	LEDDATA1	Light Emitting Diode Data register 1
144H	LEDDATA2	Light Emitting Diode Data register 2
145H	LEDDATA3	Light Emitting Diode Data register 3
146H	LEDDATA4	Light Emitting Diode Data register 4
146H	LEDDATA5	Light Emitting Diode Data register 5
147H	LEDDATA6	Light Emitting Diode Data register 6
148H	LEDDATA7	Light Emitting Diode Data register 7
14AH	C1FILTCTL	Comparer 1 Filter Control register
14BH	C1FILTPER	Comparer 1 Filter Per-scale register
14CH	C2FILTCTL	Comparer 2 Filter Control register
14DH	C2FILTPRE	Comparer 2 Filter Per-scale register
14EH	C3FILTCTL	Comparer 3 Filter Control register
14FH	C3FILTPRE	Comparer 3 Filter Per-scale register
155H	LEDDATAE	Light Emitting Diode Data register E
157H	LEDDATAF	Light Emitting Diode Data register F
158H	LEDOMS	
159H	VREFCAL0	Reference Voltage Calibration 0
15AH	VREFCAL1	Reference Voltage Calibration 1
15BH	HLVDCTL	
15CH	LEDLUM	
15DH	T3REL	Timer 3 Reload register Low
15EH	T3REH	Timer 3 Reload register High
15FH	DACM	
160H	T4L	Timer 4 register Low
161H	T4H	Timer 4 register High
162H	T4REL	Timer 4 Reload register Low
163H	T4REH	Timer 4 Reload register High
164H	T4CTL	Timer 4 Control register
165H	RC32KCAL	RC32K Calibration

BANK 2

20CH	P4	Port 4
20DH	P4LR	Port 4 Latch Register
20EH	TR4	Tri Register 4
20FH	PUR4	Pull-up Register 4
210H	P5	Port 5
211H	P5LR	Port 5 Latch Register
212H	TR5	Tri Register 5
213H	PUR5	Pull-up Register 5
214H	P6	Port 6
215H	P6LR	Port 6 Latch Register
216H	TR6	Tri Register 6
218H	PUR6	Pull-up Register 6
219H	P7	Port 7
21AH	P7LR	Port 7 Latch Register
21BH	TR7	Tri Register 7
21CH	PUR7	Pull-up Register 7
21DH	PWMCTL0	Pulse-Width-Modulation Control 0
21EH	PWMCTL1	Pulse-Width-Modulation Control 1
21FH	PP1	Pulse-Width Modulation Periods register 1
220H	PP2	Pulse-Width Modulation Periods register 2

