



# MC51F7084 用户手册

---

## 基于 8051 内核的 8 位增强型微控制器

Rev.1.2.0

本资料内容为晟矽微电子（以下简称“我公司”）版权所有。

我公司将力求本资料的内容做到准确无误，但同时保留在不通知用户的情况下，对本资料内容的修改权。如您需  
要获得最新的资料，请及时联系我公司。

我公司将尽最大努力为您提供高品质、高稳定性的产品。尽管如此，由于一般半导体器件的电气敏感性及其易受到  
外部物理损伤等固有特性存在，所以难免造成半导体器件出现故障或失效的可能。当您使用我公司产品时，有责任按  
照本资料以及相关资料中提到的“规则”来设计一个稳定及安全可靠的系统环境。另外，在遇到超规格（本资料中未  
描述到内容）的使用，请您提前咨询我公司，以免因我公司产品在一些特殊设备中或者特殊环境下的使用，导致财产  
损失、人员伤亡等严重后果的发生。因超规格的使用、未经咨询授权我公司产品的使用，我公司对此不承担任何法律  
责任。



## 目录

目录.....	2
1 主要特性.....	11
2 产品介绍.....	14
2.1 概述.....	14
2.2 系统框图.....	15
2.3 引脚排列.....	16
2.3.1 MC51F7084A0Y.....	16
2.3.2 MC51F7084A0M.....	16
2.3.3 MC51F7084A0ZQ.....	16
2.3.4 MC51F7084A0K.....	17
2.3.5 MC51F7084A0H.....	17
2.4 引脚说明.....	18
2.5 订购信息.....	20
2.6 基本概念和术语.....	21
3 电气参数.....	22
3.1 极限参数.....	22
3.2 推荐工作条件.....	22
3.3 直流特性参数.....	22
3.4 LVR 特性参数.....	24
3.5 ADC 特性参数.....	24
3.6 片上 FLASH 特性.....	25
3.7 交流电气参数.....	25
4 中央处理器.....	26
4.1 CPU 概述.....	26
4.1.1 程序状态寄存器 (PSW, 0xD0/SFR0).....	26
4.2 程序存储器 (FLASH ROM).....	27
4.3 附加数据区 (EEPROM).....	27
4.4 随机存储器 (RAM).....	27
4.5 存储器结构.....	27
4.6 寻址方式.....	28
4.7 功能寄存器映射.....	29
4.8 用户配置字.....	30
5 时钟与工作模式.....	34
5.1 系统时钟概述.....	34
5.2 内部高频 RC 振荡器.....	34
5.3 内部低频 RC 振荡器.....	35
5.4 外接晶体振荡器.....	35
5.5 时钟切换.....	35



5.6	振荡器预热和复位预热.....	36
5.7	工作模式.....	36
5.8	工作模式与时钟源状态.....	36
5.8.1	停止模式.....	36
5.8.2	休眠模式.....	37
5.9	高级能耗控制.....	37
5.10	相关寄存器.....	37
5.10.1	时钟控制寄存器 0 (CLKCR0, 0xC0/SFR0) .....	38
5.10.2	时钟控制寄存器 1 (CLKCR1, 0xC6/SFR0) .....	39
5.10.3	时钟安全寄存器 (OSCSS, 0xC5/SFR0) .....	39
5.10.4	电源控制寄存器 (PCON, 0x87/SFR0) .....	39
5.10.5	模式保护控制寄存器 (SLPCR, 0x86/SFR0) .....	40
5.10.6	高级能耗控制寄存器 (PWRCR, 0xF8/SFR0) .....	40
5.10.7	高级能耗控制寄存器 1 (PWRCR1, 0xF9/SFR0) .....	41
6	复位.....	43
6.1	复位条件.....	43
6.2	上电复位.....	44
6.3	外部复位.....	44
6.4	低电压复位.....	44
6.5	看门狗复位.....	44
6.5.1	看门狗定时器控制寄存器 (WDTCR, 0xCF/SFR0) .....	44
6.6	软件复位.....	45
6.6.1	复位标志寄存器 (RSTFLAG, 0xE8/SFR0) .....	45
6.7	复位标志寄存器.....	46
6.7.1	复位标志寄存器 (RSTFLAG, 0xE8/SFR0) .....	46
6.8	各种复位与低功耗模式的关系.....	47
7	I/O 端口.....	48
7.1	端口特性.....	48
7.2	端口相关寄存器.....	48
7.3	端口模式及数据控制.....	49
7.3.1	P0 端口模式寄存器 0 (P0MOD0, 0xC1/SFR0) .....	49
7.3.2	P1 端口模式寄存器 0 (P1MOD0, 0x91/SFR0) .....	49
7.3.3	P1 端口模式寄存器 1 (P1MOD1, 0x92/SFR0) .....	49
7.3.4	P2 端口模式寄存器 0 (P2MOD0, 0xA1/SFR0) .....	49
7.3.5	P2 端口模式寄存器 1 (P2MOD1, 0xA2/SFR0) .....	49
7.3.6	P0 端口数据寄存器 (P0, 0x80/SFR0) .....	50
7.3.7	P1 端口数据寄存器 (P1, 0x90/SFR0) .....	50
7.3.8	P2 端口数据寄存器 (P2, 0xA0/SFR0) .....	50
7.4	输入上/下拉电阻控制.....	50
7.4.1	P0 端口上拉电阻控制寄存器 (P0PU, 0xC3/SFR0) .....	50



7.4.2	P1 端口通用上拉电阻控制寄存器 (P1PU, 0x93/SFR0) .....	50
7.4.3	P2 端口上拉电阻控制寄存器 (P2PU, 0xA3/SFR0) .....	51
7.4.4	P0 端口下拉电阻控制寄存器 (P0PD, 0xC4/SFR) .....	51
7.4.5	P1 端口下拉电阻控制寄存器 (P1PD, 0x94/SFR) .....	51
7.4.6	P2 端口下拉电阻控制寄存器 (P2PD, 0xA4/SFR) .....	51
7.5	端口复用设置.....	52
8	定时器/计数器.....	53
8.1	定时器 T0/T1 .....	53
8.1.1	方式 0: 13 位计数器/定时器.....	53
8.1.2	方式 1: 16 位计数器/定时器.....	53
8.1.3	方式 2: 8 位自动重载计数器/定时器.....	54
8.1.4	方式 3: 两个 8 位计数器/定时器 (只限于定时器 T0) .....	54
8.2	T0/T1 相关寄存器.....	55
8.2.1	定时器模式寄存器 (TMOD, 0x89/SFR0) .....	55
8.2.2	定时器控制寄存器 (TCON, 0x88/SFR0) .....	56
8.2.3	定时器控制寄存器 1 (TCON1, 0x8E/SFR0) .....	56
8.2.4	定时器计数高位寄存器 0/1 (TH0, 0x8C/SFR0), (TH1, 0x8D/SFR0) .....	57
8.2.5	定时器计数低位寄存器 0/1 (TL0, 0x8A/SFR0), (TL1, 0x8B/SFR0) .....	57
8.3	定时器 T2.....	57
8.3.1	方式 0: 16 位重载方式定时器/计数器方式.....	57
8.3.2	方式 1: 16 位输入捕获方式.....	58
8.3.3	方式 3: 16 位输出比较方式.....	59
8.4	T2 相关寄存器 .....	60
8.4.1	T2 控制寄存器 (T2CON, 0xC8/SFR0) .....	61
8.4.2	T2 工作模式寄存器 (T2MOD, 0xC9/SFR0) .....	62
8.4.3	T2 控制寄存器 1 (T2CON1, 0xCE/SFR0) .....	62
8.4.4	T2 计数高位寄存器 (TH2, 0xCD/SFR0) .....	63
8.4.5	T2 计数低位寄存器 (TL2, 0xCC/SFR0) .....	63
8.4.6	T2 重载/捕获高位寄存器 (RCAPH2, 0xCB/SFR0) .....	63
8.4.7	T2 重载/捕获低位寄存器 (RCAPL2, 0xCA/SFR0) .....	63
8.5	定时器 T3.....	64
8.5.1	方式 0: 系统时钟 (同步时钟) 模式 .....	64
8.5.2	方式 1: 低频时钟 (异步时钟) 模式 .....	64
8.5.3	实时读取计数器值.....	65
8.6	T3 相关寄存器 .....	65
8.6.1	T3 控制寄存器 (T3CR, 0x95/SFR0) .....	65
8.6.2	T3 重载高位寄存器 (TH3, 0x97/SFR0) .....	66
8.6.3	T3 重载低位寄存器 (TL3, 0x96/SFR0) .....	66
8.6.4	T3 计数高位寄存器 (T3CNTH, 0xE7/SFR0) .....	67
8.6.5	T3 计数低位寄存器 (T3CNTL, 0xE6/SFR0) .....	67



9	通用型 PWM0	68
9.1	PWM0 概述	68
9.2	PWM0x 结构图	68
9.3	PWM0 相关寄存器	68
9.3.1	PWM0 控制寄存器 0 (PWM0CR0, 0xFF68/XSFR)	69
9.3.2	PWM0 周期设置寄存器 (PWM0P, 0xFF69/XSFR)	70
9.3.3	PWM0A 占空比设置寄存器 (PWM0AD, 0xFF6A/XSFR)	70
10	增强型 PWM1 阵列	71
10.1	概述	71
10.2	PWM 时基模块	71
10.2.1	PWM 工作时钟	72
10.2.2	PWM 时基工作方式	72
10.2.3	后分频	74
10.2.4	时基中断	74
10.2.5	PWM 周期寄存器的自动重载	75
10.3	PWM 波形发生模块	75
10.3.1	PWM 输出模式	75
10.3.2	边沿对齐计数模式下的 PWM1A/B/C 原始波形	75
10.3.3	中心对齐计数模式下的 PWM1A/B/C 原始波形	76
10.3.4	单次计数模式下的 PWM1A/B/C 原始波形	77
10.3.5	占空比寄存器的自动重载	77
10.3.6	占空比中断	78
10.4	PWM 死区与极性控制模块	78
10.4.1	PWM 死区的实现	79
10.4.2	PWM 极性设置	79
10.4.3	PWM 死区的自动重载	79
10.4.4	不同极性设置下带死区的 PWM 波形	80
10.5	PWM1A/B/C 输出控制与刹车检测	81
10.5.1	PWM 刹车信号输入源的选择	81
10.5.2	PWM 刹车输入信号滤波	82
10.5.3	PWM 刹车保护模式	82
10.5.4	PWM 输出控制模块	82
10.6	事件触发功能	83
10.6.1	PWM 计数器触发 Timer2	83
10.6.2	PWM 计数器触发 ADC	84
10.7	PWM 模块的中断总汇	84
10.8	注意事项	84
10.8.1	边沿对齐计数模式中占空比寄存器值为 0 或周期值	84
10.8.2	中心对齐计数模式中占空比寄存器值为 0 或周期值	87
10.8.3	周期寄存器值为 0	89



10.8.4	仿真状态的 PWM 输出.....	90
10.8.5	进入 STOP 模式, PWM1A/B/C 的波形输出状态.....	90
10.8.6	进入 SLEEP 模式, PWM1A/B/C 的波形输出状态.....	90
10.9	代码示例.....	90
10.10	增强型 PWM1 寄存器.....	91
10.10.1	PWM1 控制寄存器 0 (PWM1CR0, 0xFF48/XSFR).....	92
10.10.2	PWM1 控制寄存器 1 (PWM1CR1, 0xDA/SFR0).....	93
10.10.3	PWM1 控制寄存器 2 (PWM1CR2, 0xDB/SFR0).....	93
10.10.4	PWM1 周期寄存器高位 (PWM1PH, 0xFF4A/XSFR).....	94
10.10.5	PWM1 周期寄存器低位 (PWM1PL, 0xFF4B/XSFR).....	94
10.10.6	PWM1A 占空比寄存器低位 (PWM1ADL, 0xD2/SFR0).....	94
10.10.7	PWM1A 占空比寄存器高位 (PWM1ADH, 0xD3/SFR0).....	94
10.10.8	PWM1B 占空比寄存器低位 (PWM1BDL, 0xD4/SFR0).....	95
10.10.9	PWM1B 占空比寄存器高位 (PWM1BDH, 0xD5/SFR0).....	95
10.10.10	PWM1C 占空比寄存器低位 (PWM1CDL, 0xD6/SFR0).....	95
10.10.11	PWM1C 占空比寄存器高位 (PWM1CDH, 0xD7/SFR0).....	95
10.10.12	PWM1 输出使能寄存器 (PWM1OE, 0xD1/SFR0).....	95
10.10.13	PWM1 手动输出控制寄存器 0 (PWM1MANCR0, 0xDD/SFR0).....	96
10.10.14	PWM1 手动输出控制寄存器 1 (PWM1MANCR1, 0xDE/SFR0).....	96
10.10.15	PWM1 刹车控制寄存器 (PWM1FBCR, 0xDC/SFR0).....	96
10.10.16	PWM1 中断使能控制寄存器 (PWM1IE, 0xFF49/XSFR).....	97
10.10.17	PWM1 中断标志寄存器 (PWM1IF, 0xD8/SFR0).....	98
10.10.18	PWM1 死区控制寄存器 0 (PWM1DT0, 0xFF4D/XSFR).....	99
10.10.19	PWM1 寄存器修改重载解锁寄存器 (PWM1RLEN, 0xD9/SFR0).....	99
11	模数转换器 ADC.....	100
11.1	ADC 概述.....	100
11.2	ADC 转换时序.....	100
11.3	ADC 常规操作步骤.....	100
11.4	ADC 比较功能.....	101
11.5	ADC 连续采样.....	101
11.6	ADC 增益配置.....	101
11.7	ADC 相关寄存器.....	101
11.7.1	ADC 控制寄存器 0 (ADCCR0, 0xF1/SFR0).....	102
11.7.2	ADC 控制寄存器 1 (ADCCR1, 0xF2/SFR0).....	103
11.7.3	ADC 控制寄存器 2 (ADCCR2, 0xF3/SFR0).....	104
11.7.4	ADC 控制寄存器 3 (ADCCR3, 0xF4/SFR0).....	105
11.7.5	ADC 端口配置寄存器 1 (ADRIOS1, 0xFD/SFR0).....	105
11.7.6	定时器控制寄存器 (TCON, 0x88/SFR0).....	105
11.7.7	ADC 转换结果高位寄存器 (ADRH, 0xFF/SFR0).....	106
11.7.8	ADC 转换结果低位寄存器 (ADRL, 0xFE/SFR0).....	106



11.7.9	ADC 转换结果高位寄存器 (ADRH, 0xFF/SFR0)	106
11.7.10	ADC 转换结果低位寄存器 (ADRL, 0xFE/SFR0)	106
11.7.11	零点偏移修正寄存器 (OSADJCR, 0xF5/SFR0)	106
11.7.12	ADC 数字比较高位寄存器 (ADCMPLH, 0xFB/SFR0)	107
11.7.13	ADC 数字比较低位寄存器 (ADCMPL, 0xFA/SFR0)	107
11.8	ADC 零点偏移修调步骤	107
12	低电压检测 LVD	109
12.1	概述	109
12.2	LVD 相关寄存器	109
12.2.1	LVD 控制寄存器 (LVDCR, 0xF6/SFR0)	109
12.2.2	LVD 控制寄存器 1 (LVDCR1, 0xF7/SFR0)	110
13	增强型 UART	111
13.1	概述	111
13.2	UART0	111
13.2.1	方式 0: 8 位同步半双工	111
13.2.2	方式 1: 8 位异步全双工, 可变波特率	112
13.2.3	方式 2: 9 位异步全双工, 固定波特率	114
13.2.4	方式 3: 9 位异步全双工, 可变波特率	115
13.2.5	波特率计算	116
13.2.6	多机通讯	117
13.2.7	帧出错检测	118
13.3	UART 相关寄存器	118
13.3.1	电源控制寄存器 (PCON, 0x87/SFR0)	119
13.3.2	串口 0 控制寄存器 (SCON0, 0x98/SFR0)	119
13.3.3	串口 0 缓存寄存器 (SBUF0, 0x99/SFR0)	120
13.3.4	串口 0 从机地址寄存器 (SADDR, 0x9A/SFR0)	121
13.3.5	串口 0 地址掩码寄存器 (SADEN, 0x9B/SFR0)	121
13.3.6	串口 0 波特率控制寄存器 (BRGCR, 0x9D/SFR0)	121
13.3.7	串口 0 波特率计数器低位寄存器 (BRGH, 0x9F/SFR0)	121
13.3.8	串口 0 波特率计数器高位寄存器 (BRGL, 0x9E/SFR0)	121
13.3.9	串口 0 控制寄存器 2 (SCON2, 0x9C/SFR)	122
14	串行外设接口 SPI	123
14.1	SPI 概述	123
14.2	SPI 信号描述	123
14.3	工作模式	123
14.3.1	主机模式	123
14.3.2	从机模式	124
14.4	数据传输格式	124
14.5	出错检测	125
14.6	SPI 相关寄存器	125



14.6.1	SPI 控制寄存器 (SPCON, 0xAD/SFR0)	126
14.6.2	SPI 状态寄存器 (SPSTA, 0xAE/SFR0)	127
14.6.3	SPI 数据寄存器 (SPDAT, 0xAF/SFR0)	127
15	IIC 通讯接口	128
15.1	IIC 概述	128
15.2	IIC 数据传输	128
15.2.1	主机到从机的数据传输	128
15.2.2	从机到主机的数据传输	128
15.3	IIC 工作模式	129
15.3.1	主机模式	129
15.3.2	从机模式	129
15.4	IIC 时序说明	130
15.5	IIC 多机通讯	131
15.6	IIC 主从机地址	131
15.7	IIC 相关寄存器	132
15.7.1	IIC 控制寄存器 0 (IICCR0, 0xE1/SFR0)	132
15.7.2	IIC 控制寄存器 1 (IICCR1, 0xE2/SFR0)	133
15.7.3	IIC 状态寄存器 (IICSR, 0xE3/SFR0)	133
15.7.4	IIC 地址寄存器 (IICAR, 0xE4/SFR0)	134
15.7.5	IIC 数据寄存器 (IICDR, 0xE5/SFR0)	134
15.7.6	IIC 辅地址寄存器 (IICAAR, 0xE6/SFR0)	135
15.7.7	IIC 辅地址掩码寄存器 (IICAMR, 0xE7/SFR0)	135
16	中断	136
16.1	中断源列表	136
16.2	中断优先级	137
16.3	中断处理流程	137
16.4	外部中断	137
16.4.1	外部中断使能寄存器 (EXTEN, 0xA6/SFR0)	138
16.4.2	外部中断使能寄存器 1 (EXTEN1, 0xA5/SFR0)	138
16.4.3	外部中断控制寄存器 2 (EXTIS2, 0xB5/SFR0)	138
16.4.4	外部中断控制寄存器 1 (EXTIS1, 0xB6/SFR0)	139
16.4.5	外部中断控制寄存器 0 (EXTIS, 0xB7/SFR0)	139
16.5	中断相关寄存器	139
16.5.1	中断使能寄存器 (IEN0, 0xA8/SFR0)	140
16.5.2	中断使能寄存器 1 (IEN1, 0xAA/SFR0)	141
16.5.3	中断使能寄存器 2 (IEN2, 0xAB/SFR0)	141
16.5.4	中断优先级控制低位寄存器 (IPL0, 0xB8/SFR0)	141
16.5.5	中断优先级控制 1 低位寄存器 (IPL1, 0xBB/SFR0)	142
16.5.6	中断优先级控制 2 低位寄存器 (IPL2, 0xBD/SFR0)	142
17	循环冗余校验 CRC	143



17.1	CRC 特性.....	143
17.2	CRC 相关寄存器 .....	143
17.2.1	CRC 控制寄存器 (CRCCR, 0xFF60/XSFR) .....	143
17.2.2	CRC 结果低字节寄存器 (CRCL, 0xFF63/XSFR) .....	144
17.2.3	CRC 结果高字节寄存器 (CRCH, 0xFF62/XSFR) .....	144
17.2.4	CRC 起始地址低字节寄存器 (CRCSTL, 0xFF65/XSFR) .....	145
17.2.5	CRC 起始地址高字节寄存器 (CRCSTH, 0xFF64/XSFR) .....	145
17.2.6	CRC 结束地址低字节寄存器 (CRCEDL, 0xFF67/XSFR) .....	145
17.2.7	CRC 结束地址高字节寄存器 (CRCEDH, 0xFF66/XSFR) .....	145
18	OCD 调试和 ICP 接口.....	147
19	片上编程 IAP .....	149
19.1	IAP 相关寄存器.....	150
19.1.1	IAP 模式寄存器 (IAPMD, 0xEE/SFR0) .....	150
19.1.2	IAP 写保护寄存器 (IAPKEY, 0xEF/SFR0) .....	151
19.1.3	IAP 高 8 位数据寄存器 (IAPDBH, 0xED/SFR0) .....	151
19.1.4	IAP 低 8 位数据寄存器 (IAPDBL, 0xEC/SFR0) .....	151
19.1.5	IAP 高地址寄存器 (IAPADH, 0xEB,SFR0) .....	152
19.1.6	IAP 低地址寄存器 (IAPADL, 0xEA/SFR0) .....	152
19.1.7	IAP 控制寄存器 (IAPCR, 0xE9/SFR0) .....	152
19.1.8	IAP 控制寄存器 1 (IAPCR1, 0xDF/SFR0) .....	153
19.2	IAP 命令汇总.....	154
19.3	IAP 操作标准流程.....	155
19.3.1	EEPROM 读取.....	155
19.3.2	FLASH 读取.....	155
19.3.3	EEPROM 字节擦除 (1 字节) .....	155
19.3.4	FLASH 页擦除 (64 字节) .....	156
19.3.5	EEPROM 写入 (1 字节) .....	156
19.3.6	FLASH 写入 (n*2 字节) .....	156
19.4	Bootloader 区.....	157
19.4.1	启动选择.....	157
19.4.2	IAP 控制寄存器 1 (IAPCR1, 0xED /SFR0) .....	157
19.4.3	Bootloader 区内的擦除/读写操作.....	157
19.4.4	bootloader 应用.....	158
19.5	INFO 区.....	158
19.5.1	加密控制字解锁 .....	158
19.5.2	Flash 加密/解密操作 .....	158
19.5.3	用户/工具配置区操作 .....	159
19.6	用户主程序区 (FLASH) .....	159
19.7	User option 控制.....	159
19.8	复位说明 .....	159



---

19.9	ICP、IAP 和 ISP.....	160
20	附录.....	161
20.1	指令集总览表.....	161
20.2	特殊功能寄存器总览表 (SFR0) .....	165
20.3	特殊功能寄存器总览表 (XSFR) .....	172
21	封装信息 .....	175
21.1	TSSOP20 (0.65mm pitch) .....	175
21.2	SOP20 (1.27mm pitch) .....	175
21.3	QFN20 (3 x 3 x 0.75) .....	176
21.4	SOP16 (1.27mm pitch) .....	176
21.5	SOP8 (1.27mm pitch) .....	177
22	修订记录.....	178



# 1 主要特性

## CPU

- ◇ 1T 高速增强型 8051 内核
- ◇ 系统时钟可软件 (1/2/4/8/16/32/64/128) 分频

## 存储器

- ◇ 片上 16K 字节 Flash, 擦写 1 万次以上
- ◇ 1024 (256 data+768 xdata) 字节 RAM
- ◇ 片上 256 字节 EEPROM, 擦写 10 万次以上
- ◇ 支持代码分区保护功能 (有效防止非法读/写/擦)
- ◇ 支持 Flash 和 E2 在电路编程 (ICP) 和应用编程 (IAP)

- ◇ 支持 bootloader (ISP) 功能
- ◇ 片上 2K 字节 bootloader ROM

## 时钟源

- ◇ 片上高精度 32MHz 高频 RC 振荡器 (CPU 最高运行主频 16MHz)
- ◇ 片上 32KHz 超低功耗低频 RC 振荡器, 供看门狗定时器 T3 使用
- ◇ 外部可选接 32768Hz 晶体振荡器

## 电源管理

- ◇ 工作电压 2.5V~5.5V
- ◇ 4 种工作模式: 高速/低速/停止/休眠
- ◇ 增加高级能耗控制功能, 满足用户低功耗的需求

## 复位

- ◇ 支持 5 种复位类型: 上电复位 (POR/PDR)、外部复位 (nRST 引脚)、低电压复位 (LVR)、看门狗复位 (WDT)、软件复位
- ◇ LVR 支持电压 3 级可选: 2.5V、3.5V、4.1V, 滤波时间可配置
- ◇ 看门狗复位, 可选 8 种溢出时间

## I/O

- ◇ 最多 18 个双向通用 I/O 口 (20PIN 封装下)
- ◇ 支持 4 种模式 (输入/推挽输出/开漏输出/高阻), 内部上拉和下拉电阻独立控制

## 定时器/计数器

- ◇ 2 个 16 位定时器 T0/T1, 兼容标准 8051
- ◇ 1 个 16 位增强型定时器 T2, 兼容 8052 的 T2, 支持输入捕获和输出比较, 并支持 PWM1 触发计数
- ◇ 1 个 16 位时基定时器 T3, 支持外部 32768Hz 晶振, 并支持停止/休眠模式下溢出唤醒
- ◇ T0~T2 计数溢出可触发 ADC 采样 (T3 不支持)

## PWM 定时器

- ◇ 8 位 PWM0, 可输出 1 路周期独立占空比的 PWM, 可用作通用定时器
- ◇ 12 位 PWM1 阵列, 支持 3 路独立 PWM 通道, 并可扩展为 3 对互补 PWM 输出, 可统一调整互补输出死区时间
- ◇ PWM1 具有故障保护中断功能
- ◇ PWM1 支持三种时基计数模式: 边沿对齐计数、中心对齐计数和单次计数模式

- ◇ PWM1 提供 PWM 周期匹配、归零匹配和占空比较匹配中断
- ◇ PWM1 支持周期匹配、归零匹配信号后分频功能
- ◇ PWM1 提供寄存器重载使能位以保证寄存器重载同步
- ◇ PWM1 计数周期、归零和占空比触发 ADC 采样
- ◇ PWM1 支持触发 T2 开始计数

## 12 位高精度 ADC

- ◇ 12 位高精度逐次逼近型 ADC
- ◇ 12 通道: AIN0~AIN9 外部 10 通道+内部 3 通道
- ◇ 参考电压可选: 内部 2.0V、VDD、Vref 引脚输入
- ◇ 时钟源为系统时钟, 可配置 1/2/4/8/16/32/64/128 分频

## 1 路 UART

- ◇ 1 路 UART 模块
- ◇ 增强 UART0 支持“帧出错”检测及自动地址识别
- ◇ 支持 8 位同步半双工、8 位/9 位异步全双工等 4 种工作方式
- ◇ 自带波特率发生器

## 1 路 SPI

- ◇ 支持全双工, 3 线/4 线同步模式, 主/从机可选
- ◇ 支持主机模式错误用以防止主机冲突

## 1 路 IIC

- ◇ 支持 7 位地址编码的主从模式
- ◇ 主机模式, 通讯速率支持 100Kbps 和 400Kbps
- ◇ 从机模式, 通讯速率自动与主机同步, 最高支持 400Kbps

## 低电压检测 LVD

- ◇ 2.2V/2.4V/2.6V/2.8V/3.0V/3.2V/3.6V/4.0V
- ◇ 滤波时间可选择
- ◇ 支持 LVD 中断与唤醒

## 中断

- ◇ INT0X、T0、INT1X、T1、UART0、ADC、T2、T3、SPI、PWM1、PWM0、PWM1FB、IIC、CRC、LVD 共 15 个中断源
- ◇ 2 级中断优先级可设
- ◇ 其中 INTnX (n=0~1, X=0~4) 支持多重映射输入, 5 选 1 分别对应两个中断源 (INT0/INT1); 支持上升沿触发方式和高电平脉宽测量功能

循环冗余校验算法模块 (符合 CRC-16 标准)

## 双两线调试与编程接口

- ◇ 支持最新的两线 (Two-wire) 调试和编程接口
- ◇ 三组调试和编程接口任意三选一, 支持自动识别

## 96-bit unique ID

## 工作环境温度

- ◇ -40°C~85°C

## 封装形式

- ◇ TSSOP20/SOP20/QFN20/SOP16/SOP8



## 芯片对比

模块/功能		MC51F003A4	MC51F7084
Core	时钟	高频 2/4/16/32 配置分频	时钟源切换和分频 均由寄存器控制
存储器	ROM	16K	16K
	BootLoad	-	2K 独立存储 (占用 FLASH, 此时 FLASH 只有 14K)
	SRAM	256+256	256+768 (data+xdata)
	SFR 地址映射	SFR0+SFR1	SFR0+XSFR
	EEPROM	1024	256
时钟源	HIRC	32M 系统时钟为 16M (32M/2)	32M 系统时钟为 16M (32M/2)
	LIRC	32K	32K
	EXT	32768	32768
复位	复位	POR/LVR/RST/WDT/SOFT	POR/LVR/RST/WDT/SOFT
	LVR 电压	2.1V/2.5V/3.5V/4.1V	2.5V/3.5V/4.1V
I/O	数量	18	18
	模式	4 种 (高速/低速/停止/休眠)	4 种 (高速/低速/停止/休眠)
	上/下拉	全上拉	全上/下拉, P1/P2 各有 2 组上拉 (IIC 专用)
	数字全映射	不支持	不支持
	外部中断	2×5	2×5
定时器	16 位 T0/T1	√	支持触发 ADC 采样
	16 位 T2	√	支持触发 ADC 采样
	16 位 T3	√	√
PWM	PWM0	1 路	1 路
	PWM1	16 位 3 对互补输出	12 位 3 对互补输出 支持电机控制
ADC	输入通道	AN0~AN9, VDD/4, GND	AN0~AN9, 2.0V, VDD, GND
	参考电压	VDD+2.0V+VER	VDD+2.0V+Vref
LVD		无	有 (共 8 档)
通讯	UART	UART0/UART1	UART0 (支持自带波特率发生器) 支持 5 组 IO 映射
	SPI	1 路	1 路
	IIC	无	1 路 (两组专用上拉配套) 支持 4 组 IO 映射



模块/功能		MC51F003A4	MC51F7084
其他	FLASH 编程	ICP	ICP, IAP, ISP
	软件 LCD	支持	不支持
	unique ID	不支持	支持
	功耗	-	同 MC51F003A4
	工作温度	-40°C~85°C	-40°C~85°C
	2 线烧写/仿真口	2 组	3 组
	内核性能	-	全新 8051 内核，效能更高



## 2 产品介绍

### 2.1 概述

本产品是一款高速低功耗 1T 周期 8051 内核 8 位增强型 FLASH 微控制器芯片，较传统 8051 而言，运行更快、效率更高。

最高运行频率为 16MHz，片上集成了 16K 字节 FLASH ROM、256 字节 EEPROM 和 1024 字节 RAM（内部 RAM 256B 加上外部 RAM 768B），内置 1 高频 RC 振荡器和 1 个低频 RC 振荡器，1 组外部晶体振荡器驱动接口 32768Hz 晶体，最大包含 18 个双向 GPIO 口，集成灵活的三选一调试和编程接口，具有 4 个 16 位定时器、1 路 8 位独立 PWM 输出，3 对 16 位共周期的互补 PWM 输出（带有刹车保护功能），集成了 1 个高精度 12 位 ADC 模块，1 路 UART，以及 1 路 SPI 和 1 路 IIC，同时内部集成 4 级可选的 LVR、8 档可选的 LVD、看门狗定时器、CRC 校验等容错功能，极大地提高了芯片的可靠性。

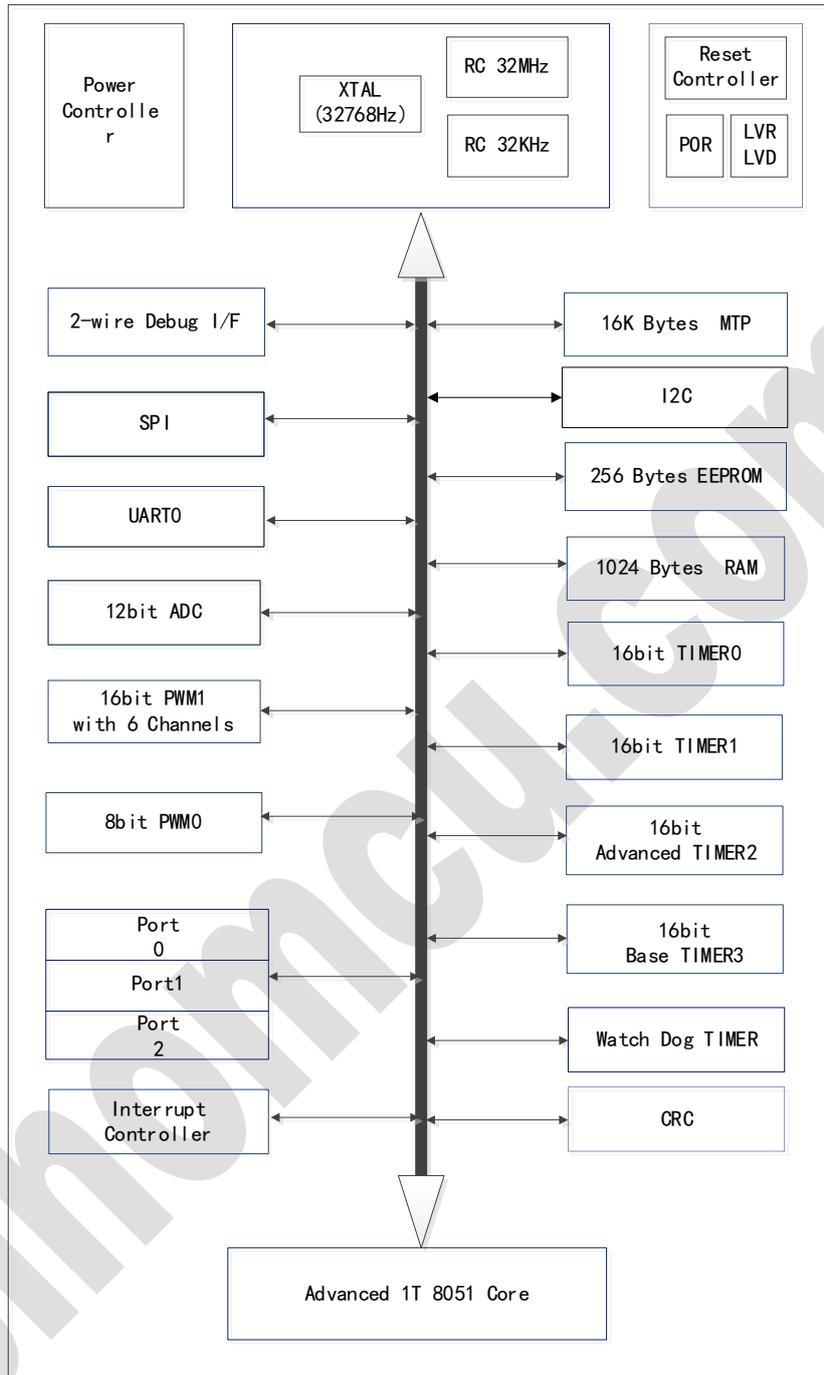
本产品工作电压为 2.5V 至 5.5V，工作温度为 -40°C 至 +85°C。支持一系列的省电模式保证低功耗应用的要求。

可提供包括 TSSOP20 在内的多种紧凑型封装形式。

本产品具有高抗干扰性和低功耗等特性，内部集成了高精度 ADC、兼容 51/52 的标准定时器、多通道 PWM、以及 UART/SPI/IIC 等通讯模块，简化了系统设计，降低了整体成本，可广泛用于家电、工控、便携设备、物联网等多种智能控制应用领域。



## 2.2 系统框图



注 1: 引脚图中, 方括号 “[ ] ” 中端口, 为烧写/调试专用口; 圆括号 “ ( ) ” 中端口, 其复用功能需在 “用户配置字” 中使能; “nXXX” 端口, n 表示该复用功能低电平有效;

注 2: 原则上同一引脚同一时刻只可选择一种功能使用, 当一个引脚同时使能了多种复用功能时, 其中优先级最高的功能有效, 其他功能则被屏蔽。即使如此, 用户在应用中也需避免同时使能引脚上的相互冲突的复用功能, 以防止发生芯片未预见的功能冲突;

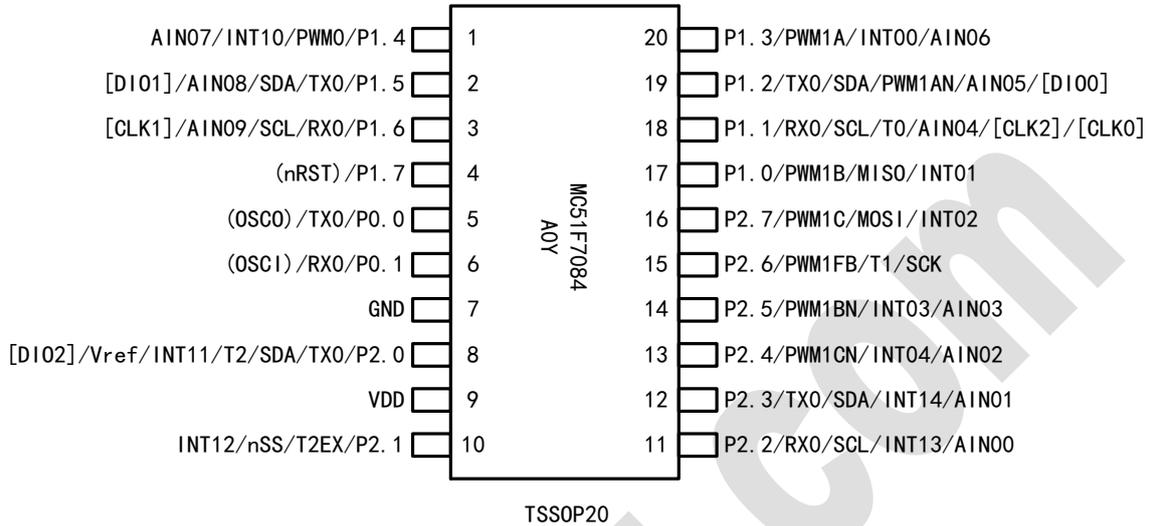
注 3: 引脚图中, 未标明的全映射数字端口为定时器相关端口 (T0/T1/T2/T2EX)、PWM 相关端口 (PWM0A/PWM0A/PWM1A/PWM1B/PWM1C/PWM1AN/PWM1BN/PWM1CN/PWM1FB)、通讯模块相关端口



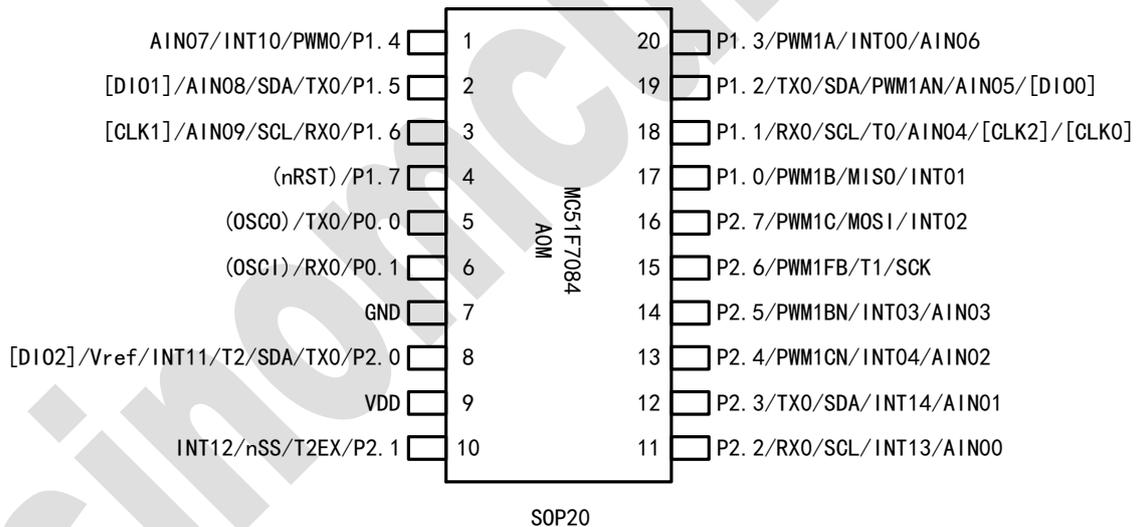
(TX0/RX0/TX1/RX1/nSS/SCK/MISO/MOSI/SCL/SDA)，详见后续“数字功能端口全映射”章节描述；

## 2.3 引脚排列

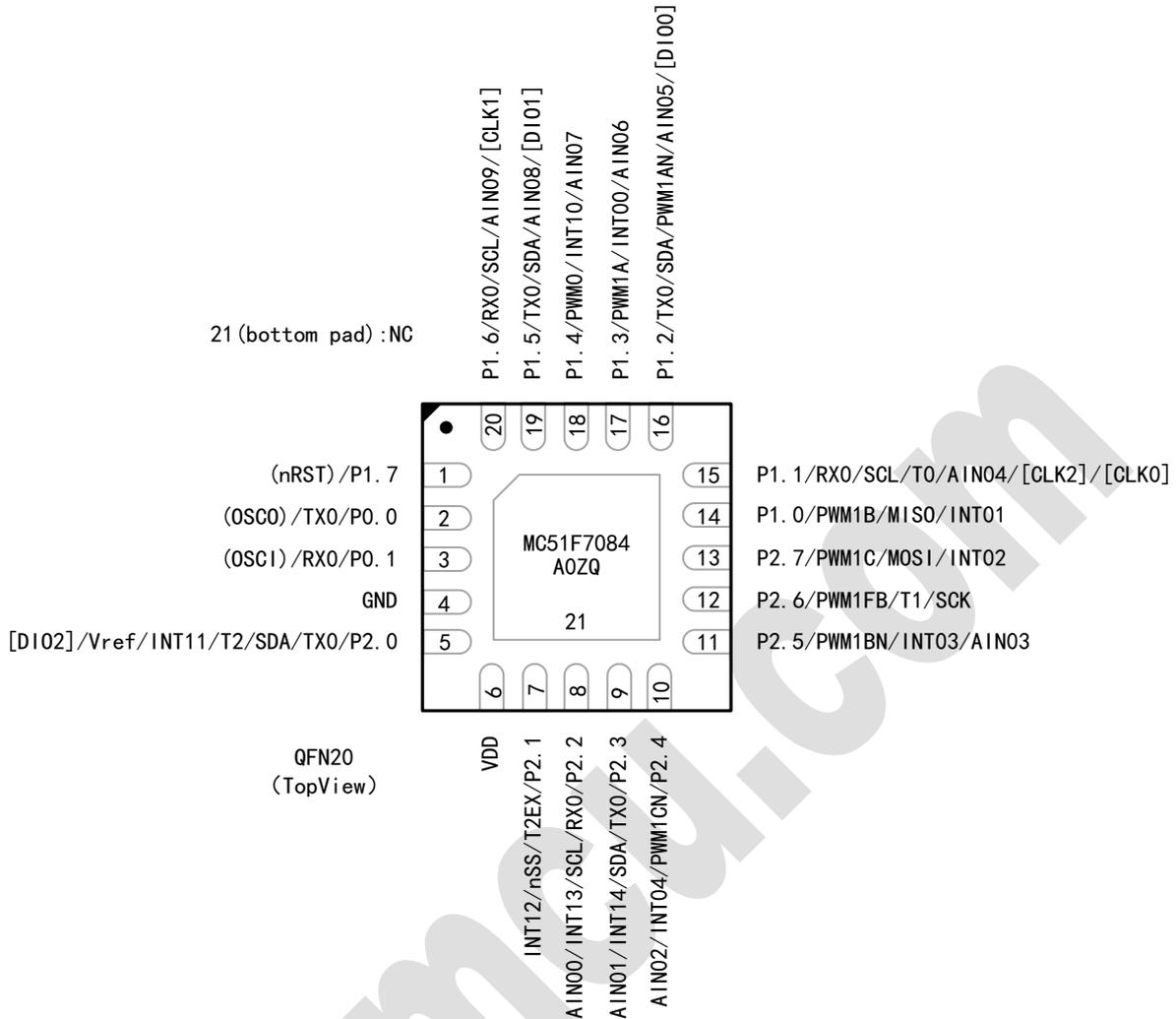
### 2.3.1 MC51F7084A0Y



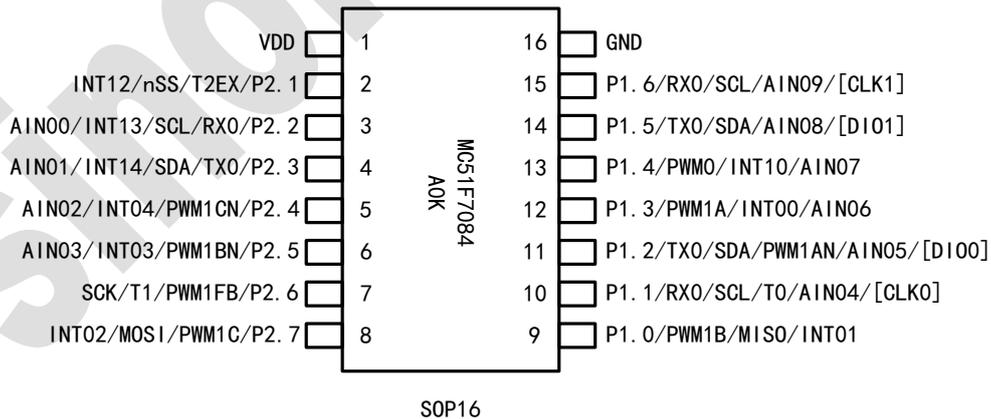
### 2.3.2 MC51F7084A0M



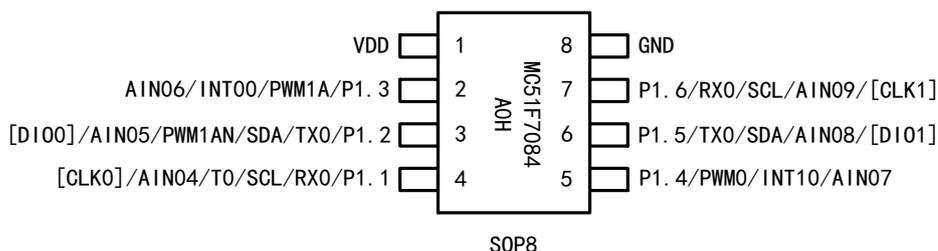
### 2.3.3 MC51F7084A0ZQ



### 2.3.4 MC51F7084A0K



### 2.3.5 MC51F7084A0H



注 1: 图中, 对于同一引脚的复用功能, 写在最外面的引脚名具有最高优先级, 优先级逐次递减, 最内侧的优先级最低; 其中用方括号“[]”括起来的功能, 代表调试/编程专用口; 其中用圆括号“()”括起来的功能, 代表这些复用功能在“用户配置字”中才能设定; 其中“nXX”的 n, 代表其为低电平有效。

注 2: 原则上同一个引脚同一时刻只可选择一种功能使用, 当一个引脚同时打开了多种复用功能时, 其中最高优先级的功能有效, 其他功能同一时刻将自动屏蔽。

## 2.4 引脚说明

IO 复用功能说明:

引脚汇总表

引脚编号				端口名称	属性	数字端口功能 (数字输出功能按照最上边优先级最低) 注 1&注 2	模拟端口功能 注 3
TSSOP20 /SSOP20	QFN20	SOP16	SOP8				
1	18	13	5	AIN07/INT10/ PWM0/ P1.4	I/O	P1.4: GPIO 组 1 的引脚 4 端口 PWM0: PWM0 独立输出口 INT10: 外部中断 1 输入通道 0 端口	AIN07: ADC 输入通道 7 端口
2	19	14	6	[DIO1]/AIN08/ SDA/TX0/P1.5	I/O	P1.5: GPIO 组 1 的引脚 5 端口 TX0: UART0 的 TXD 端口 SDA: IIC 的数据线 DIO1: 组 1 调试和编程的数据端口	AIN08: ADC 输入通道 8 端口
3	20	15	7	[CLK1]/AIN09/ SCL/RX0/P1.6	I/O	P1.6: GPIO 组 1 的引脚 6 端口 RX0: UART0 的 RXD 端口 SCL: IIC 的时钟线 CLK1: 组 1 调试和编程的时钟端口	AIN09: ADC 输入通道 9 端口
4	1	-	-	(nRST)/P1.7	I/O	P1.7: GPIO 组 1 的引脚 7 端口 nRST: 外部 RESET 输入端口(低电平有效)	
5	2	-	-	(OSCO)/ TX0/P0.0	I/O	P0.0: GPIO 组 0 的引脚 0 端口 TX0: UART0 的 TXD 端口	OSCO: 外部 RTC 晶振输出端口
6	3	-	-	(OSCI)/	I/O	P0.1: GPIO 组 0 的引脚 1 端口	OSCI: 外部 RTC



引脚编号				端口名称	属性	数字端口功能 (数字输出功能按照最上边优先级最低) 注1&注2	模拟端口功能 注3
TSSOP20 /SSOP20	QFN20	SOP16	SOP8				
				RX0/P0.1		RX0: UART0 的 RXD 端口	晶振输入端口
7	4	16	8	GND	P		GND: 芯片接地端口
8	5	-	-	[DIO2]/Vref/IN T11/T2/SDA/TX0/P2.0	I/O	P2.0: GPIO 组 2 的引脚 0 端口 TX0: UART0 的 TXD 端口 SDA: IIC 的数据线 T2: 定时器 2 的输入/输出端口 (含输出比较) INT11: 外部中断 1 输入通道 1 端口 DIO2: 组 2 调试和编程的数据端口	Vref: ADC 外部参考源输入端口
9	6	1	1	VDD	P		VDD: 芯片接电源端口
10	7	2	-	INT12/nSS/T2EX/P2.1	I/O	P2.1: GPIO 组 2 的引脚 1 端口 T2EX: 定时器 2 的重载触发/输入捕捉输入端口 nSS: SPI 的 nSS 端口(低电平有效) INT12: 外部中断 1 输入通道 2 端口	
11	8	3	-	AIN00/INT13/SCL/RX0/P2.2	I/O	P2.2: GPIO 组 2 的引脚 2 端口 RX0: UART0 的 RXD 端口 SCL: IIC 的时钟线 INT13: 外部中断 1 输入通道 3 端口	AIN00: ADC 输入通道 0 端口
12	9	4	-	AIN01/INT14/SDA/TX0/P2.3	I/O	P2.3: GPIO 组 2 的引脚 3 端口 TX0: UART0 的 TXD 端口 SDA: IIC 的数据线 INT14: 外部中断 1 输入通道 4 端口	AIN01: ADC 输入通道 1 端口
13	10	5	-	AIN02/INT04/PWM1CN/P2.4	I/O	P2.4: GPIO 组 2 的引脚 4 端口 PWM15: PWM1 通道 5 输出端口 INT04: 外部中断 0 输入通道 4 端口	AIN02: ADC 输入通道 2 端口
14	11	6	-	AIN03/INT03/PWM1BN/P2.5	I/O	P2.5: GPIO 组 2 的引脚 5 端口 PWM13: PWM1 通道 3 输出端口 INT03: 外部中断 0 输入通道 3 端口	AIN03: ADC 输入通道 3 端口
15	12	7	-	SCK/T1/PWM1FB/P2.6	I/O	P2.6: GPIO 组 2 的引脚 6 端口 PWM1FB: PWM1 的刹车控制输入端口 T1: 定时器 1 的输入/输出端口	



引脚编号				端口名称	属性	数字端口功能 (数字输出功能按照最上边优先级最低) 注1&注2	模拟端口功能 注3
TSSOP20 /SSOP20	QFN20	SOP16	SOP8				
						SCK: SPI 的 CLK 端口	
16	13	8	-	INT02/MOSI/P WM1C/ P2.7	I/O	P2.7: GPIO 端口 2 的引脚 7 PWM14: PWM1 通道 4 输出 MOSI: SPI 的 MOSI 端口 INT02: 外部中断 0 输入通道 2	
17	14	9	-	INT01/MISO/P WM1B/P1.0	I/O	P1.0: GPIO 组 1 的引脚 0 端口 PWM12: PWM1 通道 2 输出端口 MISO: SPI 的 MISO 端口 INT01: 外部中断 0 输入通道 1 端口	
18	15	10	4	[CLK0]/ [CLK2]/AIN04/ T0/SCL/RX0/P 1.1	I/O	P1.1: GPIO 组 1 的引脚 1 端口 RX0: UART0 的 RXD 端口 SCL: IIC 的时钟线 T0: 定时器 0 的输入/输出端口 CLK2: 组 2 调试和编程的时钟端口 CLK0: 组 0 调试和编程的时钟端口	AIN04: ADC 输入通道 4 端口
19	16	11	3	[DIO0]/AIN05/ PWM1AN/SDA /TX0/P1.2	I/O	P1.2: GPIO 组 1 的引脚 2 端口 TX0: UART0 的 TXD 端口 SDA: IIC 的数据线 PWM11: PWM1 通道 1 输出端口 DIO0: 组 0 调试和编程的数据端口	AIN05: ADC 输入通道 5 端口
20	17	12	2	AIN06/INT00/ PWM1A/ P1.3	I/O	P1.3: GPIO 组 1 的引脚 3 端口 PWM10: PWM1 通道 0 输出端口 INT00: 外部中断 0 输入通道 0 端口	AIN06: ADC 输入通道 6 端口

注 1: 数字输入功能 (含数字复用功能) 在 IO 配置成非高阻模式 (即输入模式/推挽输出/开漏输出) 下始终有效;

注 2: 数字输出功能 (含数字复用功能) 必须在 IO 配置成推挽输出或开漏输出模式下才能有效, 优先级按照表格中描述;

注 3: 模拟端口功能, 除了电源 (VDD/GND) 和模拟输出功能 (外部晶体), 其它模拟输入功能可以在任何情况下生效, 能与数字功能共存; 模拟输出功能 IO 必须配置成高阻模式;

注 4: 在高阻模式下, 任何数字输入和输出功能都无效

## 2.5 订购信息

产品名称	封装形式	备注
MC51F7084A0Y	TSSOP20	-
MC51F7084A0M	SOP20	引脚排列同 TSSOP20 封装



MC51F7084A0ZQ	QFN20	请注意，引脚排列位置有调整
MC51F7084A0K	SOP16	请注意，引脚排列位置有调整
MC51F7084A0H	SOP8	请注意，引脚排列位置有调整

注：产品名称中的倒数第 2 位 x 代表本产品的出货包装信息，最终的具体类型请与我司销售人员确认。

## 2.6 基本概念和术语

- ◇ R/W: 代表 Read/Write 的含义，即可读可写。当仅有 R 时，代表只读；当仅有 W 时，代表只写。
- ◇ R\*/W,R/W\*: R\*代表仅可读 0 或 1；W\*代表仅可写 0 或 1。
- ◇ 0/1/X/-: 其中 0=数字逻辑 0 (低电平)；1=数字逻辑 1 (高电平)；X=不确定 (高低电平皆可能)；-=无效或保留 (禁用)。
- ◇ B/b: B 代表 Byte 的含义，即字节，表示了 8 位数据长度；b 代表 bit 的含义，即位，表示 1 位数据长度。
- ◇ 双两线调试接口 (Two-wire): 是一种类似于 SWD 的调试接口形式，是晟矽微电子特有的调试/编程接口。为用户精简了调试/编程接口电路，方便了硬件工程师的 PCB 设计；通过两组调试接口交替仿真，让工程师可以做到“所有端口的全覆盖仿真”。
- ◇ ICP: 代表在电路编程，即 In Circuit Programming 的首字母缩写。用户可通过电路板上预留的两线调试和编程接口 (Two-wire)，在印刷电路板组装以后 (PCBA)，直接将程序下载到芯片中。
- ◇ IAP: 代表在应用编程，即 In Application Programming 的首字母缩写。只要芯片可以运行，用户就可以通过程序对芯片本身进行编程。
- ◇ FLASH 的分区保护功能: 用户可以部分或者全部保护 FLASH 区。可以防范用 FLASH 区中的代码在不可信的情况下被读出，也可以防范程序跑飞对 FLASH 的意外擦除或者内容改写，还可以为用户的二次开发提供有效手段，这是一种高可靠性的安全防护措施。



### 3 电气参数

下述无特殊说明的情况时， $T_a=25^{\circ}\text{C}$ ， $V_{DD}-GND=5V$ 。

#### 3.1 极限参数

参数	符号	值	单位
直流供电电压	VDD/GND	-0.3~5.5	V
IO 输入/输出电压	Vin/Vout	GND-0.3~VDD+0.3	V
工作环境温度	Ta	-40~85	$^{\circ}\text{C}$
储存环境温度	Tstg	-55~150	$^{\circ}\text{C}$
片上 FLASH/EEPROM 操作温度	TMTP	-40~85	$^{\circ}\text{C}$

注：流过 VDD、GND 电流总和的最大值在  $5.0V/25^{\circ}\text{C}$  下必须小于  $200\text{mA}$ 。本规格仅采用本芯片在上述条件内测试获得，对超出上述条件的“特殊应用”不做保证。若芯片长期处于极限参数的最大或者最小条件下工作，可能会直接影响器件的可靠性并加速器件的老化。

#### 3.2 推荐工作条件

参数	符号	最小值	最大值	单位	备注
工作电压 1	VDD-GND	3.5	5.5	V	CORE 可运行于 Fsys 最高频率 16MHz，建议开启 LVR=3.5V 或者 4.1V。
工作电压 2	VDD-GND	2.5	5.5	V	CORE 可运行于 Fsys 最高频率 8MHz 建议按需开启 LVR。
工作环境温度	Ta	-40	85	$^{\circ}\text{C}$	-

#### 3.3 直流特性参数

特性	符号	引脚	条件	最小	典型	最大	单位
输入低电压	Vil	所有输入脚	IO 输入模式下 (SMT 输入方式)	GND-0.3	-	0.2VDD	V
输入高电压	Vih	所有输入脚	IO 输入模式下 (SMT 输入方式)	0.8VDD	-	VDD+0.3	V
输入漏电流	Iilc	所有输入脚	IO 输入模式下，Vin=VDD 或者 GND	-1	-	1	$\mu\text{A}$
输出高电流 (推挽输出)	Ioh	所有 IO	IO 输出模式下，Voh=0.9VDD	-	10	-	mA
输出低电流 (推挽输出)	Iol	所有 IO	IO 输出模式下，Vol=0.1VDD	-	18	-	mA
内部上拉电阻	Rpu1	P0	VDD=5V, Vin=0, 使能上拉电阻	-	15	-	K $\Omega$
	Rpu2	P1,P2	VDD=5V, Vin=0, 使能通用上拉电阻	-	15	-	K $\Omega$
	Rpu3		VDD=5V, Vin=0, 使能专用上拉电阻	-	1.9	-	K $\Omega$



特性	符号	引脚	条件	最小	典型	最大	单位
			(I2C 模块中控制)				
内部下拉电阻	Rpd	P0,P1,P2	Vin=VDD=5V, 使能下拉电阻	-	15	-	KΩ
输入 RESET 低电平	Vlrst	nRST	作为 RESET 使用时	GND	-	0.2VDD	V
输入 RESET 高电平	Vhrst	nRST	作为 RESET 使用时	0.8VDD	-	VDD	V
RESET 引脚的内部上拉电阻	Rrst	nRST	作为 RESET 使用时	-50%	50K	+100%	KΩ
工作电流 (高级能耗控制全关; 所有引脚无负载; CORE 执行 NOP 指令)	Iddc	VDD	VDD=5V, F <sub>CORE</sub> =16MHz@HIRC	-	8	-	mA
			VDD=5V, F <sub>CORE</sub> =8MHz@HIRC	-	4	-	mA
			VDD=5V, F <sub>CORE</sub> =2MHz@HIRC	-	2.5	-	mA
			VDD=5V, F <sub>CORE</sub> =1MHz@HIRC	-	2.1	-	mA
			VDD=5V, F <sub>CORE</sub> =32KHz@LIRC, HIRC 关, XTAL 关	-	0.5	-	mA
工作电流 (高级能耗控制全开; 所有引脚无负载; CORE 执行 NOP 指令)	Iddc	VDD	VDD=5V, F <sub>CORE</sub> =16MHz@HIRC	-	9	-	mA
			VDD=5V, F <sub>CORE</sub> =8MHz@HIRC	-	5	-	mA
			VDD=5V, F <sub>CORE</sub> =2MHz@HIRC	-	2.6	-	mA
			VDD=5V, F <sub>CORE</sub> =1MHz@HIRC	-	2.2	-	mA
			VDD=5V, F <sub>CORE</sub> =32KHz@LIRC, HIRC 关, XTAL 关	-	0.5	-	mA
待机电流 (停止模式)	Istop	VDD	VDD=5V, F <sub>CORE</sub> =16MHz@HIRC; CORE 停止, LIRC 关, XTAL 关; WDT 关, LVR 开; 所有引脚无负载, INTn 唤醒开; 其他功能模块关	-	5	-	mA
			VDD=5V, F <sub>CORE</sub> =32KHz@LIRC; CORE 停止, HIRC 关, XTAL 关; WDT 关, LVR 开; 所有引脚无负载, INTn 唤醒开; 其他功能模块关	-	50	-	μA
待机电流 (休眠模式)	Isleep	VDD	VDD=5V, CORE 休眠, HIRC/LIRC 关, XTAL 关; WDT 关, LVR 关; 所有引脚无负载, INTn 唤醒开; 其他功能模块关	-	2	-	μA
待机后定时唤醒的平均电流 (休眠模式)	I <sub>sa</sub>	VDD	VDD=5V, CORE 休眠, HIRC 关, XTAL 关, LIRC 开, T3 的 1S 定时唤醒开; WDT 关, LVR 开; 所有引脚无负载; 其他功能模块关	-	12	-	μA
WDT 电流	I <sub>wdt</sub>	VDD	VDD=5V, CORE 休眠, HIRC/LIRC 关, XTAL 关; WDT 工作; 其他功能模块关	-	-	3	μA
LVR 电流	I <sub>lvr</sub>	VDD	VDD=5V, CORE 休眠, HIRC/LIRC 关,	-	-	3	μA



特性	符号	引脚	条件	最小	典型	最大	单位
			XTAL 关; WDT 关, LVR 开; 其他功能模块关				
RAM 保持电压	V <sub>RAM</sub>	VDD	-	-	0.7	-	V

### 3.4 LVR 特性参数

特性	符号	条件	最小	典型	最大	单位
LVR 设定电压 1	Vlvr1	LVR 使能, VDD=2.5~5.5V	-	2.5	-	V
LVR 设定电压 2	Vlvr2		-	3.5	-	V
LVR 设定电压 3	Vlvr3		-	4.1	-	V
LVR 回滞电压	Vlvrs	-	-	100	-	mV
LVR 模拟滤波时间	Tlvraf		-	100	2000	μs
LVD 模拟滤波时间	Tlvdaf		-	100	2000	μs
LVD 无模拟滤波时间	Tlvdoaf		-	5	-	μs

### 3.5 ADC 特性参数

下表为分辨率 12bit 参数

特性	符号	条件	最小	典型	最大	单位
ADC 有效工作电压	V <sub>adc</sub>	-	2.5	-	5.5	V
分辨率	Nr	12-bit 模式	-	12	-	bit
转换率	F <sub>s</sub>	12-bit 模式, 参考电压为 VDD	-	-	1	msps
		12-bit 模式, 参考电压为 VREF2.0			0.3	
积分线性误差	INL	VDD=5V, F <sub>adc</sub> =1-16MHz, 采样时间=8Clk, 12bit 精度	-	-	±2	LSB
微分线性误差	DNL	FADC=16MHz	-	-	±1	LSB
上限偏置误差	E <sub>ot</sub>	FADC=16MHz	-	±1	±4	LSB
下限偏置误差	E <sub>ob</sub>	FADC=16MHz	-	±1	±4	LSB
零点偏移误差	E <sub>z</sub>	FADC=16MHz	-	±0.5	±4	LSB
总绝对误差	E <sub>ad</sub>	FADC=16MHz	-	±4	-	LSB
转换时钟	F <sub>adc</sub>	VDD=5V	-	-	16	MHz
		VDD=3.3V	-	-	16	MHz
转换时间	T <sub>con</sub>	VDD=5V, F <sub>adc</sub> =16MHz, 采样时间=4Clk, 12-bit 模式	17	-	-	T <sub>adc</sub>
ADC 输入电压	V <sub>ain</sub>	-	GND	-	VREF/ Gain	V
ADC 输入阻抗	R <sub>ain</sub>	-	2	-	-	MΩ
ADC 输入电流	I <sub>ain</sub>	-	-	-	10	μA



特性	符号	条件	最小	典型	最大	单位
ADC 动态电流	I <sub>add</sub>	AD 转换中	-	1	1.5	mA
ADC 静态电流	I <sub>ads</sub>	ADEN 位=0	-	0.1	1	μA
模拟信号源推荐阻抗	Z <sub>ain</sub>	-	-	-	10	KΩ
ADC 启动时间	T <sub>on</sub>	参考电压为 VDD			1.5	μs
内部参考电压	V <sub>ref</sub>	@VDD ≥ 2.5V, 全温度	-2%	2.0	+2%	V
内部参考电压启动时间	T <sub>on</sub>		-	-	20	μs

