

8 位微控制器

KF8L10Z08 数据手册

产品订购信息

芯片型号	订货号	封装	FLASH	RAM (Byte)	DEE (Byte)	内部 HFOSC	外部 HF/LFOSC	8 位 定时器	16 位 定时器	16 位 PWM	12 位 ADC	CMP	SPI/I2C	RTC	工作电压 (V)	内核 版本
KF8L10	KF8L10Z08OG-I	SSOP-20	4Kx16	400	128	4M	8M/32.768k	1	3	2	16	2	1	Y	2.1~5.5	V1

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。使用方同意在由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。

KF8L10Z08 芯片使用注意事项

芯片的 ESD 防护措施

KF8L10Z08 芯片提供满足工业级 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

KF8L10Z08 芯片提供满足工业级 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

KF8L10Z08 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

KF8L10Z08 芯片提供独立电源管脚。当 KF8L10Z08 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

KF8L10Z08 芯片提供内部上电复位。对于不同的快速上电/断电或慢速上电/断电系统，内部上电复位电路可能失效，建议用户使用外部复位、断电复位、看门狗复位等，确保复位电路正常工作。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

KF8L10Z08 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

KF8L10Z08 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

KF8L10Z08 芯片提供宽范围的输入管脚电平，用户输入高电平应大于 V_{IH} 的最小值，低电平应小于 V_{IL} 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

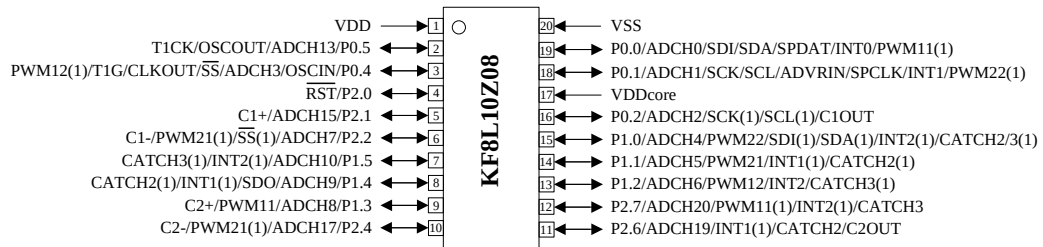
KF8L10Z08 芯片提供低功耗设计模式，用户在实际应用中可根据应用系统的要求采用各种不同的低功耗模式，包括系统工作时钟的选择和休眠模式的选择等等。

芯片的开发环境

KF8L10Z08 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

KF8L10Z08 (SSOP-20)引脚示意图:



注：（1）标志为引脚复用功能。

引脚总述

I/O	引脚名	AD	CMP	PWM	捕捉	SSCI	定时器	中断	上拉	基本
	SSOP-20									
-	1	-	-	-	-	-	-	-	-	VDD
P0.5	2	ADCH13	-	-	-	-	T1CK	IOCL5	Y	OSCOUT
P0.4	3	ADCH3	-	PWM12(1)	-	$\overline{SS}$	T1G	IOCL4	Y	CLKOUT / OSCIN
P2.0	4	-	-	-	-	-	-	-	Y	RST
P2.1	5	ADCH15	C1+	-	-	-	-	-	Y	-
P2.2	6	ADCH7	C1-	PWM21(1)	-	$\overline{SS}(1)$	-	-	Y	-
P1.5	7	ADCH10	-	-	CATCH3(1)	-	-	INT2(1)	Y	-
P1.4	8	ADCH9	-	-	CATCH2(1)	SDO	-	INT1(1)	Y	-
P1.3	9	ADCH8	C2+	PWM11	-	-	-	-	Y	-
P2.4	10	ADCH17	C2-	PWM21(1)	-	-	-	-	Y	-
P2.6	11	ADCH19	C2OUT	-	CATCH2	INT1(1)	-	-	Y	-
P2.7	12	ADCH20	-	PWM11(1)	CATCH3	INT2(1)	-	-	Y	-
P1.2	13	ADCH6	-	PWM12	CATCH3(1)	-	-	INT2	Y	-
P1.1	14	ADCH5	-	PWM21	CATCH2(1)	-	-	INT1(1)	Y	-
P1.0	15	ADCH4	-	PWM22	CATCH2/3(1)	SDI/SDA(1)	-	INT2(1)	Y	-
P0.2	16	ADCH2	C1OUT	-	-	SCK/SCL(1)	-	IOCL2	Y	-
-	17	-	-	-	-	-	-	-	-	VDDcore
P0.1	18	ADCH1	-	PWM22(1)	-	SCK/SCL	-	IOCL1 / INT1	Y	ADVRIN / SPCLK
P0.0	19	ADCH0	-	PWM11(1)	-	SDI/SDA	-	IOCL0 / INT0	Y	SPDAT
-	20	-	-	-	-	-	-	-	-	VSS

目 录

产品订购信息.....	2
KF8L10Z08 芯片使用注意事项.....	3
引脚示意图.....	5
引脚总述.....	6
目 录.....	7
1 系统概述.....	14
1.1 芯片特征	15
1.2 系统框图	16
1.3 存储器	17
1.4 系统时钟	18
1.4.1 时钟模块相关寄存器	19
1.4.1.1 系统频率控制寄存器 OSCCTL	19
1.4.1.2 系统时钟标志寄存器 OSCSTA	20
1.4.2 上电延时	21
1.4.3 内部高频振荡器	21
1.4.4 内部低频振荡器	22
1.4.5 外部高频振荡器	22
1.4.6 外部低频振荡器	22
1.4.7 时钟切换和时钟信号同步	23
1.4.8 双速启动模式	24
1.4.9 双速启动过程	24
1.4.10 外部时钟故障检测	24
1.4.11 时钟故障保护处理	25
1.5 配置位	26
1.6 在线串行编程	28
2 I/O 端口介绍.....	29
2.1 I/O 端口的读写	30
2.2 P0 口	31
2.2.1 P0 口相关的寄存器	32
2.2.1.1 P0 口状态读取寄存器 (P0)	32
2.2.1.2 P0 口输出锁存寄存器 (P0LR)	32
2.2.1.3 P0 口方向控制寄存器 (TR0)	33
2.2.1.4 P0 口上拉功能控制寄存器(PUR0).....	33
2.2.1.5 P0 口电平变化中断控制寄存(IOCL).....	34
2.2.1.6 P0 口模拟/数字口设置寄存器 (ANSE0)	34
2.2.2 P0 口各引脚内部原理功能框图	35
2.3 P1 口	36
2.3.1 P1 口相关的寄存器	37

2.3.1.1	P1 口状态读取寄存器 (P1)	37
2.3.1.2	P1 口输出锁存寄存器 (P1LR)	37
2.3.1.3	P1 口方向控制寄存器(TR1).....	38
2.3.1.4	P1 口上拉功能控制寄存器(PUR1).....	38
2.3.1.5	P1 口模拟/数字口设置寄存器 (ANSE1)	38
2.3.2	P1 口原理功能框图.....	39
2.4	P2 口	40
2.4.1	P2 口相关的寄存器.....	41
2.4.1.1	P2 口状态寄存器 (P2)	41
2.4.1.2	P2 口输出锁存寄存器 (P2LR)	41
2.4.1.3	P2 口方向控制寄存器 (TR2)	42
2.4.1.4	P2 口上拉控制寄存器 (PUR2)	42
2.4.1.5	P2 口模拟/数字口设置寄存器 (ANSE2)	42
2.4.2	P2 口原理功能框图.....	43
2.5	I/O 口功能复用	44
2.5.1	引脚复用功能控制寄存器 0 (APFCTL0)	44
2.5.2	引脚复用功能控制寄存器 1 (APFCTL1)	45
2.5.3	引脚复用功能控制寄存器 2 (APFCTL2)	46
3	存储器.....	47
3.1	程序存储器(ROM)区.....	47
3.1.1	JMP、CALL 指令	48
3.2	数据存储器(RAM)区.....	49
3.2.1	通用寄存器区.....	49
3.2.2	特殊功能寄存器(SFR)区	50
3.3	DATAEEPROM	51
3.3.1	寄存器 NVMDATAL	51
3.3.2	寄存器 NVMADDRL	51
3.3.3	寄存器 NVMCTL0/NVMCTL1	51
3.3.4	写 DATA EEPROM.....	51
3.3.5	读 DATA EEPROM.....	53
3.4	寄存器组 R _N	53
3.5	ID 地址单元	53
4	汇编指令及寻址方式.....	54
4.1	寻址方式	54
4.1.1	寄存器寻址.....	54
4.1.2	直接寻址.....	54
4.1.3	立即数寻址.....	54
4.1.4	寄存器间接寻址	55
4.1.5	位寻址.....	55
4.2	汇编指令	55
5	中断.....	56

5.1	中断相关的寄存器	58
5.1.1	中断控制寄存器(INTCTL)	58
5.1.2	中断使能寄存器 EIE1	59
5.1.3	中断使能寄存器 EIE2	60
5.1.4	中断标志寄存器 EIF1	60
5.1.5	中断标志寄存器 EIF2	61
5.1.6	中断优先级控制寄存器 IP0	62
5.1.7	中断优先级控制寄存器 IP1	62
5.1.8	中断优先级控制寄存器 IP2	63
5.1.9	电源控制寄存器 PCTL	63
5.1.10	中断响应	64
5.2	INT 中断	65
5.2.1	INT 中断对应引脚	65
5.2.2	INT0 中断	65
5.2.3	INT1 中断	66
5.2.4	INT2 中断	66
5.3	定时器中断	66
5.4	P0 口中断	66
5.5	PWM 中断	66
5.6	中断现场保护	67
6	定时/计数器	68
6.1	定时器 0(T0)	68
6.1.1	T0 原理框图	68
6.1.2	T0 相关的寄存器	69
6.1.3	OPTR 选择寄存器	69
6.1.4	T0 的使用	70
6.2	定时/计数器 1(T1)	71
6.2.1	T1 原理框图	71
6.2.2	T1 相关的寄存器	71
6.2.2.1	T1 控制寄存器	72
6.2.3	定时模式	72
6.2.4	计数模式	73
6.2.5	T1 在休眠模式下的运行	73
6.2.6	T1 分配给 PWM1/2x	73
6.3	定时/计数器 2(T2)	74
6.3.1	T2 原理框图	74
6.3.2	T2 相关寄存器	74
6.3.2.1	T2 控制寄存器	75
6.3.2.3	捕捉数据清零寄存器	76
6.3.3	T2 预分频器	77
6.3.4	T2 计数时钟选择	77
6.3.5	T2 重载功能	77
6.3.6	T2 捕捉功能	77

6.3.7	T2 中断.....	77
6.3.8	T2 工作在休眠模式.....	77
6.4	定时/计数器 3(T3).....	78
6.4.1	T3 原理框图.....	78
6.4.2	T3 相关寄存器.....	78
6.4.1.1	T3 控制寄存器.....	79
6.4.1.2	T3 控制寄存器 1.....	79
6.4.3	T3 预分频器.....	80
6.4.4	T3 计数时钟选择.....	80
6.4.5	T3 计数寄存器.....	80
6.4.6	T3 捕捉功能.....	80
6.4.7	T3 中断.....	80
6.4.8	T3 工作在休眠模式.....	80
7	模数(A/D)转换模块.....	81
7.1	AD 相关的寄存器.....	81
7.1.1	AD 控制寄存器 0(ADCCTL0).....	82
7.1.2	AD 控制寄存器 1(ADCCTL1).....	83
7.2	通道的选择.....	84
7.3	模拟输入口的配置.....	84
7.4	A/D 转换参考电压的选择.....	84
7.5	转换时钟的选择.....	84
7.6	输出格式.....	84
7.7	A/D 转换的启动和完成.....	85
7.8	复位的影响.....	85
7.9	AD 转换器的设置.....	85
8	PWM1X/2X 模块.....	86
8.1	PWM 原理.....	86
8.2	PWM1x/2x 相关的寄存器.....	87
8.2.1	PWM1x/2x 控制寄存器.....	87
8.1	PWM1x/2x 输出引脚.....	88
8.2	PWM1x/2x 周期.....	88
8.3	PWM1x/2x 占空比.....	88
8.4	PWM1x/2x 分辨率.....	89
8.5	PWM1x/2x 中断.....	89
8.6	休眠模式下的操作.....	89
8.7	复位的影响.....	89
8.8	PWM1/2 使用方法.....	90
9	模拟比较器模块.....	91
9.1	模拟比较器原理.....	91
9.2	相关寄存器.....	91
9.3	CMP1 相关寄存器.....	92

9.3.1	比较器 1 控制寄存器 C1CTL	92
9.3.2	滤波器 1 控制寄存器 C1FILTCTL	92
9.3.3	滤波器 1 采样时钟分频寄存器 C1FILTPRE	93
9.4	CMP2 相关寄存器	93
9.4.1	比较器 2 控制寄存器 C2CTL	93
9.4.2	滤波器控制寄存器 C2FILTCTL	94
9.4.3	滤波器 2 采样时钟分频寄存器 C2FILTPRE	94
9.5	极性选择	95
9.6	范围控制功能	95
9.7	滤波功能	95
9.8	比较器中断	96
9.9	复位的影响	96
10	SSCI 模块	97
10.1	概述	97
10.1.1	SSCI 模块功能引脚	97
10.2	SSCI 相关寄存器	98
10.2.1	SSCI 控制寄存器 0 (SSCICTL0)	98
10.2.2	SSCI 控制寄存器 1 (SSCICTL1)	100
10.2.3	SSCI 状态寄存器 (SSCISTA)	101
10.2.4	SSCI 屏蔽寄存器 (SSCIMSK)	102
10.2.5	SSCI I2C 地址寄存器 (SSCIADD)	103
10.3	I2C 模式	104
10.3.1	工作原理	104
10.3.2	I2C 从动模式	105
10.3.2.1	寻址	105
10.3.2.2	接收	106
10.3.2.3	发送	107
10.3.2.4	广播呼叫地址支持	109
10.3.3	I2C 主控模式	109
10.3.3.1	主控模式支持	110
10.3.3.2	I2C 主模式操作	111
10.3.3.3	波特率发生器	112
10.3.3.4	I2C 主控模式启动条件时序	112
10.3.3.5	I2C 主控模式重复启动条件时序	113
10.3.3.6	I2C 主控模式发送	114
10.3.3.7	I2C 主控模式接收	116
10.3.3.8	应答序列时序	117
10.3.3.9	停止条件序列	118
10.3.3.10	时钟仲裁	119
10.3.4	多主控器模式	119
10.3.4.1	多主机通信, 总线冲突与总线仲裁	120
10.3.4.2	启动条件期间的总线冲突	120
10.3.4.3	重复启动条件期间的总线冲突	122

10.3.4.4	停止条件期间的总线冲突.....	124
10.3.4.5	SSCI 屏蔽寄存器.....	125
10.4	SPI 模式.....	125
10.4.1	工作原理.....	126
10.4.2	使能 SPI/IO 与外部链接.....	127
10.4.3	典型连接.....	127
10.4.4	主模式.....	128
10.4.5	从模式.....	129
10.4.5.1	从动模式.....	129
10.4.5.2	从动选择同步.....	129
10.4.6	休眠模式和复位.....	130
10.4.7	SPI 四种工作模式设置.....	131
10.4.7.1	主控发送工作流程.....	131
10.4.7.2	从动接收工作流程.....	131
10.4.7.3	主控接收工作流程.....	131
10.4.7.4	从动发送工作流程.....	132
11	复位.....	133
11.1	上电复位(POR).....	133
11.2	WDT 复位.....	133
11.3	RST 复位.....	133
11.4	欠压检测复位(LVR/LPLVR).....	134
11.5	上电延时定时器.....	134
11.6	不同复位条件下对寄存器的影响.....	135
12	功耗模式.....	138
12.1	相关寄存器.....	139
12.2	运行状态下：正常运行模式.....	139
12.3	休眠状态下：深度休眠模式和空闲模式.....	139
12.4	深度休眠模式下可工作外设.....	140
12.5	休眠状态下 I/O 口注意事项.....	141
12.6	休眠状态唤醒方式.....	141
13	看门狗定时器.....	142
13.1	看门狗相关寄存器.....	142
13.1.1	看门狗预分频选择寄存器 WDTPS.....	142
13.2	看门狗的开启方式.....	143
13.3	看门狗的清除方式.....	143
13.4	看门狗的周期.....	143
14	电气规范.....	145
14.1	极限参数值.....	145
14.2	时钟温漂典型特性图.....	146
14.3	静态电流特性.....	147
14.4	功耗模式直流特性.....	148

14.5	I/O 端口电平和芯片供电电压特性	149
14.6	A/D 转换器 (ADC) 特性.....	150
14.7	光电检测器电气特性	150
14.8	振荡器模块规范	151
14.9	内部整流器	151
14.10	内部低频振荡器	151
14.11	外部低频振荡器	151
14.12	外部高频晶振	152
14.13	DATA EEPROM	152
15	直流特性图表	153
16	封装信息	159
17	附录 1 KF8L10Z08 SFR 地址映射及功能汇总	160
18	附录 2 汇编指令集	162
19	产品标识体系	164
20	版本变更记录表	165
21	ROSH 认证	166
22	声明及销售网络	167

1 系统概述

KF8L10Z08 为哈佛结构的精简指令 CPU。在这种结构中，程序和数据总线是相互独立的。指令字节长度为 16 位，大多数指令能在一个机器周期内执行完成。一共有 68 条指令，效率高，容易进行指令扩展。

芯片内集成了多种外设，包括：

1 个 8 位定时器 T0；

1 个 16 位定时器/计数器 T1；

1 个 16 位的定时/计数器 T2；

1 个 16 位的定时 T3；

4 路 8 位 PWM 模块；

1 个 12 位 ADC 模块；

2 个比较器模块；

1 个 SSCI(SPI/I2C)模块；

硬件看门狗和低电压检测及低电压复位模块等；

芯片内集成了 $400(384+16) \times 8$ 位的数据存储器 RAM、8K byte 的程序存储器。

1.1 芯片特征

● CPU

高性能哈佛结构的 RISC CPU

68 条精简指令

支持中断优先级处理

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）

工作频率为 15.625kHz~4MHz（内部高频时钟作为系统时钟源时）

● 存储器

8K byte FLASH 程序存储器

400×8 位的数据存储器

128×8 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 4MHz±1%（常温）

内部可校正低频 32kHz 时钟

支持在线串行编程，低功耗休眠模式

● I/O 口配置

输入输出：双向输入输出

内置上拉功能：P0/P1/P2 口带有弱上拉功能

电平变化中断：P0 口均有电平变化中断功能

● 定时器/计数器

定时器 0：带有 8 位预分频器的 8 位定时器/计数器

定时器 1：带门控、重载功能和预分频器的 16 位定时器/计数器

定时器 2：带有重载功能、预分频及时钟源多选的 16 位定时器/计数器

定时器 3：带有预分频及时钟源多选的 16 位定时器/计数器

● 其它外设

1 个 12 位 ADC 模块

4 路 8 位脉宽调制 PWM 模块

1 个 I2C/SPI 模块

1 个比较器模块

● 工作条件

工作电压：2.1V~5.5V

工作温度范围：-40~85℃

1.2 系统框图

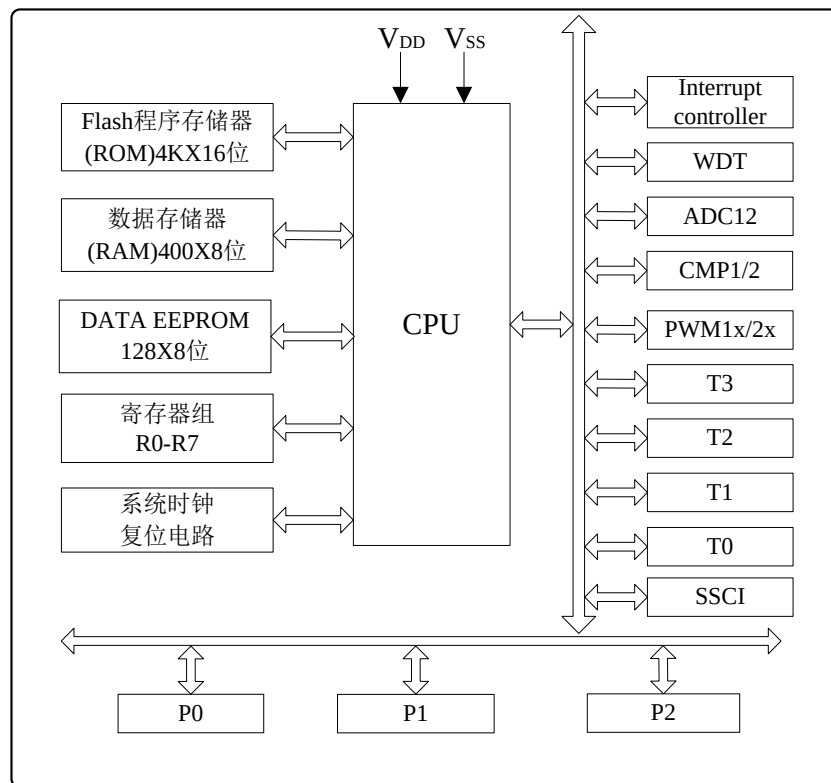


图 1.1 KF8L10Z08 系统框图

1.3 存储器

KF8L10Z08 单片机的存储器包含：程序存储器(ROM)、数据存储器(RAM)和 DATA EEPROM。

KF8L10Z08 的程序存储器空间为 8K byte，寻址范围为 0000H~0FFFH，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 2，每个区均有 128×8 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-7FH。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时钟周期，如图 1.2 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。

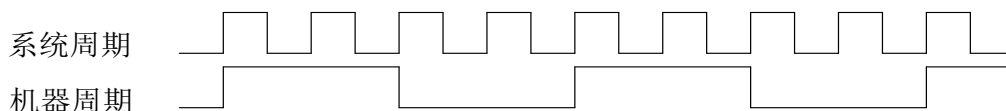


图 1.2 机器周期

KF8L10Z08 单片机提供 4 个可选时钟源：

内部高频时钟 INTHF：以内部高频振荡器为时钟源；

内部低频时钟 INTLF：以内部低频振荡器为时钟源；

外部高频时钟 EXTHF：标准晶振、陶瓷谐振器或外接最高 8MHz 的时钟源工作；

外部低频时钟 EXTLF：外接 32.768kHz 的钟表晶振。

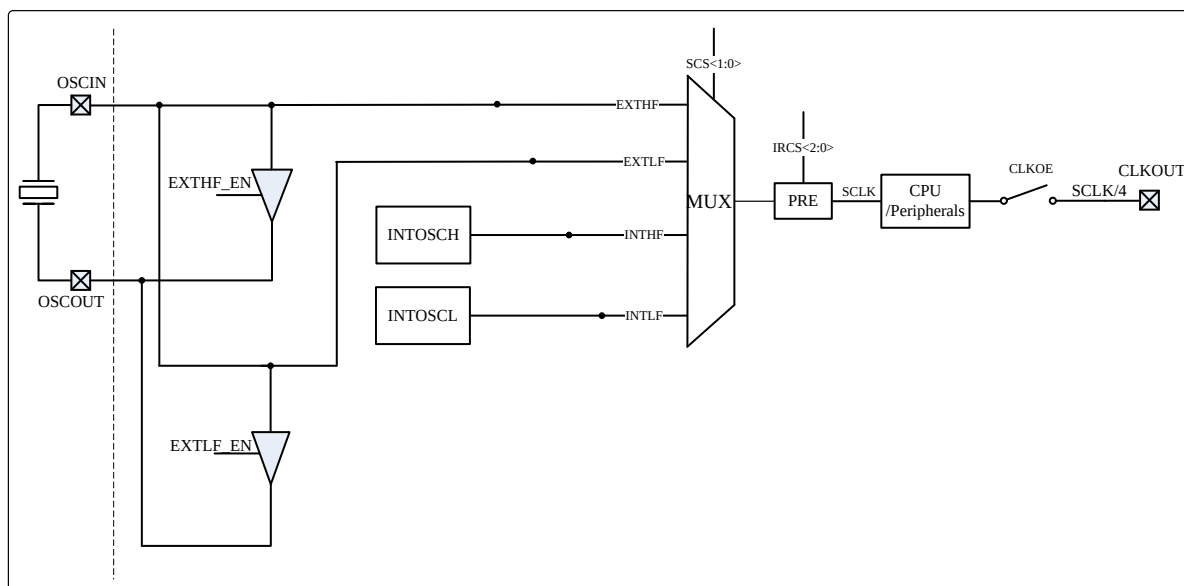


图 1.3 时钟模块原理框图

KF8L10Z08 单片机的系统时钟源可以配置为：

- 内部高频时钟 INTHF
- 内部低频时钟 INTLF
- 外部高频时钟 EXTHF
- 外部低频时钟 EXTLF

注：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
INTOSCH(INTHF)	定义为内部高频振荡器（时钟）
INTOSCL(INTLF)	定义为内部低频振荡器（时钟）
EXTHF	定义为外部高频时钟
EXTLF	定义为外部低频时钟
OSC	定义为 INTHF, INTLF, EXTHF 和 EXTLF 的集合

注：系统时钟和系统时钟源为两个概念，系统时钟定义为选定时钟源并经过分频器（IRCS<2:0>位控制）分频后用于系统工作的时钟；系统时钟源定义为系统时钟所选时钟源。

1.4.1 时钟模块相关寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
2FH	OSCCTL	CLKOE	IRCS2	IRCS1	IRCS0	SCS1	SCS0	IESO	FSCM
63H	OSCSTA	-	OSTS	HTS	LTS	-	-	SCF1	SCF0

1.4.1.1 系统频率控制寄存器 OSCCTL

寄存器1.1: OSCCTL系统频率控制寄存器(地址:2FH)

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

CLKOE: 系统时钟输出使能位
1=使能系统时钟输出
0=禁止系统时钟输出

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

图注: R=可读 W=可写 -=未用 U=未实现位

- 注 1: 在使用外部时钟且经常进入休眠模式的应用中, 双速模式使唤醒所花费的时间中不包含振荡器的起振时间, 能有效降低器件的总功耗。
- 2: 在使能故障检测功能后, 外部时钟发生故障后会自动切换到内部高频时钟, 详见1.4.11

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:63H)

	bit7						bit0	
复位值 0110--00	-	OSTS	HTS	LTS	-	-	SCF1	SCF0
	R	R	R	R	U	U	R/W	R/W

- 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>: 系统时钟标志位
00=当前系统时钟为内部高频时钟
01=当前系统时钟为内部低频时钟
10=当前系统时钟为外部低频时钟
11=当前系统时钟为外部高频时钟

图注: R=可读 W=可写 -=未用 U=未实现位

1.4.2 上电延时

KF8L10Z08 单片机的上电延时可以通过配置位 $\overline{\text{PWRT}}$ 设置，上电延时计数时钟 PWRTCLK 为内部低频时钟。

当 $\overline{\text{PWRT}}=1$ 时，上电延时 $T_{\text{pwrt}} = 2^6 \times T_{\text{pwrtclk}}$ ，约 2ms；

当 $\overline{\text{PWRT}}=0$ 时，上电延时 $T_{\text{pwrt}} = 2^{10} \times T_{\text{pwrtclk}}$ ，约 32ms。

注： T_{pwrtclk} =上电延时计数时钟周期。

1.4.3 内部高频振荡器

KF8L10Z08 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 4MHz，精度为 $\pm 1\%$ 。

OSCCAL0 、 OSCCAL1 和 OSCCAL2 为 INTOSCH 的校准寄存器，用来存放 INTHF 校准值。用户在编程时，需要在程序初始化部分将存放在程序空间 0FFFH 、 0FFEh 和 0FFDh 的校准值读出来分别存放到 OSCCAL0 、 OSCCAL1 和 OSCCAL2 中，否则会导致系统时钟不准。程序示例请参考例 1.1。

例1.1 内部高频振荡器及内部参考电压校准

```
CALL 0X0FFB ;读取内部参考电压校准值2
MOV VRECAL2, R0
NOPZ
NOPZ
CALL 0X0FFC ;读取内部参考电压校准值1
MOV VRECAL1, R0
NOPZ
NOPZ
CALL 0X0FFD ;读取内部高频振荡器校准值2
MOV OSCCAL2, R0
NOPZ
NOPZ
CALL 0X0FFE ;读取内部高频振荡器校准值1
MOV OSCCAL1, R0
NOPZ
NOPZ
CALL 0X0FFF ;读取内部高频振荡器校准值0
MOV OSCCAL0, R0
NOPZ
NOPZ
```

注：如用户使用汇编语言进行编程，需要对内部高频/低频晶振进行调用校准值操作；使用 C 语言编程，则无需再对内部高频/低频晶振进行调用校准值操作，编译器已自动调用。

1.4.4 内部低频振荡器

系统内部低频振荡器，其振荡器频率为 32kHz，它不仅可以为 SCLK 提供时钟源，还可以单独作为低频时钟信号供外设模块使用，同时可以作为内部上电延时定时器和看门狗 WDT 定时器的时钟。

RC32KCAL(6CH)为 INTLF 时钟(INTOSCL)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 FFAH 的晶振校准值存放到 RC32KCAL 中。程序示例请参考例 1.2。

例1.2 内部低频振荡器校准

```
CALL 0X0FFA ;读取内部低频振荡器校准值
MOV RC32KCAL, R0 ;校准值送RC32KCAL
```

注：如用户使用汇编语言进行编程，需要对内部高频/低频晶振进行调用校准值操作；使用 C 语言编程，则无需再对内部高频/低频晶振进行调用校准值操作，编译器已自动调用。

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、陶瓷谐振器或外接最高 8M 的时钟为外部高频时钟源。外部高频时钟 EXTHF 可以作为系统时钟 SCLK 和计数器时钟源。

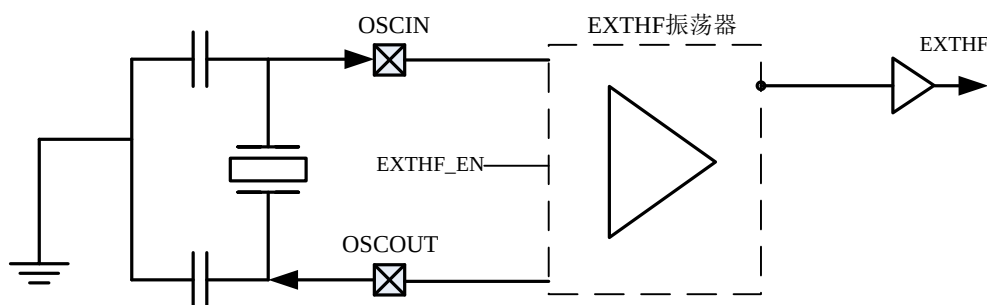


图 1.4 EXTHF 振荡器原理图

1.4.6 外部低频振荡器

如图 1.5 所示，引脚 OSCIN 和引脚 OSCOUT 外接 32.768kHz 石英晶振。EXTLF 时钟源可以作为系统时钟 SCLK 和其他外设时钟源。

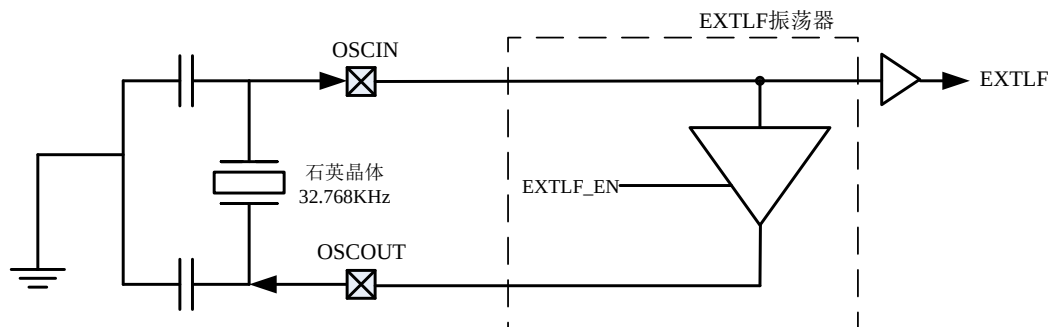


图 1.5 EXTFLF 振荡器原理图

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，即可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 EXTFLF），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。

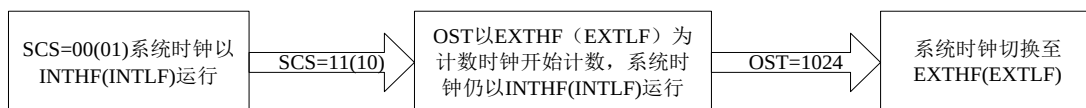


图 1.6 时钟切换流程图

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。

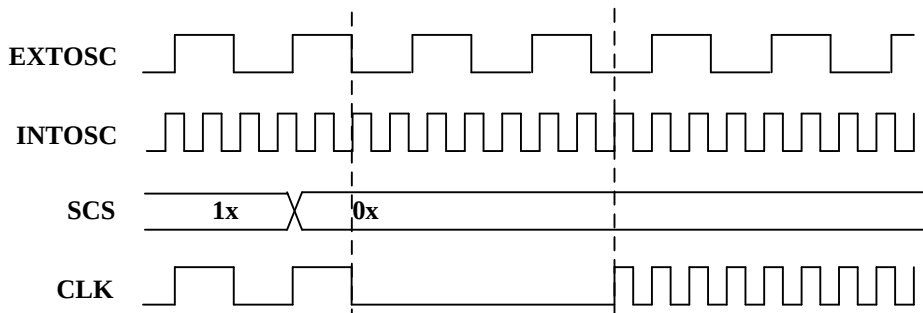


图 1.7 切换至内部时钟源时时钟信号同步时序图

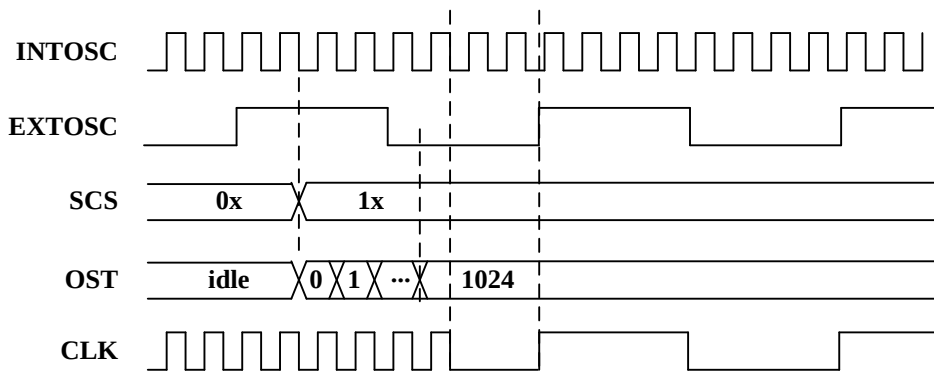


图 1.8 切换至外部时钟源时时钟信号同步时序图

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。

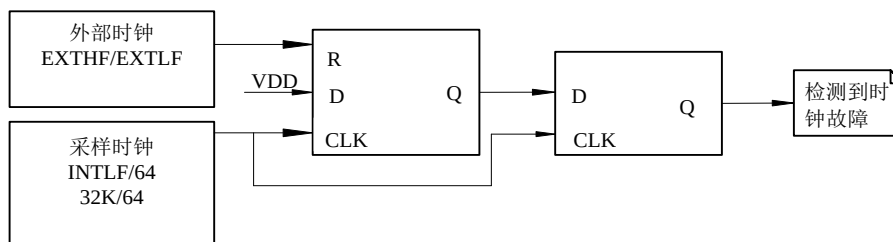


图 1.9 时钟故障检测原理图

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。

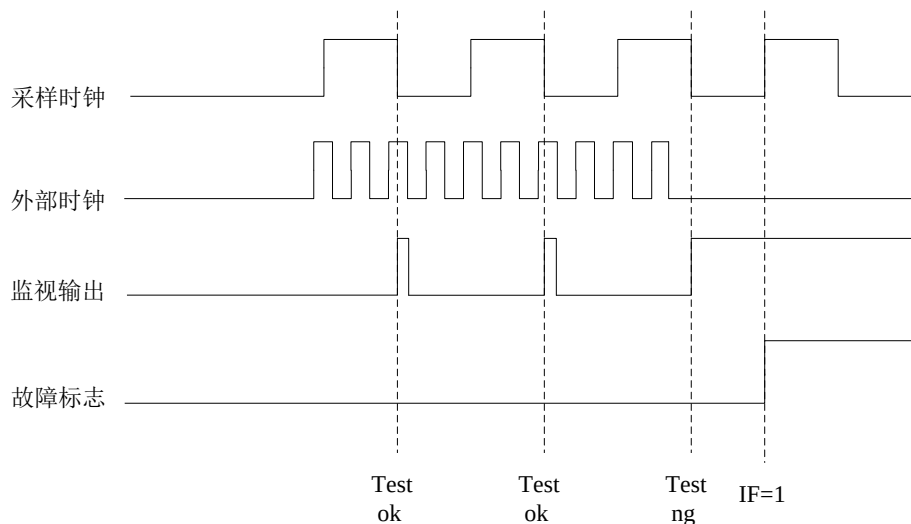


图 1.10 时钟故障检测时序

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

如寄存器 1.2 所示，用户在下载程序时，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。KF8L10Z08 的配置位映射在 2007H 和 2008H 地址单元。