221H	PWM10L	Pulse-Width Modulation 10 Low
222H	PWM10H	Pulse-Width Modulation 10 High
223H	PWM11L	Pulse-Width Modulation 11 Low
224H	PWM11H	Pulse-Width Modulation 11 High
225H	PWM12L	Pulse-Width Modulation 12 Low
226H	PWM12H	Pulse-Width Modulation 12 High
227H	PWM13L	Pulse-Width Modulation 13 Low
228H	PWM13H	Pulse-Width Modulation 13 High
229H	PWM14L	Pulse-Width Modulation 14 Low
22AH	PWM14H	Pulse-Width Modulation 14 High
22BH	PWM15L	Pulse-Width Modulation 15 Low
22CH	PWM15H	Pulse-Width Modulation 15 High
22DH	PWM16L	Pulse-Width Modulation 16 Low
22EH	PWM16H	Pulse-Width Modulation 16 High
22FH	PWM17L	Pulse-Width Modulation 17 Low
230H	PWM17H	Pulse-Width Modulation 17 High
231H	PWM18L	Pulse-Width Modulation 18 Low
232H	PWM18H	Pulse-Width Modulation 18 High
233H	PWM19L	Pulse-Width Modulation 19 Low
234H	PWM19H	Pulse-Width Modulation 19 High
235H	PWM1AL	Pulse-Width Modulation 1A Low
236H	LCDPS	Liquid Crystal Display
237H	LCDCTL	Liquid Crystal Display Control register
238H	LCDPDR	
239H	LCDFLKCTL	
23AH	LCDFLKON	
23BH	LCDFLKOFF	
23CH	LCDSE0	Liquid Crystal Display Segment enable register 0
23DH	LCDSE1	Liquid Crystal Display Segment enable register 1
23EH	LCDSE2	Liquid Crystal Display Segment enable register 2
23FH	LCDSE3	Liquid Crystal Display Segment enable register 3
240H	LCDDATA0	Liquid Crystal Display Data register 0
241H	LCDDATA1	Liquid Crystal Display Data register 1
242H	LCDDATA2	Liquid Crystal Display Data register 2
243H	LCDDATA3	Liquid Crystal Display Data register 3
244H	LCDDATA4	Liquid Crystal Display Data register 4
245H	LCDDATA5	Liquid Crystal Display Data register 5
246H	LCDDATA6	Liquid Crystal Display Data register 6
247H	LCDDATA7	Liquid Crystal Display Data register 7
248H	LCDDATA8	Liquid Crystal Display Data register 8
249H	LCDDATA9	Liquid Crystal Display Data register 9
24AH	LCDDATA10	Liquid Crystal Display Data register 10
24BH	LCDDATA11	Liquid Crystal Display Data register 11
24CH	LCDDATA12	Liquid Crystal Display Data register 12
24DH	LCDDATA13	Liquid Crystal Display Data register 13
24EH	LCDDATA14	Liquid Crystal Display Data register 14
24FH	LCDDATA15	Liquid Crystal Display Data register 15
250H	LCDDATA16	Liquid Crystal Display Data register 16
251H	LCDDATA17	Liquid Crystal Display Data register 17
252H	LCDDATA18	Liquid Crystal Display Data register 18
253H	LCDDATA19	Liquid Crystal Display Data register 19
254H	LCDDATA20	Liquid Crystal Display Data register 20
255H	LCDDATA21	Liquid Crystal Display Data register 21
256H	LCDDATA22	Liquid Crystal Display Data register 22

257H	LCDDATA23	Liquid Crystal Display Data register 23
258H	LCDDATA24	Liquid Crystal Display Data register 24
259H	LCDDATA25	Liquid Crystal Display Data register 25
25AH	LCDDATA26	Liquid Crystal Display Data register 26
25BH	LCDDATA27	Liquid Crystal Display Data register 27
25CH	LCDDATA28	Liquid Crystal Display Data register 28
25DH	LCDDATA29	Liquid Crystal Display Data register 29
25EH	LCDDATA30	Liquid Crystal Display Data register 30
25FH	LCDDATA31	Liquid Crystal Display Data register 31
260H	PWM1AH	Pulse-Width Modulation 1A High
261H	PWM1BL	Pulse-Width Modulation 1B Low
262H	PWM1BH	Pulse-Width Modulation 1B High
263H	PWM1CL	Pulse-Width Modulation 1C Low
264H	PWM1CH	Pulse-Width Modulation 1C High
265H	PWM1DL	Pulse-Width Modulation 1D Low
266H	PWM1DH	Pulse-Width Modulation 1D High
267H	PWM1EL	Pulse-Width Modulation 1E Low
268H	PWM1EH	Pulse-Width Modulation 1E High
269H	PWM1FL	Pulse-Width Modulation 1F Low
26AH	PWM1FH	Pulse-Width Modulation 1F High
BANK3		
305H	EIE4	Enable Interrupt register 4
306H	EIF4	Enable Interrupt Flag register 4
307H	IP4	Interrupt Priority 4 register
320H	U7816CTL2	Uart 7816 Control register 2
321H	U7816RXCTL2	Uart 7816 Receive Control register 2
322H	U7816TXCTL2	Uart 7816 Transmit Control register 2
323H	CLKDIV2	
324H	EGTCTL2	
325H	TZBRGH2	
326H	TZBRGL2	
327H	UARTMTH2	
328H	USLPEN2	
329H	BRCTL2	Baud Rate Control register 2
32AH	RSCTL2	Receive Status Control register 2
32BH	RXSDR2	Receive Date Register2
32CH	EUBRGH2	Enhance Universal Baud Rate Generator register High2
32DH	EUBRGL2	Enhance Universal Baud Rate Generator register Low 2
32EH	TXSDR2	Transmit Date Register 2
32FH	TSCTL2	Transmit Control register 2