### 3.6 片上 FLASH 特性

特性	符号	条件	最小	典型	最大	单位
Page sizes	S <sub>page</sub>			64		bytes
FLASH 读取电流	I <sub>read</sub>	@10MHz, VDD=5V			1.2	mA
		@32KHz, VDD=5V			20	μA
编程电流	I <sub>prog</sub>	VDD=5V			2	mA
擦除电流	I <sub>erase</sub>	VDD=5V			2	mA
编程时间	T <sub>prog</sub>			4.5	5	ms
擦除时间	T <sub>erase</sub>			4.5	6	ms
FLASH 读写循环次数	N <sub>MNDUR</sub>	-	10000	-	-	cycle
E2PROM 读写循环次数	N <sub>ENDUR</sub>	-	100000	-	-	cycle
FLASH 数据保存时间	T <sub>FRET</sub>	-	10	-	-	year

注：片上 FLASH 支持 ICP、IAP、ISP 进行操作。

### 3.7 交流电气参数

特性	符号	条件	最小	典型	最大	单位
HIRC 振荡频率	F <sub>HIRC</sub>	VDD=5V, T=25°C	-1%	32	+1%	MHz
		T=-40°C~85°C, VDD=2.5~5.5V	-5%		+5%	MHz
HIRC 起振时间	T <sub>HIRC</sub>	VDD=2.4~5.5V	-	-	20	μs
LIRC 振荡频率	F <sub>LIRC</sub>	VDD=5V	-10%	32	+10%	KHz
LIRC 起振时间	T <sub>LIRC</sub>	VDD=2.5~5.5V	-	-	150	μs
32768Hz 振荡器起振时间	T <sub>LOSC</sub>	VDD=5V	-	1	5	s
32768Hz 振荡器内部电容	C <sub>LOSC</sub>	VDD=2.5~5.5V	-	10 <sup>注</sup>	-	pF
外部复位脉冲宽度	T <sub>RST</sub>	-	10	-	-	μs

注：芯片有内置电容，晶体外部负载电容请参考所购买的晶体振荡器规格进行匹配并考虑内置电容容值。



## 4 中央处理器

### 4.1 CPU 概述

CPU 为 1T 标准的高速 8051 内核，其指令完全兼容传统 8051 单片机。

CPU 包含复位初值为 0000H 的程序计数器 PC，包含 1 个专用累加器 A、辅助寄存器 B、堆栈指针 SP 和数据指针 DPTR，以及程序状态寄存器 PSW。

堆栈指针 SP 是 1 个 8 位专用寄存器，保存堆栈栈顶的地址信息。在执行 PUSH、子程序调用、中断响应等指令时，SP 先加 1 再将数据压栈；执行 POP、RET、RETI 等指令时，数据退出堆栈后 SP 再减 1。堆栈栈顶可以是片上内部 RAM (00H~FFH) 的任意地址，系统复位后，SP 初始化为 07H，堆栈实际从 08H 地址开始，用户可自行调整堆栈栈顶地址。

程序状态寄存器 PSW 用来保存指令执行中的各种状态。

数据指针 DPTR 是 16 位专用寄存器，其高字节寄存器为 DPH 表示，低字节寄存器为 DPL 表示。在一组 DPTR 中，它们既可以作为一个 16 位寄存器来处理，也可作为两个独立 8 寄存器来处理。

#### 4.1.1 程序状态寄存器 (PSW, 0xD0/SFR0)

SBIT 寄存器	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PSW	CY	AC	F0	RS1	RS0	OV	F1	P
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7] CY - 进/借位标志位

0: 算术或逻辑运算中，最高位 (Bit7) 没有进位或借位发生；

1: 算术或逻辑运算中，最高位 (Bit7) 有进位或借位发生。

BIT[6] AC - 进/借位辅助标志位 (可在 BCD 码加减法运算时方便调整)

0: 算数逻辑运算中，半字节 (Bit3) 没有辅助进位或借位发生；

1: 算数逻辑运算中，半字节 (Bit3) 有辅助进位或借位发生。

BIT[5] F0 - 用户自定义标志位

BIT[4:3] RS[1:0] - 工作寄存器组选择位

RS[1:0]	当前使用的工作寄存器组 0~3
00	工作寄存器组 0 (00H~07H)
01	工作寄存器组 1 (08H~0FH)
10	工作寄存器组 2 (10H~17H)
11	工作寄存器组 3 (18H~1FH)

BIT[2] OV - 溢出标志位

当进行有符号 (signed) 数加减法运算时：

0: 表示一个数字未超出了累加器 A 以补码形式表示一个有符号数的范围；

1: 表示一个数字超出了累加器 A 以补码形式表示一个有符号数的范围，即超出了 -128~+127 的范围。

当执行乘法指令 (MUL AB) 运算时：

0: 表示乘积未超过 255，结果只在 A 中；

1: 表示乘积超过 255，结果在 A 和 B 中。

当执行除法指令 (DIV) 运算时：



- 0: 表示除数为非 0;  
1: 表示除数为 0。
- BIT[1] F1 – 用户自定义标志位
- BIT[0] P – 奇偶校验位
- 0: 表示累加器 A 中的二进制结果中有 1 的个数为偶数个;  
1: 表示累加器 A 中的二进制结果中有 1 的个数为奇数个。

## 4.2 程序存储器 (FLASH ROM)

程序存储器 ROM 总的大小为 16K 字节, 地址范围为 0000H~3FFFH, 该 FLASH ROM 可反复擦写 1 万次, 数据保存年限至少 10 年。

FLASH ROM 通常需要先擦除后写入。本芯片的 FLASH ROM 支持在电路编程 (ICP) 和在应用编程 (IAP)。

注: ICP 和 IAP 请参考相关章节的内容。

## 4.3 附加数据区 (EEPROM)

本芯片内置 256 bytes 附加数据区, 可作为 EEPROM 使用, 支持页擦除/字节烧写。

本芯片的 EEPROM 支持在电路编程 (ICP) 和在应用编程 (IAP), 也可通过 MOVX 指令进行寻址。

注: ICP 和 IAP 相关内容请参考相关章节的内容。

## 4.4 随机存储器 (RAM)

随机数据存储器 RAM 包括内部 256 字节和外部 768 字节共 1024 字节的 SRAM。

内部 RAM 的地址为 00H~FFH, 其中高 128 字节 (80H~FFH) 与特殊功能寄存器 SFR 重叠, 且只能间接寻址, 低 128 字节 (00H~7FH) 可直接寻址也可以间接寻址。其中最低的 32 字节 (00H~17H) 作为通用寄存器分成四组 8 个寄存器, 程序视这些寄存器为 R0 到 R7, 具体通过程序状态字 PSW 的 RS[1:0]位来选择哪一个寄存器组会被使用; 接下来的 16 个字节 (20H~2FH) 还支持位寻址; 剩下的字节 (30H~7FH) 作为通用 RAM, 可用来设置栈指针, 用作“栈”使用。

外部 RAM 的地址为 0000H~02FFFH, 这部分空间必须通过 MOVX 指令来寻址。

特殊功能寄存器 SFR 地址为 80H~FFH, 只能直接寻址。CPU 自动通过指令区分是访问 SFR 还是内部 RAM 的高 128 字节; SFR 通过 DPS 的 SFRSL0 位切换, 拓展了 SFR 的范围。

注 1: 本芯片不支持 P0 和 P2 口的外部 RAM 扩展功能。

注 2: PSW 的 RS[1:0]相关内容请参考相关章节的内容。

注 3: DPS 的 SFRSL0 相关内容请参考相关章节的内容。

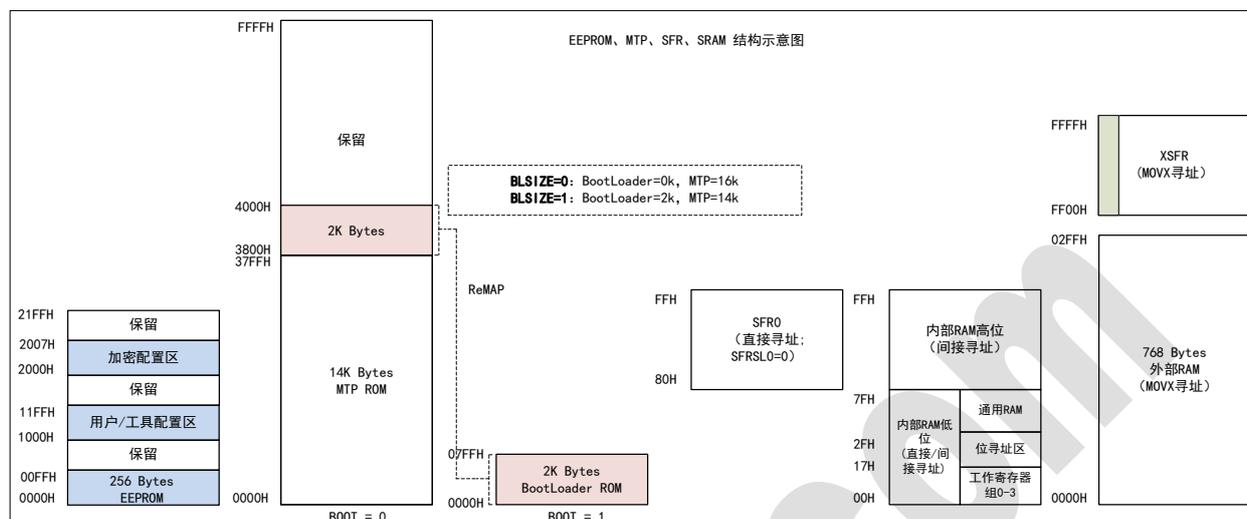
## 4.5 存储器结构

本芯片的程序存储器 FLASH、随机存储器 RAM 以及非易失性数据存储器 EEPROM 分别是独立编址的。由于没有提供外部存储器扩展的总线接口, 所以并不能像传统的 8051 通过 P2 端口进行外扩。

本芯片片上存储器的结构, 如下图所示:



## 存储器 MAP 示意图



该芯片的 FLASH 空间由 8 位宽的存储单元组成，即可以保存代码又可以保存数据。当前 FLASH 每 64Bytes 为 1 扇区，共 256 个扇区。在 FLASH “分区保护” 的时候，以扇区为操作单位，具体在“用户配置字”中设定，请参考相关章节的内容。

## ROM 空间操作汇总表

操作形式	ICP	IAP	ISP
FLASH 代码保护	支持	不支持	支持
FLASH 页擦除	支持	不支持	支持
FLASH 块擦除	支持（不保护的情况下）	不支持	支持（不保护的情况下）
FLASH 整体擦除	支持	不支持	不支持
FLASH 读/写	支持（不保护的情况下）	不支持	支持（不保护的情况下）
EEPROM 擦除	支持	支持	支持
EEPROM 读/写	支持	支持	支持

## 4.6 寻址方式

CPU 指令的寻址方式有以下 7 种：立即寻址、直接寻址、间接寻址、寄存器寻址、相对寻址、变址寻址、位寻址。

## 立即寻址

立即寻址也称为立即数，它是在指令操作数中直接给出参加运算的操作数。

## 直接寻址

在直接寻址方式中，指令操作数域给出的是参加运算操作数的地址。直接寻址方式只能用来表示特殊功能寄存器、内部数据寄存器和位地址空间。其中特殊功能寄存器和位地址空间只能用直接寻址方式访问。

## 间接寻址

间接寻址采用 R0 或 R1 前加“@”符号来表示。

## 寄存器寻址



寄存器寻址寻址时对选定的工作寄存器 R7~R0、累加器 A、通用寄存器 B、地址寄存器和进位标志 CY 中的数进行操作。其中寄存器 R7~R0 由指令码的低 3 位表示，A、B、DPTR 及进位标志 CY 隐含在指令码中，因此寄存器寻址也包含一种隐含寻址方式。寄存器工作区的选择由程序状态寄存器 PSW 中的 RS[1:0]来决定。指令操作数指定的寄存器均指当前工作区的寄存器。

#### 相对寻址

相对寻址是将程序计数器 PC 中的当前值与指令第二字节给出的数相加，其结果作为转移指令的转移地址。转移地址也成为转移目的地址，PC 中的当前值成为基地址，指令第二字节给出的数成为偏移量。由于目的地址是相对于 PC 中的基地址而言，所以这种寻址方式称为相对寻址。偏移量为带符号的数，所能表示的范围为 (+127) ~ (-128)。这种寻址方式主要用于转移指令。

#### 变址寻址

在变址寻址方式中，指令操作数指定一个存放变址基址的变址寄存器。变址寻址时，偏移量与变址基值相加，其结果作为操作数的地址。变址寄存器有程序计数器 PC 和地址寄存器 DPTR。

语句“MOVC A, @A+DPTR”表示累加器 A 为偏移量寄存器，其内容与地址寄存器 DPTR 中的内容相加，其结果作为操作数的地址，取出该单元中的数送入累加器 A 中。

#### 位寻址

位寻址是指对一些可进行位操作的内部数据存储单元 RAM 和特殊功能寄存器进行位操作时的寻址方式。在进行位操作时，借助于进位标志 CY 作为位操作累加器，指令操作数直接给出该位的地址，然后根据操作码的性质对该位进行位操作。位地址与字节直接寻址中的字节地址编码方式完全一样，主要由操作指令的性质加以区分，使用时应特别注意。

## 4.7 功能寄存器映射

特殊功能寄存器 SFR 的地址位于 80H~FFH，有些可以位寻址，有些不能位寻址。能够进行位寻址操作的寄存器的地址末位数都是“0”或“8”，这些寄存器在需要改变单个位的数值时非常方便。所有的特殊功能寄存器只能通过直接寻址方式进行寻址。

### 特殊功能寄存器列表(SFR0)

SFR 寄存器	可位寻址	不可位寻址							SFR 页
		1/9	2/A	3/B	4/C	5/D	6/E	7/F	
F8H	PWRCR	PWRCR1	ADCMPL	ADCMPLH	-	ADRIOS1	ADRL	ADRH	SFR0
F0H	B	ADCCR0	ADCCR1	ADCCR2	ADCCR3	OSADJCR	LVDCR	LVDCR1	SFR0
E8H	RSTFLAG	IAPCR	IAPADL	IAPADH	IAPDBL	IAPDBH	IAPMD	IAPKEY	SFR0
E0H	ACC	IICCR0	IICCR1	IICSR	IICAR	IICDR	IICAAR	IICAMR	SFR0
D8H	PWM1IF	PWM1RLEN	PWM1CR1	PWM1CR2	PWM1FBCR	PWM1MANCR 0	PWM1MANCR 1	IAPCR1	SFR0
D0H	PSW	PWM1OE	PWM1ADL	PWM1ADH	PWM1BDL	PWM1BDH	PWM1CDL	PWM1CDH	SFR0
C8H	T2CON	T2MOD	RCAPL2	RCAPH2	TL2	TH2	T2CON1	WDTCR	SFR0
C0H	CLKCR0	P0MOD0	-	P0PU	P0PD	OSCSS	CLKCR1	-	SFR0
B8H	IPL0	-	-	IPL1	-	IPL2	-	-	SFR0
B0H	-	-	-	-	-	EXTIS2	EXTIS1	EXTIS	SFR0
A8H	IEN0	-	IEN1	IEN2	-	SPCON	SPSTA	SPDAT	SFR0
A0H	P2	P2MOD0	P2MOD1	P2PU	P2PD	EXTEN1	EXTEN	-	SFR0
98H	SCON0	SBUF0	SADDR	SADEN	SCON2	BRGCR	BRGL	BRGH	SFR0



90H	P1	P1MOD0	P1MOD1	P1PU	P1PD	T3CR	TL3	TH3	SFR0
88H	TCON	TMOD	TL0	TL1	TH0	TH1	TCON1	-	SFR0
80H	P0	SP	DPL	DPH	T3CNTL	T3CNTH	SLPCR	PCON	SFR0
起始地址	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	SFR 页
	可位寻址	不可位寻址							

### 扩展特殊功能寄存器列表(XSFR)

XSFR 寄存器	偏移地址								SFR 页
	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	
起始地址	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	
FFF8H~FF70H	-	-	-	-	-	-	-	-	XSFR
FF68H	PWM0CR0	PWM0P	PWM0AD	-	-	-	-	-	XSFR
FF60H	CRCCR	-	CRCH	CRCL	CRCSTH	CRCSTL	CRCEDH	CRCEDL	XSFR
FF5FH~FF50H	-	-	-	-	-	-	-	-	XSFR
FF48H	PWM1CR0	PWM1IE	PWM1PH	PWM1PL	-	PWM1DT0	-	-	XSFR
FF40H~FF00H	-	-	-	-	-	-	-	-	XSFR
起始地址	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	SFR 页
	偏移地址								

## 4.8 用户配置字

芯片内部包含一块额外的 FLASH 空间用于保存芯片的配置信息。芯片为保证系统正常工作，会将关键模块的配置预先存储于此区域，在上电或其他复位发生后会将配置信息载入寄存器中，通过寄存器确定关键模块的初始工作状态。这部分配置信息，即为用户配置字。

注：“用户配置字”在芯片正常操作时不允许修改，否则芯片会出现不可预知的情况。

### 用户配置字详细说明

符号	功能说明
BOOTS	BOOT-启动 ROM 区域选择 0: 复位后 MCU 从 FLASH rom 区 0000H 启动; 1: 复位后 MCU 从 bootloader rom 区 0000H 启动; 注: 除软复位 (SWRST) 外, 其他复位此位内容加载至 IAPCR1->BOOT(BIT7)。
BLSIZE	Bootloader Rom 区空间大小选择 0: BootLoader=0k, Main FLASH=16k 1: BootLoader=2k, Main FLASH=14k 注: 若设置为 BootLoader=2k (BLSIZE=1), 烧写模式下 (ICP) 仍可以通过访问第 15/16k 空间 (3800H~3FFFH) 访问。
WDTM[1:0]	WDT 工作模式选择位:



符号	功能说明
	00: 始终关闭 WDT: “看门狗” 功能无效; 01: 自动关闭 WDT: 在停止模式/休眠模式下, 自动关闭 “看门狗” ; 10: 始终开启 WDT: 在停止模式/休眠模式下, “看门狗” 继续工作, 故需要手动喂狗。 注: 当本位选择了 “始终开启 ‘看门狗’ ” 时, 请务必按时软件喂狗, 否则将会产生 “WDT 复位” 。
RSTEN	允许外部复位操作位: 0: 外部复位功能无效: 对应的 nRST 口可用于 GPIO 或者其他复用功能, 该 PIN 不可用于外部复位; 1: 外部复位功能有效: nRST 口自动内部带上拉, 低电平时复位有效, 该 PIN 仅可用于外部复位, 其他功能无效。 注: 当本位选择了 “外部复位功能无效” 时, 本芯片将无外部复位功能可用。
IOSPEEDS	IO输出速度选择位 0: 慢速翻转 (慢开快关) 1: 快速翻转 (快开快关) 注: 慢速翻转支持最快 2MHz, 快速翻转支持最快 16MHz (IO 端口空载模式) ;
LVRSLP	休眠模式下 LVR 功能的选择位: 0: 休眠模式下 LVR 无效: LVR 功能将在 Core 进入休眠模式下, 自动关闭; 1: 休眠模式下 LVR 有效: LVR 功能始终有效; 注: “停止模式” 下, LVR 不可控, 将始终有效。
VLVRS	LVR 电压等级选择位: 101: 2.5V; 100: 3.5V; 001: 4.1V; 其他: 保留 注: 本位与 LVRSLP 位联动, 当 LVRSLP 为 “停止模式/休眠模式下 LVR 无效” 时, 则 LVR 功能将在停止模式/休眠模式下, 自动关闭。 注意: 若此位配置 2.5v 档, 用户需确保高频分频控制 OSCSS->FHS[2:0]禁止配置 000b (1 分频), 否则 CPU 16MHz 下, 低于 3.5v 有进入死区风险。 若用户用到 16MHz 主频 (OSCSS->FHS[2:0]=000b), 请务必配置 LVR 3.5v 档。
LVRAFEN	LVR 模拟滤波使能位: 0: LVR 输出无模拟滤波; 1: LVR 输出有模拟滤波;
LVRDFSEL	LVR 数字滤波选择位: 00: LVR 输出无数字滤波; 01: LVR 输出数字滤波 2 个 LIRC 时钟周期; 10: LVR 输出数字滤波 4 个 LIRC 时钟周期; 11: LVR 输出数字滤波 8 个 LIRC 时钟周期; 注: LVR 输出模拟滤波和数字滤波功能相互独立, 若同时开启则滤波时间累加。
OCDEN	OCD 调试端口运行状态使能:



符号	功能说明
	<p>0: 芯片运行状态, 关闭 OCD 调试端口;</p> <p>1: 芯片运行状态, 使能 OCD 调试端口, OCD 组由 OCDSEL 决定。</p>
OCDSEL[1:0]	<p>运行状态 OCD 组选择:</p> <p>00: 使能 OCD0 调试口 (DIO0/CLK0), 同时关闭 OCD1 和 OCD2 调试口 (默认);</p> <p>01: 使能 OCD1 调试口 (DIO1/CLK1), 同时关闭 OCD0 和 OCD2 调试口;</p> <p>10: 使能 OCD2 调试口 (DIO2/CLK2), 同时关闭 OCD0 和 OCD1 调试口;</p> <p>11: 使能 OCD0 调试口 (DIO0/CLK0), 同时关闭 OCD1 和 OCD2 调试口。</p> <p><i>注: 此位只在 OCDEN=1 时有效。</i></p> <p><i>注: OCDEN 和 OCDSEL 控制位, 不影响上电或复位的 OCD 访问控制; 仅控制芯片运行状态是否打开内部 OCD 通路, 除此之外, 还要配置对应的 OCD 组端口为输入状态, 内部 OCD 控制器才能收到控制时序。</i></p>
SC0P	<p>用户程序区 0000~0FFFH 操作权限控制:</p> <p>00: 第 0 分区加密块不加密: 对应的 0 分区加密块无限制, FLASH 内容可正常读出;</p> <p>01: 第 0 分区加密块一级加密: 对应的 0 分区一级加密限制, FLASH 内容禁止外部调试器/编程器读出; 允许 MOVN 指令在“脱机”情况下的读出、“调试”情况下 MOVN 指令读出禁止;</p> <p>1X: 第 0 分区加密块二级加密: 对应的 0 分区二级加密限制, FLASH 内容禁止外部调试器/编程器读出, 同时禁止内部 MOVN 指令任何情况下的读出;</p> <p><i>注: 该芯片中, 每 4K 字节大小设定为一个分区加密块, 根据 FLASH ROM 空间地址从低字节到高依次递增排列, 即 SC0P 对应 0x0000~0x0FFF 区域, SC1P 对应 0x1000~0x1FFF 区域, SC2P 对应 0x2000~0x2FFF 区域, SC3P 对应 0x3000~0x3FFF 区域。</i></p> <p><b><i>注: 选择一级加密和二级加密, 仿真界面下 code 回读会被禁止, MOVN 指令也会被禁止, 故不支持仿真调试。</i></b></p>
SC1P	<p>用户程序区 1000~1FFFH 操作权限控制:</p> <p>00: 第 1 分区加密块不加密:</p> <p>01: 第 1 分区加密块一级加密:</p> <p>1X: 第 1 分区加密块二级加密:</p>
SC2P	<p>用户程序区 2000~2FFFH 操作权限控制:</p> <p>00: 第 2 分区加密块不加密:</p> <p>01: 第 2 分区加密块一级加密:</p> <p>1X: 第 2 分区加密块二级加密:</p>
SC3P	<p>用户程序区 3000~3FFFH 操作权限控制:</p> <p>00: 第 3 分区加密块不加密:</p> <p>01: 第 3 分区加密块一级加密:</p> <p>1X: 第 3 分区加密块二级加密:</p>
SCBL0	<p>Bootloader 区 0000H~07FFFH 操作权限控制:</p> <p>00: Bootloader 区块不加密: Bootloader 区内容可正常读出;;</p> <p>01: Bootloader 区块一级加密: Bootloader 区一级加密限制, Bootloader 区内容禁止外部调试器/编程器读出; 允许 MOVN 指令在“脱机”情况下的读出、“调试”情况下 MOVN 指令读出禁止;</p>



符号	功能说明
	<p>1X: Bootloader 区块二级加密: Bootloader 区二级加密限制, Bootloader 区内容禁止外部调试器/编程器读出, 同时禁止内部 MOVC 指令任何情况下的读出;</p> <p>注: 选择一级加密和二级加密, 仿真界面下 code 回读会被禁止, MOVC 指令也会被禁止, 故不支持仿真调试。</p>



## 5 时钟与工作模式

### 5.1 系统时钟概述

芯片支持 3 种时钟源：

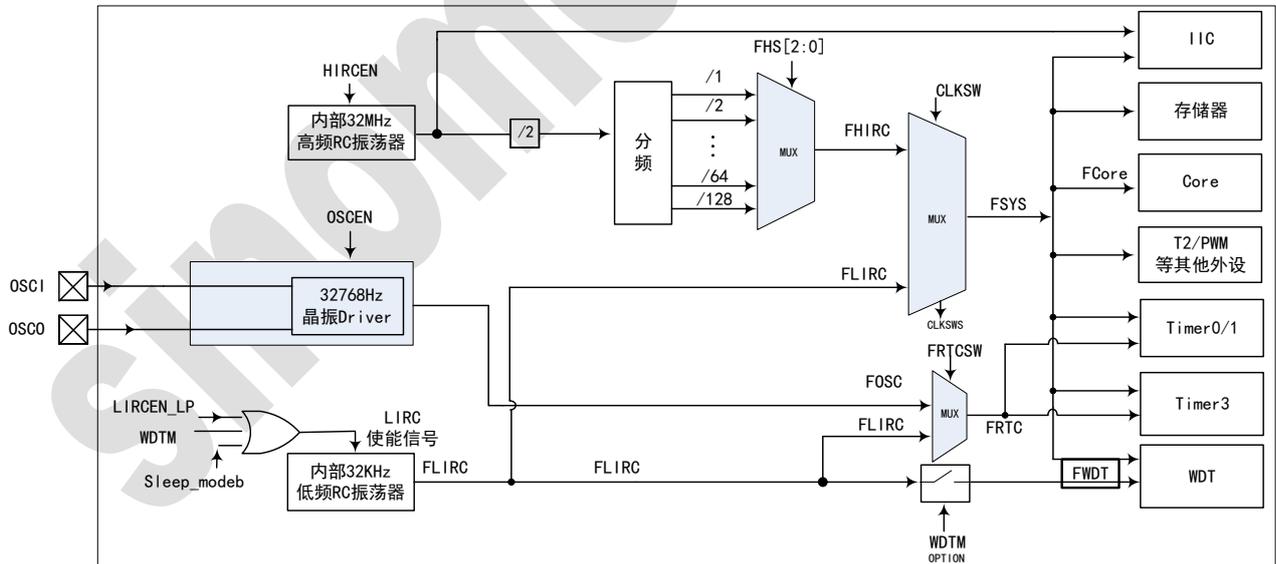
- ◇ 内部高频 RC 时钟 FHIRC 32MHz，该时钟进行 2 分频后（即 16MHz）可用作系统高频时钟源；
- ◇ 内部低频 RC 时钟 FLIRC 32KHz，可用作系统低频时钟源、或用作定时器的低频时钟 FRTC；
- ◇ 外部低频晶体时钟 FOSC 32.768KHz，用作定时器的低频时钟 FRTC；

除定时器 T0/T1/T3/IIC 外，包括 CPU 内核在内的系统其他模块均只能在系统时钟 FSYS 下工作。而 T0/T1/T3 还可在低频时钟 FRTC 下工作，且可通过寄存器位 FRTC SW 单独选择 FLIRC 或 FOSC 作为 FRTC；IIC 则只能在内部高频 RC 时钟 FHIRC 下工作。

3 种时钟源均可通过寄存器位开启或关停（LIRC 仅能设定低功耗模式下的开/关，且还受工作模式、WDT 状态和 LVR/LVD 数字滤波设置的影响），当外部晶体开启时，端口自动转为高阻态模式，对应的端口模式选择位无效。

3 种时钟源均有振荡稳定标志位用以反馈时钟源的振荡状态，只有振荡稳定后时钟源才能可靠使用。在切换系统时钟 FSYS 的时钟源时，需确认目标时钟源已开启并已稳定振荡，才可通过寄存器位进行切换。T3 切换低频时钟 FRTC 的时钟源时同样如此。

系统时钟示意图



### 5.2 内部高频 RC 振荡器

芯片内置高精度 32MHz RC 振荡器 HIRC，用户可通过寄存器 OSCSS 的 FHS[2:0]位配置成 128/64/32/16 /8/4/2/1 分频用于系统高频时钟 FHCLK。

内置的高频 RC 振荡器，可在芯片内部进行调校，出厂时即已调校至 32MHz@5V/25°C，用户使用时无需重新调校。



### 5.3 内部低频 RC 振荡器

芯片内置一个低频 RC 振荡器 LIRC，该振荡器可用于系统低频时钟 FSYS。该振荡器频率典型值为 32KHz。

FLIRC 作为 WDT 固定工作时钟。

FLIRC 还可作为定时器 T0/T1/T3 低频时钟 FRTC 的时钟源。

FLIRC 还可作为 LVR/LVD 数字滤波的时钟源。

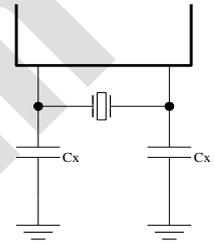
### 5.4 外接晶体振荡器

使用外部晶体作为内部模块的时钟源时，连接方式见左图。

寄存器控制位 OSCEN，选择低频晶体振荡器有效及 PIN 脚 OSC 功能有效。当外部晶体开启时，端口自动转为高阻态模式，对应的端口模式选择位无效。

低频仅支持 32768Hz 晶体，且通常 Cx 是必须的。外部低频晶体振荡器时钟还可作为定时器 T0/T1/T3 低频时钟 FRTC 的时钟源。

在实际使用中，用户应使晶体离 OSCI/OSCO 引脚的距离尽可能短，这样有助于振荡器的起振和振荡的稳定性。



下表列出典型频率晶振选用电容 Cx 的推荐值和相应最低起振电压参考值。

晶体频率 (Hz)	电容 Cx (pF)	最低起振电压 (V)
32768	10	2.5

注：因为晶体的品牌很多，电容值仅为推荐值，起振电压仅供参考，具体参数请根据实际使用的晶振性能而定。

### 5.5 时钟切换

当通过 CLKSW 切换系统时钟 FSYS 的时钟源时，系统会根据当前系统工作状态以及待切换时钟的工作状态，决定是否执行切换动作，并将通过时钟源标志位 CLKWS 反馈切换操作后的实际时钟源，用户可根据这些标志位判断切换是否成功。

注：时钟切换控制位 CLKSW，若切换条件不满足，则相应的控制位将无法被改写。

#### 时钟切换及关闭控制

执行动作	必须满足条件才能生效
高低频切换 CLKSW	1、待切换的目标时钟必须使能 2、待切换的目标时钟 RDY 位必须置位（已稳定） 3、待切换的时钟通路切换开关（CLKSW）必须与条件 1、2 匹配
关闭 HIRC (HIRCEN=0)	1、CLKSW=1 (选择 LIRC) 2、关闭 HIRC 立即生效 若 HIRC 被使用 (CLKSW=0), 则 HIRCEN 写 0 无效。
SLEEP 下关闭 LIRC (LIRCEN_LP=0)	1、SLEEP 模式 2、WDT SLEEP 下关闭 若 wdt sleep 下开启, LIRCEN_LP 关闭无效



## 5.6 振荡器预热和复位预热

### 振荡器时钟预热

振荡器类型	晶体振荡器 OSC 32.768KHz	内部 RC 高频振荡器 HIRC 32MHz
预热时钟数	$2^{14}$ clk	$2^{10}$ clk

### 复位预热

复位类型	上电/外部复位	LVR/WDT/SWRST
预热时间	10ms+振荡器预热	≈振荡器预热

芯片上电后，先进行上电预热，再进行振荡器预热，完成后运行程序。

## 5.7 工作模式

芯片及内核支持高速运行模式、低速运行模式、停止模式、休眠模式共 4 种工作模式。

工作模式	模式描述	功耗
高速运行模式	CORE 时钟运行在高频时钟（FHIRC）上，所有外设均可正常工作。	高
低速运行模式	CORE 时钟运行在低频时钟（FLIRC）上，大部分外设可正常工作。	中
停止模式	CORE 时钟停止，程序计数器（PC）挂起，但时钟源和所有外设均继续保持原有状态，用户可通过各种外设产生的中断快速唤醒 CORE。	低
休眠模式	CORE 时钟停止（FHIRC 时钟源关闭、FLIRC 时钟源可选关闭），程序计数器（PC）挂起，仅有限的外设可以继续工作，用户仅可通过这部分外设产生的中断唤醒 CORE。	最低

## 5.8 工作模式与时钟源状态

芯片上电、复位以及从休眠模式下唤醒后系统初始为高速运行模式，时钟源为 HIRC，需通过软件配置进入低速运行模式、停止模式或休眠模式。而从停止模式下唤醒后，将恢复为进入停止模式之前的工作模式。

### 5.8.1 停止模式

在高速或低速运行模式时，通过先对寄存器 SLPCR 写入“55H”、再将寄存器 PCON 中 STOP 位置“1”，则系统进入停止模式，芯片进行相应处理：

- ◇ CORE 停止工作；
- ◇ LDO1P5 开启
- ◇ 内部高/低频 RC 振荡器、外部晶体振荡器，均维持原工作状态
- ◇ RAM 内容保持不变
- ◇ 所有的输入输出端口保持不变；
- ◇ 全部外设可以根据用户的设定，停止或保持继续工作；

以下情况可退出停止模式：



- ◇ 发生一个有效的中断请求（可以参考中断章节的内容）；
- ◇ 发生一个有效的复位（可以参考中断章节的内容）；

中断请求导致的芯片退出停止模式，经过振荡等待后，会将 STOP 清 0，所有时钟源均保持原工作状态，系统恢复为进入停止模式前的工作模式。

注：复位导致的芯片退出停止模式，芯片直接复位

### 5.8.2 休眠模式

在高速或低速运行模式时，通过先对寄存器 SLPCR 写入“55H”、再将寄存器 PCON 中 SLEEP 位置“1”，则系统进入休眠模式，芯片进行相应处理：

- ◇ CORE 停止工作；
- ◇ FLASH 进入低功耗模式
- ◇ LDO1P5 进入低功耗模式
- ◇ 高频振荡器停止，低频振荡器根据 LIRCEN\_LP 位决定停止或工作；用户配置字 WDTM 选择“WDT 始终开启”，则 LIRCEN\_LP=0 无效；
- ◇ RAM 内容保持不变；
- ◇ 所有的输入输出端口保持不变；
- ◇ 部分外设可以根据用户的设定，停止或保持继续工作；

以下情况可退出休眠模式：

- ◇ 发生外部中断请求（可以参考中断章节的内容）；
- ◇ 发生定时器 T3 中断（可以参考中断章节的内容）；
- ◇ 发生 LVD 中断（可以参考中断章节的内容）
- ◇ 发生一个有效的复位（可以参考中断章节的内容）；

芯片在非复位情况下退出休眠模式，在经过振荡等待后，会将 SLEEP 清 0，并将 HIRC 作为系统时钟源以高速模式恢复工作，相应的系统时钟控制位均恢复为复位值。

注：唤醒后用户需要等待至少 100us 后才能开始进行 IAP 操作。

## 5.9 高级能耗控制

芯片支持高级能耗控制，可通过单独的寄存器控制位停止对应外设模块的内部时钟，以进一步降低芯片本身的功耗。详见高级能耗控制寄存器章节。

## 5.10 相关寄存器

系统时钟寄存器汇总表

地址 (SFRn/ H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
SFR0/F 9	PWRCR1	高级能耗控制寄存器 1	-	IIC_CLKE N	CRC_CLK EN	-	-	T3_CLKE N	T2_CLKE N	T01_CLK EN	-11- -111
SFR0/F	PWRCR	高级能耗控制寄	-	-	PWM1_C	PWM0_C	SPI_CLK	ADC_CLK	-	OCD_CL	--11



8		寄存器			LKEN	LKEN	EN	EN		KEN	11-1
SFR0/C 6	CLKCR1	时钟控制寄存器 1	-	CLKSW	FRTCSW	-	-	-	-	CLKSWS	-00- ---0
SFR0/C 5	OSCSS	时钟安全寄存器	-	FHS2	FHS1	FHS0	-	XTSPD	-	-	-001 01--
SFR0/C 0	CLKCR0	时钟控制寄存器 0	HIRCEN	-	OSCEN	LIRCEN_ LP	HIRCRDY	-	OSCRDY	LIRC RDY	1-00 1-01
SFR0/8 7	PCON	电源控制寄存器	-	-	-	UART0EN	SMOD0	SSTAT0	SLEEP	STOP	---0 0000
SFR0/8 6	SLPCR	模式保护控制寄 存器	SLPCR7	SLPCR6	SLPCR5	SLPCR4	SLPCR3	SLPCR2	SLPCR1	SLPCR0	0000 0000

### 5.10.1 时钟控制寄存器 0 (CLKCR0, 0xC0/SFR0)

SBIT 寄存器	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CLKCR0	HIRCEN	-	OSCEN	LIRCEN_LP	HIRCRDY	-	OSCRDY	LIRC RDY
R/W	R/W	-	R/W	R/W	R	-	R	R
复位值	1	-	0	0	1	-	0	1

BIT[7] HIRCEN – 内部高频 RC 振荡器 HIRC 使能位

- 0: 关闭 HIRC;
- 1: 开启 HIRC。

BIT[5] OSCEN – 外部低频晶体使能位

- 0: 关闭低频晶体振荡器;
- 1: 使能外部低频晶体振荡器。

BIT[4] LIRCEN\_LP – 低功耗模式低频振荡器使能位

- 0: 任意模式下, 当芯片进入“休眠模式”后, LIRC 自动停止; 停止模式, LIRC 始终工作;
- 1: 任意模式下, 当芯片在进入“停止模式”和“休眠模式”后, LIRC 始终工作。

注 1: 这里的“任意模式”指, “高速/低速运行模式”。

注 2: 若用户配置字中的“WDT 工作模式选择位”处于“始终开启看门狗”状态, LIRCEN\_LP 控制的 LIRC 是不能被直接关闭的, 设置 LIRCEN\_LP=0 对 LIRC 无效。

BIT[3] HIRCRDY – 内部高频 RC 振荡器 HIRC 稳定标志位

- 0: HIRC 未开启、停振或未稳定, 由硬件清 0, 可读但写入无效;
- 1: HIRC 已稳定运行, 由硬件置 1, 可读但写入无效。

BIT[1] OSCRDY – 外部 32768Hz 晶体振荡器 OSC 稳定标志位

- 0: OSC 未开启、停振或未稳定, 由硬件清 0, 可读但写入无效;
- 1: OSC 已稳定运行, 由硬件置 1, 可读但写入无效。

注: 当 T0/T1/T3 需要使用 OSC 时, 必须等待该位置 1, 置 1 说明振荡器已经稳定, 方可切换; 否则定时器计数将不准确。

BIT[0] LIRC RDY – 内部低频 RC 振荡器 LIRC 稳定标志位

- 0: LIRC 未开启、停振或未稳定, 由硬件清 0, 可读但写入无效;
- 1: LIRC 已稳定运行, 由硬件置 1, 可读但写入无效。



## 5.10.2 时钟控制寄存器 1 (CLKCR1, 0xC6/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CLKCR1	-	CLKSW	FRTCSW	-	-	-	-	CLKSWS
R/W	-	R/W	R/W	-	-	-	-	R
复位值	-	0	0	-	-	-	-	0

BIT[6] CLKSW – 系统时钟源选择位  
 0: 选择内部高频 RC 振荡器作为 FSYS 时钟源;  
 1: 选择内部低频 RC 振荡器作为 FSYS 时钟源。

BIT[5] FRTCSW – 定时器 T0/T1/T3 低频时钟 FRTC 时钟源选择位  
 0: 选择内部低频 RC 振荡器时钟 FLIRC 作为 FRTC 时钟源;  
 1: 选择外部低频晶体振荡器时钟 FOSC 作为 FRTC 时钟源。

注: FRTC 在 FOSC 和 FLIRC 之间切换时, 无论待切换的时钟源是否已正常工作, 都将直接切换。

BIT[0] CLKSWS – 系统时钟源状态位  
 0: FSYS 时钟源为高频 RC 振荡器;  
 1: FSYS 时钟源为低频 RC 振荡器。

## 5.10.3 时钟安全寄存器 (OSCSS, 0xC5/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OSCSS	-	FHS2	FHS1	FHS0	-	XTSPD	-	-
R/W	-	R/W	R/W	R/W	-	R/W*	-	-
复位值	-	0	0	1	-	1	-	-

BIT[6:4] FHS[2:0] – 高频时钟分频控制  
 000: 1 分频;  
 001: 2 分频;  
 010: 4 分频;  
 011: 8 分频;  
 100: 16 分频;  
 101: 32 分频;  
 110: 64 分频;  
 111: 128 分频。

BIT[2] XTSPD – 外部 32768Hz 振荡器加速模式控制位  
 0: 32768Hz 振荡器常规模式, 由软件清 0;  
 1: 32768Hz 振荡器加速模式, 由软件或硬件置 1。

注: 当使用 32768Hz 振荡器时, 此位在系统发生复位后, 由硬件自动置 1, 以加速 32768Hz 振荡器的起振时间; 如有需要, 此位也可由软件置 1 以便在模式切换时加速外部振荡器的起振, 或软件清 0 以降低功耗。

## 5.10.4 电源控制寄存器 (PCON, 0x87/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCON	-	-	-	UART0EN	SMOD0	SSTAT0	SLEEP	STOP
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
复位值	-	-	-	0	0	0	0	0



BIT[1] SLEEP – 休眠模式控制位  
 0: 未进入休眠模式, 退出休眠模式后硬件清 0;  
 1: 进入休眠模式 (必须紧跟解锁保护后), 用户是无法读取到 (1) 这个值的。

BIT[0] STOP – 停止模式控制位  
 0: 未进入停止模式, 退出停止模式后硬件清 0;  
 1: 进入停止模式 (必须紧跟解锁保护后), 用户是无法读取到 (1) 这个值的。

注 1: 以上两个模式控制位, 停止/休眠模式都是 CORE 的节能模式, CORE 同一时刻只会进入其中一种模式, 并不会嵌套进入; 若这两位同时置 1, CORE 优先进入休眠 SLEEP 模式。

注 2: 特别注意, 对上述两位 (PCON[1:0]) 操作时, 必须先将寄存器 SLPCR 写 “55H”, 然后立刻将 STOP/SLEEP 位置 1, 才能使系统进入停止/休眠模式, 否则下个时钟周期到来后, 本次操作失效。

### 5.10.5 模式保护控制寄存器 (SLPCR, 0x86/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SLPCR	SLPCR7	SLPCR6	SLPCR5	SLPCR4	SLPCR3	SLPCR2	SLPCR1	SLPCR0
W	W	W	W	W	W	W	W	W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] SLPCR[7:0] – 模式切换保护位

SLPCR[7:0]	模式保护允许位
55H	允许进入停止/休眠模式
其他	禁止进入停止/休眠模式

注: 该寄存器为只写寄存器, 只有先将寄存器 SLPCR 写 “55H” 再写 STOP/SLEEP 位置 1 的连续 2 条指令, 才能使系统进入停止/休眠模式, 否则下个时钟周期到来后, 寄存器 SLPCR 和 STOP/SLEEP 位中先前写入的值将被硬件清 0。

CORE 进入 STOP 停止模式, C 语言代码应用举例:

```
#include <intrins.h>
.....
sfr SLPCR = 0x86;
sfr PCON = 0x87;

SLPCR = 0x55; //解锁保护寄存器
```

CORE 进入 SLEEP 休眠模式, C 语言代码应用举例:

```
#include <intrins.h>
.....
sfr SLPCR = 0x86;
sfr PCON = 0x87;

SLPCR = 0x55; //解锁保护寄存器
```

### 5.10.6 高级能耗控制寄存器 (PWRCR, 0xF8/SFR0)

SBIT 寄存器	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
----------	-------	-------	-------	-------	-------	-------	-------	-------



PWRCR	-	-	PWM1_CLK EN	PWM0_CLK EN	SPI_CLKEN	ADC_CLKE N	-	OCD_CLKE N
R/W	-	-	R/W	R/W	R/W	R/W	-	R/W
复位值	-	-	1	1	1	1	-	1

BIT[5] PWM1\_CLKEN – PWM1 时钟控制位

- 0: 关闭 PWM1 模块的时钟;
- 1: 打开 PWM1 模块的时钟。

BIT[4] PWM0\_CLKEN – PWM0 时钟控制位

- 0: 关闭 PWM0 模块的时钟;
- 1: 打开 PWM0 模块的时钟。

BIT[3] SPI\_CLKEN – SPI 时钟控制位

- 0: 关闭 SPI 模块的时钟;
- 1: 打开 SPI 模块的时钟。

BIT[2] ADC\_CLKEN – ADC 时钟控制位

- 0: 关闭 ADC 模块的时钟;
- 1: 打开 ADC 模块的时钟。

BIT[0] OCD\_CLKEN – OCD 时钟控制位

- 0: 关闭 OCD (片上调试) 模块的时钟;
- 1: 打开 OCD (片上调试) 模块的时钟。

注: 该寄存器中的位需要在模块不使用的情况下, 才能被关闭; 否则芯片有保护机制, “关闭”动作会失败。

#### 5.10.7 高级能耗控制寄存器 1 (PWRCR1, 0xF9/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWRCR1	-	IIC_CLKEN	CRC_CLKE N	-	-	T3_CLKEN	T2_CLKEN	T01_CLKE N
R/W	-	R/W	R/W	-	-	R/W	R/W	R/W
复位值	-	1	1	-	-	1	1	1

BIT[6] IIC\_CLKEN – IIC 时钟控制位

- 0: 关闭 IIC 模块的时钟;
- 1: 打开 IIC 模块的时钟。

BIT[5] CRC\_CLKEN – CRC 时钟控制位

- 0: 关闭 CRC 模块的时钟;
- 1: 打开 CRC 模块的时钟。

BIT[2] T3\_CLKEN – T3 时钟控制位

- 0: 关闭 T3 模块的时钟;
- 1: 打开 T3 模块的时钟。

BIT[1] T2\_CLKEN – T2 时钟控制位

- 0: 关闭 T2 模块的时钟;
- 1: 打开 T2 模块的时钟。

BIT[0] T01\_CLKEN – T01 时钟控制位



0: 关闭 T01 模块的时钟;

1: 打开 T01 模块的时钟。

注: 该寄存器中的位需要在模块不使用的情况下, 才能被关闭; 否则芯片有保护机制, “关闭”动作会失败。

规划局

Sinomcu.com



## 6 复位

片上有五种复位方式

- ◇ 上电复位 POR;
- ◇ 外部复位;
- ◇ 低电压复位 LVR;
- ◇ 看门狗复位;
- ◇ 软件复位。

任何一种复位发生后，系统将会重新从 0000H 地址处开始执行指令；另外系统还会将大部分的特殊功能寄存器重置为默认复位值。并且可用通过寄存器标志位可以确定复位源。

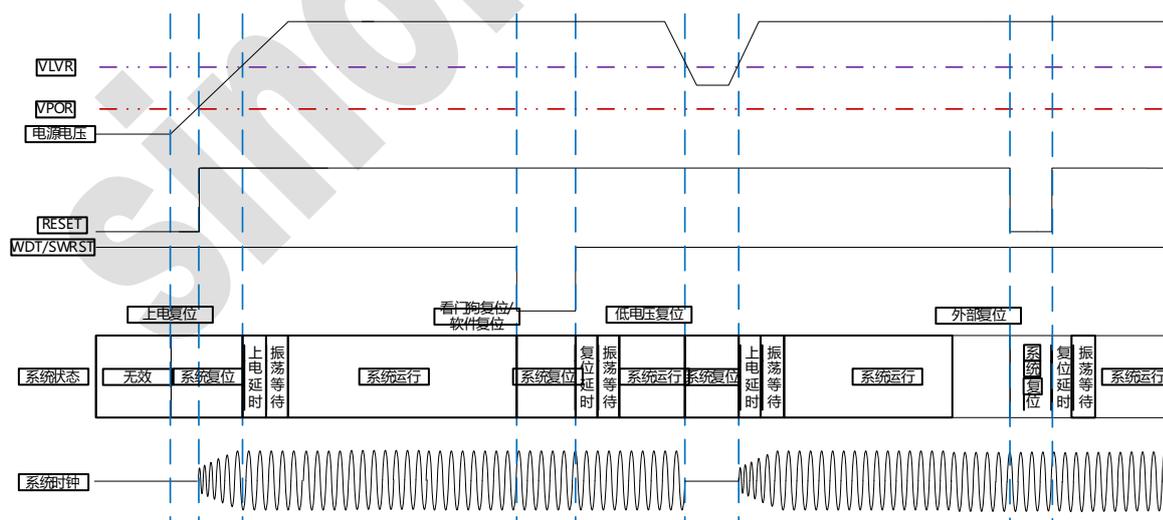
复位寄存器汇总表

地址 (SFRn/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
SFR0/E8	RSTFLAG	复位标志寄存器	PORF	LVRF	RSTF	WDTRF	SWRF	-	-	SWRST	0000 0--0
SFR0/CF	WDTCR	看门狗定时器控制寄存器	-	-	-	-	-	WDTOT2	WDTOT1	WDTOT0	---- -111

### 6.1 复位条件

CORE 根据复位方式的不同，执行不同的复位动作，根据配置信息完成相应的模块配置后，在外部条件满足时才结束复位，系统重新开始工作。

系统状态与五种复位的关系示意图





## 6.2 上电复位

在上电过程中，当 VDD 低于参考电压门限值，上电复位将保持芯片为复位模式。这种设计使芯片在 VDD 不满足执行读取存储器时，不访问程序存储器空间。如果从程序存储器读取并执行一个不确定的操作码，可能会使芯片甚至是整个系统进入错误状态。VDD 上升到参考门限电压以上，系统工作，所选的振荡器起振，程序从 0000H 开始执行，同时 PORF 上电标志位将置 1。若发生上电复位，上电标志 PORF 置 1，且上电后内部 RAM 的值为不确定，强烈建议用户初始化 RAM 再使用。

## 6.3 外部复位

外部复位功能是否开启可以通过芯片配置信息进行设置，选择外部复位功能后复位引脚的内部上拉电阻自动有效。外部复位引脚 RST 是施密特结构的，低电平有效。当外部复位引脚为高电平时，系统正常运行；为低电平时，系统产生复位。若发生外部复位，程序从 0000H 开始执行。同时外部复位 RSTF 标志位将置 1。

## 6.4 低电压复位

芯片的 LVR 电压有 4 级可选：2.1V、2.5V、3.5V、4.1V。LVR 的电压检测电路有一定的回滞特性，通常回滞电压为 0.05V 左右，则当电源电压下降到 LVR 电压时 LVR 复位有效，而电压需要上升到 LVR 电压+0.05V 时 LVR 复位才会解除。若发生低电压复位，程序从 0000H 开始执行。同时低电压 LVRF 标志位将置 1。

LVR 电压通过芯片的“用户配置字”VLVRS 进行设置，软件无法调整复位电压等级。

LVR 具有独立的模拟滤波电路和数字滤波电路，可通过用户配置字 LVRAFEN/LVRDFSEL 配置 LVR 输出的滤波时间。

## 6.5 看门狗复位

看门狗复位是一种对程序正常运行的保护机制。正常情况下，用户软件需要按时对 WDT 定时器进行清零操作，保证 WDT 不溢出。若出现异常状况，程序未按时对 WDT 定时器清零，WDT 会溢出从而产生看门狗复位，系统重新初始化，返回受控状态。若发生看门狗复位，程序从 0000H 地址开始执行。同时看门狗复位 WDTRF 标志位将被置 1。

WDT 开启与关闭通过“用户配置字”进行设置，软件无法调整；WDT 只能通过用户配置字的 WDTM 选择“始终开启 WDT”、“停止/休眠模式关闭 WDT”、“始终关闭 WDT”等 3 种工作模式。

- (1) 若选择始终开启 WDT 模式，在停止/休眠模式下 WDT 依然运行，WDT 溢出后将复位芯片。
- (2) 若选择停止/休眠模式下关闭 WDT 模式，则 WDT 在停止/休眠模式下被硬件自动关闭，并在芯片恢复运行后自动继续。
- (3) 若选择始终关闭 WDT 模式，则整个 WDT 模块功能将失效，“用户配置字”中默认为此项。

当 WDT 开启时，默认为最大溢出时间，用户可以通过改变 WDTOT[2:0] 的值，来调整的 WDT 溢出时间。因为 WDT 的时钟源来自 LIRC，故一旦在“用户配置字”中选择“始终开启 WDT”后，LIRC 将不能在软件中被关闭，将始终为 WDT 提供时钟源。

WDT 设计为一个向上计数的计数器，计数到设定值以后产生溢出，触发对应的看门狗复位事件。该计数器的值对用户不可见，但用户可以通过读/写一次 WDTCR 寄存器，实现“喂狗操作”，该动作发生的同时，硬件上自动完成一次对 WDT 计数器的重载。

### 6.5.1 看门狗定时器控制寄存器 (WDTCR, 0xCF/SFR0)



	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WDTCR	-	-	-	-	-	WDTOT2	WDTOT1	WDTOT0
R/W	-	-	-	-	-	R/W	R/W	R/W
复位值	-	-	-	-	-	1	1	1

BIT[2:0] WDTOT[2:0] – WDT 溢出时间控制位

WDT[2:0]	WDT 溢出时间
000	8ms
001	16ms
010	32ms
011	64ms
100	128ms
101	256ms
110	512ms
111	1024ms (复位值)

注 1: 上表中的“WDT 溢出时间”是一个理论的溢出时间, 实际溢出时间和 LIRC 有关, 请以 LIRC 规格为准;

注 2: 寄存器的配置 (用户软件上) 仅可控制看门狗的溢出时间; 但是看门狗定时器开启与否, 必须在“用户配置字”中设定。

注 3: 在 WDT 的溢出时间之内, 读/写一次 WDTCR 寄存器, 即完成一次“喂狗操作”; 此时硬件会自动重载一次 WDT 计数器的值, 并继续开始向下递减计数。

喂狗操作 (清 WDT), C 语言代码应用举例:

```
sfr WDTCR = 0xCF;
```

## 6.6 软件复位

软件复位是一种对可控的程序自发的复位行为, 有点类似于外部复位。正常情况下, 用户软件需要连续两次对 SWRST 位写两次, 启动软件复位。若发生软件复位, 程序从 0000H 开始执行。同时软件复位 SWRF 标志位将置 1。

注: 软件复位, 不会加载用户配置字 BOOTS 位, 其他复位 (除 wdt 复位) 会加载 BOOTS 至 IAPCR1->BOOT(BIT7)。

### 6.6.1 复位标志寄存器 (RSTFLAG, 0xE8/SFR0)

SBIT 寄存器	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RSTFLAG	PORF	LVRF	RSTF	WDTRF	SWRF	-	-	SWRST
R/W	R/W	R/W	R/W	R/W	R/W	-	-	R/W
复位值	0	0	0	0	0	-	-	0

BIT[0] SWRST – 软件复位

0: 无软件复位要求;

1: 启动软件复位 (连续写两遍有效), 系统软件复位后硬件清 0, 并将 SWRF 标志位置 1。

注 1: 只有对该位连续写 2 条相同的指令, 才能使系统进行软件复位, 否则下个时钟周期到来后, 寄存器中先前写



入的值将被硬件自动清掉，保持为 0，这是一种简单的保护机制。

注 2: 写入 1 成功后，芯片直接“软件复位”，用户查询不到该值为 1 的状态。

软件复位，C 语言代码应用举例：

```
sfr RSTFLAG = 0xE8;

RSTFLAG |= 0x01;
```

## 6.7 复位标志寄存器

系统在热复位的条件下，用户可以通过查看复位标志寄存器，确认系统复位的原因。

### 6.7.1 复位标志寄存器 (RSTFLAG, 0xE8/SFR0)

SBIT 寄存器	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RSTFLAG	PORF	LVRF	RSTF	WDTRF	SWRF	-	-	SWRST
R/W	R/W	R/W	R/W	R/W	R/W	-	-	R/W
复位值	0	0	0	0	0	-	-	0

BIT[7] PORF – 上电复位标志位

0: 未发生上电复位;

1: 曾经发生过上电复位，硬件置 1 但不会清 0，需要用户手动清。

BIT[6] LVRF – 低电压复位标志位

0: 未发生低电压复位;

1: 曾经发生过低电压复位，硬件置 1 但不会清 0，需要用户手动清。

BIT[5] RSTF – 外部复位标志位

0: 未发生外部复位;

1: 曾经发生过外部复位，硬件置 1 但不会清 0，需要用户手动清。

BIT[4] WDTRF – 看门狗复位标志位

0: 没有发生过看门狗定时器溢出复位;

1: 曾经发生过看门狗定时器溢出复位，硬件置 1 但不会清 0，需要用户手动清。

BIT[3] SWRF – 软件复位标志位

0: 没有发生过软件复位;

1: 曾经发生过软件复位，硬件置 1 但不会清 0，需要用户手动清。

注 1: 以上 5 个复位标志在置位 (=1) 后，需要手动清除，否则硬件不会自动清零 (=0)；建议用户在每次使用这些标志位之前，将以上 5 个“复位标志位”全部清零。

注 2: 若不清零，不同复位原因标志位之间可以累积记录；但对于同一复位原因标志位，前后复位之间，对已经置位 (=1) 的标志位将无影响，所以用户可能因此检查不到“重复复位”的标志位。



## 6.8 各种复位与低功耗模式的关系

### 复位与停止模式和休眠模式的关系

复位源	停止模式中 (STOP=1) 复位有效	休眠模式中 (SLEEP=1) 复位有效
POR 复位	Yes	Yes
外部复位	Yes	Yes
低电压复位	Yes	Yes <sup>注2</sup>
看门狗复位	Yes	Yes
软件复位	No	No

注 1: 通过上表中的部分复位操作, 虽然可以让芯片从停止模式和休眠模式迁移出来, 虽然这是一种退出“低功耗模式”的手段, 但是程序将从 0x0000 地址开始执行, 因此可能造成实际运行结果与用户的预想的结果截然不同。

注 2: 通过“用户配置字中”关闭休眠模式下的 LVR 功能, 则不会产生 LVR 复位。



## 7 I/O 端口

### 7.1 端口特性

芯片共有 3 组 18 个 IO 端口，包括 1 组 2 位端口 P0 和 2 组 8 位端口的 P1、P2。

每个 IO 都有 4 种输入/输出模式可选：推挽输出模式、输入模式、高阻态模式、开漏输出模式。

推挽输出模式：此模式下具有很强的输出驱动能力。

输入模式：此模式下可实现输入功能，支持不带上拉或下拉电阻、带上拉或下拉电阻等多种输入方式。

高阻态模式：此模式下既不输入也不输出，既不是高电平也不是低电平，“高阻态模式”可以用来隔绝芯片和外围电路之间的相互影响，近似于断开。

开漏输出模式：此模式下仅有输出低驱动能力，而无输出高驱动能力，可配置内部上/下拉电阻

### 7.2 端口相关寄存器

端口寄存器汇总表

地址 (SFRn/ H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
SFR0/C 4	P0PD	P0 端口下拉电阻 控制寄存器	-	-	-	-	-	-	P01PD	P00PD	---- --00
SFR0/C 3	P0PU	P0 端口上拉电阻 控制寄存器	-	-	-	-	-	-	P01PU	P00PU	---- --00
SFR0/C 1	P0MOD0	P0 端口模式寄存 器 0	-	-	-	-	P01M1	P01M0	P00M1	P00M0	---- 1010
SFR0/A 4	P2PD	P2 端口下拉电阻 控制寄存器	P27PD	P26PD	P25PD	P24PD	P23PD	P22PD	P21PD	P20PD	0000 0000
SFR0/A 3	P2PU	P2 端口上拉电阻 控制寄存器	P27PU	P26PU	P25PU	P24PU	P23PU	P22PU	P21PU	P20PU	0000 0000
SFR0/A 2	P2MOD1	P2 端口模式寄存 器 1	P27M1	P27M0	P26M1	P26M0	P25M1	P25M0	P24M1	P24M0	1010 1010
SFR0/A 1	P2MOD0	P2 端口模式寄存 器 0	P23M1	P23M0	P22M1	P22M0	P21M1	P21M0	P20M1	P20M0	1010 1010
SFR0/A 0	P2	P2 端口数据寄存 器	P27D	P26D	P25D	P24D	P23D	P22D	P21D	P20D	xxxx xxxx
SFR0/9 4	P1PD	P1 端口下拉电阻 控制寄存器	P17PD	P16PD	P15PD	P14PD	P13PD	P12PD	P11PD	P10PD	0000 0000
SFR0/9 3	P1PU	P1 端口通用上拉 电阻控制寄存器	P17PU	P16PU	P15PU	P14PU	P13PU	P12PU	P11PU	P10PU	0000 0000
SFR0/9 2	P1MOD1	P1 端口模式寄存 器 1	P17M1	P17M0	P16M1	P16M0	P15M1	P15M0	P14M1	P14M0	1010 1010
SFR0/9 1	P1MOD0	P1 端口模式寄存 器 0	P13M1	P13M0	P12M1	P12M0	P11M1	P11M0	P10M1	P10M0	1010 1010
SFR0/9	P1	P1 端口数据寄存 器	P17D	P16D	P15D	P14D	P13D	P12D	P11D	P10D	xxxx



0		器									xxxx
SFR0/8	P0	P0 端口数据寄存器	-	-	-	-	-	-	P01D	P00D	---- --XX

### 7.3 端口模式及数据控制

#### 7.3.1 P0 端口模式寄存器 0 (P0MOD0, 0xC1/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P0MOD0	-	-	-	-	P01M1	P01M0	P00M1	P00M0
R/W	-	-	-	-	R/W	R/W	R/W	R/W
复位值	-	-	-	-	1	0	1	0

#### 7.3.2 P1 端口模式寄存器 0 (P1MOD0, 0x91/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P1MOD0	P13M1	P13M0	P12M1	P12M0	P11M1	P11M0	P10M1	P10M0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	0	1	0	1	0	1	0

#### 7.3.3 P1 端口模式寄存器 1 (P1MOD1, 0x92/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P1MOD1	P17M1	P17M0	P16M1	P16M0	P15M1	P15M0	P14M1	P14M0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	0	1	0	1	0	1	0

BIT[7:0] P1nM[1:0] (n = 7 - 0) – P1 端口模式选择位，初始状态为高阻态模式

#### 7.3.4 P2 端口模式寄存器 0 (P2MOD0, 0xA1/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P2MOD0	P23M1	P23M0	P22M1	P22M0	P21M1	P21M0	P20M1	P20M0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	0	1	0	1	0	1	0

#### 7.3.5 P2 端口模式寄存器 1 (P2MOD1, 0xA2/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P2MOD1	P27M1	P27M0	P26M1	P26M0	P25M1	P25M0	P24M1	P24M0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	0	1	0	1	0	1	0

BIT[7:0] P2nM[1:0] (n = 7 - 0) – P2 端口模式选择位，初始状态为高阻态模式

#### 端口输入/输出模式对应表

PxnM1 (x=0-3; n=7-0)	PxnM0 (x=0-3; n=7-0)	IO 工作模式
0	0	输入模式
0	1	推挽输出模式
1	0	高阻态模式 (复位值)
1	1	开漏输出模式



端口数据寄存器 P0/P1/P2，全部支持位寻址。

### 7.3.6 P0 端口数据寄存器 (P0, 0x80/SFR0)

SBIT 寄存器	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P0	-	-	-	-	-	-	P01D	P00D
R/W	-	-	-	-	-	-	R/W	R/W
复位值	-	-	-	-	-	-	X	X

BIT[7:0] P0nD (n = 7 - 0) – P0 端口数据位

### 7.3.7 P1 端口数据寄存器 (P1, 0x90/SFR0)

SBIT 寄存器	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P1	P17D	P16D	P15D	P14D	P13D	P12D	P11D	P10D
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	X	X	X	X	X	X	X	X

BIT[7:0] P1nD (n = 7 - 0) – P1 端口数据位

### 7.3.8 P2 端口数据寄存器 (P2, 0xA0/SFR0)

SBIT 寄存器	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P2	P27D	P26D	P25D	P24D	P23D	P22D	P21D	P20D
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	X	X	X	X	X	X	X	X

BIT[7:0] P2nD (n = 7 - 0) – P2 端口数据位

端口输出电平对应表

PxnD[1:0] (x = 0-2; n = 7-0)	IO 输出电平
0	输出低电平
1	输出高电平
X (复位值)	输出不确定

注: P0、P1、P2 口的数据值在复位之后为不确定状态 (X)，建议用户使用之前先写入一个确定的初始电平值。

## 7.4 输入上/下拉电阻控制

P0、P1、P2 端口的每位都内置有独立的输入上拉和下拉电阻控制。无论端口处于输入或输出模式都配置有效。

### 7.4.1 P0 端口上拉电阻控制寄存器 (P0PU, 0xC3/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P0PU	-	-	-	-	-	-	P01PU	P00PU
R/W	-	-	-	-	-	-	R/W	R/W
复位值	-	-	-	-	-	-	0	0

BIT[1:0] P0nPU (n = 1 - 0) – P0 端口上拉电阻选择位

### 7.4.2 P1 端口通用上拉电阻控制寄存器 (P1PU, 0x93/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P1PU	P17PU	P16PU	P15PU	P14PU	P13PU	P12PU	P11PU	P10PU



R/W								
复位值	0	0	0	0	0	0	0	0

BIT[7:0] P1nPU (n = 7 - 0) – P1 端口通用上拉电阻选择位

#### 7.4.3 P2 端口上拉电阻控制寄存器 (P2PU, 0xA3/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P2PU	P27PU	P26PU	P25PU	P24PU	P23PU	P22PU	P21PU	P20PU
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] P2nPU (n = 7 - 0) – P2 端口上拉电阻选择位