CONFIG 配置字

R	R	R	R	R	R/P	R/P	R	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R	R/P
-	-	-	-	-	DEBUG	SWRTEN	-	CODEP	LVREN	RSTEN	PWRT	WDTEN	DATAP	-	LPLVREN	
bit15					bit8										bit0	

注：R=编程器可读 P=编程时可写 -=未实现位

DEBUG： 在线调试使能位

DEBUG=1 禁止在线调试

DEBUG=0 使能在线调试

SWRTEN： 系统保留位

该位必须置 1，不可配置为 0

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： P2.0/ RST 引脚功能选择

RSTEN=1 P2.0/ RST 引脚配置为外部复位输入

RSTEN=0 P2.0/ 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（建议上电延时使能，保证启动的稳定性）

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

DATAP=1 禁止数据存储区加密

DATAP=0 使能数据存储区加密

LPLVREN： LPLVR 使能位

LPLVREN=1 禁止 LPLVR

LPLVREN=0 使能 LPLVR

注：如需使用欠压检测 (LVR) 功能，需要将配置位的 LVREN 位置 1，LPLVREN 位置 1，PCTL 寄存器的 SLVREN 位置 1；即 LVREN=1，LPLVREN=1 且 SLVREN=1，才能使能 LVR 功能；LPLVREN 位为 LPLVR 模块的使能位，该位只在 PCTL 寄存器的 SLVREN 位为 1 时有效，即当 SLVREN=1 且 LPLVREN=0 时，才能使能 LPLVR 模块，否则无法使能。如下表：

表 1-5-1：LVR/LPLVR 使能配置方法表

功能	LVREN	LPLVREN	PCTL_SLVREN
LVR 使能配置	1	1	1
LPLVR 使能配置	0/1	0	1

1.6 在线串行编程

如图 1.11、1.12 所示，在最终应用电路中可对 KF8L10Z08 单片机进行在线串行编程、调试。实现串行编程仅需要五根线包括：时钟线(SPCLK)、数据线(SPDAT)、电源线(V_{DD})、地线(V_{SS})、复位线(RST)。

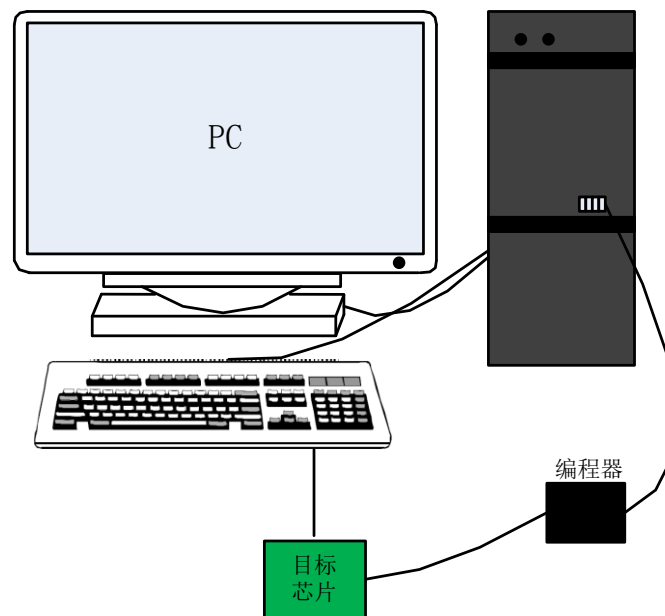


图 1.11 在线调试系统示意图

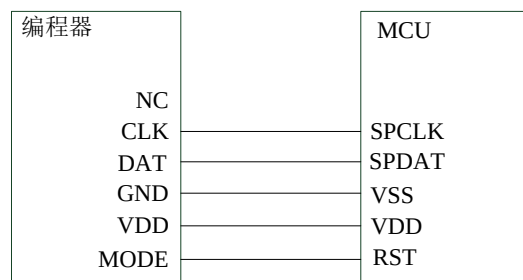


图 1.12 在线串行编程连接图

2 I/O 端口介绍

KF8L10Z08 单片机除电源正极、电源负极和 VDDcore 外，有三组通用 I/O 口，包括 P0 口、P1 口、P2 口，均可以作为通用 I/O 口使用。具体封装对应可用 IO 可以参见前面的“引脚总述”。

使用时请将未开放的管脚配置为模拟输入口或者数字输出口，不能配置为数字输入口，否则会引起单片机功耗增大。

通过一系列的 I/O 口配置寄存器完成对 I/O 口的功能配置，KF8L10Z08 单片机的通用 I/O 口控制方式简介如下：

模拟/数字口功能配置：通过 ANSE0/ANSE1/ANSE2 寄存器配置 I/O 口的功能模式；

状态读取：通过 P0/P1/P2 寄存器读取当前 I/O 口的逻辑电平；

方向控制：通过 TR0/TR1/TR2 寄存器控制 I/O 口的输入输出状态；

输出控制：通过 P0LR/P1LR/P2LR 寄存器配置 I/O 口的逻辑电平输出；

上拉功能：通过 PUR0/PUR1/PUR2 寄存器及总使能位 $\overline{PUPH}$ 配置 I/O 口的上拉功能；

电平变化中断功能（仅 P0 口）：通过 IOCL 寄存器控制 P0 口的电平变化功能。

注：当使用模拟外设并需要将特定 I/O 口配置为模拟功能时，需要将该 I/O 口配置为模拟输入状态，可分别通过 ANSE_x 寄存器（配置数字/模拟）和 TR_x 寄存器（输入/输出）完成。

注：用户在正常使用时，通常会有一些用不到的引脚或者未开放的管脚，如果直接把这些管脚悬空，而不做其他处理可能使单片机功耗增大，因此建议将那些不用的引脚设置为数字输出模式或者模拟输入口。

2.1 I/O 端口的读写

读 P0/P1/P2 口时实际为读端口电平，即读 $P_x(x=0/1/2)$ ，改变 IO 端口的输出状态通过写 $P_xLR(x=0/1/2)$ 寄存器实现。其原理框图如图 2.1 所示：

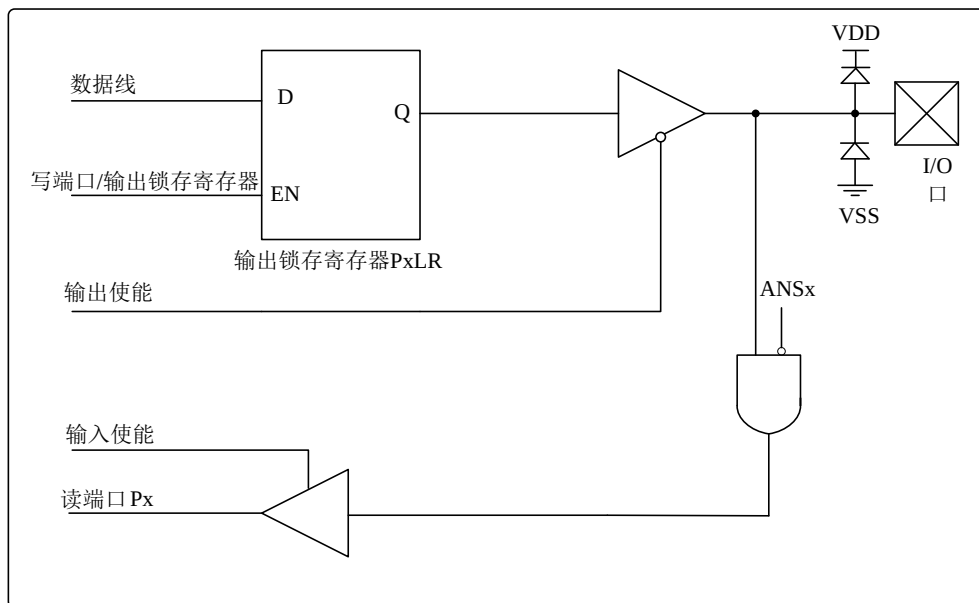


图 2.1 I/O 口读写原理图

- 注 1: I/O 口作为数字输入时，需将对应的 ANS_x 位清 0。
- 注 2: I/O 口作为模拟输入时，需将对应的 ANS_x 位置 1。
- 注 3: IO 端口作为输出时，应对 $P_xLR(x=0/1/2)$ 进行赋值(寄存器操作或位操作)，以避免 IO 端口的读-修改-写操作引起的错误。

2.2 P0 口

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。

表 2-1 P0 口各引脚功能

引脚名	I/O	引脚功能	引脚说明
P0.0/SDI/SDA /SPDAT/INT0/ ADCH0/PWM1 1(1)	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
		SDI	I2C 数据输入/输出脚
		SDA	SPI 数据输入脚
		SPDAT	编程数据输入/输出
		ADCH0	ADC 输入通道 0
		PWM11	PWM11 输出通道（复用）
		INT0	外部中断 0 输入
P0.1/SCK/SCL /ADVRIN /SPCLK/INT1/ ADCH1/PWM2 2(1)	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
		SCK	SPI 时钟脚
		SCL	I2C 时钟脚
		ADVRIN	AD 外部参考电压输入
		SPCLK	编程时钟输入
		ADCH1	ADC 输入通道 1
		PWM22(1)	PWM22 输出通道（复用）
P0.2/ADCH2/S CK(1)/SCL(1)/ C1OUT	I/O	P0.2	带上拉和电平变化中断功能的双向输入输出端口
		ADCH2	ADC 输入通道 2
		SCK (1)	SPI 时钟脚（复用）
		SCL (1)	I2C 时钟脚（复用）
		C1OUT	比较器 1 输出通道
P0.4/ $\overline{\text{T1G}}$ /CLK OUT/ $\overline{\text{SS}}$ /ADCH 3/OSCIN/PWM 12(1)	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
		$\overline{\text{T1G}}$	T1 门控信号输入
		CLKOUT	系统时钟输出
		$\overline{\text{SS}}$	SPI 模式从动选择输入
		ADCH3	ADC 输入通道 3
		PWM12(1)	PWM12 输出通道（复用）
		OSCIN	外部振荡器输入引脚 B
P0.5/T1CK /OSCOU/ADC H13	I/O	P0.5	带上拉和电平变化中断功能的双向输入输出端口
		T1CK	T1 时钟输入
		ADCH13	ADC 输入通道 13
		OSCOU	外部振荡器输入引脚 A

2.2.1 P0 口相关的寄存器

表 2-2 与 P0 端口相关的寄存器

地址	名称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
05H	P0	-	-	P05	P04	-	P02	P01	P00
45H	P0LR	-	-	P0LR5	P0LR4	-	P0LR2	P0LR1	P0LR0
25H	TR0	-	-	TR05	TR04	-	TR02	TR01	TR00
36H	IOCL	-	-	IOCL5	IOCL4	-	IOCL2	IOCL1	IOCL0
35H	PUR0	-	-	PUR05	PUR04	-	PUR02	PUR01	PUR00
41H	ANSE0	-	-	ANS05	ANS04	-	ANS02	ANS01	ANS00

2.2.1.1 P0 口状态读取寄存器（P0）

寄存器 P0 各位对应 P0 口相应引脚的状态，如寄存器 2.1 所示：

寄存器2.1: P0: P0口状态输出寄存器(地址: 05H)

复位值 --XX XXXX	bit7							bit0
	-	-	P05	P04	-	P02	P01	P00
	U	U	R/W	R/W	R/W	R/W	R/W	R/W

P0<5:4>: 读 P0 口各引脚电平

P0<2:0>: 读 P0 口各引脚电平

1 = 对应引脚为逻辑高电平

0 = 对应引脚为逻辑低电平

图注: R=可读 W=可写 -=未用 U=未实现位

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 P0 口的输出状态。

寄存器2.2: P0LR: P0口输出锁存寄存器(地址: 45H)

复位值 XXXX XXXX	bit7							bit0
	-	-	P0LR5	P0LR4	-	P0LR2	P0LR1	P0LR0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

P0LR<5:4>: 写 P0 口输出状态

P0LR<2:0>: 写 P0 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

图注: R=可读 W=可写 -=未用 U=未实现位

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。系统复位时, P0 口各引脚默认为输入口。

寄存器2.3: TR0: P0口方向控制寄存器(地址: 25H)

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

TR0<5:4>: P0 口各引脚方向控制位

TR0<2:0>: P0 口各引脚方向控制位

1 = 对应的引脚设置为输入

0 = 对应的引脚设置为输出

图注: R=可读 W=可写 -=未用 U=未实现位

2.2.1.4 P0 口上拉功能控制寄存器(PUR0)

KF8L10Z08 中 P0 口开放的五个引脚均带有上拉功能, 可通过上拉功能控制寄存器和 OPTR 寄存器中的 $\overline{\text{PUPH}}$ 来控制上拉功能是否打开。

如果要将某引脚的上拉功能打开, 需要先将 $\overline{\text{PUPH}}$ (上拉功能总使能位)位清 0, 允许 P0 口上拉功能打开, 然后再将要打开上拉功能的引脚所对应的上拉功能控制位置 1 即可。寄存器 2.4 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口上拉控制寄存器(地址: 35H)

复位值 1111 -111	bit7							bit0
	-	-	PUR05	PUR04	-	PUR02	PUR01	PUR00
	R/W	R/W	R/W	R/W	U	R/W	R/W	R/W

PUR0<5:4>: 上拉功能使能位

PUR0<2:0>: 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

图注: R=可读 W=可写 -=未用 U=未实现位

2.2.1.5 P0 口电平变化中断控制寄存(IOCL)

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，IOCL 为电平变化中断控制寄存器，将 IOCL 某位置 1 将开启对应引脚的电平变化中断功能，如果该引脚电平发生变化，不管电平变化中断是否使能，电平变化中断标志位(P0IF)都会置 1，如果全局中断使能位(AIE)和电平变化中断使能位(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: IOCL: P0口电平变化中断控制寄存器(地址:36H)

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

IOCL<5:4><2:0>: P0 口电平变化中断使能控制位

1 = 使能对应引脚的电平变化中断

0 = 禁止对应引脚的电平变化中断

注： P0 口电平变化中断是在引脚的当前电平与上次读 P0 寄存器时的电平不匹配时产生的，所以每次中断标志位(P0IF)置 1 后都要更新 P0 寄存器的值。

电平变化中断参考:

JB INTCTL,P0IF ;检测是否为P0电平变化中断
JMP INT_END ;退出中断
MOV P0 ;锁存P0口状态
CLR INTCTL,P0IF ;清零电平变化中断标志位

2.2.1.6 P0 口模拟/数字口设置寄存器 (ANSE0)

P0 口模拟/数字口设置寄存器 ANSE0 用于将 P0 口设置为模拟口或者数字口，通过将 ANSE0 某位置 1，将对应的引脚设置为模拟口，清 0 设置为数字口。

寄存器2.6: ANSE0: P0口模拟/数字口设置寄存器(地址: 41H)

复位值 1111 1111	bit7							bit0
	-	-	ANS05	ANS04	-	ANS02	ANS01	ANS00
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ANS0<2:0>: P0 口各引脚模拟/数字口设置位

ANS0<5:4>: 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

如图 2.2 所示，为 P0 口引脚内部原理功能框图。

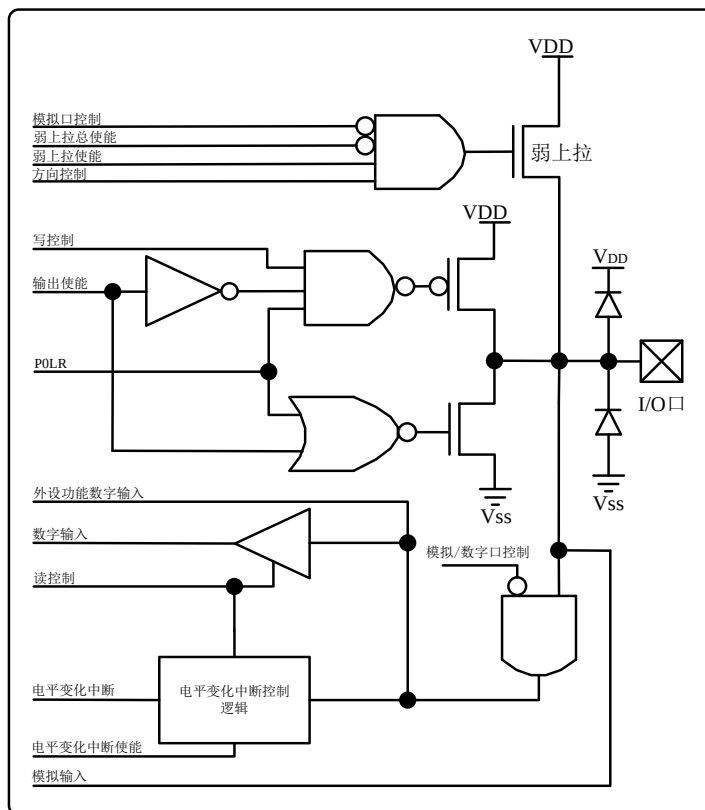


图 2.2 P0 口引脚原理框图

2.3 P1 口

P1 口均可作为普通 I/O 口。引脚功能如表 2-3 所示。

表 2-3 P1 口各引脚功能

引脚名	I/O	引脚功能	引脚说明
P1.0/PWM22/ADCH4/SDI(1)/SDA(1)/INT2(1)/CATCH2(1)/CATCH3(1)	I/O	P1.0	带上拉的双向输入输出端口
		SDI(1)	I2C 数据输入/输出脚
		SDA(1)	SPI 数据输入脚
		INT2(1)	外部中断 2 输入口（复用）
		CATCH2(1)	定时器 2 捕捉功能输入引脚（复用）
		CATCH3(1)	定时器 3 捕捉功能输入引脚（复用）
		PWM22	PWM22 输出通道
		ADCH4	ADC 输入通道 4
P1.1/ADCH5/PWM21/INT1(1)/CATCH2(1)	I/O	P1.1	带上拉的双向输入输出端口
		INT1(1)	外部中断 1 输入口（复用）
		CATCH2(1)	定时器 2 捕捉功能输入引脚（复用）
		PWM21	PWM21 输出通道
		ADCH5	ADC 输入通道 5
P1.2/PWM12/INT2/ADCH6/CATCH3(1)	I/O	P1.2	带上拉的双向输入输出端口
		INT2	外部中断 2 输入口
		CATCH3(1)	定时器 3 捕捉功能输入引脚（复用）
		PWM12	PWM12 输出通道
		ADCH6	ADC 输入通道 6
P1.3/ADCH8/PWM11/C2+	I/O	P1.3	带上拉的双向输入输出端口
		ADCH8	ADC 输入通道 8
		C2+	比较器 2 正端输入端口
		PWM11	PWM11 输出通道
P1.4/SDO/ADCH9/INT1(1)/CATCH2(1)	I/O	P1.4	带上拉的双向输入输出端口
		ADCH9	ADC 输入通道 9
		INT1(1)	外部中断 1 输入口(复用)
		CATCH2(1)	定时器 2 捕捉功能输入引脚（复用）
		SDO	SPI 数据输出端
P1.5/ADCH10/INT2(1)/CATCH3(1)	I/O	P1.5	带上拉的双向输入输出端口
		ADCH10	ADC 输入通道 10
		INT2(1)	外部中断 2 输入口(复用)
		CATCH3(1)	定时器 3 捕捉功能输入引脚（复用）

2.3.1 P1 口相关的寄存器

表 2-4 与 P1 口相关的寄存器

地址	名称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
07H	P1	-	-	P15	P14	P13	P12	P11	P10
47H	P1LR	-	-	P1LR5	P1LR4	P1LR3	P1LR2	P1LR1	P1LR0
27H	TR1	-	-	TR15	TR14	TR13	TR12	TR11	TR10
50H	PUR1	-	-	PUR15	PUR14	PUR13	PUR12	PUR11	PUR10
31H	ANSE1	-	-	ANS15	ANS14	ANS13	ANS12	ANS11	ANS10

2.3.1.1 P1 口状态读取寄存器（P1）

寄存器 P1 各位对应 P1 口相应引脚的状态，如寄存器 2.7 所示：

寄存器2.7: P1: P1口状态读取寄存器(地址：07H)

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

P1<5:0>: 读 P1 口各引脚电平

1 = 对应引脚为逻辑高电平

0 = 对应引脚为逻辑低电平

图注：R=可读 W=可写 -=未用 U=未实现位

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 P1 口的输出状态。

寄存器2.8: P1LR: P1口输出锁存寄存器(地址：47H)

复位值	bit7							bit0
xxxx xxxx	-	-	P1LR5	P1LR4	P1LR3	P1LR2	P1LR1	P1LR0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

P1LR<5:0>: 写 P1 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

2.3.1.3 P1 口方向控制寄存器(TR1)

如寄存器 2.9 所示，TR1 为 P1 口方向控制寄存器，当 TR1 某位置 1 时，将该引脚设置为输入，此时引脚为三态(悬空)，TR1 某位清 0，对应引脚设置为输出。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 1111 1111	bit7						bit0	
	-	-	TR15	TR14	TR13	TR12	TR11	TR10
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

TR1<5:0>: P1 口引脚方向控制位

1 = P1 口对应引脚被配置为输入端口

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)

KF8L10Z08 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 $\overline{\text{PUPH}}$ 来控制上拉功能是否打开。

如果要将某引脚的上拉功能打开，需要先将 $\overline{\text{PUPH}}$ (上拉功能总使能位)位清 0，允许 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 P1 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口上拉控制寄存器(地址: 50H)

复位值 1111 1111	bit7						bit0	
	-	-	PUR15	PUR14	PUR13	PUR12	PUR11	PUR10
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

PUR1<5:0>: 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

2.3.1.5 P1 口模拟/数字口设置寄存器 (ANSE1)

P1 口模拟/数字口设置寄存器 ANSE1 用于将 P1 口设置为模拟口或者数字口，通过将 ANSE1 某位置 1，将对应的引脚设置为模拟口，清 0 设置为数字口。

寄存器2.11: ANSE1: P1口模拟/数字口设置寄存器(地址: 31H)

复位值 1111 1111	bit7						bit0	
	-	-	ANS15	ANS14	ANS13	ANS12	ANS11	ANS10
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ANS1<5:0>: 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

如图 2.4 所示，为 P1 口内部原理功能框图。

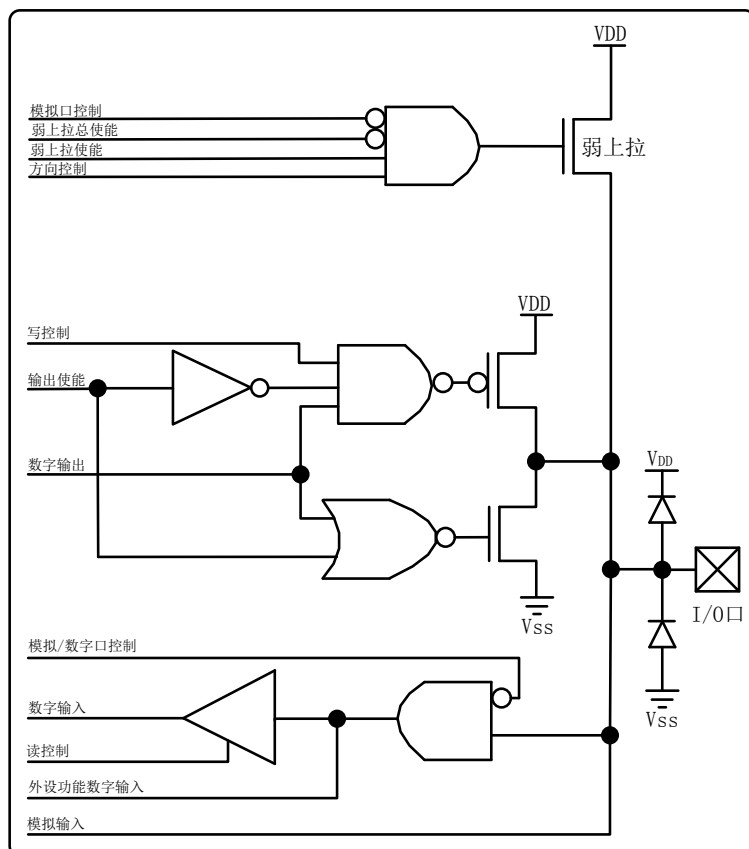


图 2.4 P1 口引脚原理功能框图

2.4 P2 口

如表 2-5 所示，KF8L10Z08 型号单片机 P2 口具有 7 个引脚。所有管脚均可作为普通 I/O 口。引脚功能如表 2-3 所示。

表 2-5 P2 口各引脚功能

引脚名	I/O	引脚功能	引脚说明
P2.0/RST	I/O	P2.0	带上拉的双向输入输出端口
		RST	外部复位输入
P2.1/ADCH15/ C1+	I/O	P2.1	带上拉的双向输入输出端口
		ADCH15	ADC 输入通道 15
		C1+	比较器 1 正端输入端口
P2.2/ $\overline{SS}$ /ADCH 7/PWM21(1)/C 1-	I/O	P2.2	带上拉的双向输入输出端口
		ADCH7	ADC 输入通道 7
		PWM21(1)	PWM21 输出端口（复用）
		C1-	比较器 1 负端输入端口
		$\overline{SS}$	SPI 模式从动选择输入
P2.4/ADCH17/ PWM21(1)/C2-	I/O	P2.4	带上拉的双向输入输出端口
		ADCH17	ADC 输入通道 17
		PWM21(1)	PWM21 输出端口（复用）
		C2-	比较器 2 负端输入端口
P2.6/ADCH19/ INT1(1)/CATC H2/C2OUT	I/O	P2.6	带上拉的双向输入输出端口
		ADCH19	ADC 输入通道 19
		INT1(1)	外部中断 1 输入口（复用）
		CATCH2	定时器 2 捕捉功能输入引脚（复用）
		C2OUT	比较器 2 结果输出端口
P2.7/ADCH20/ INT2(1)/CATC H3/PWM11(1)	I/O	P2.7	带上拉的双向输入输出端口
		ADCH20	ADC 输入通道 20
		INT2(1)	外部中断 2 输入口（复用）
		CATCH3	定时器 3 捕捉功能输入引脚（复用）
		PWM11(1)	PWM11 输出端口（复用）

2.4.1 P2 口相关的寄存器

表 2-6 与 P2 口相关的寄存器

地址	名称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
06H	P2	P27	P26	-	P24	-	P22	P21	P20
46H	P2LR	P2LR7	P2LR6	-	P2LR4	-	P2LR2	P2LR1	P2LR0
26H	TR2	TR27	TR26	-	TR24	-	TR22	TR21	TR20
5EH	PUR2	PUR27	PUR26	-	PUR24	-	PUR22	PUR21	PUR20
1DH	ANSE2	ANS27	ANS26	-	ANS24	-	ANS22	ANS21	ANS20

2.4.1.1 P2 口状态寄存器（P2）

寄存器 P2 各位对应 P2 口相应引脚的状态，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	-	P24	-	P22	P21	P20	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

P2<7:6><4><2:0>: 读 P2 口各引脚电平
1 = 对应引脚为逻辑高电平
0 = 对应引脚为逻辑低电平

图注: R=可读 W=可写 -=未用 U=未实现位

2.4.1.2 P2 口输出锁存寄存器（P2LR）

寄存器 P2LR 是 P2 口输出锁存寄存器。在 P2 口作为输出时，通过写 P2LR 寄存器来设置 P2 口的输出状态。

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	-	P2LR4	-	P2LR2	P2LR1	P2LR0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

P2LR<7:6><4><2:0>: 写 P2 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

图注: R=可读 W=可写 -=未用 U=未实现位

2.4.1.3 P2 口方向控制寄存器 (TR2)

如寄存器 2.14 所示, 通过将寄存器 TR2 中的某位置 1, 将对应管脚设置为输入口, 清 0 设置为输出口。

寄存器2.14: TR2: P2口方向控制寄存器(地址: 26H)

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

TR2<7:6><4><2:0>: P2 口各引脚方向控制位

1 = P2 口对应引脚被配置为输入端口

0 = P2 口对应引脚被配置为输出端口

图注: R=可读 W=可写 -=未用 U=未实现位

2.4.1.4 P2 口上拉控制寄存器 (PUR2)

寄存器2.15: PUR2: P2口上拉控制寄存器(地址: 5EH)

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

PUR2<7:6><4><2:0>: 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

图注: R=可读 W=可写 -=未用 U=未实现位

2.4.1.5 P2 口模拟/数字口设置寄存器 (ANSE2)

P2 口模拟/数字口设置寄存器 ANSE2 用于将 P2 口设置为模拟口或者数字口, 通过将 ANSE2 某位置 1, 将对应的引脚设置为模拟口, 清 0 设置为数字口。

寄存器2.16: ANSE2: 模拟/数字口设置寄存器(地址: 1DH)

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

ANS2<7:6><4><2:0>: 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

图注: R=可读 W=可写 -=未用 U=未实现位

2.4.2 P2 口原理功能框图

如图 2.5 所示，为 P2 口内部原理功能框图。

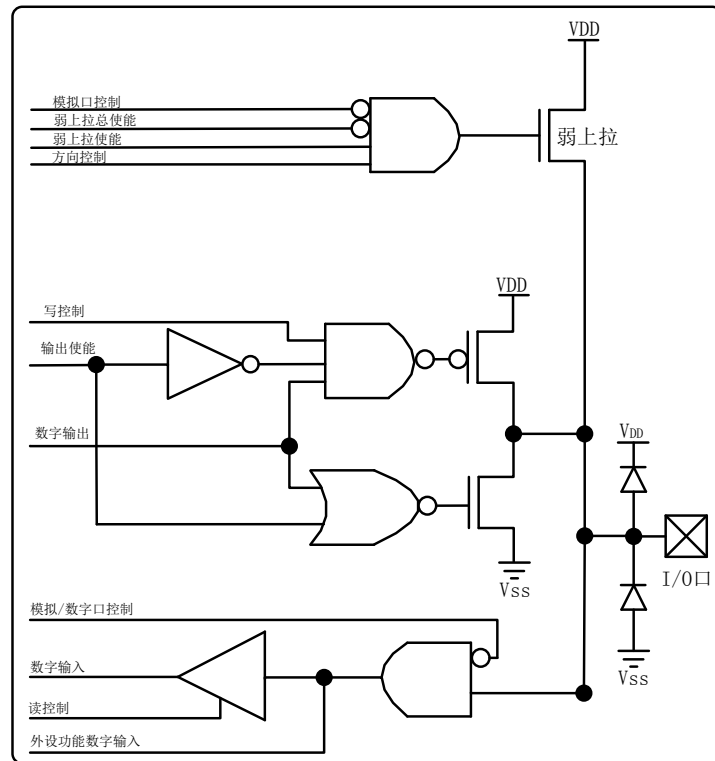


图 2.5 P2 口引脚原理功能框图

2.5 I/O 口功能复用

部分引脚功能可以通过软件配置 APFCTLx 寄存器，对所需的 I/O 口进行选择。

表 2-7 与 I/O 口功能复用相关的寄存器

地址	名称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
5BH	APFCTL0	PWM22 SEL1	PWM22 SEL0	PWM21 SEL1	PWM21 SEL0	PWM12 SEL1	PWM12 SEL0	PWM11 SEL1	PWM11 SEL0
40H	APFCTL1	-	-	SSCI SEL	SSSEL	INT2 SEL1	INT2 SEL0	INT1 SEL1	INT1 SEL0
58H	APFCTL2	-	-	-	-	CAT3 SEL1	CAT3 SEL0	CAT2 SEL1	CAT2 SEL0

2.5.1 引脚复用功能控制寄存器 0（APFCTL0）

寄存器： APFCTL0:引脚复用控制寄存器(5BH)

复位值 0000 0000	bit7				bit0			
	PWM22 SEL1	PWM22 SEL0	PWM21 SEL1	PWM21 SEL0	PWM12 SEL1	PWM12 SEL0	PWM11 SEL1	PWM11 SEL0
	W	W	W	W	W	W	W	W

PWM22SEL<1:0>: PWM22 输出功能引脚选择位

00 = PWM22 功能位于 P1.0 引脚

01 = PWM22 功能位于 P0.1 引脚

10 = 保留

11 = 保留

PWM21SEL<1:0>: PWM21 输出功能引脚选择位

00 = PWM21 功能位于 P1.1 引脚

01 = PWM21 功能位于 P2.2 引脚

10 = PWM21 功能位于 P2.4 引脚

11 = 保留

PWM12SEL<1:0>: PWM12 输出功能引脚选择位

00 = PWM12 功能位于 P1.2 引脚

01 = PWM12 功能位于 P0.4 引脚

10 = 保留

11 = 保留

PWM11SEL<1:0>: PWM11 输出功能引脚选择位

00 = PWM11 功能位于 P1.3 引脚

01 = PWM11 功能位于 P2.7 引脚

10 = PWM11 功能位于 P0.0 引脚

11 = 保留

图注：R=可读 W=可写 -=未用 U=未实现位

注：

1.该寄存器为只写寄存器，对该寄存器进行读操作时将读出无效数据；

2.该寄存器无法进行位写操作，在对该寄存器进行写操作时，请对整个寄存器进行写操作。

2.5.2 引脚复用功能控制寄存器 1 (APFCTL1)

寄存器: APFCTL1:引脚复用控制寄存器1(40H)

复位值 0000 0000	bit7				bit0			
	保留	保留	SSCISEL	SSSEL	INT2SEL1	INT2SEL0	INT1SEL1	INT1SEL0
	W	W	W	W	W	W	W	W

- SSCISEL: SSCI 模块功能引脚选择位
 0= SDI/SDA 功能位于 P0.0 引脚, SCK/SCL 功能位于 P0.1 引脚
 1= SDI/SDA 功能位于 P1.0 引脚, SCK/SCL 功能位于 P0.2 引脚
- SSSEL: SSCI 模块的 SS 功能引脚选择位
 0= $\overline{SS}$ 功能位于 P0.4 引脚
 1= $\overline{SS}$ 功能位于 P2.2 引脚
- INT2SEL<1:0>: INT2 中断输入引脚选择位
 00= INT2 中断输入功能位于 P1.2 引脚
 01= INT2 中断输入功能位于 P2.7 引脚
 10= INT2 中断输入功能位于 P1.5 引脚
 11= INT2 中断输入功能位于 P1.0 引脚
- INT1SEL<1:0>: INT1 中断输入引脚选择位
 00= INT1 中断输入功能位于 P0.1 引脚
 01= INT1 中断输入功能位于 P1.1 引脚
 10= INT1 中断输入功能位于 P2.6 引脚
 11= INT1 中断输入功能位于 P1.4 引脚

图注: R=可读 W=可写 -=未用 U=未实现位

注:

- 1.该寄存器为只写寄存器, 对该寄存器进行读操作时将读出无效数据;
- 2.该寄存器无法进行位写操作, 在对该寄存器进行写操作时, 请对整个寄存器进行写操作。

2.5.3 引脚复用功能控制寄存器 2 (APFCTL2)

寄存器: APFCTL2:引脚复用控制寄存器(58H)

	bit7							bit0
复位值 ---- 0000	-	-	-	-	CAT3 SEL1	CAT3 SEL0	CAT2 SEL0	CAT2 SEL0
	U	U	U	U	W	W	W	W

CAT3SEL<1:0>: T3 捕捉功能输入引脚选择位
 00 = CATCH3 功能位于 P2.7 引脚
 01 = CATCH3 功能位于 P1.5 引脚
 10 = CATCH3 功能位于 P1.2 引脚
 11 = CATCH3 功能位于 P1.0 引脚

CAT2SEL<1:0>: T2 捕捉功能输入引脚选择位
 00 = CATCH2 功能位于 P2.6 引脚
 01 = CATCH2 功能位于 P1.4 引脚
 10 = CATCH2 功能位于 P1.1 引脚
 11 = CATCH2 功能位于 P1.0 引脚

图注: R=可读 W=可写 -=未用 U=未实现位

注:

- 1.该寄存器为只写寄存器，对该寄存器进行读操作时将读出无效数据;
- 2.该寄存器无法进行位写操作，在对该寄存器进行写操作时，请对整个寄存器进行写操作。

3 存储器

如图 3.1 所示, KF8L10Z08 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 8K 字节的 FLASH 存储器;

数据存储器由特殊功能寄存器和通用寄存器组成, 特殊功能寄存器空间为 128×8 位, 通用数据寄存器空间为 $400(384+16) \times 8$ 位。另外 KF8L10Z08 中还有一些其它存储器, 包括: 工作寄存器组 R0~R7、8 级硬件堆栈、ID 地址单元等。