附录 4 全双工异步模式的典型波特率和误差值

SYNC=0,HBRG=0,BRG16=0									
波特率 时钟频率		300	1200	2400	9600	10417	19.2K	57.6K	116.2K
20.000MHz	实际波特率	-	-	2404	9615	10417	-	-	-
	误差(%)	-	-	0.16	0.16	0.00	-	-	-
	EUBRGL 值(十六进制)	-	-	81H	20H	1DH	-	-	-
16.000MHz	实际波特率	-	1202	2404	9615	10417	19231	-	-
	误差(%)	-	0.16	0.16	0.16	0.00	0.16	-	-
	EUBRGL 值(十六进制)	-	CFH	67H	19H	17H	0CH	-	-
8.000MHz	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	-	-	-
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	-	-	-	-	-	-
512.00 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 时钟频率		300	1200	2400	9600	10417	19.2K	57.6K	116.2K
20.000MHz	实际波特率	-	-	-	9541	10417	19230	56818	113.6K
	误差(%)	-	-	-	-0.6	0.00	0.16	-1.35	0.03
	EUBRGL 值(十六进制)	-	-	-	82H	77H	41H	15H	0AH
16.000MHz	实际波特率	-	-	-	9615	10417	19231	58824	111.1K
	误差(%)	-	-	-	0.16	0.00	0.16	2.12	3.55
	EUBRGL 值(十六进制)	-	-	-	67H	5FH	33H	10H	08H

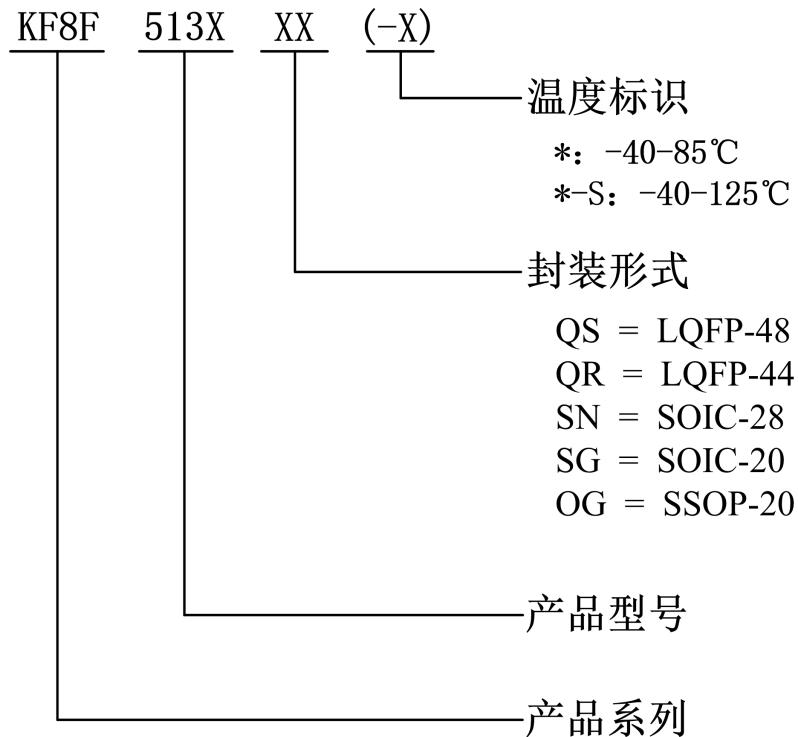
8.000MHz	实际波特率	—	—	2404	9615	10417	19231	55556	—
	误差(%)	—	—	0.16	0.16	0. 00	0.16	-3.55	—
	EUBRGL 值(十六进制)	—	—	CFH	33H	2FH	19H	08H	—
4.000MHz	实际波特率	—	1202	2404	9615	10417	19.23K	—	—
	误差(%)	—	0.16	0.16	0.16	0. 00	0.16	—	—
	EUBRGL 值(十六进制)	—	CFH	67H	19H	17H	0CH	—	—
2.000MHz	实际波特率	—	1202	2404	9615	10417	—	—	—
	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	—	—	—
512.000KHz	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 时钟频率		300	1200	2400	9600	10417	19.2K	57.6K	116.2K
20.000MHz	实际波特率	-	-	-	9541	10417	19230	56818	113.6K
	误差(%)	-	-	-	-0.6	0.00	0.16	-1.35	0.03
	EUBRGL 值 (十六进制)	-	-	-	82H	77H	41H	15H	0AH
16.000MHz	实际波特率	—	—	—	9615	10417	19231	58824	111.1K
	误差(%)	—	—	—	0.16	0.00	0.16	2.12	3.55
	EUBRGL 值 (十六进制)	—	—	—	67H	5FH	33H	10H	08H
8.000MHz	实际波特率	299.9	1999	2404	9615	10417	19231	55556	—
	误差(%)	-0.02	-0.08	0.16	0.16	0.00	0.16	-3.55	—
	EUBRGL 值 (十六进制)	0682H	01A0H	CFH	33H	3FH	19H	08H	—
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	—	—	—
512.000KHz	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 时钟频率		300	1200	2400	9600	10417	19.2K	57.6K	116.2K
20.000MHz	实际波特率	300	1200	2399	9597	10417	19157	57.47K	113.6K
	误差(%)	0.00	0.00	0.03	0.03	0.00	0.22	0.22	0.03
	EUBRGH: EUBRGL	411AH	1046H	0823H	0208H	01DFH	0104H	0056H	002BH
16.000MHz	实际波特率	300	1200	2401	9615	10417	19.23K	57971	117.6K
	误差(%)	0.00	0.01	0.04	0.16	0.00	0.16	0.64	2.12
	EUBRGH: EUBRGL	3414H	0D04H	0681H	019FH	017FH	00CFH	0044H	0021H
8.000MHz	实际波特率	300.0	1200	2401	9615	10417	19.23K	57.14K	117.6K
	误差(%)	0.00	-0.02	0.04	0.16	0.00	0.16	-0.79	2.12
	EUBRGH: EUBRGL	1A0A H	0682H	0340H	00CFH	00BFH	0067H	0022H	0010H
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	—	—
512.000KHz	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