#### 7.4.4 P0 端口下拉电阻控制寄存器 (P0PD, 0xC4/SFR)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P0PD	-	-	-	-	-	-	P01PD	P00PD
R/W	-	-	-	-	-	-	R/W	R/W
复位值	-	-	-	-	-	-	0	0

BIT[1:0] P0nPD (n = 1 - 0) – P0 端口下拉电阻选择位

#### 7.4.5 P1 端口下拉电阻控制寄存器 (P1PD, 0x94/SFR)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P1PD	P17PD	P16PD	P15PD	P14PD	P13PD	P12PD	P11PD	P10PD
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] P1nPD (n = 7 - 0) – P1 端口下拉电阻选择位

#### 7.4.6 P2 端口下拉电阻控制寄存器 (P2PD, 0xA4/SFR)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P2PD	P27PD	P26PD	P25PD	P24PD	P23PD	P22PD	P21PD	P20PD
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] P2nPD (n = 7 - 0) – P2 端口下拉电阻选择位

端口上拉电阻选择对应表

PxnPU (x = 0-2; n = 7-0)	上拉电阻选择
0	无上拉电阻控制 (复位值)
1	输入带内部上拉

端口下拉电阻选择对应表

PxnPD (x = 0-2; n = 7-0)	下拉电阻选择
0	无下拉电阻控制 (复位值)
1	输入带内部下拉



## 7.5 端口复用设置

P0、P1、P2 的部分端口可以复用为其他功能模块的输入/输出端口，芯片默认的优先级按引脚排列中所示外部最高内部最低的原则，用户可根据需要通过寄存器配置成需要功能。

具体内容，请参考相关章节的内容。

注 1: 数字输入功能 (含数字复用功能) 在 IO 配置成非高阻模式 (即输入模式/推挽输出/开漏输出) 下始终有效;

注 2: 数字输出功能 (含数字复用功能) 必须在 IO 配置成推挽输出或开漏输出模式下才能有效, 优先级按照《引脚汇总表》;

注 3: 模拟端口功能, 除了电源 (VDD/GND) 和模拟输出功能 (外部晶体), 其它模拟输入功能可以在任何情况下生效, 能与数字功能共存; 模拟输出功能 IO 必须配置成高阻模式;

注 4: 在高阻模式下, 任何数字输入和输出功能都无效



## 8 定时器/计数器

### 8.1 定时器 T0/T1

芯片包含 2 个兼容 8051 标准的 16 位定时器 T0 和 T1，定时器的两个 8 位数据寄存器（THx\TLx, x=0,1）可作为一个 16 位寄存器来访问。

定时器具有 4 种工作方式，通过定时器模式寄存器的方式选择位（TxM[1:0]）来选择定时器 Tx 的工作方式。

#### 8.1.1 方式 0：13 位计数器/定时器

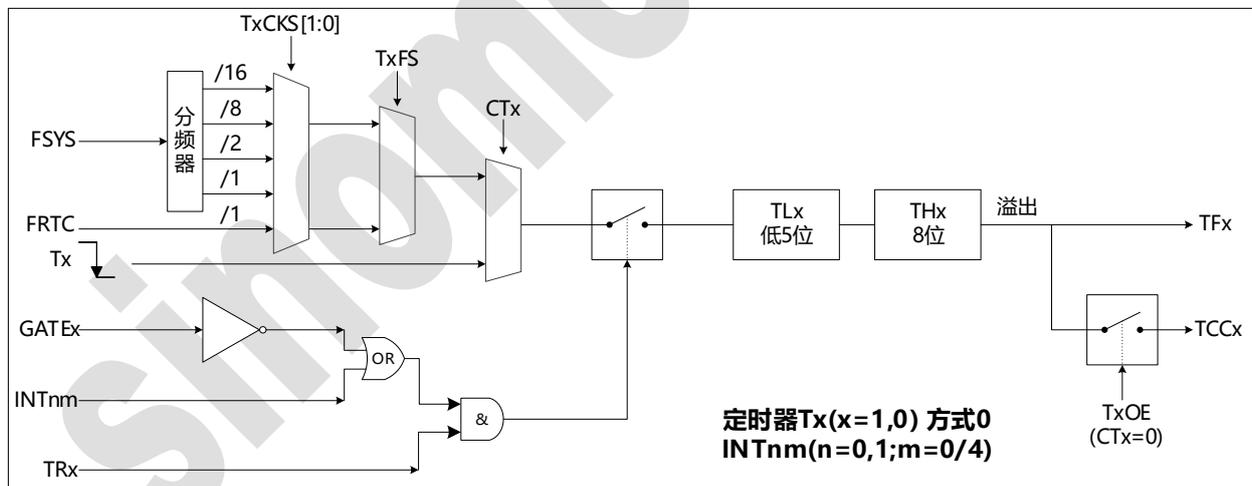
在方式 0 中，定时器 Tx (x=0,1) 为 13 位计数器/定时器。递增计数寄存器 THx[7:0]存放 13 位计数器/定时器的高 8 位，TLx[4:0]存放低 5 位，而 TLx[7:5]的值不确定，在读取时应被忽略。当 13 位计数寄存器溢出时，溢出标志 TFX 将被置 1，如果 Tx 中断被允许，将会产生定时器 Tx 中断。

CTx 位选择计数器/定时器 Tx 的时钟源。如果 CTx = 1，Tx 用作计数器，Tx 引脚将被设置成输入端口（需其他复用功能未复用该端口），Tx 的下降沿将使 Tx 计数寄存器加 1；如果 CTx = 0，Tx 用作定时器，可选择低频时钟 FRTC 或系统时钟 Fsys（或其分频）作为 Tx 的时钟源。

*注：寄存器位 FRTC SW，控制 T0/T1 的低频时钟源，可选择 FLOSC 或 FLIRC 作为 T0/T1 的低频时钟源。*

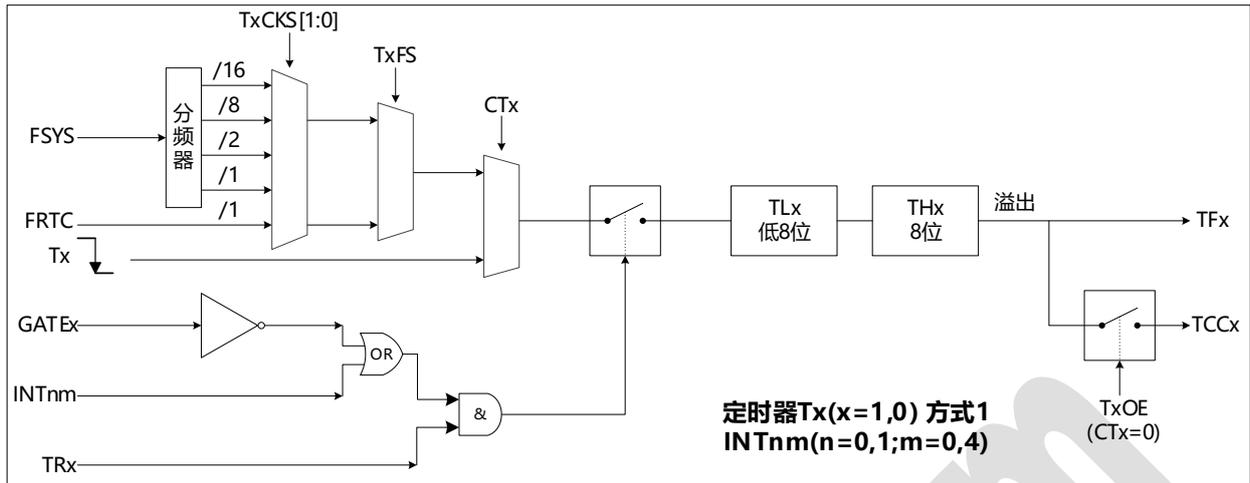
当 GATEx=0，或者 GATEx=1 且 INTnm 有效时，TRx 置 1 将启动定时器。GATEx 置 1，允许定时器 Tx 由外部输入信号 INTnm 控制，用于测量 INTnm 引脚上输入高电平的脉冲宽度。TRx 位置 1 不强行复位定时器，即 TRx 置 1 时计数寄存器将从上次 TRx 清 0 时的值恢复计数。所以在启动定时器之前应设定计数寄存器的复位值。

当 Tx 用作定时器时，可设置寄存器 TCON1 中的 TxOE 位使 Tx 计数值溢出时可以对 Tx 引脚电平自动翻转。仅当 TxOE 被置 1，Tx 引脚自动设置为输出有效（需其他复用功能未复用该端口）。



#### 8.1.2 方式 1：16 位计数器/定时器

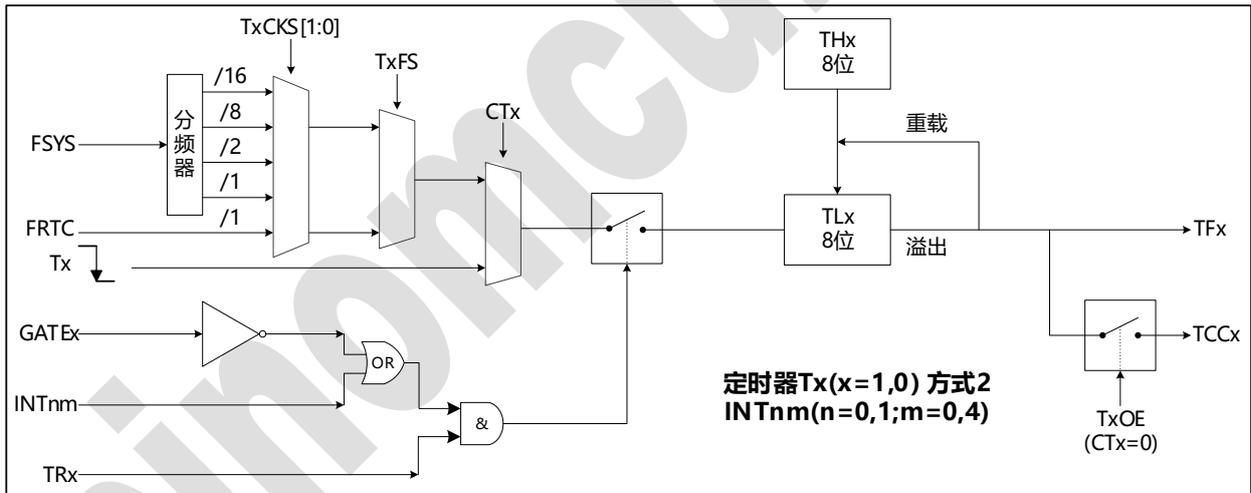
除了使用 16 位计数寄存器 THx/TLx 外，方式 1 的运行与方式 0 相同，打开和设置计数器/定时器也与方式 0 相同。



### 8.1.3 方式 2: 8 位自动重载计数器/定时器

方式 2 中, 定时器 Tx 用作 8 位自动重载计数器/定时器。TLx 存放计数值, THx 存放重载值。当递增计数器 TLx 溢出至 0x00 时, 溢出标志 TFx 将被置 1, 寄存器 THx 的值重新载入计数器 TLx 中, 而 THx 的重载值不变。如果 Tx 中断被允许, 将会产生定时器 Tx 中断。在此方式下, 首次打开定时器开始计数之前, 应将 TLx 初始化为重载值, 确保首次溢出时间与后续一致。

除自动重载功能以外, 方式 2 中的计数器/定时器的使能和设置与方式 1 和 0 基本相同。

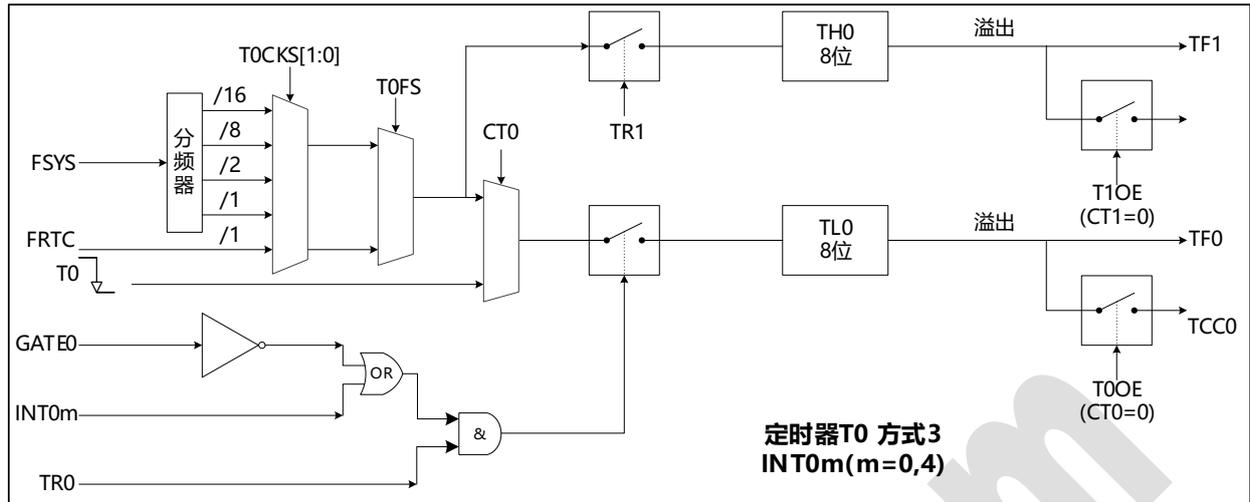


### 8.1.4 方式 3: 两个 8 位计数器/定时器 (只限于定时器 T0)

在方式 3 中, 定时器 T0 用作两个独立的 8 位计数器/定时器, 分别由 TL0 和 TH0 控制。TL0 使用定时器 T0 的控制 (TCON 中) 和状态 (TMOD 中) 位: TR0、CT0、GATE0 和 TF0。TL0 可采用振荡器时钟信号用作定时器, 也可采用外部输入信号用作计数器。

TH0 只能用作定时器功能, 时钟源来自振荡器时钟信号。TH0 借用定时器 T1 的控制位 TR1 控制使能, 溢出时借用定时器 T1 溢出标志 TF1 置 1, 产生 T1 中断。

当 T0 工作在方式 3 时, 定时器 T1 可以工作在方式 0、1 或 2, 但是无法将 TF1 标志置 1, 无法产生中断。TH1 和 TL1 只能用作定时器功能, 时钟源来自振荡器时钟信号, GATE1 无效。因为 TR1 被定时器 0 占用, 所以定时器 T1 改由方式设置来控制使能/关闭。设为方式 0/1/2 时 T1 使能, 设为方式 3 时 T1 被关闭。



## 8.2 T0/T1 相关寄存器

T0/T1 寄存器汇总表

地址 (SFRn/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
SFR0/8E	TCON1	定时器控制寄存器 1	T1FS	T1OE	T1CKS1	T1CKS0	T0FS	T0OE	T0CKS1	T0CKS0	0000 0000
SFR0/8D	TH1	定时器计数高位寄存器 1	TH17	TH16	TH15	TH14	TH13	TH12	TH11	TH10	0000 0000
SFR0/8C	TH0	定时器计数高位寄存器 0	TH07	TH06	TH05	TH04	TH03	TH02	TH01	TH00	0000 0000
SFR0/8B	TL1	定时器计数低位寄存器 1	TL17	TL16	TL15	TL14	TL13	TL12	TL11	TL10	0000 0000
SFR0/8A	TL0	定时器计数低位寄存器 0	TL07	TL06	TL05	TL04	TL03	TL02	TL01	TL00	0000 0000
SFR0/89	TMOD	定时器模式寄存器	GATE1	CT1	T1M1	T1M0	GATE0	CT0	T0M1	T0M0	0000 0000
SFR0/88	TCON	定时器控制寄存器	TF1	TR1	TF0	TR0	-	-	ADCCMP1 F	ADCIF	0000 --00

### 8.2.1 定时器模式寄存器 (TMOD, 0x89/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TMOD	GATE1	CT1	T1M1	T1M0	GATE0	CT0	T0M1	T0M0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7,3] GATE<sub>x</sub> - 定时器/计数器 Tx 门控制位 (x=1,0)

0: TR<sub>x</sub> 置 1 时, 定时器 Tx 立刻开始计数;

1: TR<sub>x</sub> 置 1 且外部 INT<sub>xn</sub> (x=1,0 与 Tx 保持一致; n=4) 信号同时有效 (具体可参考中断章节内容), 定时器 Tx 才开始计数。



注：当  $GATE_x=1$  时， $T0/T1$  只能通过  $INT04$  和  $INT14$  测量外部信号的脉宽，且仅测量高电平的脉宽。

BIT[6,2] CTx – Tx 的计数器/定时器方式选择位 ( $x=1,0$ )

0: 定时器方式，振荡器时钟信号触发，即内部有一个 clock，Tx 的计数器递增一次；

1: 计数器方式，外部输入下降沿触发，即 Tx 端口上一个下降沿的有效信号，Tx 的计数器递增一次，此时 Tx 口作为 Tx 定时器/计数器的专用输入口，且不能做其它复用功能。

BIT[5:4] T1M[1:0] – 定时器/计数器 T1 工作方式选择位

T1M[1:0]	定时器 T1 工作方式
00	方式 0
01	方式 1
10	方式 2
11	禁止 (T1 将停止工作)

注：T1 没有工作方式 3，用户强制使用 T1 的方式 3，T1 将停止工作。

BIT[1:0] TOM[1:0] – 定时器/计数器 T0 工作方式选择位

TOM[1:0]	定时器 T0 工作方式
00	方式 0
01	方式 1
10	方式 2
11	方式 3

### 8.2.2 定时器控制寄存器 (TCON, 0x88/SFR0)

SBIT 寄存器	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TCON	TF1	TR1	TF0	TR0	-	-	ADCCMPIF	ADCIF
R/W	R/W	R/W	R/W	R/W	-	-	R/W	R/W
复位值	0	0	0	0	-	-	0	0

BIT[7,5] TFx – 定时器/计数器 Tx 溢出标志位 ( $x=1,0$ )

0: 定时器/计数器 Tx 无溢出，当中断响应后硬件清 0，可由软件清 0；

1: 定时器/计数器 Tx 溢出时，由硬件置 1。

BIT[6,4] TRx – 定时器/计数器 Tx 使能位 ( $x=1,0$ )

0: 定时器/计数器 Tx 中止递增，THx 和 TLx 寄存器的数值将停在“中止”时候的数值；

1: 定时器/计数器 Tx 开始递增，或者 (THx 和 TLx 寄存器不重新赋值) 继续递增。

### 8.2.3 定时器控制寄存器 1 (TCON1, 0x8E/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TCON1	T1FS	T1OE	T1CKS1	T1CKS0	T0FS	T0OE	T0CKS1	T0CKS0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7,3] TxFS – 定时器/计数器方式时，Tx 时钟源选择位 ( $x=1,0$ )

0: Tx 时钟源为系统时钟  $F_{sys}$ ；

1: Tx 时钟源为低频时钟  $F_{RTC}$ ；

BIT[6,2] TxOE – 定时器/计数器 Tx 输出使能位 ( $x=1,0$ ) (CTx=0 时有效)

0: 禁止 Tx 溢出时翻转 Tx 端口电平，端口用于通用 IO 或其他复用模块；



1: 使能 Tx 端口复用于定时器 Tx, Tx 溢出时翻转端口电平, 并且第 1 次反转输出的是高电平 (逻辑 1), 第 2 次反转输出的是低电平 (逻辑 0), 之后依次交替反转。

BIT[5:4,1:0] TxCKS[1:0] - 定时器/计数器 Tx (x=1,0) 时钟分频选择位

TxCKS[1:0]	Tx 频率 (TxFS=0)	Tx 频率 (TxFS=1)
00	$F_{SYS}$	$F_{RTC}$
01	$F_{SYS}/2$	禁止 (T0/T1 将停止工作)
10	$F_{SYS}/8$	
11	$F_{SYS}/16$	

注: Tx 的工作频率  $F_{Tx}$ , 与用户选择的实际频率和分频比有关; 在低频 (TxFS=1) 的情况下, 用户仅有一档 (TxCKS[1:0]=00) 可选择, 若强行选择了“禁止”状态, 否则 Tx 将出现不可预知的情况。

#### 8.2.4 定时器计数高位寄存器 0/1 (TH0, 0x8C/SFR0), (TH1, 0x8D/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
THx	THx7	THx6	THx5	THx4	THx3	THx2	THx1	THx0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] THx[7:0] - 定时器/计数器 Tx 高 8 位计数器 (x=1,0)

#### 8.2.5 定时器计数低位寄存器 0/1 (TL0, 0x8A/SFR0), (TL1, 0x8B/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TLx	TLx7	TLx6	TLx5	TLx4	TLx3	TLx2	TLx1	TLx0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] TLx[7:0] - 定时器/计数器 Tx 低 8 位计数器 (x=1,0)

注: 无论是读或者写 TLx/THx 的值, 必须先停止 (TRx=0), 否则 Tx 将产生不可预知的情况。

### 8.3 定时器 T2

芯片包含 1 个 16 位增强型定时器 T2, 兼容 52 系列中标准型 T2, 但功能更强。T2 是一个向上递增型定时器/计数器。T2 具有 4 种工作在模式: 自动重载模式、捕获模式、波特率发生器模式、比较模式。

T2 工作方式	T2CON1		T2CON
	Bit 6	Bit 5	Bit 0
	CM2	BR2EN	CPRL2
方式 0: 16 位重载方式	0	0	0
方式 1: 16 位捕获方式	0	0	1
方式 3: 16 位比较方式	1	0	0
无效状态	其他值		

#### 8.3.1 方式 0: 16 位重载方式定时器/计数器方式

T2 的这种工作方式和 T0/T1 的工作方式 2 类似, 只不过 T0/T1 是 8 位的, 而 T2 则是 16 位的。

T2 的自动重载模式中, 时钟源由 CM2 位决定是对  $F_{SYS}$  计数, 还是对 T2 端口输入信号计数。RCAPH2 和 RCAPL2 保



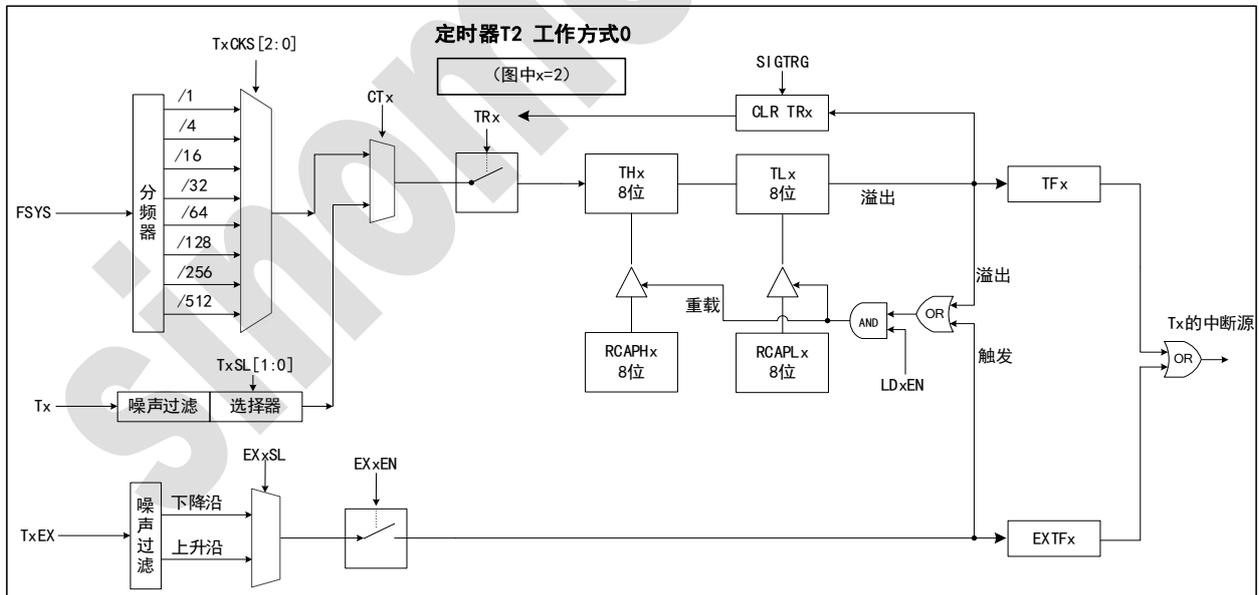
存着重复装载的数值。一旦产生了重载触发事件，硬件会自动把 RCAPH2 和 RCAPL2 中的值分别装入 TH2 和 TL2 中。重载触发事件指，T2 定时器溢出 (TF2=1) 或者是外部捕获信号有效 (T2EX 端口出现有效信号, EXTF2=1)，两者都可产生 T2 中断，并重新装载 TH2/TL2，使其开始新一轮的计数。

标准情况：使能 T2，将 T2 设置为方式 0，设置时钟频率、定时器模式 (CT2=0) 等等参数。启动 T2 (TR2=1)，打开重载功能 LD2EN=1，且禁止外部捕获信号 T2EX 输入 (EX2EN=0)，该情况不产生外部触发事件 (T2EX 端口无有效信号，始终是 EXTF2=0)，仅通过 T2 定时器产生溢出 (TF2=1) 产生触发事件的情况。向上递增计数中，TH2/TL2 (从 RCMPH2/RCMPL2 装载) 的值，开始递增计数直到 0x0 产生溢出，最终产生 TF2 标志位置位 (TF2=1) 和 T2 中断。

特殊情况一：用户觉得内部时钟源不好，可以改变时钟源。当 CT2=1 时，时钟源切换到外部，通过 T2 口送入。用户还可通过 T2SL 位，决定有效信号的方式，当 T2SL=01B 时上升沿有效，当 T2SL=10B 时下降沿有效，当 T2SL=11B 时双沿有效。当选择外部时钟时，由于内部系统时钟采样同步，外部最大不能超过  $F_{sys}/2$ 。

特殊情况二：不重载或者 T2EX 加入触发重载控制。当 LD2EN 位=0 关闭重载功能，则 T2 仅就做一次性定时任务，T2 的溢出和外部 T2EX 的有效信号都不再重载 TH2 和 TL2。若用户需要再次开启 T2 的方式 0，则先让 TR2=0 停止 T2，对 RCAPH2/RCAPL2 重新赋值后 (对 TH2/TL2 进行赋值无效)，让 TR2=1 再次启动 T2，T2 方可正常工作。当 T2OE=1，T2EX 端口的信号可以参与触发重载，外部触发信号与 T2 溢出的信号同一优先级。若 T2EX 有效信号和 T2 溢出信号同时来到，两者都会产生中断，且 2 个标志位的产生，就意味着会有 2 次中断请求，用户需要根据自己的应用编写不同的应用程序，以满足自己的实际需要。EX2SL 控制有效信号的类型，0 为下降沿有效，1 为上升沿有效。当通过外部 T2EX 触发重载了 TH2/TL2 并产生中断，其中断标志位是 EXTF2，而不是 TF2，用户可以通过查询 EXTF2 标志位/TF2 标志位，具体识别本次 T2 中断事件的类型。

特殊情况三：当 SIGTRG 为 0 时，定时器定时溢出后会自动重载继续定时 (TR2 不会变为 0)；当 SIGTRG 为 1 时，定时器定时溢出后会自动停止定时器 (将 TR2 变为 0)，可以利用这个方式将 PWM1，TIMER2 和 ADC 组成一个事件级联 (如：PWM1 周期触发 TIMER2 单次定时，TIMER2 定时溢出触发 ADC)



### 8.3.2 方式 1: 16 位输入捕获方式

输入捕获功能，主要用于测量脉冲宽度或周期。在 T2 的捕获方式中，时钟源由 CT2 位决定是对  $F_{sys}$  或其分频计数，还是对 T2 端口输入信号计数。T2 的 T2EX 端口的输入由 EX2EN 位控制，当在该方式下打开了 T2EX 端口后，芯片始终通过监视并捕获 T2EX 端口上的有效信号。RCAPH2 和 RCAPL2 保存最近一次捕获的结果。一旦产生了输入捕获事件，硬件自动把 TH2 和 TL2 中的当前值分别送到 RCAPH2 和 RCAPL2 中保存。输入捕获事件指，T2EX 上产生一个有效的输



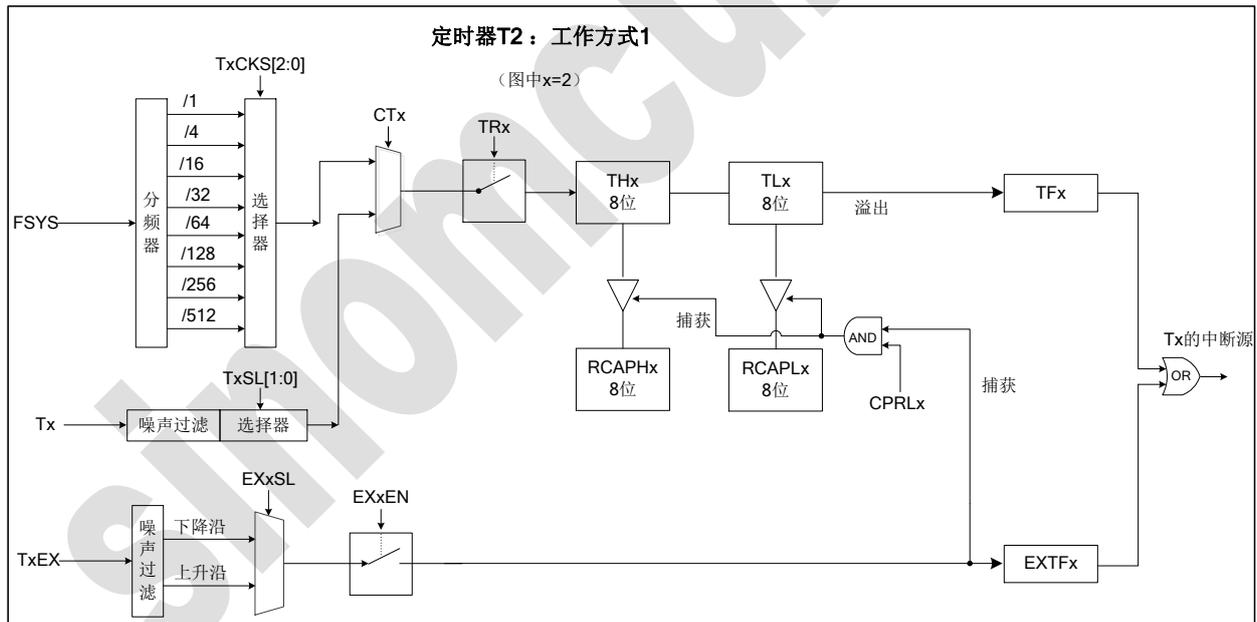
入捕获信号有效，并被芯片识别（T2 硬件对 EXTF2 标志位置 1）的过程。注意若捕获事件发生前的事件间隔超过最大计数，同样会产生 T2 定时器的溢出（TF2 标志位置 1）引起 T2 中断，但用户可以通过标志位的不同进行区分。另外，当 RCAPH2 和 RCAPL2 中保存了 TH2 和 TL2 的溢出值（0x0），也是一种异常情况，用户应该做特别处理。

标准情况：使能 T2，将 T2 设置为方式 1，设置时钟频率、定时器模式（CT2=0）等等参数。开启 T2EX 端口（EX2EN=1），并选择捕获极性为上升沿 EX2SL=1（下降沿 EX2SL=0）有效，并持续对其进行监视。启动 T2（TR2=1），TH2/TL2 从 0x0 向上开始计数。当 T2EX 端口上产生有效的触发信号（EXTF2=1）时，硬件立即将 TH2/TL2 的值保存到 RCAPH2/RCAPL2 中，同时产生 EXTF2 标志位和 T2 中断。每一次的 T2EX 有效信号都会触发一次捕获，并覆盖上一次的 RCAPH2/RCAPL2 值。

特殊情况一：用户觉得内部时钟源不好，可以改变时钟源。当 CT2=1 时，时钟源切换到外部，通过 T2 口送入。用户还可通过 T2SL 位，决定有效信号的方式，当 T2SL=01B 时上升沿有效，当 T2SL=10B 时下降沿有效，当 T2SL=11B 时双沿有效。当选择外部时钟时，由于内部系统时钟采样同步，外部最大不能超过  $F_{sys}/2$ 。

特殊情况二：假设测量 T2EX 端口上的一个低电平，但由于测量的波形时间特别长，且超过了 T2 的最大计数时间。T2 启动（TR2=1）开始计数，T2EX 端口下降沿执行第一次捕获，上升沿未到达之前，将产生 T2 溢出，发生 TF2 中断。用户可以在进入中断后通过标志位判定中断原因，并记录溢出次数，等待 T2EX 上的触发信号到来。这样，用户的中断处理函数增加一定溢出处理，便可以可以实现长时间间隔的有效沿捕获。假设，首次触发捕获后，经过了 3 个溢出时间中断后，有效的上升沿到来，那么本次测量 2 次有效沿宽度的时间计数值如下：

有效沿宽度=RCAP1(最后一次中断取得的 RCAPH2 和 RCAPL2)+0xFFFF×3（前 3 次 T2 的溢出时间）- RCAP0(首次捕获值 RCAPH2 和 RCAPL2)。



### 8.3.3 方式 3：16 位输出比较方式

在 T2 的比较模式中，时钟源仅来源于对  $F_{sys}$  或其分频，T2EX 端口作输入无效。T2 端口仅作输出使用（若用户不想输出也可以不输出），不可作为时钟输入，但 T2 端口最终是否输出，可由 T2OE 位控制，且该端口的输出初始电平值由 T2SL[1:0]位控制。在 T2 端口允许输出的前提下，每次比较事件后 T2 端口上可以指定输出一个高/低电平（CM2ON=0，初始电平由 T2SL 决定），或者控制其翻转一次电平（CM2ON=1）。

比较模式下，T2 从 0 开始计数，TH2 和 TL2 计数到与 RCMPH2 和 RCMPL2 的值相同后，发生输出比较匹配事件，硬件产生一次 TF2 置位的过程，若 LD2EN=1，则 TH2 和 TL2 重载 0x0 继续计数。若 LD2EN=0，TH2/TL2 计数至

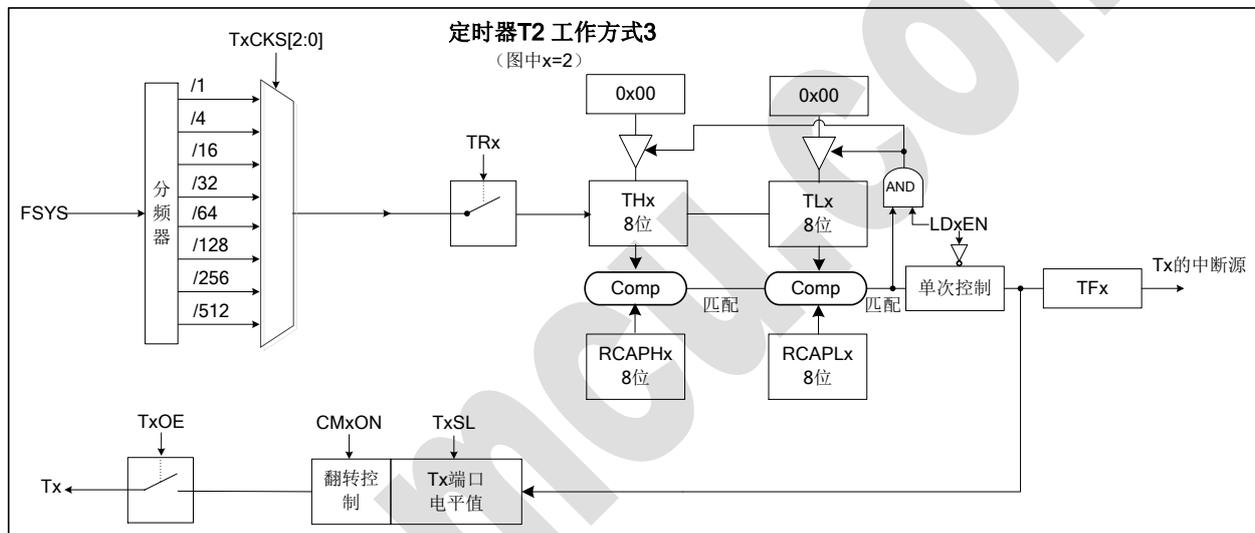


比较匹配时不自动重载。当中断打开，每次比较匹配（TF2 置位）将触发中断。输出比较功能，可用于产生类方波。

标准情况：使能 T2，将 T2 设置为方式 3，设置时钟频率、定时器模式（CT2=0）等等参数。重载有效（LD2EN=1），T2 启动（TR2=1），向上递增计数时，TH2/TL2 从 0x0 开始计到 RCMPL2/RCMPH2 中的值产生比较匹配事件，则 TF2 标志位置位（TF2=1）和产生 T2 中断，并将 0x0 载入 TH2/TL2 重新开始计数；允许外部输出 T2OE=1，且当 T2SL0=0 时，CM2ON=0，第一次比较匹配事件发生后 T2 端口上的电平由初始低电平翻转为高电平，第二次比较匹配事件后则由高电平翻转为低电平，之后交替翻转。

特殊情况：重载无效（LD2EN=0），T2 启动（TR2=1），基本同“标准情况”，每次比较匹配事件发生后，T2 端口上输出的电平翻转一次后保持不变。当 T2OE=0 时，将没有输出，T2 端口可用作其它复用功能，这样的话仅实现了一个定时中断功能。

注：初始电平的配置（T2SL0 和 CM2ON）会立即生效，T2OE 开启，对应的电平状态即会传递至 GPIO，若要实现更准确的单脉冲输出，请配置 GPIO 电平为预期脉冲电平的相反电平，T2SL0 和 CM2ON 配置脉冲初始电平，然后尽可能的同时启动 T2OE 和 TR2。



## 8.4 T2 相关寄存器

T2 寄存器汇总表

地址 (SFRn/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
SFR0/CE	T2CON1	T2 控制寄存器 1	CM2ON	CM2	-	-	-	EX2SL	SIGTRG	T2EN	00--000
SFR0/CD	TH2	T2 计数高位寄存器	TH27	TH26	TH25	TH24	TH23	TH22	TH21	TH20	00000000
SFR0/CC	TL2	T2 计数低位寄存器	TL27	TL26	TL25	TL24	TL23	TL22	TL21	TL20	00000000
SFR0/CB	RCAPH2	T2 重载/捕获高位寄存器	RCAPH27	RCAPH26	RCAPH25	RCAPH24	RCAPH23	RCAPH22	RCAPH21	RCAPH20	00000000
SFR0/CA	RCAPL2	T2 重载/捕获低位寄存器	RCAPL27	RCAPL26	RCAPL25	RCAPL24	RCAPL23	RCAPL22	RCAPL21	RCAPL20	00000000



SFR0/C 9	T2MOD	工作模式寄存器	LD2EN	T2CK2	T2CK1	T2CK0	T2OE	T2SL1	T2SL0	-	0000 000-
SFR0/C 8	T2CON	T2 控制寄存器	TF2	EXTF2	-	-	EX2EN	TR2	CT2	CPRL2	00-- 0000

#### 8.4.1 T2 控制寄存器 (T2CON, 0xC8/SFR0)

SBIT 寄存器	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T2CON	TF2	EXTF2	-	-	EX2EN	TR2	CT2	CPRL2
R/W	R/W	R/W	-	-	R/W	R/W	R/W	R/W
复位值	0	0	-	-	0	0	0	0

BIT[7] TF2 – 定时器/计数器 T2 溢出标志位

0: T2 无溢出, 中断响应后仅可由软件清 0;

1: T2 溢出时由硬件置 1。

BIT[6] EXTF2 – 定时器/计数器 T2 重载触发/输入捕获标志位

0: T2EX 上无外部信号输入/对外输出信号, 必须由软件清 0;

1: T2EX 上检测到有效的外部信号输入/产生有效的对外输出信号, T2 做重载触发/输入捕获的同时, 由硬件置 1。

注: 当 T2 工作在方式 2 时, EXTF2 恒为 0, 软件写 1 无效。

BIT[3] EX2EN – T2EX 端口作为重载触发/输入捕获使能位

0: 禁止 T2EX 端口作为重载触发/输入捕获功能, 或者用于通用 IO 或其他复用模块 (由 T2CON 的 C/T2 位决定);

1: 使能 T2EX 端口作为重载触发/输入捕获功能, 具体是重载触发还是输入捕获功能, 由此时 T2 的具体工作方式决定 (工作方式 0=重载触发; 工作方式 1=输入捕获), 激活 T2 执行一次重载触发/输入捕获的事件。

注: 有效触发的形式, 由 EX2SL 位决定, 即 EX2SL=0 为下降沿触发, EX2SL=1 为上升沿触发。

BIT[2] TR2 – 定时器/计数器 T2 使能位

0: 停止定时器/计数器 T2;

1: 开始定时器/计数器 T2, 或者再次写入 1 后复位 T2, 让 T2 重新开始计数, 即定时器将从 0x0/0xFFFF 开始向上/向下计数, 具体计数方向视 DC2EN 位而定。

BIT[1] CT2 – T2 的计数器/定时器方式选择位

0: T2 工作于定时器方式, T2 的时钟源来自内部;

1: T2 工作于计数器方式, T2 端口作为 T2 的外部时钟源专用输入口, 且不能做其它复用功能。

BIT[0] CPRL2 – 定时器/计数器 T2 重载触发/输入捕获方式选择位

0: T2 工作于带重载触发功能的定时器/计数器方式;

1: T2 工作于带输入捕获功能的定时器/计数器方式。

#### T2 工作方式整理表

T2 工作方式	T2CON1	T2CON
	Bit 6	Bit 0
	CM2	CPRL2
方式 0: 16 位重载方式	0	0



方式 1: 16 位捕获方式	0	1
方式 3: 16 位比较方式	1	0
无效状态	其他值	

注: 当 CM2、BR2EN、CPRL2 同时置位时, 比较模式优先于波特率模式优先于捕获模式。

#### 8.4.2 T2 工作模式寄存器 (T2MOD, 0xC9/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T2MOD	LD2EN	T2CKS2	T2CKS1	T2CKS0	T2OE	T2SL1	T2SL0	-
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	-
复位值	0	0	0	0	0	0	0	0

BIT[7] LD2EN - 定时器/计数器 T2 溢出重载控制位

0: 在重载模式 (方式 0/方式 1/方式 3) 中, T2 溢出后不自动重载 TH2/TL2;

1: 在重载模式 (方式 0/方式 1/方式 3) 中, T2 溢出后自动重载 TH2/TL2, 使能后将连续重载。

BIT[6:4] T2CKS[2:0] - 定时器/计数器 T2 时钟分频选择位

T2CKS[2:0]	T2 频率 (CT2M=0)
000	Fsys
001	Fsys/4
010	Fsys/16
011	Fsys/32
100	Fsys/64
101	Fsys/128
110	Fsys/256
111	Fsys/512

BIT[3] T2OE - 定时器/计数器 T2 输出使能位

0: 禁止 T2 溢出时翻转 T2 端口电平, 端口用于通用 IO 或其他复用模块;

1: 使能 T2 端口复用于定时器 T2, T2 溢出时翻转一次端口电平, 初始电平可由 T2SL[1:0]控制。

BIT[2:1] T2SL[1:0] - T2 端口输入/输出信号选择位

T2SL[1:0]	T2 端口有效输入信号选择 <sup>注1</sup>	T2 端口初始输出信号选择 <sup>注2</sup>
00	无效	禁止
01	上升沿有效	高电平
10	下降沿有效	低电平
11	双边沿同时有效	禁止

注 1: 仅当 C/T2 位置 1 后, T2 端口上的输入信号才受控于该控制位。

注 2: 仅在输出比较的情况下, 即“工作方式 3”, 且 T2OE=1 时, 该位决定了 T2 端口的初始电平值, 若输出比较模式中, 用户需要了解当前 T2 端口的实际电平值, 可以通过查询对应的 (P2.0) 端口的数据值得。

#### 8.4.3 T2 控制寄存器 1 (T2CON1, 0xCE/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T2CON1	CM2ON	CM2	-	-	-	EX2SL	SIGTRG	T2EN
R/W	R/W	R/W	-	-	-	R/W	R/W	R/W
复位值	0	0	-	-	-	0	0	0



- BIT[7] CM2ON – 比较方式下 T2 端口输出电平翻转控制位  
 0: T2 端口在比较模式（方式 3）下，每次触发后 T2 端口自动翻转一次输出电平；  
 1: T2 端口在比较模式（方式 3）下，每次触发后 T2 端口输出（初始）电平保持不变。

注：T2 端口上的初始电平由 T2SL 决定。

- BIT[6] CM2 – 比较方式选择位  
 0: T2 工作于（方式 0、方式 1）非 16 位输出比较模式；  
 1: T2 工作于（方式 3）16 位输出比较模式。
- BIT[2] EX2SL – T2EX 端口输入捕获/重载触发信号有效沿/电平选择位  
 0: T2EX 端口作为输入捕获或者重载触发用时，下降沿为有效输入信号；  
 1: T2EX 端口作为输入捕获或者重载触发用时，上升沿为有效输入信号。

EX2SL	(当 EX2EN=1) T2EX 端口有效输入信号选择
1	上升沿有效
0	下降沿有效

- BIT[1] SIGTRG – Timer2 单次计数使能位  
 0: 连续计数模式；  
 1: 单次计数模式。
- BIT[0] T2EN – 定时器/计数器 T2 模块使能位  
 0: 关闭 T2 模块；  
 1: 打开 T2 模块。

#### 8.4.4 T2 计数高位寄存器 (TH2, 0xCD/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TH2	TH27	TH26	TH25	TH24	TH23	TH22	TH21	TH20
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

- BIT[7:0] TH2[7:0] – 定时器/计数器 T2 高 8 位计数器设定值

#### 8.4.5 T2 计数低位寄存器 (TL2, 0xCC/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TL2	TL27	TL26	TL25	TL24	TL23	TL22	TL21	TL20
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

- BIT[7:0] TL2[7:0] – 定时器/计数器 T2 低 8 位计数器设定值

注：无论是读或者写 TL2/TH2 的值，必须先停止 (TR2=0)，否则 T2 将产生不可预知的情况。

#### 8.4.6 T2 重载/捕获高位寄存器 (RCAPH2, 0xCB/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RCAPH2	RCAPH27	RCAPH26	RCAPH25	RCAPH24	RCAPH23	RCAPH22	RCAPH21	RCAPH20
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

- BIT[7:0] RCAPH2[7:0] – 定时器/计数器 T2 重载/捕获高 8 位寄存器

#### 8.4.7 T2 重载/捕获低位寄存器 (RCAPL2, 0xCA/SFR0)



	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RCAPL2	RCAPL27	RCAPL26	RCAPL25	RCAPL24	RCAPL23	RCAPL22	RCAPL21	RCAPL20
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] RCAPL2[7:0] – 定时器/计数器 T2 重载/捕获低 8 位寄存器

注：无论是读或者写 RCAPL2/RCAPH2 的值，必须先停止（TR2=0），否则 T2 将产生不可预知的情况。

## 8.5 定时器 T3

芯片包含 1 个 16 位基础定时器 T3，T3 是一个向上递增型定时器/计数器，可选择系统时钟（同步时钟）或低频时钟 FRTC（异步时钟）。

### 8.5.1 方式 0：系统时钟（同步时钟）模式

当 T3 作定时使用时，可通过 CT3 位 (=0) 设定，T3 选择系统时钟（同步时钟）模式。此时，其时钟源为系统时钟或其分频，通过 T3CKS 设置，可选择多种分频（具体值参考 T3 控制寄存器说明）；启动 TR3，T3 从 TL3/TH3 设定的初值开始递增计数，溢出后产生 T3 中断，并重新加载 TL3/TH3 设定的初值重新开始递增计数。

### 8.5.2 方式 1：低频时钟（异步时钟）模式

当 T3 作定时使用时，可通过 CT3 位 (=1) 设定，T3 选择低频时钟（异步时钟）模式。此时，通过寄存器位 FRTC SW 设定其低频时钟 FRTC 为外部 32768Hz 振荡器或其分频，还是选择内部 32KHz 低频振荡器或其分频，通过 T3CKS 设置，可选择多种分频（具体值参考 T3 控制寄存器说明）。方式 1 类似于方式 0，它们的本质类似，只是时钟的来源不同。

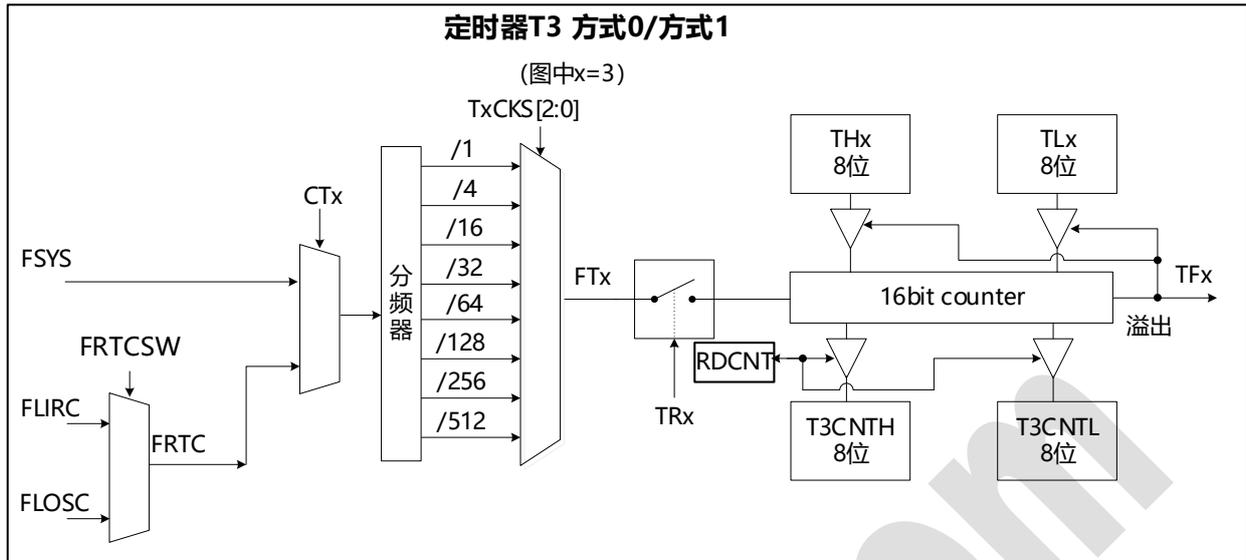
通过 CLKCR1 寄存器位 FRTC SW，可以选择内置 32KHz 低频振荡器或外部 32768Hz 晶体振荡器作为时钟源，通过设定 LIRCEN\_LP 位确定 T3 是否在停止模式（STOP）/休眠模式（SLEEP）下具备唤醒功能。

当 LIRCEN\_LP=1，停止模式/休眠模式下选择的低频时钟源（FIRC 或 FLOSC）继续工作，当寄存器 TH3/TL3 溢出后会产生 T3 中断，并通过中断从而实现在停止模式/休眠模式下唤醒芯片的功能。

当 LIRCEN\_LP=0，停止模式下低频时钟源（FLIRC 或 FLOSC）将继续工作，则 T3 在停止模式具备唤醒芯片的功能；休眠模式下 FLIRC 会被强制停止，此时 T3 不具备唤醒芯片的功能；休眠模式下 FLOSC 如若处于使能状态，此时 T3 具备唤醒芯片的功能。

若选择外部 32768Hz 振荡器作为时钟源时（需通过 CLKCR1 寄存器位 FRTC SW 设置相关值），需连接外部 32768Hz 振荡器到对应的 OSCI/OSCO 端口，这种模式下可以实现对时间的精准计时。

用户应充分考虑在停止模式/休眠模式下是否需要使用 T3 作为唤醒源，并做相应的设定，否则无论是内置 32KHz 还是外部 32768Hz 振荡器，始终“开启”为 T3 提供时钟源的情况下，都会增加芯片的功耗。



### 8.5.3 实时读取计数器值

T3 定时器内部时钟涉及异步设计，若用户需要实时读取动态计数器值，必须按照下面步骤：

- 1、先写 RDCNT=1，启动 T3CNT 读取同步；
- 2、等待 RDCNT 变为 0，同步完成后，硬件会清零 RDCNT (RDCNT=0)；
- 3、此时用户可以读取 T3CNT 的值 (T3CNT[H:L])。

## 8.6 T3 相关寄存器

T3 寄存器汇总表

地址 (SFR n/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
SFR0 /97	TH3	T3 重载高位寄存器	TH37	TH36	TH35	TH34	TH33	TH32	TH31	TH30	0000 000
SFR0 /96	TL3	T3 重载低位寄存器	TL37	TL36	TL35	TL34	TL33	TL32	TL31	TL30	0000 000
SFR0 /95	T3CR	T3 控制寄存器	TR3	CT3	T3F	RDCNT	T3EN	T3CK2	T3CK1	T3CK0	0000 0000
SFR0 /85	T3CNTH	T3 计数高位寄存器	CNT15	CNT14	CNT13	CNT12	CNT11	CNT10	CNT9	CNT8	0000 0000
SFR0 /84	T3CNTL	T3 计数低位寄存器	CNT7	CNT6	CNT5	CNT4	CNT3	CNT2	CNT1	CNT0	0000 0000

### 8.6.1 T3 控制寄存器 (T3CR, 0x95/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T3CR	TR3	CT3	T3F	RDCNT	T3EN	T3CK2	T3CK1	T3CK0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7] TR3 - 定时器/计数器 T3 使能位



- 0: 停止定时器/计数器 T3;  
1: 打开定时器/计数器 T3, 每次写入 1, 将从 TH3 和 TL3 的起始值开始递增计数;
- BIT[6] CT3 - T3 模式选择位  
0: 时钟源为系统时钟或其分频;  
1: 时钟源为低频时钟 FRTC (内部 32KHz 振荡器或者外部 32768Hz 振荡器或其分频), 具体由 CLKCR1 寄存器位 FRTCSW 决定。
- 注 2: 当该位=1 时, 若外部没有 32768Hz 振荡器, (LIRCEN\_LP=0) 32KHz 内部振荡器也停止时, T3 将停止工作。
- BIT[5] T3F - 定时器/计数器 T3 溢出标志位  
0: 定时器/计数器 T3 无溢出, 中断响应后硬件清 0, 也可由软件清 0;  
1: 定时器/计数器 T3 溢出时, 由硬件置 1。
- BIT[4] RDCNT - T3 计数器读取同步控制  
0: 此位置 1 后清零, 表示 T3CNT 已完成读取同步, 用户可以读取 T3CNT 寄存器的值; 写 0 无效  
1: 软件写 1, 启动 T3CNT 读同步, 同步完成后硬件清零。
- BIT[3] T3EN - 定时器/计数器 T3 模块使能位  
0: 关闭定时器/计数器 T3 模块;  
1: 打开定时器/计数器 T3 模块。
- BIT[2:0] T3CKS[2:0] - T3 基础定时器时钟分频选择位  
对应频率选择列表:

T3CKS[2:0]	分频
000	Fsys
001	Fsys/4
010	Fsys/16
011	Fsys/32
100	Fsys/64
101	Fsys/128
110	Fsys/256
111	Fsys/512

### 8.6.2 T3 重载高位寄存器 (TH3, 0x97/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TH3	TH37	TH36	TH35	TH34	TH33	TH32	TH31	TH30
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] TH3[7:0] - 定时器/计数器 T3 高 8 位计数器设定重载值

### 8.6.3 T3 重载低位寄存器 (TL3, 0x96/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TL3	TL37	TL36	TL35	TL34	TL33	TL32	TL31	TL30
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0



BIT[7:0] TL3[7:0] – 定时器/计数器 T3 低 8 位计数器设定重载值

注：写操作时：必须先写 TL3，再写 TH3；否则 TL3 的值不能立即反映到内部 counter 上；读操作时：必须先读 TL3，再读 TH3，否则将读取到错误的值。

#### 8.6.4 T3 计数高位寄存器 (T3CNTH, 0xE7/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T3CNTH	CNT15	CNT14	CNT13	CNT12	CNT11	CNT10	CNT9	CNT8
R/W	R	R	R	R	R	R	R	R
复位值	0	0	0	0	0	0	0	0

BIT[7:0] CNT[15:8] – 定时器/计数器 T3 高 8 位计数器值，只读

#### 8.6.5 T3 计数低位寄存器 (T3CNTL, 0xE6/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T3CNTL	CNT7	CNT6	CNT5	CNT4	CNT3	CNT2	CNT1	CNT0
R/W	R	R	R	R	R	R	R	R
复位值	0	0	0	0	0	0	0	0

BIT[7:0] CNT[7:0] – 定时器/计数器 T3 低 8 位计数器值，只读

注 1: T3 运行中需要读计数器时，需先置位 RDCNT (RDCNT)，待 RDCNT 硬件清 0 后，读取 T3CNT[H:L]的值，否则读取到错误的值。



## 9 通用型 PWM0

### 9.1 PWM0 概述

片上提供 1 个 8 位通用型 PWM 模块 PWM0，可设置独立中断向量入口，可输出 2 路共周期独立占空比的 PWM (PWM0A)，输出极性可选，输出引脚为 PWM0A。该 PWM 也可用作具有重载功能的通用定时器使用。

PWM0 周期的计算公式：

$$\text{PWM0周期} = \frac{\text{PWM0P} + 1}{F_{\text{sys}}}$$

注：PWM0 周期的时钟设定，固定为  $F_{\text{SYS}}$  或  $F_{\text{SYS}}$  的分频，最大不应该超过 16MHz（待实测）。

PWM0x (x=A) 占空比（默认为高电平的时间）的计算公式：

$$\text{PWM0x占空比} = \frac{\text{PWM0xD} + 1}{\text{PWM0P} + 1}$$

注 1：假设为（默认）高电平有效，每个周期之间存在一个 clock（由 1 到 0）切换的时间；当  $\text{PWM0xD} < \text{PWM0P}$ ，占空比为  $\text{PWM0xD} + 1$  的高电平时间；当  $\text{PWM0xD} \geq \text{PWM0P}$  时，占空比为最大 100%，输出恒为 1。

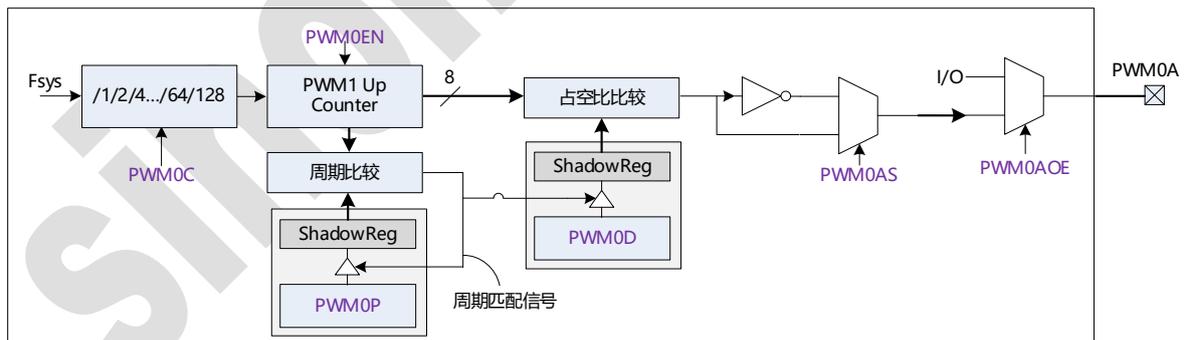
注 2：在 PWM0x 输出波形时，若当前周期或占空比被修改，不会立即生效，需要等到下个周期来到才会相应改变。

注 3：占空比所代表的电平，由 PWM0xS 位决定，用户可以切换。当 PWM0xS 设为 1 时，这里的 PWM0x 占空比指“低电平”所占的时间。

注 4：当用户需要 PWM0x 输出恒 0/1 时，推荐下面两种做法：A. 在 PWM 功能下反向控制极限位，并以 100% 占空比输出，此时将获得需要的电平值；B. 给 Px 端口指定一个 0/1 电平，并通过改变 PWM0xOE 位退出 PWM 功能，切换至 GPIO 的输出功能，直接输出指定的电平值。

### 9.2 PWM0x 结构图

PWM0A 结构示意图



### 9.3 PWM0 相关寄存器

PWM0 寄存器汇总表

地址 (SFRn/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
XSFR/	PWM0CR0	PWM0 控制寄	PWM0EN	PWM0PIE	PWM0IF	PWM0AO	PWM0AS	PWM0C2	PWM0C1	PWM0C0	0000



FF68		寄存器				E					0000
XSFR/ FF69	PWM0P	PWM0 周期设置寄存器	PWM0P7	PWM0P6	PWM0P5	PWM0P4	PWM0P3	PWM0P2	PWM0P1	PWM0P0	0000 0000
XSFR/ FF6A	PWM0AD	PWM0A 占空比设置寄存器	PWM0AD 7	PWM0AD 6	PWM0AD 5	PWM0AD 4	PWM0AD 3	PWM0AD 2	PWM0AD 1	PWM0AD 0	0000 0000

### 9.3.1 PWM0 控制寄存器 0 (PWM0CR0, 0xFF68/XSFR)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM0CR0	PWM0EN	PWM0PIE	PWM0IF	PWM0AOE	PWM0AS	PWM0C2	PWM0C1	PWM0C0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7] PWM0EN – PWM0 使能位

0: 关闭 PWM0 模块，关闭后 PWM 停止计数，输出也将停止；

1: 打开 PWM0 模块，PWM0 计数器从 0 开始向上计数，PWM0 对应的占空比较器不断比较 PWM0D 与 PWM0CNT 的值，当两者相同时反转一次输出波形，比较 PWM0P 与 PWM0CNT 的值，当周期结束时再反转一次，并继续新一轮周期的比较反转。

注 1: PWM0EN 即使计数的控制位，配置 PWM0 时须先配置周期寄存器和占空比寄存器，再设置 PWM0EN 位为 1。

注 2: PWM0CNT 是一个内部寄存器，对用户不可见，向上自增，将在 PWM0EN 启动后开始计数或者重新计数。

BIT[6] PWM0PIE – PWM0 周期中断使能

0: 关闭；

1: 开启。

注: 此控制位与 IEN1 的 PWMIE 联动控制，若要使能 PWM0P 中断，必须同时置位 PWM0PIE 和 PWMIE；若要禁止 PWM0P 中断，仅需清零 PWMIE 和 PWM0PIE 任意一位。PWM0P 复用同一个中断入口。

BIT[5] PWM0IF – PWM0 中断标志位

0: PWM0 定时器未溢出（计数未超过周期寄存器值），由软件清 0；

1: PWM0 定时器已溢出（计数超过周期寄存器值），由硬件置 1。

BIT[4] PWM0AOE – PWM0A 输出使能位

0: 禁止 PWM0A 复用功能，端口用于通用 IO 或其他复用模块；

1: 打开 PWM0A 复用功能，作为 PWM0A 输出波形用。

注 1: PWM0EN 设置为 0 时，硬件会同时关闭 PWM0AOE，需要再次开启 PWM，需要重新配置 PWM0EN=1 和 PWM0AOE=1。

注 2: 当禁止 PWM0A 复用功能时，PWM0 可用作为“具有重载功能的通用定时器”使用。

BIT[3] PWM0AS – PWM0A 输出极性选择位

0: PWM0A 输出占空比有效期间为高电平；

1: PWM0A 输出占空比有效期间为低电平。

BIT[2:0] PWM0C[2:0] – PWM0 工作时钟频率选择位

PWM0C[2:0]	PWM0 工作频率
000	Fsys
001	Fsys/2
010	Fsys/4



011	Fsys/8
100	Fsys/16
101	Fsys/32
110	Fsys/64
111	Fsys/128

注：在 PWM0 输出波形时，请勿随意修改频率，否则会影响 PWM0 的输出结果；建议按先关闭 PWM0，修改后再打开 PWM0 的顺序操作。

### 9.3.2 PWM0 周期设置寄存器 (PWM0P, 0xFF69/XSFR)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM0P	PWM0P7	PWM0P6	PWM0P5	PWM0P4	PWM0P3	PWM0P2	PWM0P1	PWM0P0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] PWM0P[7:0] – 设置 PWM0 的周期。

### 9.3.3 PWM0A 占空比设置寄存器 (PWM0AD, 0xFF6A/XSFR)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM0AD	PWM0AD7	PWM0AD6	PWM0AD5	PWM0AD4	PWM0AD3	PWM0AD2	PWM0AD1	PWM0AD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] PWM0AD [7:0] – 设置 PWM0A 的占空比，即 PWM0 周期内 PWM0A 先输出有效电平的时钟个数。

