RJM8L151S 系列

用户手册

文档版本 V1.1

武汉瑞纳捷半导体有限公司

Wuhan Runjet Semiconductor Co.,Ltd

武汉瑞纳捷半导体有限公司

重要声明

本文档必须以用户光盘的形式或者加密 E-MAIL 的形式发布给某个个人或组织，并且以其官方 E-MAIL 地址作为水印标记。禁止其他的发布形式。

本文档没有任何形式的担保、立场表达或其他暗示，若有任何因本文档或其中提及的产品所有资讯所引起的直接或间接损失，瑞纳捷公司及所属员工恕不为其担保任何责任。武汉瑞纳捷半导体有限公司保留不发出通知就改变其产品或产品说明书的权利，保留不更新本文档以反映这些改变的权利。

目录

1. 系统概述 11

1.1. 功能框图 11
1.2. 电源框图 11

2. 处理器 12

2.1. 概述 12
2.2. 主要特性 12
2.3. 内部数据存储器 12
2.3.1. 介绍 12
2.3.2. 特殊功能寄存器 12

3. FLASH 程序存储器和数据 14

3.1. 概述 14
3.2. 术语 14
3.3. FLASH 主要特性 14
3.4. 存储器组织结构 14
3.4.1. 数据区（XDATA） 15
3.4.2. 主程序区（CODE） 15
3.5. 存储器保护 15
3.6. FLASH 寄存器 16
3.6.1. 控制寄存器 FLASH_CTRL (偏移：21H) 16
3.6.2. DMA 源地址高字节寄存器 FLASH_DMA_CTRL0(偏移：29H) 17
3.6.3. DMA 源地址低字节寄存器 FLASH_DMA_CTRL1(偏移：2AH) 17
3.6.4. DMA 目的地址高字节寄存器 FLASH_DMA_CTRL2(偏移：2BH) 17
3.6.5. DMA 目的地址低字节寄存器 FLASH_DMA_CTRL3(偏移：2CH) 17
3.6.6. DMA 传输长度 FLASH_DMA_CTRL4(偏移：2EH) 17
3.6.7. FLASH 解锁控制寄存器 FLASH_LOCK_0（偏移：B8H） 17
3.6.8. FLASH 解锁控制寄存器 FLASH_LOCK_1（偏移：B9H） 17
3.6.9. FLASH 解锁控制寄存器 FLASH_LOCK_2(偏移：BAH) 18
3.6.10. FLASH 解锁控制寄存器 FLASH_LOCK_3(偏移：BBH) 18
3.6.11. SCREMAP 控制寄存器 SCRREMAP（偏移：12H） 18
3.7. FLASH 操作说明 18
3.7.1. 解锁操作 18
3.7.2. 擦除操作 19
3.7.3. 编程操作 19
3.8. MEMORY 保护配置 19
3.8.1. 保护操作 19

4. 电源检测控制 20

4.1. 电源 20
4.2. 上电复位（POR）/掉电复位(PDR) 20
4.3. 欠压复位（BOR） 20
4.4. 可编程电压检测（PVD） 20
4.5. 电压检测控制寄存器 20
4.5.1. 电压检测及复位使能寄存器 SCRSTEN(偏移：16H) 21
4.5.2. 电压检测档位选择寄存器 PVD_CFG_REG(偏移：F6H) 21
4.5.3. PVD 中断控制寄存器 PVD_INT_REG（偏移：CBH） 21
4.5.4. BOR 控制寄存器 BOR_CTRL（偏移：2FH） 22
4.6. 电压检测操作说明 22
4.6.1. PVD 初始化流程 22