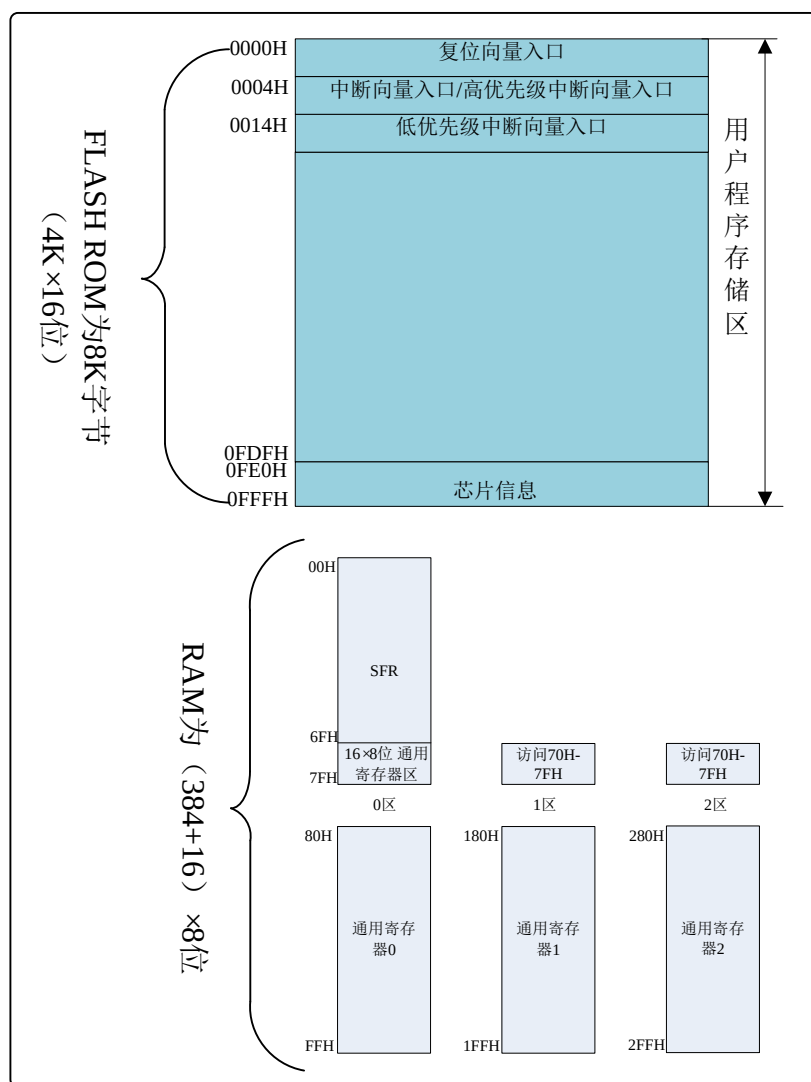


图 3.1 存储器组织图

3.1 程序存储器(ROM)区

KF8L10Z08 有一个 13 位的程序计数器, 实现了 8K byte 的程序存储空间, 地址为 0000H~0FFFH, 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。

如图 3.2 所示, 程序计数器(PC)的低 8 位(PC<7:0>)来自特殊功能寄存器 PCL, 高 5 位(PC<12:8>)来自 PCH 寄存器。在任何复位发生后 PC 值将被清 0。在有任何未屏蔽中断发生后 PC 值将指向 0004H 或 0014H 地址。图 3.3 为程序存储器区的地址映射图。

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

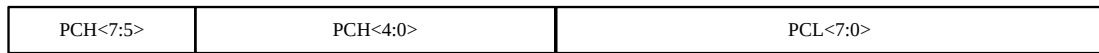


图 3.2 程序计数器 (PC)

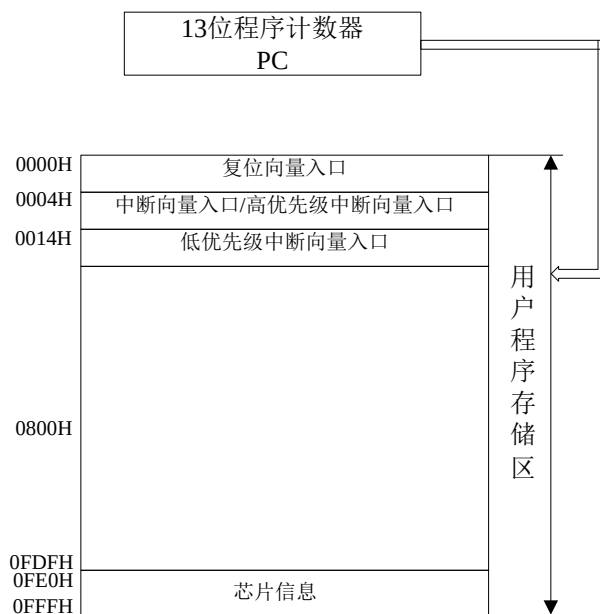


图 3.3 KF8L10Z08 程序存储器映射

3.1.1 JMP、CALL 指令

KF8L10Z08 系列单片机的 JMP、CALL 指令编码如下:

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.bit4 以及指令所带立即数 (#data12), 如图 3.4 所示。

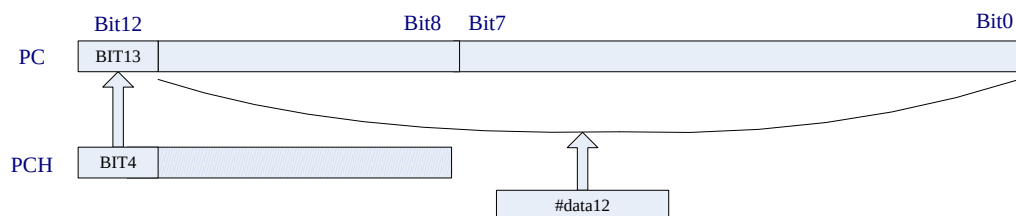


图 3.4 执行 JMP、CALL 指令时 PC 的变化图

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。

3.2 数据存储器(RAM)区

如图 3.5 所示, KF8L10Z08 中的数据存储器由 4 个区组成, 每个区的空间都是 128 字节, 其中 1 个区用作特殊功能寄存器区(SFR)使用; 另外 3 个存储器区为通用寄存器区, 由用户支配。SFR 地址空间为 00H~7FH, 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。

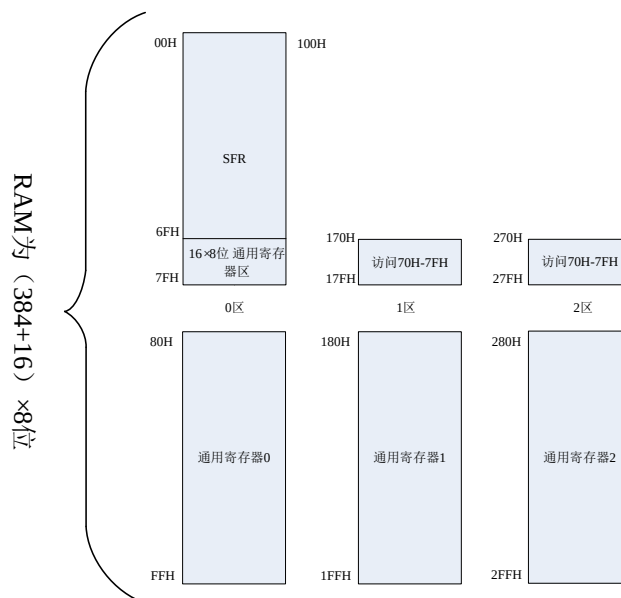


图 3.5 数据存储器地址映射图

3.2.1 通用寄存器区

如图 3.5 所示, 通用寄存器的空间为 400 字节, 0 区至 2 区通过 PSW 中的 RP0(PSW.5)、RP1(PSW.6)位进行选择, 如表 3.1 所示。

表 3-1 通用寄存区地址

RP<1:0>	通用寄存器区	地址
00	通用寄存器 0 区	80H~FFH
01	通用寄存器 1 区	180H~1FFH
10	通用寄存器 2 区	280H~2FFH
11	系统保留	

3.2.2 特殊功能寄存器(SFR)区

KF8L10Z08 内部的 I/O 口控制、定时/计数器、中断等各种控制寄存器和状态寄存器都称为特殊功能寄存器。附录 1 列出 SFR 的地址映射及复位初始值等。

状态字寄存器(PSW): 如寄存器 3.1 所示,PSW 的低三位是算术运算标志位,在进行加、减等运算时对它们产生影响(具体请参考汇编指令部分)。 $\overline{TO}$ 和 $\overline{PD}$ 是复位状态位,当单片机有复位或看门狗超时、执行休眠等指令时,会对这两位产生影响。 $RP0$ 、 $RP1$ 为通用存储器区选择位。

寄存器3.1: PSW: 状态字寄存器(地址: 03H)

bit7				bit0			
复位值 0001 1xxx	-	$RP1$	$RP0$	$\overline{TO}$	$\overline{PD}$	Z	CY
	保留	R/W	R/W	R	R	R/W	R/W

$RP<1:0>$: 通用存储器区选择位

00 = 通用寄存器0区

01 = 通用寄存器1区

10 = 通用寄存器2区

11 = 保留

$\overline{TO}$: 超时标志位

1 = 在上电复位、CWD $\overline{T}$ 指令或IDLE指令执行之后

0 = WDT超时被清0

$\overline{PD}$: 上电复位标志位

1 = 上电复位或执行CWD $\overline{T}$ 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)

图注: R=可读 W=可写 -=未用 U=未实现位

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

KF8L10Z08 片内的 DATA EEPROM 存储器最大容量为 128×8 位，地址范 00H~7FH，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。DATA EEPROM 通过如下 4 组寄存器操作。

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	NVMADDRL	DATA EEPROM 地址指针低 8 位							
3CH	NVMCTL0	NVM 控制寄存器 0							
3DH	NVMCTL1	NVM 控制寄存器 1							
39H	NVMDATAL	NVM 数据寄存器低 8 位							

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 NVMADDRL

DATA EEPROM 最大容量为 128×8 位，地址范围 0~127，只需 7 位地址线参与译码。NVMADDRL 寄存器用来存放要写入 DATA EEPROM 的 7 位的地址信息，最高位不参与译码。

3.3.3 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，NVMADDRL 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 NVMADDRL 中，然后向 NVMCTL0 写入固定的读命令，把要读的数据送到 NVMDATAL 中。

3.3.4 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 NVMADDRL，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：

```
MOV R5, BANK           ;保存当前寄存器存储区
CLR BANK               ;切换到Bank0区
MOV DATA_BANK, R5     ;该样例要求DATA_BANK在0区,否则添加切区

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

MOV R5, OSCCTL          ;保存当前的时钟状态
MOV DATA_OSCCTL, R5   ;该样例要求DATA_OSCCTL在0区,否则添加切区
MOV R5, #0X30          ;切换到250kHz
MOV OSCCTL, R5

;;以下时序不可更改
MOV R5, #0X04
MOV NVMCTL0, R5
MOV R5, #0X69
MOV NVMCTL1, R5
MOV R5, #0X96
MOV NVMCTL1, R5
SET NVMCTL0, 1
NOPZ
NOPZ
NOPZ
NOPZ
NOPZ
NOPZ
NOPZ
NOPZ
NOPZ
NOPZ
NOPZ
MOV R5, #0X00          ;关闭DATA EEPROM的写操作，防止意外写
MOV NVMCTL0, R5

NOP
JB EIF1, EEIF
JMP $-1
CLR EIF1, EEIF
NOP

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

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

MOV R5, DATA_BANK     ;BANK区还原
MOV BANK, R5
```

以上指令中的立即数 0X04, 0X69, 0X96, 0X00 是固定不变的。如果未完全按照上述顺序(先将 0X04 写入 NVMCTL0, 再将 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 DATA EEPROM 时请严格按照样例程序进行操作，否则可能导致无法正常写入的情况。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：CPU写DATA EEPROM时,不管DATAP设置为何值，都能写入正确的数据

写 DATA EEPROM 的步骤：

1. 把系统时钟调整到 500K 或 250K;
2. 将要写入的数据送到 NVMDATAL;
3. 将对应的 DATA EEPROM 地址送到 NVMADDRL;
4. 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。用户需等待 6ms 用来执行擦除和写入数据的命令，此期间 CPU 仍可正常工作；
5. 重复执行步骤 2、3、4，可以执行其他地址的写入。

3.3.5 读 DATA EEPROM

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

```
MOV R0, #0X01
MOV NVMCTL0, R0
NOPZ
MOV R0, NVMDATAL ;R0=NVMDATAL
```

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 NVMDATAL。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。NVMDATAL 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 NVMADDRL 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 NVMDATAL。

3.4 寄存器组 Rn

KF8L10Z08 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

KF8L10Z08 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 0FE0H~0FFFH，用于存放芯片校准信息。

4 汇编指令及寻址方式

4.1 寻址方式

KF8L10Z08 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。

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。
指令中含有一个操作数，寻址方式为直接寻址。

4.1.3 立即数寻址

在指令中的操作数为立即数。

例：

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

4.1.4 寄存器间接寻址

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

例:

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

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

4.1.5 位寻址

指令中的操作数是寄存器的某位,这样的寻址方式称为位寻址。

例:

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

4.2 汇编指令

KF8L10Z08 系列单片机汇编指令共有 68 条,除子程序调用、子程序返回、中断返回、部分跳转指令为双周期指令外,其余指令均为单周期指令。所有指令都占两个字节。

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

5 中断

KF8L10Z08 单片机的中断源有：

- INT0/1/2 中断
- T0/1/2/3 溢出中断
- P0 口引脚电平变化中断
- A/D 中断
- CMP1/2 中断
- PWM1/2 中断
- SPI 中断
- I2C 中断
- DATA EEPROM 写操作(EE)中断
- SSCI 模块 I2C 总线冲突(BCL)中断

在本单片机中有 2 个中断优先级，多个中断源，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 寄存器（见章节 5.1.6）的 IPEN 位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。

中断逻辑如图 5.2 所示，KF8L10Z08 在使用外设中断时需要将外设中断使能位（AIE）置 1 使能外设中断功能。下列中断属于外设中断：

- INT1/2 中断
- T1/2/3 溢出中断
- A/D 中断
- PWM1/2 中断
- SPI 中断
- I2C 中断
- DATA EEPROM 写操作(EE)中断
- SSCI 模块 I2C 总线冲突(BCL)中断

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（INTCTL 寄存器相关见章节 5.1.1）。

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，退出低优先级

中断时置 1 AIEL；进高优先级中断时清 AIEH，退出高优先级中断时置 1 AIEH。（AIEH=0 时禁止所有中断）。如果两级中断同时发生，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，等高优先级中断事件结束后再处理低优先级的中断事件。

其中断优先级工作原理如图 5.1:

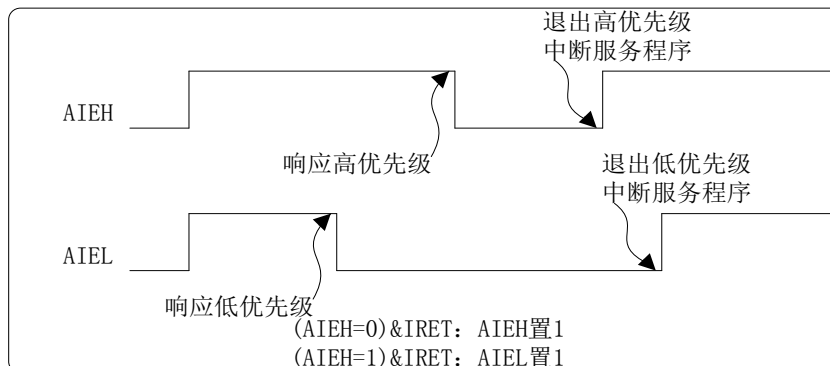


图 5.1 中断优先级工作原理图

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 IP0 和 IP1 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。

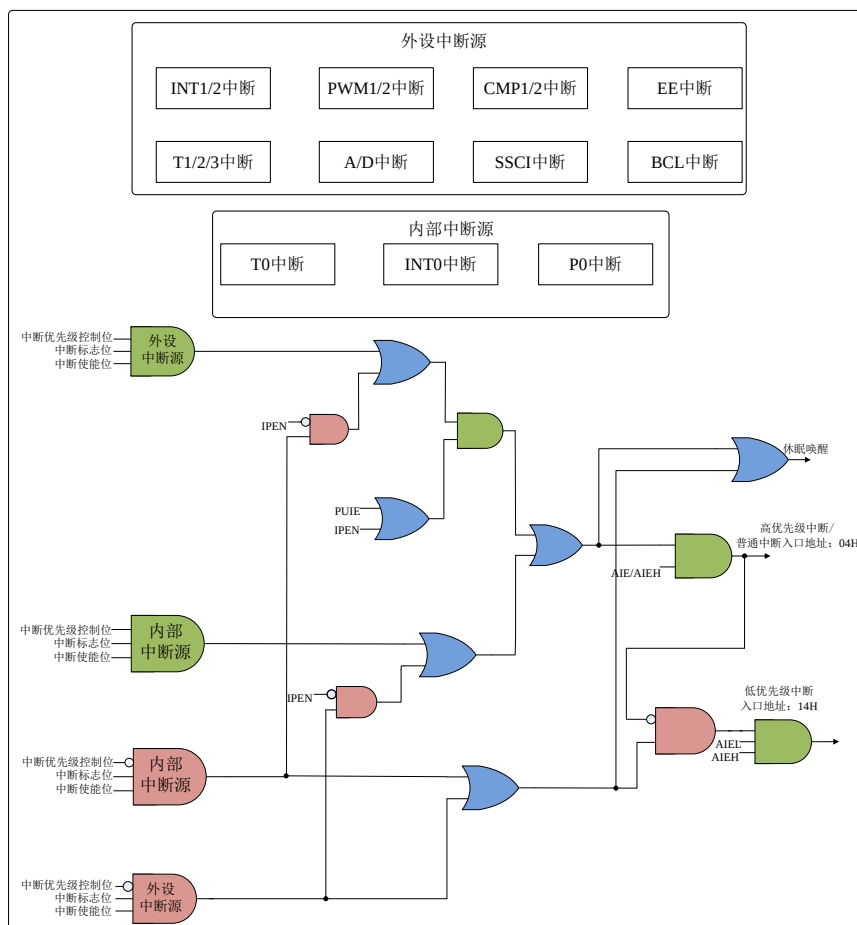


图 5.2 中断逻辑

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	TOIE	INTOIE	POIE	TOIF	INTOIF	POIF
2CH	EIE1	EEIE	ADIE	INT2IE	INT1IE	C1IE	PWM2IE	T2IE	T1IE
2DH	EIE2	T3IE	C2IE	-	-	SSCIE	BCLIE	-	-
0CH	EIF1	EEIF	ADIF	INT2IF	INT1IF	C1IF	PWM2IF	T2IF	T1IF
0DH	EIF2	T3IF	C2IF	-	-	SSCIF	BCLIF	-	-
22H	IP0	-	-	-	-	-	PT0	PINT0	PP0
23H	IP1	PEE	PADC	PINT2	PINT1	PC1	PPWM2	PT2	PT1
24H	IP2	PT3	PC2	-	-	PSSCI	PBCL	-	-
2EH	PCTL	-	-	-	SLVREN	IPEN	SWDTEN	$\overline{\text{POR}}$	$\overline{\text{LVR}}$

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 5.2 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 5.2 所示。

注：1.当中断条件满足时，无论相应的中断使能位或者全局中断使能位AIE的状态如何，中断标志位将被硬件置1。

2.中断条件满足时，中断标志位通过硬件置1,而清零则需要软件完成。

寄存器5.1: INTCTL: 中断控制寄存器(地址: 0BH)

复位值 0000 0000	bit7				bit0			
	AIE/AIEH	PUIE/AIEL	TOIE	INTOIE	POIE	TOIF	INTOIF	POIF
	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

图注: R=可读 W=可写 -=未用 U=未实现位

5.1.2 中断使能寄存器 EIE1

如寄存器 5.2 所示, EIE1 是一个可读写的寄存器, 它包含:

寄存器5.2: EIE1: 中断使能寄存器(地址: 2CH)

复位值 0000 0000	bit7						bit0
	EEIE	ADIE	INT2IE	INT1IE	C1IE	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

C1IE: CMP1 中断使能位

- 1 = 使能 CMP1 中断
0 = 禁止 CMP1 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
- T1IE: T1 中断或 PWM1 中断使能位
1 = 使能 T1 中断或 PWM1 中断
0 = 禁止 T1 中断或 PWM1 中断

图注: R=可读 W=可写 --未用 U=未实现位

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 0000 0000	bit7				bit0			
	T3IE	C2IE	-	-	SSCIE	BCLIE	-	-
	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
- C2IE: CMP2 中断使能位
1 = 使能 CMP2 中断
0 = 禁止 CMP2 中断
- SSCIE: SSCI 中断使能位
1 = 允许 SSCI 中断
0 = 禁止 SSCI 中断
- BCLIE: BCL 中断使能位
1 = 允许 BCL 中断
0 = 禁止 BCL 中断

图注: R=可读 W=可写 --未用 U=未实现位

5.1.4 中断标志寄存器 EIF1

如寄存器 5.3 所示, 中断标志寄存器包含:

寄存器5.3: EIF1: 外设中断标志寄存器(地址: 0CH)

复位值 0000 0000	bit7				bit0			
	EEIF	ADIF	INT2IF	INT1IF	C1IF	PWM2IF	T2IF	T1IF
	R/W	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 完成操作
	0 = 独立 EE 未完成操作
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 引脚产生外部中断
	0 = INT2 引脚未产生外部中断
INT1IF:	INT1 中断标志位
	1 = INT1 引脚产生外部中断
	0 = INT1 引脚未产生外部中断
C1IF:	CMP1 中断标志位
	1 = CMP1 输出结果翻转，产生中断
	0 = CMP1 输出结果未翻转
PWM2IF:	PWM2 中断标志位
	1 = PWM2 使能时，T1H 和 PP2 匹配
	0 = PWM2 使能时，T1H 和 PP2 不匹配
T2IF:	T2 与 PP5 匹配中断标志位
	1 = 发生了 T2 与 PP5 匹配
	0 = 未发生了 T2 与 PP5 匹配
T1IF:	T1 寄存器溢出标志位或 PWM1 中断标志位
	1 = T1 寄存器溢出或 T1L 和 PP1 匹配（PWM1 使能）
	0 = T1 寄存器未溢出或 T1L 和 PP1 不匹配（PWM1 使能）

图注：R=可读 W=可写 -=未用 U=未实现位

5.1.5 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	C2IF	-	-	SSCIIF	BCLIF	-	-
	R/W	R/W	R/W	R/W	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							
C2IF:	CMP2 中断标志位							
	1 = CMP2 输出结果翻转，产生中断							
	0 = CMP2 输出结果未翻转							
SSCIIF:	SSCI 中断标志位							
	1 = SSCI 产生了中断							
	0 = SSCI 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							

图注：R=可读 W=可写 -=未用 U=未实现位

5.1.6 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。如寄存器 5.4 所示，中断优先级控制寄存器 IP0 包含：

寄存器5.4: IP0: 中断优先级控制寄存器0(地址: 22H)

bit7					bit0		
复位值 ---- -000	-	-	-	-	PT0	PINT0	PP0
	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

图注：R=可读 W=可写 -=未用 U=未实现位

5.1.7 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。如寄存器 5.5 所示，中断优先级控制寄存器 IP1 包含：

寄存器5.5: IP1: 中断优先级控制寄存器1(地址: 23H)

bit7				bit0				
复位值 0000 0000	PEE	PADC	PINT2	PINT1	PC1	PPWM2	PT2	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: AD 中断高优先级控制位

1 = AD 中断为高优先级

0 = AD 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

PC1: CMP1 中断高优先级控制位

- 1 = CMP1 中断为高优先级
0 = CMP1 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
- PT1: T1 中断或 PWM1 中断高优先级控制位
1 = T1 中断或 PWM1 中断为高优先级
0 = T1 中断或 PWM1 中断为低优先级

图注: R=可读 W=可写 -=未用 U=未实现位

5.1.8 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。如寄存器 5.8 所示, 中断优先级控制寄存器 IP2 包含:

寄存器5.8: IP2: 中断优先级控制寄存器2(地址: 24H)

		bit7				bit0			
复位值	0000 0000	PT3	PC2	-	-	PSSCI	PBCL	-	-
		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								
PC2:	CMP2 中断高优先级控制位 1 = CMP2 中断为高优先级 0 = CMP2 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								
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								

图注: R=可读 W=可写 -=未用 U=未实现位

5.1.9 电源控制寄存器 PCTL

寄存器5.6: PCTL: 电源控制寄存器(地址: 2EH)

		bit7				bit0			
复位值	---1 000x	-	-	-	SLVREN	IPEN	SWDTEN	POR	LVR
		R/W	R/W	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, 即为优先级模式
0 = 禁止中断优先级功能, 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
当配置字的 WDTEN=1 时
为无关位
- $\overline{\text{POR}}$: 上电复位状态位
1 = 未发生上电复位
0 = 发生了上电复位
- $\overline{\text{LVR}}$: 欠压复位状态位
1 = 未发生欠压复位
0 = 已发生欠压复位

图注: R=可读 W=可写 --未用 U=未实现位

5.1.10 中断响应

当 IPEN=1 时, 为优先级模式, 中断被响应后:

1. 返回地址压入堆栈;
2. 中断入口地址载入 PC;
3. 在中断被响应前, AIEH 和 AIEL 位的设置会影响到中断响应的过程:
 - 将 AIEH 和 AIEL 均置 1, 可进入高优先级或优先级中断。进入的是高或低优先级中断时, AIEH 或 AIEL 由硬件自动清 0, 执行高或低优先级中断服务程序, 执行指令 IRET 退出高或低优先级中断, 硬件自动将 AIEH 或 AIEL 置 1;
 - 只将 AIEH 置 1, 则直接进入高优先级中断, AIEH 由硬件自动清 0, 继续执行中断服务子程序, 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;
4. 跳转到中断发生处继续执行下面的程序。

注: 中断的响应有一定的响应时间:

1. 如果已经进入了低优先级中断, 再有高优先级中断, 此时为中断嵌套;
2. 如果还没有进入低优先级中断, 同时有高优先级中断产生, 则直接进入高优先级中断, 相当于两个中断同时产生。

当 IPEN=0 时, 为普通模式, 中断被响应后:

1. 返回地址压入堆栈;
2. 中断入口地址载入 PC;
3. AIE 位将被硬件清 0 以禁止其它中断;
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, 同时硬件自动将 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 INT 中断对应引脚

APFCTL1 寄存器的 INT1SEL<1:0>位和 INT2SEL<1:0>位可以分别选择 INT1 中断和 INT2 中断的对应引脚位置；INT0/1/2 中断对应的引脚如下表所示：

表 5-2: INT 中断引脚对应表

中断	APFCTL1 寄存器配置	引脚位置
INT0	-	P0.0
INT1	INT1SEL<1:0>=00	P0.1
	INT1SEL<1:0>=01	P1.1
	INT1SEL<1:0>=10	P2.6
	INT1SEL<1:0>=11	P1.4
INT2	INT2SEL<1:0>=00	P1.2
	INT2SEL<1:0>=01	P2.7
	INT2SEL<1:0>=10	P1.5
	INT2SEL<1:0>=11	P1.0

5.2.2 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(见章节 6.1.1)中的 INT0SE 位设置触发边沿，INT0SE 置 1,将 INT0 设置为上升沿触发，清零设置为下降沿触发。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。

5.2.3 INT1 中断

INT1 中断通过寄存器 EIE1 中的 INT1IE 位置 1 使能 INT1 中断。通过 PWMCTL 中的 INT1SE 位(相关寄存器见章节 8.1.2)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。

5.2.4 INT2 中断

INT2 中断通过寄存器 EIE1 中的 INT2IE 位置 1 使能 INT2 中断。通过 PWMCTL 中的 INT2SE 位(相关寄存器见章节 8.1.2)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. 通过 APFCTL1 寄存器的 INTxSEL<1:0>位选择 INTx 对应引脚;
2. 将对应的 INTx 引脚设置为数字输入口;
3. 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。

5.3 定时器中断

下列两个条件任何一个满足时, 定时/计数器 Tx 对应的中断标志位 TxIF 置 1:

- Tx 计数寄存器发生溢出
- Tx 计数寄存器与重载寄存器发生匹配 (如果有重载功能)

通过将 TxIE 位置 1/清 0 可使能/禁止该中断。当 IPEN 和 PTx 置 1 时, 定时器 Tx 中断配置为高优先级中断。

5.4 P0 口中断

P0 口引脚的输入电平变化将使 P0IF(INTCTL.0)位置 1。通过设置/清除 P0IE(INTCTL.3) 位, 可使能/禁止该中断。且该端口各引脚可通过 IOCL 寄存器来对每个引脚进行配置。有关 P0 口中断的操作, 请参考 P0 口电平变化中断控制寄存器的部分。当 IPEN 和 PP0 均置 1 时, P0 口中断配置为高优先级中断。

5.5 PWM 中断

使能 PWM1x/2x 后, T1L 分配给 PWM1x 进行计数, T1H 分配给 PWM2x 进行计数,

当 T1L/H 与 PP1/2 匹配时，会触发相应的中断标志位 T1IF 和 PWM2IF。如果使能 T1IE 或者 PWM2IE，则会触发中断（AIE、PUIE 置 1）。当 IPEN 和 PT1 置 1 时，PWM1 中断配置为高优先级中断；当 IPEN 和 PPWM2 置 1 时，PWM2 中断配置为高优先级中断。

详见 PWM 部分。

5.6 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

KF8L10Z08 单片机提供如下定时/计数器：

1 个 8 位的定时器 T0；

3 个 16 位的定时/计数器 T1/T2/T3。

6.1 定时器 0(T0)

T0 是一个 8 位的定时器，T0 时钟源为系统时钟频率的四分频： $F_{T0}=F_{sys}/4$ ；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，并将 T0 中断标志位 T0IF 置 1，此时如果 T0 中断使能（ $AIE=1 \& T0IE=1$ ），程序将进入中断行。

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。

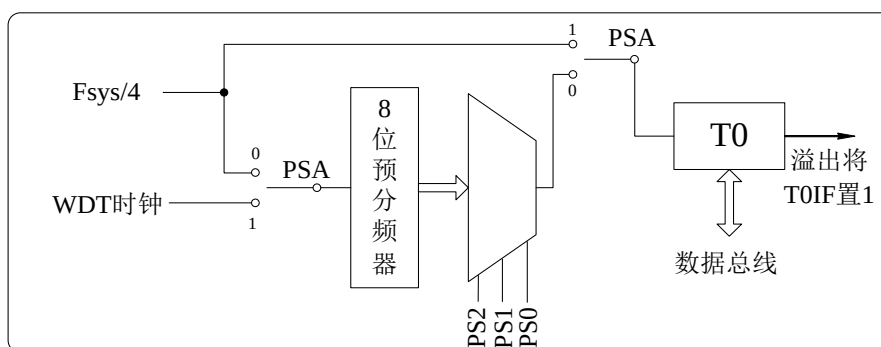


图 6.1 T0 原理框图

6.1.2 T0 相关的寄存器

表 6-1 与 T0 相关的寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
01H	T0	8 位计数器							
21H	OPTR	$\overline{\text{PUPH}}$	INT0SE	T0CS	T0SE	PSA	PS2	PS1	PS0

6.1.3 OPTR 选择寄存器

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

寄存器6.1: OPTR: 选择寄存器(地址: 21H)

复位值	bit7							bit0
1111 1111	$\overline{\text{PUPH}}$	INT0SE	T0CS	T0SE	PSA	PS2	PS1	PS0
	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1

$\overline{\text{PUPH}}$ I/O 口上拉功能总使能位

1 = 禁止 I/O 口所有上拉功能

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 为上升沿触发

0 = INT0/P0.0 为下降沿触发

T0CS: 保留位

T0SE: 保留位

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

注:

1. 请勿将保留位置 1，否则将引起 T0 无法正常工作。
2. 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 CWDIT 指令)。否则，可能引起芯片的异常复位。

6.1.4 T0 的使用

T0 在使用时通过以下步骤进行设置：

1. 如果需要分频，则将预分频器分配给 T0，并设置分频比；
2. 给 T0 寄存器设置初始值；
3. 如果使用中断方式则将 TOIE 和 AIE 位置 1。

6.2 定时/计数器 1(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机器时钟 $F_{sys}/4$, 当 T1 与内部机器时钟一起使用时, 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 $\overline{T1SY}$ 位 (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(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。

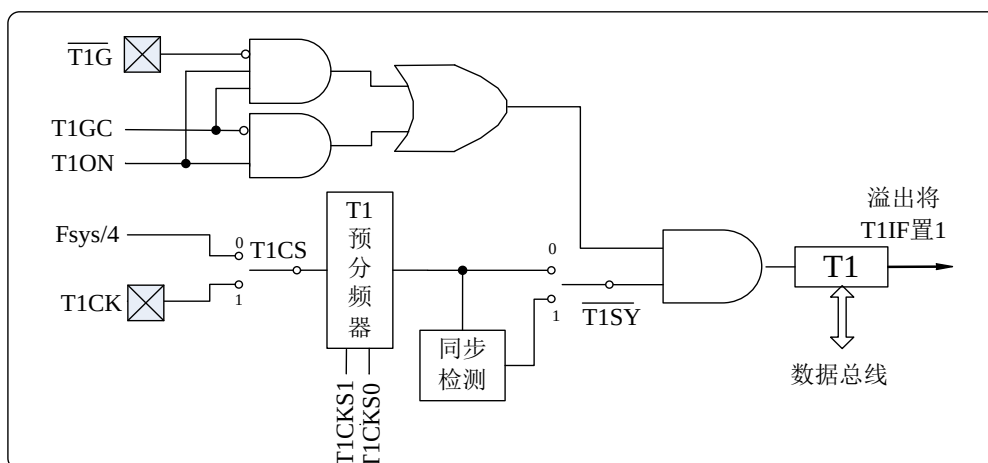


图 6.2 T1 原理框图

6.2.2 T1 相关的寄存器

表 6-2 与 T1 相关的寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
0EH	T1L	T1 低 8 位							
0FH	T1H	T1 高 8 位							
10H	T1CTL	-	T1GC	T1CKS1	T1CKS0	-	$\overline{T1SY}$	T1CS	T1ON

6.2.2.1 T1 控制寄存器

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

寄存器6.2: T1CTL: T1控制寄存器(地址: 10H)

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

T1GC: T1 门控使能位

如果 T1ON=0 则该位被忽略

如果 T1ON=1 则:

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

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

$\overline{T1SY}$: 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 T1CK/P0.5

0 = 定时模式，T1 时钟为机器时钟 Fsys/4

T1ON: T1 启动控制位

1 = 启动 T1

0 = 停止 T1

图注: R=可读 W=可写 -=未用 U=未实现位

6.2.3 定时模式

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

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

6.2.4 计数模式

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

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

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

6.2.5 T1 在休眠模式下的运行

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

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

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

6.2.6 T1 分配给 PWM1/2x

当使用 PWM1x/2x 时需要用到 T1，单片机将 T1L，T1IE，T1IF 分配给 PWM1x，T1H 分配给 PWM2x，具体使用方法参见 PWM1x/2x 部分。

6.3 定时/计数器 2(T2)

T2 是一个 16 位的定时/计数器，T2 的低 8 位在寄存器 T2L 中，高 8 位在寄存器 T2H 中，当 T2 计数值达到 65535 后，T2 的值再加 1 就会产生溢出，将 T2 中断标志位置 1。

T2 带有重载功能。当使能 T2 重载功能时，T2 计数器计数到 T2 重载寄存器中设置的值时，T2 计数器将清 0 重新开始计数，且将 T2 中断标志位置 1。

T2 带有捕捉功能。通过 APFCTL2 寄存器的 CAT2SEL<1:0>位选择 T2 捕捉功能的输入引脚。当检测到 CATCH2 引脚发生上升沿或者下降沿时，T2 计数值 T2H/T2L 将被捕捉到 T2CCPH/T2CCPL 寄存器中，且将 T2 中断标志位置 1。

T2 属于外部单元，因此在使用 T2 中断时，需将 PUIE 位置 1，使能外设中断。

6.3.1 T2 原理框图

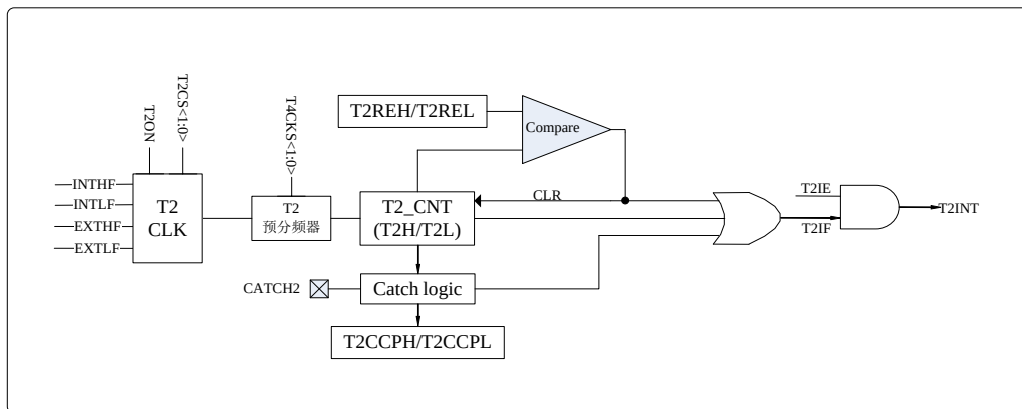


图 6.3 T2 原理框图

6.3.2 T2 相关寄存器

表 6-5 与 T2 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
4BH	T2CTL	T2REN	T2EXTHEN	T2CKS1	T2CKS0	T2EXTLEN	T2CS1	T2CS0	T2ON
58H	APFCTL2	-	-	-	-	CAT3SEL1	CAT3SEL0	CAT2SEL1	CAT2SEL0
5CH	CCPCLR	-	-	-	-	-	-	T3CCPCLR	T2CCPCLR
4AH	T2H	T2 高 8 位							
49H	T2L	T2 低 8 位							
48H	T2REH	T2 重载寄存器高 8 位							
44H	T2REL	T2 重载寄存器低 8 位							
5AH	T2CCPH	T2 捕捉寄存器高 8 位							
57H	T2CCPL	T2 捕捉寄存器低 8 位							