注：在 PWM0A 配置时，建议先设置周期和占空比寄存器，再配置控制寄存器。



## 10 增强型 PWM1 阵列

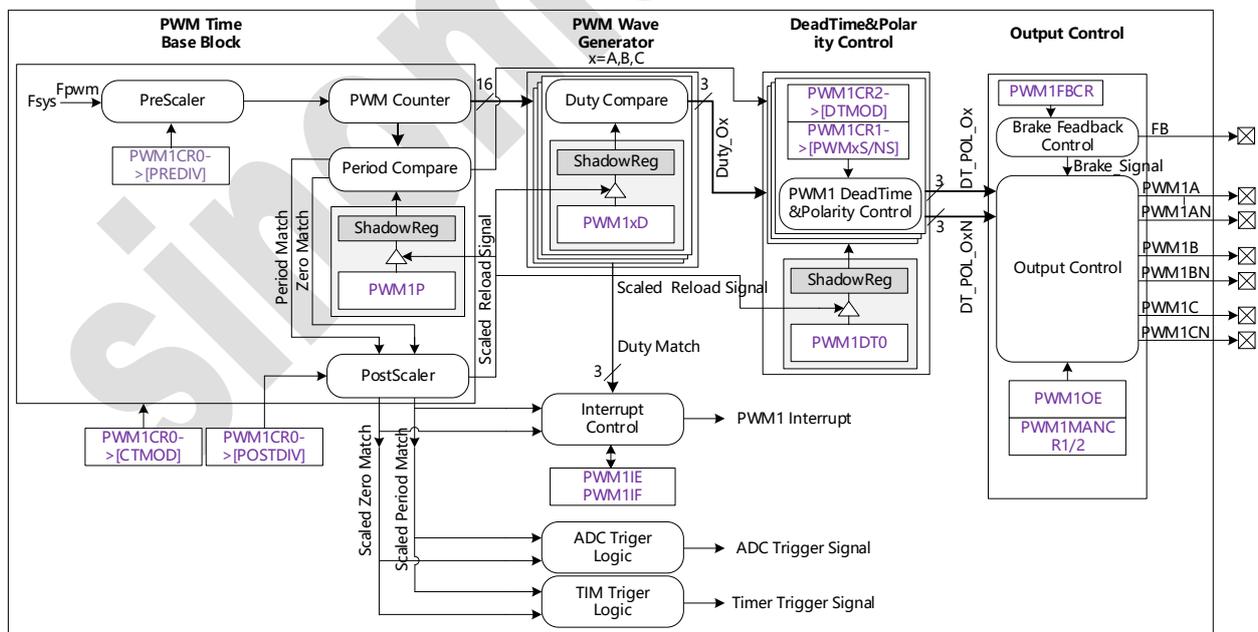
### 10.1 概述

- ◇ 支持电机控制专用
- ◇ 重要寄存器受保护寄存器 PWMRLN 控制
- ◇ 12 位时基计数器
- ◇ 三种时基计数模式：边沿对齐计数、中心对齐计数和单次计数模式
- ◇ 两种波形输出模式：互补模式与独立模式
- ◇ 提供 PWM 周期匹配、归零匹配和占空比比较匹配中断
- ◇ 3 对互补 PWM 输出或 3 路独立 PWM 输出，输出极性可选择
- ◇ 提供故障检测功能可紧急关闭 PWM 输出
- ◇ 提供寄存器重载使能位以保证寄存器重载同步
- ◇ 4 档预分频/后分频功能
- ◇ 可切换手动控制 PWM 输出
- ◇ 可选择自动触发 Timer 或 ADC 转换

芯片内建一个 12 位电机控制 PWM 阵列。PWM 模块由时基模块（PWM Time Base Block）、波形发生模块（Wave Generator）、死区&极性控制模块（Dead&Polarity Control）、刹车检测（Brake Feedback Control）以及输出控制模块（Output Control）组成。

时基模块中通过一个 16 位计数器（PWM Counter）对 PWM 时钟（PWM Clock）计数，计数器的值与周期寄存器比较产生周期匹配信号、与 0 比较产生归零信号、与占空比寄存器比较产生占空比匹配信号，结合波形发生模块产生原始的六路 PWM 信号 Duty\_Ox 和 Duty\_OxN（x=A,B,C），该原始 PWM 信号经过死区&极性控制模块后产生带死区和极性的 PWM 信号 DT\_POL\_Ox 和 DT\_POL\_OxN（x= A,B,C），最后的输出控制模块和刹车检测模块决定是否将 DT\_POL\_Ox 和 DT\_POL\_OxN（x= A,B,C）信号输出至 PWM 引脚 PWM1x/PWM1xN（x=A,B,C）上。

PWM 模块方框图



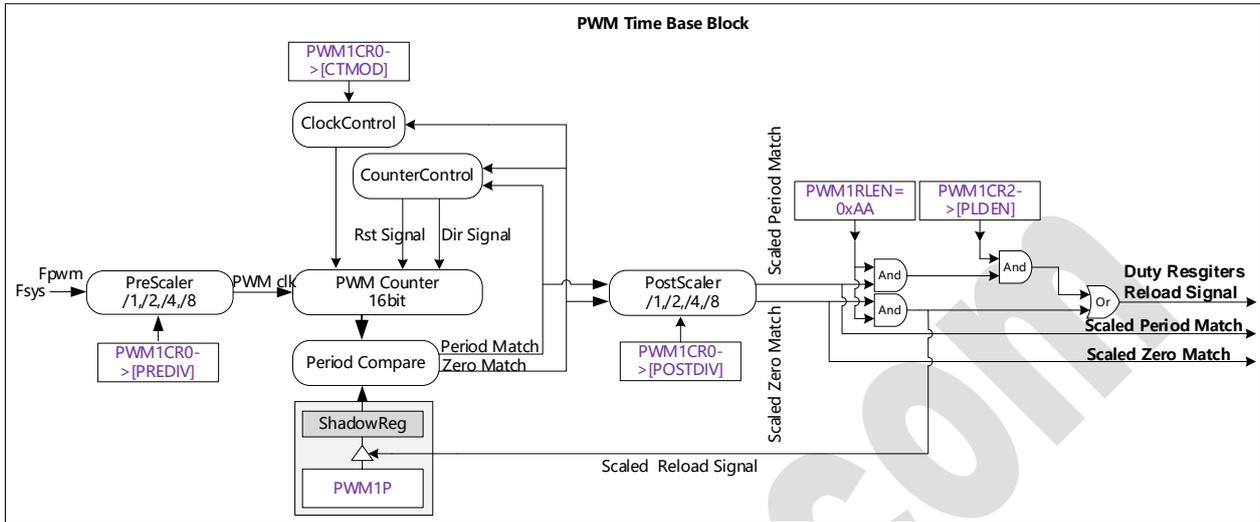
### 10.2 PWM 时基模块

PWM 模块中的时基模块（Time Base Block）主要由一个 16 位的计数器结合一个预分频器和一个后分频器组成。



如图所示，波形发生器（Wave Generator）用时基模块中 16 位计数器的值与占空比寄存器比较并结合死区/极性逻辑（DeadTime&Polarity Control）和输出控制逻辑（Output Control）最终产生 PWM 波形。

时基模块逻辑框图



### 10.2.1 PWM 工作时钟

PWM1 时钟源为系统时钟  $F_{sys}$  ( $F_{pwm}$ ) 经过预分频提供给 PWM 时基计数器。PWMCR0 寄存器的 PREDIV 控制位控制分频系数 1/1、1/2、1/4 或 1/8 分频。

通过软件置位/清零 PWM1OE 寄存器中的 PWM1EN 位可以使能/禁止 PWM 时基模块。PWM1EN 位从 0 置 1 时，PWM 计数器（PWM Counter）开始对 PWM 时钟从 0 开始向上计数。PWMEEN 位清 0 时，预分频计数器和 PWM 计数器都将清 0。

注：任何复位都将导致 PWM1 时钟预分频计数器和 PWM 计数器清零。

### 10.2.2 PWM 时基工作方式

PWM1 时基中 12 位的 PWM 计数器有 3 种工作方式，由 PWMCR0 寄存器中的 CTMOD[1:0]控制位确定。

#### 边沿对齐计数模式

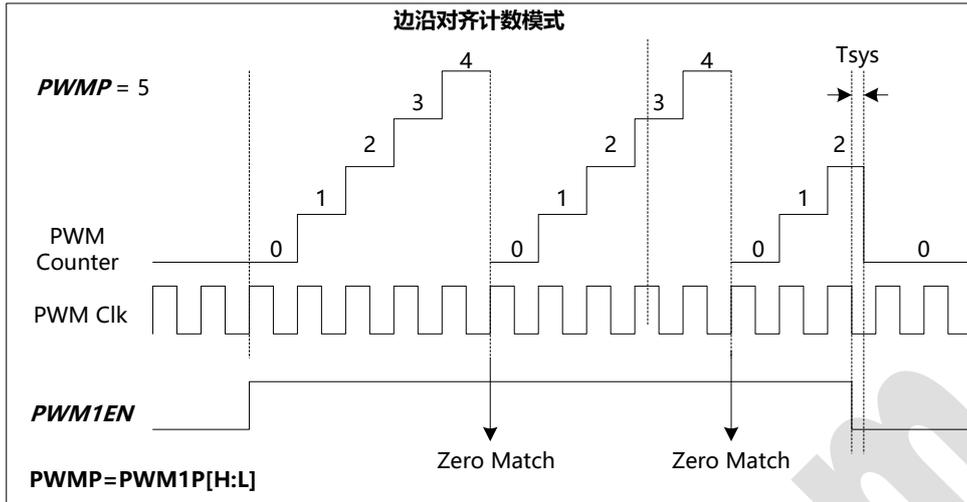
设置 PWMCR0 中 CTMOD[1:0]为 00，时基模块中的 PWM 计数器将工作在边沿对齐计数模式下。

此模式下，设置 PWM1EN 位置 1，PWM 计数器将从 0 开始向上计数，直到与 PWM 周期寄存器 PWM1PL/H 匹配，然后 PWM 计数器复位为 0 并继续向上计数，如此往复。

PWM 计数器复位为 0 的时刻产生归零匹配（Zero Match）。

设置 PWM1EN 位清 0，PWM Counter 将停止计数并在下一个系统时钟边沿复位为 0。

边沿对齐计数模式时序图（PWM1PL/H = 5）



### 中心对齐计数模式

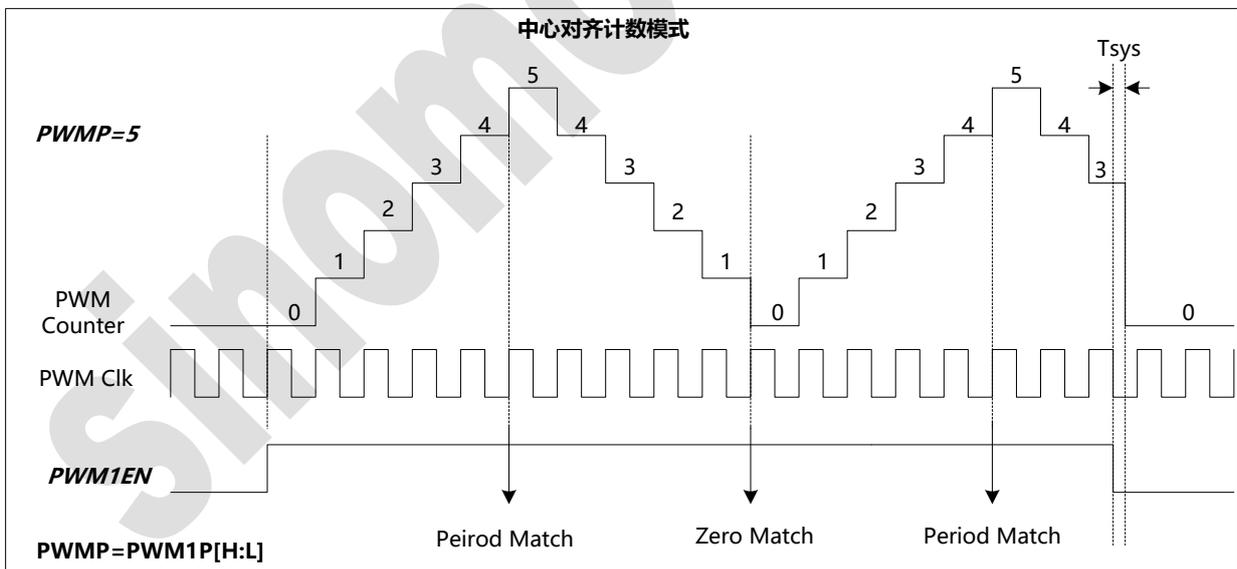
设置 PWMCR0 中 CTMOD[1:0]为 01，时基模块中的 PWM 计数器将工作在在中心对齐计数模式下。

此模式下，设置 PWM1EN 位置 1，PWM 计数器将从 0 开始向上计数，直到与 PWM 周期寄存器 PWM1PL/H 匹配，然后 PWM 计数器开始向下计数直至归 0，如此往复。

PWM 计数器向下计数至零的时刻产生归零匹配（Zero Match），PWM 计数器与周期寄存器匹配时刻产生周期匹配（Period Match）。

设置 PWM1EN 位清 0，PWM Counter 将停止计数并在下一个系统时钟边沿复位为 0。

中心对齐计数模式时序图（PWM1PL/H = 5）



### 单次计数模式

设置 PWM1CR0 中 CTMOD[1:0]为 10 或 11，时基模块中的 PWM 计数器将工作在单次计数模式下。

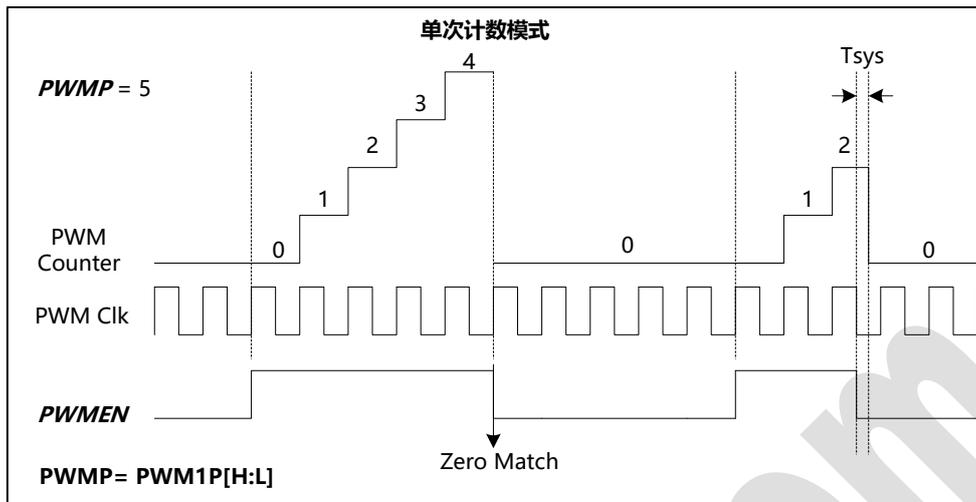
此模式下，设置 PWM1EN 位置 1，PWM 计数器将从 0 开始向上计数，直到与 PWM 周期寄存器 PWM1PL/H 匹配时，PWM 计数器复位为 0，同时 PWM1EN 位将由硬件清 0，PWM1 输出高阻状态。

PWM 计数器复位为 0 的时刻产生归零匹配（Zero Match）。

设置 PWM1EN 位清 0，PWM Counter 将停止计数并在下一个系统时钟边沿复位为 0。



单次计数模式时序图 (PWM1PL/H = 5)



### 10.2.3 后分频

PWM 计数器运行时产生的周期匹配信号 (Period Match) 和归零匹配信号 (Zero Match) 经过后分频计数器 (PostScaler), 可以实现 1/1、1/2、1/4 和 1/8 分频。通过寄存器 PWM1CR0 中的 POSTDIV[1:0]位可以选择以上四档后分频。若设置 POSTDIV[1:0] = 00 则后分频系数为 1/1, 周期匹配信号和归零匹配信号不会被分频。

后分频应用场景:

- 1、通过后分频配置, 可以实现多个周期更新一次占空比, 不需要每个周期更新 PWM 的占空比
- 2、后分频后的周期匹配信号 (Scaled Period Match) 和归零匹配信号 (Scaled Zero Match) 结合 PWM1RLEN 寄存器值用于控制周期寄存器 PWM1PL/H 以及占空比寄存器 PWM1xD[H,L]/PWM1xND[H,L](x=A,B,C)的重载。
- 3、后分频后的周期匹配信号 (Scaled Period Match) 和归零匹配信号 (Scaled Zero Match) 可用于产生归零匹配中断和周期匹配中断, 详见“时基中断”章节。
- 4、后分频后的周期匹配信号 (Scaled Period Match) 和归零匹配信号 (Scaled Zero Match) 可用于触发 ADC, Timer2。在中心对齐模式下触发 timer2, 可用分频后的周期匹配信号触发、或分频后的归零匹配信号触发, 也可以同时选择两者。中心对齐模式触发 ADC, 可以选择在分频后的周期匹配信号触发, 或分频后的归零匹配信号触发, 也可以同时选择两者。而在边沿对齐模式下, 只能用分频后的归零匹配信号触发。

*注: 后分频只对周期匹配信号和归零匹配信号有效, 对占空比匹配时刻无效, 比如, 允许占空比中断和归零匹配信号中断的情况下, 设置后分频为 8 分频时, 归零匹配信号进中断会在 8 个 PWM 周期后才进一次中断, 而占空比匹配信号则依然会在每个 PWM 周期触发中断。*

### 10.2.4 时基中断

时基中断包括分频后的周期匹配信号所触发的周期中断和分频后的归零匹配信号所触发的归零中断。

分频后的周期匹配信号发生时将令寄存器 PWM1IF 中的 PWMPIF 位置 1, 此时若 PWM1IE 中的 PWMPIE 位为 1 则将触发周期中断;

分频后的归零匹配信号发生时将令寄存器 PWM1IF 中的 PWMZIF 位置 1, 此时若 PWM1IE 中的 PWMZIE 位为 1 则将触发归零中断。PWMPIF 位和 PWMZIF 位需由软件写 0 清除。

PWM1CR0 中 POSTDIV[1:0]段等于 0 时, PWM 计数器的周期匹配信号和归零匹配信号不会被分频, 故周期匹配信号每次发生时都将令 PWMPIF 位置 1, 归零匹配信号每次发生时都将令 PWMZIF 位置 1。



### 10.2.5 PWM 周期寄存器的自动重载

周期寄存器 PWM1PL 和 PWM1PH 都有一个缓存寄存器 (ShadowReg)，用户能够读写 PWM1PL/H 但不能操作其缓存寄存器。PWM 时基计数器运行时实际上是与周期寄存器的 ShadowReg 进行比较以产生周期匹配信号。

由于存在缓冲寄存器，因此对周期寄存器的读写有特殊规定：

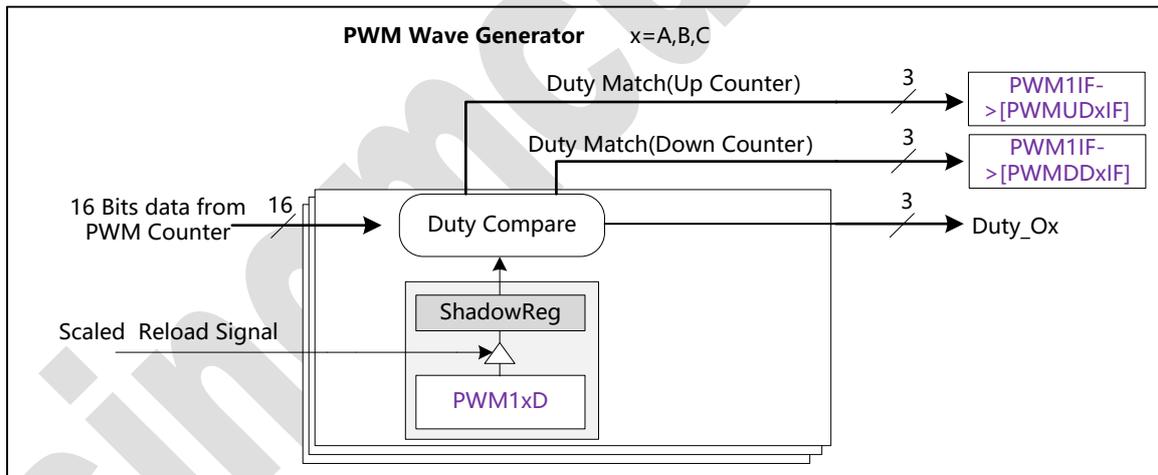
当 PWM1RLEN 寄存器的值设为 0x55 时，才允许对这些寄存器进行修改，然后只有当 PWM1RLEN 寄存器中的值设为 0xAA 时，分频后的归零匹配信号发生时才会将 PWM1PL 和 PWM1PH 寄存器锁存至对应的缓存寄存器 (ShadowReg) 中，这样可以避免寄存器修改过程中发生重载，也避免寄存器修改立即生效可能会出现毛刺。因此对上述寄存器的修改，必须要有对 PWM1RLEN 寄存器的配合操作才能完成。

*注：为适应 keil 中 C 语言 16 位寄存器赋值顺序，上述 2 个 8 位寄存器赋值必须先赋值高位寄存器 PWM1PH 再赋值低位寄存器 PWM1PL 才能正确赋值成功；如果顺序相反只能赋值 PWM1PL 寄存器，PWM1PH 为之前旧值。*

## 10.3 PWM 波形发生模块

PWM 波形发生模块使用 PWM 时基模块中 PWM 计数器的值与 3 个 16 位占空比缓存寄存器的值进行比较，从而产生原始的 3 路 PWM 波形：Duty\_Ox (x=A,B,C)。该原始 3 路 PWM 波形信号经过死区&极性逻辑和输出控制逻辑最终在 6 个 PWM 引脚输出。

波形发生模块逻辑框图



### 10.3.1 PWM 输出模式

PWM 模块中包含有 3 个波形发生模块，对应于 3 对 PWM 输出：PWM1A/PWM1AN、PWM1B/PWM1BN 和 PWM1C/PWM1CN。

16 位 PWM 计数器与占空比寄存器 PWM1xDL/H (x=A,B,C, 下同) 比较以产生 Duty\_Ox 波形，PWM1x/PWM1xN 引脚上输出的 PWM 波形使用的是同一占空比寄存器 PWM1xDL/H。

由寄存器 PWM1CR1 设置 PWM1x/PWM1xN 引脚上输出波形的极性、由寄存器 PWM1DT0 设置死区 (详见 9.5 章节“PWM 死区与极性控制模块”)。

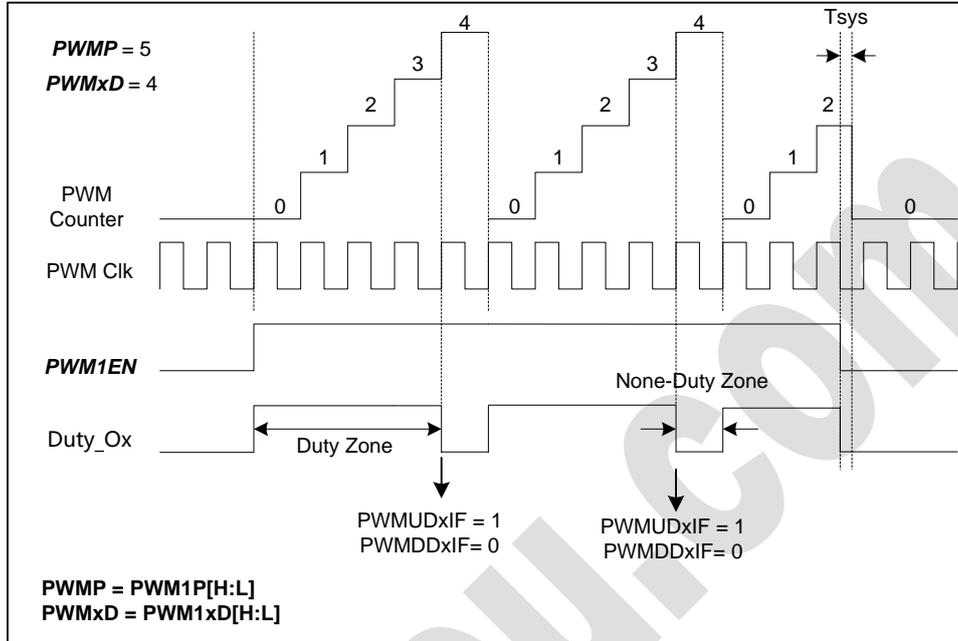
### 10.3.2 边沿对齐计数模式下的 PWM1A/B/C 原始波形



设置 PWM1CR0 中 CTMOD[1:0] = 00，将 PWM 时基设置为边沿对齐计数模式。

PWM 原始输出波形 Duty\_Ox (x=A,B,C, 下同) 如图所示。PWM1EN 置 1 后，Duty\_Ox 波形变高，PWM 计数器与占空比寄存器 PWM1xDL/H 匹配后 Duty\_Ox 波形将变低直至 PWM 计数器归零，如此往复。

边沿对齐计数模式下的 PWM 原始波形



### 10.3.3 中心对齐计数模式下的 PWM1A/B/C 原始波形

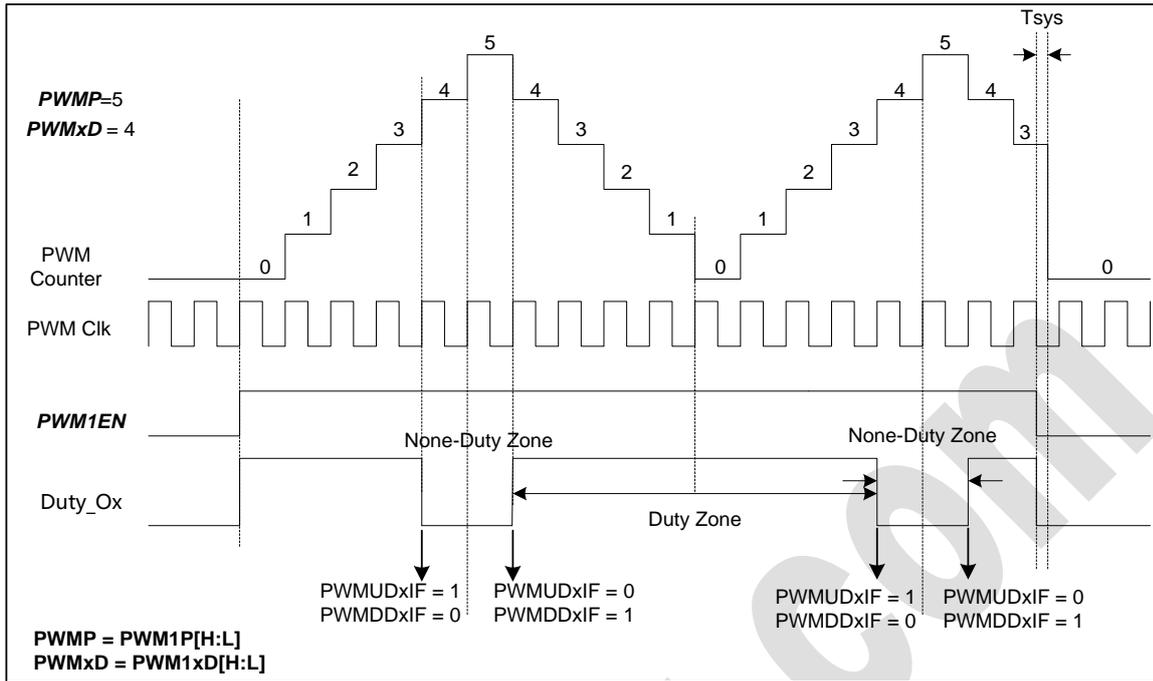
设置 PWM1CR0 中 CTMOD[1:0] = 01，将 PWM 时基设置为中心对齐计数模式（对称模式）。

PWM 原始输出波形 Duty\_Ox 如图所示。PWM1EN 置 1 后，Duty\_Ox 波形保持为高，PWM 计数器增计数时与 PWM1xDL/H 匹配后 Duty\_Ox 波形将变低直至 PWM 计数器减计数时再次与 PWM1xDL/H 匹配，如此往复。

注：x=A,B,C



中心对齐计数模式下的 PWM 原始波形 (对称波形输出)

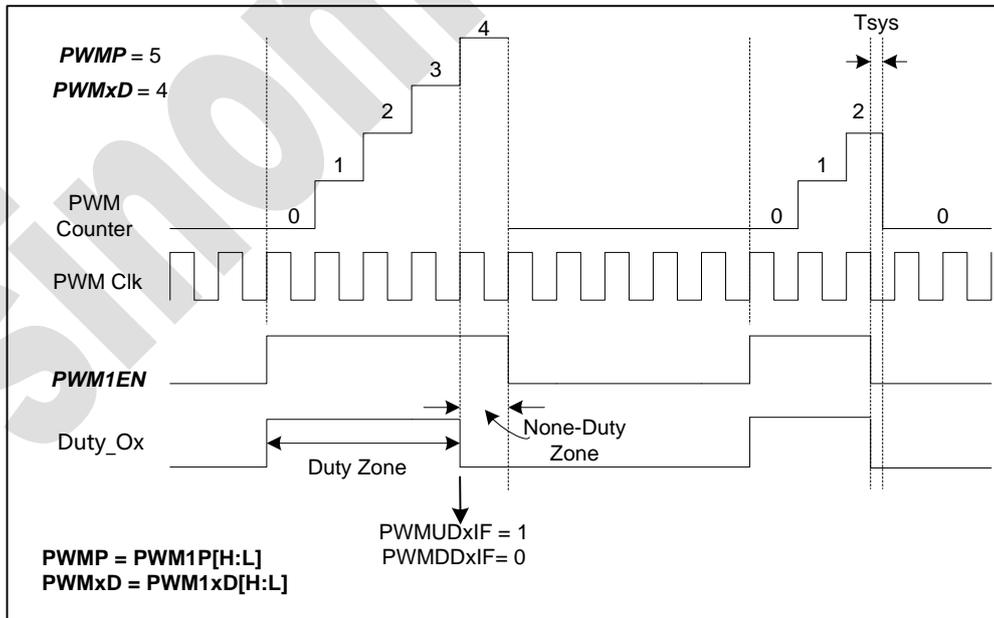


10.3.4 单次计数模式下的 PWM1A/B/C 原始波形

设置 PTCON 中 CTMOD[1:0] = 10 或 11, 将 PWM 时基设置为单次计数模式。

PWM 原始输出波形 Duty\_Ox (x=A,B,C, 下同) 如图所示。PWM1EN 置 1 后, Duty\_Ox 波形变高, PWM 计数器增计数时与 PWM1xDL/H 匹配后 Duty\_Ox 波形将变低。

单次计数模式下的 PWM 原始波形



10.3.5 占空比寄存器的自动重载

3 个占空比寄存器都带有对应的缓存寄存器 (ShadowReg), 16 位的 PWM 计数器实际上是与 ShadowReg 比较从



而控制 PWM 原始波形的变化。用户能够读写 3 个占空比寄存器，却不能操作其对应的缓存寄存器。

由于存在缓冲寄存器，和周期寄存器一样，对占空比寄存器的读写有特殊规定：

当 PWM1RLEN 寄存器的值设为 0x55 时，才允许对占空比寄存器进行修改，然后只有当 PWM1RLEN 寄存器中的值设为 0xAA 时，且分频后的归零信号发生时才会将占空比寄存器值锁存至对应的缓存寄存器（ShadowReg）中，这样可以避免寄存器修改过程中发生重载，也避免寄存器修改立即生效可能会出现毛刺。因此对上述寄存器的修改，必须要有对 PWM1RLEN 寄存器的配合操作才能完成。

另外，在中心对齐模式下，和周期寄存器不同，占空比寄存器除了可以用归零匹配信号重载外，还可以用周期匹配信号重载，如果将 PWM1CR2 中 PLDEN 位为 1，则分频后的周期匹配信号硬件自动将 3 个占空比寄存器锁存至对应的缓存寄存器中（称之为重载）。

因此，在边沿计数模式和单次计数模式下，由于时基模块只有归零匹配信号，故 PWM 计数器每次复位为 0 时都将发生一次占空比寄存器的重载；中心对齐计数模式下，时基模块能产生归零匹配以及周期匹配信号，故 PWM 计数器每次复位为 0 以及与周期寄存器匹配时都可以重载占空比寄存器。

*注：为适应 keil 中 C 语言 16 位寄存器赋值顺序，上述 2 个 8 位寄存器赋值必须先赋值高位寄存器 PWM1ADH/PWM1BDH/PWM1CDH 再赋值低位寄存器 PWM1ADL/PWM1BDL/PWM1CDL 才能正确赋值成功；如果顺序相反只能赋值 PWM1ADL/PWM1BDL/PWM1CDL 寄存器，PWM1ADH/PWM1BDH/PWM1CDH 为之前旧值。*

### 10.3.6 占空比中断

在 PWM 时基计数器向上计数时，不论在何种模式，当计数值与占空比寄存器 PWM1xDL/H 匹配时，PWM1IF 寄存器中 PWMUDxIF 位硬件置 1，同时 PWMDDxIF 位硬件清 0，此时若 PWM1IE 寄存器中的 PWMUDxIE 位为 1，则触发占空比中断。

在中心对齐计数模式中，PWM 时基计数器向下计数时，当计数值与 PWM1xDL/H 匹配时，PWM1IF 寄存器中 PWMDDxIF 位硬件置 1，同时 PWMUDxIF 位硬件清 0，此时若 PWM1IE 寄存器中的 PWMDDxIE 位为 1，则触发占空比中断。

*注 1：占空比中断在 PWM 计数器与 PWM1xDL/H 值匹配时产生占空比中断标志。*

*注 2：PWM1IF 寄存器 PWMDDxIF 位，仅在在中心对齐计数模式中，PWM 时基计数器向下计数时产生；PWM1IF 寄存器中 PWMUDxIF 位在任何模式的时基计数器向上计数产生，包括中心对齐计数模式。*

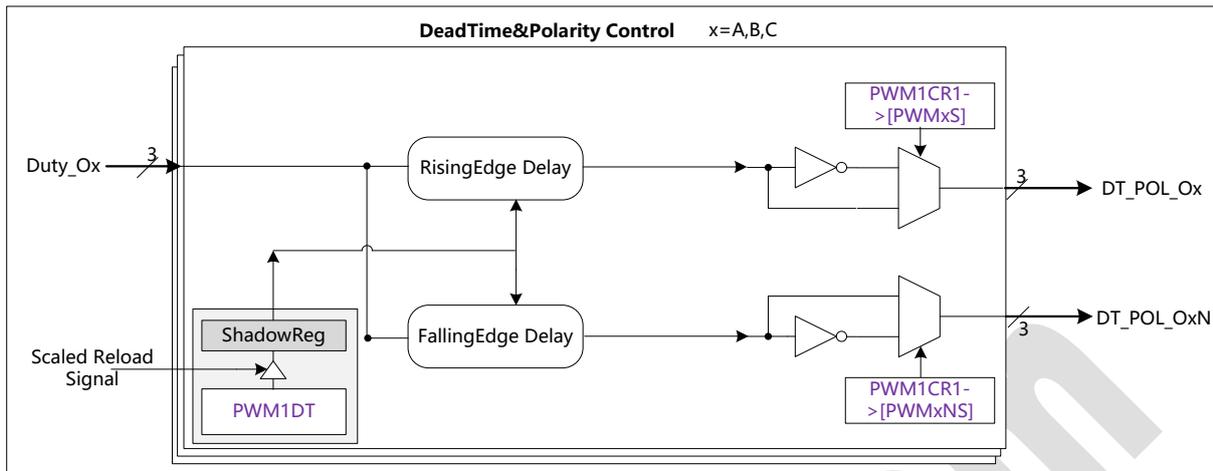
*注 3：PWMDDxIF 和 PWMUDxIF，任一置位硬件清零另一标志位，除此之外软件可写 0 清零。*

## 10.4 PWM 死区与极性控制模块

波形发生模块（Wave Generator）产生的 PWM 原始波形 Duty\_Ox 经过死区与极性控制模块（Dead&Polarity Control）产生最终的 PWM 波形 DT\_POL\_Ox 和 DT\_POL\_OxN。带有死区与极性的 PWM 信号 DT\_POL\_Ox/DT\_POL\_OxN 由输出控制模块（Output Control）和刹车检测模块（Brake Feedback Control）决定是否由引脚 PWM1x/PWM1xN 送出，（x = A,B,C）。



## 死区与极性控制模块的逻辑框图



## 10.4.1 PWM 死区的实现

如死区与极性控制模块框图所示，死区控制逻辑通过将 PWM 原始信号 Duty\_Ox 的上升沿和下降沿进行延迟控制实现。PWM 原始信号 Duty\_Ox 将经过死区逻辑，即 PWM1x/PWM1xN 引脚将引入死区。(x=A,B,C)

死区时间由寄存器 PWM1DT0 决定，上升沿和下降沿死区时间相同。

死区延迟时间 (Valid Edge Delay) =  $\text{PWM1DT0} \times T_{\text{PWM CLK}}$

PWM1DT0 = 0x0 时，原始 PWM 波形对应的有效沿不会触发死区延时。

注 1: 在不同的极性设置下，插入死区后的 PWM 波形也将不同，详见“不同极性设置下带死区的 PWM 波形”章节。

## 10.4.2 PWM 极性设置

为了有效说明 PWM 引脚上波形的极性，将一个 PWM 周期划分为“占空比时区 (Duty Zone)”与“非占空比时区 (None-Duty Zone)”，各种时基计数模式下，占空比时区和非占空比时区的划分参见章节 9.4.4 框图所示。由寄存器 PWM1CR1 中的位 PWMxS/PWMxNS 来决定占空比时区期间 DT\_POL\_Ox 和 DT\_POL\_OxN 的电平。

PWMxS 控制 DT\_POL\_Ox 的极性。PWMxS = 0 时，DT\_POL\_Ox 的占空比时区期间为高电平，非占空比时区期间为低电平；PWMxS = 1 时，DT\_POL\_Ox 的占空比时区期间为低电平，非占空比时区期间为高电平。

PWMxNS 控制 DT\_POL\_OxN 的极性。PWMxNS = 0 时，DT\_POL\_OxN 的占空比时区期间为低电平，非占空比时区期间为高电平；PWMxNS = 1 时，DT\_POL\_OxN 的占空比时区期间为高电平，非占空比时区期间为低电平。

结合章节 9.3.2~9.3.4 框图所示的 PWM 原始波形 Duty\_Ox 和章节 9.5 模块框图可以清晰的反映死区与极性的实现。

## 10.4.3 PWM 死区的自动重载

PWM 死区寄存器 (PWM1DT0)，也和占空比、周期一样，有对应的缓冲寄存器 (ShadowReg)，自动重载功能类似于占空比的自动重载，可以在周期匹配和归零匹配信号发生时重载。

由于存在缓冲寄存器，和占空比寄存器一样，对死区寄存器的读写有特殊规定：

当 PWM1RLEN 寄存器的值设为 0x55 时，才允许对死区寄存器 (PWM1DT0) 进行修改，然后只有当 PWM1RLEN 寄存器中的值设为 0xAA 时，且分频后的归零信号发生时才会将对应寄存器值锁存至对应的缓存寄存器 (ShadowReg) 中，这样可以避免寄存器修改过程中发生重载，也避免寄存器修改立即生效可能会出现毛刺。因此对上述寄存器的修改，必须要有对 PWM1RLEN 寄存器的配合操作才能完成。

另外，在中心对齐模式下，和占空比寄存器相同，死区寄存器除了可以用归零匹配信号重载外，还可以用周期匹



配信号重载，如果将 PWM1CR2 中 PLDEN 位为 1，则分频后的周期匹配信号硬件自动将对应比寄存器锁存至对应的缓存寄存器中（称之为重载）。

因此，在边沿计数模式和单次计数模式下，由于时基模块只有归零匹配信号，故 PWM 计数器每次复位为 0 时都将发生一次死区寄存器的重载；中心对齐计数模式下，时基模块能产生归零匹配以及周期匹配信号，故 PWM 计数器每次复位为 0 以及与周期寄存器匹配时都可以重载死区寄存器。

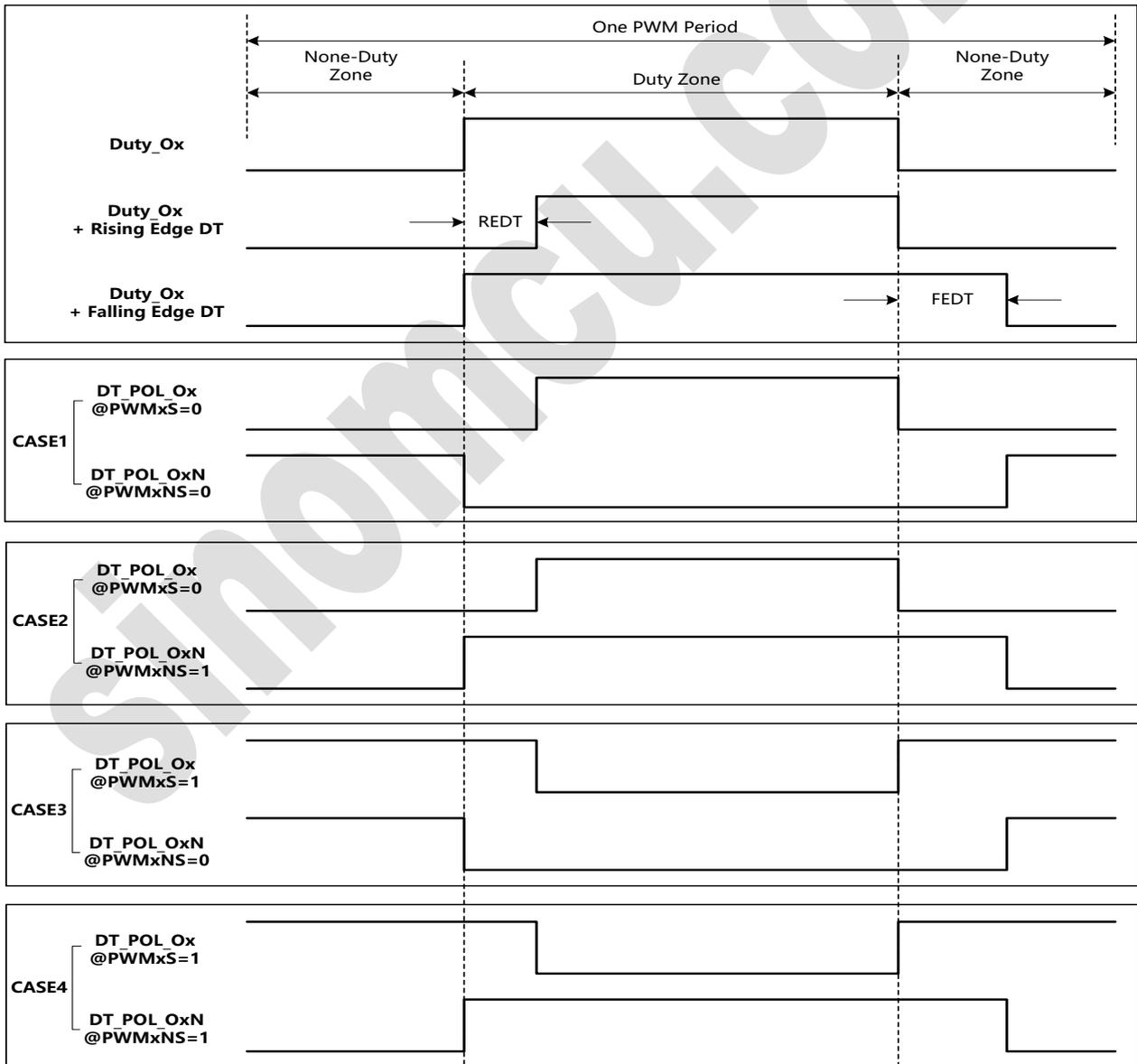
注：PWM1CR2 中 PLDEN 位，对占空比和死区同步控制，共享控制位。

### 10.4.4 不同极性设置下带死区的 PWM 波形

下图给出了一对 PWM 输出 DT\_POL\_Ox/DT\_POL\_OxN 波形，所有的极性组合下带有死区的波形。

注：PWM 启动第一个周期前，增加死区保护；中心对齐模式，计数 0 启动也会有额外增加死区控制，以防初始态上下桥开关管切换导致的瞬态导通。

不同极性设置下的死区波形



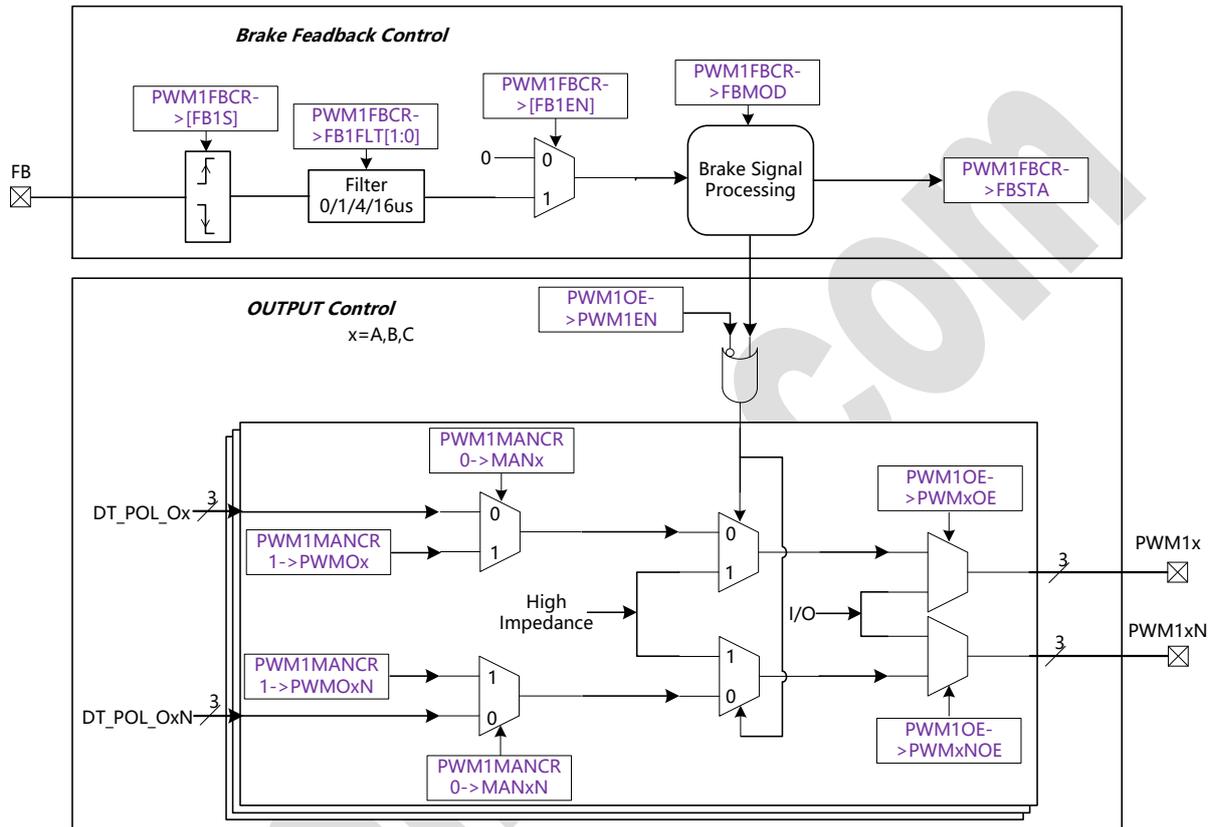
注：图中的 REDT 表示上升沿死区，FEDT 表示下降沿死区。



## 10.5 PWM1A/B/C 输出控制与刹车检测

PWM 输出控制与刹车检测模块 (Output Control & Brake Feedback Control) 用于决定最终的 PWM 波形 DT\_POL\_Ox/DT\_POL\_OxN 是否在 PWM1x/PWM1xN 引脚上输出。

输出控制与刹车检测模块的逻辑框图



刹车检测模块 (Brake Feedback Control), 当 PWM1FBCR 寄存器中 FB1EN 位为 1 时, 使能该模块功能, 若 FB1EN 为 0, 则禁止该模块功能。

PWM 刹车检测应用场景:

当刹车事件发生时 (比如过流) 可以切断 PWM 的输出, 进入无效驱动状态 (输出高阻态), 进而达到保护外部功率器件的目的, 由于硬件控制, 因此响应速度非常快。参见章节 9.6<输出控制与刹车检测模块的逻辑框图>, 刹车检测模块结合输出控制模块 (Output Control) 能够实现前述的保护功能。

### 10.5.1 PWM 刹车信号输入源的选择

可以选择 FB 引脚输入作为刹车检测输入信号, 参见章节 9.5<输出控制与刹车检测模块的逻辑框图>。若刹车信号来自 FB 引脚, 可以选择高电平或低电平有效, 对应的有效电平发生时将自动切断 PWM 的输出。

若 FB1EN=1, 则 FB 引脚输入接入刹车检测输入信号, 一旦 FB 引脚输入变高 (FB1S=0 时) 或变低 (FB1S=1 时), 并保持滤波时间, 则六路 PWM 引脚 PWM1x/PWM1xN 立即输出高阻态 (x=A,B,C)。如果 FB1EN=0, 则禁止 FB 引脚的刹车检测功能, FB 引脚作为普通 IO 使用。

PWM 应用场景: 逆变桥控制

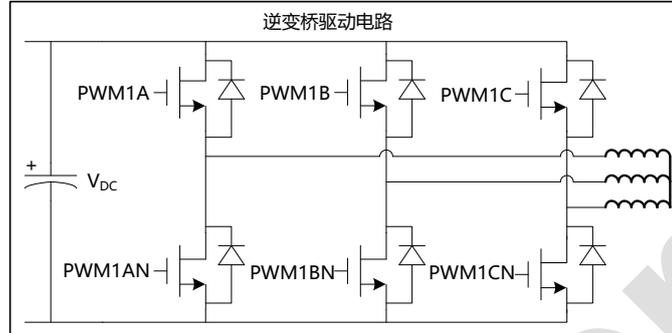
下图为逆变桥控制电路, 六个管子为高电平驱动的功率管。



设置 PWMxS/PWMxNS=0 (x=A,B,C), 则 PWM1A/B/C 的输出在占空比时区期间为高电平, PWM1AN/BN/CN 的输出在占空比时区期间为低电平 (见 “PWM 极性的设置” 章节);

PWM1A 与 PWM1AN 互补输出、PWM1B 与 PWM1BN 互补输出、PWM1C 与 PWM1CN 互补输出。

使能刹车检测后, 若发生刹车事件, 则六路 PWM 立即输出高阻态。



### 10.5.2 PWM 刹车输入信号滤波

#### FB 引脚输入信号

当 FB1EN = 1 时, FB 引脚输入接入刹车检测输入信号, 此时可以通过设置寄存器 PWM1FBCR 中的 FB1FLT[1:0]位调整此输入信号的滤波时间。

FB1FLT [1:0] = 00b 时, 无滤波作用; 刹车检测输入检测到有效信号立即触发保护, 自动切断 PWM 的输出。

FB1FLT [1:0] = 01/10/11b 时, 对应滤波时间为 1 $\mu$ s、4 $\mu$ s、16 $\mu$ s; 刹车检测输入的有效信号电平至少保持 FB1FLT [1:0]定义的时间长度, 才会触发保护, 自动切断 PWM 的输出; 检测到的无效电平也要至少保持 FB1FLT [1:0]定义的时间长度, 刹车检测才认为 FB 引脚的有效电平信号消失, PWM 输出根据保护模式 (章节 9.6.3 介绍) 执行动作。

当 FB1S=0 时, FB 刹车检测信号变为低电平有效; 当 FB1S=1 时, FB 刹车检测信号变为高电平有效。

注: 根据功率管的特性, 恰当的设置滤波时间可以滤除刹车检测输入信号上的噪声。

### 10.5.3 PWM 刹车保护模式

根据刹车保护的解除方式不同分两种刹车保护模式: 锁存模式。

#### 锁存模式

设置 PWM1FBCR 寄存器中 FBMOD=0, 配置刹车保护模式为锁存模式。

在锁存模式下, 当刹车检测模块检测到有效刹车信号, 立即触发保护, 六路 PWM 引脚 PWM1x/PWM1xN (x=A,B,C) 立即输出高阻态。FBSTA 作为标志位使用, 硬件置位, 软件清零; FBSTA 在检测到有效刹车信号时置位。

若有效刹车信号没有消失, 六路 PWM 引脚 PWM1x/PWM1xN (x=A,B,C) 将保持输出高阻态, 软件无法清除 FBSTA 标志位, 六路 PWM 引脚无法恢复正常输出。

当有效刹车信号消失后, 六路 PWM 引脚 PWM1x/PWM1xN (x=A,B,C) 亦不会恢复正常输出, 只有软件将 FBSTA 位清 0 后, PWM1x/PWM1xN 才会在 FBSTA 被清零之后的最近一次 PWM 计数器归零匹配或周期匹配时 (同时后分频计数也归零) 恢复正常输出 (边沿对齐模式中, 在 PWM 计数器归零匹配时恢复输出; 中心对齐模式中, 在 PWM 计数器与周期匹配或归零匹配时恢复输出)。

### 10.5.4 PWM 输出控制模块



## PWM 输出引脚与 IO 功能复用

PWM 引脚 PWM1x/PWM1xN (x=A,B,C) 与 IO 功能复用, 由寄存器 PWM1OE 控制。令 PWM1OE 中的 PWMxOE 位为 1, 可以将 PWM1x 引脚设置为 PWM 输出功能, 反之为 IO 功能; 令 PWM1OE 中的 PWMxNOE 位为 1, 可以将 PWM1xN 引脚设置为 PWM 输出功能, 反之为 IO 功能。参见章节 9.6<输出控制与刹车检测模块的逻辑框图>。

*注: PWM 引脚 PWM1x/PWM1xN (x=A,B,C) 与 IO 功能复用, 首先需要 IO 数字输出设置并选择 PWM1 功能才会有效。*

## 手动控制 PWM 输出

当 PWMxOE/PWMxNOE 位为 1, 且 PWM1EN 位为 1 时, 若 PWMMANCR0 寄存器中的 MANx/MANxN 位为 1, 则引脚 PWM1x/PWM1xN (x=A,B,C) 将输出 PWMMANCR1 寄存器中 PWMOx/PWMOxN 位的值; 若 PWMMANCR0 寄存器中的 MANx/MANxN 位为 0, 则引脚 PWM1x/PWM1xN (x=A,B,C) 将输出 PWM 波形 DT\_POL\_Ox/DT\_POL\_OxN。

此功能和 I/O 口的输出功能类似, 唯一的区别是, 在手动控制 PWM 输出模式下, 其输出会受到 PWM 刹车检测输入信号控制, 而在 I/O 输出模式下, 不受该信号控制。参见章节 9.6<输出控制与刹车检测模块的逻辑框图>。

更改 PWMMANCR0 和 PWMMANCR1 寄存器时, 若 PWM1CR2 的 OSYNC 位为 0 时, 引脚 PWM1x/PWM1xN (x=A,B,C) 输出的改变与系统时钟同步, 输出状态改变立即生效; 若 PWM1CR2 的 OSYNC 位为 1 时, 引脚 PWM1x/PWM1xN (x=A,B,C) 输出的改变与 PWM 周期同步, 输出状态在 PWM 计数归零时生效。

*注: 无论 MANx/MANxN 是否为 1, 当寄存器 PWM1OE 中的 PWMxOE/PWMxNOE 为 0 时, 引脚 PWM1x/PWM1xN (x=A,B,C) 都将成为普通 IO 口。*

## 时基停止时 PWM 引脚的输出

PWM1OE 中 PWM1EN 位为 0 时, 时基模块将停止工作, 同时若 PWM1OE 中 PWMxOE/PWMxNOE 位为 1, 则 PWMx/PWMx1 引脚将输出高阻态。具体参见章节 9.6<输出控制与刹车检测模块的逻辑框图>。

## 10.6 事件触发功能

PWM 时基模块产生的分频后的归零匹配信号 (Scaled Zero Match) 和分频后的周期匹配信号 (Scaled Period Match) 可以用来触发启动 ADC 或者 Timer2。

### 10.6.1 PWM 计数器触发 Timer2

配置寄存器 PWM1CR2 中的 TRGTIM[1:0]位且 Timer2 工作在自动触发模式 (SIGTRG=1), 可以设置 PWM 计数器自动触发启动 Timer2。触发方式配置参见下表。

#### PWM 计数器匹配触发启动 TIMER2 控制

PWM1CR2-> TRGTIM[1:0]	T2CON1-> SIGTRG	自动触发启动 Timer2 单次方式
00	x	PWM 模块不会触发启动 Timer2
01	1	中心对齐模式下, 分频后的 PWM 周期匹配信号启动 Timer2 定时, 溢出后自动停止定时 边沿对齐或单次模式下, 分频后的 PWM 归零匹配信号启动 Timer2 定时, 溢出后自动停止定时



10	1	中心对齐模式下，分频后的 PWM 归零匹配信号启动 Timer2 定时，溢出后自动停止定时 边沿对齐或单次模式下，分频后的 PWM 归零匹配信号启动 Timer2 定时，溢出后自动停止定时
11	1	中心对齐模式下，分频后的 PWM 周期匹配和归零匹配信号均启动 Timer2 定时，溢出后自动停止定时 边沿对齐或单次模式下，分频后的 PWM 归零匹配信号启动 Timer2 定时，溢出后自动停止定时

### 10.6.2 PWM 计数器触发 ADC

若寄存器 PWM1CR2 中的 ZTRGAD = 1，则分频后的归零信号可以自动触发一次 ADC 序列的转换（如果 ADC 模块中的 ADCCR0->ADEN 位和 ADCCR2->ADCTRG2 位被允许，将会启动一次序列转换）；若 ZTRGAD = 0，则归零信号不会自动触发 ADC 序列。

PWM1CR2-> ZTRGAD	ADCCR2-> ADCTRG2	ADCCR0-> ADEN	自动触发启动 ADC 方式
x	0	0	ADC 工作在非自动触发模式
0	1	1	PWM 模块分频后归零信号不会触发启动 ADC
1	1	1	PWM 模块分频后归零信号会触发启动 ADC

若寄存器 PWM1CR2 中的 PTRGAD = 1，则分频后的周期匹配信号可以自动触发一次 ADC 序列的转换（如果 ADC 模块中的 ADCCR0->ADEN 位和 ADCCR4->ADCTRG3 位被允许，将会启动一次序列转换）；若 PTRGAD = 0，则周期匹配信号不会自动触发 ADC 序列。

PWM1CR2-> PTRGAD	ADCCR2-> ADCTRG2	ADCCR0-> ADEN	自动触发启动 ADC 方式
x	0	0	ADC 工作在非自动触发模式
0	1	1	PWM 模块分频后周期匹配信号不会触发启动 ADC
1	1	1	PWM 模块分频后周期匹配信号会触发启动 ADC

上述，PWM 自动触发启动 AD，必须配置 ADC 模块中的 ADCCR0->ADEN 位和 ADCCR2->ADCTRG2 位被允许。

## 10.7 PWM 模块的中断总汇

PWM 模块的中断包括时基模块产生的 PWM 计数器归零中断、PWM 计数器值与周期寄存器匹配产生的周期中断（详见“时基中断”章节）以及 PWM 计数器值与占空比寄存器 PWM1xDL/H (x=A,B,C) 匹配时产生的占空比中断（详见“占空比中断”章节）。这些中断共享一个中断向量（详见“中断”章节）。

## 10.8 注意事项

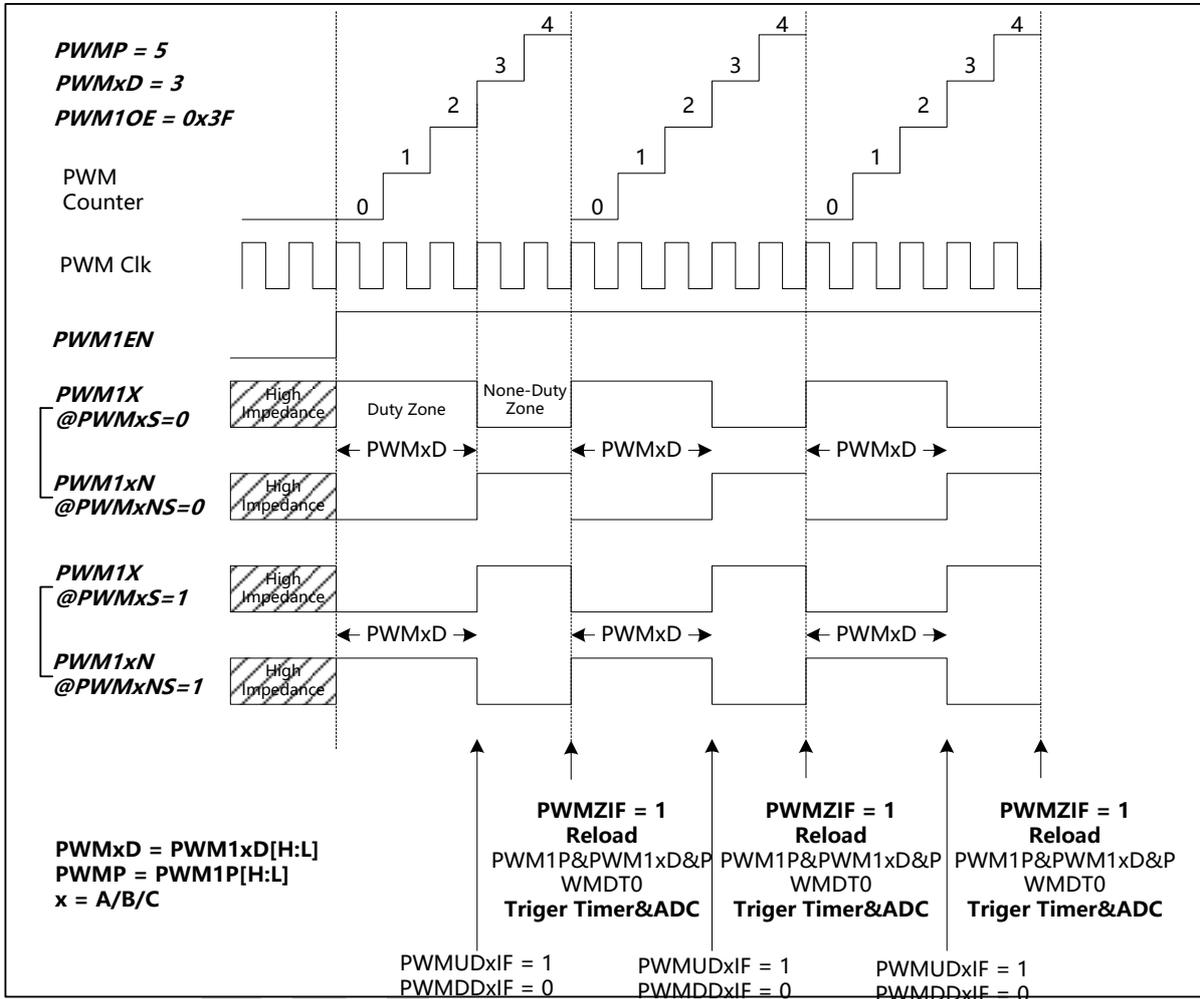
### 10.8.1 边沿对齐计数模式中占空比寄存器值为 0 或周期值

边沿对齐计数模式中，若占空比寄存器为 0 或周期值，则根据不同的极性设置，PWMx/PWMx1 应输出占空比为 0% 或 100% 的波形。令 PWMOE = 0xBF、PWMDT0 = 0 在边沿对齐时基计数模式下，不同占空比值的 PWM 引脚输出的波形、中断标志触发位置、事件触发位置、寄存器重载位置如图 8-1-16~图 8-1-18 所示。



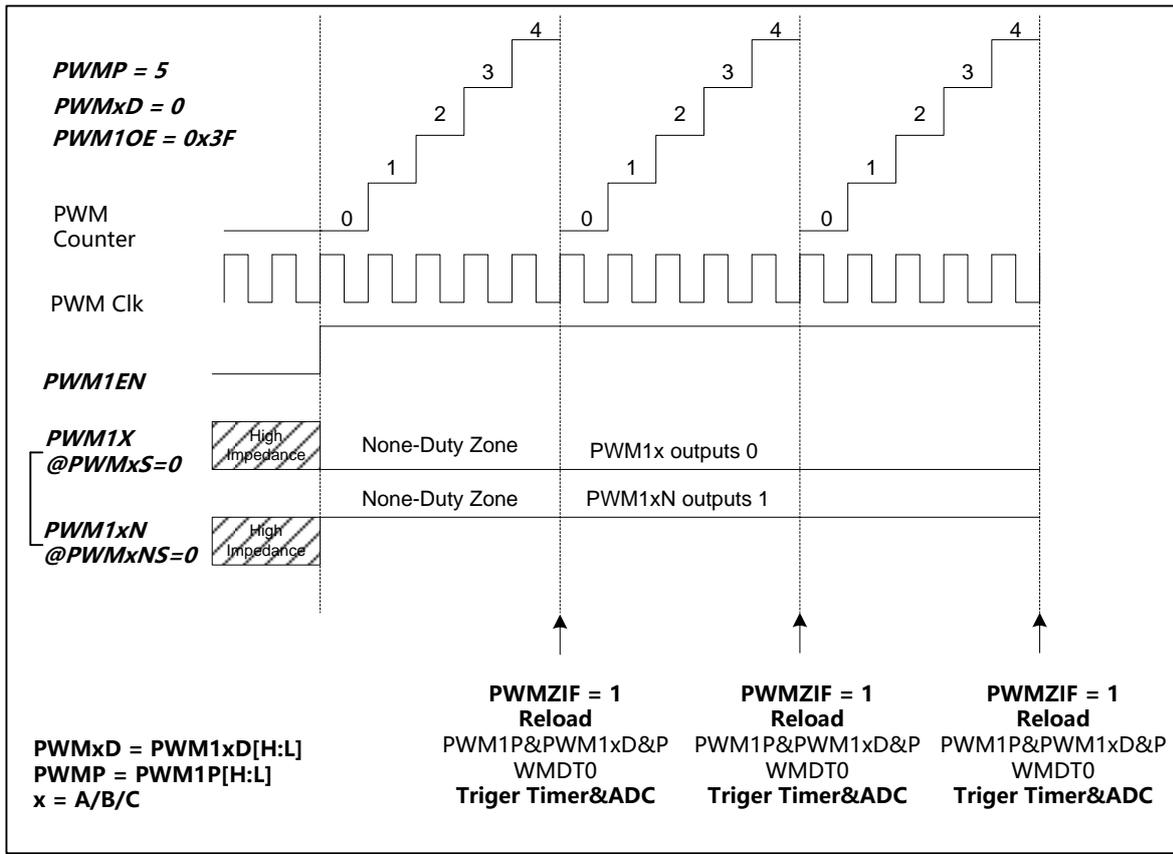
注: PWM 时基 Enable 时刻, 虽然时基计数器为 0, 但不会产生归零(zero match)信号, 因此不会置 PWMZIF 标志, 也不会产生事件触发信号, 而寄存器重载信号在 PWMEN=0 时是立即发生的。