产品标识体系



产品系列: KF8F = KF 系列 8 位单片机

产品型号: 513x = 513x 型

封装形式: QS = LQFP-48
QR = LQFP-44
SN = SOIC-28
SG = SOIC-20
OG = SSOP-20

温度标识: *: -40~85°C
*-S: -40~125°C

例: 订货号 KF8F5134QS 为 KF8F5134 型号 LQFP-48 封装, 温度范围-40~85°C;
订货号 KF8F5134QR-S 为 KF8F5134 型号 LQFP-44 封装, 温度范围-40~125°C。

版本信息

数据手册版本号	更新内容概述	更新日期
V0	初版	2018-10-8
V0.1	增加 F5133-TSSOP28 封装, 修改寄存器错误信息	2019-01-08
V0.2	更新 ADCCTL2 寄存器说明中 ADC 通道顺序错误; 添加 T3BUZOE 寄存器配置信息; 修改 T4BUZOE 说明错误;	2019-01-18
V0.3	增加 F5133-SSOP20 封装; 修改 ADC 章节描述错误; 增加 T3 和 T4 的蜂鸣器输出功能说明; 修改 LED 章节描述错误, 更新 LED 通道章节内容。	2019-02-27
V0.4		
V0.5	修改 LED 章节 COMEN 位描述错误。	2019-03-07
V0.6	更新 SSOP20 封装尺寸信息。 更新产品标识体系章节。	2019-07-22
V0.7	更新电气特性数据; 添加休眠模式章节。	2019-09-10
V0.8	更新产品订购信息。	2020-08-22
V0.9	更新 PWM3 周期公式	2020-11-9

ROSH 认证

本产品已通过 ROSH 检测。

声明及销售网络

销售及服务网点

上海 TEL:021-50275927

地址 上海浦东张江集电港龙东大道 3000 号 1 幢 906 室 B1 座