6.3.2.1 T2 控制寄存器

如寄存器 6.5 所示，T2 控制寄存器（T2CTL）用于启动/禁止 T2 以及选择 T2 模块的不同功能特性。

寄存器6.5: T2CTL: T2控制寄存器(地址: 4BH)

复位值 0000 0000	bit7						bit0	
	T2REN	T2EXTHE N	T2CKS1	T2CKS0	T2EXTLE N	T2CS1	T2CS0	T2ON
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

T2REN: T2 重载功能使能位

0 = 禁止 T2 重载功能

1 = 使能 T2 重载功能

T2EXTHEN: 外部高频时钟使能位

1 = 使能外部高频时钟

0 = 未使能外部高频时钟

T2CKS<1:0>: T2 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

T2EXTLEN: 外部低频时钟使能位

1=使能外部低频时钟

0=未使能外部低频时钟

T2CS<1:0>: T2 定时/计数模式选择

00 = T2 时钟为内部高频时钟 INTHF

01 = T2 时钟为内部低频时钟 INTLF

10 = T2 时钟为外部高频时钟 EXTHF

11 = T2 时钟为外部低频时钟 EXTLE

T2ON: T2 启动控制位

1 = 启动 T2

0 = 停止 T2

图注：R=可读 W=可写 -=未用 U=未实现位

6.3.2.3 捕捉数据清零寄存器

寄存器: CCPCLR: 捕捉数据清零寄存器(地址: 5CH)

	bit7						bit0	
复位值 0000 0000	-	-	-	-	-	-	T3CCPCLR	T2CCPCLR
	W	W	W	W	W	W	W	W

- T3CCPCLR: T3 捕捉数据清零位
0 = T3CCPH/T3CCPL 寄存器处于复位状态
1 = T3CCPH/T3CCPL 寄存器退出复位状态
- T2CCPCLR: T2 捕捉数据清零位
0 = T2CCPH/T2CCPL 寄存器处于复位状态
1 = T2CCPH/T2CCPL 寄存器退出复位状态

6.3.3 T2 预分频器

如寄存器 6.5 所示，T2 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。T2CKS 位(T1CTL<5:4>)对预分频计数器进行控制。T2 预分频计数器不能直接进行读写操作，可通过写入 T2H 或 T2L 使预分频计数器清 0。

6.3.4 T2 计数时钟选择

用户可以通过 T2CTL 寄存器的 T2CS<1:0>位来选择 T2 的计数时钟，KF8L10Z08 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。

6.3.5 T2 重载功能

置位 T2CTL 寄存器的 T2REN 位使能 T2 重载功能。通过设置 T2REH/T2REL 寄存器来设置 T2 的重载点。当设置 T2REH/T2REL 寄存器且使能重载功能后，T2 计数器从 0 开始计数，当计数到 T2REH/T2REL 寄存器设置的值时，T2 计数器清 0 并重新开始计数，T2IF 置 1。

T2REN 位置 1 时会载入一次 T2REH/T2REL 寄存器的值，当置位 T2REN 且在 T2 计数过程中对 T2REH/T2REL 寄存器写入新值时，T2 模块会在下一次重载时载入新的重载点。

6.3.6 T2 捕捉功能

通过 APFCTL2 寄存器的 CAT2SEL<1:0>位选择 T2 捕捉功能的输入引脚。CATCHCTL 寄存器的 T2CCP<1:0>位用于选择 T2 捕捉功能的关闭/打开及上升/下降沿发生捕捉事件。当捕捉时间发生时，T2H/T2L 寄存器内的值将被写入 T2CCPH/T2CCPL 寄存器中。

在使用 T2 捕捉功能之前，需要将 CCPCLR 寄存器的 T2CCPCLR 位置 1，将 T2CCPH/T2CCPL 寄存器退出复位状态。由于 T2CCPH/T2CCPL 寄存器为只读寄存器，用户可通过 CCPCLR 寄存器的 T2CCPCLR 位来对 T2CCPH/T2CCPL 寄存器进行清零操作。

6.3.7 T2 中断

在两种情况下会使 T2 中断标志位置 1：

1. 当 T2 计数值溢出；
2. 当 T2 发生重载；
3. 当 P2.6 口发生捕捉事件时。

6.3.8 T2 工作在休眠模式

当选择低频时钟信号作为 T2 的计数时钟时，T2 模块可以工作在休眠模式下。当使能 T2 中断时，系统会在 T2 溢出或者重载时唤醒。

6.4 定时/计数器 3(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 带有捕捉功能。通过 APFCTL2 寄存器的 CAT3SEL<1:0>位选择 T3 捕捉功能的输入引脚。当检测到 CATCH3 引脚发生上升沿或者下降沿时，T3 计数值 T3H/T3L 将被捕捉到 T3CCPH/T3CCPL 寄存器中，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

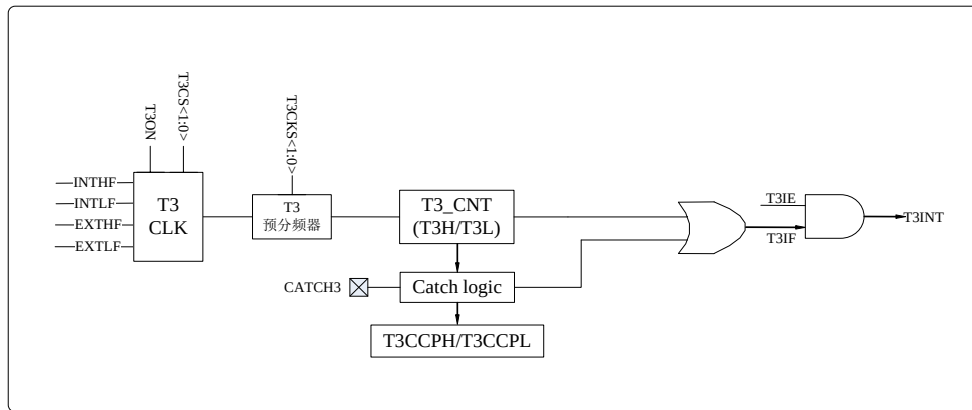


图 6.4 T3 原理框图

6.4.2 T3 相关寄存器

表 6-5 与 T3 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
4EH	T3CTL	-	T3EXTHEN	T3CKS1	T3CKS0	T3EXTLEN	T3CS1	T3CS0	T3ON
54H	T3CTL1	T3CLR	-	-	-	-	-	-	-
58H	CATCHCTL	-	-	-	-	T3CCP1	T3CCP0	T2CCP1	T2CCP0
5CH	CCPCLR	-	-	-	-	-	-	T3CCPCLR	T2CCPCLR
5FH	T3H	T3 高 8 位							
4FH	T3L	T3 低 8 位							
55H	T3CCPH	T3 捕捉寄存器高 8 位							
56H	T3CCPL	T3 捕捉寄存器低 8 位							

6.4.1.1 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)

	bit7						bit0	
复位值	-	T3EXTHE N	T3CKS1	T3CKS0	T3EXTLE N	T3CS1	T3CS0	T3ON
0000 0000	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

T3EXTHE: 外部高频时钟使能位

1 = 使能外部高频时钟

0 = 未使能外部高频时钟

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

T3EXTLE: 外部低频时钟使能位

1=使能外部低频时钟

0=未使能外部低频时钟

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

图注：R=可读 W=可写 -=未用 U=未实现位

6.4.1.2 T3 控制寄存器 1

寄存器6.5: T3CTL1: T3控制寄存器1(地址: 54H)

	bit7						bit0
复位值	T3CLR	保留	保留	保留	保留	保留	保留
0111 1000	R/W	R	R	R	R	R/W	R/W

T3CLR: T3 计数寄存器复位位
0 = T3 计数寄存器 T3H/T3L 处于复位状态
1 = T3 计数寄存器 T3H/T3L 退出复位状态

图注：R=可读 W=可写 -=未用 U=未实现位

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, KF8L10Z08 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 计数寄存器

T3H 和 T3L 分别为 T3 计数器的高 8 位和低 8 位寄存器, 用户可通过读取这两个寄存器以得到当前计数值; 用户还可在使能 T3 计数器之前对 T3H/T3L 寄存器写初值, 也可在使能 T3 计数器之后, 在 T3 计数过程中对 T3H/T3L 寄存器进行写操作以覆盖当前计数值。

T3CTL1 寄存器的 T3CLR 位为 T3 计数寄存器 T3H/T3L 的复位位, 当 T3CLR=0 时, T3H/T3L 寄存器处于复位状态, 寄存器值为复位值 00H, 且无法被写入; 在对 T3H/T3L 寄存器进行写操作之前, 需要将 T3CLR 位置 1; 要使 T3 正常计数, T3CLR 应保持为 1, 否则 T3H/T3L 被复位, 无法正常计数。

6.4.6 T3 捕捉功能

通过 APFCTL2 寄存器的 CAT3SEL<1:0>位选择 T3 捕捉功能的输入引脚。CATCHCTL 寄存器的 T3CCP<1:0>位用于选择 T3 捕捉功能的关闭/打开及上升/下降沿发生捕捉事件。当捕捉时间发生时, T3H/T3L 寄存器内的值将被写入 T3CCPH/T3CCPL 寄存器中。

在使用 T3 捕捉功能之前, 需要将 CCPCLR 寄存器的 T3CCPCLR 位置 1, 将 T3CCPH/T3CCPL 寄存器退出复位状态。由于 T3CCPH/T3CCPL 寄存器为只读寄存器, 用户可通过 CCPCLR 寄存器的 T3CCPCLR 位来对 T3CCPH/T3CCPL 寄存器进行清零操作。

6.4.7 T3 中断

在如下 2 种情况下会使 T3 中断标志位置 1:

1. 当 T3 计数值溢出;
2. T3 发生捕捉事件。

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 溢出时唤醒。

7 模数(A/D)转换模块

模数(A/D)转换模块可将模拟输入信号转换为 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、VDDCORE 或施加在 ADVRIN 引脚上的电压作为转换使用的参考电压。图 7.1 显示了 KF8L10Z08 中 A/D 转换模块的结构框图。

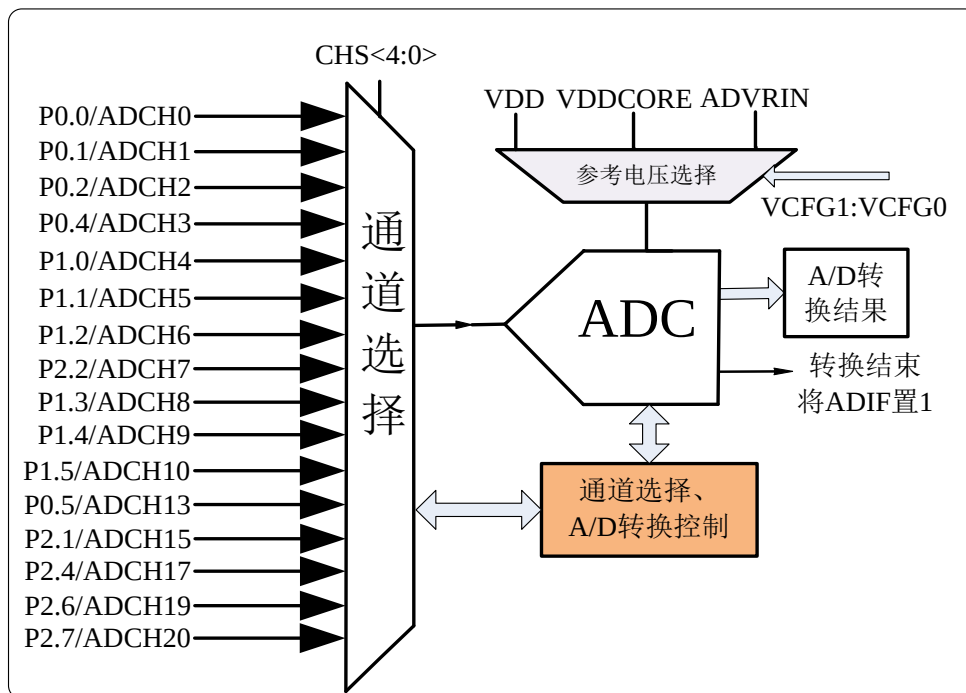


图 7.1 AD 模块结构框图

7.1 AD 相关的寄存器

表 7-1 与 AD 转换相关的寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
1FH	ADCCTL0	ADLR	CHS4	CHS3	CHS2	CHS1	CHS0	START	ADEN
3FH	ADCCTL1	-	ADCS2	ADCS1	ADCS0	VCFG1	VCFG0	ADTEST	ADCIM
41H	ANSE0	-	-	ANS05	ANS04	-	ANS02	ANS01	ANS00
31H	ANSE1	-	-	ANS15	ANS14	ANS13	ANS12	ANS11	ANS10
1EH	ADCDATAH	AD 高 8 位							
3EH	ADCDATAL	AD 低 8 位							

7.1.1 AD 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 0000 0000	bit7						bit0
	ADLR	CHS4	CHS3	CHS2	CHS1	CHS0	ADEN
	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ADLR: A/D 转换结果输出格式选择位

ADLR=1 结果右对齐

ADLR=0 结果左对齐

CHS<4:0>: 模拟通道选择位

00000 = 通道 00(ADCH0/P0.0)

00001 = 通道 01(ADCH1/P0.1)

00010 = 通道 02(ADCH2/P0.2)

00011 = 通道 03(ADCH3/P0.4)

00100 = 通道 04(ADCH4/P1.0)

00101 = 通道 05(ADCH5/P1.1)

00110 = 通道 06(ADCH6/P1.2)

00111 = 通道 07(ADCH7/P2.2)

01000 = 通道 08(ADCH8/P1.3)

01001 = 通道 09(ADCH9/P1.4)

01010 = 通道 10(ADCH10/P1.5)

01011 = 保留

01100 = 保留

01101 = 通道 13(ADCH13/P0.5)

01110 = 保留

01111 = 通道 15(ADCH15/P2.1)

10000 = 通道 16(ADCH16/P1.2)

10001 = 通道 17(ADCH17/P2.4)

10010 = 保留

10011 = 通道 19(ADCH19/P2.6)

10100 = 通道 20(ADCH20/P2.7)

其他 = 系统保留

START: A/D 转换状态位

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

0 = A/D 转换结束或者未进行

ADEN: A/D 工作使能位

1 = 使能 A/D 转换模块工作

0 = A/D 转换器关闭且不消耗工作电流

7.1.2 AD 控制寄存器 1(ADCCTL1)

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

寄存器7.2: ADCCTL1: A/D控制寄存器1(地址: 3FH)

复位值	bit7						bit0	
-000 0000	-	ADCS2	ADCS1	ADCS0	VCFG1	VCFG0	ADTEST	ADCIM
	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ADCS<2:0>: A/D 转换时钟选择位

000 = $F_{ad}=F_{sys}/2$

001 = $F_{ad}=F_{sys}/8$

010 = $F_{ad}=F_{sys}/32$

x11 = 系统保留

100 = $F_{ad}=F_{sys}/4$

101 = $F_{ad}=F_{sys}/16$

110 = $F_{ad}=F_{sys}/64$

VCFG<1:0>: A/D 转换参考电压选择位

00 = ADC 参考电压悬空

01 = VDD 作为 ADC 参考电压

10 = ADVRIN 作为 ADC 参考电压

11 = VDDCORE 作为 ADC 参考电压

ADTEST: 保留位

ADCIM: 保留位，该位复位为 0，使用 AD 时必须将该位置 1。

7.2 通道的选择

如图 7.1 所示，KF8L10Z08 中的 A/D 转换模块的输入可以通过配置 ADCCTL0 寄存器的 CHS<4:0>位进行选择。

7.3 模拟输入口的配置

在选择 IO 口作为 A/D 转换的输入时，需要将对应的引脚配置为模拟输入口。通过将寄存器 ANSEx 的相应位置 1 将对应的引脚配置为模拟口，然后把寄存器 TRx 的对应位置 1 把该引脚配置为输入口，此时该引脚被设置为模拟输入口。

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

7.4 A/D 转换参考电压的选择

KF8L10Z08 中 ADC 模块的参考电压可以选择如下 3 种：

- 电源电压(VDD)
- 内部参考电压(VDDCORE)
- 外部参考电压(ADVRIN)

7.5 转换时钟的选择

完成一次 A/D 转换所需要的时间为 $13T_{ad}$ ($T_{ad}=1/F_{ad}$)。如寄存器 7.2 所示，可通过软件方式设置 ADCS 位(ADCCTL1<6:4>)选择转换时钟源，共有 7 种时钟选项。 T_{ad} 和 F_{ad} 分别为 A/D 转换时钟周期和频率。

7.6 输出格式

KF8L10Z08 中 A/D 转换的结果为 12 位二进制数，A/D 转换结果寄存器为两个 8 位的寄存器。用户可以通过 ADLR(ADCCTL0.7)设置转换结果输出格式，ADLR 置 1 输出为右对齐，ADLR 清 0 输出为左对齐。如图 7.2 所示。

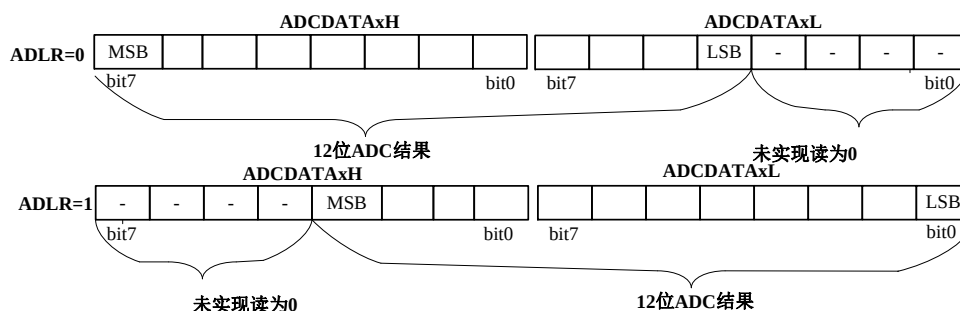


图 7.2 ADC 结构对其方式

7.7 A/D 转换的启动和完成

先将 ADEN 位置 1，然后将 START 位(ADCCTL0<1>)置 1 即可启动 A/D 转换。当转换结束时，A/D 模块将：

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

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

7.8 复位的影响

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

7.9 AD 转换器的设置

启动 A/D 转换器时的设置：

1. 选择 A/D 采样输入通道，设置 A/D 转换结果对齐方式；
2. 将对应的 A/D 采样输入通道设置为模拟输入模式；
3. 将 ADCCTL0 寄存器的 ADEN 位置 1，并且将 ADCCTL1 寄存器的 ADCIM 位置 1，使能 ADC 模块；
4. 选择参考电压和 A/D 采样时钟频率，打开 A/D 转换；
5. 如果采用中断方式，使能 A/D 转换中断；
6. 等待 AD 所需的采集时间；
7. 将 START 位置 1 启动 A/D 转换；
8. 查询 AD 是否转换完成(START=0)或进入 AD 中断；
9. 读取 AD 转换结果。

8 PWM1x/2x 模块

脉宽调制（PWM）是一种通过在完全开启和完全关闭状态之间进行快速切换而为负载供电的方案。PWM 信号类似于方波，信号的高电平部分视为开启状态，信号的低电平部分视为关闭状态。高电平部分（也称为脉宽）可以随时间而变，并以步幅为单位进行定义。施加的步幅数量越多（这会增大脉宽），为负载提供的电量就越多。施加的步幅数量降低时（这会缩短脉宽），提供的电量就会下降。PWM 周期定义为一个完整周期的持续时间，或者开启和关闭时间相加的总时间。

KF8L10Z08 单片机具有 4 路 8 位的 PWM 模块 PWM11/PWM12/PWM21/PWM22。4 路 PWM 结构均相同。

8.1 PWM 原理

启动 PWM 后，在对应的 PWM1x(或 PWM2x)引脚输出 PWM 脉冲。PWM 脉冲的频率和占空比通过 PP1(或 PP2)和 PWM1x(或 PWM2x)设置。

图 8.1 显示了 PWM 逻辑框图。其中 PP1 为 PWM1x 模块的周期寄存器，PWM1xL 为 PWM1x 模块占空比设置寄存器，使用 PWM 时需要将定时器 1 配置给 PWM 做定时用，其中 T1L、T1IE 和 T1IF 分配给 PWM1x，T1H 分配给 PWM2x。启动 PWM1x 后，当 T1L 计数值和 PP1 相等时，PWM 引脚被置 1，此时 T1L 被清 0，重新开始计数，当 T1L 的计数值和 PWM1x 相等时，PWM 引脚清 0(如图 8.2 所示)。改变 PP1 和 PWM1x 的值可产生不同的 PWM1x 周期和 PWM1x 占空比。PWM2x 模块的工作原理和 PWM1x 模块完全一致。

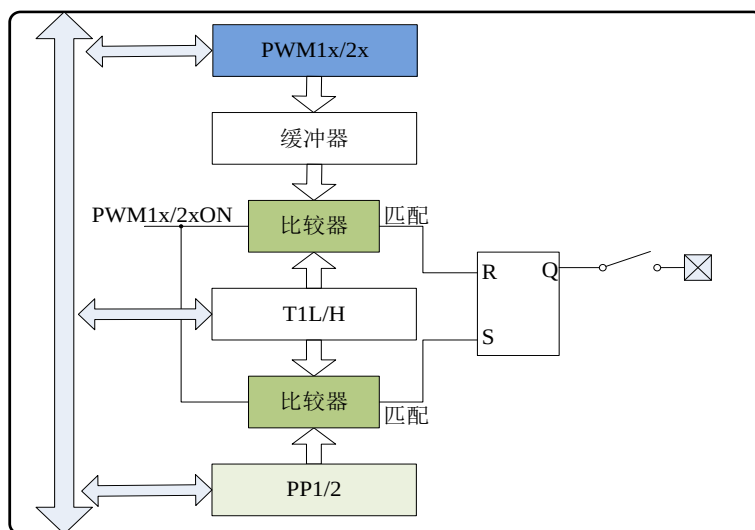


图 8.1 PWM1x/2x 逻辑框图

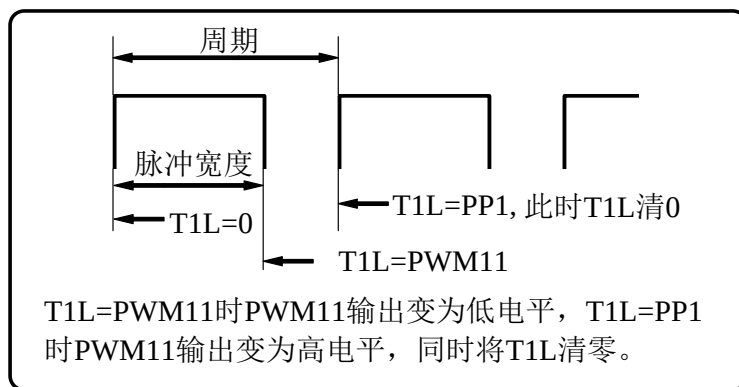


图 8.2 PWM11 输出波形图

8.2 PWM1x/2x 相关的寄存器

表 8-1 与 PWM 相关的寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
5BH	APFCTL0	PWM22 SEL1	PWM22 SEL0	PWM21 SEL1	PWM21 SEL0	PWM12 SEL1	PWM12 SEL0	PWM11 SEL1	PWM11 SEL0
13H	PWM11	PWM11 占空比寄存器							
60H	PWM12	PWM12 占空比寄存器							
16H	PP1	PWM1x 周期寄存器							
32H	PP2	PWM2x 周期寄存器							
33H	PWM21	PWM21 占空比寄存器							
61H	PWM22	PWM22 占空比寄存器							
15H	PWMCTL	INT2SE	INT1SE	-	-	PWM22 ON	PWM12O N	PWM21ON	PWM11ON

8.2.1 PWM1x/2x 控制寄存器

寄存器8.1: PWMCTL: PWM启动控制寄存器 (地址: 15H)

复位值 1100 0000	bit7				bit0			
	INT2SE	INT1SE	保留	保留	PWM22O N	PWM12O N	PWM21O N	PWM11O N
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	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							
PWM22ON:	PWM22 启动控制位							
	1 = 启动 PWM22							
	0 = 禁止 PWM22							
PWM12ON:	PWM12 启动控制位							
	1 = 启动 PWM12							

0 = 禁止 PWM12

PWM21ON: PWM21 启动控制位

1 = 启动 PWM21

0 = 禁止 PWM21

PWM11ON: PWM11 启动控制位

1 = 启动 PWM11

0 = 禁止 PWM11

图注：R=可读 W=可写 -=未用 U=未实现位

注：保留位只能写 0，不能写 1。

8.1 PWM1x/2x 输出引脚

PWM1x/2x 的输出引脚可以通过 APFCTL0 寄存器进行配置选择。

表 8-2 PWM1x/2x 功能引脚对应表

功能	APFCTL0 寄存器配置	对应引脚位置
PWM11	PWM11SEL<1:0>=00	P1.3
	PWM11SEL<1:0>=01	P2.7
	PWM11SEL<1:0>=10	P0.0
	PWM11SEL<1:0>=11	-
PWM12	PWM12SEL<1:0>=00	P1.2
	PWM12SEL<1:0>=01	P0.4
	PWM12SEL<1:0>=10	-
	PWM12SEL<1:0>=11	-
PWM21	PWM21SEL<1:0>=00	P1.1
	PWM21SEL<1:0>=01	P2.2
	PWM21SEL<1:0>=10	P2.4
	PWM21SEL<1:0>=11	-
PWM22	PWM22SEL<1:0>=00	P1.0
	PWM22SEL<1:0>=01	P0.1
	PWM22SEL<1:0>=10	-
	PWM22SEL<1:0>=11	-

8.2 PWM1x/2x 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 8.1 进行计算。

★ 式8.1:
$$\text{PWM周期} = (\text{PPx} + 1) \cdot 4 \cdot \text{Tsys} \cdot (\text{T1预分频比}) \quad (\text{x}=1、2)$$

8.3 PWM1x/2x 占空比

占空比描述开启时间与关闭时间之间以百分比形式表示的比例，0% 代表完全关闭，100% 代表完全开启。占空比越低，对应的供电量就越低；占空比越高，对应的供电量就越高。

PWM 占空比通过 PWM1x/2x(地址: 13H/60H/33H/61H)设置，可写入一个 8 位的值到 PWM1x/2x 来设置占空比。脉冲宽度和占空比通过式 8.2 和式 8.3 计算：

$$\star \text{ 式8.2: } \text{脉冲宽度} = \text{PWMx} \cdot 4 \cdot T_{\text{sys}} \cdot (T1 \text{ 预分频比}) \quad (x=11、12、21、22)$$

$$\star \text{ 式8.3: } \text{占空比} = \frac{\text{脉冲宽度}}{\text{PWM周期}} = \frac{\text{PWMx}}{\text{PPx}+1} \quad (x=11、12、21、22)$$

8.4 PWM1x/2x 分辨率

PWM 分辨率定义可以在单个 PWM 周期中出现的最大步幅数量。分辨率越高，就可以越精确地控制脉宽时间，从而更精确地控制在负载上的供电量。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。KF8L10Z08 中当 PP1/2 为 255 时，PWM 的最大分辨率为 8 位。分辨率的计算公式如式 8.4 所示。

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

8.5 PWM1x/2x 中断

PWM2x 有一个专门的中断使能位 PWM2IE 和中断标志位 PWM2IF，当启动 PWM2 后，会对 PWM2IF 产生影响，而 PWM1x 和定时器 T1 共用中断使能位 T1IE 和中断标志位 T1IF。

在 PWM1x/2x 启用后，当 T1L/H 的计数值与 PWM1xL/2xL 的值匹配后，其对应的输出引脚变为低电平。当 T1L/H 的计数值与 PP1/2 的值匹配后，其对应的输出引脚变为高电平，同时将 T1L/H 清 0，将 T1IF/PWM2IF 置 1，如果允许 T1 或 PWM2 中断，将会转入对应的中断子程序中。

8.6 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。

8.7 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 PWM1x/2x 使用的寄存器进入其复位状态。

8.8 PWM1/2 使用方法

PWM1/2 工作的设置应按照以下步骤:

1. 配置 APFCTL0 寄存器, 选择 PWM_{xx} 的输出引脚。
2. 将 PWM 输出引脚的状态配置为输入, 禁止 IO 口输出。
3. 赋 PP1 或 PP2 寄存器的初值以设置 PWM_{xx} 的 PWM 周期。
4. 赋 PWM_{xx} 寄存器的初值以设置 PWM_{xx} 的占空比。
5. 配置并启动定时器/计数器 T1:
 - 配置 T1CTL 寄存器的 T1CKS1 和 T1CKS0 以选择 T1 的预分频比;
 - 将 T1L/H 清 0;
 - 将 T1CTL 寄存器的 T1ON 位置 1 以启动 T1。
6. 将 PWMCTL 寄存器的 PWM_{xx}ON 置 1 以启动 PWM_{xx}。
7. 将 PWM 输出 IO 口状态配置为数字输出, 输出 PWM 波形。

9 模拟比较器模块

KF8L10Z08 含有 2 路模拟比较器 CMP1 和 CMP2; 2 个比较器分别带有数字滤波功能和中断。

9.1 模拟比较器原理

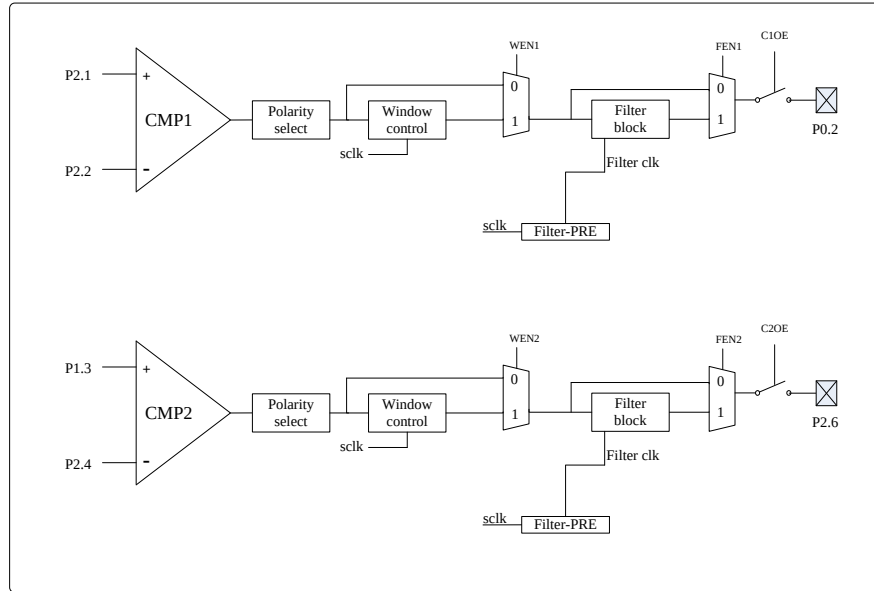


图 9.1 模拟比较器原理框图

9.2 相关寄存器

表 9-1 相关寄存器

地址	寄存器	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
19H	C1CTL	C1OE	C1M	C1P1	C1P0	-	-	C1OUT	C1EN
1AH	C2CTL	C2OE	-	-	-	-	-	C2OUT	C2EN
08H	C1FILTCTL	INV1	WEN1	FEN1	-	-	F1CNT2	F1CNT1	F1CNT0
69H	C2FILTCTL	INV2	WEN2	FEN2	-	-	F2CNT2	F2CNT1	F2CNT0
09H	C1FILTPRE	FP17	FP16	FP15	FP14	FP13	FP12	FP11	FP10
6AH	C2FILTPRE	FP27	FP26	FP25	FP24	FP23	FP22	FP21	FP20
2CH	EIE1	EEIE	ADIE	INT2IE	INT1IE	C1IE	PWM2IE	T2IE	T1IE
2DH	EIE2	T3IE	C2IE	-	-	SSCIE	BCLIE	-	-
0CH	EIF1	EEIF	ADIF	INT2IF	INT1IF	C1IF	PWM2IF	T2IF	T1IF
0DH	EIF2	T3IF	C2IF	-	-	SSCIF	BCLIF	-	-
23H	IP1	PEE	PADC	PINT2	PINT1	PC1	PPWM2	PT2	PT1
24H	IP2	PT3	PC2	-	-	PSSCI	PBCL	-	-

9.3 CMP1 相关寄存器

9.3.1 比较器 1 控制寄存器 C1CTL

寄存器: C1CTL: 比较器1控制寄存器(地址: 19H)

复位值 0000 0000	bit7						bit0	
	C1OE	C1M	C1P1	C1P0	-	-	C1OUT	C1EN
	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W

- C1OE: CMP1 输出使能位
1 = 允许 CMP1 输出到 P0.2 口, 此时需将 P0.2 引脚配置成数字输出端口
0 = 禁止 CMP1 结果输出到 I/O 口
- C1M: 保留位
- C1P<1:0>: 保留位
- C1OUT: CMP1 输出结果状态位
INV1=0 时
1 = C1IN+ > C1IN-
0 = C1IN+ < C1IN-
INV1=1 时
0 = C1IN+ > C1IN-
1 = C1IN+ < C1IN-
- C1EN: 比较器 1 使能位
1 = 使能比较器 1
0 = 关闭比较器 1

图注: R=可读 W=可写 -=未用 U=未实现位

注: 请勿将保留位置 1, 否则将引起比较器无法正常工作。

9.3.2 滤波器 1 控制寄存器 C1FILTCTL

寄存器13.3: C1FILTCTL: 滤波器控制寄存器 (08H)

复位值 0000 0000	bit7						bit0	
	INV1	WEN1	FEN1	-	-	F1CNT2	F1CNT1	F1CNT0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- INV1: 比较器 1 输出极性选择位
0 = 比较器正常输出: C1IN+>C1IN-=1, C1IN+<C1IN-=0
1 = 比较器输出极性转换: C1IN+>C1IN-=0, C1IN+<C1IN-=1
- WEN1: 比较器 1 范围控制使能位
0 = 禁止范围控制功能
1 = 使能范围控制功能
- FEN1: 滤波器 1 使能位
0 = 禁止滤波功能

1 = 使能滤波功能

F1CNT<2:0>:滤波器 1 取样数量选择位

000 = 滤波器 1 不采样, 禁止滤波功能

001 = 滤波器 1 采样 1 次, 并输出结果

010 = 滤波器 1 连续 2 次采样结果一致, 输出采样值, 否则保持;

011 = 滤波器 1 连续 3 次采样结果一致, 输出采样值, 否则保持;

100 = 滤波器 1 连续 4 次采样结果一致, 输出采样值, 否则保持;

101 = 滤波器 1 连续 5 次采样结果一致, 输出采样值, 否则保持;

110 = 滤波器 1 连续 6 次采样结果一致, 输出采样值, 否则保持;

111 = 滤波器 1 连续 7 次采样结果一致, 输出采样值, 否则保持;

图注: R=可读 W=可写 --未用 U=未实现位

注: 每次采样的间隔时间通过采样时钟分频寄存器 C1FILTPRE 设置。

9.3.3 滤波器 1 采样时钟分频寄存器 C1FILTPRE

寄存器13.4: C1FILTPRE: 滤波器采样时钟分频寄存器(地址: 09H)

复位值	bit7						bit0	
0000 0000	FP17	FP16	FP15	FP14	FP13	FP12	FP11	FP10
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

图注: R=可读 W=可写 --未用 U=未实现位

★ 式11.1: 滤波器1采样时钟频率 = $\frac{SCLK}{FP1<7:0>+1}$

注: (1) 当 FP1<7:0>=0 时, 滤波采样时钟即为系统时钟 SCLK。

9.4 CMP2 相关寄存器

9.4.1 比较器 2 控制寄存器 C2CTL

寄存器: C2CTL: C2CTL控制寄存器(地址: 1AH)

复位值	bit7						bit0	
0000 0000	C2OE	-	-	-	-	-	C2OUT	C2EN
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

C2OE: CMP2 输出使能位

1 = 允许 CMP2 输出到 P2.6 口, 此时需将 P2.6 引脚配置成数字输出端口

0 = 禁止 CMP2 结果输出到 I/O 口

C2OUT: CMP2 输出结果状态位

INV2=0 时

1 = C2IN+ > C2IN-

0 = C2IN+ < C2IN-

INV2=1 时

0 = C2IN+ > C2IN-

1 = C2IN+ < C2IN-

C2EN: 比较器 2 使能位

1 = 使能比较器 2

0 = 关闭比较器 2

图注: R=可读 W=可写 -=未用 U=未实现位

9.4.2 滤波器控制寄存器 C2FILTCTL

寄存器13.7: C2FILTCTL: 滤波器控制寄存器 (69H)

复位值 0000 0000	bit7							bit0
	INV2	WEN2	FEN2	-	-	F2CNT2	F2CNT1	F2CNT0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