边沿对齐计数模式、占空比大于 0&小于周期值





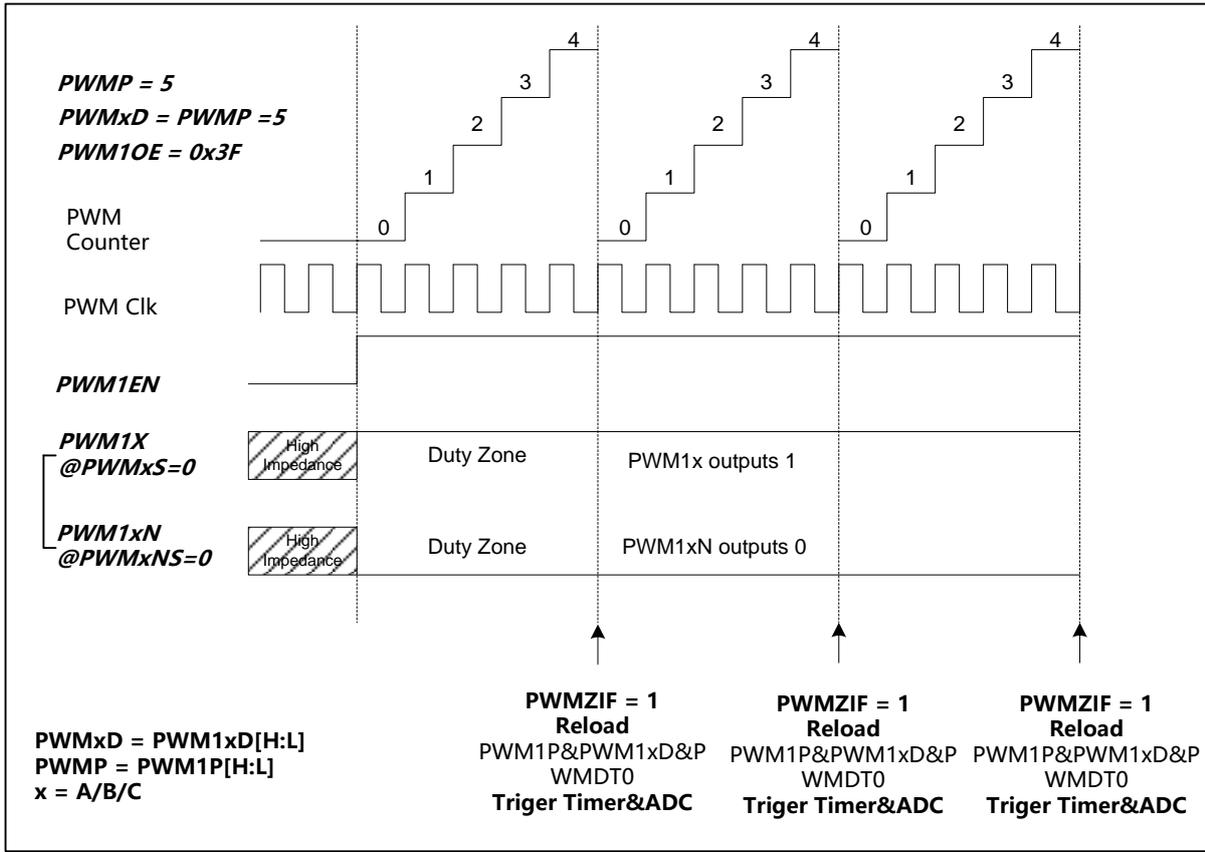
边沿对齐计数模式，占空比=0



Sinomicro



边沿对齐计数模式，占空比=周期值



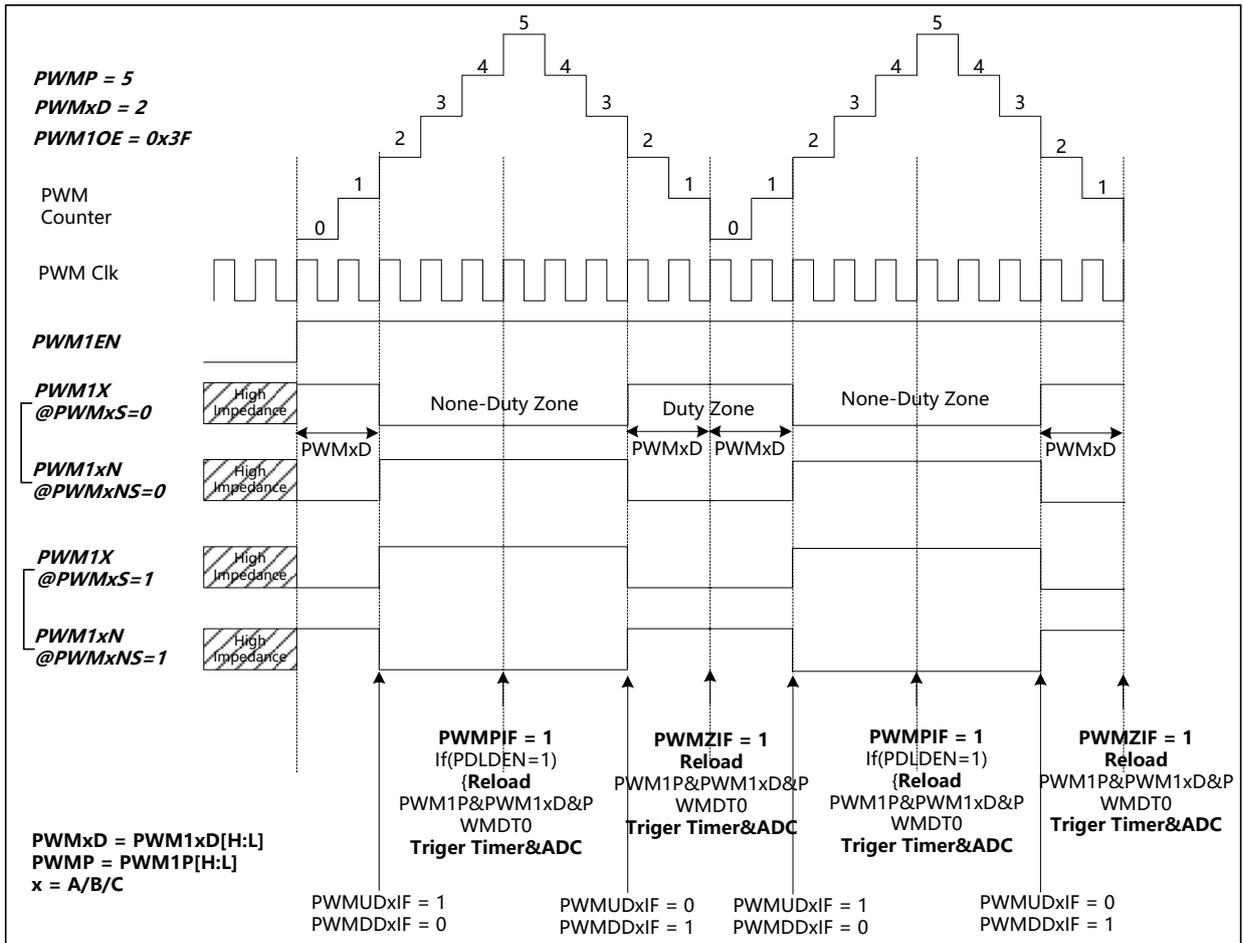
### 10.8.2 中心对齐计数模式中占空比寄存器值为 0 或周期值

中心对齐计数模式中，若占空比寄存器为 0 或周期值，则根据不同的极性设置，PWM1x/PWM1xN 应输出占空比为 0%或 100%的波形。令 PWM1OE = 0xBF、PWM1DT0 = 0，在边沿对齐计数模式下，不同占空比值的 PWM 引脚输出的波形、中断标志触发位置、事件触发位置、寄存器重载位置如下图所示。

注：在中心对齐计数模式中，当占空比寄存器为 0 或周期值时，占空比较匹配中断标志不会置起。



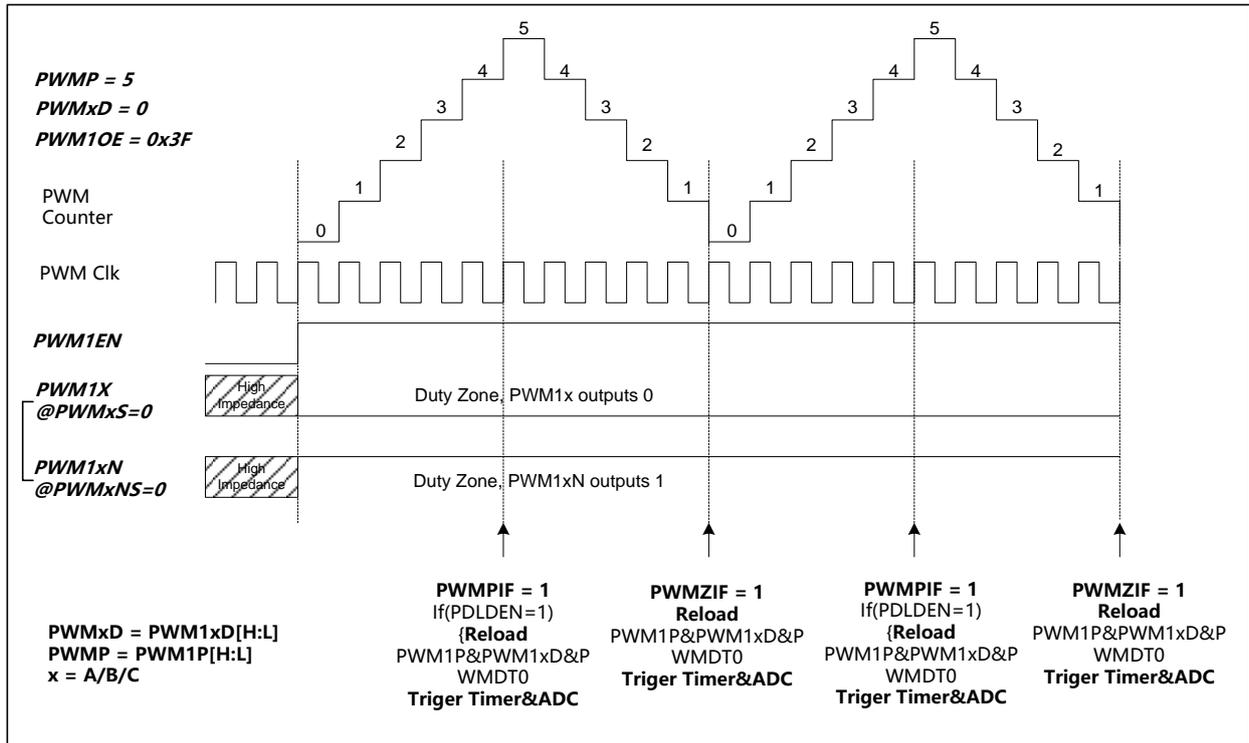
中心对齐计数模式、占空比大于 0&小于周期值



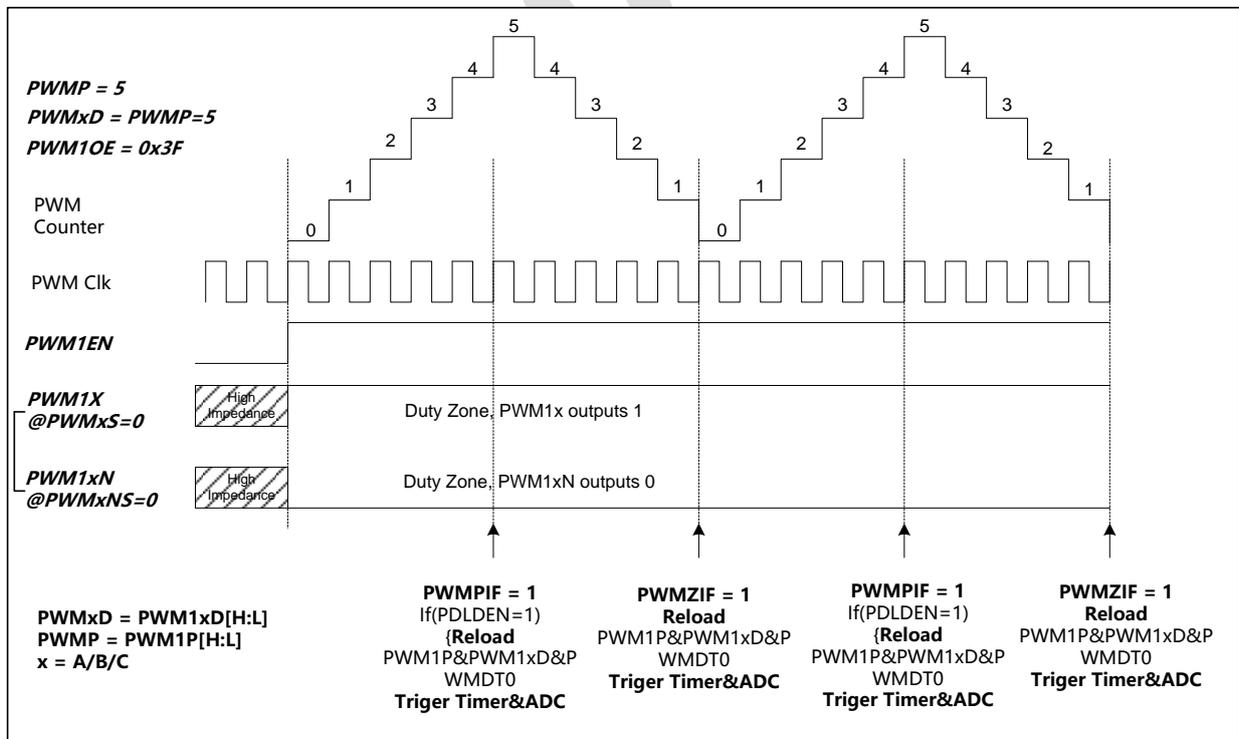
Sinom



中心对齐计数模式、占空比=0



中心对齐计数模式、占空比=周期值



10.8.3 周期寄存器值为 0

不管时基计数器在何种模式，若周期寄存器为 0，则根据不同的极性设置，PWM1x/PWM1xN 应输出占空比为 0% 的波形。此时，所有中断标志都不会产生，也不会产生事件触发信号，但重载信号会产生。



#### 10.8.4 仿真状态的 PWM 输出

支持在线仿真，在仿真状态运行和正常状态运行没有区别，如果在仿真状态停止运行或单步运行，PWM 输出会切换为高阻态，保证不会误触发外部功率管。

#### 10.8.5 进入 STOP 模式，PWM1A/B/C 的波形输出状态

进入 STOP 模式后，PWM 输出口不受影响，比如 PWM0 口原先输出 1K 的方波，进入 STOP 模式后会继续输出 1K 的方波，但是如果设置了 PWM1 中断，会唤醒 STOP 模式，继续执行 IDLE 后面的程序。

#### 10.8.6 进入 SLEEP 模式，PWM1A/B/C 的波形输出状态

进入 SLEEP 模式后，PWM 输出口会输出高阻状态，即使设置了 PWM1 中断，也不能唤醒 SLEEP 模式，SLEEP 模式必须通过其他方式唤醒，具体参考工作模式章节。

### 10.9 代码示例

```
#include "MC51Fxxxxx.h" /*用户开启内部 16.6M HIRC*/
void Pwm1_Isr(void) interrupt 11 using 1
{
    PWM1IF = 0;
    PWM1RLEN = 0x55;           //解保护，允许修改 PWM 受保护寄存器
    PWM1AD16++;               //每次中断将 PWM 占空比加 1
    PWM1RLEN = 0xAA;         //启动加载
}
void System_Initial()
{
    EA = 1;
    IEN1 |= 0x04;
}
void Pwm1_Initial()
{
    PWM1RLEN = 0x55;           //解保护，允许修改 PWM 受保护寄存器
    PWM1OE = 0x3f;           //将 6 个 I/O 口设置为 PWM 输出口
    PWM1CR0 = 0x00;          //PWM 时钟选择系统时钟，设置边沿对齐模式，预分频为 1，后分频为
1
    PWM1CR1 = 0x00;          //PWM 输出互补模式 (PWM1x 与 PWM1xN 波形互补，x=A,B,C)
    PWM1P16 = 6000;          //PWM 的周期设为 6000 Fsys clk
    PWM1AD16 = 0;            //PWM1A 占空比设置为 0 Fsys clk
    PWM1BD16 = 1000;         //PWM1B 占空比设置为 1000 Fsys clk
    PWM1CD16 = 3000;         //PWM1C 占空比设置为 3000 Fsys clk
    PWM1IE = 0x40;           //PWM 归零匹配中断允许
    PWM1OE |= 0x80;          //打开 PWM 总时基
    PWM1RLEN = 0xAA;         //启动加载，将 PWM 带缓存的寄存器值载入 (周期，占空比，死区等)
```

以下是输出 PWM 波形的一个简单示例程序



## 10.10 增强型 PWM1 寄存器

PWM1 模块所使用寄存器如下：

功能	名称	寄存器描述
PWM 计数器时钟（时基）设置	PWM1CR0	设置 PWM 时基的预分频、PWM 模块中控制信号的后分频以及 PWM 计数器的运行模式
PWM 控制寄存器	PWM1CR1	设置 PWM 波形的输出极性及输出模式
	PWM1CR2	PWM 触发事件的使能、重载
PWM 输出控制	PWM1OE	设置 PWM 引脚输出 PWM 模块产生的信号或者作为普通 IO 口
	PWM1MANCR0	设置 PWM 模块输出 PWM 波形或直接输出 PWM1MANCR1 中定义的电平
	PWM1MANCR1	定义 PWM 模块直接输出的电平
PWM 死区控制	PWM1DT0	设置 PWM 原始波形上升沿/下降沿触发死区时间
刹车保护	PWM1FBCR	设置刹车保护功能
PWM 中断	PWM1IE	PWM 中断使能位控制
	PWM1IF	PWM 中断标志位
寄存器重载使能控制	PWM1RLEN	禁止/使能占空比寄存器/周期寄存器/PWM 时基预分频的重载
PWM 周期寄存器	PWM1PL	16 位周期寄存器
	PWM1PH	
PWM 占空比寄存器	PWM1ADL	独立输出模式 PWM1xDL/H(x=A,B,C)控制 PWM1x(x=A,B,C)通道原始波形 互补输出模式+对称波形 PWM1xDL/H(x=A,B,C)同时控制 PWM1x 和 PWM1xN (x=A,B,C)通道原始波形
	PWM1ADH	
	PWM1BDL	
	PWM1BDH	
	PWM1CDL	
	PWM1CDH	

## PWM1 寄存器汇总表

地址 (SFRn /H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
SFR0/DE	PWM1MANCR1	PWM1 手动输出控制寄存器 1	-	-	PWMO CN	PWMO BN	PWMO AN	PWMO C	PWMO B	PWMO A	--00 0000
SFR0/DD	PWM1MANCR0	PWM1 手动输出控制寄存器 0	-	-	MANCN	MANBN	MANAN	MANC	MANB	MANA	--00 0000
SFR0/DC	PWM1FBCR	PWM1 刹车控制寄存器	-	-	FB1EN	FB1S	FB1FLT 1	FB1FLT 0	-	FBSTA	--00 00-0
SFR0/	PWM1	PWM1 控制寄	TRGTI	TRGTI	PTRGA	ZTRGA	PLDEN	OSYNC	-	-	0000



DB	CR2	寄存器 2	M1	M0	D	D					00--
SFR0/ DA	PWM1 CR1	PWM1 控制寄存器 1	-	-	PWMC NS	PWMB NS	PWMA NS	PWMCS	PWMB S	PWMA S	--00 0000
SFR0/ D9	PWM1 RLEN	PWM1 寄存器修改重载控制寄存器	PWMRL 7	PWMRL 6	PWMRL 5	PWMRL 4	PWMRL 3	PWMRL 2	PWMRL 1	PWMRL 0	0000 0000
SFR0/ D8	PWM1 F	PWM1 中断标志寄存器	PWMP F	PWMP F	PWMD DCIF	PWMD DCIF	PWMD DBIF	PWMD DBIF	PWMD DAIF	PWMD DAIF	0000 0000
SFR0/ D7	PWM1 CDH	PWM1C 占空比寄存器高位	-	-	-	-	PWMC D11	PWMC D10	PWMC D9	PWMC D8	---- 0000
SFR0/ D6	PWM1 CDL	PWM1C 占空比寄存器低位	PWMC D7	PWMC D6	PWMC D5	PWMC D4	PWMC D3	PWMC D2	PWMC D1	PWMC D0	0000 0000
SFR0/ D5	PWM1 BDH	PWM1B 占空比寄存器高位	-	-	-	-	PWMB D11	PWMB D10	PWMB D9	PWMB D8	---- 0000
SFR0/ D4	PWM1 BDL	PWM1B 占空比寄存器低位	PWMB D7	PWMB D6	PWMB D5	PWMB D4	PWMB D3	PWMB D2	PWMB D1	PWMB D0	0000 0000
SFR0/ D3	PWM1A DH	PWM1A 占空比寄存器高位	-	-	-	-	PWMA D11	PWMA D10	PWMA D9	PWMA D8	---- 0000
SFR0/ D2	PWM1A DL	PWM1A 占空比寄存器低位	PWMA D7	PWMA D6	PWMA D5	PWMA D4	PWMA D3	PWMA D2	PWMA D1	PWMA D0	0000 0000
SFR0/ D1	PWM1 OE	PWM1 输出使能寄存器	PWMP N	-	PWMC NOE	PWMB NOE	PWMA NOE	PWMC OE	PWMB OE	PWMA OE	0-00 0000
XSFR/ FF48	PWM1 CR0	PWM1 控制寄存器 0	-	-	CTMOD 1	CTMOD 0	PREDIV 1	PREDIV 0	POSTDIV V1	POSTDIV V0	0-00 0000
XSFR/ FF49	PWM1 E	PWM1 中断使能控制寄存器	PWMP E	PWMP E	PWMD DCIE	PWMD DCIE	PWMD DBIE	PWMD DBIE	PWMD DAIE	PWMD DAIE	0000 0000
XSFR/ FF4A	PWM1 PH	PWM1 周期寄存器高位	-	-	-	-	PWMP 1	PWMP 0	PWMP 9	PWMP 8	0000 0000
XSFR/ FF4B	PWM1 PL	PWM1 周期寄存器低位	PWMP 7	PWMP 6	PWMP 5	PWMP 4	PWMP 3	PWMP 2	PWMP 1	PWMP 0	0000 0000
XSFR/ FF4D	PWM1 DT0	PWM1 死区控制寄存器 0	PWMD T07	PWMD T06	PWMD T05	PWMD T04	PWMD T03	PWMD T02	PWMD T01	PWMD T00	0000 0000

### 10.10.1 PWM1 控制寄存器 0 (PWM1CR0, 0xFF48/XSFR)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1CR0	-	-	CTMOD1	CTMOD0	PREDIV1	PREDIV0	POSTDIV1	POSTDIV0
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
复位值	-	-	0	0	0	0	0	0

BIT[5:4] CTMOD[1:0] – 计数工作模式选择

00: 边沿对齐计数模式, 在此模式下, 时基计数器只会发生归零匹配

01: 中心对齐计数模式, 在此模式下, 时基计数器会发生归零匹配和周期匹配

10: 单次计数模式, 在此模式下, 时基计数器只会发生归零匹配

11: 保留



- BIT[3:2] PREDIV[1:0] – 时钟预分频选择
- 00: PWM 计数器时钟 =  $F_{pwm}/1$
  - 01: PWM 计数器时钟 =  $F_{pwm}/2$
  - 10: PWM 计数器时钟 =  $F_{pwm}/4$
  - 11: PWM 计数器时钟 =  $F_{pwm}/8$

注:  $F_{pwm}$  由 CLKSEL 控制, 选择系统时钟  $F_{sys}$  或 HIRC48M 时钟。

- BIT[1:0] POSTDIV[1:0] – 时钟后分频选择
- 00: 无后分频功能
  - 01: 中断、重载信号与事件触发信号将 2 分频
  - 10: 中断、重载信号与事件触发信号将 4 分频
  - 11: 中断、重载信号与事件触发信号将 8 分频

### 10.10.2 PWM1 控制寄存器 1 (PWM1CR1, 0xDA/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1CR1	-	-	PWMCNS	PWMBNS	PWMANS	PWMCS	PWMBBS	PWMAS
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
复位值	-	-	0	0	0	0	0	0

- BIT[5,4,3] PWMxNS(x=A,B,C) – PWMxN 通道输出极性选择, x=A,B,C
- 0: PWMxN 占空比期间输出低电平, 其余时间输出高电平
  - 1: PWMxN 占空比期间输出高电平, 其余时间输出低电平

- BIT[2,1,0] PWMxS(x=A,B,C) – PWMx 通道输出极性选择, x=A,B,C
- 0: PWMx 占空比期间输出高电平, 其余时间输出低电平
  - 1: PWMx 占空比期间输出低电平, 其余时间输出高电平

### 10.10.3 PWM1 控制寄存器 2 (PWM1CR2, 0xDB/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1CR2	TRGTIM1	TRGTIM0	PTRGAD	ZTRGAD	PLDEN	OSYNC	-	-
R/W	R/W	R/W	R/W	R/W	R/W	R/W	-	-
复位值	0	0	0	0	0	0	-	-

- BIT[7:6] TRGTIM[1:0] – PWM 计数器匹配触发启动 TIMER2 控制
- 00: PWM 模块不会触发启动 Timer2
  - 01: 中心对齐模式下, 分频后的 PWM 周期匹配信号启动 Timer2  
边沿对齐或单次模式下, 分频后的 PWM 归零匹配信号启动 Timer2
  - 10: 中心对齐模式下, 分频后的 PWM 归零匹配信号启动 Timer2  
边沿对齐或单次模式下, 分频后的 PWM 归零匹配信号启动 Timer2
  - 11: 中心对齐模式下, 分频后的 PWM 周期匹配和归零匹配信号均启动 Timer2  
边沿对齐或单次模式下, 分频后的 PWM 归零匹配信号启动 Timer2  
(Timer2 的 PWM 触发控制开启后, 上述设置才有效)
- BIT[5] PTRGAD – PWM 计数器与周期匹配启动 AD 控制
- 0: PWM 模块周期匹配时不会触发启动 AD
  - 1: PWM 时基计数器与周期寄存器匹配时启动 AD



- (ADC 的 ADEN 和 ADCTRG3 位置 1 的情况下)
- BIT[4] ZTRGAD – PWM 计数器归 0 匹配启动 AD 控制
- 0: PWM 模块归 0 时刻不会触发启动 AD
- 1: PWM 时基计数器归 0 时刻启动 AD
- (ADC 的 ADEN 和 ADCTRG3 位置 1 的情况下)
- BIT[3] PLDEN - PWM 计数器周期匹配时重载占空比使能位: (只在中心对齐模式下有效)
- 0: 周期匹配时刻不允许重载占空比寄存器
- 1: 周期匹配时刻允许重载占空比寄存器

注:

- 1、无论寄存器如何设置，周期匹配时不会重载 PWM 周期寄存器。
- 2、只在中心对齐模式下周期匹配发生时，由 PLDEN 控制，占空比寄存器是否重载。
- 2: 归 0 匹配时会自动重载周期寄存器、占空比寄存器、死区寄存器，此重载动作不受上述控制位影响，PWM1RLEN = 0XAA 条件依然有效。

- BIT[2] OSYNC – 手动修改 PWMx/PWMxN 输出同步位
- (手动修改通过设置 PWMMANCR0 寄存器实现，任一寄存器的修改,何时生效都由此寄存器控制)
- 0: 手动修改后立即生效
- 1: 手动修改与 PWM 周期同步后生效

#### 10.10.4 PWM1 周期寄存器高位 (PWM1PH, 0xFF4A/XSFR)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1PH	-	-	-	-	PWMP11	PWMP10	PWMP9	PWMP8
R/W	-	-	-	-	R/W	R/W	R/W	R/W
复位值	-	-	-	-	0	0	0	0

- BIT[7:0] PWMP[11:8] – PWM1 周期寄存器高位

#### 10.10.5 PWM1 周期寄存器低位 (PWM1PL, 0xFF4B/XSFR)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1PL	PWMP7	PWMP6	PWMP5	PWMP4	PWMP3	PWMP2	PWMP1	PWMP0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

- BIT[7:0] PWMP[7:0] – PWM1 周期寄存器低位

#### 10.10.6 PWM1A 占空比寄存器低位 (PWM1ADL, 0xD2/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1ADL	PWMAD7	PWMAD6	PWMAD5	PWMAD4	PWMAD3	PWMAD2	PWMAD1	PWMAD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

- BIT[7:0] PWMAD[7:0] – PWM1A 占空比寄存器低位

#### 10.10.7 PWM1A 占空比寄存器高位 (PWM1ADH, 0xD3/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1ADH	-	-	-	-	PWMAD11	PWMAD10	PWMAD9	PWMAD8
R/W	-	-	-	-	R/W	R/W	R/W	R/W



复位值	-	-	-	-	0	0	0	0
-----	---	---	---	---	---	---	---	---

BIT[7:0] PWMAD[11:8] – PWM1A 占空比寄存器高位

#### 10.10.8 PWM1B 占空比寄存器低位 (PWM1BDL, 0xD4/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1BDL	PWMBD7	PWMBD6	PWMBD5	PWMBD4	PWMBD3	PWMBD2	PWMBD1	PWMBD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] PWMMD[7:0] – PWM1B 占空比寄存器低位

#### 10.10.9 PWM1B 占空比寄存器高位 (PWM1BDH, 0xD5/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1BDH	-	-	-	-	PWMBD11	PWMBD10	PWMBD9	PWMBD8
R/W	-	-	-	-	R/W	R/W	R/W	R/W
复位值	-	-	-	-	0	0	0	0

BIT[7:0] PWMMD[11:8] – PWM1B 占空比寄存器高位

#### 10.10.10 PWM1C 占空比寄存器低位 (PWM1CDL, 0xD6/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1CDL	PWMCD7	PWMCD6	PWMCD5	PWMCD4	PWMCD3	PWMCD2	PWMCD1	PWMCD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] PWMMD[7:0] – PWM1C 占空比寄存器低位

#### 10.10.11 PWM1C 占空比寄存器高位 (PWM1CDH, 0xD7/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1CDH	-	-	-	-	PWMCD11	PWMCD10	PWMCD9	PWMCD8
R/W	-	-	-	-	R/W	R/W	R/W	R/W
复位值	-	-	-	-	0	0	0	0

BIT[7:0] PWMMD[11:8] – PWM1C 占空比寄存器高位

#### 10.10.12 PWM1 输出使能寄存器 (PWM1OE, 0xD1/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1OE	PWM1EN	-	PWMCNOE	PWMBNOE	PWMANOE	PWMCOE	PWMBOE	PWMAOE
R/W	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	-	0	0	0	0	0	0

BIT[7] PWM1EN – PWM1 总使能位

0: 关闭 PWMA/B/C/AN/BN/CN 时钟

1: 打开 PWMA/B/C/AN/BN/CN 时钟

BIT[5,4,3] PWMxNOE(x=A,B,C) – PWM1xN 输出使能位(x=A,B,C)

0: PWM1xN 引脚作为普通 I/O 口(x=A,B,C)

1: PWM1xN 引脚作为 PWM 输出口,输出 PWM 波形或 PWMOxN 定义的电平(x=A,B,C)

BIT[2,1,0] PWMxOE(x=A,B,C) – PWM1x 输出使能位(x=A,B,C)



0: PWM1x 引脚作为普通 I/O 口(x=A,B,C)

1: PWM1x 引脚作为 PWM 输出口,输出 PWM 波形或 PWMOx 定义的电平(x=A,B,C)

#### 10.10.13 PWM1 手动输出控制寄存器 0 (PWM1MANCR0, 0xDD/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1MANCR0	-	-	MANCN	MANBN	MANAN	MANC	MANB	MANA
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
复位值	-	-	0	0	0	0	0	0

BIT[5,4,3] MANxN(x=A,B,C) – PWM1xN 输出手动控制位, x=A,B,C

(只有当 PWMxNOE=1 时有效)

0: PWMxN 口输出 PWM 波形

1: PWMxN 口输出由 PWMMANCR2 寄存器中的 PWMOxN 位控制

BIT[2,1,0] MANx(x=A,B,C) – PWM1x 输出手动控制位, x=A,B,C

(只有当 PWMxOE=1 时有效)

0: PWMx 口输出 PWM 波形

1: PWMx 口输出由 PWMMANCR2 寄存器中的 PWMOx 位控制

#### 10.10.14 PWM1 手动输出控制寄存器 1 (PWM1MANCR1, 0xDE/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1MANCR1	-	-	PWMOCN	PWMOBN	PWMOAN	PWMOA	PWMOB	PWMOA
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
复位值	-	-	0	0	0	0	0	0

BIT[5,4,3] PWMOxN(x=A,B,C) – MANxN = 1 时, 决定 PWMxN 口的输出电平, x=A,B,C

0: 输出 0

1: 输出 1

BIT[2,1,0] PWMOx(x=A,B,C) – MANxN = 1 时, 决定 PWMxN 口的输出电平, x=A,B,C

0: 输出 0

1: 输出 1

注: 必须先写 PWMMANCR1 寄存器, 后写 PWMMANCR0 寄存器。

#### 10.10.15 PWM1 刹车控制寄存器 (PWM1FBCR, 0xDC/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1FBCR	-	-	FB1EN	FB1S	FB1FLT1	FB1FLT0	-	FBSTA
R/W	-	-	R/W	R/W	R/W	R/W	-	R/W
复位值	-	-	0	0	0	0	-	0

BIT[5] FB1EN – 刹车信号 1 使能控制位

0: 刹车信号 1 功能禁止

1: 刹车信号 1 功能打开, 输入源为 FB 引脚

BIT[4] FB1S – 刹车信号 1 有效电平选择位

0: 刹车信号 1 FB 输入高电平有效



1: 刹车信号 1 FB 输入低电平有效  
 BIT[3:2] FB1FLT[1:0] - 刹车信号 1 输入滤波选择

- 00: 无滤波
- 01: 滤波时间常数为 1 $\mu$ s (24clocks@HIRC/2)
- 10: 滤波时间常数为 4 $\mu$ s (96 clock@HIRC/2)
- 11: 滤波时间常数为 16 $\mu$ s (384 clock@HIRC/2)

注: 上述滤波常数时间不是精确值, 仅供参考。时钟数根据 HRC 频率设计调整

BIT[0] FBSTA - 刹车输入检测标志位

- 0: PWM1A/B/C 通道处于正常输出状态
- 1: 检测到刹车有效输入信号, PWM1A/B/C 通道处于停止输出状态, 如果在锁存模式, 可软件清 0, 恢复 PWM 输出

#### 10.10.16 PWM1 中断使能控制寄存器 (PWM1IE, 0xFF49/XSFR)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1IE	PWMP1E	PWMZIE	PWMDDCI	PWMUDCI	PWMDDBI	PWMUDBI	PWMDDAI	PWMUDAI
			E	E	E	E	E	E
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7] PWMP1E - PWM 时基周期匹配中断允许位 (只在中心对齐模式下有效)

- 0: 禁止 PWM 时基周期匹配中断
- 1: 允许 PWM 时基周期匹配中断

BIT[6] PWMZIE - PWM 时基归零中断允许位

- 0: 禁止 PWM 时基归零中断
- 1: 允许 PWM 时基归零中断

BIT[5] PWMDDCI - PWM 时基计数器减计数时与占空比寄存器 PWM1CDL/H 匹配时中断使能位 (只在中心对齐模式下有效)

- 0: 禁止 PWM 计数器减计数时与 PWM1CDL/H 匹配时触发中断
- 1: 允许 PWM 计数器减计数时与 PWM1CDL/H 匹配时触发中断

BIT[4] PWMUDCI - PWM 时基计数器增计数时与占空比寄存器 PWM1CDL/H 匹配时中断使能位

- 0: 禁止 PWM 计数器增计数时与 PWM1CDL/H 匹配时触发中断
- 1: 允许 PWM 计数器增计数时与 PWM1CDL/H 匹配时触发中断

BIT[3] PWMDDBI - PWM 时基计数器减计数时与占空比寄存器 PWM1BDL/H 匹配时中断使能位 (只在中心对齐模式下有效)

- 0: 禁止 PWM 计数器减计数时与 PWM1BDL/H 匹配时触发中断
- 1: 允许 PWM 计数器减计数时与 PWM1BDL/H 匹配时触发中断

BIT[2] PWMUDBI - PWM 时基计数器增计数时与占空比寄存器 PWM1BDL/H 匹配时中断使能位

- 0: 禁止 PWM 计数器增计数时与 PWM1BDL/H 匹配时触发中断
- 1: 允许 PWM 计数器增计数时与 PWM1BDL/H 匹配时触发中断

BIT[1] PWMDDAI - PWM 时基计数器减计数时与占空比寄存器 PWM1ADL/H 匹配时中断使能位 (只在中心对齐模式下有效)



- 0: 禁止 PWM 计数器减计数时与 PWM1ADL/H 匹配时触发中断  
 1: 允许 PWM 计数器减计数时与 PWM1ADL/H 匹配时触发中断
- BIT[0] PWMUDAIE – PWM 时基计数器增计数时与占空比寄存器 PWM1ADL/H 匹配时中断使能位  
 0: 禁止 PWM 计数器增计数时与 PWM1ADL/H 匹配时触发中断  
 1: 允许 PWM 计数器增计数时与 PWM1ADL/H 匹配时触发中断

#### 10.10.17 PWM1 中断标志寄存器 (PWM1IF, 0xD8/SFR0)

SBIT 寄存器	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1IF	PWMPIF	PWMZIF	PWMDDCI F	PWMUDCI F	PWMDDBI F	PWMUDBI F	PWMDDAIF	PWMUDAI F
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

- BIT[7] PWMPIF – PWM 时基周期匹配中断标志位 (只在中心对齐模式下有效)  
 0: 无 PWM 时基周期匹配中断  
 1: 发生 PWM 时基周期匹配中断
- BIT[6] PWMZIF – PWM 时基归零中断标志位  
 0: 无 PWM 时基归零中断  
 1: 发生 PWM 时基归零中断
- BIT[5] PWMDDCIF – PWM 时基计数器减计数时与占空比寄存器 PWM1CDL/H 匹配时中断标志位 (只在中心对齐模式下有效)  
 0: 未发生中断或发生过但被 PWMUDCIF 信号清 0  
 1: 发生中断
- BIT[4] PWMUDCIF – PWM 时基计数器增计数时与占空比寄存器 PWM1CDL/H 匹配时中断标志位  
 0: 未发生中断或发生过但被 PWMDDCIF 信号清 0  
 1: 发生中断
- BIT[3] PWMDDBIF – PWM 时基计数器减计数时与占空比寄存器 PWM1BDL/H 匹配时中断标志位 (只在中心对齐模式下有效)  
 0: 未发生中断或发生过但被 PWMUDBIF 信号清 0  
 1: 发生中断
- BIT[2] PWMUDBIF – PWM 时基计数器增计数时与占空比寄存器 PWM1BDL/H 匹配时中断标志位  
 0: 未发生中断或发生过但被 PWMDDBIF 信号清 0.  
 1: 发生中断
- BIT[1] PWMDDAIF – PWM 时基计数器减计数时与占空比寄存器 PWM1ADL/H 匹配时中断标志位 (只在中心对齐模式下有效)  
 0: 未发生中断或发生过但被 PWMUDAIF 信号清 0.  
 1: 发生中断
- BIT[0] PWMUDAIF – PWM 时基计数器增计数时与占空比寄存器 PWM1ADL/H 匹配时中断标志位  
 0: 未发生中断或发生过但被 PWMDDAIF 信号清 0.  
 1: 发生中断

注 1: 即使 PWM 中断允许位为 0, 中断标志位也会置 1, 但中断不会响应。



注 2: PWMDDxIF 置 1 时, 硬件会自动将 PWMUDxIF 清零; 同样, PWMUDxIF 置 1 时, 硬件会自动将 PWMDDxIF 清零; 因此通过判断这两个标志位可以判断目前 PWM 波形处在有效状态还是无效状态。

注 3: 如果设置 PWM 周期过短 (例如: 2) 或向上与向下匹配接近 (例如: 周期为 100, 匹配为 99, 具体看实际效果), 增计数标志 PWMUDxIF 刚建立起来进入中断或唤醒后, 减计数也触发会导致 PWMDDxIF 建立并清除 PWMUDxIF 标志, 应用时请注意该情况发生

注 4: 由于 PWMDDxIF 和 PWMUDxIF 的互清机制, 所以 PWMDDxIE 和 PWMUDxIE 在应用时需要同时打开, 以保证唤醒或中断时一定有对应标志置起; 如若只打开一个中断, 由于两者是互清, 如果周期短或向上与向下匹配接近就会导致对应标志可能无法查询到

#### 10.10.18 PWM1 死区控制寄存器 0 (PWM1DT0, 0xFF4D/XSFR)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1DT0	PWMDT07	PWMDT06	PWMDT05	PWMDT04	PWMDT03	PWMDT02	PWMDT01	PWMDT00
R/W								
复位值	0	0	0	0	0	0	0	0

BIT[7:0] PWMDT0[7:0] – PWM1 死区控制寄存器 0 低位

#### 10.10.19 PWM1 寄存器修改重载解锁寄存器 (PWM1RLLEN, 0xD9/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1RLLEN	PWMRL7	PWMRL6	PWMRL5	PWMRL4	PWMRL3	PWMRL2	PWMRL1	PWMRL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] PWMRL[7:0] – 寄存器修改和重载控制寄存器

0x55: 允许软件对模块寄存器的修改

0xAA: 允许模块带缓冲的寄存器的重载

注 1: PWM 模块的寄存器, 除了中断标志寄存器 PWM1IF 和手动控制寄存器 PWMMANCR1 外, 其它寄存器的修改有限制条件, 只有当 PWM1RLLEN=0x55 时才允许软件修改, 否则修改无效。

注 2: PWM 模块中有一些寄存器带有缓冲寄存器, 包括占空比寄存器、周期寄存器。这些寄存器在软件修改后不会立即生效, 只有在归零或周期溢出时硬件重载, 才会真正生效。是否允许这些寄存器重载同样受控制, 只有当 PWM1RLLEN=0xAA 时才允许重载。



## 11 模数转换器 ADC

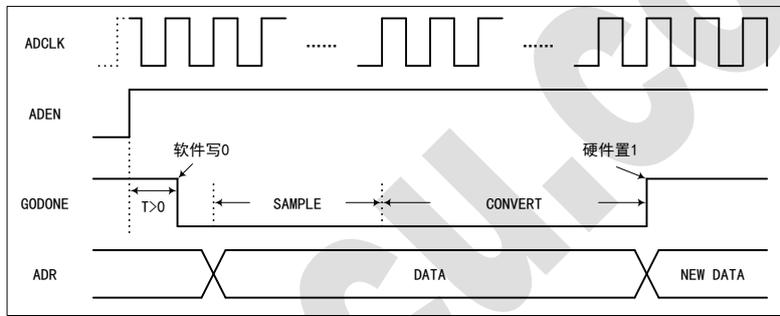
### 11.1 ADC 概述

ADC 模块为 12 位高精度逐次逼近型 ADC，包含外部 10 通道和内部 2 通道（VDD/4、GND）。ADC 时钟源为系统时钟  $F_{sys}$ ，可选择 1/2/4/8/16/32/64/128 分频；ADC 参考电压可配置：VDD、内部 2.0V、外部  $V_{ref}$  输入电压。

ADC 模块通过 ADEN 使能，通过 ADCHS 选择转换的模拟通道，ADCKS 选择转换速度，GODONE 为 ADC 启动位及转换结束标志位。当 GODONE 标志为 1 时，对该位写 0 将启动模数转换，转换时间有  $F_{sys}/1/2/4/8/32/64/128$  可选，采样周期有 3/4/8/16/32/64/128/256 可选，转换结果放在 ADRH/ADRL 中，GODONE 将自动置 1，同时中断标志 ADCIF 置 1；当 ADEOCIE=1 且 ADCIE=1 时，产生中断请求。

### 11.2 ADC 转换时序

ADC 转换时序如下图所示：



注：SAMPLE 时间可选 3/4/8/16/32/64/128/256 个 ADCLK，CONVERT 时间为 12 个 ADCLK。

### 11.3 ADC 常规操作步骤

AD 转换常规操作步骤：

- (1) 设置相应端口为输入端口，关闭上/下拉电阻；
- (2) 通过端口数模控制寄存器，关闭相应端口的数字 I/O 功能；
- (3) 若转换时钟可选，则设置 ADCKS，选择适当的 ADC 转换时钟；
- (4) 若采样时间可选，则设置 ADSPS，选择适当的 ADC 采样时间；
- (5) 若参考电压可选，则设置 ADVRS，选择适当的参考电压；
- (6) 若数据格式可选，则设置 ADRSEL，选择 ADC 转换结果的数据格式；
- (7) ADEN 置 1，使能 ADC 模块；
- (8) 设置 ADCHS，选择 ADC 转换通道；
- (9) 延时等待电路稳定后，GODONE 写 0，启动 AD 转换；
- (10) 等待 GODONE 硬件置 1（或利用 ADC 中断）；
- (11) 读取 ADC 转换结果（ADRH、ADRL）；
- (12) 重复执行（8）~（11），对不同的通道进行转换或对同一通道进行多次转换。

注 1：AD 转换过程中或者 ADEN 未使能时，ADRH/ADRL 中的数据未知，应在 ADEN 置 1 使能且 AD 转换结束 GODONE 硬件置 1 的情况下读取 AD 转换数据。

注 2：若 ADC 模块已经使能（ADEN 置 1），参考电压由 ADC 切换至内部参考电压 2.0V（设置  $VRS[1:0]=0x00$ ），



需要等待一定时间 ( $> 200\mu\text{s}$ )，参考电压稳定后，再能启动 ADC 转换 ( $GODONE = 0$ )。

注 3: 若 ADC 模块关闭 ( $ADEN$  清 0)，则 ADC 端口配置寄存器  $ADRIOS1$  中的使能位均自动失效，相应端口自动恢复为通用 IO 或其他复用端口。

## 11.4 ADC 比较功能

ADC 模块支持数字比较器功能， $ADCMPLH:ADCMPL$  寄存器设置比较值， $ADCMPEM$  位控制数字比较器使能， $ADCMPLD$  位控制数值比较关系。

设置  $ADCMPEM=1$ ，使能数字比较功能，如果  $ADCMPLD=0$ ，ADC 转换结果小于等于( $ADCMPLH:ADCMPL$ )值，则置位  $ADCCMPIF$ ；如果  $ADCMPLD=1$ ，ADC 转换结果大于( $ADCMPLH:ADCMPL$ )值，则置位  $ADCCMPIF$ 。

当  $ADCMPEM=1$  且  $ADCIE=1$  时，置位  $ADCCMPIF$  会引发 ADC 比较功能中断

## 11.5 ADC 连续采样

ADC 模块支持连续采样功能， $ADCONT$  位控制连续采样使能。

当通过软件或硬件触发 ADC 完成采样后（即  $GODONE$  自动置 1，同时中断标志  $ADCIF$  置 1），如果此时  $ADCONT=1$ ，ADC 模块将会自动开始新一轮采样，以此类推循环连续采样；直到  $ADCONT$  设置为 0 并完成当前 ADC 采样后才会停止采样。

注：当 ADC 连续采样进行时，将  $ADCEN$  关闭后再使能，连续采样不会恢复，需要进行一次新的软件或硬件触发才能重新开始连续采样

## 11.6 ADC 增益配置

ADC 增益可设置为  $1x$ 、 $0.75x$ 、 $0.5x$ 、 $0.25x$ 。在  $1x$  模式下，ADC 的输入全范围电压由  $VREF(2.0)$  决定。在其他模式下，当输入电压等于  $[VREF(2.0)/\text{增益}]$  时，达到全范围电压。

例如，在  $0.5x$  模式下，全范围输入电压为  $VREF(2.0) / 0.5 = 2 \times VREF(4.0)$ 。当选择较小的  $VREF$  电压时，较低的增益设置可以用于获得较高的输入电压范围，或者用于测量  $VREF$  和电源电压之间的输入电压。

ADC 的增益设置由  $ADCCR1$  的  $ADGAIN$  控制。

注：即使采用较低的增益设置，大于电源电压也不能由 ADC 直接测量。

## 11.7 ADC 相关寄存器

ADC 寄存器汇总表

地址 (SFRn /H)	寄存器 名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
SFR0/ FF	ADRH	( $ADRSEL=0$ ) ADC 转换结果	ADR11	ADR10	ADR9	ADR8	ADR7	ADR6	ADR5	ADR4	xxxx xxxx



		高位寄存器 (ADRSEL=1) ADC 转换结果 高位寄存器	-	-	-	-	ADR11	ADR10	ADR9	ADR8	
SFR0/ FE	ADRL	(ADRSEL=0) ADC 转换结果 低位寄存器	-	-	-	-	ADR3	ADR2	ADR1	ADR0	xxxx
		(ADRSEL=1) ADC 转换结果 低位寄存器	ADR7	ADR6	ADR5	ADR4	ADR3	ADR2	ADR1	ADR0	xxxx
SFR0/ FD	ADRIOS1	ADC 端口配置 寄存器 1	-	-	EX_VREF _EN	-	-	-	-	-	--0- ----
SFR0/ FB	ADCMPL	ADC 数字比较 高位寄存器	ADCMP1 5	ADCMP1 4	ADCMP1 3	ADCMP1 2	ADCMP1 1	ADCMP10	ADCMP9	ADCMP8	0000 0000
SFR0/ FA	ADCMPL	ADC 数字比较 低位寄存器	ADCMP7	ADCMP6	ADCMP5	ADCMP4	ADCMP3	ADCMP2	ADCMP1	ADCMP0	0000 0000
SFR0/ F5	OSADJCR	零点偏移修正 寄存器	OSADJEN	OSADJTD	OSADJT5	OSADJT4	OSADJT3	OSADJT2	OSADJT1	OSADJT 0	0xxx xxxx
SFR0/ F4	ADCCR3	ADC 控制寄存 器 3	ADGAIN1	ADGAIN0	-	-	-	-	-	ADVREF EN	00-- ---0
SFR0/ F3	ADCCR2	ADC 控制寄存 器 2	ADEOCIE	ADCMPIE	ADCMPE N	ADCMPI R	ADCONT	ADTRG2	ADTRG1	ADTRG0	0000 0000
SFR0/ F2	ADCCR1	ADC 控制寄存 器 1	ADRSEL	ADCKS2	ADCKS1	ADCKS0	-	ADSPS2	ADSPS1	ADSPS0	0000 -111
SFR0/ F1	ADCCR0	ADC 控制寄存 器 0	ADCHS3	ADCHS2	ADCHS1	ADCHS0	VRS1	VRS0	GODONE	ADEN	1111 1110
SFR0/ 88	TCON	定时器控制寄 存器	TF1	TR1	TF0	TR0	-	-	ADCCMPI F	ADCIF	0000 --00

11.7.1 ADC 控制寄存器 0 (ADCCR0, 0xF1/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCCR0	ADCHS3	ADCHS2	ADCHS1	ADCHS0	VRS1	VRS0	GODONE	ADEN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	1	1	1	1	1	1	0

BIT[7:4] ADCHS[3:0] – ADC 模拟通道选择

ADCHS[3:0]	ADC 模拟通道选择
0000	AIN0
0001	AIN1
0010	AIN2
0011	AIN3
0100	AIN4
0101	AIN5
0110	AIN6



0111	AIN7
1000	AIN8
1001	AIN9
1010	禁止
1011	禁止
1100	禁止
1101	内部 2.0V
1110	AN14 (内部接 VDD)
1111	AN15 (内部接地)

BIT[3:2] VRS[1:0] – ADC 参考电压选择

VRS[1:0]	ADC 参考电压
00	VDD
01	保留
10	保留
11	内部 2.0V

注：当模拟通道“ADCHS”选择“内部 2.0V”，“VRS”配置将无效，参考电压强制为 VDD。

BIT[1] GODONE – ADC 启动位及转换结束标志位

- 0: AD 启动或者 AD 正在转换中，转换结束后硬件自动置 1；
- 1: AD 转换已结束，同时作为 AD 转换完成的查询标志。

注：该位写入 0 时，启动一次 ADC 转换；当未转换完成时，再写入 0，则上次转换中止，开始一次新的转换；1 可以被写入，但其对 AD 模块无效，即不会影响 AD 转换结果。

BIT[0] ADEN – ADC 功能使能位

- 0: 关闭 ADC 模块；
- 1: 使能 ADC 模块。

### 11.7.2 ADC 控制寄存器 1 (ADCCR1, 0xF2/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCCR1	ADRSEL	ADCKS2	ADCKS1	ADCKS0	-	ADSPS2	ADSPS1	ADSPS0
R/W	R/W	R/W	R/W	R/W	-	R/W	R/W	R/W
复位值	0	0	0	0	-	1	1	1

BIT[7] ADRSEL – ADC 转换数据格式选择

- 0: ADC 转换结果 12 位数据，高 8 位存入 ADRH[7:0]、低 4 位存入 ADRL[3:0]；
- 1: ADC 转换结果 12 位数据，高 4 位存入 ADRH[3:0]、低 8 位存入 ADRL[7:0]。

BIT[6:4] ADCKS[2:0] – AD 转换时钟选择

ADCKS[2:0]	AD 转换时钟 $F_{ADC}$
000	$F_{sys}$
001	$F_{sys}/2$
010	$F_{sys}/4$
011	$F_{sys}/8$
100	$F_{sys}/16$



101	Fsys/32
110	Fsys/64
111	Fsys/128

BIT[3:0] ADSPS[3:0] – ADC 采样时间选择

ADSPS[2:0]	SAMPLE 时间
000	3 个 adcclk
001	4
010	8
011	16
100	32
101	64
110	128
111	256

### 11.7.3 ADC 控制寄存器 2 (ADCCR2, 0xF3/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCCR2	ADEOCIE	ADCMPIE	ADCPEN	ADCMPDIR	ADCONT	ADTRG2	ADTRG1	ADTRG0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7] ADEOCIE – ADC 转换完成中断使能位

0: 无效;

1: 使能。

BIT[6] ADCMPIE – ADC 比较功能中断使能位

0: 无效;

1: 使能。

BIT[5] ADCPEN – ADC 比较功能使能位

0: 无效;

1: 使能。

BIT[4] ADCMPDIR – ADC 比较功能数值大小关系选择位

0: 小于等于比较值;

1: 大于比较值。

BIT[3] ADCONT – ADC 连续采样功能使能位

0: 无效;

1: 使能。

BIT[2] ADTRG2 – PWM1 触发 ADC 启动使能位

0: 无效;

1: 使能。

BIT[1:0] ADTRG[1:0] – TIM0-2 触发 ADC 启动选择位

00: 任意 Timer 都不触发

01: Timer0 溢出时触发



10: Timer1 溢出时触发

11: Timer2 溢出时触发

注: ADTRG2 控制的 PWM1、ADTRG[1:0]控制的 Timer (3 选 1 或不选择) 触发这二类触发可以同时使能并有效;

#### 11.7.4 ADC 控制寄存器 3 (ADCCR3, 0xF4/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCCR3	ADGAIN1	ADGAIN0	-	-	-	-	-	ADVREFEN
R/W	R/W	R/W	-	-	-	-	-	R/W
复位值	0	0	-	-	-	-	-	0

BIT[7:6] ADGAIN[1:0] – ADC 输入电压增益选择

00: PGA 增益 1x

01: PGA 增益 0.75x

10: PGA 增益 0.5x

11: PGA 增益 0.25x

BIT[0] ADVREFEN – ADC VREF 使能位

0: 关闭;

1: 使能。

#### 11.7.5 ADC 端口配置寄存器 1 (ADRIOS1, 0xFD/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADRIOS1	-	-	EX_VREF_	-	-	-	-	-
R/W	-	-	R/W	-	-	-	-	-
复位值	-	-	0	-	-	-	-	-

BIT[5] EX\_VREF\_EN – ADC 外部 Vref 输入使能位

0: 禁止 ADC 外部 Vref 输入功能, 端口用于通用 IO 或其他复用模块;

1: 端口用作 ADC 外部 Vref 输入。

注: 当该位置 1 时, Vref 引脚输入有效, 且 ADC 模块强制将 Vref 电压作为 AD 的参考电压使用, 这时无视 ADC 参考电压选择位 (VRS[1:0]) 的值。

#### 11.7.6 定时器控制寄存器 (TCON, 0x88/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TCON	TF1	TR1	TF0	TR0	-	-	ADCCMPIF	ADCIF
R/W	R/W	R/W	R/W	R/W	-	-	R/W	R/W
复位值	0	0	0	0	-	-	0	0

BIT[1] ADCCMPIF – ADC 模拟比较生效中断标志位

0: 未发生 ADC 模拟比较中断, 由软件清 0;

1: 发生 ADC 模拟比较中断, 由硬件置 1。

BIT[0] ADCIF – ADC 中断标志位

0: 未发生 ADC 中断, 由软件清 0;

1: 发生 ADC 中断, 由硬件置 1。

注: 注: 此位默认值 1, 首次使用请先清零此位。



当 ADRSEL = 0 时:

### 11.7.7 ADC 转换结果高位寄存器 (ADRH, 0xFF/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADRH	ADR11	ADR10	ADR9	ADR8	ADR7	ADR6	ADR5	ADR4
R/W	R	R	R	R	R	R	R	R
复位值	X	X	X	X	X	X	X	X

BIT[7:0] ADR[11:4] – ADC 转换结果高 8 位

### 11.7.8 ADC 转换结果低位寄存器 (ADRL, 0xFE/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADRL	-	-	-	-	ADR3	ADR2	ADR1	ADR0
R/W	R	R	R	R	R	R	R	R
复位值	X	X	X	X	X	X	X	X

BIT[3:0] ADR[3:0] – ADC 转换结果低 4 位

当 ADRSEL = 1 时:

### 11.7.9 ADC 转换结果高位寄存器 (ADRH, 0xFF/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADRH	-	-	-	-	ADR11	ADR10	ADR9	ADR8
R/W	R	R	R	R	R	R	R	R
复位值	X	X	X	X	X	X	X	X

BIT[3:0] ADR[11:8] – ADC 转换结果高 4 位

### 11.7.10 ADC 转换结果低位寄存器 (ADRL, 0xFE/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADRL	ADR7	ADR6	ADR5	ADR4	ADR3	ADR2	ADR1	ADR0
R/W	R	R	R	R	R	R	R	R
复位值	X	X	X	X	X	X	X	X

BIT[7:0] ADR[7:0] – ADC 转换结果低 8 位

注: 芯片复位后, ADC 结果寄存器中的值为随机值。

### 11.7.11 零点偏移修正寄存器 (OSADJCR, 0xF5/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OSADJCR	OSADJEN	OSADJTD	OSADJT5	OSADJT4	OSADJT3	OSADJT2	OSADJT1	OSADJT0
R/W								
复位值	0	X	X	X	X	X	X	X

BIT[7] OSADJEN – ADC 零点偏移修调模式使能位, 复位值为出厂配置

0: 禁止 ADC 零点偏移修调模式 (默认);

1: 使能 ADC 零点偏移修调模式, 此模式下 OSADJTD & OSADJT[5:0] 的值将对 AD 的结果进行自动纠偏。



注：建议在使能AD零点偏移修调模式前，“用户”通过软件方式进行一次修调，以保证AD转换得到的结果最佳，具体步骤可以参考下面的说明。

BIT[6] OSADJTD – ADC 零点偏移修调方向选择位

0: 负向修调，即根据修调电压减小转换值（转换结果大于理论值时应选择负向修调）；

1: 正向修调，即根据修调电压增加转换值（转换结果小于理论值时应选择正向修调）。

BIT[5:0] OSADJT[5:0] – ADC 零点偏移修调电压选择位

OSADJT[5:0]	修调电压（典型值）
000000	0.5mV
000001	1.0mV
...	...
111110	30.5mV
111111	31.0mV

注：在ADC的修调结果中OSADJT[5:0]=000000B, OSADJTD=0或1, 两者效果是一样的，它们的偏移电压都是0mV；最大修调电压为±31.0mV。

#### 11.7.12 ADC 数字比较高位寄存器 (ADCMPH, 0xFB/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCMPGH	ADCMP15	ADCMP14	ADCMP13	ADCMP12	ADCMP11	ADCMP10	ADCMP9	ADCMP8
R/W	R/W	R/W						
复位值	0	0	0	0	0	0	0	0

BIT[3:0] ADCMP[15:8] – ADC 上限比较高 8 位

#### 11.7.13 ADC 数字比较低位寄存器 (ADCMPL, 0xFA/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCMPGL	ADCMP7	ADCMP6	ADCMP5	ADCMP4	ADCMP3	ADCMP2	ADCMP1	ADCMP0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] ADCMP[7:0] – ADC 上限比较低 8 位

### 11.8 ADC 零点偏移修调步骤

AD 模块零点偏移修调步骤：

- 将ADC的输入通道选择为内部接地（AN15），设置OSADJEN=1打开修调功能；设置一个基本的ADC时钟(建议ADC工作频率在1MHz为宜)、采样时间（建议ADC采样时间在8个Clk为宜）等；
- 设置OSADJTD=0, OSADJT[3:0]=0000B, 进行一次ADC转换；  
若ADC转换结果为0（实际值≤0），则跳至步骤（4），进行正向修调；  
若ADC转换结果不为0，则执行步骤（3）；
- 将OSADJT[3:0]的值，加1后，再次进行ADC转换；  
若ADC转换结果为0，则跳至步骤（6）；  
若ADC转换结果不为0，则循环执行步骤（3），直到ADC转换结果为0或OSADJT[3:0]递增到最大值1111B，跳至步骤（6）；



(4) 设置 OSADJTD =1, OSADJT[3:0]=1111B, 进行一次 ADC 转换;

若 ADC 转换结果为 0, 则跳至步骤 (6);

若 ADC 转换结果不为 0, 则执行步骤 (5);

(5) 将 OSADJT[3:0]的值, 减 1 后, 再次进行 ADC 转换;

若 ADC 转换结果为 0, 则跳至步骤 (6);

若 ADC 转换结果不为 0, 则循环执行步骤 (5), 直到 ADC 转换结果为 0 或 OSADJT[3:0]递减到 0000B, 跳至步骤 (6);

(6) OSADJTD 和 OSADJT[3:0]的值, 即为零点偏移最佳修调结果, 修调流程结束, 后续 ADC 工作时可直接使用该值, 不需要再次进行修调 (除非该值丢失, 如断电等情况)。

*注: 修调过程工作条件变化以及 ADC 有效位数, 可能影响修调的最佳效果, 用户可以通过增加单次步骤中的 ADC 采样次数或多次修调择优方式处理。*



## 12 低电压检测 LVD

### 12.1 概述

芯片内置低电压检测模块 LVD，可通过寄存器位 LVDEN 开启，通过 LVDVS 选择电压检测测量值。当 VDD 电压低于电压检测测量值时检测状态标志位 LVDF 将被置 1，同时中断标志 LVDIF 置 1 触发 LVD 中断；因 LVD 电路有回滞特性（回滞电压典型值为 6%），被检测电压需上升至电压检测测量值+6%后 LVDF 才被清 0（但中断标志不会自动清 0）。

LVD 具有独立的模拟滤波电路和数字滤波电路，可通过寄存器位 LVDAFLTE/LVDDFLTS 控制选择 LVD 输出的滤波时间。

注：开启 LVD、切换电压检测测量值等操作，需待电路稳定（时间>2ms）后 LVD 输出才有效，LVD 输出响应时间不超过 200 $\mu$ s。

### 12.2 LVD 相关寄存器

#### LVD 寄存器汇总表

地址 (SFR n/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
SFR0 /F7	LVDCR 1	LVD 控制寄存器 1	-	-	-	-	LVDDFL TS2	LVDDFL TS1	LVDDFL TS0	LVDAFL TE	---- 0000
SFR0 /F6	LVDCR	LVD 控制寄存器	LVDEN	-	LVDIF	-	LVDVS2	LVDVS1	LVDVS0	LVDF	0-0- 000-

#### 12.2.1 LVD 控制寄存器 (LVDCR, 0xF6/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LVDCR	LVDEN	-	LVDIF	-	LVDVS2	LVDVS1	LVDVS0	LVDF
R/W	R/W	-	R/W	-	R/W	R/W	R/W	R
复位值	0	-	0	-	0	0	0	X

BIT[7] LVDEN – 低电压检测 LVD 使能位

0: 关闭 LVD;

1: 开启 LVD;

BIT[5] LVDIF – LVD 中断标志位

0: 未触发 LVD 中断;

1: 已触发 LVD 中断，需软件清 0;

BIT[3:1] LVDVS[2:0] – LVD 电压检测测量值选择位

LVDVS[2:0]	LVD 电压检测测量值 (典型值)
000	4.0V
001	3.6V
010	3.2V
011	3.0V



100	2.8V
101	2.6V
110	2.4V
111	2.2V

BIT[0] LVDF – LVD 检测状态标志位

0: 被检测电压高于电压检测量值, 或 LVD 关闭;

1: 被检测电压低于电压检测量值;

### 12.2.2 LVD 控制寄存器 1 (LVDCR1, 0xF7/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LVDCR1	-	-	-	-	LVDDFLTS2	LVDDFLTS1	LVDDFLTS0	LVDAFLTE
R/W	-	-	-	-	R/W	R/W	R/W	R/W
复位值	-	-	-	-	0	0	0	0

BIT[2:1] LVDDFLTS[2:0] – LVD 数字滤波选择位

LVDDFLTS[2:0]	LVD 数字滤波时间选择
000	无数字滤波
001	16 个 F <sub>sys</sub> 时钟周期
010	32 个 F <sub>sys</sub> 时钟周期
011	64 个 F <sub>sys</sub> 时钟周期
100	2 个 FLIRC 时钟周期
101	4 个 FLIRC 时钟周期
110	6 个 FLIRC 时钟周期
111	8 个 FLIRC 时钟周期

注: 若选用 F<sub>sys</sub> 作为 LVD 数字滤波时钟源, 则需系统时钟源正常工作, 否则 LVD 将无法输出有效信号。

BIT[0] LVDAFLTE – LVD 模拟滤波使能位

0: LVD 输出无模拟滤波;

1: LVD 输出有模拟滤波;



## 13 增强型 UART

### 13.1 概述

片上包含 1 路 UART：

UART0 兼容 8051 标准的增强型通用异步收发器 UART，波特率可选择为系统时钟分频或者自带波特率发生器。

UART0 的增强功能包括帧出错检测以及自动地址识别，UART0 支持 8 位同步半双工（方式 0）、8 位异步全双工（方式 1）、9 位异步固定波特率（方式 2）、9 位异步可变波特率（方式 3）4 种工作方式。

### 13.2 UART0

UART0 有 4 种工作方式。在通讯之前必须先初始化串口控制寄存器 SCON，选择 UART 的工作方式和波特率。

在所有四种方式中，任何将串口缓冲寄存器 SBUF0 作为目标寄存器的写操作都会启动发送。在方式 0 中由条件 RI=0 和 REN=1 初始化接收，将在 TxD 引脚上产生 1 个时钟信号，然后在 RxD 引脚上串行移入/移出 8 位数据。在其他方式中则利用外部输入的起始位来初始化接收（如果 REN=1），外部发送器通过发送起始位开始通信。

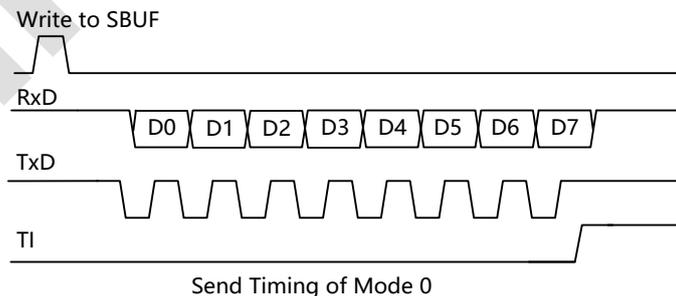
#### UART 方式列表

SM[0:1]	方式	类型	波特率	帧长度	起始位	停止位	第 9 位
00	0	同步	$F_{sys}/4$ 或 $F_{sys}/12$	8 位	无	无	无
01	1	异步	专用波特率发生器	10 位	1	1	无
10	2	异步	$F_{sys}/32$ 或 $F_{sys}/64$	11 位	1	1	0/1
11	3	异步	专用波特率发生器	11 位	1	1	0/1

#### 13.2.1 方式 0：8 位同步半双工

方式 0 支持与外部设备的同步通信。UART 通过 TxD 引脚发送移位时钟，在 RxD 引脚上收发串行数据。因此这个方式是串行通信的半双工方式，在此方式中，每帧收发 8 位，低位先接收/发送。

通过设置寄存器 SCON 中的 SM2 位为 0 或 1，波特率固定为系统时钟的 1/12 或 1/4。当 SM2 位为 0 时，串行端口以系统时钟的 1/12 运行。当 SM2 位置 1 时，串行端口以系统时钟的 1/4 运行。与标准 8051 唯一不同的是，芯片在方式 0 中有可变波特率。

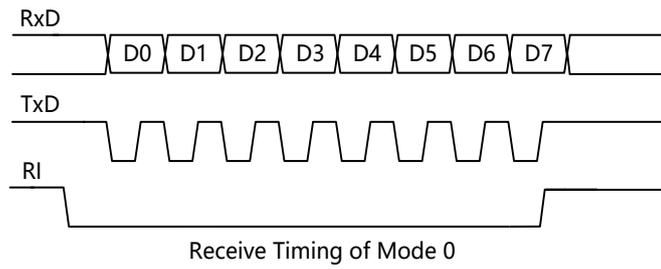


UART 模块通过 TxD 引脚输出同步时钟，通过 RxD 引脚将数据读入或移出串行端口。任何将 SBUF0 作为目标寄存器的写操作都会启动发送。下一个系统时钟 Tx 控制块开始发送。数据转换发生在移位时钟的上升沿，移位寄存器的内容逐次从左往右移位，空位置 0。当移位寄存器中的所有 8 位都发送后，TX 控制模块停止发送操作，然后在下一个系统时钟的上升沿将 TI 置 1。

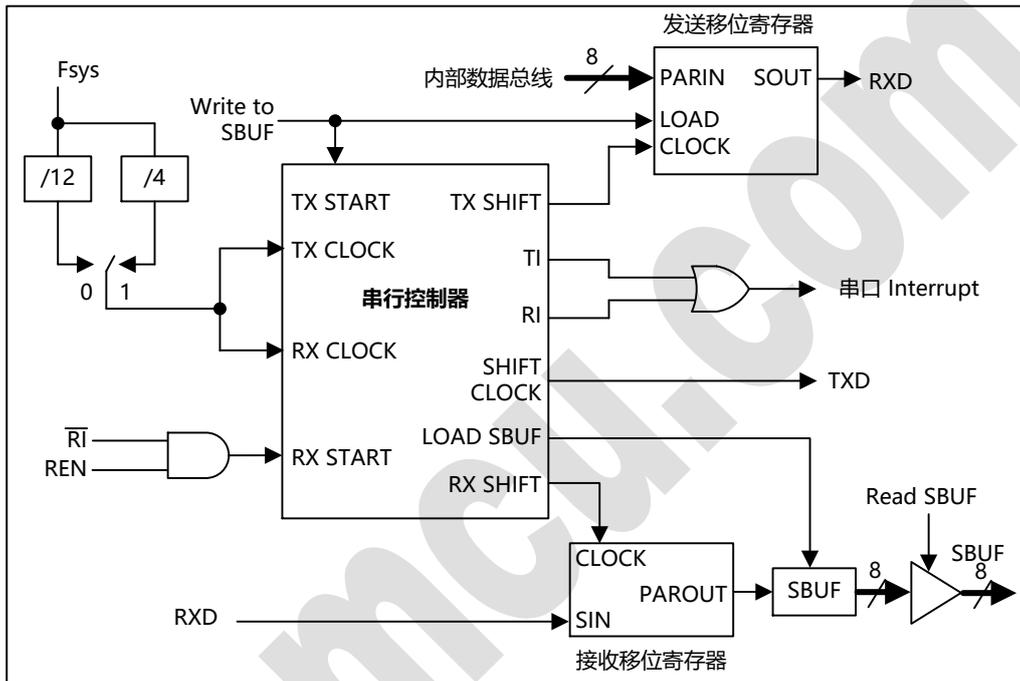
REN 位置 1 和 RI 位清 0 将初始化接收。下一个系统时钟启动接收，在移位时钟的上升沿锁存数据，接收转换寄存器的内容逐次向左移位。当所有 8 位都接收到接收移位寄存器中后，RX 控制模块停止接收，然后在下一个系统时钟的上



升沿上 RI 置 1，直到被软件清 0 才允许接收。



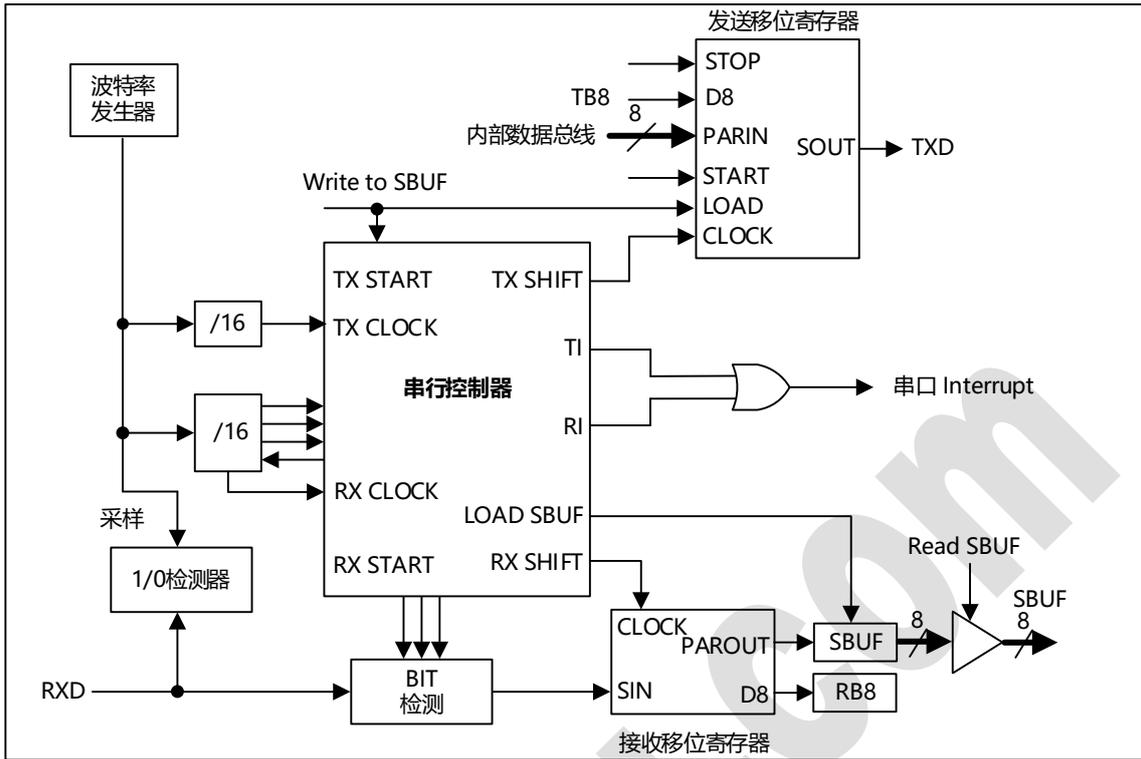
方式 0 功能块框图如下图所示：



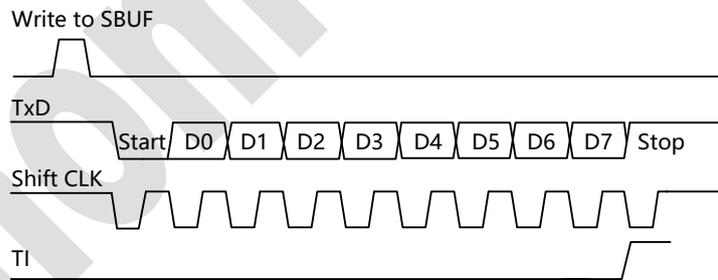
### 13.2.2 方式 1: 8 位异步全双工，可变波特率

方式 1 提供 10 位全双工异步通信，10 位由一个起始位（逻辑 0）、8 个数据位（低位在前）和一个停止位（逻辑 1）组成。在接收时，这 8 个数据位存储在 SBUF0 中而停止位储存在 RB8 位中。方式 1 中的波特率是可变的，串行收发波特率为自带波特率发生器产生。

方式 1 功能块框图如下图所示：



任何将 SBUF0 作为目标寄存器的写操作都会启动发送，实际上发送是从 16 分频计数器中的下一次跳变之后的系统时钟开始的，因此位时间与 16 分频计数器是同步的，与对 SBUF0 的写操作不同步。起始位首先在 TxD 引脚上移出，然后是 8 位数据位。在发送移位寄存器中的所有 8 位数据都发送完后，停止位在 TxD 引脚上移出，在停止位发出的同时 TI 标志置 1。

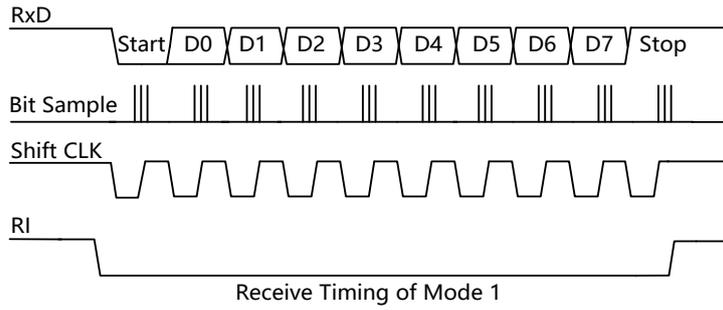


Send Timing of Mode 1

只有 REN 位置 1 时才允许接收。当 RxD 引脚检测到下降沿时串行口开始接收串行数据。为此，芯片对 RxD 不断采样，采样速率为波特率的 16 倍。当检测下降沿时，16 分频计数器立即复位，这有助于 16 分频计数器与 RxD 引脚上的串行数据位同步。16 分频计数器把每一位的时间分为 16 个状态，在第 7、8、9 状态时，位检测器对 RxD 端的电平进行采样。为抑制噪声，在这 3 个状态采样中至少有 2 次采样值一致数据才被接收。如果所接收的第一位不是 0，说明这位不是一帧数据的起始位，该位被忽略，接收电路被复位，等待 RxD 引脚上另一个下降沿的到来。若起始位有效，则移入移位寄存器，并接着移入其它位到移位寄存器。8 个数据位和 1 个停止位移入之后，移位寄存器的内容被分别装入 SBUF0 和 RB8 中，RI 置 1，但必须满足下列条件：

- (1) RI=0;
- (2) SM2=0, 或者接收的停止位=1;

如果这些条件被满足，那么停止位移入 RB8，8 个数据位移入 SBUF0，RI 被置 1。否则接收的帧会丢失。这时，接收器将重新去探测 RxD 端是否另一个下降沿。用户必须用软件清除 RI，然后才能再次接收。

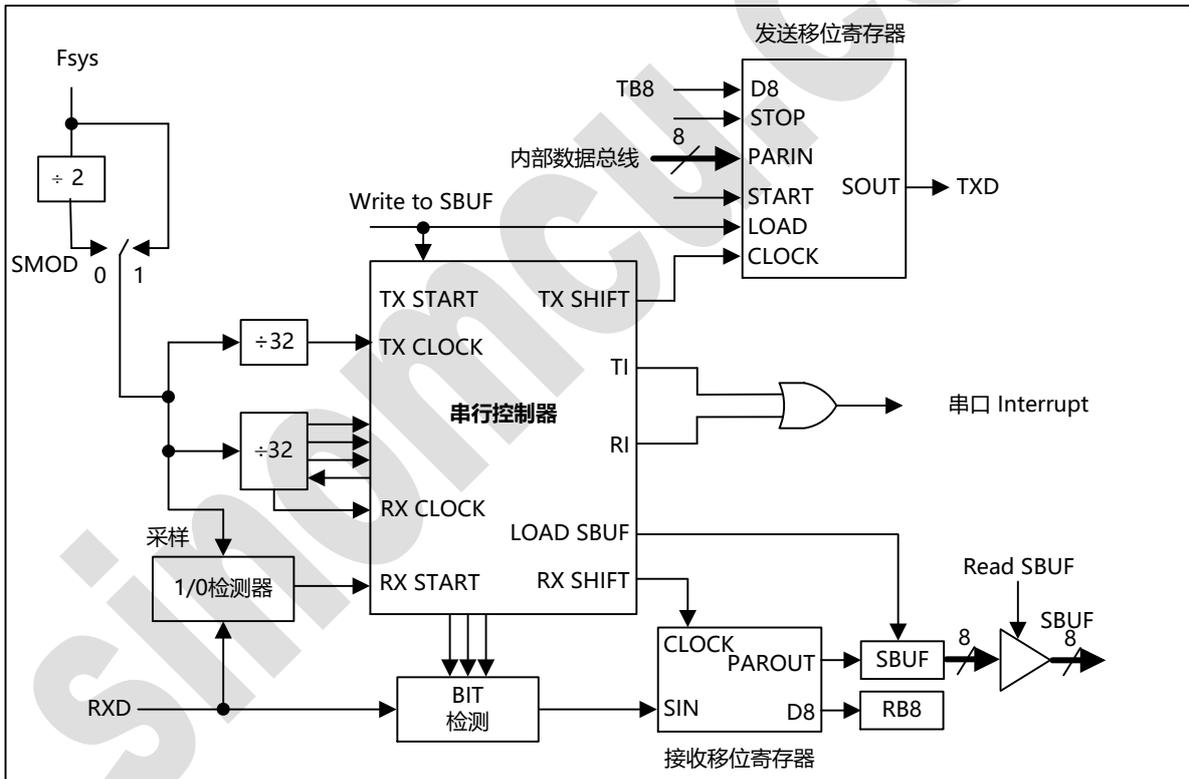


Receive Timing of Mode 1

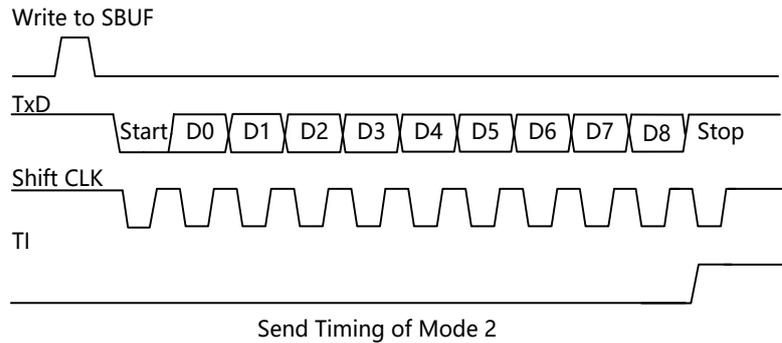
### 13.2.3 方式 2：9 位异步全双工，固定波特率

这个方式使用异步全双工通信中的 11 位。一帧由一个起始位（逻辑 0），8 个数据位（低位在前），一个可编程的第 9 数据位和一个停止位（逻辑 1）组成。方式 2 支持多机通信和硬件地址识别（详见多机通讯章节）。在数据传送时，第 9 数据位（SCON 中的 TB8）可以写 0 或 1，例如，可写入 PSW 中的奇偶位 P，或用作多机通信中的数据/地址标志位。当接收到数据时，第 9 数据位进入 RB8 而停止位不保存。

方式 2 功能块框图如下所示：



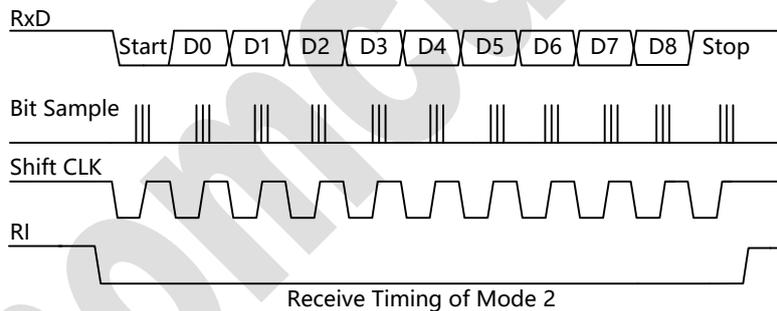
任何将 SBUF0 作为目标寄存器的写操作都会启动发送，同时也将 TB8 载入到发送移位寄存器的第 9 位中。实际上发送是从 16 分频计数器中的下一次跳变之后的系统时钟开始的，因此时间与 16 分频计数器是同步的，与对 SBUF0 的写操作不同步。起始位首先在 TxD 引脚上移出，然后是第 9 位数据。在发送转换寄存器中的所有 9 位数据都发送完后，停止位在 TxD 引脚上移出，在停止位开始发送时 TI 标志置 1。



只有 REN 位置 1 时才允许接收。当 RxD 引脚检测到下降沿时串行口开始接收串行数据。为此，芯片对 RxD 不断采样，采样速率为波特率的 16 倍。当检测下降沿时，16 分频计数器立即复位。这有助于 16 分频计数器与 RxD 引脚上的串行数据位同步。16 分频计数器把每一位的时间分为 16 个状态，在第 7、8、9 状态时，位检测器对 RxD 端的电平进行采样。为抑制噪声，在这 3 个状态采样中至少有 2 次采样值一致数据才被接收。如果所接收的第一位不是 0，说明这位不是一帧数据的起始位，该位被忽略，接收电路被复位，等待 RxD 引脚上另一个下降沿的到来。若起始位有效，则移入移位寄存器，并接着移入其它位到移位寄存器。9 个数据位和 1 个停止位移入之后，移位寄存器的内容被分别装入 SBUF0 和 RB8 中，RI 置 1，但必须满足下列条件：

- (1) RI=0;
- (2) SM2=0 或者接收的第 9 位=1，且接收的字节符合实际从机地址；

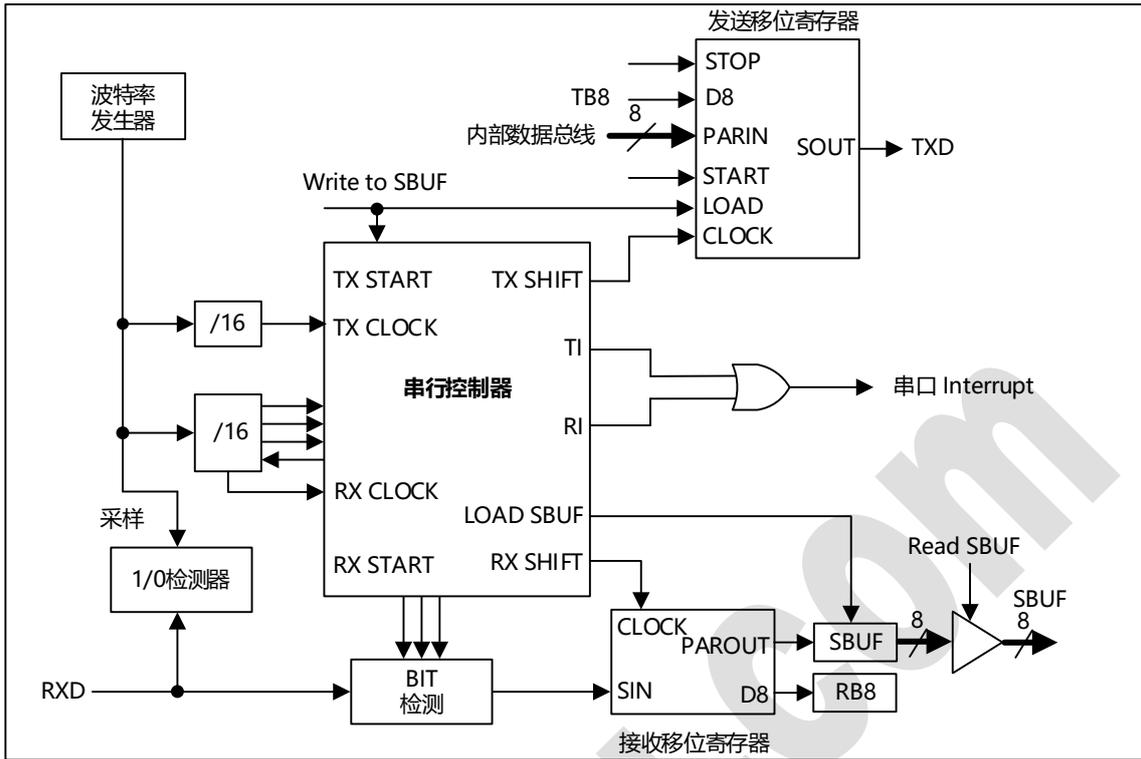
如果这些条件被满足，那么第 9 位移入 RB8，8 位数据移入 SBUF0，RI 被置 1。否则接收的数据帧会丢失。在停止位的当中，接收器回到寻找 RxD 引脚上的另一个下降沿。用户必须用软件清除 RI，然后才能再次接收。



### 13.2.4 方式 3：9 位异步全双工，可变波特率

方式 3 使用方式 2 的传输协议以及方式 1 的波特率产生方式。

方式 3 功能块框图如下所示：



### 13.2.5 波特率计算

#### UART 方式 0

波特率可编程为系统时钟的 1/12 或 1/4，由 SM2 位决定。当 SM2 为 0 时，串行端口在系统时钟的 1/12 下运行；当 SM2 为 1 时，串行端口在系统时钟的 1/4 下运行。

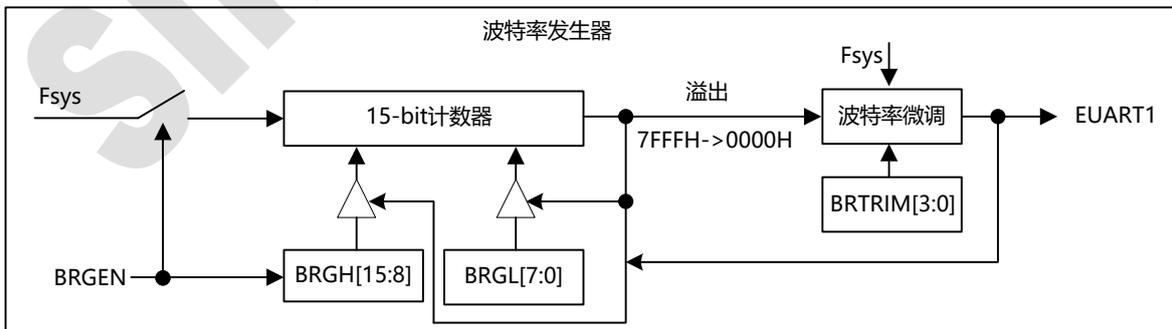
$$\text{UART0波特率} = \frac{F_{\text{sys}}}{4}$$

或者

$$\text{UART0波特率} = \frac{F_{\text{sys}}}{12}$$

#### UART 方式 1 和方式 3

自带波特率发生器且波特率可微调，精度为一个系统时钟。



波特率公式如下：

$$\text{波特率} = \frac{F_{\text{sys}}}{16 \times (65536 - [BRGH, BRGL]) + BRTRIM}$$

举例：Fsys = 12MHz，计算 115200bps 波特率如下：

$$12000000/16/115200 = 6.51$$



[BRGH, BRGL] = 65536 - 6 = 65530

由波特率公式,  $115200 = 12000000 / (16 \times 6 + \text{BRTRIM})$ , 得  $\text{BRTRIM} = 8.2$ , 取整 8。

实际波特率 =  $12000000 / (16 \times 6 + 8) = 115384.6\text{bps}$ , 误差 0.16%。

## UART 方式 2

UART0 的波特率为:

$$\text{UART0 波特率} = \frac{F_{\text{sys}}}{64} \times 2^{\text{SMOD0}}$$

其中 SMOD0 位根据用户设定值可以取 0 或者 1, 起到倍频的作用。

### 13.2.6 多机通讯

#### 软件地址识别

方式 2 和方式 3 有一个专门的适用于多机通讯的功能。在这两个方式下, 接收的是 9 位数据, 第 9 位移入 RB8 中, 然后再来一位停止位。UART 可以这样设定: 当接收到停止位时, 只有在 RB8=1 的条件下, 串行口中断才会有效 (请求标志 RI 置 1)。可通过将 SCON 寄存器的 SM2 位置 1 使 UART 具有这个功能。

在多机通讯系统中, 以如下所述来利用这一功能。当主机要发送一数据块给几个从机中的一个时, 先送出一地址字节, 以辨认目标从机。地址字节与数据字节可用第 9 数据位来区别, 地址字节的第 9 位为 1, 数据字节的第 9 位为 0。

如果从机 SM2 为 1, 则不会响应数据字节中断。地址字节可以中断所有从机, 这样, 每一个从机都检查所接收到的地址字节, 以判别自己是不是目标从机。被寻到的从机将 SM2 位清 0, 并准备接收即将到来的数据字节, 当接收完毕时, 从机再一次将 SM2 置 1。没有被寻址的从机, 则维持其 SM2 位为 1, 忽略到来的数据字节, 继续做自己的事情。

*注: 在方式 0 中, SM2 用来选择波特率加倍。在方式 1 中, SM2 用来检测停止位是否有效, 如果 SM2 = 1, 接收中断不会响应直到接收到一个有效的停止位。*

#### 自动 (硬件) 地址识别

在方式 2 和方式 3 中, SM2 置 1 将使 UART 在如下状态运行: 当 1 个停止位被接收时, 如果载入 RB8 的第 9 数据位为 1 (地址字节) 并且接收到的数据字节符合 UART 的从机地址, UART 产生一个中断。接着, 从机应将 SM2 清零, 以接收后续的数据字节。

在 9 位方式下要求第 9 位为 1 以表明该字节是地址而非数据。当主机要发送一组数据给几个从机中的一个时, 必须先发送目标从机的地址。所有从机在等待接收地址字节时, 为了确保仅在接收地址字节时产生中断, SM2 位必须置 1。自动地址识别的特点是只有地址匹配的从机才能产生中断, 地址比较通过硬件完成而不是软件。

中断产生后, 地址相匹配的从机清零 SM2, 继续接收数据字节。地址不匹配的从机不受影响, 将继续等待接收和它匹配的地址字节。一旦全部信息接收完毕, 地址匹配的从机应该再次把 SM2 置 1, 忽略所有传送的非地址字节, 直到接收到下一个地址字节。

使用自动地址识别功能时, 主机可以通过调用给定的从机地址选择与一个或多个从机通信。使用广播地址可以联系所有的从机。有两个特殊功能寄存器用来定义从机地址 (SADDR) 和地址屏蔽 (SADEN)。从机地址是一个 8 位的字节, 存于 SADDR 寄存器中。SADEN 用于定义 SADDR 内位的有效与否, 如果 SADEN 中某一位为 0, 则 SADDR 中相应位的被忽略, 如果 SADEN 中某一位置 1, 则 SADDR 中相应位的将用于得到给定的从机地址。这可以使用户在不改变 SADDR 寄存器中的从机地址的情况下灵活地寻址多个从机。使用给定地址可以识别多个从机而排除其他的从机。

	从机 1	从机 2
SADDR	10100100	10100111
SADEN (为 0 的位被忽略)	11111010	11111001
实际从机地址	10100x0x	10100xx1
广播地址 (SADDR 或 SADEN)	1111111x	11111111



从机 1 和从机 2 给定地址的最低位是不同的。从机 1 忽略了最低位，而从机 2 的最低位是 1。因此只与从机 1 通讯时，主机必须发送最低位为 0 的地址（10100000）。类似地，从机 1 的第 1 位为 0，从机 2 的第 1 位被忽略。因此，只与从机 2 通讯时，主机必须发送第 1 位为 1 的地址（10100011）。如果主机希望同时与两从机通讯，则第 0 位为 1，第 1 位为 0，第 2 位被两从机都忽略，此时有两个不同的地址用于选定两个从机（1010 0001 和 1010 0101）。

主机可以通过广播地址与所有从机同时通讯。这个地址等于 SADDR 和 SADEN 的逻辑或，结果中的 0 表示该位被忽略。多数情况下，广播地址为 0xFFh，该地址可被所有从机应答。

系统复位后，SADDR 和 SADEN 两个寄存器初始化为 0，这两个结果设定了给定地址和广播地址为 XXXXXXXX（所有位都被忽略）。这有效地去除了多处机通讯的特性，禁止了自动寻址方式。这样的 UART 将对任何地址都产生应答，兼容了不支持自动地址识别的 8051 控制器。用户可以按照上面提到的方法实现软件识别地址的多机通讯。

### 13.2.7 帧出错检测

当寄存器 PCON 中的 SSTAT 位为逻辑 1 时，帧出错检测功能才有效。3 个错误标志位被置 1 后，只能通过软件清零，尽管后续接收的帧没有任何错误也不会自动清零。

注：SSTAT=1 时是访问状态位（FE，RXOV 和 TXCOL），SSTAT=0 时是访问方式选择位（SM0，SM1 和 SM2）。  
发送冲突

当一个发送正在进行，而此时软件写数据到 SBUF0 寄存器，则发送冲突位 TXCOL 置 1。如果发生了冲突，新数据会被忽略，不会被写入发送缓冲器。

接收溢出

在接收缓冲器中数据未被读取之前，RI 被清 0，此时如果又有新的数据存入接收缓冲器，则接收溢出位 RXOV 位置 1。如果发生了接收溢出，接收缓冲器中原来的数据将丢失。

帧出错

如果检测到一个无效（低）停止位，则帧出错位 FE 置 1。

暂停检测

当连续检测到 11 个位都为低电平时时，则认为检测到一个暂停。由于暂停条件同样满足帧错误条件，因此检测到暂停时也会报告帧错误。一旦检测到暂停条件，UART 将进入空闲状态并一直保持，直至接收到有效停止位（RXD 引脚上出现上升沿）。

## 13.3 UART 相关寄存器

UART 寄存器汇总表

地址 (SFR n/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位 值 (B)
SFR0 /9F	BRGH	串口 0 波特率 计数器高位寄 存器	BRG15	BRG14	BRG13	BRG12	BRG11	BRG10	BRG9	BRG8	0000 000
SFR0 /9E	BRGL	串口 0 波特率 计数器低位寄 存器	BRG7	BRG6	BRG5	BRG4	BRG3	BRG2	BRG1	BRG0	0000 000
SFR0	BRGCR	串口 0 波特率	BRGEN				BRTRI	BRTRI	BRTRI	BRTRI	0---



/9D		控制寄存器					M3	M2	M1	M0	0000
SFR0 /9C	SCON2	串口 0 控制寄存器 2	-	-	-	-	TEN	PINSEL 2	PINSEL 1	PINSEL 0	---- 0000
SFR0 /9B	SADEN	串口 0 地址掩码寄存器	SADEN 7	SADEN 6	SADEN 5	SADEN 4	SADEN 3	SADEN 2	SADEN 1	SADEN 0	0000 0000
SFR0 /9A	SADDR	串口 0 从机地址寄存器	SADDR 7	SADDR 6	SADDR 5	SADDR 4	SADDR 3	SADDR 2	SADDR 1	SADDR 0	00000 000
SFR0 /99	SBUF0	串口 0 缓存寄存器	SBUF7	SBUF6	SBUF5	SBUF4	SBUF3	SBUF2	SBUF1	SBUF0	xxxx xxxx
SFR0 /98	SCON0	串口 0 控制寄存器	SM0/FE	SM1/RX OV	SM2/TX COL	REN	TB8	RB8	TI	RI	00000 000
SFR0 /87	PCON	电源控制寄存器	-	-	-	UART0 EN	SMOD0	SSTAT0	SLEEP	STOP	0--0 0100

### 13.3.1 电源控制寄存器 (PCON, 0x87/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCON	-	-	-	UART0EN	SMOD0	SSTAT0	SLEEP	STOP
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
复位值	-	-	-	0	0	0	0	0

BIT[4] UART0EN – UART0 模块使能位

- 0: 关闭 UART0 模块，端口用于通用 IO 或其他复用模块；
- 1: 打开 UART0 模块。

BIT[3] SMOD0 – UART0 波特率倍增位

- 0: UART0 波特率保持现状；
- 1: UART0 波特率增加一倍。

BIT[2] SSTAT0 – UART0 寄存器功能选择位

- 0: 寄存器 SCON[7:5]功能为 SM[0:2]；
- 1: 寄存器 SCON[7:5]功能为 FE, RXOV, TXCOL。

*特别注意：上述控制位 (PCON[7:2]) 皆不受 SLPCR 保护寄存器的影响，可以直接读/写。*

### 13.3.2 串口 0 控制寄存器 (SCON0, 0x98/SFR0)

SBIT 寄存器	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SCON0	SM0/FE	SM1/RXOV	SM2/TXCO L	REN	TB8	RB8	TI	RI
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

UART0 在 SSTAT=0 时:

BIT[7:6] SM[1:0] – UART 方式控制位 (SSTAT=0)

SM[0:1]	UART 工作方式	适用
00	方式 0: 同步方式，固定波特率	UART0
01	方式 1: 8 位异步方式，可变波特率	UART0
10	方式 2: 9 位异步方式，固定波特率	UART0



11	方式 3: 9 位异步方式, 可变波特率	UART0
----	----------------------	-------

BIT[5] SM2 或者 SM2\_1 – UART 功能设定位 (SSTAT=0)

SM2	方式 0	方式 1	方式 2/3
0	波特率=Fsys/12	禁止停止位确认检验, 停止位将置 RI 为 1 产生中断	任何字节均会置 RI 为 1 产生中断
1	波特率=Fsys/4	允许停止位确认检验, 只有有效的停止位(1)才能置 RI 为 1 产生中断	只有寻址字节(第 9 位=1)能置 RI 为 1 产生中断

UART0 在 SSTAT=1 时:

BIT[7] FE – 帧出错标志位 (SSTAT=1)

- 0: 无帧出错, 由软件清 0;
- 1: 发生帧出错, 由硬件置 1。

BIT[6] RXOV – 接收溢出标志位 (SSTAT=1)

- 0: 无接收溢出, 由软件清 0;
- 1: 接收溢出, 由硬件置 1。

BIT[5] TXCOL – 发送冲突标志位 (SSTAT=1)

- 0: 无发送冲突, 由软件清 0;
- 1: 有发送冲突, 由硬件置 1。

注: 以上 3 个控制位 (Bit[7-5]), 因模块本身的差异, 仅在 UART0 中存在, UART1 中并不存在。

BIT[4] REN – 接收器允许位

- 0: 禁止接收;
- 1: 允许接收。

BIT[3] TB8 – 方式 2/3 时发送数据的第 9 位

- 0: 方式 2/3 发送数据的第 9 位为 0;
- 1: 方式 2/3 发送数据的第 9 位为 1。

BIT[2] RB8 – 方式 1/2/3 时接收数据的第 9 位 (停止位或数据位)

- 0: 方式 1/2/3 时接收数据的第 9 位为 0;
- 1: 方式 1/2/3 时接收数据的第 9 位为 1。

BIT[1] TI – 发送中断标志位

- 0: 由软件清 0;
- 1: 在方式 0 的第 8 位最后, 或在其他方式的停止位开始, 由硬件置 1。

BIT[0] RI – 接收中断标志位

- 0: 由软件清 0;
- 1: 在方式 0 的第 8 位最后, 或在其他方式的停止位开始, 由硬件置 1。

### 13.3.3 串口 0 缓存寄存器 (SBUF0, 0x99/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SBUF0	SBUF7	SBUF6	SBUF5	SBUF4	SBUF3	SBUF2	SBUF1	SBUF0
R/W								
复位值	X	X	X	X	X	X	X	X

BIT[7:0] SBUF[7:0] – UART 数据缓存, SBUF0 访问两个寄存器: 1 个移位寄存器和 1 个接收锁存寄存器。SBUF0



的写入将发送字节到移位寄存器中，然后开始从端口发送；SBUF0 的读取将返回接收锁存寄存器中的内容

注：UART0 的发送和接收用缓存，使用了同一个缓存寄存器，写入时是一个通道，读取时是另一个通道，故用户编写软件中需要注意。

### 13.3.4 串口 0 从机地址寄存器 (SADDR, 0x9A/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SADDR	SADDR7	SADDR6	SADDR5	SADDR4	SADDR3	SADDR2	SADDR1	SADDR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] SADDR[7:0] – UART 的从机地址

### 13.3.5 串口 0 地址掩码寄存器 (SADEN, 0x9B/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SADEN	SADEN7	SADEN6	SADEN5	SADEN4	SADEN3	SADEN2	SADEN1	SADEN0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] SADENn – SADDRn 检验控制位 (n=7-0)

0: 忽略 SADDRn 位;

1: 检验 SADDRn 位是否对应接收地址。

### 13.3.6 串口 0 波特率控制寄存器 (BRGCR, 0x9D/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
BRGCR	BRGEN	-	-	-	BRTRIM3	BRTRIM2	BRTRIM1	BRTRIM0
R/W	R/W	-	-	-	R/W	R/W	R/W	R/W
复位值	0	-	-	-	0	0	0	0

BIT[7] BRGEN – 串口 0 波特率发生器使能控制位

0: 关闭;

1: 使能;

BIT[3:0] BRTRIM[3:0] – 串口 1 波特率微调控制位

波特率微调，单位为系统时钟 F<sub>sys clock</sub>，设置为 0 表示不做微调。

### 13.3.7 串口 0 波特率计数器低位寄存器 (BRGH, 0x9F/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
BRGH	BRG15	BRG14	BRG13	BRG12	BRG11	BRG10	BRG9	BRG8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

### 13.3.8 串口 0 波特率计数器高位寄存器 (BRGL, 0x9E/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
BRGL	BRG7	BRG6	BRG5	BRG4	BRG3	BRG2	BRG1	BRG0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0



BIT[7:0] BRG[15:0] – 串口 1 波特率计数器控制寄存器

### 13.3.9 串口 0 控制寄存器 2 (SCON2, 0x9C/SFR)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SCON2	-	-	-	-	TEN	PINSEL2	PINSEL1	PINSEL0
R/W	-	-	-	-	R/W	R/W	R/W	R/W
复位值	-	-	-	-	0	0	0	0

BIT[3] TEN – 发送使能位

0: TX0 引脚无效;

1: TX0 引脚有效。

BIT[2:0] PINSEL[2:0] – TX0/RX0 管脚选择位

000: TX0 选择 P2.3, RX0 选择 P2.2;

001: TX0 选择 P1.5, RX0 选择 P1.6;

010: TX0 选择 P1.2, RX0 选择 P1.1;

011: TX0 选择 P2.0, RX0 选择 P1.1 (仅 20 pin 封装支持);

1XX: TX0 选择 P0.0, RX0 选择 P0.1。



## 14 串行外设接口 SPI

### 14.1 SPI 概述

片上 SPI 支持全双工，3 线/4 线同步传输，主从机操作模式可选，LSB/MSB 优先可选，带芯片中断主模式故障出错标志，带写入冲突标志保护等功能。

串行外部设备接口（SPI）是一种高速串行通信接口，允许芯片与外部设备（包括芯片）进行全双工同步串行信号通信。

### 14.2 SPI 信号描述

主输入从输出（MOSI）：该信号表示主设备、从设备之间，数据流经主设备串行发送到从设备，主设备输出，从设备输入。

主输入从输出（MISO）：该信号表示主设备、从设备之间，数据流经从设备串行发送到主设备，从设备输出，主设备输入。注意：在 4 线 SPI 通信中从设备必须被选中，方可发送数据；若未被选中 MISO 引脚将处于高阻状态。

串行时钟（SCLK）：该信号用作控制 MOSI 和 MISO 线上输入出数据的同步移动，每 8 个时钟周期 MOSI 和 MISO 线上传送一个字节，如果从设备未被选中 SCLK 信号将被此设备忽略。注意：只有主设备才能产生 SCLK 信号。

从设备选择信号（SS）：当做从设备使用时，每一个 SPI 从设备都带有一个片选信号，该信号低电平有效，当引脚信号由高到低反转时表明该从设备被选中，开始与 SPI 主设备进行通信。为了防止 SPI 通信冲突，同一时刻仅允许一个主设备与从设备通信。当做主设备使用时，该引脚可另作他用，从设备设备时交给 SPI 主设备控制。

### 14.3 工作模式

片上的 SPI 在工作中，仅可配置为主/从模式中的一种，对应于 3 线/4 线 SPI 模式，参考做法如下：

通信模式	本机 SPI 设备	本机 SPI 的 SS 脚	其他 SPI 设备	其他 SPI 设备 SS 脚
一主一从	主模式	可另作他用	从模式	拉低（片选信号始终有效）
一主多从				通过 GPIO 控制，通信时分别拉低（片选有效），切换不同从设备
一从一主	从模式	由其他 SPI 设备控制是否拉低（片选有效）	主模式	可另作他用

#### 14.3.1 主机模式

当 SPCON 寄存器的 MSTR 位置 1 时，SPI 模块工作在主机模式。整个 SPI 系统中只允许一个主机启动传输，每次传输总是由主机发起。

在 SPI 作为主机模式中，向 SPI 的数据寄存器 SPDAT 写入数据后，数据将会写入发送移位缓冲器。如果发送移位寄存器已经存在一个数据，那么主 SPI 产生一个 WCOL 信号以表明写入太快。但是发送移位寄存器中的数据不会受到影响，发送也不会中断。另外如果发送移位寄存器为空，那么主设备立即按照 SCLK 的时钟频率串行地移出发送寄存器中的数据到 MOSI 上。当传送完毕，SPSTA 寄存器中的 SPIIF 位置 1。如果 SPI 中断被允许，当 SPIIF 位置 1，会产生一个中断请求。

当主设备通过 MOSI 传送数据给从设备时，相应的从设备同时也通过 MISO 将其发送移位寄存器的内容传送给主设



备的接收移位寄存器，实现全双工。因此，SPIIF 标志位置 1 既表示数据发送完成，又表示接收完成。用户可以从 SPDAT 寄存器中读出接收的数据，并清除 SPIIF。

### 14.3.2 从机模式

当 SPCON 寄存器的 MSTR 位清 0 时，SPI 模块工作在从机模式。在数据传送之前，从设备的 SS 引脚必须拉低，且必须保持低电平直到一个字节数据传送完毕。当从机设备的 SS 管脚不为低时，主机设备不能与从机交换数据。在数据传输开始前和数据传输完成前，SS 管脚都需要保持低电平状态。如果 SS 变为高电平，SPI 模块将被迫进入闲置状态。如果 SS 管脚在传输的过程被置高，那么传输将被取消，接收移位缓存区里剩下的位数将变高，同时也将进入闲置状态。

在从机模式下，按照主机的 SCLK 的时钟信号，数据通过 MOSI 管脚由主机向从机传输，通过 MISO 管脚由从机向主机传输。一个位计数器记录 SCLK 的边沿数，当数据移位寄存器移入 8 位数据（一字节）；同时，在移位寄存器接收到 8 位数据（一字节），数据将移到接收数据缓存器，同时 SPIIF 置 1。若 SPI 中断被允许，当 SPIIF 置 1 时，会产生一个中断请求。此时接收移位寄存器将保持且 SPIIF 位置 1，这样 SPI 从设备将不会接收到任何数据，直到 SPIIF 位清 0。SPI 从设备必须在主设备开始一次新的数据传输前，将要传的数据写入发送移位寄存器。如果发送开始前未写入数据，从设备将传送“0x00”给主设备。如果写 SPDAT 操作发生在传输过程中，那么 SPI 从设备的 WCOL 标志位置 1，表示写 SPDAT 冲突。但是移位寄存器的数据不受影响，本次传输也不会中断。

## 14.4 数据传输格式

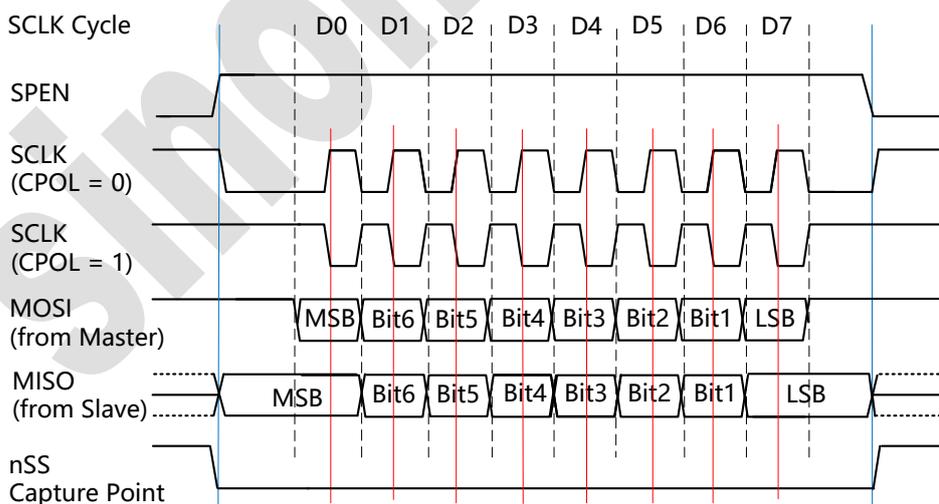
通过软件设置 SPCON 寄存器的 CPOL 位和 CPHA 位，用户可以选择 SPI 时钟 SCLK 的极性和相位的组合。

CPOL 位定义时钟的极性，即空闲时的电平状态。

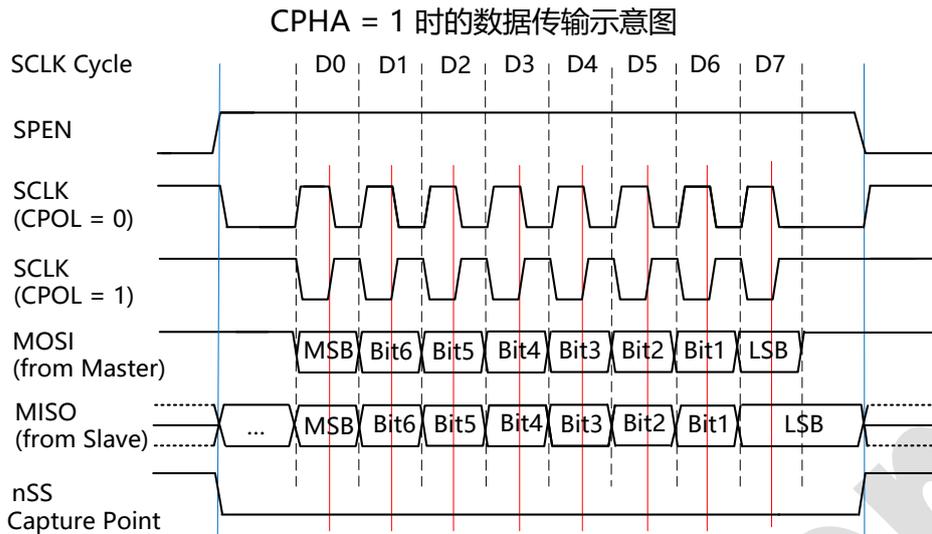
CPHA 位定义时钟的相位，即定义了允许数据采样移位的时钟边沿。在 SPI 通信中主从通讯的两个设备，时钟极性相位的设置应一致。

当 CPHA=0，SCLK 的第一个捕获数据，从设备必须在 SCLK 的第一个沿之前将数据准备好。

CPHA = 0 时的数据传输示意图



当 CPHA=1，主设备在 SCLK 的第一个沿将数据输出到 MOSI 上，从设备把 SCLK 的第一沿作为开始发送信号，SCLK 的第二个沿开始捕获数据，因此用户必须在第一个 SCLK 的两个沿内完成写 SPDAT 的操作。这种数据传输格式是一个主设备一个从设备之间通讯的首选。



## 14.5 出错检测

出错情形一：模式故障（MODF 位）

SPI 主模式下的模式故障出错表明 SS 引脚上的电平状态与实际的设备模式不一致。SPSTA 寄存器中 MODF 位置 1 后，表明系统控制存在多主设备冲突的问题。这种情况下，由于 SPCON 寄存器的 SS 引脚禁止位（SSDIS）清 0，SS 引脚信号为低时，MODF 标志位置 1。然而，对于只有一个主设备的系统来说，主设备的 SS 引脚被拉低，那是绝对不允许的。这种情况下，为防止 MODF 置 1，可使 SPCON 寄存器中的 SSDIS 位置 1，让 SS 引脚作为普通 I/O 口或是其它功能引脚。

此时，在重启 SPI 通信前，用户必须将 MODF 位以软件方式清 0，将 SPCON 寄存器中的 MSTR 位和 SPSTA 寄存器的 SPEN 位置 1，重新启动主模式。

*注：当 SSDIS=0 且 MODF=1 时将产生 SPI 错误中断。*

出错情形二：写冲突（WCOL 位）

在发送数据序列期间写入 SPDAT 寄存器会引起写的冲突，SPSTA 寄存器中的 WCOL 标志位置 1。WCOL 位置 1 不会引起中断，发送也不会中止。该标志位仅用于通知用户，SPI 数据写入的时机不对，WCOL 位需由软件清 0。

出错情形三：超限情况（SPIOV 位）

当作为主设备或从设备时，软件尚未清除 SPIIF 位，此时主或从设备又试图发送/接受几个数据字节时，发生超限情况。在这种情况下，接收移位寄存器保持原有数据，SPIIF 置 1，同样 SPI 设备直到 SPIIF 被清除后才会再接收新的数据。在 SPIIF 位被清除之前继续调用中断，SPIOV 位置 1 不会引起中断，发送也不会中止。该标志位仅用于通知用户，仍有接收的 SPI 数据未读取，SPIOV 位需由软件清 0。

## 14.6 SPI 相关寄存器

SPI 寄存器汇总表

地址 (SFR n/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位 值 (B)



SFR0 /AF	SPDAT	SPI 数据寄存器	SPDAT 7	SPDAT 6	SPDAT 5	SPDAT 4	SPDAT 3	SPDAT 2	SPDAT 1	SPDAT 0	xxxx xxxx
SFR0 /AE	SPSTA	SPI 状态寄存器	SPEN	SPIF	MODF	WCOL	SPIOV	-	-	-	0000 0---
SFR0 /AD	SPCON	SPI 控制寄存器	DIR	MSTR	CPHA	CPOL	SSDIS	SPR2	SPR1	SPR0	0000 0000

#### 14.6.1 SPI 控制寄存器 (SPCON, 0xAD/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SPCON	DIR	MSTR	CPHA	CPOL	SSDIS	SPR2	SPR1	SPR0
R/W								
复位值	0	0	0	0	0	0	0	0

BIT[7] DIR – 传输方向选择位

- 0: 发送时, 由 MSB 开始依次发送;
- 1: 发送时, 由 LSB 开始依次发送。

BIT[6] MSTR – SPI 主从选择位

- 0: 片上 SPI 做从设备使用;
- 1: 片上 SPI 做主设备使用。

注: 如果要设置 MSTR=1, 务必先让 SSDIS=1; 或者在 SSDIS=0 的状态, 但请提前上拉外部 SS 引脚, 否则 SPI 保护功能生效, 并不能正常进入主模式, 该位无法置 1, 仍保持 0 (从模式状态)。

BIT[5] CPHA – SPI 时钟相位控制位

- 0: SPI 周期的第一个电平变化沿采集数据;
- 1: SPI 周期的第二个电平变化沿采集数据。

BIT[4] CPOL – SPI 时钟极性控制位

- 0: 在 Idle 状态下 SCK 处于低电平;
- 1: 在 Idle 状态下 SCK 处于高电平。

BIT[3] SSDIS – SPI 片选信号 SS 控制位

- 0: 在主/从模式下, 打开 SS 引脚;
- 1: 在主/从模式下, 关闭 SS 引脚, SS 脚位可另作他用。

注: 如果 SSDIS 置 1, 不产生 MODF 中断请求; 在从模式下, 若 CPHA 为 0, 该位无效。

BIT[2:0] SPR[2:0] – SPI 时钟速率选择位

SPR[2:0]	SPI 工作时钟 $F_{SPI}$
000	$F_{sys}/2$
001	$F_{sys}/4$
010	$F_{sys}/8$
011	$F_{sys}/16$
100	$F_{sys}/32$
101	$F_{sys}/64$
110	$F_{sys}/128$
111	$F_{sys}/256$



注：作从模式时，SPI 的时钟由主机决定，但不应超过 8Mbit/s；作主模式时，SPI 的时钟由上表决定，最大设置也不应超过 8Mbit/s。

#### 14.6.2 SPI 状态寄存器 (SPSTA, 0xAE/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SPSTA	SPEN	SPIF	MODF	WCOL	SPIOV	-	-	-
R/W	R/W	R/W	R/W	R/W	R/W	-	-	-
复位值	0	0	0	0	0	-	-	-

BIT[7] SPEN – SPI 模块使能位  
 0：关闭 SPI 模块，端口用于通用 IO 或其他复用模块；  
 1：打开 SPI 模块，MOSI、MISO、SCLK 端口有效，开始数据的收发。

注：nSS 端口的使能与否，直接由 SSDIS 位来控制。

BIT[6] SPIF – SPI 数据传送标志位  
 0：由软件清 0；  
 1：表明数据传输已完成，由硬件置 1。

BIT[5] MODF – 模式故障标志位  
 0：由软件清 0；  
 1：表明 SS 引脚电平与 SPI 模式不匹配，由硬件置 1。

BIT[4] WCOL – 写入冲突标志位  
 0：表明没有写入冲突发生，由软件清 0；  
 1：表明已检测到写入冲突发生，由硬件置 1。

BIT[3] SPIOV – 接收超限标志位  
 0：表明以处理接收超限，由软件清 0；  
 1：表明已检测到接收超限，由硬件置 1。

#### 14.6.3 SPI 数据寄存器 (SPDAT, 0xAF/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SPDAT	SPDAT7	SPDAT6	SPDAT5	SPDAT4	SPDAT3	SPDAT2	SPDAT1	SPDAT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	X	X	X	X	X	X	X	X

BIT[7:0] SPDAT[7:0] – SPI 数据缓存寄存器

注 1：写入 SPDAT 的数据被放置到发送移位寄存器中，而读取 SPDAT 时将获得接收移位寄存器中的数据，本质上对应了两个不同的寄存器，因此用户无法直接读取到自己写入的值。

注 2：SPDAT 的复位值为随机值，因此作主模式发送数据前必须设定其初值；当关闭 SPI 功能后，再读取 SPDAT 的内容无效。



## 15 IIC 通讯接口

### 15.1 IIC 概述

本芯片内置 1 组 IIC 功能。

芯片内置 1 个 IIC 总线通讯模块，支持 7 位地址编码主从模式的 IIC 总线通讯。IIC 总线通讯接口为时钟线 SCL 和数据线 SDA 的双向两线接口，IIC 使能后复用的 I/O 端口用作 SCL/SDA，此时为输入/开漏输出口，输出时其内部上拉电阻控制位依然有效，可选择内部或外接合适的上拉电阻，以匹配选定的通讯速率（最高支持 400Kbps）。

注：IIC 模块的时钟源为内部高频时钟  $F_{HIRC}$ ，仅当 HIRC 工作时，IIC 才可正常工作。

### 15.2 IIC 数据传输

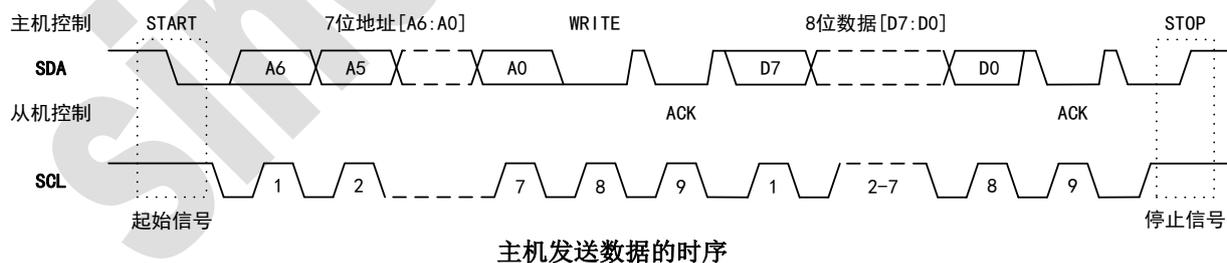
总线空闲时，数据线 SDA 和时钟线 SCL 均为高电平。SDA 电平在 SCL 高电平期间由高变低的下降沿表示起始信号 START，而 SDA 电平在 SCL 高电平期间由低变高的上升沿则表示停止信号 STOP。START/STOP 信号以及 SCL 上的时钟信号均由主机发送，而数据线 SDA 上的数据则由主从双方同步于 SCL 时钟进行单向传输。数据传输时，SDA 电平在 SCL 高电平期间必须保持稳定，只有在 SCL 为低电平时，SDA 电平才允许变化。

一帧数据传输以一个起始信号 START 开始，以一个停止信号 STOP 或重复起始信号 RE-START 结束，一个重复起始信号 RE-START 也是下一帧数据传输的开始（需从机支持重复起始信号 RE-START），期间总线不被释放。

每一帧数据传输时需先由主机发送一个以 7 位从机地址和 1 位读/写命令组成的控制字节，再由主机或从机发送一个或多个数据字节。一个完整字节的传输需 9 个时钟，前 8 个时钟传输 8 位字节内容（最高位最先传输），第 9 个时钟则为应答时钟，此时 SDA 上的电平即为接收方返回的应答信号，低电平表示应答（ACK），高电平表示非应答（NACK）。

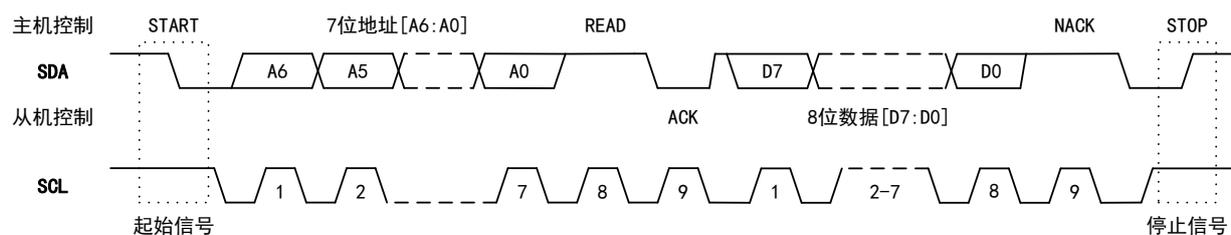
#### 15.2.1 主机到从机的数据传输

主机先发送起始信号 START，再发送一个包含“写”命令的控制字节，从机返回一个 ACK，然后主机开始发送数据字节，从机接收完每一个数据字节后均返回一个 ACK，主机在接收到最后一个字节从机返回的 ACK 后，发送停止信号 STOP 结束本次数据传输。



#### 15.2.2 从机到主机的数据传输

主机先发送起始信号 START，再发送一个包含“读”命令的控制字节，从机返回一个 ACK，然后主机开始接收从机发送的数据字节，并在接收完每一个数据字节后均返回一个 ACK，当主机不再接收数据时则在接收完从机上一个字节后返回一个 NACK，接着发送停止信号 STOP 结束本次数据传输。



主机接收数据的时序

## 15.3 IIC 工作模式

IIC 总线接口模块支持 7 位地址编码的主从模式，可通过寄存器位 IICMOD 选择主机工作模式或从机工作模式。主机模式下，总线时钟从 SCL 端口输出，通讯速率可通过寄存器位 IICSPD 选择 100Kbps 或 400Kbps；从机模式下，总线时钟由 SCL 端口输入，通讯速率在同步采样方式（IICSYN=0）时需通过 IICSPD 选择 100Kbps 或 400Kbps，而在异步采样方式（IICSYN=1）时最高可支持 400Kbps。

### 15.3.1 主机模式

主机模式下，IIC 模块可单独发送地址帧（包含起始信号）、数据帧、或停止信号。

主机发送地址帧时，会将地址寄存器 IICAR 中内容作为地址（高 7 位）和命令（最低位）发送；

主机发送数据帧时，会将发送缓冲器（IICTD\_buf）中内容作为数据发送，而在接收数据帧时，则将接收的数据存入接收缓冲器（IICRD\_buf）中，IICTD\_buf 和 IICRD\_buf 均需通过数据寄存器 IICDR 进行读/写操作，写 IICDR 为写 IICTD\_buf，读 IICDR 则为读 IICRD\_buf。

主机发送完地址或数据帧（第 9 位接收完从机应答信号）、主机接收完数据帧（第 9 位发送完应答信号）、或主机发送完停止信号后，中断标志 IICIF 将被置 1 触发 IIC 中断。

### 15.3.2 从机模式

从机模式下，IIC 模块在总线上检测到 IIC 起始信号后开始接收主机发送的地址及命令（缓存在接收缓冲器中），若地址与 IICAR 中预设的从机地址（若使能广播地址则还包括地址 00H）匹配，则在第 9 位应答时钟期间将寄存器位 TACKS 内的应答内容发送给主机。若发送的应答内容为 NACK，则发送完成后重置 IIC 模块以等待总线上的下一次起始信号；若发送的应答内容为 ACK，则发送完成后地址匹配标志位 SADMF 将被置 1，同时 IIC 模块会将 SCL 电平拉为低电平（清中断标志位就会导致 SADMF 标志位清零），中断标志 IICIF 将被置 1 触发 IIC 中断。

清 0 中断标志 IICIF 可将 SCL 线释放，此前可通过读写命令标志位 SCMDF 准备接收或发送数据。从机接收或发送数据帧时，也会将接收的数据存入接收缓冲器（IICRD\_buf）中或将发送缓冲器（IICTD\_buf）中内容发送。接收/发送完数据并发送/接收完应答信号后，数据帧收发完成标志位 SDTFF 将被置 1，同时 IIC 模块会再次将 SCL 电平拉为低电平（下一次检测到 SCL 电平上升沿时 SDTFF 将被清 0），中断标志 IICIF 将被置 1 触发 IIC 中断。清 0 中断标志 IICIF 释放 SCL 线，可使主机继续传输数据帧或发送停止信号。

检测到总线上的停止信号后，从机模式通讯结束。

*注：从机模式地址匹配时，接收到的地址信息缓存在 IICRD\_buf 中，因此若使能广播地址，则可通过 IICDR 读取 IICRD\_buf 中的地址信息，判断当前所响应的地址是广播地址 00H 还是预设于 IICAR 中的从机地址。*