INV2: 比较器 2 输出极性选择位 comparator1 invert bit

INV2=0, 比较器正常输出: CIN+>CIN-=1, CIN+<CIN-=0

INV2=1, 比较器输出极性转换: CIN+>CIN-=0, CIN+<CIN-=1

WEN2: 比较器 2 范围控制使能位 window control enable bit

WEN2=0, 禁止范围控制功能

WEN2=1, 使能范围控制功能

FEN2: 滤波器 2 使能位 filter block enable bit

FEN2=0, 禁止滤波功能

FEN2=1, 使能滤波功能

F2CNT<2:0>: 滤波器 2 取样数量选择位 filter sample count bits

F2CNT =000, 滤波器 2 不采样, 禁止滤波功能

F2CNT =001, 滤波器 2 采样 1 次, 并输出结果

F2CNT =010, 滤波器 2 连续 2 次采样结果一致, 输出采样值, 否则保持;

F2CNT =011, 滤波器 2 连续 3 次采样结果一致, 输出采样值, 否则保持;

F2CNT =100, 滤波器 2 连续 4 次采样结果一致, 输出采样值, 否则保持;

F2CNT =101, 滤波器 2 连续 5 次采样结果一致, 输出采样值, 否则保持;

F2CNT =110, 滤波器 2 连续 6 次采样结果一致, 输出采样值, 否则保持;

F2CNT =111, 滤波器 2 连续 7 次采样结果一致, 输出采样值, 否则保持;

图注: R=可读 W=可写 -=未用 U=未实现位

9.4.3 滤波器 2 采样时钟分频寄存器 C2FILTPRE

寄存器13.8: C2FILTPRE: 滤波器采样时钟分频寄存器(地址: 6AH)

复位值 0000 0000	bit7							bit0
	FP27	FP26	FP25	FP24	FP23	FP22	FP21	FP20
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

图注: R=可读 W=可写 -=未用 U=未实现位

★ 式11.2: 滤波器2采样时钟频率 = $\frac{SCLK}{FP2<7:0>+1}$

注：（1）当 $FP2<7:0>=0$ 时，滤波采样时钟即为系统时钟 SCLK。

9.5 极性选择

模拟比较器的输出状态可以通过 CxFILTCTL 寄存器的 INVx 位结果选择，以比较器 1 为例，如表 11-2 所示。INV1=0, 比较器正常输出: $C1IN+>C1IN-=1$, $C1IN+<C1IN-=0$; INV1=1, 比较器输出极性转换: $C1IN+>C1IN-=0$, $C1IN+<C1IN-=1$ 。

表 13-2 比较器 1 输出状态与输入状态

输入条件	极性选择	比较器输出
$C1IN+ > C1IN-$	INV1 = 0	C1OUT = 1
$C1IN+ < C1IN-$	INV1 = 0	C1OUT = 0
$C1IN+ > C1IN-$	INV1 = 1	C1OUT = 0
$C1IN+ < C1IN-$	INV1 = 1	C1OUT = 1

9.6 范围控制功能

范围控制功能是利用系统时钟作为筛选时钟，当时钟电平为高时，比较器结果输出；当时钟电平为低时，比较器结果保持。通过 CxFILTCTL 寄存器的 WENx 位来使能/禁止范围控制功能。

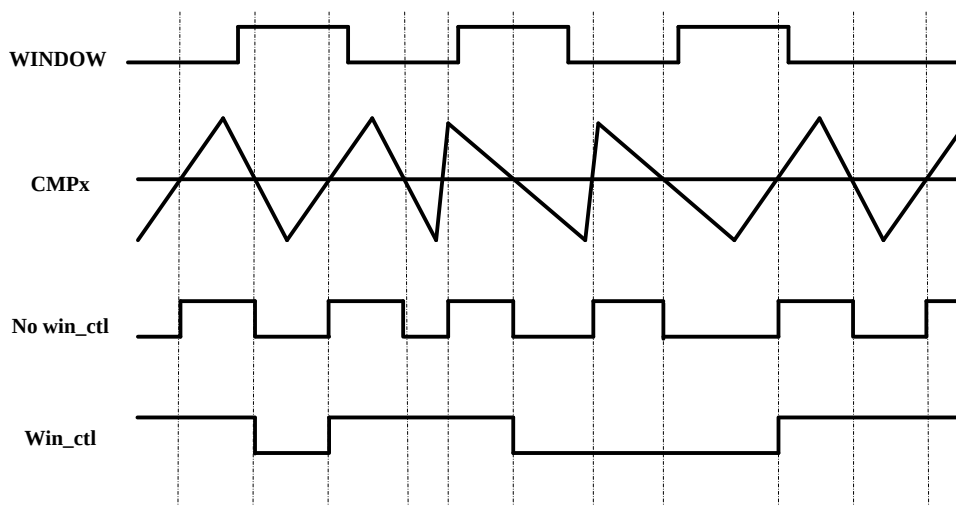


图 13.2 模拟比较器范围控制

9.7 滤波功能

当比较器的输入电压的压差很小时，比较器的输出会发生震荡。内部和外部的寄生效应，和信号线、电源线与其它系统部分间的信号耦合造成了比较器的输出震荡。输出震荡降低了比较器的精度，影响了比较输出的结果。通过 CxFILTCTL 寄存器的 FENx 位选择将比较器的输出通过滤波器，可以减少比较器震荡带来的误差。

滤波器使用方法：

- 通过 CxFILTCTL 寄存器的 FENx 位来使能滤波功能；
- 通过 $FPx<7:0>$ 位选择滤波器采样时钟；
- 通过 $FxCNT<2:0>$ 位设置滤波器的采样次数。

9.8 比较器中断

当比较器的输出值发生变化时，相应的比较器中断标志位就会置 1，此时将产生中断。用户如果使用中断需要打开外设中断使能位 PUIE、全局中断使能位 AIE 以及相应的比较器中断使能位 CxIE；将 PCx 位置 1，还可将比较器中断设置成高优先级中断。

注：用户在清零比较器中断标志位（C1IF/C2IF）之前，必须先读取相应比较器的输出位 CxOUT，再将中断标志位清 0，如例 13.1 所示。

例 13.1：进入比较器 1 中断后的操作

```
MOV R0 , C1CTL  
CLR EIF1 , C1IF
```

9.9 复位的影响

当器件复位时，强制将寄存器 CxCTL 置为复位状态，这使比较器和电阻分压参考电压被“强制”设置为关闭状态。

10 SSCI 模块

10.1 概述

KF8L10Z08 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

10.1.1 SSCI 模块功能引脚

SSCI 模块相关的引脚功能如下:

- SDI/SDA
- SCK/SCL
- SDO
- SS

其中, SDA 和 SCL 为 I2C 的数据与时钟引脚功能:

- SDA: I2C 串行数据信号
- SCL: I2C 串行时钟信号

SPI 相关引脚功能如下:

- SDO: SPI 的串行数据输出信号
- SDI: SPI 的串行数据输入信号
- SCK: SPI 的串行时钟
- $\overline{SS}$: SPI 的从动选择信号

用户可通过 APFCTL1 寄存器的 SSCISEL 位和 SSSEL 位来选择 SSCI 相关功能引脚的位置, 详细如下表所示:

表 10-0 SSCI 模块功能引脚对应表

功能	寄存器配置	对应引脚位置
SDI/SDA	APFCTL1_SSCISEL=0	P0.0
	APFCTL1_SSCISEL=1	P1.0
SCK/SCL	APFCTL1_SSCISEL=0	P0.1
	APFCTL1_SSCISEL=1	P0.2
SDO	-	P1.4
SS	APFCTL1_SSSEL=0	P0.4
	APFCTL1_SSSEL=1	P2.2

10.2 SSCI 相关寄存器

表 10-1 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
66H	SSCICTL0	SSCIWCFL	SSCIOV	SSCIEN	SSCICKP	SSCIMOD3	SSCIMOD2	SSCIMOD1	SSCIMOD0
67H	SSCICTL1	SSCICALLN	SSCIACKSTA	SSCIACKDAT	SSCIACKEN	SSCIRCEN	STOPEN	RESTARTEN	STARTEN
65H	SSCISTA	SAMPLE	CKEKE	SSCIDA	SSCISTOP	SSCISTART	SSCIRW	SSCIUA	SSCIBUF
68H	SSCIBUFR	SSCI 数据接收缓冲/发送寄存器							
64H	SSCIADD	SSCI 的 I2C 地址寄存器							
	SSCIMSK	SSCIMSK7	SSCIMSK6	SSCIMSK5	SSCIMSK4	SSCIMSK3	SSCIMSK2	SSCIMSK1	SSCIMSK0
40H	APFCTL1	-	-	SSCI SEL	SSSEL	INT2 SEL1	INT2 SEL0	INT1 SEL1	INT1 SEL0

10.2.1 SSCI 控制寄存器 0 (SSCICTL0)

寄存器12.1: SSCICTL0: SSCI控制寄存器0(地址:66H)

bit7								bit0	
复位值	SSCIWCFL	SSCIOV	SSCIEN	SSCICKP	SSCIMOD3	SSCIMOD2	SSCIMOD1	SSCIMOD0	
0000 0000	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。 在主动模式下, 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 = 保留

0100 = SPI 从动模式, 时钟 = SCK 引脚。使能 $\overline{SS}$ 引脚控制。

0101 = SPI 从动模式, 时钟 = SCK 引脚。禁止 $\overline{SS}$ 引脚控制。 $\overline{SS}$ 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=SCLK/(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

图注: R=可读 W=可写 -=未用 U=未实现位

10.2.2 SSCI 控制寄存器 1 (SSCICTL1)

寄存器12.2: SSCICTL1: SSCI控制寄存器1(地址:67H)

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

在发送模式模式下:

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）

在接收模式模式下: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）

在接收模式模式下:

1 = 在SDA 和SCL 引脚启动应答序列，发送SSCIACKDAT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）

在发送模式模式下:

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

图注: R=可读 W=可写 --未用 U=未实现位

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)。

10.2.3 SSCI 状态寄存器 (SSCISTA)

寄存器12.3: SSCISTA: SSCI状态寄存器(地址:65H)

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

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。

I2C主模式下

1 = 读

0 = 写

I2C从模式下

1 = 发送正在进行

0 = 发送未进行

该位与STARTEN、RESTARTEN、STOPEN、SSCIRCEN或SSCIACKEN位的或运算结果指示SSCI是否处于空闲状态。

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

图注: R=可读 W=可写 -=未用 U=未实现位

10.2.4 SSCI屏蔽寄存器（SSCIMSK）

寄存器12.4: SSCIMSK: SSCI屏蔽寄存器(地址:64H)

	bit7							bit0
复位值	SSCIMSK	SSCIMSK	SSCIMSK	SSCIMSK	SSCIMSK	SSCIMSK	SSCIMSK	SSCIMSK
1111 1111	7	6	5	4	3	2	1	0
	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

图注: R=可读 W=可写 -=未用 U=未实现位

注:当SSCICTL0位SSCIMOD<3:0>=1001时，不能对SSCIADD寄存器进行读和写，任何对SSCIADD寄存器（地址也是12EH）的读和写操作均是对SSCIMSK寄存器操作。

10.2.5 SSCI I2C 地址寄存器 (SSCIADD)

在10位I2C从动模式下，该地址寄存器是复用的。

寄存器12.5: SSCIADD: I2C地址寄存器(地址:64H)

	bit7							bit0
复位值	SSCIADD	SSCIADD	SSCIADD	SSCIADD	SSCIADD	SSCIADD	SSCIADD	SSCIADD
0000 0000	7	6	5	4	3	2	1	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

10位从动模式下——高地址字节:

SSCIADD<7:3>: 未使用，SSCIADD存放高地址字节时，未使用这5位，为无关位。主器件发送的位模式由I2C规范制定必须等于11110，但是这这些位由硬件进行比较且不受该寄存器中的值的影响

SSCIADD<2:1>: 保存10位地址的高两位

SSCIADD0: 未使用，为无关位，初始化时写0

10位从动模式下——低地址字节:

SSCIADD<7:0>: 10位地址的低8位

7位从动模式下:

SSCIADD<7:1>: 7位地址

SSCIADD0: 未使用，为无关位，初始化时写0

图注: R=可读 W=可写 -=未用 U=未实现位

注: (1)在I2C主控模式下，波特率计算公式= $SCLK/(4*(SSCIADD+1))$;
(2)在I2C模式下，不支持SSCIADD寄存器的值为0、1或2的情况。

10.3 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

10.3.1 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（SSCICTL0<5>）置1以使能SSCI模块的功能。

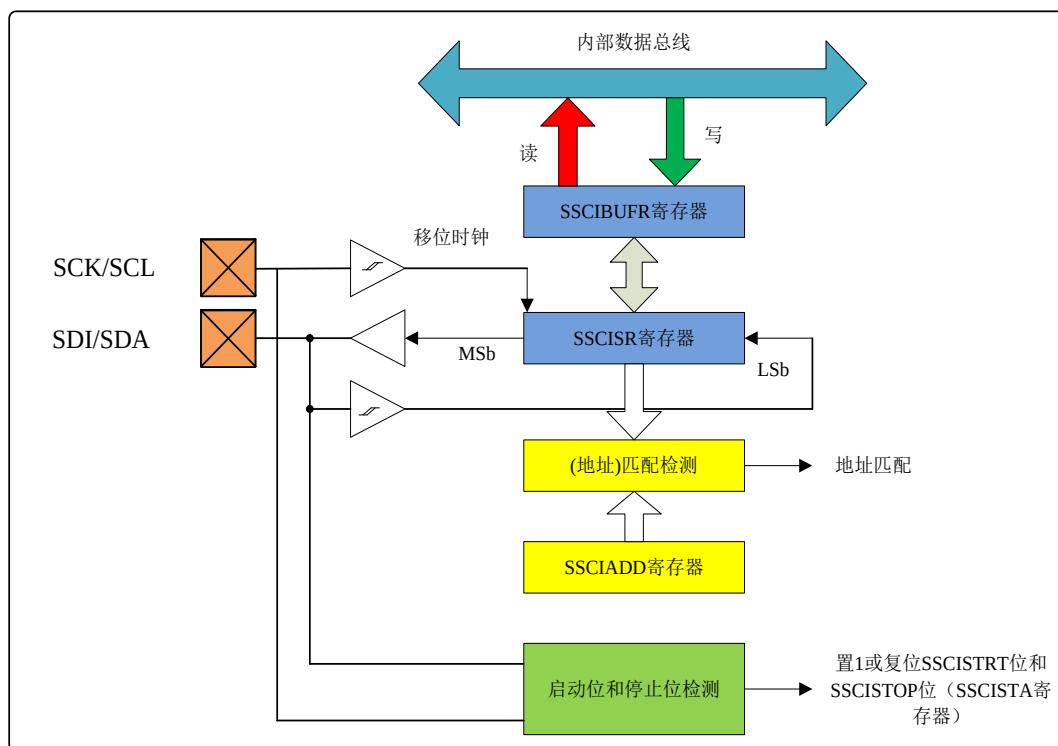


图10.1 I2C模式方框图

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（SSCICTL0）
- SSCI控制寄存器1（SSCICTL1）
- 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）

SSCICTL0 寄存器用于控制I2C 的工作。可通过设置四个模式选择位（SSCICTL0<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。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。

10.3.2 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（SSCCTL0<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。表12-2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 10-2 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。

10.3.2.1 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10.3）。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。

10.3.2.2 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（SSCICTL0<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。

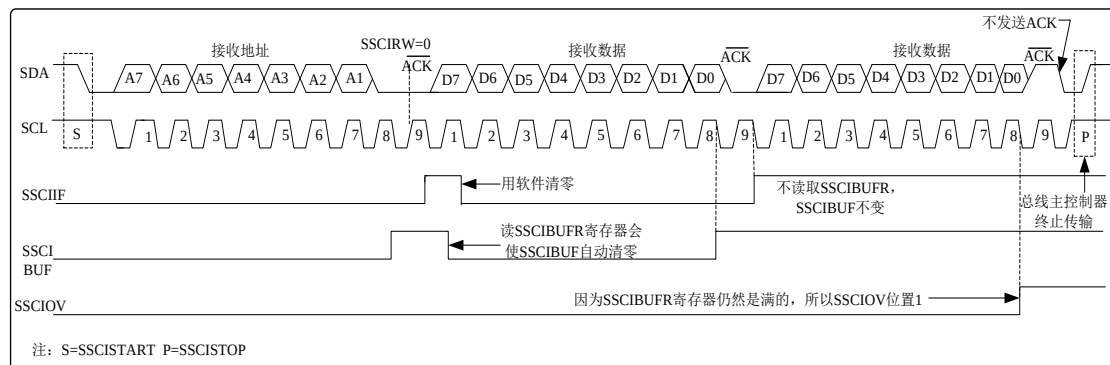


图10.2 从动模式时序（接收，7位地址）

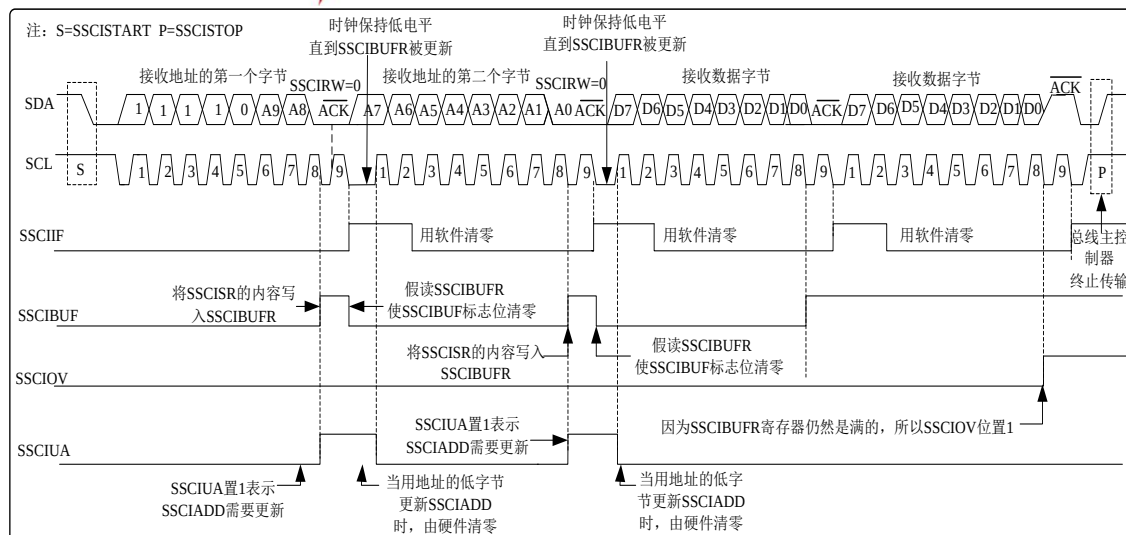


图 10.3 从动模式时序（接收，10 位地址）

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中断使能位；
- 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 $\overline{ACK}$ 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。

10.3.2.3 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（SSCICTL0<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（SSCICTL0<4>）置1使能SCL引脚。

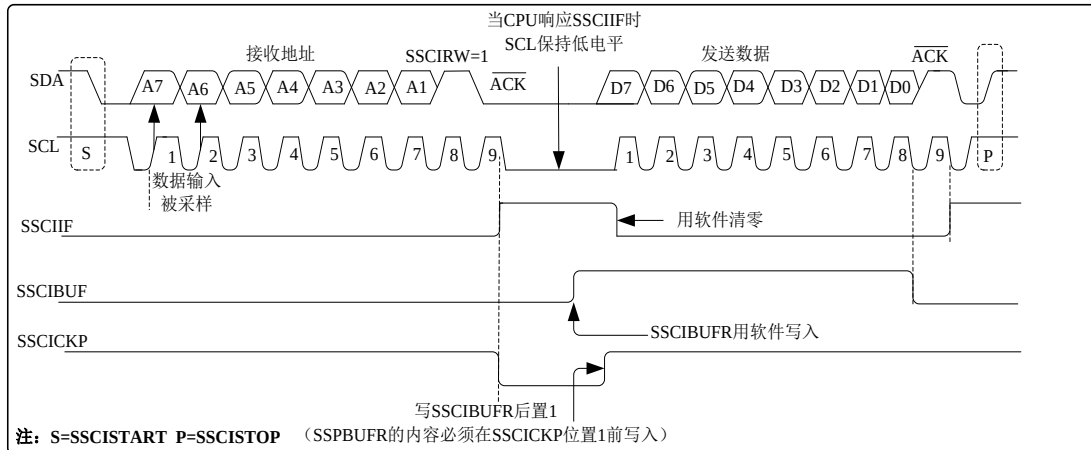


图 10.4 从动模式时序（发送，7 位地址）

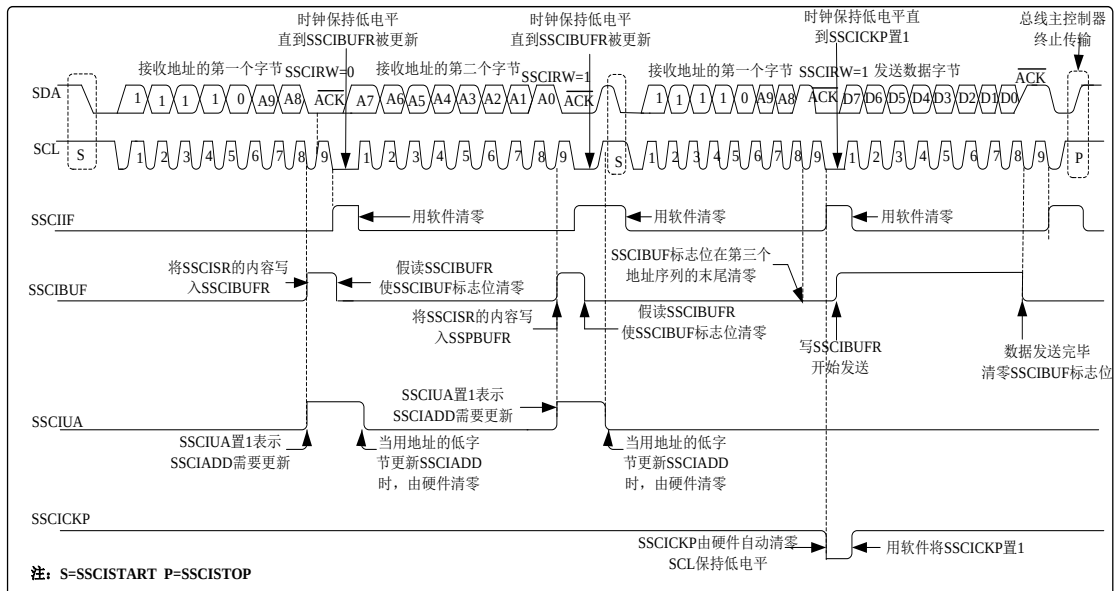


图 10.5 从动模式时序（发送，10 位地址）

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SSCIF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予主控制器同步。

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（SSCICTL0<4>）置1使能SCL引脚。

10.3.2.4 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（SSCICTL1<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10.6所示。

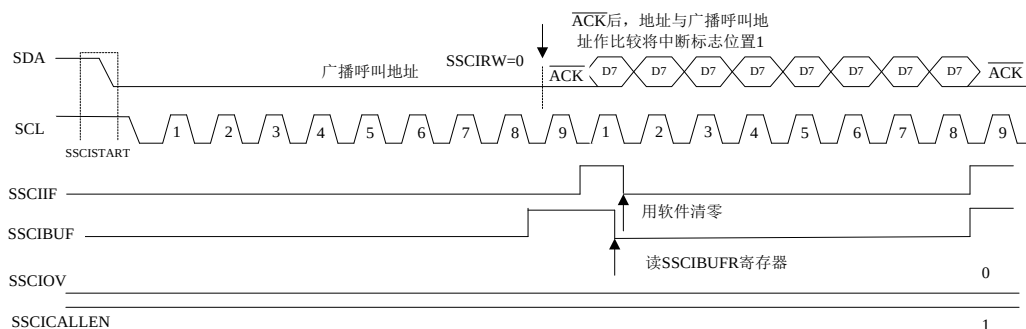


图10.6 从动模式广播呼叫地址时序（7或10位地址模式）

10.3.3 I2C 主控模式

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TR0<1:0>位来控制。输出电平始终为低电平，而与P0.<1:0>的值无关。因此当发送数据时，对于SDA线，必须将TR00置0（输出），对于SCL线，也要将TR0<1>位置0（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。

10.3.3.1 主控模式支持

通过设置SSCICTL0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。图10.7 为I2C主模式框图。

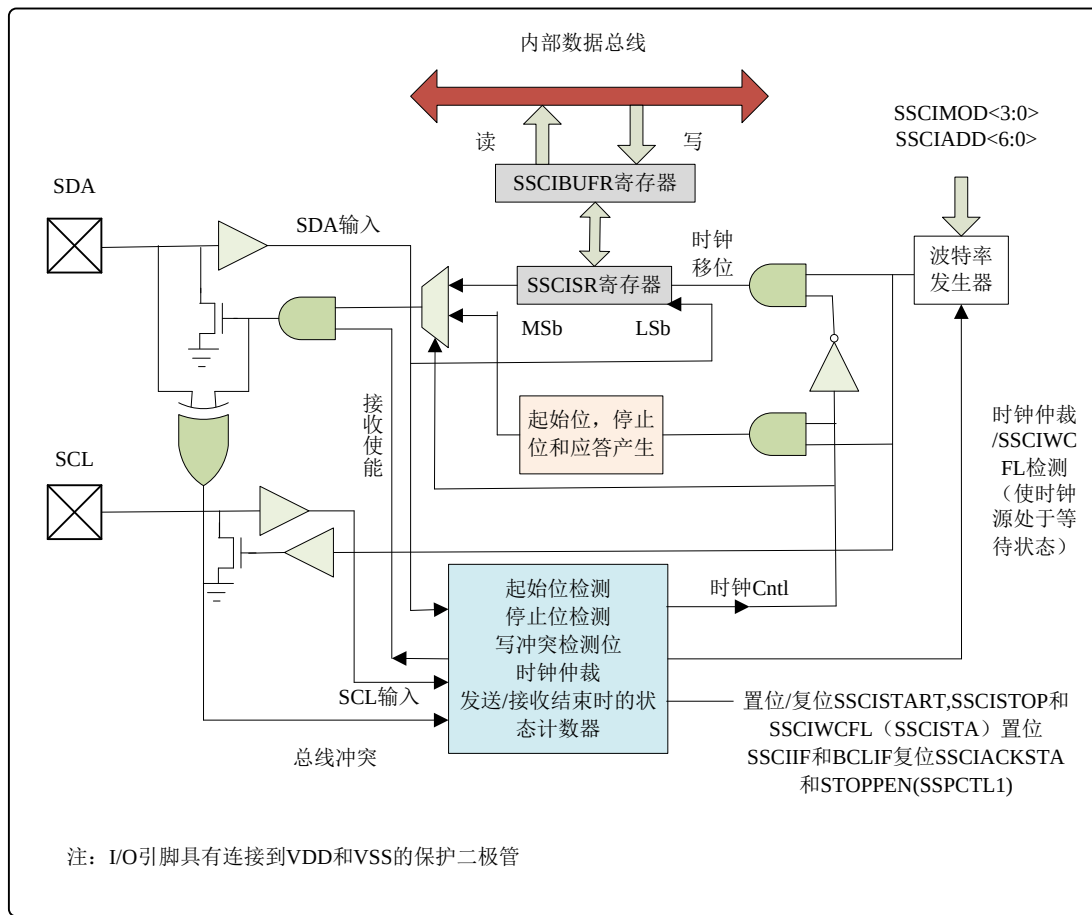


图10.7 I2C主模式框图

10.3.3.2 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。在主机发送器模式下，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、400kHz或1MHz。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 (SSCICTL1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SSCICTL1 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SSCICTL1 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（SSCICTL1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。

10.3.3.3 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10.8 和图10.9 所示。

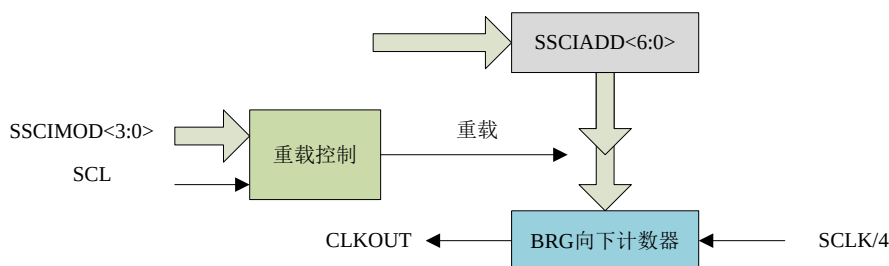


图 10.8 波特率发生器框图

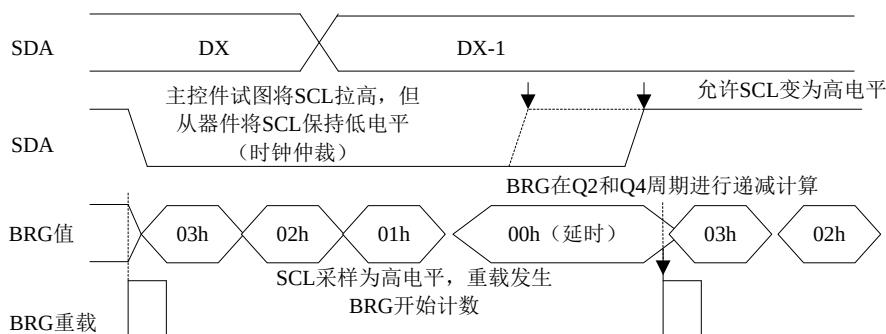


图 10.9 带有时钟仲裁的波特率发生器时序

10.3.3.4 I2C 主控模式启动条件时序

要发起启动条件，用户应将SSCICTL1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（SSCISTA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，SSCICTL1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SSCICTL1 的低5 位进行写操作。

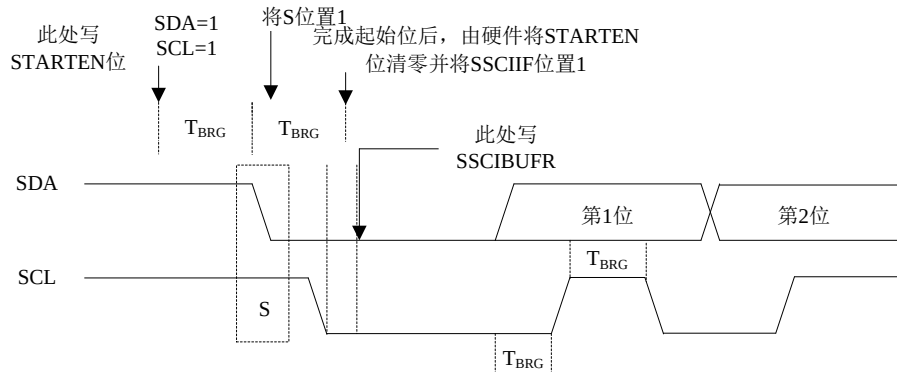


图10.10 第一个启动位时序

10.3.3.5 I2C 主控模式重复启动条件时序

将RESTARTEN位（SSCICTL1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位置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（SSCICTL1 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，SSCIIF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SSCIIF 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SSCICTL1 的低5位进行写操作。

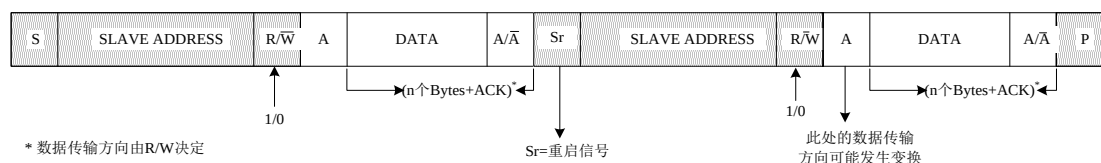


图10.11 I2C协议复合数据帧格式

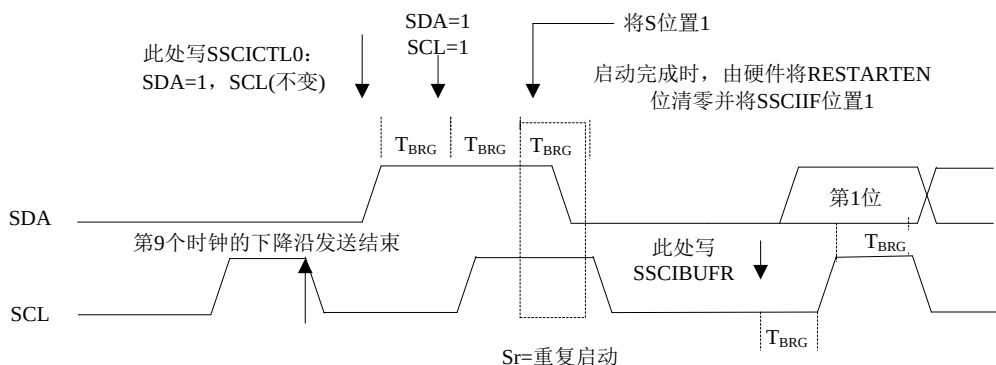


图10.12 重复启动条件时序波形

10.3.3.6 I2C 主控模式发送

I2C 协议中 SDI 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。图 10.13 所示是 I2C 协议典型写数据帧格式。

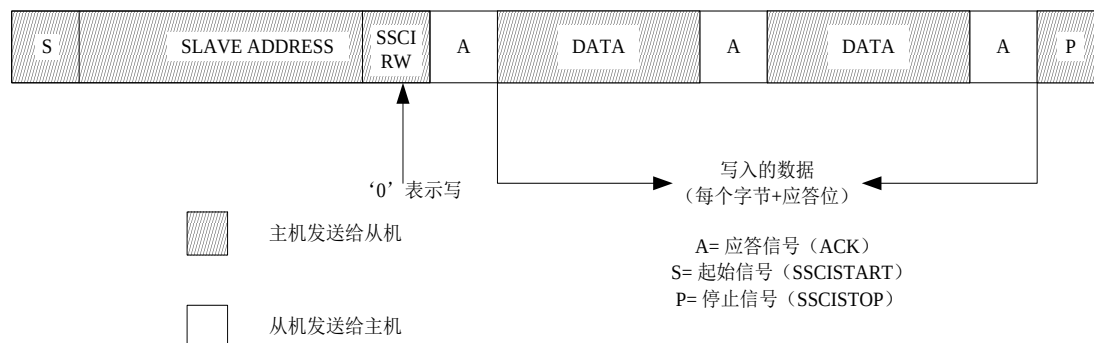


图 10.13 I2C 协议典型写数据帧格式

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 (SSCICTL1 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 (SSCISTA 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）。

图 10.14 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。

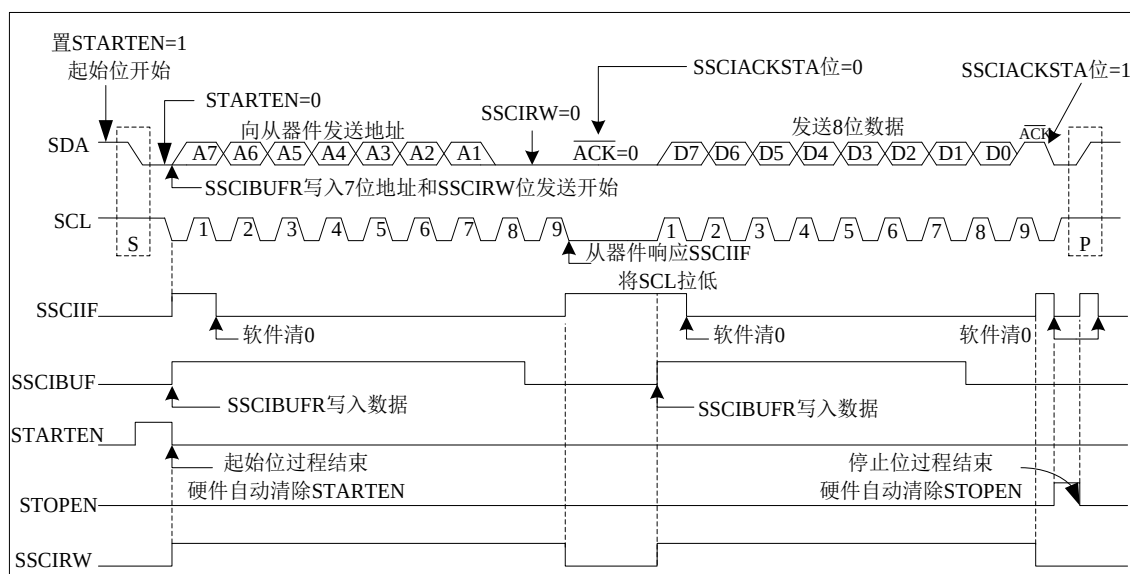


图 10.14 硬件 I2C 主模式发送数据时序图 (7 位地址)

10.3.3.7 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。如图10.15为I2C协议典型读数据帧格式。