从机模式还可选择同步或异步采样方式，同步方式可增加采样准确率，异步方式则支持休眠模式下检测总线信号，

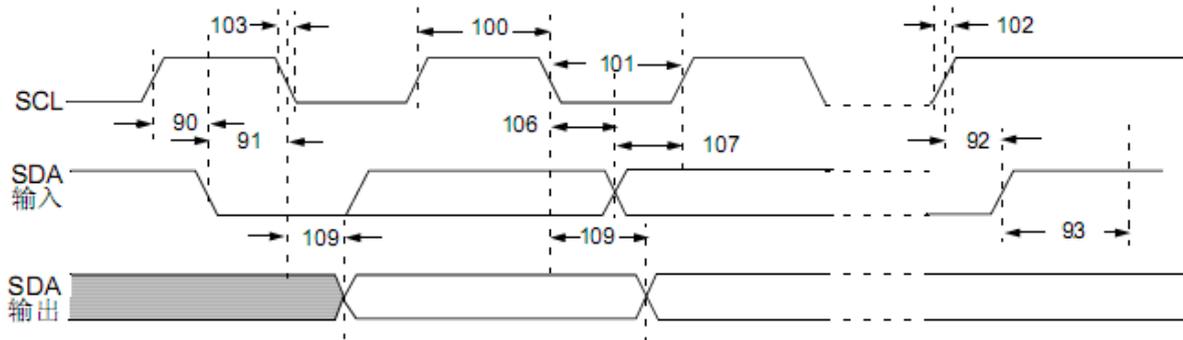


当有地址匹配或接收完成时将会唤醒 CPU。同步/异步方式的切换需在 SCL 被拉住后进行，且切换后需等待  $t_{\text{HIGH}}$  时间后方可释放 SCL 线。

#### 应答寄存器位

无论主机模式还是从机模式，发送应答信号时将发送寄存器 TACKS 中内容、接收到的应答内容保存在寄存器位 RACKF 中。

### 15.4 IIC 时序说明



#### 主机模式信号输出时间定义

标号	参数	符号	典型值 (TYP)		单位
			100Kbps	400Kbps	
90	重复起始条件的建立时间	$T_{\text{SU;STA}}$	6	1.5	$\mu\text{s}$
91	起始条件的保持时间	$t_{\text{HD;STA}}$	4	1	$\mu\text{s}$
101	SCL 时钟的低电平周期	$t_{\text{LOW}}$	6	1.5	$\mu\text{s}$
100	SCL 时钟的高电平周期	$t_{\text{HIGH}}$	4	1	$\mu\text{s}$
102	SDA 和 SCL 上升沿时间	$T_{\text{R}}$	1000	300	ns
103	SDA 和 SCL 下降沿时间	$T_{\text{F}}$	300	300	ns
109	时钟输出有效时间	$T_{\text{AA}}$	5	1	$\mu\text{s}$
92	停止条件的建立时间	$t_{\text{SU;STO}}$	4	1	$\mu\text{s}$
93	停止条件的保持时间	$t_{\text{HD;STO}}$	6	1.5	$\mu\text{s}$

#### 主机模式信号输入时间要求

标号	参数	符号	典型值 (TYP)		单位
			100Kbps	400Kbps	
106	数据输入保持时间	$T_{\text{HD;DAT}}$	1	0.5	$\mu\text{s}$
107	数据输入建立时间	$T_{\text{SU;DAT}}$	300	150	ns



## 从机模式信号输入时间要求

标号	参数	符号	典型值 (TYP)		单位
			100Kbps	400Kbps	
90	重复起始条件的建立时间	$T_{SU;STA}$	4.7	0.6	$\mu\text{s}$
91	起始条件的保持时间	$t_{HD;STA}$	4	0.6	$\mu\text{s}$
101	SCL 时钟的低电平周期	$t_{LOW}$	4.7	1.3	$\mu\text{s}$
100	SCL 时钟的高电平周期	$t_{HIGH}$	4	0.6	$\mu\text{s}$
107	数据输入建立时间	$t_{SU;DAT}$	300	150	ns
92	停止条件的建立时间	$t_{SU;STO}$	4	0.6	$\mu\text{s}$
93	停止条件的保持时间	$t_{HD;STO}$	4	0.6	$\mu\text{s}$

## 15.5 IIC 多机通讯

IIC 模块支持多主机通讯，可实现时钟同步和总线仲裁功能。

主机模式下，IIC 模块发送时钟的高电平后会检测 SCL 线，直至 SCL 为高后开始高电平宽度计时，在高电平保持时间达到后才发送时钟的低电平，若在高电平计时期间 SCL 线被其他设备拉低，IIC 模块则将直接进入低电平宽度计时，如此 SCL 线上时钟的低电平时间由所有主机中最长的低电平时长决定，而高电平时间由最短的高电平时长决定，从而实现多主机在 SCL 线上的时钟同步。

主机或从机模式下，IIC 模块发送数据的高电平后会进行总线仲裁，即检测 SDA 线上的电平状态，若仍为低电平，则表示高电平未发送成功，已有其他设备占用总线，此时 IIC 模块将释放总线，总线仲裁状态标志 IICARLO 将被置 1。

主机模式下发送起始信号或重复起始信号时，IIC 模块释放 SCL 总线时检测到 SDA 为低电平和 IIC 模块拉低 SDA 时检测到 SCL 为低电平，总线仲裁状态标志 IICARLO 将被置 1。

主机模式下发送停止信号时，IIC 模块释放 SDA 时检测到 SCL 为低电平和停止信号发送完毕后检测到 SDA 为低电平，总线仲裁状态标志 IICARLO 将被置 1。

主机模式下在发送地址帧/数据帧、发送应答时，从机模式下在发送数据帧或发送应答时，在向 SDA 发送高电平，但在 SCL 的上升沿却从 SDA 上检测到低电平，总线仲裁状态标志 IICARLO 将被置 1。

只有使能位 IICEN 清 0 才能将状态标志 IICARLO 清 0。

## 15.6 IIC 主从机地址

IIC 模块中有设置对应的主/从机地址功能 (IICAR 和 IICAAR)。

主机模式下只有一组主机地址 IICAR 为发送的地址，从机模式下可以有从机地址 IICAR (为预设的从机地址) 和 IICAAR (IICAAR 为预设的从机辅地址)，辅地址 IICAAR 根据对应地址掩码寄存器 IICAMR 控制从机辅地址哪几位生效，如果全部设置为不检验，辅地址就自动无效。



## 15.7 IIC 相关寄存器

IIC 寄存器汇总表

地址 (SFR n/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位 值 (B)
SFR0 /E7	IICAMR	IIC 辅地址掩码 寄存器	IICAM6	IICAM5	IICAM4	IICAM3	IICAM2	IICAM1	IICAM0	-	0000 000-
SFR0 /E6	IICAAAR	IIC 辅地址寄存 器	IICAA6	IICAA5	IICAA4	IICAA3	IICAA2	IICAA1	IICAA0	-	0000 000-
SFR0 /E5	IICDR	IIC 数据寄存器	IICD7	IICD6	IICD5	IICD4	IICD3	IICD2	IICD1	IICD0	0000 0000
SFR0 /E4	IICAR	IIC 地址寄存器	IICA6	IICA5	IICA4	IICA3	IICA2	IICA1	IICA0	MCMDS	0000 0000
SFR0 /E3	IICSR	IIC 状态寄存器	TACKS	IICARL O	IICSTR F	IICSTP F	SADMF	SDTFF	SCMDF	RACKF	0000 0000
SFR0 /E2	IICCR1	IIC 控制寄存器 1	IICIF	-	-	MNAST PE	MTSAA	MTSTP	MTDTA	MRDTA	0--0 0000
SFR0 /E1	IICCR0	IIC 控制寄存器 0	IICEN	IICMOD	IICSPD	IICSYN	IICGCE	IICCHS 1	IICCHS 0	IICRUS	0000 0000

## 15.7.1 IIC 控制寄存器 0 (IICCR0, 0xE1/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IICCR0	IICEN	IICMOD	IICSPD	IICSYN	IICGCE	IICCHS1	IICCHS0	IICRUS
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	0	0	0	0	0	0

BIT[7] IICEN - IIC 使能位

0: 关闭 IIC, 端口用作通用 I/O;

1: 使能 IIC, IICCHS 选定的端口用作 SCL/SDA;

注: 应用 IIC 模块, 选定的端口需使其数字 I/O 功能。使能 IIC 前, 软件将端口转为输入/开漏输出口, 输出功能时根据应用外围电路配置上拉功能;

BIT[6] IICMOD - IIC 工作模式选择位

0: 主机模式;

1: 从机模式;

BIT[5] IICSPD - IIC 通讯速率选择位 (实际速率受芯片及外围电路影响)

0: IIC 通讯速率为 100Kbps;

1: IIC 通讯速率为 400Kbps;

BIT[4] IICSYN - IIC 从机模式采样方式选择位

0: 同步采样;

1: 异步采样;

BIT[3] IICGCE - IIC 从机模式广播呼叫使能位

0: 从机模式下不支持广播呼叫功能, 即不响应广播地址 00H;



1: 从机模式下支持广播呼叫功能, 即响应广播地址 00H;

注: 做从机时, 即使不使能广播模式, 只要从机地址配置为 0x00, 仍能触发正常通信。

BIT[2:1] IICCHS[1:0] – IIC 通讯接口选择位

00: IIC 通讯的时钟/数据接口为 P1.1/P1.2;

01: IIC 通讯的时钟/数据接口为 P1.6/P1.5;

10: IIC 通讯的时钟/数据接口为 P2.2/P2.3;

11: IIC 通讯的时钟/数据接口为 P1.1/P2.0; (仅 20 pin 封装支持)

BIT[0] IICRUS – IIC 端口内部上拉电阻选择位 (被选上拉电阻仍受端口上拉电阻控制位影响)

0: 选择内部通用上拉电阻 (15K);

1: 选择内部 IIC 专用上拉电阻 (1.8K);

### 15.7.2 IIC 控制寄存器 1 (IICCR1, 0xE2/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IICCR1	IICIF	-	-	MNASTPE	MTSAA	MTSTP	MTDTA	MRDTA
R/W	R/W	-	-	R/W	R/W	R/W	R/W	R/W
初始值	0	-	-	0	0	0	0	0

BIT[7] IICIF – IIC 中断标志位

0: 未触发 IIC 中断;

1: 已触发 IIC 中断, 需软件清 0。

注 1: 该标志位为总标志位, “IICSR” 中 “IICARLO、IICSTRF、SADMF、SDTFF” 任意一个标志触发都能置起该总标志。

注 2: 作为 IIC 主机时, “IICSTPF” 无法触发 “IICIF”; 作为 IIC 从机时, “IICSTPF” 可以触发 “IICIF”

BIT[4] MNASTPE – 主机模式发送完 NACK 或接收到 NACK 后自动发送 STOP 信号使能位

0: 发送完 NACK 或接收到 NACK 后, 不自动发送 STOP 信号;

1: 发送完 NACK 或接收到 NACK 后, 自动发送 STOP 信号;

BIT[3] MTSAA – 主机模式发送 START 信号和地址帧再接收应答控制位

0: 操作未开始或已完成, 写 1 开始发送起始信号和地址帧再接收从机应答信号;

1: 发送起始信号和地址帧再接收从机应答信号操作中, 完成后自动清 0;

BIT[2] MTSTP – 主机模式发送 STOP 停止信号控制位

0: 操作未开始或已完成, 写 1 开始发送停止信号;

1: 发送停止信号操作中, 完成后自动清 0;

BIT[1] MTDTA – 主机模式发送数据帧再接收应答控制位

0: 操作未开始或已完成, 写 1 开始发送数据帧再接收从机应答信号;

1: 发送数据帧再接收从机应答信号操作中, 完成后自动清 0;

BIT[0] MRDTA – 主机模式接收数据帧再发送应答控制位

0: 操作未开始或已完成, 写 1 开始接收从机数据帧再发送应答信号;

1: 接收从机数据帧再发送应答信号操作中, 完成后自动清 0;

### 15.7.3 IIC 状态寄存器 (IICSR, 0xE3/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IICSR	TACKS	IICARLO	IICSTRF	IICSTPF	SADMF	SDTFF	SCMDF	RACKF



R/W	R/W	R	R	R	R	R	R	R
初始值	0	0	0	0	0	0	0	0

- BIT[7] TACKS – 应答信号发送内容选择位  
 0: 待发送的应答信号为应答 (ACK);  
 1: 待发送的应答信号为非应答 (NACK);
- BIT[6] IICARLO – IIC 总线仲裁状态标志位  
 0: IIC 总线上未检测到仲裁失败;  
 1: IIC 总线上检测到仲裁失败 (仅在 IIC 关闭后清 0);
- BIT[5] IICSTRF – IIC 总线 START 信号检测状态标志位  
 0: 总线上未检测到起始信号;  
 1: 总线上检测到起始信号 (检测到停止信号后清 0);
- BIT[4] IICSTPF – IIC 总线 STOP 信号检测状态标志位  
 0: 总线上未检测到停止信号;  
 1: 总线上检测到停止信号 (检测到起始信号后清 0);
- BIT[3] SADMF – 从机模式地址匹配状态标志位  
 0: 从机模式接收到的地址帧中地址不匹配, 或接收到数据帧;  
 1: 从机模式接收到地址帧, 且地址与本机预设地址匹配;
- BIT[2] SDTFF – 从机模式数据帧收发完成状态标志位  
 0: 从机模式数据帧接收/发送中, 或未接收到数据帧;  
 1: 从机模式数据帧接收/发送完成;
- BIT[1] SCMDF – 从机模式读写命令接收内容标志位  
 0: 从机模式接收到的命令为“读”命令 (逻辑“0”);  
 1: 从机模式接收到的命令为“写”命令 (逻辑“1”);
- BIT[0] RACKF – 应答信号接收内容标志位  
 0: 接收到的应答信号为应答 (ACK);  
 1: 接收到的应答信号为非应答 (NACK);

注: 该标志位在写起始信号或停止信号后自动清零。

#### 15.7.4 IIC 地址寄存器 (IICAR, 0xE4/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IICAR	IICA6	IICA5	IICA4	IICA3	IICA2	IICA1	IICA0	MCMDS
R/W								
初始值	0	0	0	0	0	0	0	0

BIT[7:1] IICA[6:0] – IIC 通讯地址, 主机模式下为发送的地址, 从机模式下为预设的从机地址

- BIT[0] MCMDS – 主机模式下发送的读/写命令控制位  
 0: 待发送的命令为“写”命令 (逻辑“0”);  
 1: 待发送的命令为“读”命令 (逻辑“1”);

#### 15.7.5 IIC 数据寄存器 (IICDR, 0xE5/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IICDR	IICD7	IICD6	IICD5	IICD4	IICD3	IICD2	IICD1	IICD0



R/W								
初始值	0	0	0	0	0	0	0	0

BIT[7:0] IICD[7:0] – IIC 通讯数据，写操作为写发送缓冲器数据，读操作为读接收缓冲器数据

注：对 IICDR 读和写操作的目的寄存器不是同一寄存器。

#### 15.7.6 IIC 辅地址寄存器 (IICAAAR, 0xE6/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IICAAAR	IICAA6	IICAA5	IICAA4	IICAA3	IICAA2	IICAA1	IICAA0	-
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	0	0	0	0	0	0

BIT[7:1] IICAA[6:0] – IIC 通讯辅地址，主机模式下无效，从机模式下为预设的从机辅地址

#### 15.7.7 IIC 辅地址掩码寄存器 (IICAMR, 0xE7/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IICAMR	IICAM6	IICAM5	IICAM4	IICAM3	IICAM2	IICAM1	IICAM0	-
R/W	R/W							
初始值	0	0	0	0	0	0	0	0

BIT[7:1] IICAM[6:0] – 从机模式 IIC 通讯从地址掩码控制位 (n=6-0)

0: 从机模式下，检验预设地址位 IICAA[6:0]与接收的对应地址位是否匹配；

1: 从机模式下，不检验接收的对应地址位；



## 16 中断

芯片的中断包括外部中断 INT0X (有 5 个通道共用该中断源)、INT1X (有 5 个通道共用该中断源)、定时器中断 (T0、T1、T2、T3)、UART0 中断、ADC 转换中断、PWM0 中断、PWM1 中断、UART1 中断、SPI 中断、PWM1 刹车保护中断 (PWM1FB)、IIC 中断、CRC 中断和 LVD 中断共 15 个可用中断源, 分为 2 个中断优先级。

每个中断均具有独立的中断标志位, 中断使能位、中断向量和优先级设置位, 还可以通过 1 个全局中断使能位将所有中断同时禁止。

### 16.1 中断源列表

芯片的中断源及其中断向量、相关控制位等列表如下:

中断源	触发条件	中断标志	中断使能位	优先级设置位	中断向量	中断号 (C51)	查询优先级	中断标志清除方式	停止模式唤醒	休眠模式唤醒
Reset	-	-	-	-	0000H	-	0(最高)	-	Yes	Yes
INT0X	上升沿/下降沿/双沿	EXT0IF0~EXT0IF4	EX0IE	EX0IPL	0003H	0	1	软件清除	Yes	Yes
T0	T0 溢出	TF0	T0IE	T0IPL	000BH	1	2	硬件自动	Yes	No
INT1X	上升沿/下降沿/双沿	EXT1IF0~EXT1IF4	EX1IE	EX1IPL	0013H	2	3	软件清除	Yes	Yes
T1	T1 溢出	TF1	T1IE	T1IPL	001BH	3	4	硬件自动	Yes	No
UART0	UART0 收/发完成	RI0/TI0	UART0IE	UART0IPL	0023H	4	5	软件清除	Yes	No
SPI	SPI 收发完成/模式故障	SPIIF/SSDIS=0 且 MODF=1	SPIIE	SPIIPL	002BH	5	6	软件清除	Yes	No
ADC	ADC 转换完成	ADCIF	ADCIE+ADEOCIE	ADCIPL	0033H	6	7	软件清除	Yes	No
	ADC 比较触发	ADCCMPIF	ADCIE+ADCMP IE							
T2	T2 溢出、比较/T2 重载、捕获	TF2/EXTF2	T2IE	T2IPL	003BH	7	8	软件清除	Yes	No
T3	T3 溢出	TF3	T3IE	T3IPL	0043H	8	9	硬件自动	Yes	Yes
PWM1	PWM1 溢出	PWMPIF	PWM1IE+PWMP IE	PWM1IPL	004BH	9	10	软件清除	Yes	No
	PWM1 归零	PWMZIF	PWM1IE+PWW ZIE							
	PWM1C 通道占向下计数匹配	PWMDDCI F	PWM1IE+PWMDDCIE							
	PWM1C 通道占向上计数匹配	PWMUDCI F	PWM1IE+PWWUDCIE							
	PWM1B 通道占向下计数匹配	PWMDDBI F	PWM1IE+PWMDDBIE							
	PWM1B 通道占向上计数匹配	PWМУDBI	PWM1IE+							



	上计数匹配	F	PWMUDBIE							
	PWM1A 通道占向 下计数匹配	PWMDDAI F	PWM1IE+ PWMDDAIE							
	PWM1A 通道占向 上计数匹配	PWMUDAI F	PWM1IE+ PWMUDAIE							
保留	-	-	-	-	0053H	10	11	软件清除	Yes	No
PWM0	PWM0 溢出	PWM0IF	PWM0IE + PWM0PIE	PWM0IPL	005BH	11	12	硬件自动	Yes	No
PWM1FB	PWM1FB 上的高/ 低电平	FBSTA	PWM1FBIE	PWM1FBIPL	0063H	12	13	软件清除	Yes	No
IIC	IIC 收/发完成	IICIF	IICIE	IICIPL	006BH	13	14	软件清除	Yes	No
CRC	CRC 校验完成	CRCIF	CRCIE	CRCIPL	0073H	14	15	软件清除	No	No
保留	-	-	-	-	007BH	15	16	软件清除	Yes	No
LVD	LVD 电压检测	LVDIF	LVDIE	LVDIPL	0083H	16	17	软件清除	Yes	Yes

## 16.2 中断优先级

芯片的中断具有 2 个中断优先级，中断源的中断请求可设定为高优先级或低优先级，通过优先级的设定可实现中断服务程序的 2 级嵌套。

低优先级的中断服务程序能被高优先级请求所中断，反之则不行；中断服务程序不能被同一级的中断请求所中断。当多个同一优先级的中断同时产生中断请求时，根据预定的查询优先级顺序响应。

## 16.3 中断处理流程

当一个中断产生并且被芯片响应，则主程序运行被中断，将执行下述操作：

1. 当前正在执行的指令执行完；
2. PC 值被压入堆栈，保护现场；
3. 中断向量地址载入程序计数器 PC；
4. 执行相应的中断服务程序；
5. 中断服务程序结束并 RETI；
6. 将 PC 值出栈，并返回执行中断前的程序继续执行。

在此过程中，系统不会立即执行其它同一优先级的中断，但会保留所发生的中断请求，在当前中断处理结束后，再执行一条指令，之后转去执行新的中断请求。

当用户开启优先级中断，高优先级中断可以打断低优先级的中断，本芯片可以实现 2 级中断嵌套。

## 16.4 外部中断

本芯片有 2 个外部中断源 INT0 和 INT1，每个中断源分别有 5 路外部输入，对应端口 INTxy (xy = 0-1; y = 0-4)。外部中断为边沿触发，配置控制位 ExnIS[1:0] (n=0-1) 选择不同的触发沿（不触发/上升沿/下降沿/双沿）。用户可以通过 EXTEN/EXTEN1 寄存器灵活地在 4 个口中选择一个或多个端口作为输入，端口 INT0y (y=0-4) 复用同一个中断源 INT0，端口 INT1y (y=0-4) 复用同一个中断源 INT1。

若设置触发沿为下降沿，一个周期内引脚输入连续采样为高电平，下个周期采样为低电平，则对应中断标志置起，



发出中断请求；若多个中断请求同时发生，同一个中断源只响应一个中断请求，用户可以在中断处理程序中进行标志区分；中断标志需要软件清零，未清零标志，硬件将会在中断退出后再次发出中断请求。

同一个中断源的 5 路输入，当其中 1 路触发已进入中断处理程序，不影响新的中断标志建立，只是不会响应新的中断请求。

注：外部中断输入通道内建最小 100ns 模拟滤波，高温低压条件下滤波偏大（不超过 500ns）。外部中断沿信号的高低电平必须大于（1 个系统时钟周期+模拟滤波时间），才能保证有效的触发。

EXTIS2 寄存器的 EXT0A 位为 INT0 的总使能开关，EXT1A 位为 INT1 的总使能开关。

#### 16.4.1 外部中断使能寄存器 (EXTEN, 0xA6/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EXTEN	-	-	-	EXT0EN4	EXT0EN3	EXT0EN2	EXT0EN1	EXT0EN0
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
复位值	-	-	-	0	0	0	0	0

#### 16.4.2 外部中断使能寄存器 1 (EXTEN1, 0xA5/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EXTEN1	-	-	-	EXT1EN4	EXT1EN3	EXT1EN2	EXT1EN1	EXT1EN0
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
复位值	-	-	-	0	0	0	0	0

BIT[7~0] EXTnENm (n=0-1; m=0-7) – 外部中断 INTxy (x=0-1; y=0-4) 端口使能位

0: 禁止对应某一路 INTxy (x=0-1; y=0-4) 端口的复用功能，端口用于通用 IO 或其他复用模块；

1: 使能对应某一路 INTxy (x=0-1; y=0-4) 端口的复用功能，端口作为外部中断输入用。

注 1: 要使能对应的 INTxy (x=0-1; y=0-4) 端口，还必须先使能 EXTnA (n=0-1) 位。中断源 INTx (x=0-1)，输入信号有效后将产生对应的 EXTnIFm (n=0-1; m=0-4) 标志位置 1。

注 2: 当使用 T0/T1 测量电平宽度时，T0/T1 不能用作他用，且只通过 INT04/INT14，即这两个外设联动工作，资源同时被使用。此时 INT04 使用 T0，计算的方式可对应 T0 的方式 0-3，四种方式；而 INT14 使用 T1，但计算的方式可对应 T1 的方式 0-2，仅三种方式。

#### 16.4.3 外部中断控制寄存器 2 (EXTIS2, 0xB5/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EXTIS2	EXT0A	EXT1A	-	-	EXT1IS1	EXT1IS0	EXT0IS1	EXT0IS0
R/W	R/W	R/W	-	-	R/W	R/W	R/W	R/W
复位值	0	0	-	-	0	0	0	0

BIT[7], BIT[6] EXTnA (n=0-1) – 外部中断 INTxy (x=0,1; y=0-4) 总使能位

0: 禁止所有物理 INTxy (x=0-1; y=0-4) 端口上外部中断功能，即关闭 INTx 模块；

1: 使能所有物理 INTx (x=0-1; y=0-4) 端口上外部中断功能，即打开 INTx 模块。

BIT[3:2] EXT1IS[1:0] – 外部中断 INT1y (y=0~4) 触发方式选择位

BIT[1:0] EXT0IS[1:0] – 外部中断 INT0y (y=0~4) 触发方式选择位

EXT0IS[1:0]/ EXT1IS[1:0]	INTxy (x=0,1) 的端口触发方式
00	不触发
01	下降沿触发
10	上升沿触发



11	双沿触发
----	------

注：同一个中断源  $INT_x$  ( $x = 0-1$ ) 而言，对  $INT_{xy}$  ( $x = 0,1; y = 0-4$ ) 这 5 个端口将使用共同的触发方式，不可单独设置。

EXnIS[1:0]	在使用 T0/T1 脉冲宽度测量方式下，INT04/INT14 的端口触发方式
XX	始终有效（用于测量高电平的脉冲宽度）

注：在使用 T0/T1 脉冲宽度测量方式下，仅 INT04 和 INT14 两个端口可用，且 EXnIS[1:0] 控制位不参与控制，此时可用于测量高电平的脉冲宽度；在 INT04 对应于 T0 的资源，INT14 则对应于 T1 的资源。

#### 16.4.4 外部中断控制寄存器 1 (EXTIS1, 0xB6/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EXTIS1	-	-	-	EXT1IF4	EXT1IF3	EXT1IF2	EXT1IF1	EXT1IF0
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
复位值	-	-	-	0	0	0	0	0

#### 16.4.5 外部中断控制寄存器 0 (EXTIS, 0xB7/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EXTIS	-	-	-	EXT0IF4	EXT0IF3	EXT0IF2	EXT0IF1	EXT0IF0
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
复位值	-	-	-	0	0	0	0	0

BIT[7~0] EXTnIFm ( $n = 0-1; m = 0-7$ ) - 外部中断  $INT_{xy}$  ( $x = 0,1; y = 0-4$ ) 标志位

0: 在对应物理  $INT_{xy}$  ( $x = 0,1; y = 0-4$ ) 端口上没有外部中断产生，由软件清 0；

1: 在对应物理  $INT_{xy}$  ( $x = 0,1; y = 0-4$ ) 端口上有外部中断产生，由硬件置 1，若同时 EXnIE ( $n = 0,1$ ) 位置位，同时将产生  $INT_x$  ( $x = 0,1$ ) 中断，则芯片会响应该中断请求。

注 1: 对  $EXT0ifm$  ( $m = 0~4$ )，对应于 INT0 中断源； $EXT1ifm$  ( $m = 0~4$ )，对应于 INT1 中断源；每个物理端口标志独立建立，但每个中断源的 5 个子中断口并不会嵌套，同一个中断源不会同时响应多个中断请求。

注 2: 这些标志位置 1 后需要软件清 0。

### 16.5 中断相关寄存器

#### 中断寄存器汇总表

地址 (SFRn/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
SFR0/BD	IPL2	中断优先级控制 2 低位寄存器	-	-	-	-	-	-	LVDIPL	-	---- --0-
SFR0/BB	IPL1	中断优先级控制 1 低位寄存器	CRCIPL	IICIPL	PWM1FB I IPL	PWM0IP L	-	PWM1IP L	T3IPL	T2IPL	0000 -000
SFR0/B8	IPL0	中断优先级控制 低位寄存器	-	ADCIPL	SPIIPL	UART0IP L	T1IPL	EXT1IPL	T0IPL	EXT0IPL	-000 0000
SFR0/B7	EXTIS	外部中断控制寄存器	-	-	-	EXT0IF4	EXT0IF3	EXT0IF2	EXT0IF1	EXT0IF0	---0 0000
SFR0/	EXTIS1	外部中断控制寄	-	-	-	EXT1IF4	EXT1IF3	EXT1IF2	EXT1IF1	EXT1IF0	---0



B6		寄存器 1									0000
SFR0/ B5	EXTIS2	外部中断控制寄存器 2	EXT1A	EXT0A	-	-	EXT1IS1	EXT1IS0	EXT0IS1	EXT0IS0	00-- 0000
SFR0/ AB	IEN2	中断使能寄存器 2	-	-	-	-	-	-	LVDIE	-	---- --00
SFR0/ AA	IEN1	中断使能寄存器 1	CRCIE	IICIE	PWM1FB IE	PWM0IE	-	PWM1IE	T3IE	T2IE	0000 -000
SFR0/ A8	IEN0	中断使能寄存器	EA	ADCIE	SPIIE	UART0IE	T1IE	EXT1IE	T0IE	EXT0IE	0000 0000
SFR0/ A6	EXTEN	外部中断使能寄存器	-	-	-	EXT0EN4	EXT0EN3	EXT0EN2	EXT0EN1	EXT0EN0	---0 0000
SFR0/ A5	EXTEN1	外部中断使能寄存器 1	-	-	-	EXT1EN4	EXT1EN3	EXT1EN2	EXT1EN1	EXT1EN0	---0 0000

### 16.5.1 中断使能寄存器 (IEN0, 0xA8/SFR0)

SBIT 寄存器	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IEN0	EA	ADCIE	SPIIE	UART0IE	T1IE	EXT1IE	T0IE	EXT0IE
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7] EA – 总中断使能位

0: 关闭所有中断;

1: 打开中断, 但具体中断使能控制由相应的控制位决定。

BIT[6] ADCIE – ADC 中断使能位

0: 关闭 ADC 中断;

1: 打开 ADC 中断。

BIT[5] SPIIE – SPI 中断允许控制位

0: 禁止 SPI 中断;

1: 允许 SPI 中断, 当标志位 SPIIF=1 直接产生正常中断或者 SSDIS=0 且 MODF=1 时将产生 SPI 错误中断。

BIT[4] UART0IE – UART0 中断使能位

0: 关闭 UART0 中断;

1: 打开 UART0 中断。

BIT[3] T1IE – T1 中断使能位

0: 关闭 T1 中断;

1: 打开 T1 中断。

BIT[2] EXT1IE – INT1 中断使能位

0: 关闭 INT1 中断;

1: 打开 INT1 中断。

BIT[1] T0IE – T0 中断使能位

0: 关闭 T0 中断;

1: 打开 T0 中断。

BIT[0] EXT0IE – INT0 中断使能位



0: 关闭 INTO 中断;

1: 打开 INTO 中断。

### 16.5.2 中断使能寄存器 1 (IEN1, 0xAA/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IEN1	CRCIE	IICIE	PWM1FBIE	PWM0IE	-	PWM1IE	T3IE	T2IE
R/W	R/W	R/W	R/W	R/W	-	R/W	R/W	R/W
复位值	0	0	0	0	-	0	0	0

BIT[7] CRCIE – CRC 中断使能位

0: 禁止 CRC 中断;

1: 打开 CRC 中断。

BIT[6] IICIE – IIC 中断使能位

0: 禁止 IIC 中断;

1: 打开 IIC 中断。

BIT[5] PWM1FBIE – PWM1FB 中断使能位

0: 禁止 PWM1FB 中断;

1: 打开 PWM1FB 中断。

BIT[4] PWM0IE – PWM0 中断使能位

0: 禁止 PWM0 中断;

1: 打开 PWM0 中断。

*注: 必须与 PWM0 模块的中断使能同时使能, 中断使能有效*

BIT[2] PWM1IE – PWM1 总中断使能位

0: 禁止 PWM1 总中断;

1: 打开 PWM1 总中断。

BIT[1] T3IE – T3 中断使能位

0: 关闭 T3 中断;

1: 打开 T3 中断。

BIT[0] T2IE – T2 中断使能位

0: 关闭 T2 中断;

1: 打开 T2 中断。

*注: 该中断源可同时受 TF2 和 EXTF2 两个中断标志位影响, 但原则上 T2 的这两个功能不可同时使用。*

#### 1.1.1 中断使能寄存器 2 (IEN2, 0xAB/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IEN2	-	-	-	-	-	-	LVDIE	-
R/W	-	-	-	-	-	-	R/W	-
复位值	-	-	-	-	-	-	0	-

BIT[1] LVDIE – LVD 中断使能位

0: 关闭 LVD 中断;

1: 打开 LVD 中断;

### 16.5.3 中断优先级控制低位寄存器 (IPL0, 0xB8/SFR0)



SBIT 寄存器	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IPL0	-	ADCIPL	SPIIPL	UART0IPL	T1IPL	EX1IPL	T0IPL	EX0IPL
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	-	0	0	0	0	0	0	0

#### 16.5.4 中断优先级控制 1 低位寄存器 (IPL1, 0xBB/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IPL1	CRCIPL	IICIPL	PWM1FBIP L	PWM0IPL	-	PWM1IPL	T3IPL	T2IPL
R/W	R/W	R/W	R/W	R/W	-	R/W	R/W	R/W
复位值	0	0	0	0	-	0	0	0

#### 16.5.5 中断优先级控制 2 低位寄存器 (IPL2, 0xBD/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IPL2	-	-	-	-	-	-	LVDIPL	-
R/W	-	-	-	-	-	-	R/W	-
复位值	-	-	-	-	-	-	0	-

优先级等级配置对照表

IPL0/IPL1/IPL2	中断优先级等级
0	等级 0 (低)
1	等级 1 (高)

注：“高优先级中断”可以打断正在执行的“低优先级中断”，实现多层中断嵌套；在同一优先级情况下则按其“中断源列表”上的顺序排列。特别注意，一个中断产生后，若用户不进行处理，相同优先级的任何新中断将不会再产生。

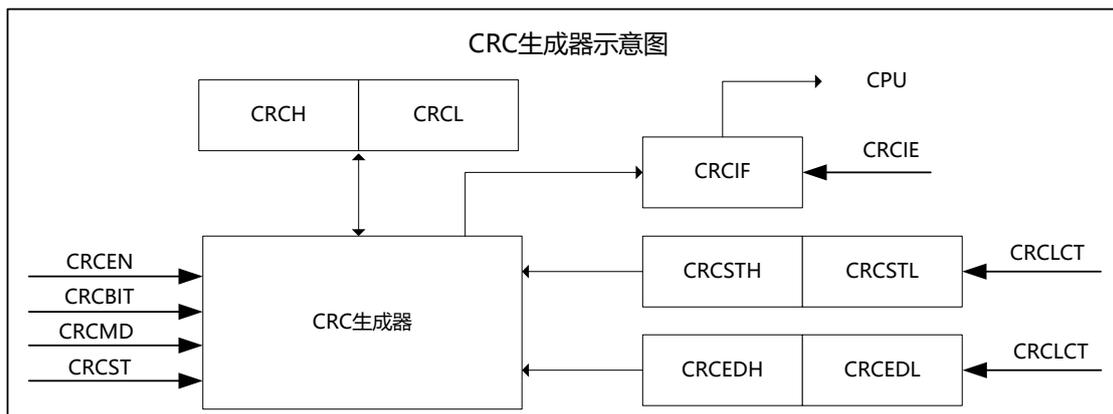


## 17 循环冗余校验 CRC

循环冗余校验(Cyclic Redundancy Check, CRC)是一种根据输入数据产生简短固定位数校验码的一种散列函数，主要用来提高系统的可靠性。

### 17.1 CRC 特性

集成硬件 16 位 CRC 单元，符合 CRC-16 标准，生成多项式为  $X_{16} + X_{15} + X_2 + 1$ ，即 0x8005，初值可设为任意值，默认为 0xFFFF。



### 17.2 CRC 相关寄存器

CRC 寄存器汇总表

地址 (SFRn /H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
XSFR/FF60	CRCCR	CRC 控制寄存器	CRCEN	-	CRCIF	CRCST	CRCMD	CRCLCT	-	CRCEN C	0-00 00-0
XSFR/FF62	CRCH	CRC 结果高字节寄存器	CRCH7	CRCH6	CRCH5	CRCH4	CRCH3	CRCH2	CRCH1	CRCH0	1111 1111
XSFR/FF63	CRCL	CRC 结果低字节寄存器	CRCL7	CRCL6	CRCL5	CRCL4	CRCL3	CRCL2	CRCL1	CRCL0	1111 1111
XSFR/FF64	CRCSTH	CRC 起始地址高字节寄存器	CRCST H7	CRCST H6	CRCST H5	CRCST H4	CRCST H3	CRCST H2	CRCST H1	CRCST H0	0000 0000
XSFR/FF65	CRCSTL	CRC 起始地址低字节寄存器	CRCST L7	CRCST L6	CRCST L5	CRCST L4	CRCST L3	CRCST L2	CRCST L1	CRCST L0	0000 0000
XSFR/FF66	CRCEDH	CRC 结束地址高字节寄存器	CRCED H7	CRCED H6	CRCED H5	CRCED H4	CRCED H3	CRCED H2	CRCED H1	CRCED H0	1111 1111
XSFR/FF67	CRCEDL	CRC 结束地址低字节寄存器	CRCED L7	CRCED L6	CRCED L5	CRCED L4	CRCED L3	CRCED L2	CRCED L1	CRCED L0	1111 1111

#### 17.2.1 CRC 控制寄存器 (CRCCR, 0xFF60/XSFR)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-------	-------	-------	-------	-------	-------	-------	-------



CRCCR	CRCEN	-	CRCIF	CRCST	CRCMD	CRCLCT	-	CRCENC
R/W	R/W	-	R/W	R/W	R/W	R/W	-	R/W
复位值	0	-	0	0	0	0	-	0

BIT[7] CRCEN – CRC 模块使能控制位

0: CRC 模块关闭;

1: CRC 模块打开。

BIT[5] CRCIF – CRC 中断标志位

0: 未启动或启动 CRC 校验后未完成, 由软件清 0;

1: 本次 CRC 校验计算完成, 由硬件置 1, 若同时 CRCIE 位置位, 将产生 CRC 中断, 则芯片会响应该中断请求。

BIT[4] CRCST – CRC 校验计算启动位

0: 本次 CRC 校验完成, 由硬件清 0;

1: 启动一次 CRC 校验或者处于 CRC 校验计算当中, 由软件置 1。

注: 写的话, 0 可以写入, 但无实际意义; 读的话, 该位根据 CRC 模块的实际状态返回 1/0。

BIT[3] CRCMD – CRC 校验模式选择位

0: 对整个 FLASHROM 区域或者整个 EEPROM 区域做 CRC 校验;

1: 对指定起始地址和结束地址的区域做 CRC 校验。

注: 该位=0 时, 起始地址寄存器和结束地址寄存器设定无效; =1 时, 起始地址寄存器和结束地址寄存器设定有效。

BIT[2] CRCLCT – CRC 校验区域选择位

0: 对 FLASHROM 区域做 CRC 校验;

1: 对 EEPROM 区域做 CRC 校验。

注: 这里 FLASHROM 有效区域指 0x0000~0x3FFF 的地址范围; EEPROM 有效区域指 0x000~0x0FF 的地址范围。

BIT[0] CRCENC – CRC 加密相关控制位

0: 应用于 FLASH 无加密的状态, 对指定起始地址和结束地址的区域做 CRC 校验;

1: 应用于 FLASH 有加密的状态, 结束地址无效。

注: 对于 FLASH 不加密的区域 (CRCENC=0) 做 CRC 校验时, 用户可以指定任意的起始地址和结束地址; 对于 FLASH 加密的区域 (CRCENC=1) 做 CRC 校验时, 用户只要指定一个起始地址即可, 无需指定结束地址, 芯片会自动对当前地址所在的扇区做 CRC 校验, 并给出 CRC 计算结果, 此时结束地址并不参与本次 CRC 的运算。

### 17.2.2 CRC 结果低字节寄存器 (CRCL, 0xFF63/XSFR)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CRCL	CRCL7	CRCL6	CRCL5	CRCL4	CRCL3	CRCL2	CRCL1	CRCL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	1	1	1	1	1	1	1

BIT[7:0] CRCL[7:0] – CRC 结果低字节

### 17.2.3 CRC 结果高字节寄存器 (CRCH, 0xFF62/XSFR)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CRCH	CRCH7	CRCH6	CRCH5	CRCH4	CRCH3	CRCH2	CRCH1	CRCH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	1	1	1	1	1	1	1



BIT[7:0] CRCH[7:0] – CRC 结果高字节

注：CRC[15:0]的内容可以在 CRC 校验启动前，由用户写入任意值，作为 CRC 初值，默认为 0xFFFF；其内容，在 CRC 校验计算过程中，不允许用户修改，否则 CRC 结果将不可预知。

#### 17.2.4 CRC 起始地址低字节寄存器 (CRCSTL, 0xFF65/XSFR)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CRCSTL	CRCSTL7	CRCSTL6	CRCSTL5	CRCSTL4	CRCSTL3	CRCSTL2	CRCSTL1	CRCSTL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] CRCSTL[7:0] – CRC 起始地址低字节

#### 17.2.5 CRC 起始地址高字节寄存器 (CRCSTH, 0xFF64/XSFR)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CRCSTH	CRCSTH7	CRCSTH6	CRCSTH5	CRCSTH4	CRCSTH3	CRCSTH2	CRCSTH1	CRCSTH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] CRCSTH[7:0] – CRC 起始地址高字节

#### 17.2.6 CRC 结束地址低字节寄存器 (CRCEDL, 0xFF67/XSFR)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CRCEDL	CRCEDL7	CRCEDL6	CRCEDL5	CRCEDL4	CRCEDL3	CRCEDL2	CRCEDL1	CRCEDL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	1	1	1	1	1	1	1

BIT[7:0] CRCEDL[7:0] – CRC 结束地址低字节

#### 17.2.7 CRC 结束地址高字节寄存器 (CRCEDH, 0xFF66/XSFR)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CRCEDH	CRCEDH7	CRCEDH6	CRCEDH5	CRCEDH4	CRCEDH3	CRCEDH2	CRCEDH1	CRCEDH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	1	1	1	1	1	1	1

BIT[7:0] CRCEDH[7:0] – CRC 结束地址高字节

注1：对起始地址和结束地址的设定，当 CRCLCT=1 (EEPROM) 时必须满足  $0x000 \leq CRCST[15:0] \leq CRCED[15:0] \leq 0x3FF$  的条件，当 CRCLCT=0 (FLASH) 时必须满足  $0x0000 \leq CRCST[15:0] \leq CRCED[15:0] \leq 0x3FFF$  的条件，否则 CRC 结果将不可预知；若 CRCST[15:0]和 CRCED[15:0]没有落在实际的物理地址上，CRC 结果也将不可预知。

注2：本芯片中，对“指定加密 FLASHROM 区域的地址范围”或者“包含加密 FLASHROM 区域的地址范围”，不能被 CRC 模块进行正确的 CRC 校验，故最终得到的 CRC 结果将是一个异常值。因此，当做 CRC 校验时，FLASH 区域当中全部或部分区域选择了“加密”时，推荐做法有下面两种：

- ✧ 对整个 FLASHROM 区域进行 CRC 校验，即先设置 CRCMD=0, CRCLCT=0, CRCENC=X, 再做 CRC 运算。
- ✧ 对分别对扇区进行 CRC 校验，通过指定（起始地址落在的某个加密扇区的）一个扇区进行 CRC 校验，不用指定结束地址，即先设置 CRCMD=1, CRCLCT=0, CRCENC=1, 再做 CRC 运算。

注3：CRC 模块计算区域的大小有限制，必须大于等于 4 个字节 (FLASH 是 4 字节；EEPROM 是 3 字节)；小于该字节数，CRC 计算结果将不可预知。



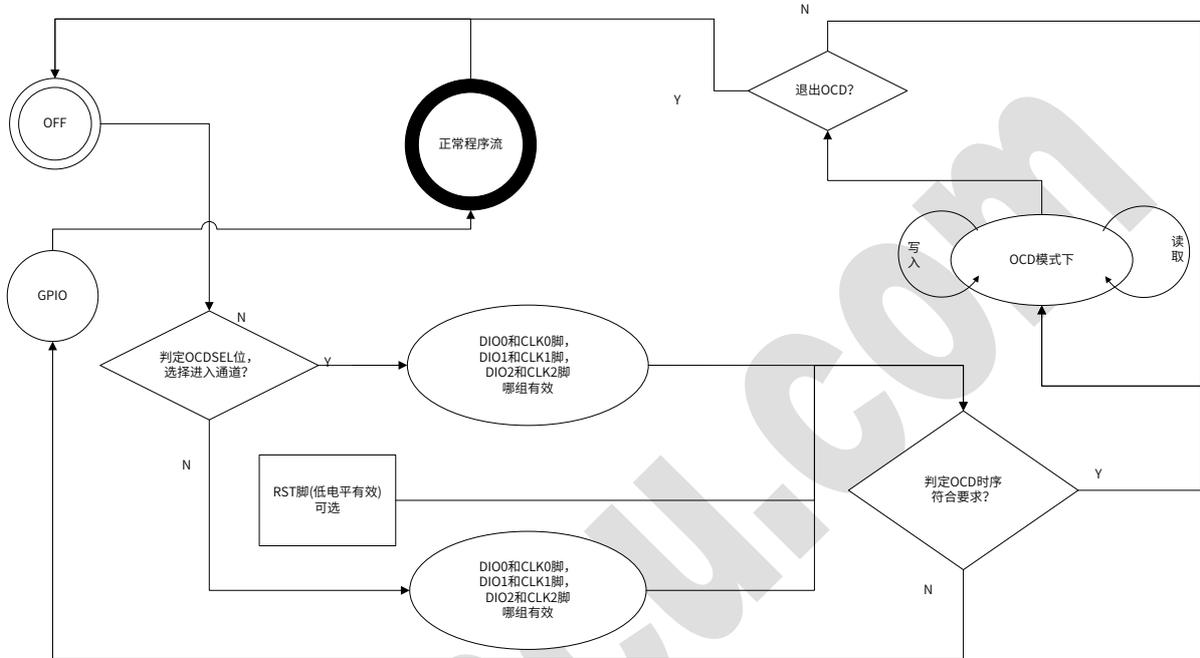
Sinomcu.com



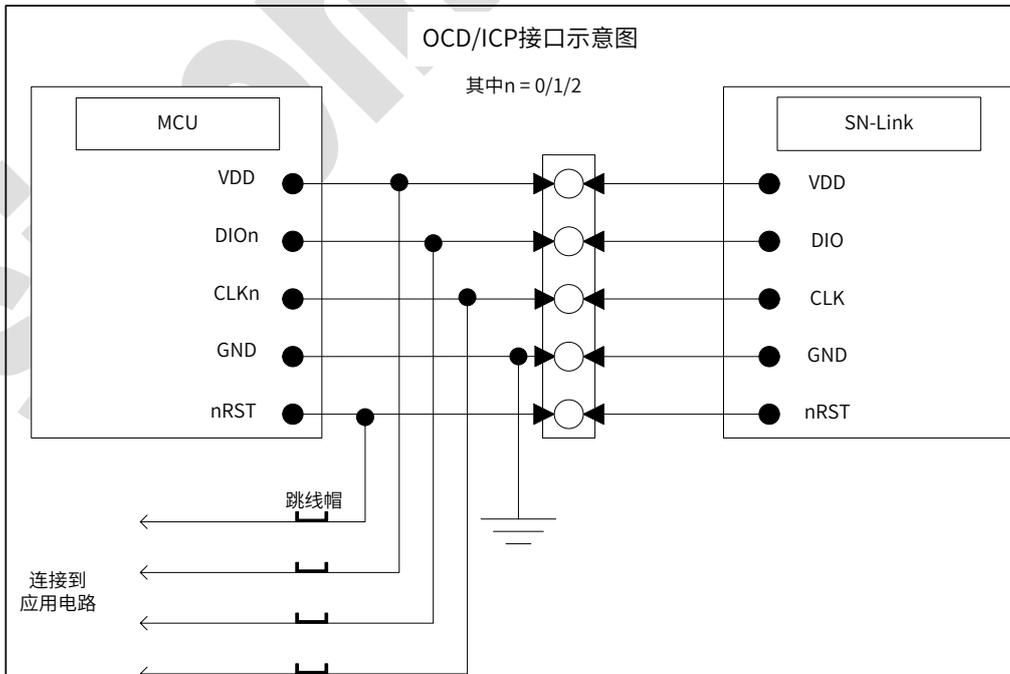
# 18 OCD 调试和 ICP 接口

本芯片芯片内嵌片上调试 (OCD) 功能和在电路编程 ICP 接口, 这为用户提供了低成本的调试方法, 并且本芯片充分考虑了用户的需求, 提供了精简的 2 线调试接口电路, 以及自动三选一的两组调试实物接口 (在用户配置字中设定, 默认为第一组调试接口)。当一组连接后, 另两组自动失效。在引脚复用的场合可以真正做到全仿真。

OCD状态机示意图



OCD/ICP接口示意图





## 片上调试 OCD 和在电路编程 ICP 接口整理

名称	对应 PIN	名称	对应 PIN	名称	对应 PIN	要求
电源正	VDD	电源正	VDD	电源正	VDD	必须
DIO2	P2.0	DIO1	P1.5	DIO0	P1.2	(三选一)
CLK2	P1.1	CLK1	P1.6	CLK0	P1.1	必须
nRST	P3.2	nRST	P3.2	nRST	P3.2	可选
电源负	GND	电源负	GND	电源负	GND	必须

注:  $DIO_n$  和  $CLK_n$  ( $n=0, 1, 2$ ) 的选择与“用户”实际电路预留调试的接口有关, 正常情况下选择其中一对使用即可。



## 19 片上编程 IAP

片上 FLASH IP 包含 3 个单元：主程序区（Main FLASH）、Info Block 区和 BootLoader 区。

*注：唤醒后用户需要等待至少 100us 后才能开始进行 IAP 操作。*

### Main FLASH 存储区

存储用户主代码，支持用户程序在带电工作中实时地读出或写入数据。当然，用户也可以通过相关的烧写工具，由外部 debug 接口直接操作。通过选择位 INFOS=0 进行选择 Main FLASH 区；在读数据时，利用 MOV<sub>C</sub> 指令间接寻址；写入数据时，通过操作地址寄存器 IAPAD、数据寄存器 IAPDB 进行。详细参见章节：IAP 操作标准流程。

FLASH 空间大小 16K bytes，64 bytes/页，支持页擦除/页烧写操作。

### Info Block 区

芯片数据存储区，可以存储芯片出厂数据，用户数据及配置信息，掉电数据保持。特定条件下，支持用户程序在带电工作中实时地读出或写入数据。当然，用户也可以通过相关的烧写工具，由外部 debug 接口直接操作。

通过选择位 INFOS=1 进行选择 Info Block 区。

Info Block 区，包括 E2prom 区、用户/工具配置区、加密配置区和出厂配置区。

此区域的操作流程及操作权限，请参见后续相关章节描述。

*注：出厂配置区仅在出厂时，由 sinomcu 写入；用户及烧写工具不允许操作此区域（操作无效）。*

### EEPROM 数据存储区

存储用户出厂数据及需要掉电保存的相关数据，支持用户程序在带电工作中实时地读出或写入数据。用户也可以通过相关的烧写工具，由外部 debug 接口直接操作。通过选择位 INFOS=1 进行选择 Info Block 区；在读数据时，利用 MOV<sub>C</sub> 指令间接寻址；写入数据时，通过操作地址寄存器 IAPAD、数据寄存器 IAPDB 进行。详细参见章节：IAP 操作标准流程。

EEPROM 空间大小 256 bytes，256 bytes/页，共 1 页，独立 EEPROM 编址，支持字节擦除。

### 用户/工具配置区

存储用户配置字，共 64bytes，64 bytes/页，共 1 页。通过选择位 INFOS=1 进行选择 Info Block 区，支持页擦除/页烧写操作，不支持字节擦除。

相关操作，请参见后续相关章节描述。

*注：用户禁止操作非有效区域，参见用户配置字章节。*

### 加密配置区

存储分区加密配置字，共 64bytes，64bytes/页，共 1 页。通过选择位 INFOS=1 进行选择 Info Block 区，支持页擦除/页烧写操作，不支持字节擦除。

相关操作，请参见后续相关章节描述。

*注：用户禁止操作非有效区域，参见用户配置字章节。*

### BootLoader 区

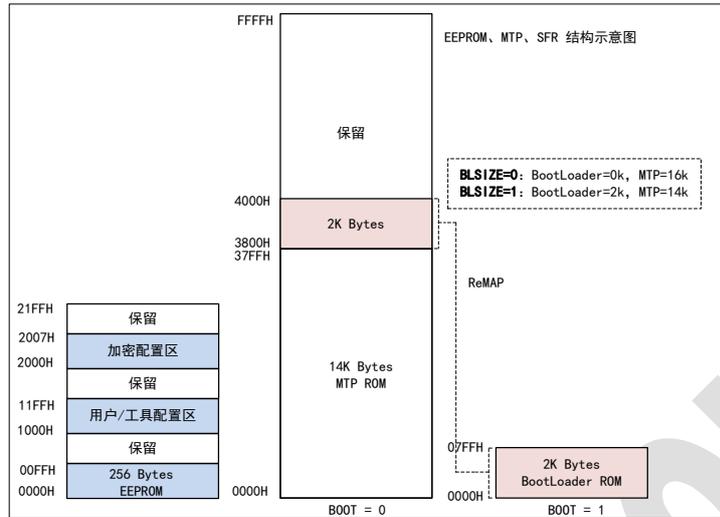
用于存储启动代码，功能与主程序区相似，只是空间大小不同，64bytes/页；此区域可实现 ISP 功能。

此区域物理上使用设计划分主程序区，通过配置用户配置字 BLSIZE 位 bootload 和 FLASH 之间关系大小，BOOTS 选择代码从 Bootloader 区启动或从 FLASH 启动。



注1: Main FLASH 区和 Info Block 区, 执行擦除操作后, 数据 bit 变为 0; 编程时写 0 变为 0, 写 1 变为 1。

注2: 数据寄存器 IAPDB 为数据锁存器, 对 IAPDB 的“写”操作实际将数据写入数据锁存区的对应地址中。



## 19.1 IAP 相关寄存器

IAP 操作寄存器汇总表

地址 (SFRn/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
SFR0/EF	IAPKEY	IAP 写保护寄存器	IAPKEY7	IAPKEY6	IAPKEY5	IAPKEY4	IAPKEY3	IAPKEY2	IAPKEY1	IAPKEY0	0000 0000
SFR0/EE	IAPMD	IAP 模式寄存器	IAPEN	-	CLRPL	-	-	PAGEERS	-	PROWR	0-0- -0-0
SFR0/ED	IAPDBH	IAP 高 8 位数据寄存器	IAPDBH7	IAPDBH6	IAPDBH5	IAPDBH4	IAPDBH3	IAPDBH2	IAPDBH1	IAPDBH0	0000 0000
SFR0/EC	IAPDBL	IAP 低 8 位数据寄存器	IAPDBL7	IAPDBL6	IAPDBL5	IAPDBL4	IAPDBL3	IAPDBL2	IAPDBL1	IAPDBL0	0000 0000
SFR0/EB	IAPADH	IAP 高地址寄存器	IAPAD15	IAPAD14	IAPAD13	IAPAD12	IAPAD11	IAPAD10	IAPAD9	IAPAD8	0000 0000
SFR0/EA	IAPADL	IAP 低地址寄存器	IAPAD7	IAPAD6	IAPAD5	IAPAD4	IAPAD3	IAPAD2	IAPAD1	IAPAD0	0000 0000
SFR0/E9	IAPCR	IAP 控制寄存器	IAPSTR	-	-	BLRS	IAPTRIG2	IAPTRIG1	IAPTRIG0	INFOS	0-0- 0000
SFR0/DF	IAPCR1	IAP 控制寄存器 1	BOOT	MFREF	IAPERR	-	PCFEN	UCFEN	-	MFREN	-00- 00-0

### 19.1.1 IAP 模式寄存器 (IAPMD, 0xEE/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IAPMD	IAPEN	-	CLRPL	-	-	PAGEERS	-	PROWR
R/W	R/W	-	R/W	-	-	R/W	-	R/W
复位值	0	-	0	-	-	0	-	0



- BIT[7] IAPEN – IAP 使能位  
 0: 由硬件自动清 0, 关闭 IAP 功能 (写一遍有效);  
 1: 打开 IAP 功能 (连续写两遍有效)。

注: 只有对该位连续写 2 条相同的指令, 才能使系统进行打开 IAP 使能位, 否则下个时钟周期到来后, 寄存器位中先前写入的值将被硬件自动清掉, 保持为 0, 这是一种简单的保护机制; 关闭时写一次 0 即可。

- BIT[5] CLRPL – IAP 清除页锁存  
 0: 由硬件自动清 0, 表示 IAP 清除页锁存流程已结束;  
 1: 写 1 后启动一次 IAP 清除页锁存。

- BIT[2] PAGEERS – 页擦除模式位  
 0: 由硬件自动清 0, 无 (写一遍有效);  
 1: 页擦除模式 (连续写两遍有效)。

注: 只有对该位连续写 2 条相同的指令, 才能使系统进入页擦模式, 否则下个时钟周期到来后, 寄存器位中先前写入的值将被硬件自动清掉, 保持为 0; 同样完成本次页擦模式后也会自动清 0, 这是一种简单的保护机制; 关闭时写一次 0 即可。

- BIT[0] PROWR – 编程 (写入) 模式位  
 0: 无 (写一遍有效);  
 1: 字节编程模式 (连续写两遍有效)。

注: 特别注意, IAPMD 寄存器中的每一位置 1 的话, 都需要连续写 2 次操作。另外, 以上 2 种 EEPROM/FLASH 的操作模式, 在同一时刻, 本芯片只允许进入一种模式执行, 否则芯片会出现不可预知的情况。

### 19.1.2 IAP 写保护寄存器 (IAPKEY, 0xEF/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IAPKEY	IAPKEY7	IAPKEY6	IAPKEY5	IAPKEY4	IAPKEY3	IAPKEY2	IAPKEY1	IAPKEY0
R/W	W	W	W	W	W	W	W	W
复位值	0	0	0	0	0	0	0	0

- BIT[7:0] IAPKEY[7:0] – IAP 保护锁  
 IAP 保护用, 当用户在启动 IAP 操作的时候, 必须先向其写入 0x5A, 紧接着写 0xA5, 解除 IAP 保护, 对 IAPCR 的操作才有效; 否则本次操作视为无效。

注: IAPKEY 为只写寄存器, 不可读取其值, 其保护的为 IAPCR 寄存器中的 IAPSTR 位和 IAPTRIG[2:0] 位。

### 19.1.3 IAP 高 8 位数据寄存器 (IAPDBH, 0xED/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IAPDBH	IAPDBH7	IAPDBH6	IAPDBH5	IAPDBH4	IAPDBH3	IAPDBH2	IAPDBH1	IAPDBH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

- BIT[7:0] IAPDBH[7:0] – IAP 写入高 8 位数据。

### 19.1.4 IAP 低 8 位数据寄存器 (IAPDBL, 0xEC/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IAPDBL	IAPDBL7	IAPDBL6	IAPDBL5	IAPDBL4	IAPDBL3	IAPDBL2	IAPDBL1	IAPDBL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0



BIT[7:0] IAPDBL[7:0] – IAP 写入低 8 位数据。

### 19.1.5 IAP 高地址寄存器 (IAPADH, 0xEB,SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IAPADH	IAPAD15	IAPAD14	IAPAD13	IAPAD12	IAPAD11	IAPAD10	IAPAD9	IAPAD8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[5:0] IAPAD[15:8] – IAP 写入地址的高 8 位。

### 19.1.6 IAP 低地址寄存器 (IAPADL, 0xEA/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IAPADL	IAPAD7	IAPAD6	IAPAD5	IAPAD4	IAPAD3	IAPAD2	IAPAD1	IAPAD0
R/W								
复位值	0	0	0	0	0	0	0	0

BIT[3:0] IAPAD[7:0] – IAP 写入地址的低 8 位。

### 19.1.7 IAP 控制寄存器 (IAPCR, 0xE9/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IAPCR	IAPSTR	-	-	BLRS	IAPTRIG2	IAPTRIG1	IAPTRIG0	INFOS
R/W	R/W	-	-	R/W	R/W	R/W	R/W	R/W
复位值	0	-	-	0	0	0	0	0

BIT[7] IAPSTR – IAP 操作启动控制位

0: 由硬件清 0, 表示当前不在 IAP 流程中, 或者 IAP 流程已结束;

1: 写 1 后启动一次 IAP 操作流程。

BIT[4] BLRS – IAP 读写区操作区域选择位, 软件置位清零, 需 INFOS=0, 此位才有效;

0: IAP 擦写及 MOVC 指令操作区域选择 FLASH 区;

1: IAP 擦写及 MOVC 指令操作区域选择 bootloader 区;

注: 当 INFOS=1, IAP 擦写及 MOVC 指令操作区域为 info block 区; 若 INFOS=0, 此位无效, IAP 擦写及 MOVC 指令操作区域固定为 FLASH 区。

BIT[3] IAPTRIG2 – IAP 操作执行控制位 2

0: 由硬件清 0, 无 IAP 操作, 或者当前 IAP 操作已结束;

1: 写 1 后, 硬件开始执行一次 IAP 操作。

BIT[2] IAPTRIG1 – IAP 操作执行控制位 1

0: 由硬件清 0, 无 IAP 操作, 或者当前 IAP 操作已结束;

1: 写 1 后, 硬件开始执行一次 IAP 操作。

BIT[1] IAPTRIG0 – IAP 操作执行控制位 0

0: 由硬件清 0, 无 IAP 操作, 或者当前 IAP 操作已结束;

1: 写 1 后, 硬件开始执行一次 IAP 操作。

注: 特别注意, 为了防止芯片误动作, 以上四个位的设置需要配合 IAPKEY 寄存器的设置联动操作, 否则设置无效, 详细可以参考“IAP 操作标准流程”。



BIT[0] INFOS – IAP 读写区操作区域选择位，软件置位清零  
 0: IAP 擦写及 MOVX 指令操作区域选择 FLASH 区；  
 1: IAP 擦写及 MOVX 指令操作区域选择 EEPROM 区。

注 1: 因 FLASHROM 和 info block 地址重叠，芯片通过 INFOS 位识别 MOVX 指令访问的是 INFO 区还是 FLASH rom 区，擦写操作也通过此位识别 INFO 区还是 FLASH 区。

注 2: FLASH 和 bootloader 地址重叠，通过 BLRS 区分 FLASH 还是 bootloader

注 3: 当用 MOVX 指令读取“用户配置字”中设置了代码区加密功能时 (ENCR 位=1)，FLASH/bootloader 区读取到的内容将为零。

### 19.1.8 IAP 控制寄存器 1 (IAPCR1, 0xDF/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IAPCR1	BOOT	MFREF	IAPERR	-	PCFEN	UCFEN	BLREN	MFREN
R/W	R/W	R	R/W	-	R/W	R/W	R/W	R/W
复位值	0	0	0	-	0	0	0	0

BIT[7] BOOT– 启动 ROM 区域选择

0: 复位后 MCU 从 FLASH 区 0000H 启动；  
 1: 复位后 MCU 从 bootloader rom 区 0000H 启动；

注: 该位由复位 (软件复位除外) 后读取 USER\_OPT3->BOOTS (BIT7) 内容载入，软件复位 (SWRST) 此位保持不变。

注: CPU 复位后，硬件根据 BOOT 位 (非 BOOTS 位) 决定启动开始于 FLASH rom 还是 bootloader rom。

BIT[6] MFREF– main FLASH 区全擦标志位，硬件置 1 或清 0

0: FLASH 区未执行全部擦除；  
 1: FLASH 区执行了全部擦除操作 (对全部 FLASH 空间执行页擦除，此位会置位)。

BIT[5] IAPERR– IAP 错误标志位，硬件置 1，软件清 0

0: 无错误发生，或软件清零；  
 1: 发生 IAP 错误。  
 满足以下任意条件，硬件在 IAPSTR 置 1 (IAP 启动) 后置位 IAPERR。

- 1) 访问地址超限 (不存在的区域)
- 2) 访问地址为禁止访问区
- 3) 擦除或烧写的区域使能未开启
- 4) IAP 的命令未正确执行 (无效或执行错误)

注: 指令未正确执行: IAPKEY 的 4 次解锁 (IAPSRT、IAPTRIG0、IAPTRIG1、IAPTRIG3) 未按照正确顺序操作。

注: 此 IAPERR 标志，仅在 IAP 模块操作及 MOVX 指令操作满足上述条件才会置位；如果 PC 指针超限，此位不会置位，CPU 执行 nop 指令。

注: 此位软件清零，或 POR 复位，其他复位不能复位此标志位

BIT[3] PCFEN– 加密配置区 (password configuration block) 擦/写操作使能

0: 禁止擦/写加密配置区；  
 1: 允许擦/写加密配置区。

BIT[2] UCFEN– 用户配置区 (User configuration block) 擦/写操作使能



0: 禁止擦/写用户配置区;

1: 允许擦/写用户配置区。

BIT[1] BLREN– Bootloader rom 擦/写操作使能

0: 禁止擦/写 Bootloader rom 区;

1: 允许擦/写 Bootloader rom 区。

BIT[0] MFREN– FLASH rom 擦/写操作使能

0: 禁止擦/写 FLASH 区;