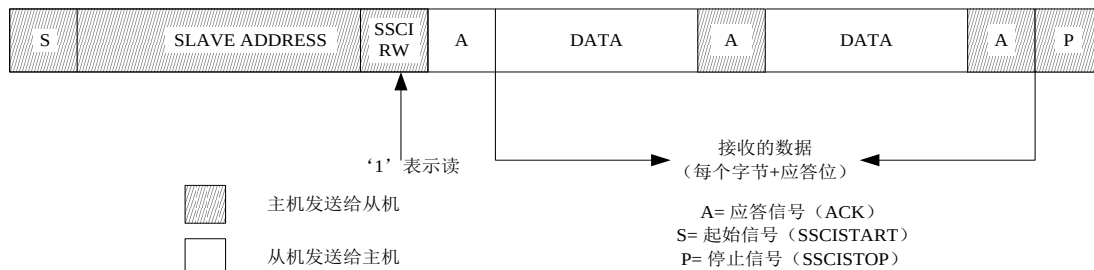


图10.15 I2C协议典型读数据帧格式

通过编程接收使能位SSCIRCEN（SSCICTL1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（SSCICTL1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SSCIBF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)

图10.16 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。

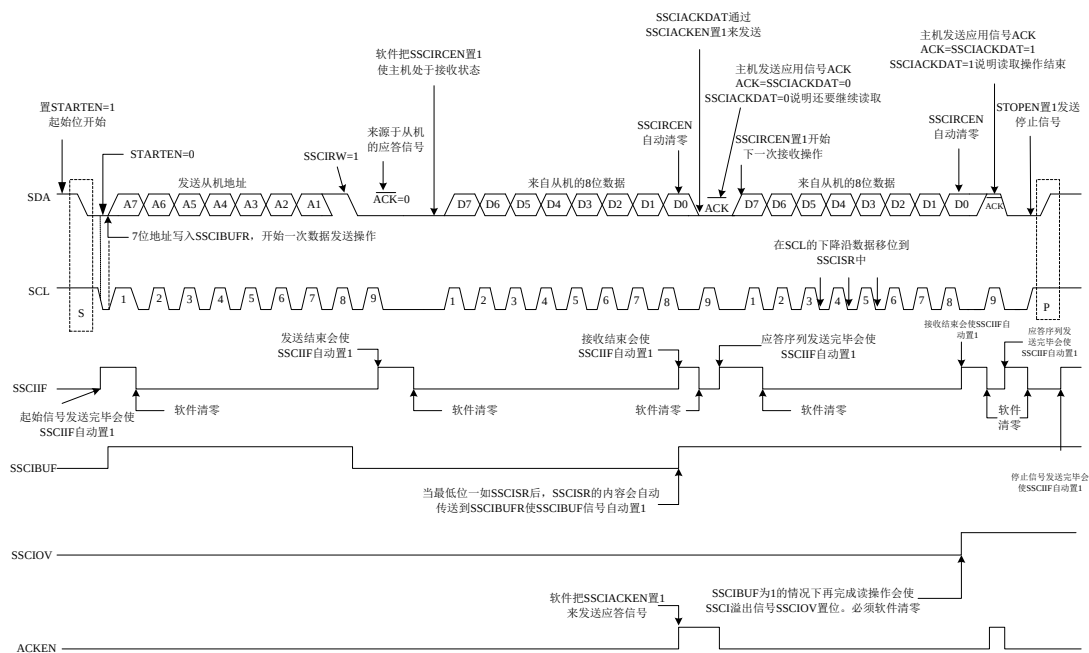


图10.16 I2C主模式接收数据时序图（7位地址）

10.3.3.8 应答序列时序

将应答序列使能位SSCIACKEN（SSCICTL1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

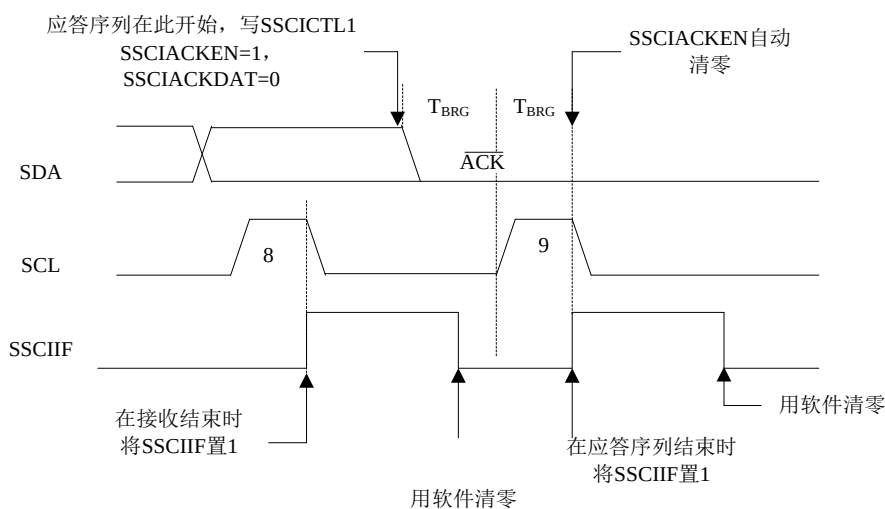
如果用户在应答序列正在进行时写SSCIBUFR，SSCIWCFL 将被置1 且缓冲器的内容保持不变（未发生写操作）。

10.3.3.9 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（SSCICTL1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，主控制器将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后，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（SSCISTA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）。



注：T_{BRG} = 一个波特率发生器周期

图10.17 应答序列时序波形

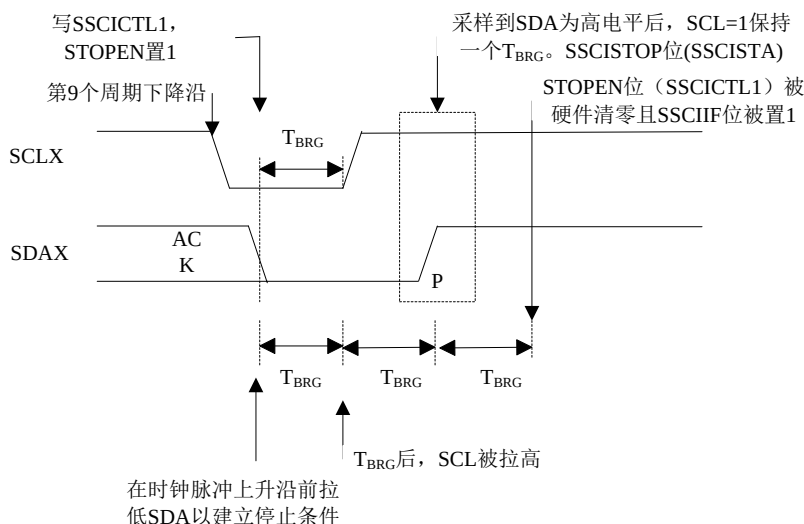


图10.18 停止条件接收或发送模式

10.3.3.10 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。

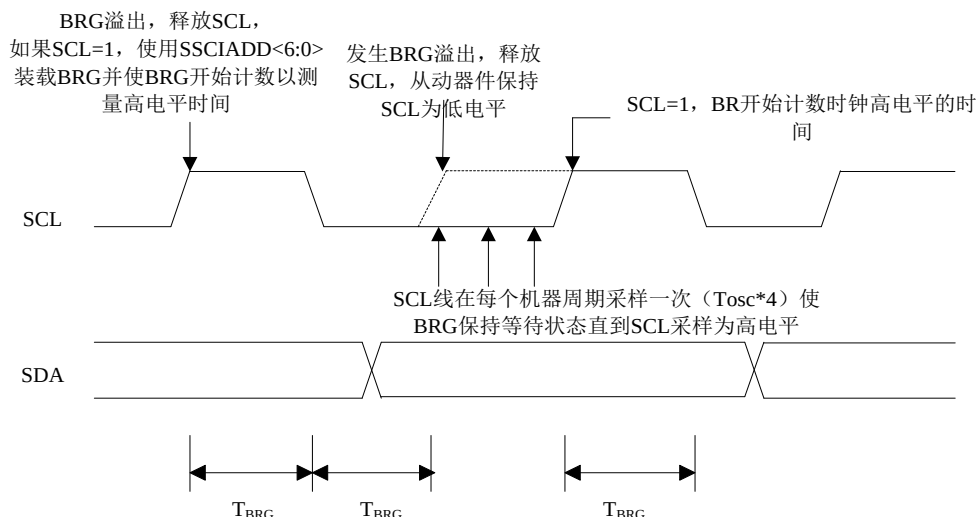


图 10.19 主控发送模式下的时钟仲裁时序

10.3.4 多主控制器模式

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（TR1<1:0> 位置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。

10.3.4.1 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，SSCCTL1 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，SSCIF 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。

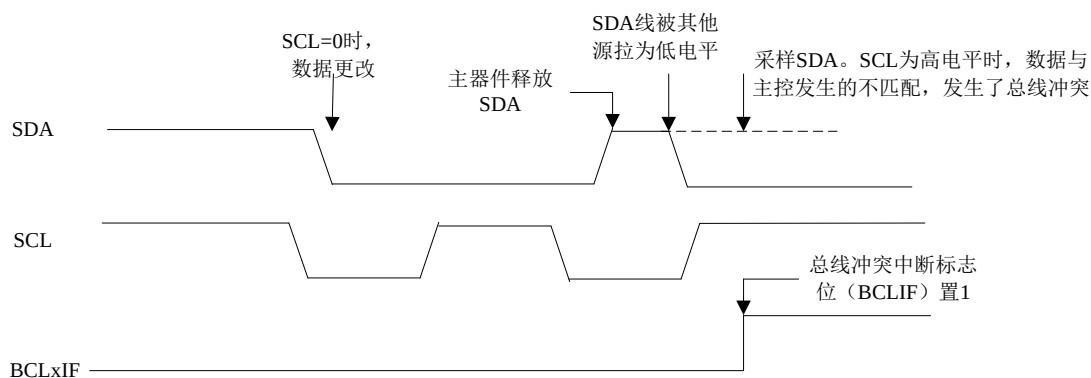


图 10.20 发送和应答时的总线冲突时序

10.3.4.2 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 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。

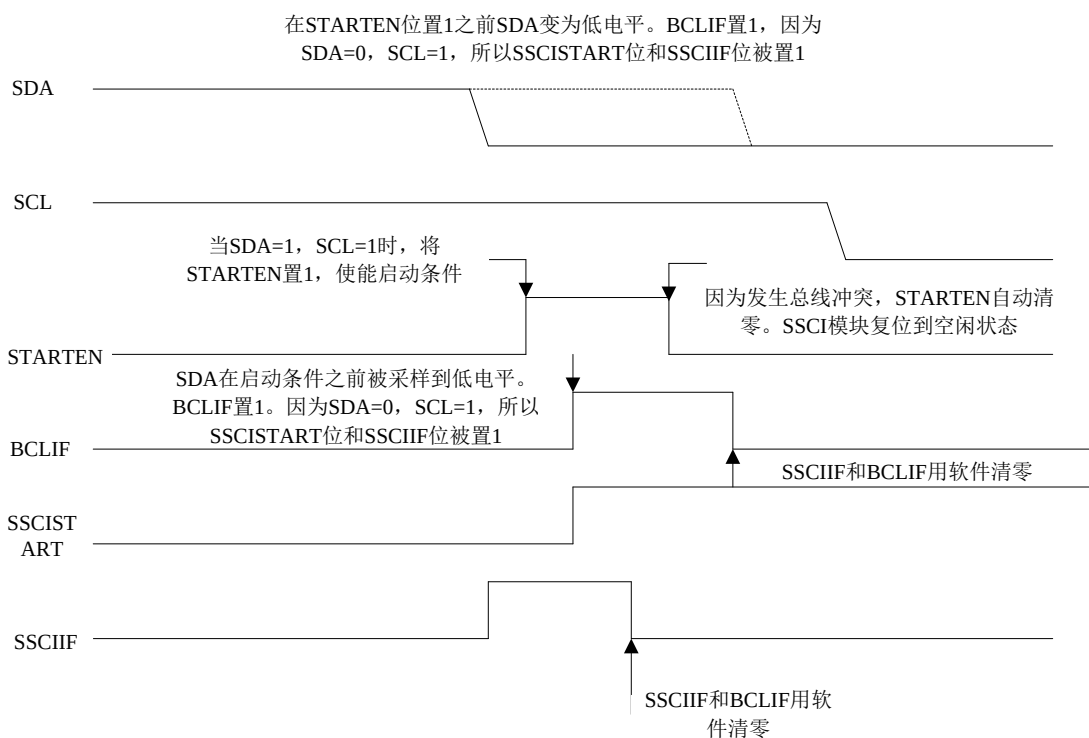


图10.21 启动条件期间的总线冲突（仅SDA）

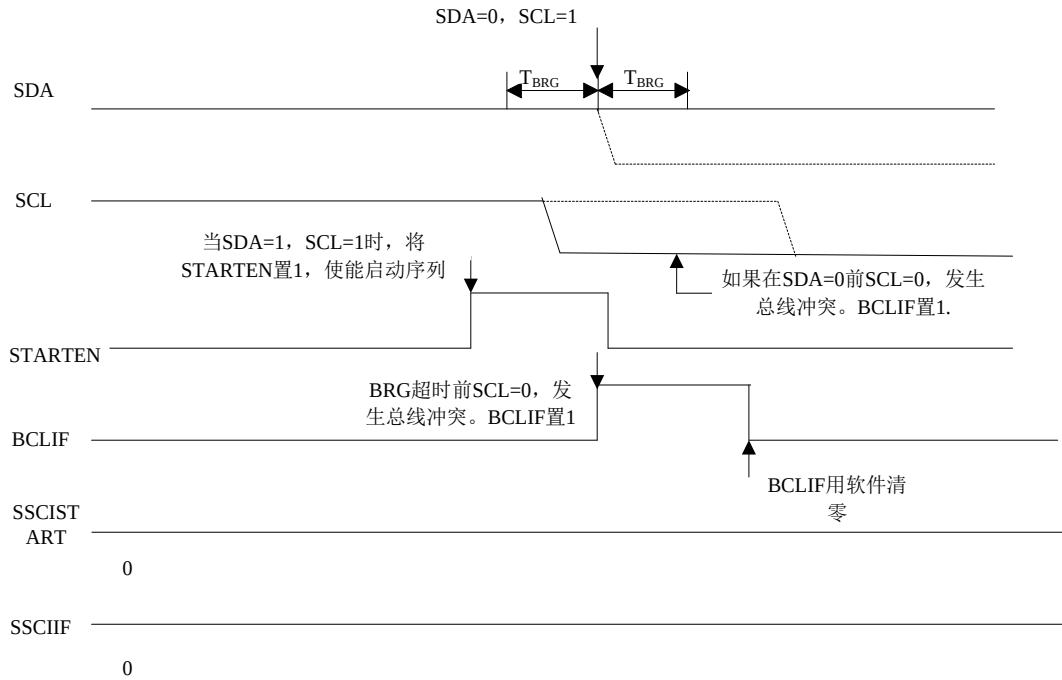


图10.22 启动条件期间的总线冲突（SCL=0）

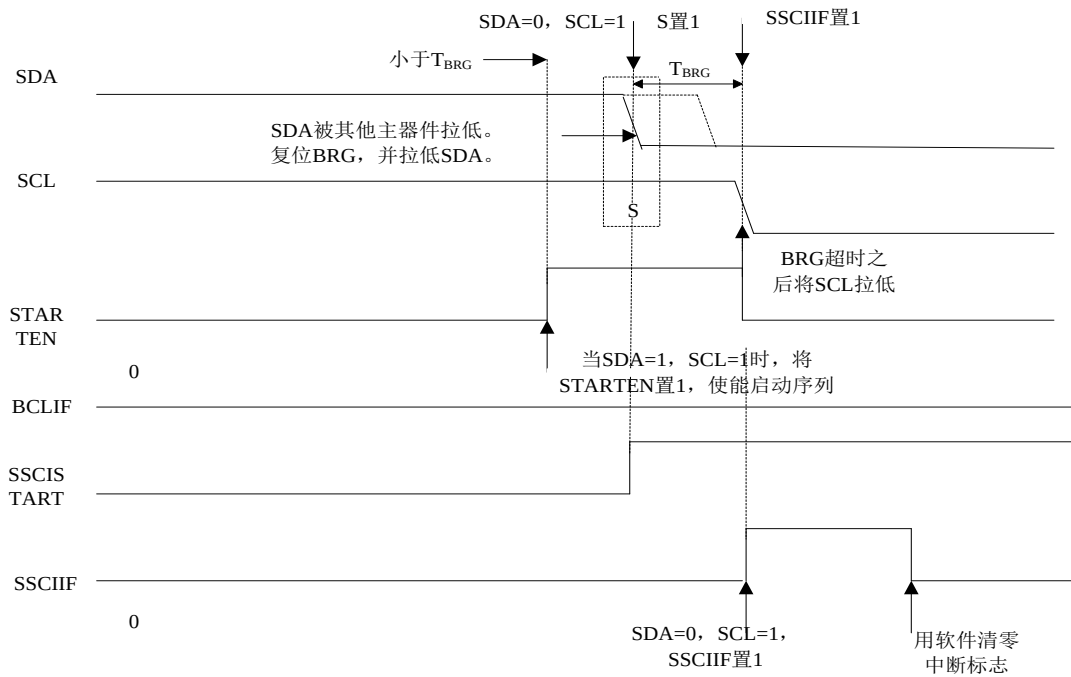


图10.23 启动条件期间由SDA仲裁引起的BRG复位

10.3.4.3 重复启动条件期间的总线冲突

在下列情况中, 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。

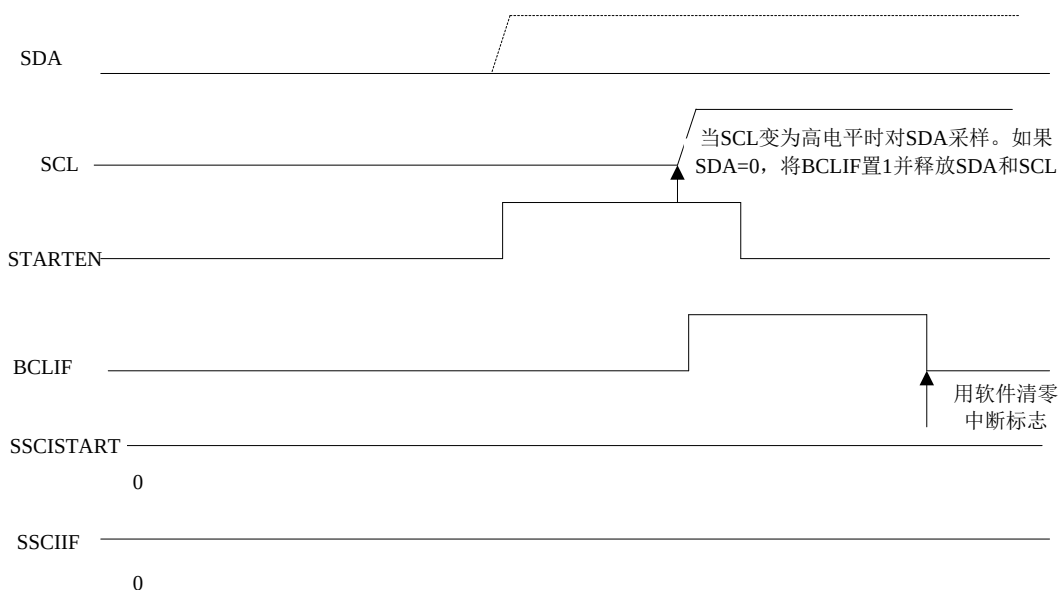


图 10.24 重复启动条件期间的总线冲突（情形 1）

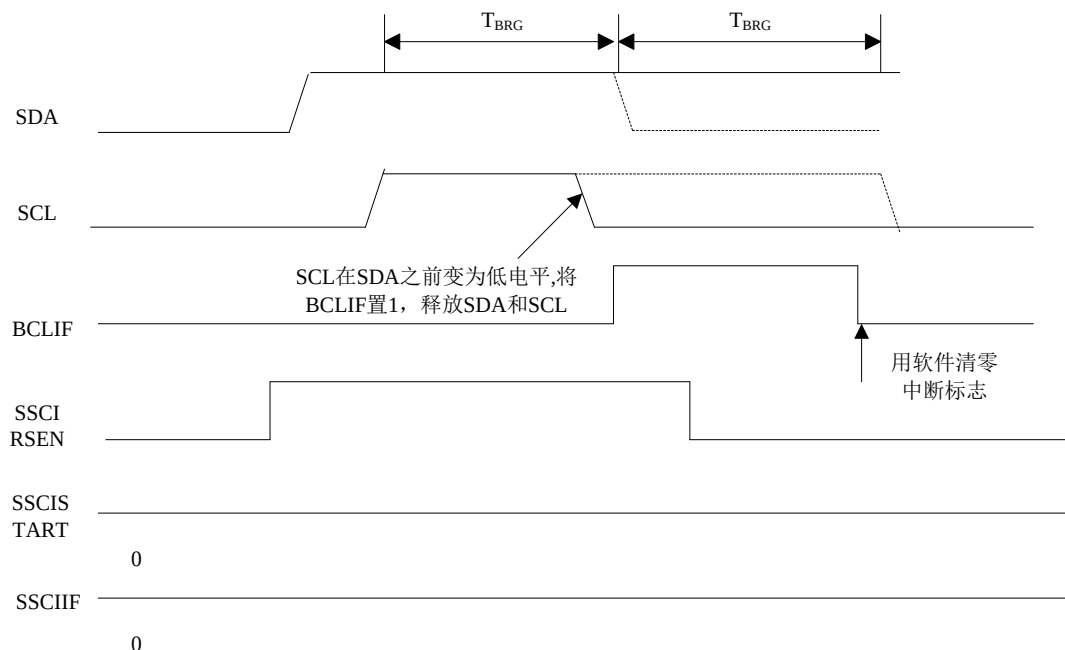


图 10.25 重复启动条件期间的总线冲突（情形 2）

10.3.4.4 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。

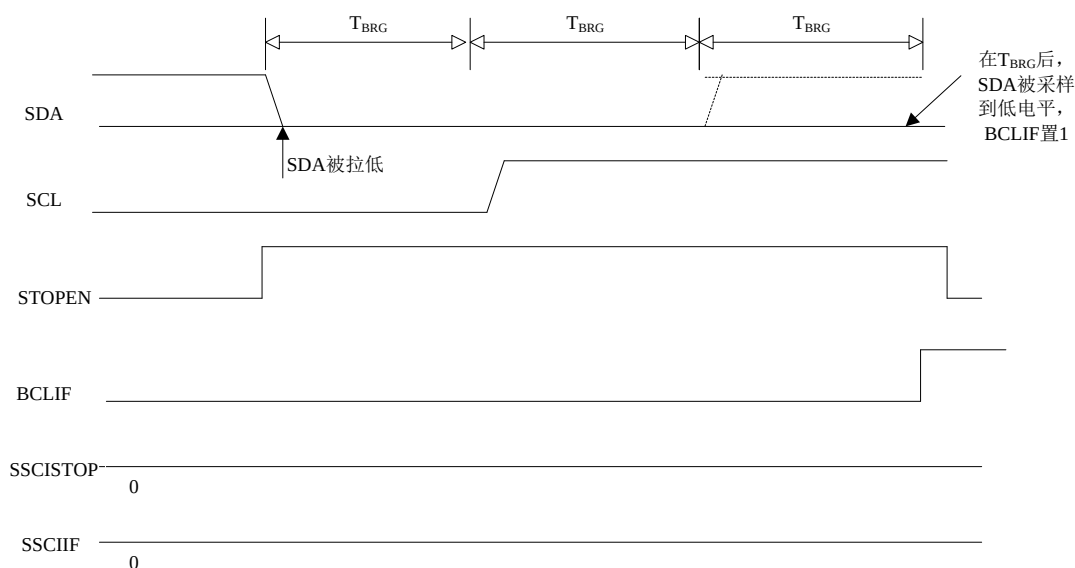


图 10.26 停止条件期间的总线冲突（情形 1）

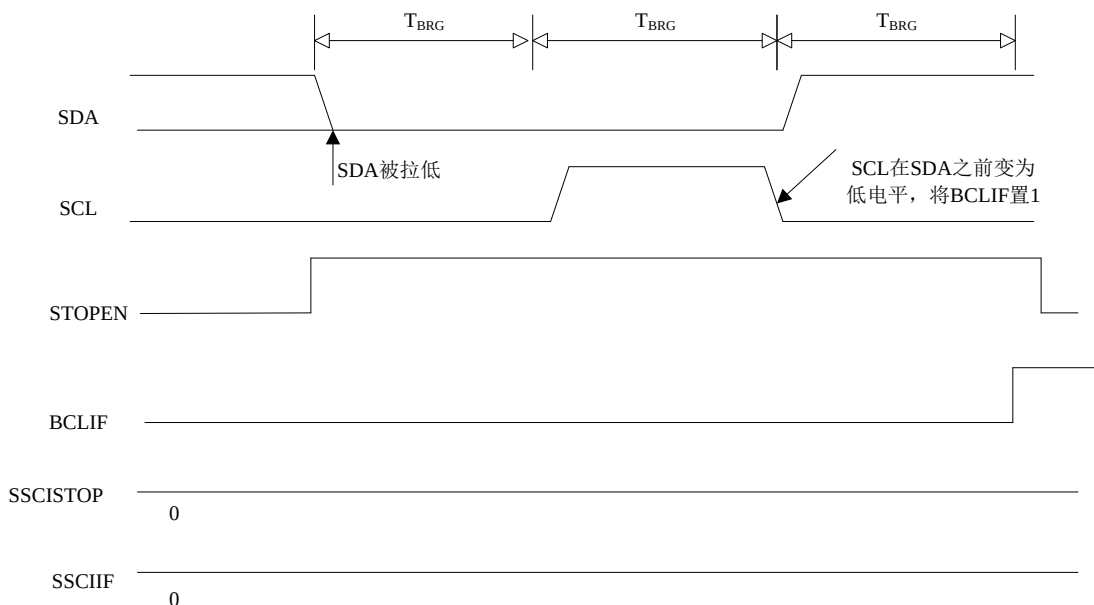


图 10.27 停止条件期间的总线冲突（情形 2）

10.3.4.5 SSCI 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SSCIADD <7:1> 进行地址比较。
- 10 位地址模式: 仅与 SSCIADD <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。

10.4 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 的原理框图如图 10.28 所示：

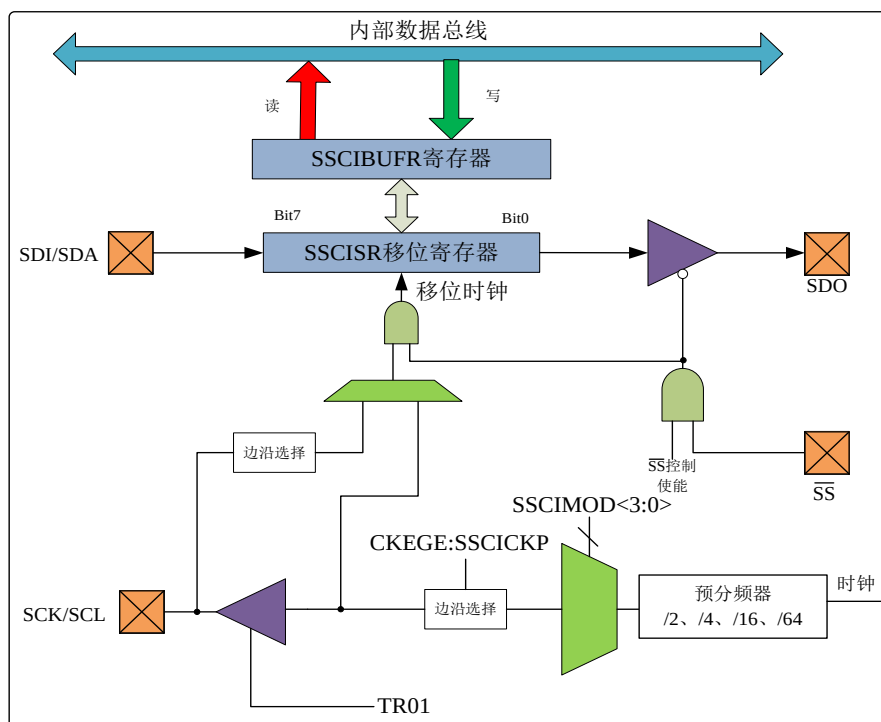


图 10.28 SPI 原理框图

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 (SSICCTL0<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。

10.4.1 工作原理

在用户初始化 SPI 时,需要先通过对相应的控制位 (SSICCTL0<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 SSCHIF 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 SSCIWCF (SSICCTL0<7>) 会置 1。此时用户必须用软件将 SSCIWCF 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。

10.4.2 使能 SPI/IO 与外部链接

要使能串行端口，SSCI 使能位 SSCIEN（SSCICTL0<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 SSCICTL0 寄存器，然后将 SSCIEN 位置 1。这将把 SDI、SDO、SCK 和 $\overline{SS}$ 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数据方向位。即：

- ◆ SDI，必须为数字输入状态（由 SPI 模块自动控制）。
- ◆ 对于 SDO，必须配置为数字输出状态。
- ◆ 对于 SCK（主控模式），必须配置为数字输出状态。
- ◆ 对于 SCK（从动模式），必须配置为数字输入状态。
- ◆ 对于 $\overline{SS}$ ，必须配置为数字输入状态。

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。

10.4.3 典型连接

下图10.29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

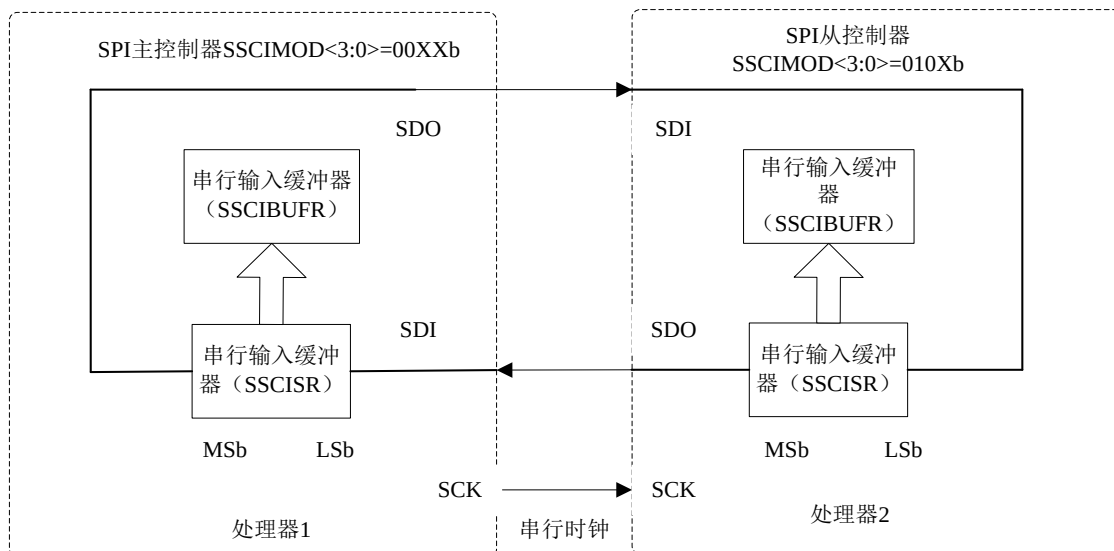


图 10.29 SPI 主从控制器连接

10.4.4 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 10.29 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（SSCICTL0<4>）进行适当的编程来选择时钟极性。图 10.30、图 10.31、图 10.32 和图 10.33 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：

- ☐ T2/2
- ☐ SCLK/4
- ☐ SCLK/16
- ☐ SCLK/64

下图 10.30 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。

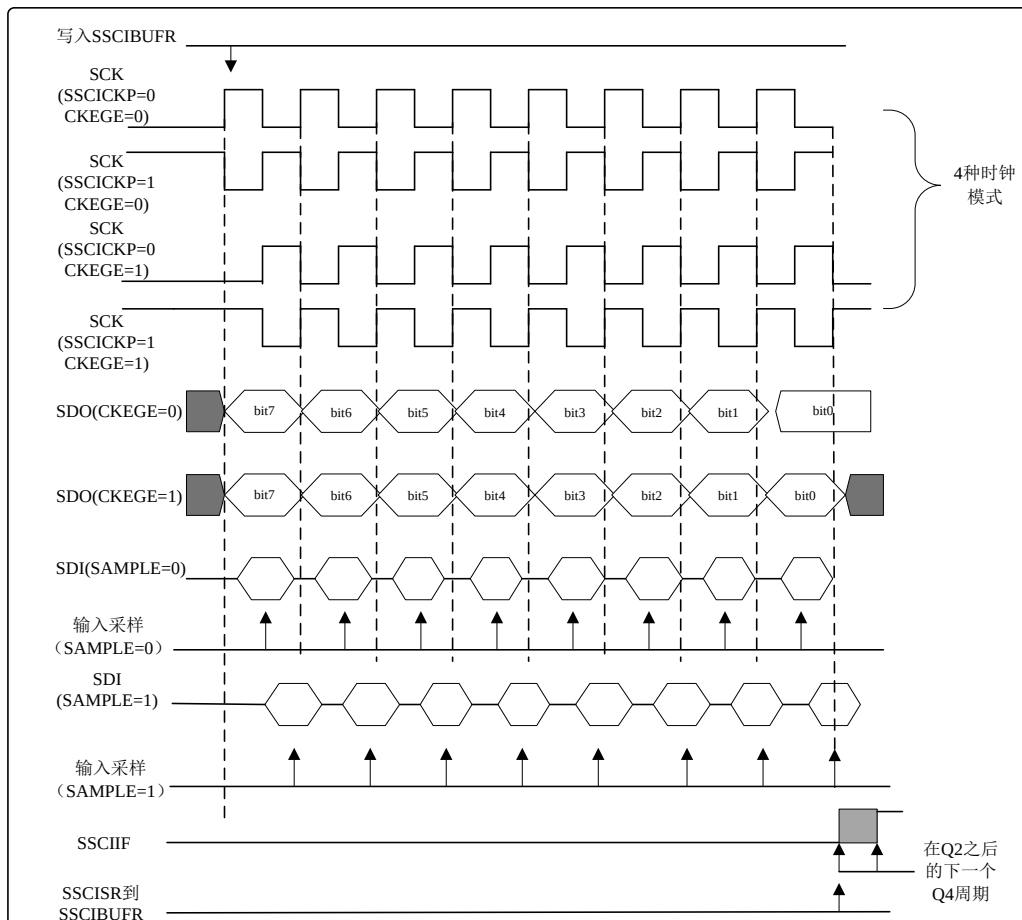


图 10.30 SPI 主控模式时序图

10.4.5 从模式

10.4.5.1 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。

10.4.5.2 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（SSCICTL0<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:

- 1) 当SPI处于从动模式，并且 $\overline{SS}$ 引脚控制使能（SSCICTL0<3:0>=0100）时，如果 $\overline{SS}$ 引脚置为VDD电平将使SPI模块复位。
- 2) 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）。

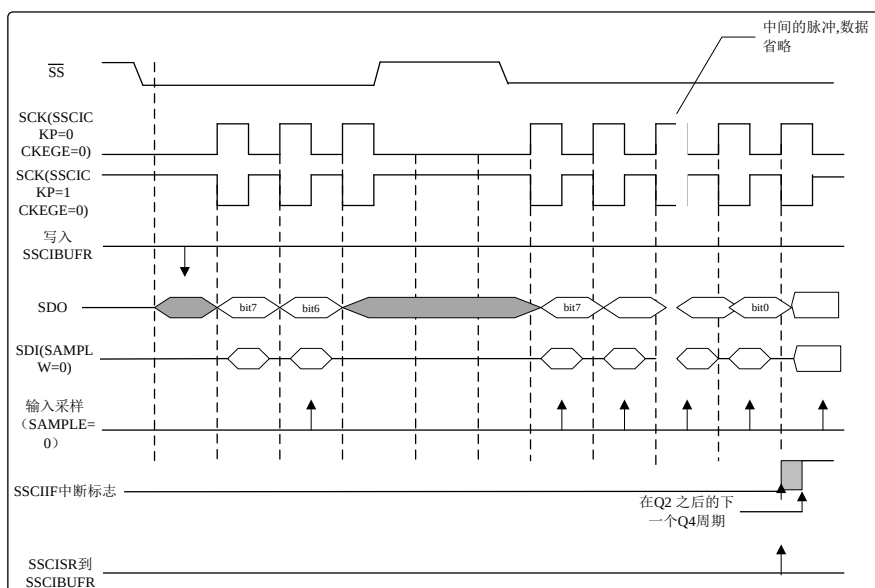


图10.31 从动时序图

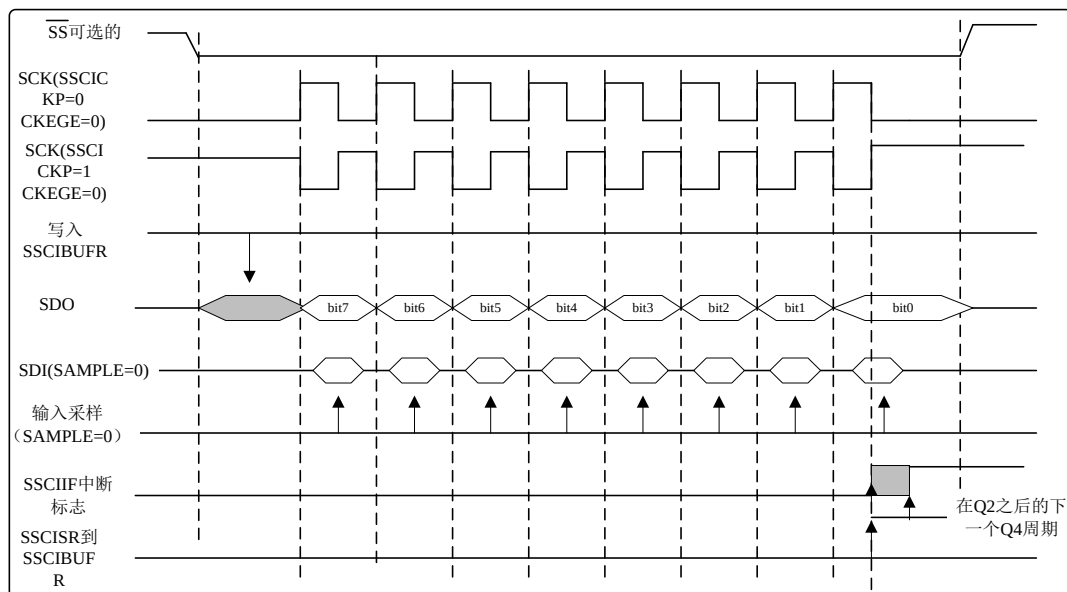


图10.32 SPI模式时序图（从动模式且CKEGE=0）

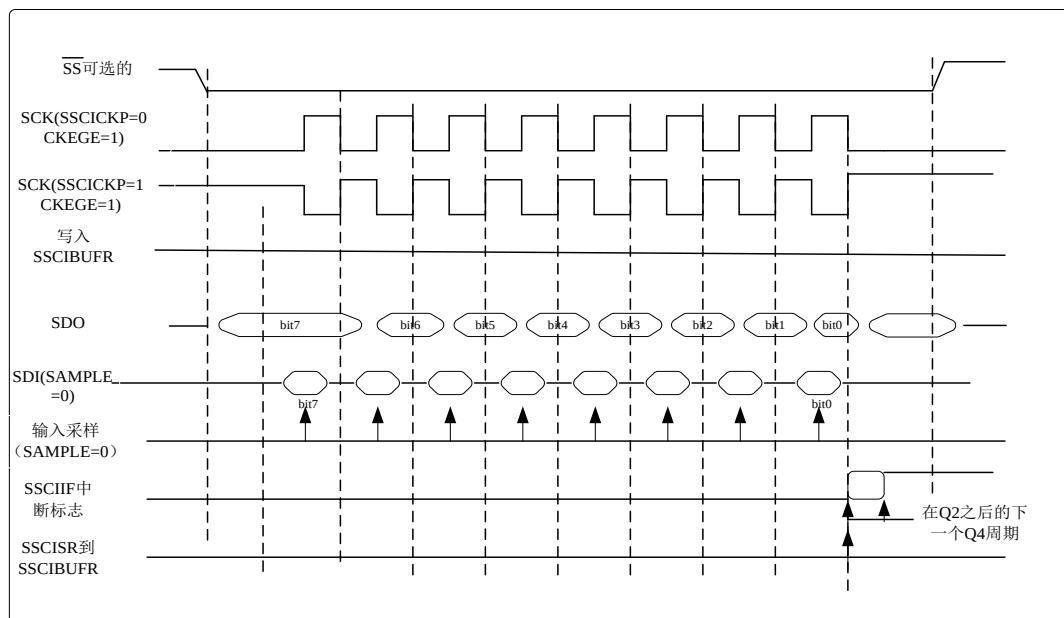


图 10.33 SPI 模式时序图（从动模式且 CKEGE=1）

10.4.6 休眠模式和复位

休眠模式下的工作

在主机模式下，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。

10.4.7 SPI 四种工作模式设置

10.4.7.1 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 SSCICTL0 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 SSCIIF 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 SSCIIF 自动置 1 (软件清零), 如果需要中断, 则使能相应中断使能位;

10.4.7.2 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 SSCICTL0 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, SSCIIF 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;

10.4.7.3 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 SSCICTL0 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, SSCHIF 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;

10.4.7.4 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 SSCICTL0 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 SSCHIF 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 SSCHIF 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 复位

KF8L10Z08 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。

这些寄存器的状态在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”。图 11.1 给出了片内复位电路的简化结构方框图。

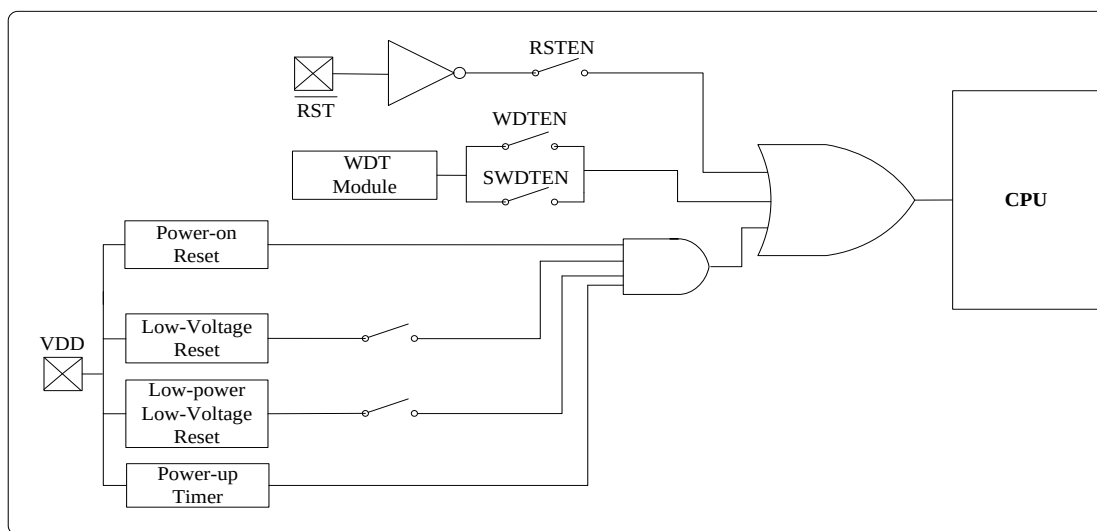


图 11.1 片内复位电路简化框图

注：上电延时定时器只对上电复位(POR)和欠压复位(LVR)有效；
LVR 和 LPLVR 的使能方式见 1.5 配置位章节。

11.1 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。

11.2 WDT 复位

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

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

11.3 RST 复位

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

在 $\overline{\text{RST}}$ 复位时，KF8L10Z08 器件有一个噪声滤波器用于滤出 $\overline{\text{RST}}$ 引脚上的噪声干扰，图 11.2 是建议 $\overline{\text{RST}}$ 复位电路。

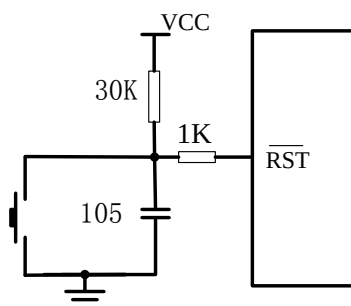


图 11.2 建议 $\overline{\text{RST}}$ 复位电路

11.4 欠压检测复位(LVR/LPLVR)

KF8L10Z08 型单片机片内配备一个欠压检测复位电路（LVR）和一个低功耗欠压检测复位电路（LPLVR）。通过编程时设定配置位中的 LVREN 位、LPLVREN 位和 PCTL 寄存器中的 SLVREN 位可以禁止/使能 LVR 或者 LPLVR，具体配置方法见配置位章节。

如果 V_{DD} 跌落至 V_{LVR} 以下且持续时间大于 T_{LVR} (T_{LVR} 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 V_{DD} 上升到 V_{LVR} 以上，如果上电延时配置位打开，将启动上电延时定时器，提供一个长度约为 33ms 的延时时间。

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

如果在上电延时定时器运行过程中发生 V_{DD} 跌落至 V_{LVR} 以下的情况，器件将返回欠压检测复位状态且上电延时定时器被重新初始化。直到 V_{DD} 上升至 V_{LVR} 以上时，上电延时定时器启动一个复位延时，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。

低功耗欠压检测复位电路（LPLVR）原理和欠压检测复位电路(LVR)一致，详细特性见电气规范章节。

11.5 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 33ms 的延时时间，通过配置位的 $\overline{\text{PWRT}}$ 位来打开。上电延时定时器的定时时钟为系统内部低频振荡器，上电延时定时器使单片机在 V_{DD} 上升到适当电平后才投入正常运行。

由于 V_{DD} 、温度、制造工艺、内部震荡器频率等的变化，不同单片机的上电延时时间有所差异。

11.6 不同复位条件下对寄存器的影响

表 11-1 寄存器在各种复位发生后的状态

名称	地址	上电复位	RST 复位 欠压复位	中断唤醒 WDT 超时唤醒
T0	01H	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	02H	0000 0000	0000 0000	PC+1
PSW	03H	000q 1xxx	000q uuuu	uuuq uuuu
P0	05H	--xx xxxx	-- uu uuuu	--uu uuuu
P2	06H	xxxx xxxx	uuuu uuuu	uuuu uuuu
P1	07H	xxxx xxxx	uuuu uuuu	uuuu uuuu
C1FILTCTL	08H	0000 0000	0000 0000	uuuu uuuu
C1FILTPRE	09H	0000 0000	0000 0000	uuuu uuuu
PCH	0AH	---0 0000	---0 0000	---u uuuu
INTCTL	0BH	0000 0000	0000 0000	uuuu uuuu
EIF1	0CH	0000 0000	0000 0000	uuuu uuuu
EIF2	0DH	0000 0000	0000 0000	uuuu uuuu
T1L	0EH	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1H	0FH	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1CTL	10H	-000 -000	-000 -000	-uuu -uuu
PWM11	13H	xxxx xxxx	uuuu uuuu	uuuu uuuu
PWMCTL	15H	1100 0000	1100 0000	uuuu uuuu
PP1	16H	1111 1111	1111 1111	uuuu uuuu
C1CTL	19H	0000 0000	0000 0000	uuuu uuuu
C2CTL	1AH	0000 0000	0000 0000	uuuu uuuu
VRECAL1	1CH	0111 1111	0111 1111	uuuu uuuu
ANSE2	1DH	1111 1111	1111 1111	uuuu uuuu
ADCDATAH	1EH	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCCTL0	1FH	0000 0000	0000 0000	uuuu uuuu
OPTR	21H	1111 1111	1111 1111	uuuu uuuu
IP0	22H	---- -000	---- -000	---- -uuu
IP1	23H	0000 0000	0000 0000	uuuu uuuu
IP2	24H	0000 0000	0000 0000	uuuu uuuu
TR0	25H	1111 1111	1111 1111	uuuu uuuu
TR2	26H	1111 1111	1111 1111	uuuu uuuu
TR1	27H	1111 1111	1111 1111	uuuu uuuu
VRECAL2	2AH	0110 0011	0110 0011	uuuu uuuu
VRECTL1	2BH	---0 0000	---0 0000	---u uuuu
EIE1	2CH	0000 0000	0000 0000	uuuu uuuu
EIE2	2DH	0000 0000	0000 0000	uuuu uuuu
PCTL	2EH	---1 000x	---1 00uq	---u uuuu
OSCCTL	2FH	0010 0000	0010 0000	uuuu uuuu
OSCCAL0	30H	1000 0000	1000 0000	uuuu uuuu
ANSE1	31H	1111 1111	1111 1111	uuuu uuuu
PP2	32H	1111 1111	1111 1111	uuuu uuuu

PWM21	33H	xxxx xxxx	սսսս սսսս	սսսս սսսս
PUR0	35H	1111 -111	1111 -111	սսսս -սսս
IOCL	36H	0000 0000	0000 0000	սսսս սսսս
OSCCAL1	37H	0000 -001	0000 -001	սսսս -սսս
NVMDATAH	38H	0000 0000	0000 0000	սսսս սսսս
NVMDATAL	39H	0000 0000	0000 0000	սսսս սսսս
NVMADDRH	3AH	0000 0000	0000 0000	սսսս սսսս
NVMADDRL	3BH	0000 0000	0000 0000	սսսս սսսս
NVMCTL0	3CH	---- x000	---- q000	---- սսսս
NVMCTL1	3DH	---- ----	---- ----	---- ----
ADCDATAL	3EH	xxxx xxxx	սսսս սսսս	սսսս սսսս
ADCCTL1	3FH	-000 0000	-000 0000	-սսս սսսս
APFCTL1	40H	0000 0000	0000 0000	սսսս սսսս
ANSE0	41H	1111 1111	1111 1111	սսսս սսսս
SLPCTL	43H	0000 0000	0000 0000	սսսս սսսս
T2REL	44H	xxxx xxxx	սսսս սսսս	սսսս սսսս
P0LR	45H	xxxx xxxx	սսսս սսսս	սսսս սսսս
P2LR	46H	xxxx xxxx	սսսս սսսս	սսսս սսսս
P1LR	47H	xxxx xxxx	սսսս սսսս	սսսս սսսս
T2REH	48H	xxxx xxxx	սսսս սսսս	սսսս սսսս
T2L	49H	xxxx xxxx	սսսս սսսս	սսսս սսսս
T2H	4AH	xxxx xxxx	սսսս սսսս	սսսս սսսս
T2CTL	4BH	0000 0000	0000 0000	սսսս սսսս
OSCCAL2	4CH	0011 1111	0011 1111	սսսս սսսս
T3CTL	4EH	0000 0000	0000 0000	սսսս սսսս
T3L	4FH	0000 0000	0000 0000	սսսս սսսս
PUR1	50H	1111 1111	1111 1111	սսսս սսսս
T3CTL1	54H	0111 1000	0111 1000	սսսս սսսս
T3CCPH	55H	0000 0000	0000 0000	սսսս սսսս
T3CCPL	56H	0000 0000	0000 0000	սսսս սսսս
T2CCPL	57H	0000 0000	0000 0000	սսսս սսսս
APFCTL2	58H	---- 0000	---- 0000	---- սսսս
T2CCPH	5AH	0000 0000	0000 0000	սսսս սսսս
APFCTL0	5BH	0000 0000	0000 0000	սսսս սսսս
CCPCLR	5CH	0000 0000	0000 0000	սսսս սսսս
PUR2	5EH	1111 1111	1111 1111	սսսս սսսս
T3H	5FH	0000 0000	0000 0000	սսսս սսսս
PWM12	60H	xxxx xxxx	սսսս սսսս	սսսս սսսս
PWM22	61H	xxxx xxxx	սսսս սսսս	սսսս սսսս
WDTPS	62H	---- 0100	---- 0100	---- 0100
OSCSTA	63H	0110 --00	0110 --00	սսսս --սս
SSCIMSK	64H	1111 11111	1111 11111	սսսս սսսսս
SSCIADD		0000 0000	0000 0000	սսսս սսսս
SSCISTA	65H	0000 0000	0000 0000	սսսս սսսս
SSCICTL0	66H	0000 0000	0000 0000	սսսս սսսս
SSCICTL1	67H	0000 0000	0000 0000	սսսս սսսս

SSCIBUFR	68H	0000 0000	0000 0000	uuuu uuuu
C2FILTCTL	69H	0000 0000	0000 0000	uuuu uuuu
C2FILTPRE	6AH	0000 0000	0000 0000	uuuu uuuu
RC32KCAL	6CH	0000 0000	0000 0000	uuuu uuuu
XTALCAL	6DH	0101 0100	0101 0100	uuuu uuuu

注：u = 不变，x = 未知，- = 未实现位，读为 0; q = 视具体条件而定。

表 11-2 不同复位条件下对标志位的影响

$\overline{\text{POR}}$	$\overline{\text{LVR}}$	$\overline{\text{TO}}$	$\overline{\text{PD}}$	复位方式
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 $\overline{\text{RST}}$ 复位
u	u	1	0	休眠模式中的 $\overline{\text{RST}}$ 复位

图注: u = 未发生变化

注：u = 不变。

12 功耗模式

KF8L10Z08 可以通过执行一条 IDLE 指令进入休眠状态。休眠状态下，系统时钟及内部高频时钟被禁止，程序停止运行，RAM 保持。

为了满足用户在不同使用条件下对功耗的要求，KF8L10Z08 在休眠状态下提供了两种不同的功耗模式。

- 运行状态下：
 - 正常运行模式 (Normal run mode, NRM)
- 休眠状态下：
 - 深度休眠模式 (Deep sleep mode, DSLP)
 - 空闲模式 (Idle mode, IDLE)

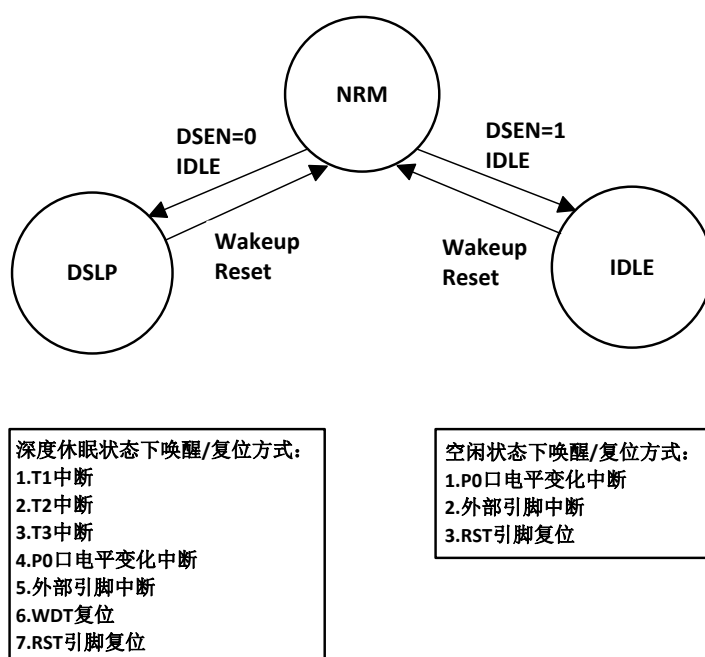


图 13-1: KF8L10Z08 功耗模式状态图

12.1 相关寄存器

SLPCTL 寄存器的 DSEN 位用于选择休眠状态下的功耗模式。

寄存器13.1: SLPCTL: 休眠模式控制寄存器(43H)

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

DSEN: 休眠模式选择位

1 = 执行 IDLE 指令后, 芯片将进入空闲模式

0 = 执行 IDLE 指令后, 芯片将进入深度休眠模式

图注: R=可读 W=可写 -=未用 U=未实现位

注: 请勿将除 DSEN 位以外的保留位配置为 1, 否则可能引起单片机无法正常工作。

12.2 运行状态下: 正常运行模式

在正常运行模式 (normal run mode) 下, MCU 正常工作, 系统时钟、高频、低频时钟均可使用, 所有外设均可使用。用户可以按照功能需求对 MCU 进行设置以达到最佳性能。

12.3 休眠状态下: 深度休眠模式和空闲模式

KF8L10Z08 在休眠状态下提供了两种不同的功耗模式。当寄存器位 DSEN=0 条件下执行 IDLE 指令, MCU 将进入深度休眠模式; 当 DSEN=1 条件下执行 IDLE 指令, 则 MCU 将进入空闲模式。

在深度休眠模式下, 内部整流器被关闭, 系统时钟及内部高频时钟被禁止, 程序停止运行, RAM 保持; 定时/计数器 T1 可使用 T1CK、T2 和 T3 允许使用内部低频时钟或者外部低频晶振进行计数; WDT 允许使用内部低频时钟工作。在深度休眠模式下, 可以通过 T1 中断、T2 中断、T3 中断、P0 口电平变化中断、外部引脚中断或者 WDT 来唤醒 MCU; 也可以通过 RST 引脚对 MCU 进行复位。

注: 进入深度休眠模式时, 如果无外设被使能, 可通过 P0 口电平变化中断、外部引脚中断 INT0/1/2 或者 RST 引脚对 MCU 进行唤醒或者复位。

在空闲模式下, 内部整流器被关闭, 所有时钟被禁止, 程序停止运行, 所有外设被禁止, RAM 保持。在空闲模式下, 可以通过 P0 口电平变化中断或者外部引脚中断来唤醒 MCU; 也可以通过 RST 引脚对 MCU 进行复位。

表 13-1：功耗模式列表

功耗模式	配置	描述	状态	唤醒/复位方式
正常运行模式	DSEN=0	系统时钟/高频时钟/低频时钟均可使用； 所有外设均可使用。	RUN	–
深度休眠模式	DSEN=0 执行 IDLE 指令	系统时钟/高频时钟被禁止； 低频时钟允许使用； 内部整流器被关闭； 允许外设： WDT/T1/T2/T3。	DSLP	T1 中断； T2 中断； T3 中断； P0 口电平变化中断； 外部引脚中断（INT0/1/2）； WDT 复位； RST 引脚复位。
空闲模式	DSEN =1 执行 IDLE 指令	所有时钟被禁止； 所有外设被禁止； 内部整流器被关闭；	IDLE	P0 口电平变化中断； 外部引脚中断（INT0/1/2）； RST 引脚复位。

12.4 深度休眠模式下可工作外设

KF8L10Z08 提供定时/计数器 T1/T2/T3 以及 WDT 可以在深度休眠模式下工作。

根据下表配置外设，执行 IDLE 指令后，外设将继续工作。

表 13-1：外设休眠工作配置列表

外设	深度休眠模式下工作配置方式
T1	- T1 可选工作时钟配置方式： - T1CTL.T1CS=1（T1 选择 T1CK 作为计数时钟） - 将 P0.5 配置位数字输入口 - T1 计数器使能配置方式： T1CTL.T1ON=1 (使能 T1 计数)
T2	- T2 可选工作时钟配置方式： - T2CTL.T2CS<1:0>=01（自动使能内部低频振荡器） - T2CTL.T2CS<1:0>=11（自动使能外部 32.768K 低频晶振） - T2 计数器使能配置方式： T2CTL.T2ON=1 (使能 T2 计数)
T3	- T3 可选工作时钟配置方式： - T3CTL.T3CS<1:0>=01（自动使能内部低频振荡器） - T3CTL.T3CS<1:0>=11（自动使能外部 32.768K 低频晶振） - T3 计数器使能配置方式： T3CTL.T3ON=1 (使能 T3 计数)
WDT	配置位 WDTEN=1 或者 PCTL.SWDTEN=1 (默认使用内部低频振荡器)

12.5 休眠状态下 I/O 口注意事项

为使休眠状态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。

12.6 休眠状态唤醒方式

单片机进入休眠状态后由于工作的需要，要将单片机从休眠状态唤醒，在 KF8L10Z08 中可通过以下方式将单片机从休眠状态唤醒：

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 内部中断
4. P0 口电平变化中断
5. 外设中断 (T1/T2/T3/INT1/INT2)

深度休眠模式下，上述 5 项均可使用；空闲模式下，由于所有时钟和外设均被禁止，看门狗唤醒和外设中断唤醒将无法使用。

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状态唤醒，并不会导致复位。可通过状态寄存器中的 $\overline{TO}$ 和 $\overline{PD}$ 位来确定单片机唤醒的原因。上电时 $\overline{PD}$ 位将被置 1，而当器件从休眠状态唤醒时，该位将被清 0。 $\overline{TO}$ 位则在 WDT 唤醒发生时被清 0。

在使用中断方式唤醒时，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 指令之后再加一条 NOP 指令即可。

13 看门狗定时器

为了防止单片机在正常工作时程序跑飞，KF8L10Z08 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。

13.1 看门狗相关寄存器

表 13-1 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	SWDTE N	$\overline{\text{POR}}$	$\overline{\text{LVR}}$
62H	WDTPS	-	-	-	-	WDTPS3	WDTPS2	WDTPS1	WDTPS0

13.1.1 看门狗预分频选择寄存器 WDTPS

寄存器13.1: WDTPS: WDT预分频选择寄存器(地址:62H)

	bit7							bit0
复位值 ---- 0100	-	-	-	-	WDTPS3	WDTPS2	WDTPS1	WDTPS0
	R/W	U	U	U	R/W	R/W	R/W	R/W

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
 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

13.2 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。

13.3 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 CWDTC 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，状态字寄存器 PSW 中的 T0 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。

13.4 看门狗的周期

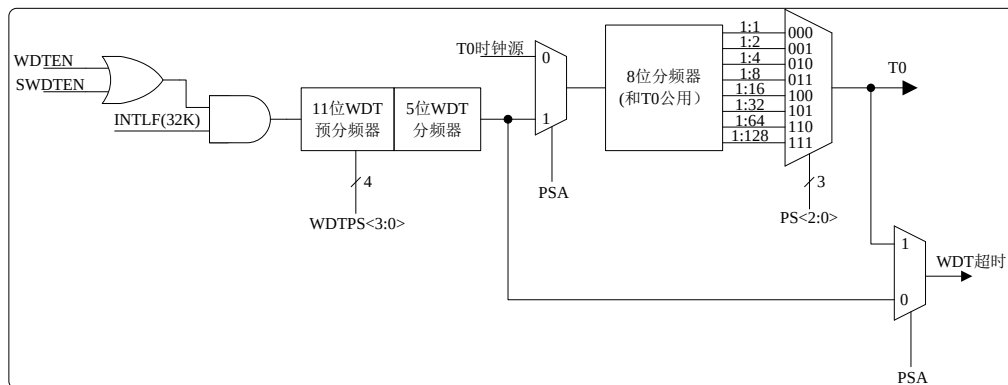


图 13.1 看门狗定时器框图

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 31.25K），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 CWDTC 指令）。否则，可能引起芯片的异常复位。

14 电气规范

14.1 极限参数值

极限参数值		
序号	参数说明	参数范围
1	偏置电压下的环境温度	-40 ℃ ~85 ℃
2	储存温度	-65 ℃ ~150 ℃
3	VDD 相对于VSS 的电压	-0.3V~+6.0V
4	MODE 相对于Vss 的电压	-0.3V~ +12.5V
5	其它引脚相对于VSS 的电压	-0.3V~VDD+0.3V
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

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

14.2 时钟温漂典型特性图

INTHF振荡器频率精度与温度、电压之间的关系图如下：



图 14.1 INTHF 振荡器频率精度与温度、电压之间的关系

14.3 静态电流特性

表14-1 芯片INTHF静态电流(I_{DD})特性

测试条件: 25 °C						
序号	测试条件		最小值	典型值	最大值	单位
	振荡频率	V _{DD} (V)				
2	4MHz	5.0	–	223.71	345	μ A
		4.0	–	223.60	345	
		3.3	–	224.09	345	
		3.0	–	224.56	345	
		2.5	–	229.29	345	
4	2MHz	5.0	–	146.67	220	
		4.0	–	146.51	220	
		3.3	–	146.65	220	
		3.0	–	146.86	220	
		2.5	–	149.15	220	
5	1MHz	5.0	–	107.84	160	
		4.0	–	107.60	160	
		3.3	–	107.60	160	
		3.0	–	107.71	160	
		2.5	–	108.87	160	
6	500kHz	5.0	–	88.40	130	
		4.0	–	88.11	130	
		3.3	–	88.06	130	
		3.0	–	88.11	130	
		2.5	–	88.70	130	
7	250kHz	5.0	–	78.76	120	
		4.0	–	78.45	120	
		3.3	–	78.38	120	
		3.0	–	78.36	120	
		2.5	–	78.66	120	
8	62.5kHz	5.0	–	73.99	110	
		4.0	–	73.65	110	
		3.3	–	73.59	110	
		3.0	–	73.54	110	
		2.5	–	73.66	110	
9	15.625kHz	5.0	–	71.60	110	
		4.0	–	71.28	110	
		3.3	–	71.21	110	
		3.0	–	71.11	110	
		2.5	–	71.16	110	

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

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

14.4 功耗模式直流特性

表 14-2 芯片功耗模式直流特性

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	深度休眠模式 (DSLP)	IMOD=0 所有外设及时钟源禁止	5.0	—	1.13	—	μ A
			3.3	—	0.75	—	
			3.0	—	0.68	—	
2	深度休眠模式 (DSLP)	WDT_ON (INTLF_ON)	5.0	—	1.49	—	
			3.3	—	1.08	—	
			3.0	—	1.01	—	
3	深度休眠模式 (DSLP)	T2_ON (INTLF_ON)	5.0	—	1.50	—	
			3.3	—	1.10	—	
			3.0	—	1.01	—	
4	深度休眠模式 (DSLP)	T2_ON (XTAL32.768K_ON) WDT_ON (INTLF_ON)	5.0	—	1.92	—	
			3.3	—	1.51	—	
			3.0	—	1.43	—	
5	深度休眠模式 (DSLP)	T2_ON (XTAL32.768K_ON)	5.0	—	1.55	—	
			3.3	—	1.16	—	
			3.0	—	1.10	—	
6	空闲模式 (IDLE)	IMOD=1 所有外设及时钟源禁止	5.0	—	0.77	—	
			3.3	—	0.47	—	
			3.0	—	0.42	—	

注：外部低频晶振 XTAL 和定时器 T2 联合使用可模拟 RTC 工作。

14.5 I/O 端口电平和芯片供电电压特性

表 14-3 芯片 IO 端口电平特性

工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$						
符号	参数说明	测试条件	最小值	典型值	最大值	单位
V_{IL}	输入低电平		V_{SS}		0.6	V
	I/O引脚: 采用TTL缓冲器		V_{SS}	-	$0.15V_{DD}$	
	采用施密特缓冲触 发器		V_{SS}		$0.2 V_{DD}$	
V_{IH}	输入高电平		$V_{DD} - 0.6$		V_{DD}	V
	I/O端口: 采用TTL缓冲器		$0.8 V_{DD}$	-	V_{DD}	
	采用施密特缓冲触 发器					
V_{OL}	输出低电压		-	-	0.6	V
V_{OH}	输出高电压		$V_{DD} - 0.6$	-	-	V

表 14-4 芯片供电电压特性

工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$						
符号	参数说明	测试条件	最小值	典型值	最大值	单位
V_{DD}	电源电压	$F_{OSC} \leq 4\text{Mhz}$, 使能上电延时功能	2.1	-	5.5	V
S_{VDD}	确保能够产生 内部上电复位 信号的VDD 上升速率		0.0625	-	-	V/ms
V_{LVR}	欠压检测电压	使能欠压检测功能	2.1	-	2.3	V
V_{POR}	确保能够产生 内部上电复位 信号的VDD 起 始电压	关闭欠压检测功能	2.0	-	2.1	V
V_{LPLVR}	欠压复位电压	使能LPLVR	1.75		1.9	V

注意: V_{DDcore} 和 V_{SS} 必须接电容, 建议值为 $4.7\mu\text{F}$ 。

14.6 A/D 转换器（ADC）特性

表 14-5 A/D 转换器（ADC）特性

工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$						
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_{AIN}^*	满量程范围		V_{SS}	-	V_{REF}	V
T_{CNV}^*	AD转换时间		-	11	-	T_{AD}
$TsOPAlc^*$	AD转换速率		-	-	400	KSPS
I_q^*	AD工作电流			550		μA

注：（1）*表示该数据为设计值。

（2）KF8L10Z08 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。

14.7 光电检测器电气特性

表14-6 光电检测器电气特性

工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$						
符号	参数说明	测试条件	最小值	典型值	最大值	单位
F_{conv}^*	检测转换因数 编程范围	一级增益选15X	0.025		0.8	V/nA
$F_{conv_step}^*$	检测转换因数 编程步进	一级增益选15X	1.5		3	dB
$F_{conv_temp}^*$	检测转换因数 温漂	一级增益选15X		1.5		%
F_{ampL}^*	下截止频率			0.45		kHz
F_{ampH}^*	上截止频率			4.5		kHz
V_{dsup}^*	D_{sup} 电压	选择P驱动		3.3		V
I_{dsup}^*	I_{dsup}		-10		0	μA
I_q^*	检测器工作电 流	$VDD=3.3V$		190		μA

*表示该数据为设计值。

14.8 振荡器模块规范

表 14-7 振荡器模块特性

测试条件（特别声明除外）： 工作温度 25℃							
参数编号	符号	特性	最小值	典型值	最大值	单位	备注
1	Fosc	振荡频率	3.96	4	4.04	MHz	$2.1V \leq V_{DD} \leq 5.5V @ 25^{\circ}C$

14.9 内部整流器

表 14-8 内部整流器模块特性

工作温度 $-40^{\circ}C \leq T_A \leq +85^{\circ}C$						
符号	参数说明	测试条件	最小值	典型值	最大值	单位
Vddcore	数字部分供电电源	外接4.7uF		1.8		V

14.10 内部低频振荡器

表 14-9 内部低频振荡器特性

工作温度 $-40^{\circ}C \leq T_A \leq +85^{\circ}C$						
符号	参数说明	测试条件	最小值	典型值	最大值	单位
Freq	频率			32		kHz
TC	温度漂移		-5		5	%
I _{DD}	静态电流	VDD=3.3V	-	0.4	1	μ A

14.11 外部低频振荡器

表 14-10 外部 32.768kHz 低频晶振特性

工作温度 $-40^{\circ}C \leq T_A \leq +85^{\circ}C$						
符号	参数说明	测试条件	最小值	典型值	最大值	单位
Freq	频率			32.768		kHz
I _{DD}	静态电流	VDD=3.3V	-	0.4	-	μ A

14.12 外部高频晶振

表 14-11 外部高频晶振特性

工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$						
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

14.13 DATA EEPROM

表 14-12 DATA EEPROM

工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$						
符号	参数说明	测试条件	最小值	典型值	最大值	单位
1	DATA EEPROM 可靠工作电压		2.1	-	5.5	V
2	DATA EEPROM 可靠工作频率		-	-	250	kHz
3	可擦写次数	25°C			1000000	次
4	擦写时间				6	ms

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 V_{DD} 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15.1： 不同 V_{DD} 时INTHF的典型 I_{DD} — F_{OSC} 关系曲线图