1: 允许擦/写 FLASH 区。

## 19.2 IAP 命令汇总

IAP 操作		IAPCR1 区域擦写门控				IAPCR 访问区选择		IAPMD IAP 使能&擦写模式 选择			IAPAD[H:L] 16 位地址	IAPDB 16/8 位数据
		PCFEN	UCFEN	BLREN	MFREN	BLRS	INFOS	IAPEN	PAGERS	PROWR	IAPAD[16:0]	IAPDB[15:8] / IAPDB[15:0]
96 位 UID	字节读	x	x	x	x	0	1	x	0	0	1828H~ 1833H	只读
FLASH rom	页擦	x	x	x	1	0	0	1	1	0	页地址,低 9 位清 0	0000H
	字写	x	x	x	1	0	0	1	0	1	待写字节地址	待写字节数据
	字节读	x	x	x	x	0	0	x	0	0	Movc 指令读取	Movc 指令读 取
Bootload er rom	字写	x	x	1	x	1	0	1	0	1	待写字节地址	待写字节数据
	字节读	x	x	x	x	1	0	x	0	0	Movc 指令读取	Movc 指令读 取
User cfg block	页擦	x	1	x	x	0	1	1	1	0	页地址,1000H	0000H
	字写	x	1	x	x	0	1	1	0	1	待写字节地址	待写字节数据
	字节读	x	x	x	x	0	1	x	0	0	Movc 指令读取	Movc 指令读 取
PSW cfg block	页擦	1	x	x	x	0	1	1	1	0	页地址,2000H	0000H
	字写	1	x	x	x	0	1	1	0	1	待写字节地址	待写字节数据
	字节读	x	x	x	x	0	1	x	0	0	Movc 指令读取	Movc 指令读 取
EEPROM block	页擦	x	x	x	x	0	1	1	1	0	页地址,0000H	00H
	字节写	x	x	x	x	0	1	1	0	1	待写字节地址	待写字节数据
	字节读	x	x	x	x	0	1	x	0	0	Movc 指令读取	Movc 指令读 取

注 1: 仅 FLASH 区支持全擦操作, 对 FLASH 执行全擦, 仅在 bootloader 区执行代码时有效, 当在 FLASH 区执行代码, 并对 FLASH 执行全擦将被忽略 (无效)。

注 2: FLASH 地址区间 0000H~3FFFH;



*Bootloader 地址区间 0000H~07FFH;*  
*96 位 UID 地址区间 1828H~1833H*  
*User cfg block 地址区间 1000H~11FFH*  
*PSW cfg block 地址区间 2000H~21FFH*  
*EEPROM block 地址区间 0000H~00FFH*

## 19.3 IAP 操作标准流程

### 19.3.1 EEPROM 读取

标准操作流程，如下：

- (1) 首先，关闭中断；若使用看门狗的话，必须将看门狗溢出时间调整到足够大；
- (2) 写 INFOS=1；
- (3) 用 MOVC 指令读取对应的 EEPROM 地址，若需要可以重复操作多次；
- (4) 完成后，退出本次 IAP 操作，打开中断；若使用看门狗的话，复原看门狗溢出时间。

### 19.3.2 FLASH 读取

标准操作流程，如下：

- (1) 首先，关闭中断；若使用看门狗的话，必须将看门狗溢出时间调整到足够大；
- (2) 写 INFOS=0, BLRS=0（默认情况下）；
- (3) 用 MOVC 指令读取对应的 FLASH 地址，若需要可以重复操作多次；
- (4) 完成后，退出本次 IAP 操作，打开中断；若使用看门狗的话，复原看门狗溢出时间。

### 19.3.3 EEPROM 字节擦除（1 字节）

标准操作流程，如下：

- (1) 首先，关闭中断；若使用看门狗的话，必须将看门狗溢出时间调整到足够大；
- (2) 写 INFOS=1；
- (3) 将 IAPEN 位重复执行 2 次置 1 操作，将 CLRPL 重复执行 2 次置 1 操作；
- (4) 向 IAPADL 中写入页擦除的 EEPROM 字节的地址低 8 位；
- (5) 向 IAPADH 中写入页擦除的 EEPROM 字节的地址高 8 位；
- (6) 向 IAPDBH 中写入 0 值；
- (7) 将 IAPEN 位重复执行 2 次置 1 操作，将 PAGEERS 重复执行 2 次置 1 操作；
- (8) 对 IAPKEY[7:0]写入 0x5A，再写入 0xA5，解锁后对 IAPSTR 位置 1；
- (9) 对 IAPKEY[7:0]写入 0x5A，再写入 0xA5，解锁后对 IAPTRIG0 位置 1；
- (10) 对 IAPKEY[7:0]写入 0x5A，再写入 0xA5，解锁后对 IAPTRIG1 位置 1；
- (11) 对 IAPKEY[7:0]写入 0x5A，再写入 0xA5，解锁后对 IAPTRIG2 位置 1；
- (12) 若一切正常，在 EEPROM 页擦除的典型时间后，原本置 1 的 IAPSTR 位将硬件清 0，表示本次页擦除成功，并退出本次 IAP 流程，用户可以通过查询该标志位来确认本次操作的完成情况；
- (13) 完成后，退出本次 IAP 操作，打开中断；若使用看门狗的话，复原看门狗溢出时间。



### 19.3.4 FLASH 页擦除 (64 字节)

标准操作流程, 如下:

- (1) 首先, 关闭中断; 若使用看门狗的话, 必须将看门狗溢出时间调整到足够大;
- (2) 写 INFOS=0, BLRS=0, MFREN/BLREN=1;
- (3) 将 IAPEN 位重复执行 2 次置 1 操作, 将 CLRPL 重复执行 2 次置 1 操作;
- (4) 向 IAPADL 中写入页擦除的 FLASH 字节的地址低 8 位; (地址必须是该页首地址)
- (5) 向 IAPADH 中写入页擦除的 FLASH 字节的地址高 8 位;
- (6) 向 IAPDBL 中写入 0 值; (数据填入必须是先低字节后高字节)
- (7) 向 IAPDBH 中写入 0 值;
- (8) 将 IAPEN 位重复执行 2 次置 1 操作, 将 PAGEERS 重复执行 2 次置 1 操作;
- (9) 对 IAPKEY[7:0]写入 0x5A, 再写入 0xA5, 解锁后对 IAPSTR 位置 1;
- (10) 对 IAPKEY[7:0]写入 0x5A, 再写入 0xA5, 解锁后对 IAPTRIG0 位置 1;
- (11) 对 IAPKEY[7:0]写入 0x5A, 再写入 0xA5, 解锁后对 IAPTRIG1 位置 1;
- (12) 对 IAPKEY[7:0]写入 0x5A, 再写入 0xA5, 解锁后对 IAPTRIG2 位置 1;
- (13) 若一切正常, 在 FLASH 页擦除的典型时间后, 原本置 1 的 IAPSTR 位将硬件清 0, 表示本次页擦除成功, 并退出本次 IAP 流程, 用户可以通过查询该标志位来确认本次操作的完成情况;
- (14) 完成后, 退出本次 IAP 操作, 打开中断; 若使用看门狗的话, 复原看门狗溢出时间。

注: 除 EEPROM 其他 INFO 区 (用户配置字、加密配置字) 操作同 FLASH。

### 19.3.5 EEPROM 写入 (1 字节)

标准操作流程, 如下:

- (1) 首先, 关闭中断, 将看门狗溢出时间调整到足够大;
- (2) 写 INFOS=1;
- (3) 将 IAPEN 位重复执行 2 次置 1 操作, 将 CLRPL 重复执行 2 次置 1 操作;
- (4) 向 IAPADL 中写入预备写入的 EEPROM 字节的地址低 8 位;
- (5) 向 IAPADH 中写入预备写入的 EEPROM 字节的地址高 8 位;
- (6) 向 IAPDBH 中写入预备写入的新数值;
- (7) 保持 IAPEN 位打开, 将 PROWR 位重复执行 2 次置 1 操作;
- (8) 对 IAPKEY[7:0]写入 0x5A, 再写入 0xA5, 解锁后对 IAPSTR 位置 1;
- (9) 对 IAPKEY[7:0]写入 0x5A, 再写入 0xA5, 解锁后对 IAPTRIG0 位置 1;
- (10) 对 IAPKEY[7:0]写入 0x5A, 再写入 0xA5, 解锁后对 IAPTRIG1 位置 1;
- (11) 对 IAPKEY[7:0]写入 0x5A, 再写入 0xA5, 解锁后对 IAPTRIG2 位置 1;
- (12) 若一切正常, 在 FLASH 编程的典型时间后, 原本置 1 的 IAPSTR 位将硬件清 0, 表示本次 EEPROM 字节写入成功, 并退出本次 IAP 流程, 用户可以通过查询该标志位来确认本次操作的完成情况;
- (13) 完成后, 退出本次 IAP 操作, 打开中断; 若使用看门狗的话, 复原看门狗溢出时间。

### 19.3.6 FLASH 写入 (n\*2 字节)

标准操作流程, 如下:

- (1) 首先, 关闭中断, 将看门狗溢出时间调整到足够大;



- (2) 写 INFOS=0, BLRS=0, MFREN/BLREN=1;
- (3) 将 IAPEN 位重复执行 2 次置 1 操作, 将 CLRPL 重复执行 2 次置 1 操作;
- (4) 向 IAPADL 写入预备写入的 FLASH 数据的起始地址低 8 位; (起始地址低 8 位的 bit0 必须为 0)
- (5) 向 IAPADH 写入预备写入的 FLASH 数据的起始地址高 8 位;
- (6) 向 IAPDBL 写入预备写入的数据低 8 位; (数据填入必须是先低字节后高字节)
- (7) 向 IAPDBH 写入预备写入的数据高 8 位;
- (8) 重复步骤 (6) (7), 写入下一数据, 直到写完预备写入的所有数据 (不能超过该页地址限制);
- (9) 保持 IAPEN 位打开, 将 PROWR 位重复执行 2 次置 1 操作;
- (10) 对 IAPKEY[7:0]写入 0x5A, 再写入 0xA5, 解锁后对 IAPSTR 位置 1;
- (11) 对 IAPKEY[7:0]写入 0x5A, 再写入 0xA5, 解锁后对 IAPTRIG0 位置 1;
- (12) 对 IAPKEY[7:0]写入 0x5A, 再写入 0xA5, 解锁后对 IAPTRIG1 位置 1;
- (13) 对 IAPKEY[7:0]写入 0x5A, 再写入 0xA5, 解锁后对 IAPTRIG2 位置 1;
- (14) 若一切正常, 在 FLASH 编程的典型时间后, 原本置 1 的 IAPSTR 位将硬件清 0, 表示本次 FLASH 编程成功, 并退出本次 IAP 流程, 用户可以通过查询该标志位来确认本次操作的完成情况;
- (15) 完成后, 退出本次 IAP 操作, 打开中断; 若使用看门狗的话, 复原看门狗溢出时间。

## 19.4 Bootloader 区

Bootloader区为完全独立于FLASH (主程序区) 存储区, 空间大小为2K。

通过USER\_OPT3->BOOTS和IAPCR1->BOOT控制CPU从bootloader区启动执行代码或从FLASH区执行代码。

### 19.4.1 启动选择

本产品提供 2 种启动选择, 通过 IAPCR1->BOOT (bit7) 控制选择 cpu 从 FLASH rom 0000H 启动或 bootloader rom 0000H 启动。

复位后 (软件复位除外), user\_opt3->BOOTS (bit7) 加载到 IAPCR1->BOOT (bit7); 软件复位不执行此加载。

### 19.4.2 IAP 控制寄存器 1 (IAPCR1, 0xED/SFR0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IAPCR1	BOOT		IAPERR	-	PCFEN	UCFEN	BLREN	MFREN
R/W	R/W		R/W	-	R/W	R/W	R/W	R/W
复位值	0		0	-	0	0	0	0

BIT[7] BOOT- 启动 ROM 区域选择

0: 复位后 MCU 从 FLASH rom 区 0000H 启动;

1: 复位后 MCU 从 bootloader rom 区 0000H 启动;

注: 该位由复位 (软件复位除外) 后读取 USER\_OPT3->BOOTS (BIT7) 内容载入, 软件复位 (SWRST) 此位保持不变。

注: CPU 复位后, 硬件根据 BOOT 位 (非 BOOTS 位) 决定启动开始于 FLASH rom 还是 bootloader rom。

### 19.4.3 Bootloader 区内的擦除/读写操作

在 bootloader 区内, 通过扇区擦除功能, 可以擦除用户程序区和类 EEPROM 区的任意扇区, 以及 bootloader 区



(但对程序运行所在的 page 执行 page Erase 无效)，可以擦除 INFO 区内用户/工具配置区（用户配置字）和加密配置区（加密配置字），不能擦除 INFO 区内出厂配置区。

在 bootloader 区内，不支持整体擦除功能；

在 bootloader 区内，通过 IAP 功能，可以向主程序存储区及 bootloader 区的空白区域写入数据，也可以向 INFO 区内的类 EEPROM 区和用户/工具配置区的空白区域写入数据。

在 bootloader 区内，可以用 MOVC 指令读取用户程序区、EEPROM 区以及 bootloader 区内的内容。在 bootloader 区内，也可以读取用户/工具配置区、加密配置区和出厂配置区的 96 位 UID。

必须指出，对 INFO 区内的用户配置字的写操作只针对有效地址有效，禁止对保留地址进行写操作。

以上操作仅限于目标区域没有进行代码保护的情况。当目标区域处于代码保护模式时，参见 *FLASH 存储器* 部分关于代码保护方式的描述。

#### 19.4.4 bootloader 应用

bootloader 区可实现 ISP 功能，用户可以把引导程序代码放在 bootloader 区（最大 2K），用户的引导代码可以通过烧录器或 ICP 下载至 bootloader 区。

一般应用中，ISP 通过 PC 端工具和 MCU 连接并通讯，PC 可以通过串口传输待更新的用户代码至 MCU。引导区代码接收并调用 IAP 模块烧录至指定用户代码区。

相关的引导代码例程及 PC 端软件，请参见<待补充>。

### 19.5 INFO 区

EEPROM 区，支持 IAP 扇区擦除和字节写入/读取。

用户/工具配置区，支持 IAP 扇区擦除和字写入/读取。

出厂配置区，不支持 IAP 扇区擦除和字写入/读取；但 96 位 UID 支持只读操作。

加密配置区，仅在 bootloader 区执行 FLASH 全部扇区擦除后，加密位才支持 IAP 扇区擦除和字写入；读取不受此条件限制。

在访问 INFO 区之前，应首先将寄存器 IAPCR->INFOS 控制位置 1；在访问 INFO 区之后，应将 IAPCR->INFOS 位清 0。INFO 区内只能存放数据，不能存放用户程序。CPU 不会到此区域内取指令。

#### 19.5.1 加密控制字解锁

加密控制字(加密配置区)，在未执行所有的 FLASH 区全部擦除动作前，不支持擦写操作（操作无效）；

IAP 方式下，满足下述两个条件，解锁加密控制字擦写权限：

- 1、程序在 bootloader 区执行
- 2、执行所有的 FLASH 全部扇区擦除

注 1：烧写器进行 ICP 方式烧写时，与 IAP 不同，有专有的烧写模式及时序。

注 2：当程序在 FLASH 区执行时，不允许对用户/工具配置区、出厂配置区、加密配置区进行擦写操作（操作无效）。

#### 19.5.2 Flash 加密/解密操作

Main flash 区的加密/解密操作如下：

- 1、解锁加密控制字擦写权限（参见 18.5.1 加密控制字解锁）。



- 2、根据需求写入对应控制字（加密/解密），同 EEPROM 区操作。
- 3、芯片复位，重新加载新的配置字，并生效。

注：被加密的 FLASH 区不支持扇区擦除和字节写入/读取；

### 19.5.3 用户/工具配置区操作

用户配置字操作如下：

- 1、对用户/工具配置区，执行扇区擦除操作；
- 2、根据需求写入对应控制字，同 EEPROM 区操作。
- 3、芯片复位，重新加载新的配置字，并生效。

## 19.6 用户主程序区（FLASH）

主程序区内，通过 IAP 扇区擦除功能，可以擦除用户程序区（除正在执行代码所在的扇区以及加密的区块）、bootloader 区（加密后不可擦除）、用户配置区和 EEPROM 区的任意扇区；

注 1：主程序区内，对当前执行代码所在扇区，执行扇区擦除，操作将无效；

注 2：主程序区内，对被加密的区块执行擦除无效

注 3：主程序区内，禁止擦写分区加密控制字。

用户程序区内，对 info 区操作，必须符合对应条件，请参见《4.3 INFO 区》相关说明。

用户程序区内的擦除和读写操作参见 IAP 章节。

## 19.7 User option 控制

USER\_OPT3 配置字

地址/位	符号	功能说明
1006H	USER_OPT3	代码加密/OTP 使能/ OCD 访问控制
BIT[7]	BOOTS	BOOT- 启动 ROM 区域选择 0: 复位后 MCU 从 FLASH rom 区 0000H 启动； 1: 复位后 MCU 从 bootloader rom 区 0000H 启动； 注：除软复位（SWRST）外，其他复位此位内容加载至 IAPCR1->BOOT(BIT7)。

## 19.8 复位说明

软件复位与其他复位区别：

- 1、Warmup 时间不同，请参见用户手册《复位预热章节》；
- 2、Option 加载不同，除软件复位的其他复位，会将 USER\_OPT3->BOOTS 值加载至 IAPCR1->BOOT(BIT7)，BOOT 控制 CPU 启动位置：FLASH rom 或 bootloader rom 的 0000H。软件复位的 option 加载相位，不加载 BOOTS 位，BOOT 位保持不变，其他 option 加载同其他复位。



## 19.9 ICP、IAP 和 ISP

ICP:

通过 JTAG 进入 ctrl 模式，再进入烧写模式

操作地址：0000H~3FFFH，访问 16K 的 FLASH 空间；

实现方式：通过仿真器或烧写器，根据用户设置区域（main FLASH）直接烧写 16K FLASH 空间。

IAP:

通过片上的 IAP 模块及对应的 IAP 命令操作方式，配合用户代码实现指定区域 rom 擦写；

可操作地址如下：

0000H~3FFFH；

ISP:

通过协议指令控制 MCU 通讯接口连接，操作 FLASH 进行烧录。



## 20 附录

## 20.1 指令集总览表

助记符	说明	长度(B)	周期(T)
ADD A,Rn	寄存器 Rn 和累加器 ACC 相加, 结果存到 ACC	1	2
ADD A, direct	直接寻址字节和累加器 ACC 相加, 结果存到 ACC	2	2
ADD A, @Ri	内部 RAM Ri 和累加器 ACC 相加, 结果存到 ACC	1	3
ADD A, #data	立即数和累加器 ACC 相加, 结果存到 ACC	2	1
ADDC A,Rn	寄存器 Rn 和累加器 ACC 相加 (带 C 标志), 结果存到 ACC	1	2
ADDC A, direct	直接寻址字节和累加器 ACC 相加 (带 C 标志), 结果存到 ACC	2	2
ADDC A, @Ri	内部 RAM Ri 和累加器 ACC 相加 (带 C 标志), 结果存到 ACC	1	2
ADDC A, #data	立即数和累加器 ACC 相加 (带 C 标志), 结果存到 ACC	2	1
SUBB A,Rn	累加器 ACC 减寄存器 Rn (带 C 标志), 结果存到 ACC	1	1
SUBB A, direct	累加器 ACC 减直接寻址字节 (带 C 标志), 结果存到 ACC	2	2
SUBB A, @Ri	累加器 ACC 减内部 RAM Ri (带 C 标志), 结果存到 ACC	1	3
SUBB A, #data	累加器 ACC 减立即数 (带 C 标志), 结果存到 ACC	2	1
INCA	累加器 ACC 加 1, 结果存到 ACC	1	1
INC Rn	寄存器 Rn 加 1, 结果存到 Rn	1	2
INC direct	直接寻址字节加 1, 结果存到直接选址字节	2	2
INC @Ri	内部 RAM Ri 加 1, 结果存到 Ri	1	3
INC DPTR	数据指针 DPTR/DPRT1 加 1, 结果存到 DPTR/DPRT1	1	1
DEC A	累加器 ACC 自减 1, 结果存到 ACC	1	1
DEC Rn	寄存器 Rn 自减 1, 结果存到 Rn	1	2
DEC direct	直接寻址字节减 1, 结果存到直接选址字节	2	2
DEC @Ri	内部 RAM Ri 减 1, 结果存到 Ri	1	3
MUL AB 8 X 8	累加器 ACC 乘寄存器 B	1	2
16 X 8			-
DIV AB 8 / 8	累加器 ACC 除寄存器 B	1	2
16 / 8			-
DA A	BCD 码调整	1	1
ANL A, Rn	累加器 ACC 与寄存器 Rn, 结果存到 ACC	1	2



ANL A, direct	累加器 ACC 与直接寻址字节, 结果存到 ACC	2	2
ANL A, @Ri	累加器 ACC 与内部 RAM Ri, 结果存到 ACC	1	3
ANL A, #data	累加器 ACC 与立即数, 结果存到 ACC	2	1
ANL direct, A	直接寻址字节与累加器 ACC, 结果存到直接寻址字节	2	2
ANL direct, #data	直接寻址字节与立即数, 结果存到直接寻址字节	3	2
ORL A, Rn	累加器 ACC 或寄存器 Rn, 结果存到 ACC	1	2
ORL A, direct	累加器 ACC 直接寻址字节, 结果存到 ACC	2	2
ORL A, @Ri	累加器 ACC 或内部 RAM Ri, 结果存到 ACC	1	3
ORL A, #data	累加器 ACC 或立即数, 结果存到 ACC	2	1
ORL direct, A	直接寻址字节或累加器 ACC, 结果存到直接寻址字节	2	2
ORL direct, #data	直接寻址字节或立即数, 结果存到直接寻址字节	3	2
XRL A, Rn	累加器 ACC 异或寄存器 Rn, 结果存到 ACC	1	2
XRL A, direct	累加器 ACC 直接寻址字节, 结果存到 ACC	2	2
XRL A, @Ri	累加器 ACC 异或内部 RAM Ri, 结果存到 ACC	1	3
XRL A, #data	累加器 ACC 异或立即数, 结果存到 ACC	2	1
XRL direct, A	直接寻址字节异或累加器 ACC, 结果存到直接寻址字节	2	2
XRL direct, #data	直接寻址字节异或立即数, 结果存到直接寻址字节	3	2
CLR A	对 ACC 清零	1	1
CPL A	对 ACC 取反	1	1
RL A	累加器 ACC 循环左移	1	1
RLC A	累加器 ACC 循环左移 (带 C 标志)	1	1
RR A	累加器 ACC 循环右移	1	1
RRC A	累加器 ACC 循环右移 (带 C 标志)	1	1
SWAP A	交换累加器 ACC 的高低半字节, 结果存到 ACC	1	1
MOV A, Rn	将寄存器 Rn 送到累加器 ACC	1	1
MOV A, direct	将直接寻址字节送到累加器 ACC	2	1
MOV A, @Ri	将内部 RAM Ri 送到累加器 ACC	1	2
MOV A, #data	将立即数送到累加器 ACC	2	1
MOV Rn, A	将累加器 ACC 送到寄存器 Rn	1	1
MOV Rn, direct	将直接寻址直接送到寄存器 Rn	2	2
MOV Rn, #data	将立即数送到寄存器 Rn	2	1
MOV direct, A	将累加器 ACC 送到直接寻址字节	2	1
MOV direct, Rn	将寄存器 Rn 送到直接寻址字节	2	2
MOV direct1, direct2	将直接寻址字节 2 送到直接寻址字节 1	3	2
MOV direct, @Ri	将内部 RAM Ri 送到直接寻址字节	2	3
MOV direct, #data	将立即数送到直接寻址字节	3	1



MOV @Ri, A	将累加器 ACC 送到内部 RAM Ri	1	2
MOV @Ri, direct	将直接寻址字节送到内部 RAM Ri	2	3
MOV @Ri, #data	将立即数送到内部 RAM Ri	2	2
MOV DPTR, #data16	将 16 位立即数送到数据指针	3	1
MOVC A, @A+DPTR	将代码程序送到累加器 ACC (相对数据指针)	1	2
MOVC A, @A+PC	将代码程序送到累加器 ACC (相对程序计数器)	1	2
MOVX A, @Ri	通过内部 RAM Ri 将 8 位外存送到累加器 ACC	1	2
MOVX A, @DPTR	通过数据指针 DPTR 将 16 位外存送到累加器 ACC	1	1
MOVX @Ri, A	通过内部 RAM Ri 将累加器 ACC 送到 8 位外存	1	2
MOVX @DPTR, A	通过数据指针 DPTR 将累加器 ACC 送到 16 位外存	1	1
PUSH direct	直接寻址字节压入栈顶	2	2
POP direct	栈顶弹至直接寻址字节	2	2
XCH A, Rn	累加器 ACC 与寄存器 Rn 交换	1	2
XCH A, direct	累加器 ACC 与直接寻址字节交换	2	2
XCH A, @Ri	累加器 ACC 与内部 RAM Ri 交换	1	3
XCHD A, @Ri	累加器 ACC 低 4 位与内部 RAM Ri 低 4 位交换	1	3
ACALL addr11	2KB 内绝对调用	2	2
LCALL addr16	64KB 内长调用	3	1
RET	子程序返回	1	3
RETI	中断返回	1	3
AJMP addr11	2KB 内绝对转移	2	1
LJMP addr16	64KB 内长转移	3	5
SJMP rel	相对短转移	2	1
JMP @A+DPTR	相对长转移	1	2
JZ rel (不发生转移) (发生转移)	累加器为零转移	2	2 2
JNZ rel (不发生转移) (发生转移)	累加器为非零转移	2	2 2
JC rel (不发生转移) (发生转移)	C 置位转移	2	2 2
JNC rel (不发生转移) (发生转移)	C 清零转移	2	2 2
JB bit, rel(不发生转移) (发生转移)	直接寻址位置位转移	3	2 2
JNB bit, rel (不发生转移) (发生转移)	直接寻址位清零转移	3	2 2
JBC bit, rel (不发生转移) (发生转移)	直接寻址位置位转移并清该位	3	2 2
CJNE A, direct, rel (不发生转移)	累加器与直接寻址字节不等转移	3	2



(发生转移)			2
CJNE A, #data, rel (不发生转移)	累加器与立即数不等转移	3	2
(发生转移)			2
CJNE Rn, #data, rel (不发生转移)	寄存器与立即数不等转移	3	2
(发生转移)			2
CJNE @Ri, #data, rel (不发生转移)	内部 RAM 与立即数不等转移	3	3
(发生转移)			3
DJNZ Rn, rel (不发生转移)	寄存器减 1 不为零转移	2	2
(发生转移)			2
DJNZ direct, rel (不发生转移)	直接寻址字节减 1 不为零转移	3	2
(发生转移)			2
NOP	空操作	1	1
CLR C	C 清零	1	1
CLR bit	直接寻址位清零	2	2
SETB C	C 置位	1	1
SETB bit	直接寻址位置位	2	2
CPL C	C 取反	1	1
CPL bit	直接寻址位取反	2	2
ANL C, bit	C 逻辑与直接寻址位	2	2
ANL C, /bit	C 逻辑与直接寻址位的反	2	2
ORL C, bit	C 逻辑或直接寻址位	2	2
ORL C, /bit	C 逻辑或直接寻址位的反	2	2
MOV C, bit	直接寻址位送 C	2	2
MOV bit, C	C 送直接寻址位	2	2



## 20.2 特殊功能寄存器总览表 (SFR0)

地址 (H)	SFR 号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
FF	SFR0	ADRH	ADC 转换结果高位寄存器	ADR11	ADR10	ADR9	ADR8	ADR7	ADR6	ADR5	ADR4	xxxx xxxx
				-	-	-	-	ADR11	ADR10	ADR9	ADR8	---- xxxx
FE	SFR0	ADRL	ADC 转换结果高位寄存器	-	-	-	-	ADR3	ADR2	ADR1	ADR0	---- xxxx
				ADR7	ADR6	ADR5	ADR4	ADR3	ADR2	ADR1	ADR0	xxxx xxxx
FD	SFR0	ADRIOS1	ADC 端口配置寄存器 1	-	-	EX_VREF_EN	-	-	-	AN9EN	AN8EN	--0- ----
FC	SFR0	-	-	-	-	-	-	-	-	-	-	---- ----
FB	SFR0	ADCMPLH	ADC 数字比较高位寄存器	ADCMP15	ADCMP14	ADCMP13	ADCMP12	ADCMP11	ADCMP10	ADCMP9	ADCMP8	0000 0000
FA	SFR0	ADCMPL	ADC 数字比较低位寄存器	ADCMP7	ADCMP6	ADCMP5	ADCMP4	ADCMP3	ADCMP2	ADCMP1	ADCMP0	0000 0000
F9	SFR0	PWRRCR1	高级能耗控制寄存器 1	-	IIC_CLKEN	CRC_CLKEN	-	-	T3_CLKEN	T2_CLKEN	T01_CLKEN	-11- -111
F8	SFR0	PWRRCR	高级能耗控制寄存器	-	-	PWM1_CLKEN	PWM0_CLKEN	SPI_CLKEN	ADC_CLKEN	-	OCD_CLKEN	--11 11-1
F7	SFR0	LVDCR1	LVD 控制寄存器 1	-	-	-	-	LVDDFLTS2	LVDDFLTS1	LVDDFLTS0	LVDAFLTE	---- 0000
F6	SFR0	LVDCR	LVD 控制寄存器	LV DEN	-	LV DIF	-	LV DVS2	LV DVS1	LV DVS0	LV DF	0-0- 000-
F5	SFR0	OSADJCR	零点偏移修正寄存器	OSADJEN	OSADJTD	OSADJT5	OSADJT4	OSADJT3	OSADJT2	OSADJT1	OSADJT0	0xxx xxxx
F4	SFR0	ADCCR3	ADC 控制寄存器 3	ADGAIN1	ADGAIN0	-	-	-	-	-	ADVREFEN	00-- ----0
F3	SFR0	ADCCR2	ADC 控制寄存器 2	ADEOCIE	ADCMPIE	ADCM PEN	ADCM PDIR	ADCONT	ADTRG2	ADTRG1	ADTRG0	0000 0000
F2	SFR0	ADCCR1	ADC 控制寄存器 1	ADRSEL	ADCKS2	ADCKS1	ADCKS0	-	ADSPS2	ADSPS1	ADSPS0	0000 -111
F1	SFR0	ADCCR0	ADC 控制寄存器 0	ADCHS3	ADCHS2	ADCHS1	ADCHS0	VRS1	VRS0	GODONE	ADEN	1111 1110
F0	SFR0	B	B 寄存器	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	0000 0000



地址 (H)	SFR 号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
EF	SFR0	IAPKEY	IAP 写保护寄存器	IAPKEY7	IAPKEY6	IAPKEY5	IAPKEY4	IAPKEY3	IAPKEY2	IAPKEY1	IAPKEY0	0000 0000
EE	SFR0	IAPMD	IAP 模式寄存器	IAPEN	-	CLRPL	-	-	PAGEERS	-	PROWR	0-0- -0-0
ED	SFR0	IAPDBH	IAP 高 8 位数据寄存器	IAPDBH7	IAPDBH6	IAPDBH5	IAPDBH4	IAPDBH3	IAPDBH2	IAPDBH1	IAPDBH0	0000 0000
EC	SFR0	IAPDBL	IAP 低 8 位数据寄存器	IAPDBL7	IAPDBL6	IAPDBL5	IAPDBL4	IAPDBL3	IAPDBL2	IAPDBL1	IAPDBL0	0000 0000
EB	SFR0	IAPADH	IAP 高地址寄存器	IAPAD15	IAPAD14	IAPAD13	IAPAD12	IAPAD11	IAPAD10	IAPAD9	IAPAD8	0000 0000
EA	SFR0	IAPADL	IAP 低地址寄存器	IAPAD7	IAPAD6	IAPAD5	IAPAD4	IAPAD3	IAPAD2	IAPAD1	IAPAD0	0000 0000
E9	SFR0	IAPCR	IAP 控制寄存器	IAPSTR	-	-	BLRS	IAPTRIG2	IAPTRIG1	IAPTRIG0	INFOS	0--0 0000
E8	SFR0	RSTFLAG	复位标志寄存器	PORF	LVRF	RSTF	WDTRF	SWRF	-	-	SWRST	0000 0--0
E7	SFR0	IICAMR	IIC 辅地址掩码寄存器	IICAM6	IICAM5	IICAM4	IICAM3	IICAM2	IICAM1	IICAM0	-	0000 0000
E6	SFR0	IICAAR	IIC 辅地址寄存器	IICAA6	IICAA5	IICAA4	IICAA3	IICAA2	IICAA1	IICAA0	-	0000 0000
E5	SFR0	IICDR	IIC 数据寄存器	IICD7	IICD6	IICD5	IICD4	IICD3	IICD2	IICD1	IICD0	0000 0000
E4	SFR0	IICAR	IIC 地址寄存器	IICA6	IICA5	IICA4	IICA3	IICA2	IICA1	IICA0	MCMDS	0000 0000
E3	SFR0	IICSR	IIC 状态寄存器	TACKS	IICARLO	IICSTRF	IICSTPF	SADMF	SDTFF	SCMDF	RACKF	---0 0000
E2	SFR0	IICCR1	IIC 控制寄存器 1	-	-	-	MNASTPE	MTSAA	MTSTP	MTDTA	MRDTA	-000 0000
E1	SFR0	IICCR0	IIC 控制寄存器 0	IICEN	IICMOD	IICSPD	IICSYN	IICGCE	IICCHS1	IICCHS0	IICRUS	0000 0000
E0	SFR0	ACC	累加器	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	0000 0000
DF	SFR0	IAPCR1	IAP 控制寄存器寄存器 1	-	MFREF	IAPERR	-	PCFEN	UCFEN	-	MFREN	-00- 00-0
DE	SFR0	PWM1MANCR1	PWM1 手动输出控制寄存器 1	-	-	PWMOCN	PWMOBN	PWMOAN	PWMOC	PWMOB	PWMOA	--00 0000



地址 (H)	SFR 号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
DD	SFR0	PWM1MANCR0	PWM1 手动输出控制寄存器 0	-	-	MANCN	MANBN	MANAN	MANC	MANB	MANA	--00 0000
DC	SFR0	PWM1FBCR	PWM1 刹车控制寄存器	-	-	FB1EN	FB1S	FB1FLT1	FB1FLT0	-	FBSTA	--00 00-0
DB	SFR0	PWM1CR2	PWM1 控制寄存器 2	TRGTIM1	TRGTIM0	PTRGAD	ZTRGAD	PLDEN	OSYNC	-	-	0000 00--
DA	SFR0	PWM1CR1	PWM1 控制寄存器 1	-	-	PWMCNS	PWMBNS	PWMANS	PWMCs	PWMBs	PWMAS	--00 0000
D9	SFR0	PWM1RLEN	PWM1 寄存器修改重载控制寄存器	PWMRL7	PWMRL6	PWMRL5	PWMRL4	PWMRL3	PWMRL2	PWMRL1	PWMRL0	0000 0000
D8	SFR0	PWM1IF	PWM1 中断标志寄存器	PWMPIF	PWMZIF	PWMDDCIF	PWMUDCIF	PWMDDbIF	PWMUDbIF	PWMDDAIF	PWMUDAIF	0000 0000
D7	SFR0	PWM1CDH	PWM1C 占空比寄存器高位	-	-	-	-	PWMCD11	PWMCD10	PWMCD9	PWMCD8	---- 0000
D6	SFR0	PWM1CDL	PWM1C 占空比寄存器低位	PWMCD7	PWMCD6	PWMCD5	PWMCD4	PWMCD3	PWMCD2	PWMCD1	PWMCD0	0000 0000
D5	SFR0	PWM1BDH	PWM1B 占空比寄存器高位	-	-	-	-	PWMBD11	PWMBD10	PWMBD9	PWMBD8	---- 0000
D4	SFR0	PWM1BDL	PWM1B 占空比寄存器低位	PWMBD7	PWMBD6	PWMBD5	PWMBD4	PWMBD3	PWMBD2	PWMBD1	PWMBD0	0000 0000
D3	SFR0	PWM1ADH	PWM1A 占空比寄存器高位	-	-	-	-	PWMAD11	PWMAD10	PWMAD9	PWMAD8	---- 0000
D2	SFR0	PWM1ADL	PWM1A 占空比寄存器低位	PWMAD7	PWMAD6	PWMAD5	PWMAD4	PWMAD3	PWMAD2	PWMAD1	PWMAD0	0000 0000
D1	SFR0	PWM1OE	PWM1 输出使能寄存器	PWM1EN	-	PWMCNOE	PWMBNOE	PWMANOE	PWMCoe	PWMBOe	PWMAOe	0-00 0000
D0	SFR0	PSW	程序状态寄存器	CY	AC	F0	RS1	RS0	OV	F1	P	0000 0000



地址 (H)	SFR 号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
CF	SFR0	WDTCR	看门狗定时器控制寄存器	-	-	-	-	-	WDTOT2	WDTOT1	WDTOT0	----111
CE	SFR0	T2CON1	T2 控制寄存器 1	CM2ON	CM2	-	-	-	EX2SL	SIGTRG	T2EN	00--000
CD	SFR0	TH2	T2 计数高位寄存器	TH27	TH26	TH25	TH24	TH23	TH22	TH21	TH20	0000 0000
CC	SFR0	TL2	T2 计数低位寄存器	TL27	TL26	TL25	TL24	TL23	TL22	TL21	TL20	0000 0000
CB	SFR0	RCAPH2	T2 重载/捕获高位寄存器	RCAPH27	RCAPH26	RCAPH25	RCAPH24	RCAPH23	RCAPH22	RCAPH21	RCAPH20	0000 0000
CA	SFR0	RCAPL2	T2 重载/捕获低位寄存器	RCAPL27	RCAPL26	RCAPL25	RCAPL24	RCAPL23	RCAPL22	RCAPL21	RCAPL20	0000 0000
C9	SFR0	T2MOD	工作模式寄存器	LD2EN	T2CKS2	T2CKS1	T2CKS0	T2OE	T2SL1	T2SL0	-	0000 000-
C8	SFR0	T2CON	T2 控制寄存器	TF2	EXTF2	-	-	EX2EN	TR2	CT2	CPRL2	00--0000
C7	SFR0	-	-	-	-	-	-	-	-	-	-	-----
C6	SFR0	CLKCR1	时钟控制寄存器 1	-	CLKSW	FRTCSW	-	-	-	-	CLKSWS	-00- ---0
C5	SFR0	OSCSS	时钟安全寄存器	-	FHS2	FHS1	FHS0	-	XTSPD	-	-	-001 01--
C4	SFR0	P0PD	P0 端口下拉电阻控制寄存器	-	-	-	-	-	-	P01PD	P00PD	---- --00
C3	SFR0	P0PU	P0 端口上拉电阻控制寄存器	-	-	-	-	-	-	P01PU	P00PU	---- --00
C2	SFR0	-	-	-	-	-	-	-	-	-	-	-----
C1	SFR0	P0MOD0	P0 端口模式寄存器 0	-	-	-	-	P01M1	P01M0	P00M1	P00M0	---- 1010
C0	SFR0	CLKCR0	时钟控制寄存器 0	HIRCEN	-	OSCCEN	LIRCEN_LP	HIRC RDY	-	OSCCRDY	LIRC RDY	1-00 1-01
BF	SFR0	-	-	-	-	-	-	-	-	-	-	-----



地址 (H)	SFR 号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
BE	SFR0	-	-	-	-	-	-	-	-	-	-	----
BD	SFR0	IPL2	中断优先级控制 2 低位寄存器	-	-	-	-	-	-	LVDIPL	-	----0-
BC	SFR0	-	-	-	-	-	-	-	-	-	-	----
BB	SFR0	IPL1	中断优先级控制 1 低位寄存器	CRCIPL	IICIPL	PWM1FBIPL	PWM0IPL	-	PWM1IPL	T3IPL	T2IPL	0000-000
BA	SFR0	-	-	-	-	-	-	-	-	-	-	----
B9	SFR0	-	-	-	-	-	-	-	-	-	-	----
B8	SFR0	IPL0	中断优先级控制低位寄存器	-	ADCIPL	SPIIPL	UART0IPL	T1IPL	EXT1IPL	T0IPL	EXT0IPL	-000 0000
B7	SFR0	EXTIS	外部中断控制寄存器	-	-	-	EXT0IF4	EXT0IF3	EXT0IF2	EXT0IF1	EXT0IF0	---0 0000
B6	SFR0	EXTIS1	外部中断控制寄存器 1	-	-	-	EXT1IF4	EXT1IF3	EXT1IF2	EXT1IF1	EXT1IF0	---0 0000
B5	SFR0	EXTIS2	外部中断控制寄存器 2	EXT1A	EXT0A	-	-	EXT1IS1	EXT1IS0	EXT0IS1	EXT0IS0	00-- 0000
B4	SFR0	-	-	-	-	-	-	-	-	-	-	----
B3	SFR0	-	-	-	-	-	-	-	-	-	-	----
B2	SFR0	-	-	-	-	-	-	-	-	-	-	----
B1	SFR0	-	-	-	-	-	-	-	-	-	-	----
B0	SFR0	-	-	-	-	-	-	-	-	-	-	----
AF	SFR0	SPDAT	SPI 数据寄存器	SPDAT7	SPDAT6	SPDAT5	SPDAT4	SPDAT3	SPDAT2	SPDAT1	SPDAT0	xxxx xxxx
AE	SFR0	SPSTA	SPI 状态寄存器	SPEN	SPIF	MODF	WCOL	SPIOV	-	-	-	0000 0---



地址 (H)	SFR 号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
AD	SFR0	SPCON	SPI 控制寄存器	DIR	MSTR	CPHA	CPOL	SSDIS	SPR2	SPR1	SPR0	0000 0000
AC	SFR0	-	-	-	-	-	-	-	-	-	-	---- ----
AB	SFR0	IEN2	中断使能寄存器 2	-	-	-	-	-	-	LVDIE	-	---- --00
AA	SFR0	IEN1	中断使能寄存器 1	CRCIE	IICIE	PWM1FBIE	PWM0IE	-	PWM1IE	T3IE	T2IE	0000 -000
A9	SFR0	-	-	-	-	-	-	-	-	-	-	---- ----
A8	SFR0	IEN0	中断使能寄存器	EA	ADCIE	SPIIE	UART0IE	T1IE	EX1IE	T0IE	EX0IE	0000 0000
A7	SFR0	-	-	-	-	-	-	-	-	-	-	---- ----
A6	SFR0	EXTEN	外部中断使能寄存器	-	-	-	EXT0EN4	EXT0EN3	EXT0EN2	EXT0EN1	EXT0EN0	---0 0000
A5	SFR0	EXTEN1	外部中断使能寄存器 1	-	-	-	EXT1EN4	EXT1EN3	EXT1EN2	EXT1EN1	EXT1EN0	---0 0000
A4	SFR0	P2PD	P2 端口下拉电阻控制寄存器	P27PD	P26PD	P25PD	P24PD	P23PD	P22PD	P21PD	P20PD	0000 0000
A3	SFR0	P2PU	P2 端口上拉电阻控制寄存器	P27PU	P26PU	P25PU	P24PU	P23PU	P22PU	P21PU	P20PU	0000 0000
A2	SFR0	P2MOD1	P2 端口模式寄存器 1	P27M1	P27M0	P26M1	P26M0	P25M1	P25M0	P24M1	P24M0	1010 1010
A1	SFR0	P2MOD0	P2 端口模式寄存器 0	P23M1	P23M0	P22M1	P22M0	P21M1	P21M0	P20M1	P20M0	1010 1010
A0	SFR0	P2	P2 端口数据寄存器	P27D	P26D	P25D	P24D	P23D	P22D	P21D	P20D	xxxx xxxx
9F	SFR0	BRGH	串口 0 波特率计数器高位寄存器	BRG15	BRG14	BRG13	BRG12	BRG11	BRG10	BRG9	BRG8	00000000
9E	SFR0	BRGL	串口 0 波特率计数器低位寄存器	BRG7	BRG6	BRG5	BRG4	BRG3	BRG2	BRG1	BRG0	00000000
9D	SFR0	BRGCR	串口 0 波特率控制寄存器	BRGEN				BRTRIM3	BRTRIM2	BRTRIM1	BRTRIM0	0--- 0000



地址 (H)	SFR 号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
9C	SFR0	SCON2	串口 0 控制寄存器 2	-	-	-	-	TEN	PINSEL2	PINSEL1	PINSEL0	---- 0000
9B	SFR0	SADEN	串口 0 地址掩码寄存器	SADEN7	SADEN6	SADEN5	SADEN4	SADEN3	SADEN2	SADEN1	SADEN0	0000 0000
9A	SFR0	SADDR	串口 0 从机地址寄存器	SADDR7	SADDR6	SADDR5	SADDR4	SADDR3	SADDR2	SADDR1	SADDR0	0000 0000
99	SFR0	SBUF0	串口 0 缓存寄存器	SBUF7	SBUF6	SBUF5	SBUF4	SBUF3	SBUF2	SBUF1	SBUF0	xxxx xxxx
98	SFR0	SCON0	串口 0 控制寄存器	SM0/FE	SM1/RXOV	SM2/TXCOL	REN	TB8	RB8	TI	RI	0000 0000
97	SFR0	TH3	T3 重载高位寄存器	TH37	TH36	TH35	TH34	TH33	TH32	TH31	TH30	0000 0000
96	SFR0	TL3	T3 重载低位寄存器	TL37	TL36	TL35	TL34	TL33	TL32	TL31	TL30	0000 0000
95	SFR0	T3CR	T3 控制寄存器	TR3	CT3	T3F	RDCNT	T3EN	T3CKS2	T3CKS1	T3CKS0	0000 0000
94	SFR0	P1PD	P1 端口下拉电阻控制寄存器	P17PD	P16PD	P15PD	P14PD	P13PD	P12PD	P11PD	P10PD	0000 0000
93	SFR0	P1PU	P1 端口通用上拉电阻控制寄存器	P17PU	P16PU	P15PU	P14PU	P13PU	P12PU	P11PU	P10PU	0000 0000
92	SFR0	P1MOD1	P1 端口模式寄存器 1	P17M1	P17M0	P16M1	P16M0	P15M1	P15M0	P14M1	P14M0	1010 1010
91	SFR0	P1MOD0	P1 端口模式寄存器 0	P13M1	P13M0	P12M1	P12M0	P11M1	P11M0	P10M1	P10M0	1010 1010
90	SFR0	P1	P1 端口数据寄存器	P17D	P16D	P15D	P14D	P13D	P12D	P11D	P10D	xxxx xxxx
8F	SFR0	-	-	-	-	-	-	-	-	-	-	-----
8E	SFR0	TCON1	定时器控制寄存器 1	T1FS	T1OE	T1CKS1	T1CKS0	T0FS	T0OE	T0CKS1	T0CKS0	0000 0000
8D	SFR0	TH1	定时器计数高位寄存器 1	TH17	TH16	TH15	TH14	TH13	TH12	TH11	TH10	0000 0000
8C	SFR0	TH0	定时器计数高位寄存器 0	TH07	TH06	TH05	TH04	TH03	TH02	TH01	TH00	0000 0000



地址 (H)	SFR 号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
8B	SFR0	TL1	定时器计数低位寄存器 1	TL17	TL16	TL15	TL14	TL13	TL12	TL11	TL10	0000 0000
8A	SFR0	TL0	定时器计数低位寄存器 0	TL07	TL06	TL05	TL04	TL03	TL02	TL01	TL00	0000 0000
89	SFR0	TMOD	定时器模式寄存器	GATE1	CT1	T1M1	T1M0	GATE0	CT0	T0M1	T0M0	0000 0000
88	SFR0	TCON	定时器控制寄存器	TF1	TR1	TF0	TR0	-	-	ADCCMPIF	ADCIF	0000 --00
87	SFR0	PCON	电源控制寄存器	-	-	-	UART0EN	SMOD0	SSTAT0	SLEEP	STOP	---0 0000
86	SFR0	SLPCR	模式保护控制寄存器	SLPCR7	SLPCR6	SLPCR5	SLPCR4	SLPCR3	SLPCR2	SLPCR1	SLPCR0	0000 0000
85	SFR0	T3CNTH	T3 计数高位寄存器	CNT15	CNT14	CNT13	CNT12	CNT11	CNT10	CNT9	CNT8	0000 0000
84	SFR0	T3CNTL	T3 计数低位寄存器	CNT7	CNT6	CNT5	CNT4	CNT3	CNT2	CNT1	CNT0	0000 0000
83	SFR0	DPH	数据指针高字节	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	0000 0000
82	SFR0	DPL	数据指针低字节	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	0000 0000
81	SFR0	SP	堆栈指针	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	0000 0111
80	SFR0	P0	P0 端口数据寄存器	-	-	-	-	-	-	P01D	P00D	-----xx

### 20.3 特殊功能寄存器总览表 (XSFR)

地址 (H)	SFR 号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
FF6A	XSFR	PWM0AD	PWM0A 占空比设置寄存器	PWM0AD7	PWM0AD6	PWM0AD5	PWM0AD4	PWM0AD3	PWM0AD2	PWM0AD1	PWM0AD0	0000 0000
FF69	XSFR	PWM0P	PWM0 周期寄设置寄存器	PWM0P7	PWM0P6	PWM0P5	PWM0P4	PWM0P3	PWM0P2	PWM0P1	PWM0P0	0000 0000



地址 (H)	SFR 号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
FF68	XSFR	PWM0CR0	PWM0 控制寄存器 0	PWM0EN	PWM0PIE	PWM0IF	PWM0AOE	PWM0AS	PWM0C2	PWM0C1	PWM0C0	0000 0000
FF67	XSFR	CRCEDL	CRC 结束地址低字节寄存器	CRCEDL7	CRCEDL6	CRCEDL5	CRCEDL4	CRCEDL3	CRCEDL2	CRCEDL1	CRCEDL0	1111 1111
FF66	XSFR	CRCEDH	CRC 结束地址高字节寄存器	CRCEDH7	CRCEDH6	CRCEDH5	CRCEDH4	CRCEDH3	CRCEDH2	CRCEDH1	CRCEDH0	1111 1111
FF65	XSFR	CRCSTL	CRC 起始地址低字节寄存器	CRCSTL7	CRCSTL6	CRCSTL5	CRCSTL4	CRCSTL3	CRCSTL2	CRCSTL1	CRCSTL0	0000 0000
FF64	XSFR	CRCSTH	CRC 起始地址高字节寄存器	CRCSTH7	CRCSTH6	CRCSTH5	CRCSTH4	CRCSTH3	CRCSTH2	CRCSTH1	CRCSTH0	0000 0000
FF63	XSFR	CRCL	CRC 结果低字节寄存器	CRCL7	CRCL6	CRCL5	CRCL4	CRCL3	CRCL2	CRCL1	CRCL0	1111 1111
FF62	XSFR	CRCH	CRC 结果高字节寄存器	CRCH7	CRCH6	CRCH5	CRCH4	CRCH3	CRCH2	CRCH1	CRCH0	0000 0000
FF61	XSFR	-	-	-	-	-	-	-	-	-	-	-----
FF60	XSFR	CRCCR	CRC 控制寄存器	CRCCEN	-	CRCCIF	CRCCST	CRCCMD	CRCLCT	-	CRCCENC	0000 00-0
FF4D	XSFR	PWM1DT0	PWM1 死区控制寄存器 0	PWM1DT07	PWM1DT06	PWM1DT05	PWM1DT04	PWM1DT03	PWM1DT02	PWM1DT01	PWM1DT00	0000 0000
FF4C	XSFR	-	-	-	-	-	-	-	-	-	-	-----
FF4B	XSFR	PWM1PL	PWM1 周期寄存器低位	PWMP7	PWMP6	PWMP5	PWMP4	PWMP3	PWMP2	PWMP1	PWMP0	0000 0000
FF4A	XSFR	PWM1PH	PWM1 周期寄存器高位	PWMP15	PWMP14	PWMP13	PWMP12	PWMP11	PWMP10	PWMP9	PWMP8	0000 0000
FF49	XSFR	PWM1IE	PWM1 中断使能控制寄存器	PWMP1E	PWMPZIE	PWMPDCCIE	PWMPUDCCIE	PWMPDDBIE	PWMPUDBIE	PWMPDDAIE	PWMPUDAIE	0000 0000



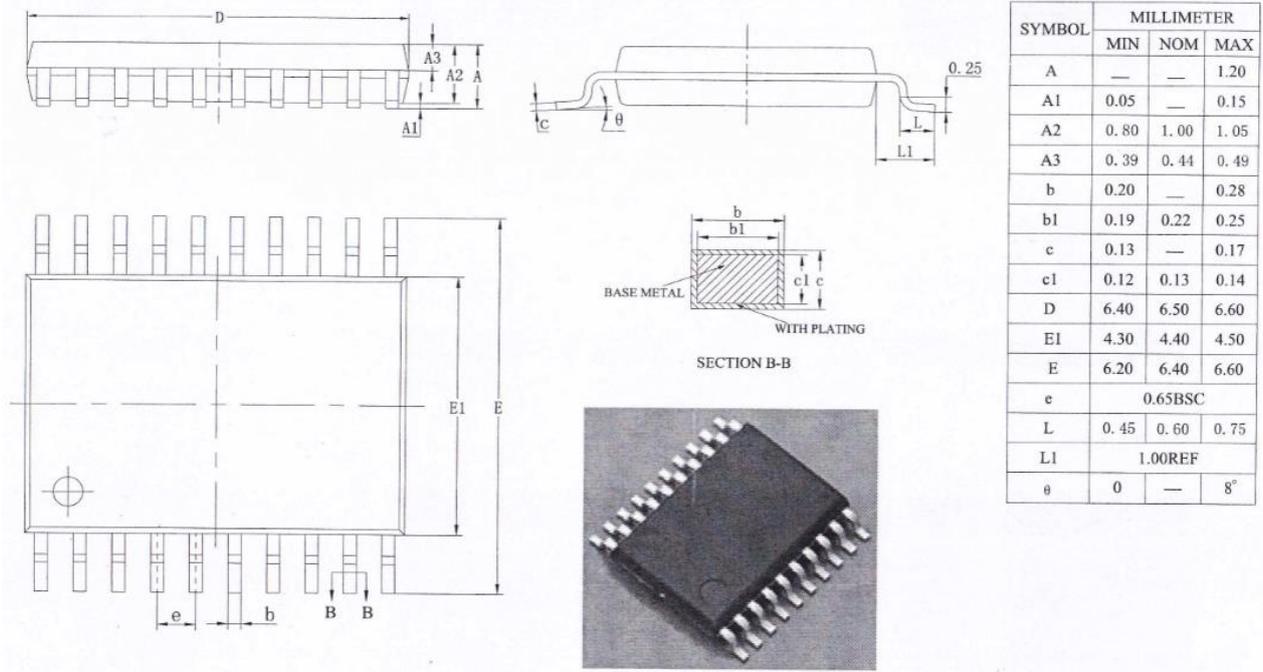
地址 (H)	SFR 号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
FF48	XSFR	PWM1CR0	PWM1 控制寄存器 0	CLKSEL	-	CTMOD1	CTMOD0	PREDIV1	PREDIV0	POSTDIV1	POSTDIV0	0-00 0000

Sinomcu.com

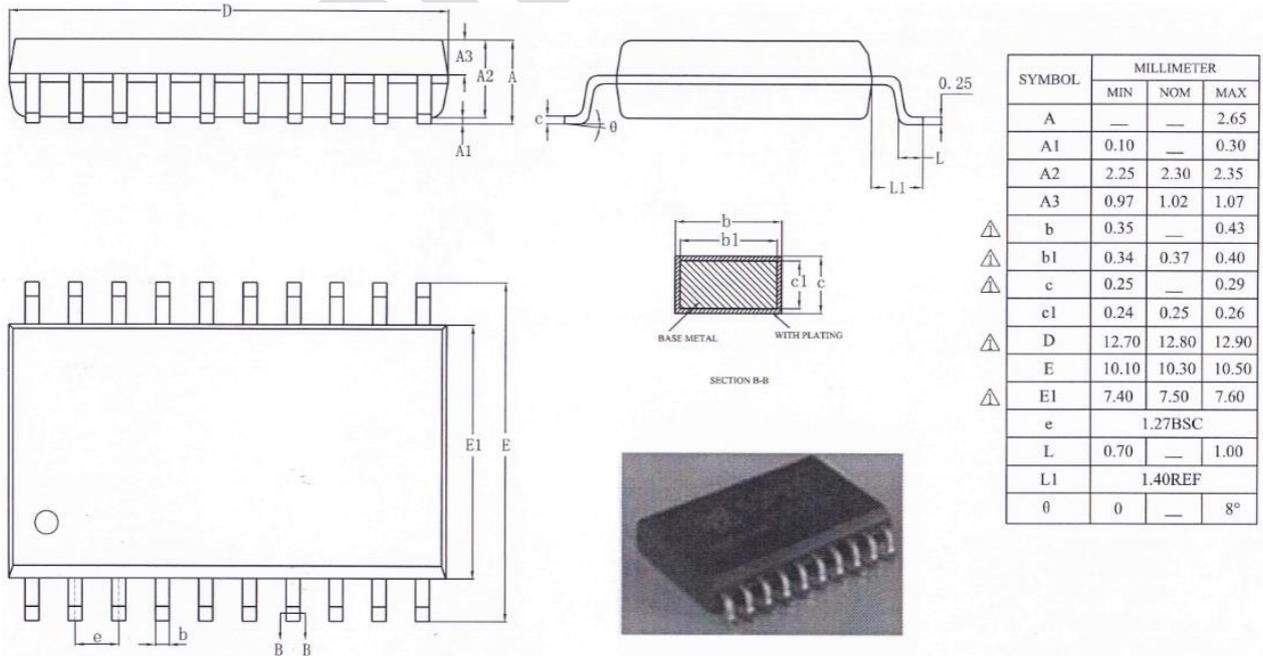


## 21 封装信息

### 21.1 TSSOP20 (0.65mm pitch)

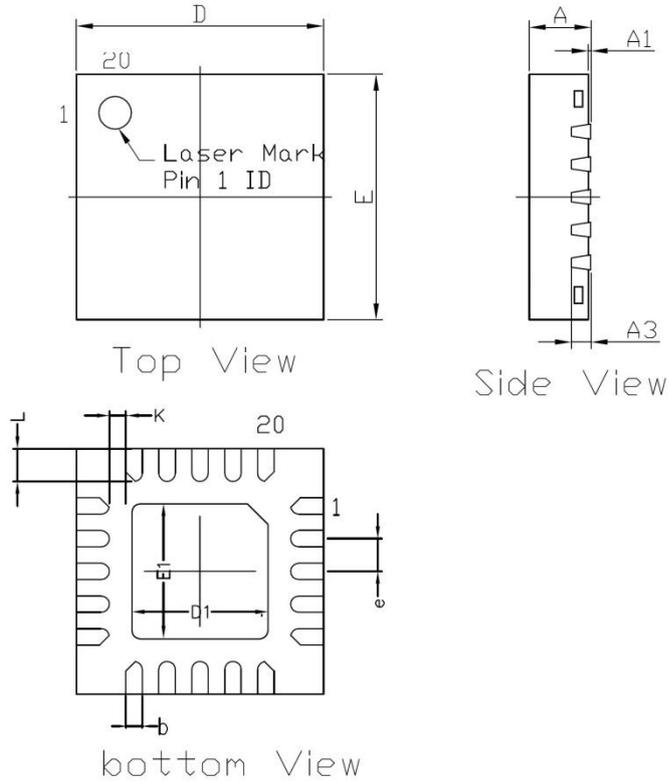


### 21.2 SOP20 (1.27mm pitch)



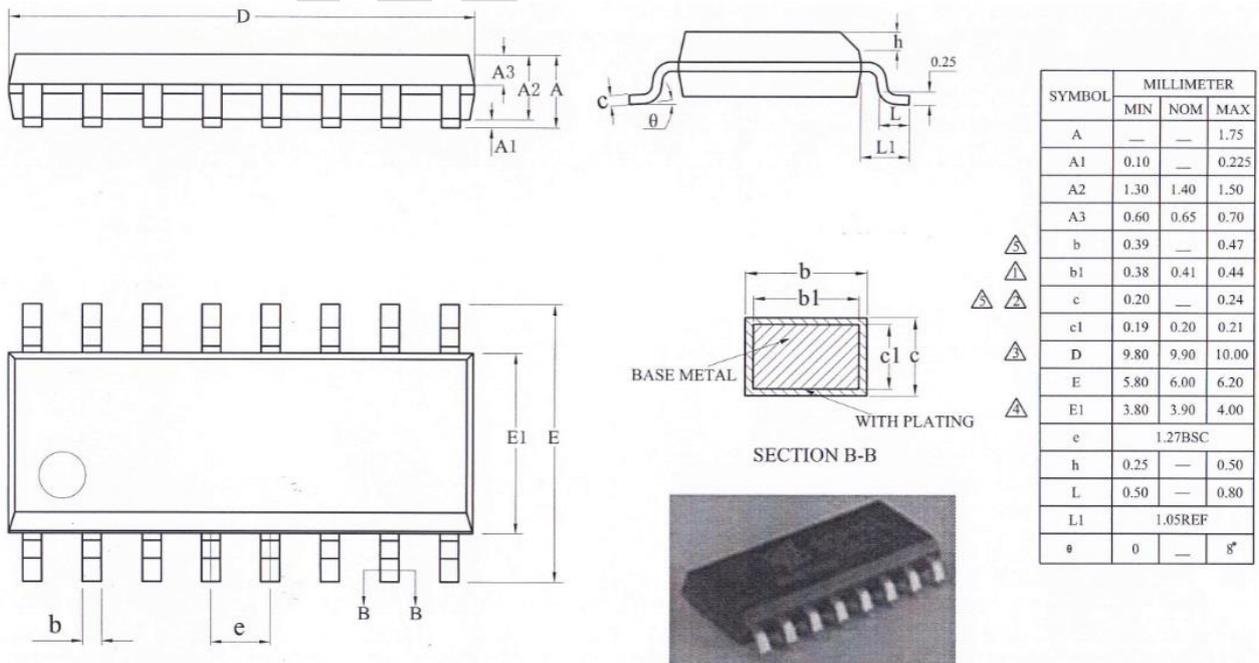


21.3 QFN20 (3 x 3 x 0.75)



标注	尺寸	最小	标准	最大
A		0.70	0.75	0.80
A1		0.00	—	0.05
A3		0.203REF		
b		0.15	—	0.25
D		2.90	3.00	3.10
E		2.90	3.00	3.10
D1		1.55	1.65	1.75
E1		1.55	1.65	1.75
e		0.40TYP		
K		0.20	—	—
L		0.30	0.40	0.50

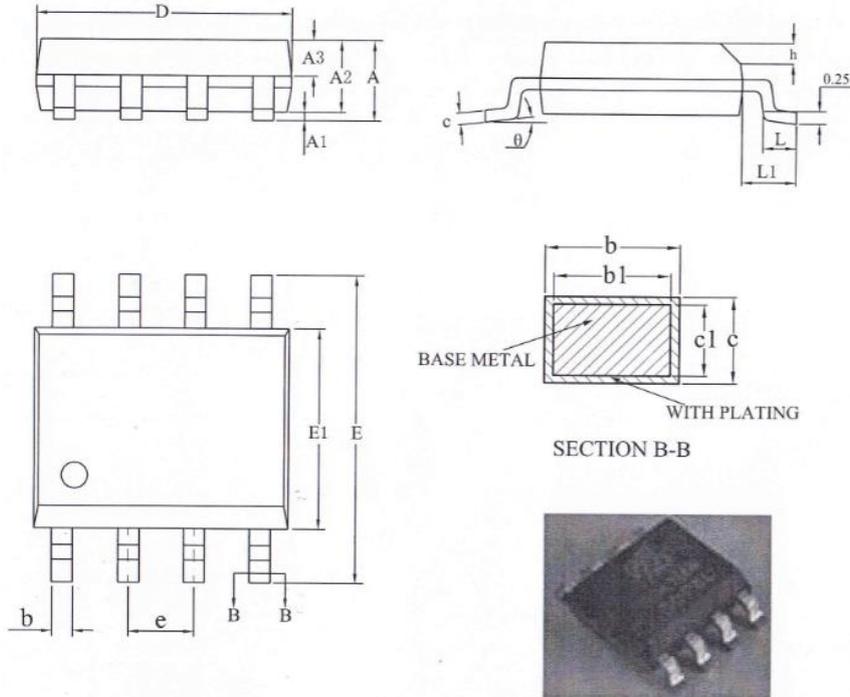
21.4 SOP16 (1.27mm pitch)



SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	1.75
A1	0.10	—	0.225
A2	1.30	1.40	1.50
A3	0.60	0.65	0.70
b	0.39	—	0.47
b1	0.38	0.41	0.44
c	0.20	—	0.24
c1	0.19	0.20	0.21
D	9.80	9.90	10.00
E	5.80	6.00	6.20
E1	3.80	3.90	4.00
e	1.27BSC		
h	0.25	—	0.50
L	0.50	—	0.80
L1	1.05REF		
θ	0	—	8°



21.5 SOP8 (1.27mm pitch)



- ⚠
- ⚠
- ⚠
- ⚠
- ⚠
- ⚠

SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	1.75
A1	0.10	—	0.225
A2	1.30	1.40	1.50
A3	0.60	0.65	0.70
b	0.39	—	0.47
b1	0.38	0.41	0.44
c	0.20	—	0.24
c1	0.19	0.20	0.21
D	4.80	4.90	5.00
E	5.80	6.00	6.20
E1	3.80	3.90	4.00
e	1.27BSC		
h	0.25	—	0.50
L	0.50	—	0.80
L1	1.05REF		
$\theta$	0	—	8°



## 22 修订记录

版本号	修订者	修订日期	修订内容
V1.0.0	Sinomcu	2023-03-01	1、初版
V1.1.0	Sinomcu	2023-09-21	1、勘误
V1.2.0	Sinomcu	2024-04-09	1、正式发布版

Sinomcu.com