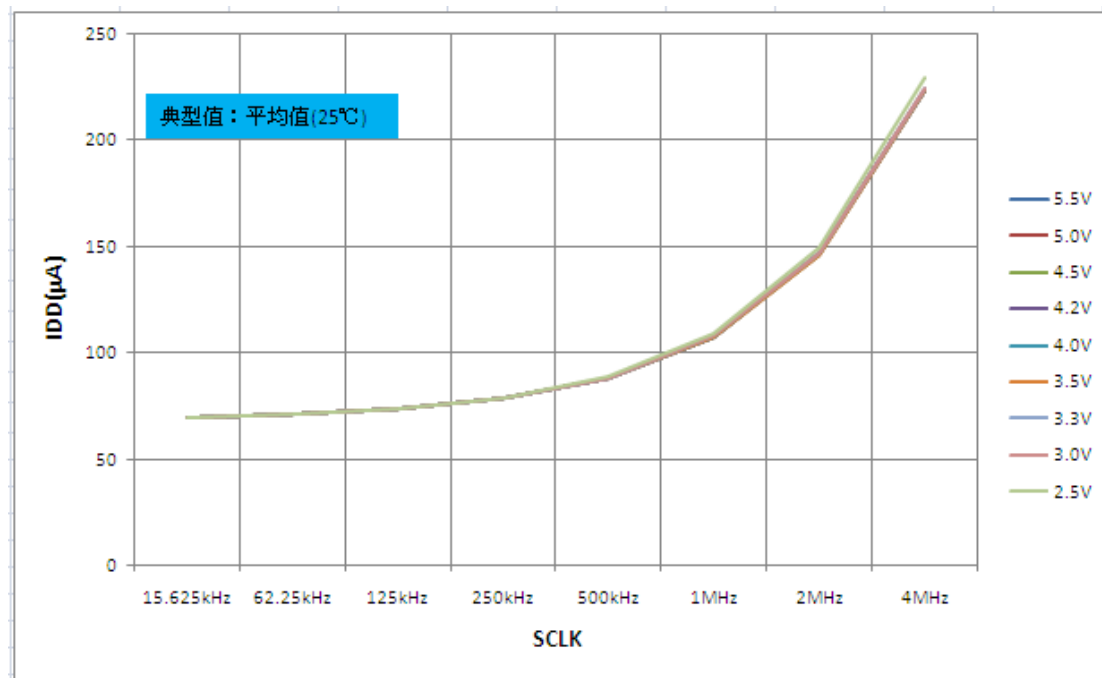


图 15.2： 不同 V_{DD} 时典型 I_{IDLE} — V_{DD} 关系曲线图

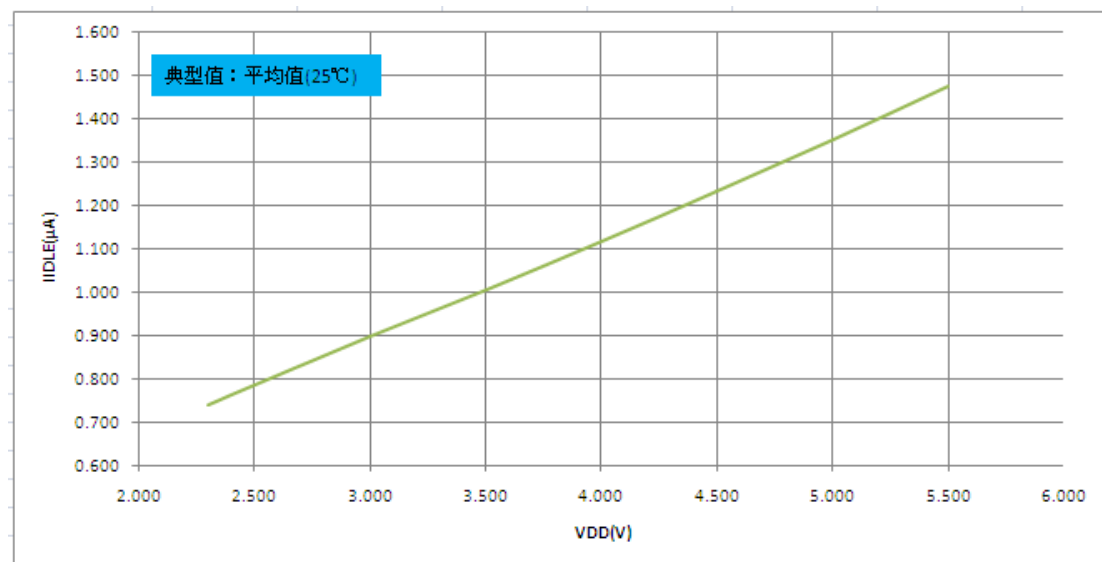


图15.3: 不同VDD时深度休眠电流 $I_{IDLEPUR}$ — VDD 关系曲线图

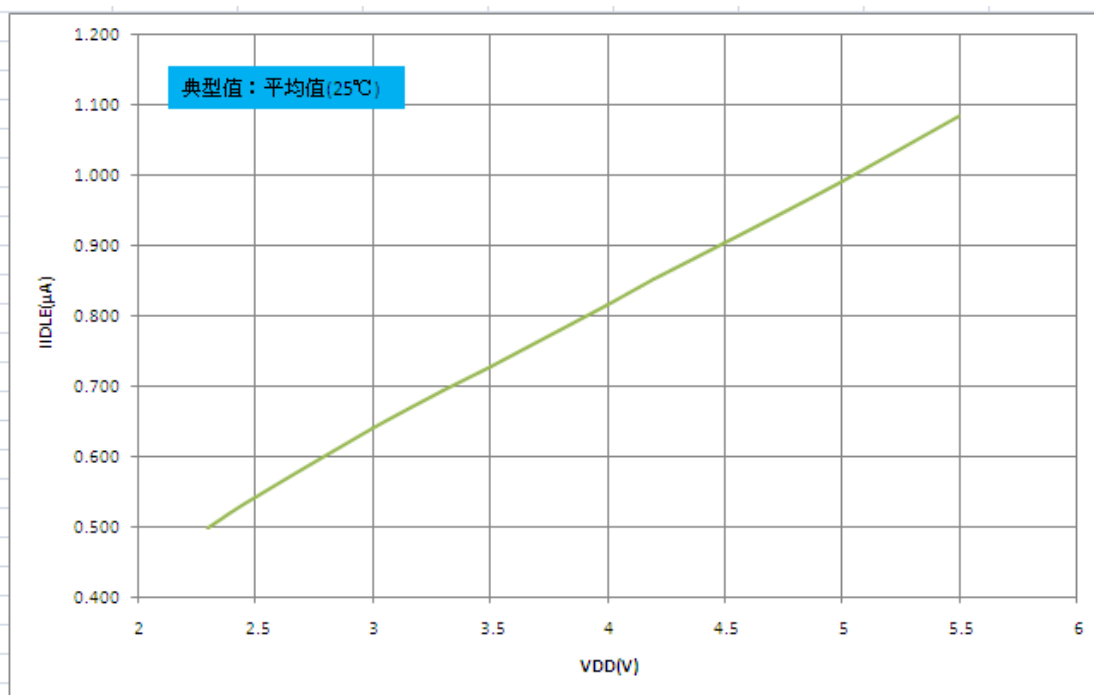
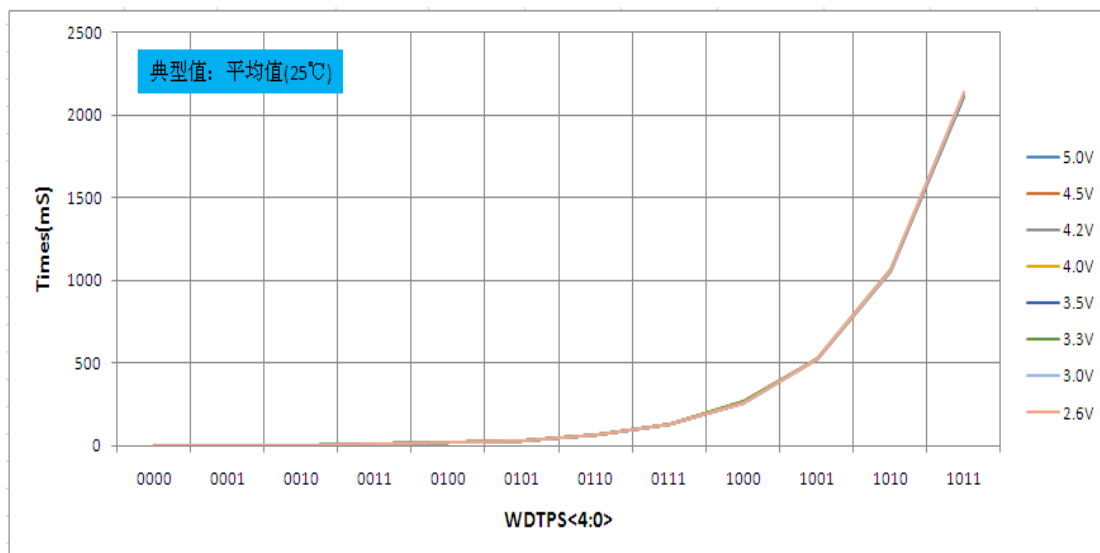


图15.4: 不同VDD时看门狗预分频比 — 周期关系曲线图



注意: 当 WDTPS<4:0>=0000 时, WDT 溢出时间为 1.2ms

图15.5: 看门狗电流 — VDD关系曲线图

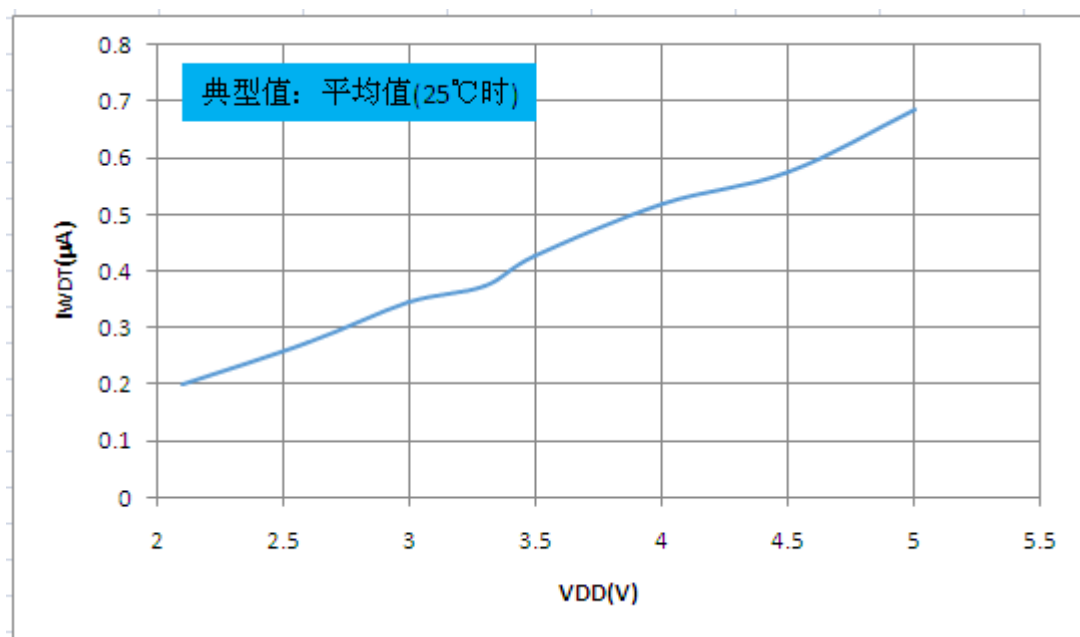


图 15.6: 欠压复位电流 I_{LVR} — VDD 关系曲线图

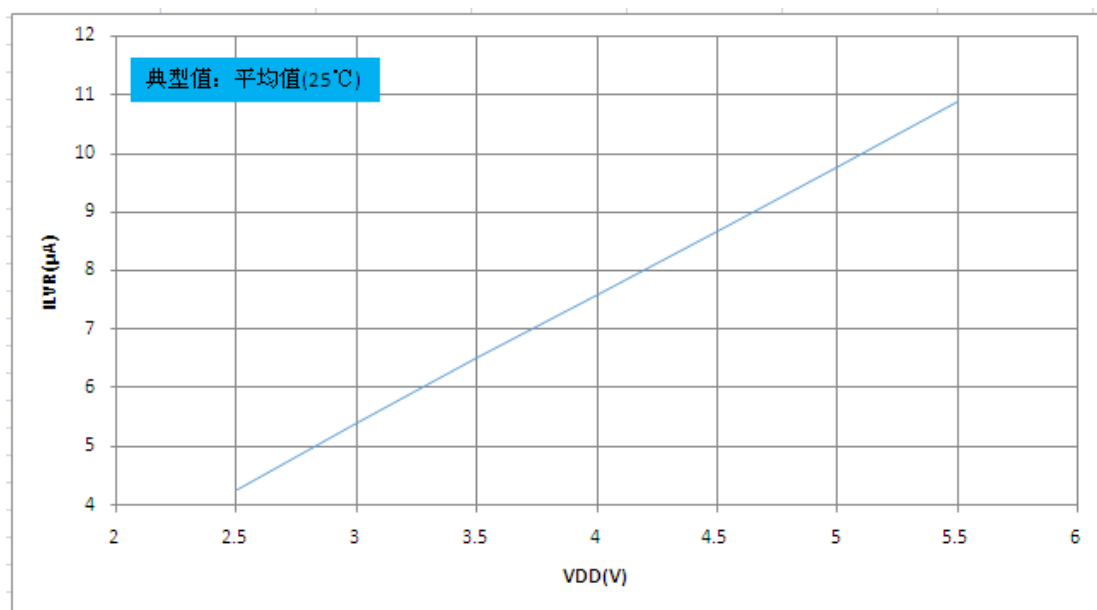


图15.7: 不同温度时 V_{OH} — I_{OH} 关系曲线图 ($V_{DD} = 5.0V$)

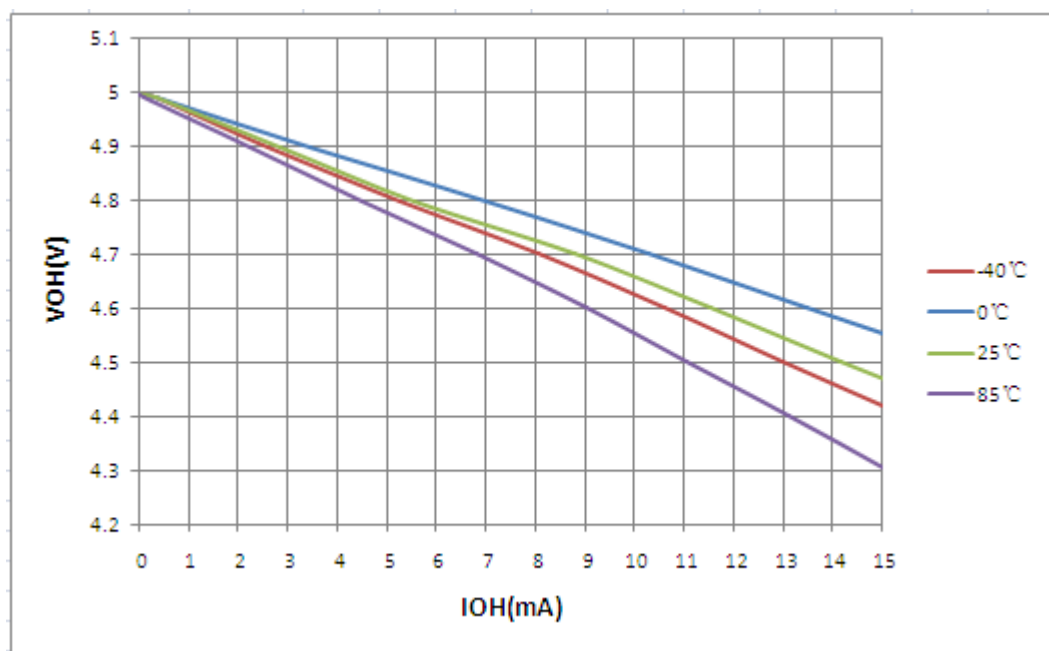


图15.8: 不同温度时 V_{OH} — I_{OH} 关系曲线图 ($V_{DD} = 3.3V$)

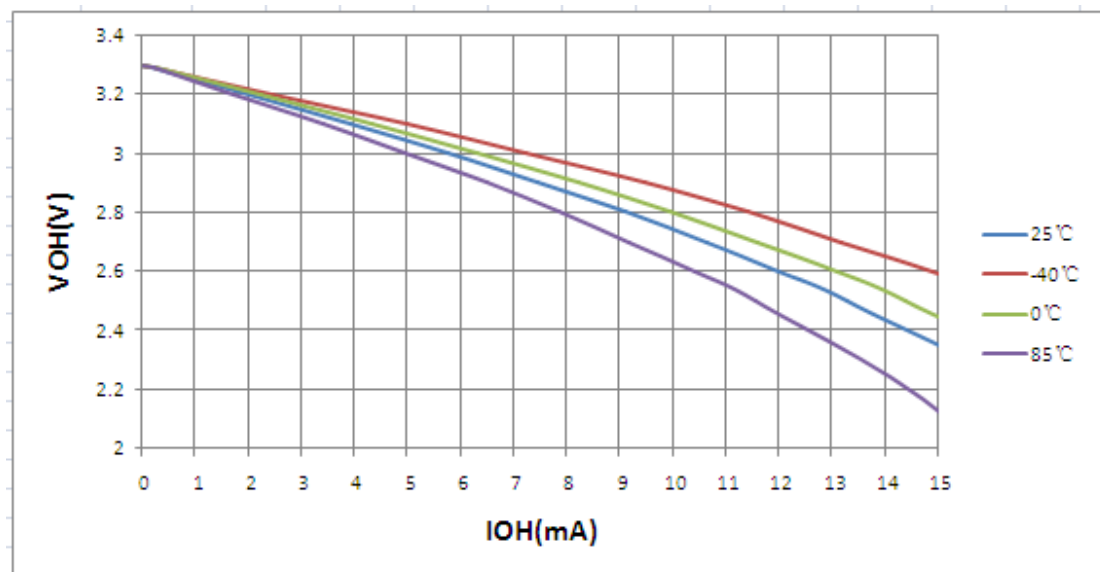


图15.9: 不同温度时VOL — IOL 关系曲线图 (VDD = 5.0V)

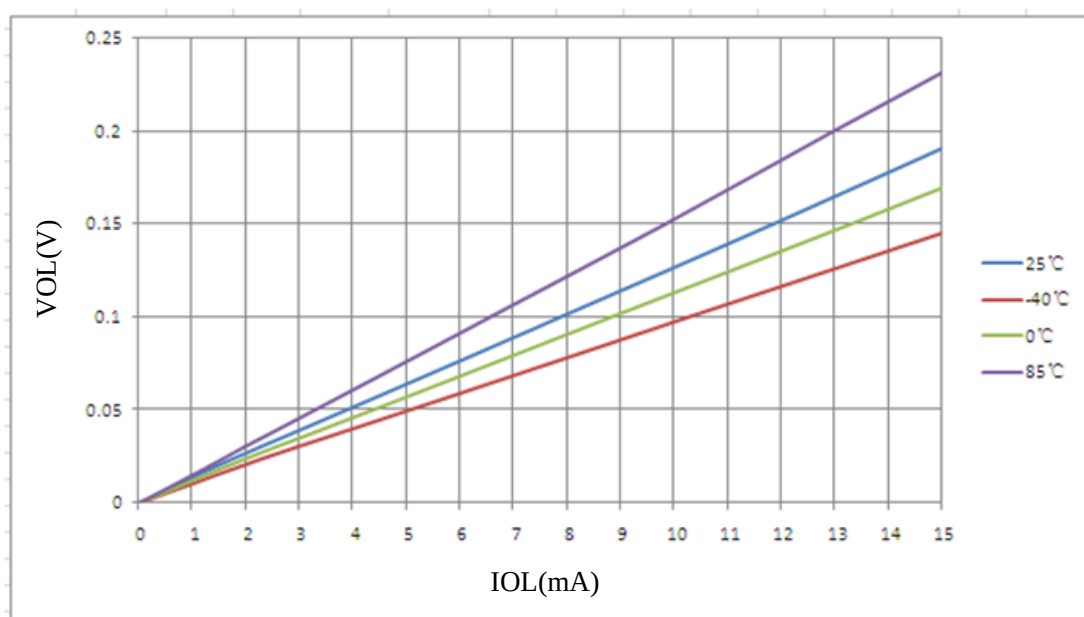


图15.10: 不同温度时VOL — IOL 关系曲线图 (VDD = 3.3V)

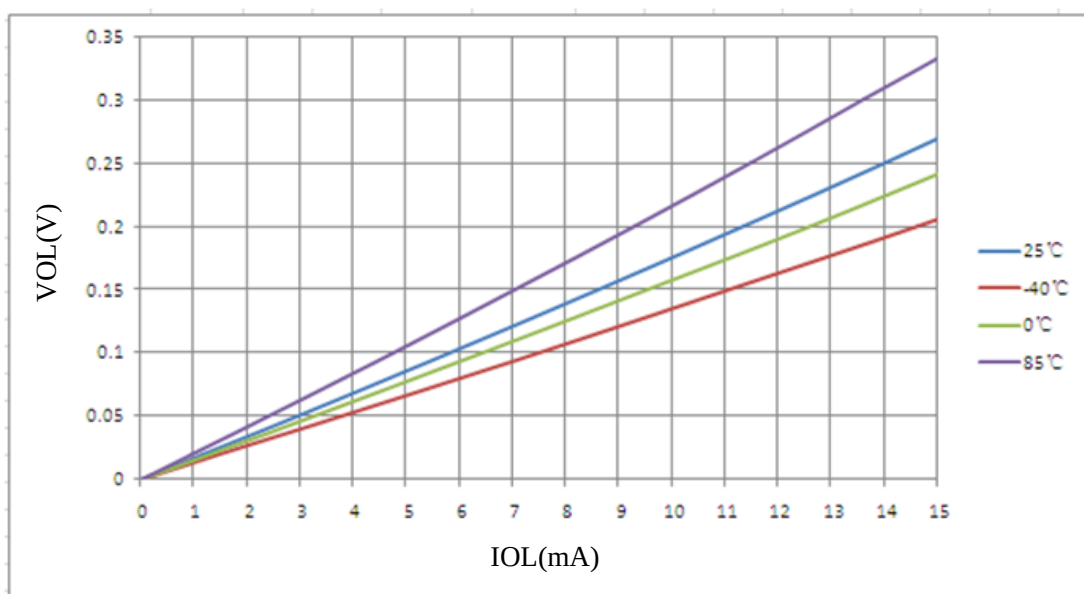


图15.11: 不同温度时VOP—IOP 关系曲线图 (VDD =5.0V)

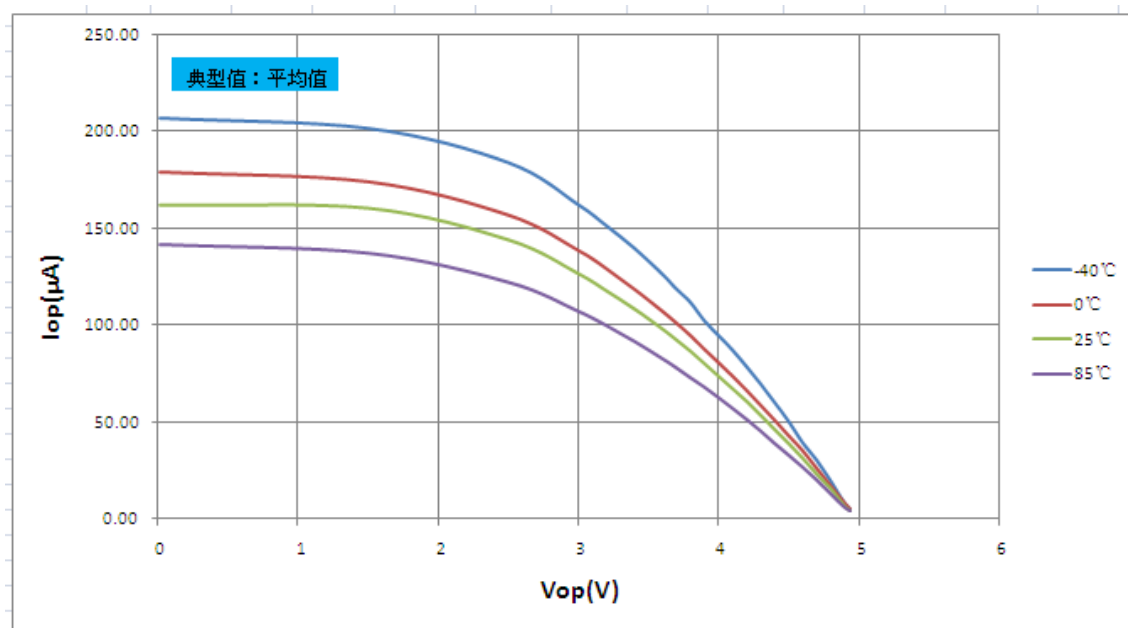
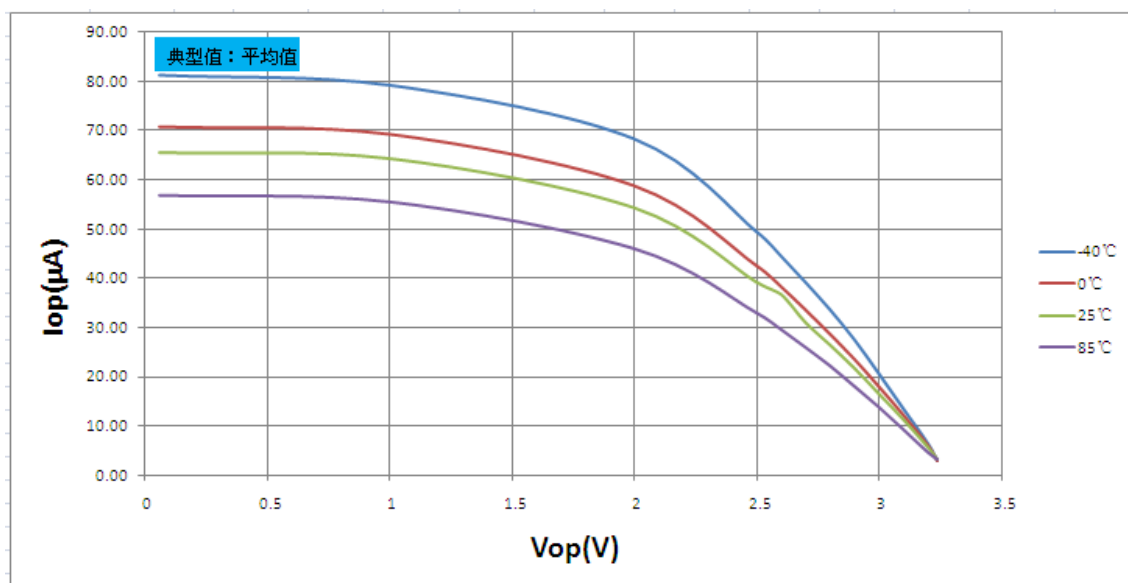
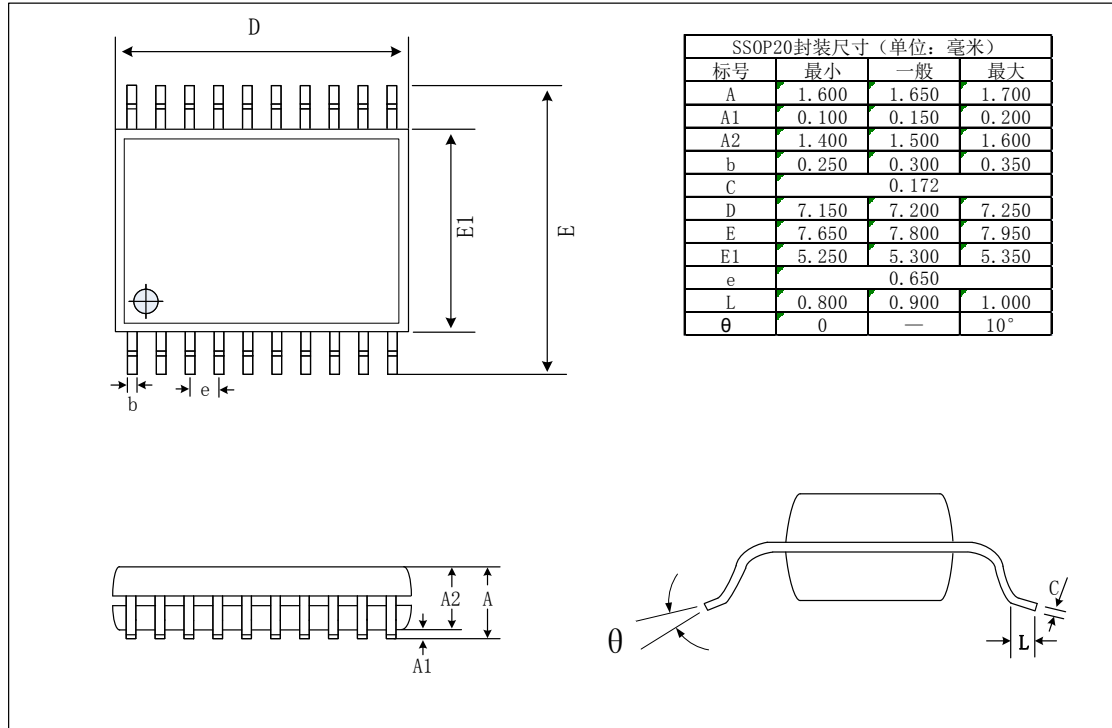


图 15.12: 不同温度时 VOP—IOP 关系曲线图 (VDD =3.3V)



16 封装信息

20 脚 SSOP 封装



17 附录 1 KF8L10Z08 SFR 地址映射及功能汇总

地址	名称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	复位初值
01H	T0	定时/计数器 0(T0)寄存器								xxxx xxxx
02H	PCL	程序计数器(PC)低字节								0000 0000
03H	PSW	-	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	CY	0001 1xxx
05H	P0	-	-	P05	P04	-	P02	P01	P00	-- xx xxxx
06H	P2	P27	P26	P25	P24	-	P22	P21	P20	xxxx xxxx
07H	P1	-	P16	P15	P14	P13	P12	P11	P10	xxxx xxxx
08H	C1FILTCTL	INV1	WEN1	FEN1	-	-	F1CNT2	F1CNT1	F1CNT0	0000 0000
09H	C1FILTPRE	FP17	FP16	FP15	FP14	FP13	FP12	FP11	FP10	0000 0000
0AH	PCH	-	-	-	程序计数器(PC)高字节					---0 0000
0BH	INTCTL	AIE/AIEH	PUIE/AIEL	T0IE	INT0IE	P0IE	T0IF	INT0IF	P0IF	0000 0000
0CH	EIF1	EEIF	ADIF	INT2IF	INT1IF	C1IF	PWM2IF	T2IF	T1IF	0000 0000
0DH	EIF2	T3IF	C2IF	-	-	SSC1IF	BCLIF	-	-	0000 0000
0EH	T1L	定时/计数器 T1 低字节寄存器								xxxx xxxx
0FH	T1H	定时/计数器 T1 高字节寄存器								xxxx xxxx
10H	T1CTL	-	T1GC	T1CKS1	T1CKS0	-	$\overline{T1SY}$	T1CS	T1ON	-000 -000
13H	PWM11	PWM11 占空比寄存器								xxxx xxxx
15H	PWMCTL	INT2SE	INT1SE	-	-	PWM22ON	PWM12ON	PWM21ON	PWM11ON	1100 0000
16H	PP1	PWM1 周期寄存器								1111 1111
19H	C1CTL	C1OE	C1M	C1P1	C1P0	-	-	C1OUT	C1EN	0000 0000
1AH	C2CTL	C2OE	-	-	-	-	-	C2OUT	C2EN	0000 0000
1CH	VRECAL1	内部参考电压校准值寄存器								0111 1111
1DH	ANSE2	ANS27	ANS26	ANS25	ANS24	-	ANS22	ANS21	ANS20	1111 1111
1EH	ADCDATAH	ADC 数据寄存器高字节								xxxx xxxx
1FH	ADCCCTL0	ADLR	CHS4	CHS3	CHS2	CHS1	CHS0	START	ADEN	0000 0000
21H	OPTR	$\overline{PUPH}$	INT0SE	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111
22H	IP0	-	-	-	-	-	PT0	PINT0	PP0	---- -000
23H	IP1	PEE	PADC	PINT2	PINT1	PC1	PPWM2	PT2	PT1	0000 0000
24H	IP2	PT3	PC2	-	-	PSSC1	PBCL	-	-	0000 0000
25H	TR0	-	-	TR05	TR04	-	TR02	TR01	TR00	1111 1111
26H	TR2	TR27	TR26	TR25	TR24	-	TR22	TR21	TR20	1111 1111
27H	TR1	-	TR16	TR15	TR14	TR13	TR12	TR11	TR10	1111 1111
2AH	VRECAL2	LFSEL	-	P18CAL5	P18CAL4	P18CAL3	P18CAL2	P18CAL1	P18CAL0	0110 0011
2BH	VRECTL1	-	-	-	-	VREOE	P18OE	VREEN	-	---0 0000
2CH	EIE1	EEIE	ADIE	INT2IE	INT1IE	C1IE	PWM2IE	T2IE	T1IE	0000 0000
2DH	EIE2	T3IE	C2IE	-	-	SSC1IE	BCLIE	-	-	0000 0000
2EH	PCTL	-	-	-	SLVREN	IPEN	SWDTEN	$\overline{POR}$	$\overline{LVR}$	---1 000x
2FH	OSCCCTL	CLKOE	IRCS2	IRCS1	IRCS0	SCS1	SCS0	IESO	FSCM	0010 0000
30H	OSCCAL0	晶振校准值寄存器 0								1000 0000
31H	ANSE1	-	ANS16	ANS15	ANS14	ANS13	ANS12	ANS11	ANS10	1111 1111
32H	PP2	PWM2 周期寄存器								1111 1111
33H	PWM21	PWM21 占空比寄存器								xxxx xxxx
35H	PUR0	-	-	PUR05	PUR04	-	PUR02	PUR01	PUR00	1111 -111
36H	IOCL	-	-	IOCL5	IOCL4	-	IOCL2	IOCL1	IOCL0	0000 0000
37H	OSCCAL1	晶振校准值寄存器 1								0000 -001
38H	NVMDATAH	NVMDATAH 数据高 8 位								0000 0000
39H	NVMDATAL	NVMDATAL 数据低 8 位								0000 0000
3AH	NVMADDRH	NVMADDRH 地址指针高 8 位								0000 0000
3BH	NVMADDRL	NVMADDRL 地址指针低 8 位								0000 0000
3CH	NVMCTL0	NVM 控制寄存器 0								---- x000
3DH	NVMCTL1	NVM 控制寄存器 1								---- ----

3EH	ADCDATAL	ADC 数据寄存器低字节								xxxx xxxx
3FH	ADCCTL1	-	ADCS2	ADCS1	ADCS0	VCFG1	VCFG0	ADTEST	ADCIM	-000 0000
40H	APFCTL1	-	-	SSCISEL	SSSEL	INT2SEL1	INT2SEL0	INT1SEL1	INT1SEL0	0000 0000
41H	ANSE0	-	-	ANS05	ANS04	-	ANS02	ANS01	ANS00	1111 1111
43H	SLPCTL	-	-	-	-	-	DSEN	-	-	0000 0000
44H	T2REL	T2 重载寄存器低 8 位								xxxx xxxx
45H	P0LR	-	-	P0LR5	P0LR4	-	P0LR2	P0LR1	P0LR0	xxxx xxxx
46H	P2LR	P2LR7	P2LR6	P2LR5	P2LR4	-	P2LR2	P2LR1	P2LR0	xxxx xxxx
47H	P1LR	-	P1LR6	P1LR5	P1LR4	P1LR3	P1LR2	P1LR1	P1LR0	xxxx xxxx
48H	T2REH	T2 重载寄存器高 8 位								xxxx xxxx
49H	T2L	T2 低 8 位寄存器								xxxx xxxx
4AH	T2H	T2 高 8 位寄存器								xxxx xxxx
4BH	T2CTL	T2REN	T2EXTHEN	T2CKS1	T2CKS0	T2EXTLEN	T2CS1	T2CS0	T2ON	0000 0000
4CH	OSCCAL2	晶振校准值寄存器 2								0011 1111
4EH	T3CTL	-	T3EXTHEN	T3CKS1	T3CKS0	T3EXTLEN	T3CS1	T3CS0	T3ON	0000 0000
4FH	T3L	T3 低位寄存器								0000 0000
50H	PUR1	-	PUR16	PUR15	PUR14	PUR13	PUR12	PUR11	PUR10	1111 1111
54H	T3CTL1	T3CLR	-	-	-	-	-	-	-	0111 1000
55H	T3CCPH	T3 捕捉寄存器高 8 位								0000 0000
56H	T3CCPL	T3 捕捉寄存器低 8 位								0000 0000
57H	T2CCPL	T2 捕捉寄存器低 8 位								0000 0000
58H	APFCTL2	-	-	-	-	CAT3SEL1	CAT3SEL0	CAT2SEL1	CAT2SEL0	---- 0000
5AH	T2CCPH	T2 捕捉寄存器高 8 位								0000 0000
5BH	APFCTL0	PWM22SEL1	PWM22SEL0	PWM21SEL1	PWM21SEL0	PWM12SEL1	PWM12SEL0	PWM11SEL1	PWM11SEL0	0000 0000
5CH	CCPCLR	-	-	-	-	-	-	T3CCPCLR	T2CCPCLR	0000 0000
5EH	PUR2	PUR27	PUR26	PUR25	PUR24	-	PUR22	PUR21	PUR20	1111 1111
5FH	T3H	T3 高位寄存器								0000 0000
60H	PWM12	PWM12 占空比寄存器								xxxx xxxx
61H	PWM22	PWM22 占空比寄存器								xxxx xxxx
62H	WDTPS	-	-	-	-	WDTPS3	WDTPS2	WDTPS1	WDTPS0	---- 0100
63H	OSCSTA	-	OSTS	HTS	LTS	-	-	SCF1	SCF0	0110 -00
64H	SSCIMSK	SSCIMSK7	SSCIMSK6	SSCIMSK5	SSCIMSK4	SSCIMSK3	SSCIMSK2	SSCIMSK1	SSCIMSK0	1111 1111
	SSCIADD	SSCI 的 I2C 地址寄存器								0000 0000
65H	SSCISTA	SAMPLE	CKEGE	SSCIDA	SSCISTOP	SSCISTART	SSCIRW	SSCIUA	SSCIBUF	0000 0000
66H	SSCICTL0	SSCIWCFL	SSCIOV	SSCIEN	SSCICKP	SSCIMOD3	SSCIMOD2	SSCIMOD1	SSCIMOD0	0000 0000
67H	SSCICTL1	SSCICALLEN	SSCIACKSTA	SSCIACKDAT	SSCIACKEN	SSCIRCEN	STOPEN	RESTARTEN	STARTEN	0000 0000
68H	SSCIBUFR	SSCI 数据接收缓冲/发送寄存器								0000 0000
69H	C2FILTCTL	INV2	WEN2	FEN2	-	-	F2CNT2	F2CNT1	F2CNT0	0000 0000
6AH	C2FILTPRE	FP27	FP26	FP25	FP24	FP23	FP22	FP21	FP20	0000 0000
6CH	RC32KCAL	内部低频振荡器频率校准寄存器								0000 0000
6DH	XTALCAL	XTAL4	XTAL3	XTAL2	XTAL1	XTAL0	LPRC2	LPRC1	LPRC0	0101 0100

注：“-”表示未用的存储单元 “x”表示不定

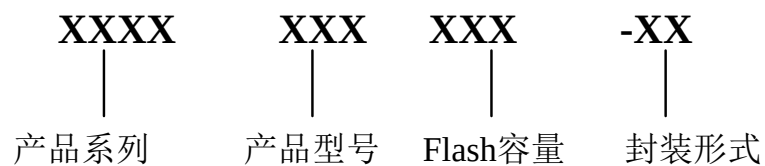
18 附录 2 汇编指令集

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

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

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

19 产品标识体系



产品系列: KF8L= KF 系列低功耗 8 位单片机

产品型号: 10Z = 10 系列

Flash 大小: 08 = 8Kbyte

封装形式: OG = SSOP20
 -I = 工作温度范围为-40~85℃

20 版本变更记录表

数据手册版本号	更新内容	更新日期

21 ROHS 认证

本产品已通过 ROHS 检测。

22 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