Copyright © 2022 武汉瑞纳捷半导体有限公司

4.6.2. BOR 初始化流程 22
5. 低功耗模式 24
5.1. 概述 24
5.2. 低功耗特性 24
5.3. HALT 24
5.3.1. HALT 模式唤醒源 24
5.3.2. 进入 HALT 模式 25
5.3.3. HALT 模式唤醒 25
5.3.4. HALT 模式寄存器 25
5.4. ACTIVE_HALT 27
5.4.1. ACTIVE_HALT 模式唤醒源 27
5.4.2. 进入 ACTIVE_HALT 模式 27
5.4.3. ACTIVE_HALT 模式唤醒 27
5.5. WAIT 27
5.5.1. WAIT 模式唤醒源 27
5.5.2. 进入 WAIT 模式 28
5.5.3. WAIT 模式唤醒 28
5.6. LOW_POWER_WAIT 28
5.6.1. LOW_POWER_WAIT 模式唤醒源 28
5.6.2. 进入 LOW_POWER_WAIT 模式 29
5.6.3. LOW_POWER_WAIT 模式唤醒 29
5.7. LOW_POWER_RUN 29
5.7.1. 进入 LOW_POWER_RUN 模式 29
5.8. HALT/ACTIVE_HALT 操作说明 29
5.8.1. HALT/ACTIVE_HALT 休眠与唤醒配置 29
6. 复位（RST） 30
6.1. 复位电路 30
6.2. 内部复位源 30
6.2.1. 上/掉电复位（POR/PDR） 30
6.2.2. 欠压复位（BOR） 30
6.2.3. 看门狗复位 30
6.2.4. 模块软件复位 30
6.2.5. 系统软复位 30
6.3. 复位寄存器 30
6.3.1. 复位查询寄存器 SCU(偏移：11H) 31
6.3.2. 电压检测及复位使能寄存器 SCRSTEN(偏移：16H) 31
6.3.3. 复位控制寄存器 RSTCON0(偏移：18H) 31
7. 时钟控制 33
7.1. 主时钟源 33
7.1.1. HSE 34
7.1.2. HSO 34
7.1.3. HSI 34
7.1.4. LSE 34
7.1.5. LSI 34
7.2. 系统时钟源选择 34
7.2.1. 系统启动 34
7.2.2. 系统时钟切换 34
7.3. RTC 时钟选择 34
7.4. 外设时钟选择 34
7.5. 时钟控制寄存器 35
7.5.1. 系统时钟选择寄存器 SCSYS（偏移：15H） 35

Copyright © 2022 武汉瑞纳捷半导体有限公司

7.5.2. OSC 控制寄存器 LP_CTRL 1(偏移：10H) 35
7.5.3. 时钟控制寄存器 SCCM0（偏移：13H） 36
7.5.4. 时钟控制寄存器 SCCM1（偏移：14H） 37
7.5.5. 时钟控制寄存器 SCCM2（偏移：19H） 37
7.5.6. LSI 时钟控制寄存器 LSI_CTRL（偏移：C0H） 38

8. 通用输入输出口（GPIO）及端口复用 39

8.1. 概述 39
8.2. GPIO 主要特性 39
8.3. I/O 的配置和使用 39
8.3.1. 普通输入/输出 I/O 39
8.3.2. 作为外部中断源 39
8.4. I/O 管脚复用寄存器 39
8.4.1. GPIOx 管脚上拉寄存器 Px_PU（偏移：70H、71H、72H、73H 和 74H） 40
8.4.2. GPIOx 管脚输入使能控制寄存器 Px_IEN（偏移：7FH、CDH、CEH、CFH 和 FFH） 40
8.4.3. 管脚复用模拟功能或数字功能选择寄存器 AD_SEL1（偏移：C8H） 40
8.4.4. 管脚复用模拟功能或数字功能选择寄存器 AD_SEL2（偏移：C9H） 41
8.4.5. 管脚复用控制寄存器 IOCFG_CTRL0（偏移：F0H） 42
8.4.6. 管脚复用控制寄存器 IOCFG_CTRL1（偏移：F7H） 42
8.4.7. 管脚复用控制寄存器 IOCFG_CTRL2（偏移：F8H） 43
8.4.8. 管脚复用控制寄存器 IOCFG_CTRL3（偏移：F9H） 43
8.4.9. 管脚复用控制寄存器 IOCFG_CTRL4（偏移：FAH） 43
8.4.10. 管脚复用控制寄存器 IOCFG_CTRL5（偏移：FBH） 44
8.4.11. 管脚复用控制寄存器 IOCFG_CTRL6（偏移：FCH） 44
8.4.12. 管脚复用控制寄存器 IOCFG_CTRL7（偏移：FDH） 45
8.4.13. 管脚复用控制寄存器 IOCFG_CTRL8（偏移：FEH） 45
8.4.14. GPIO 功能复用表 46
8.5. GPIO 寄存器 48
8.5.1. GPIOx 数据寄存器 Px（偏移：80H、90H、A0H、B0H、91H） 49
8.5.2. GPIOx 输出使能寄存器 Px_OEN（偏移：A4H、A5H、A6H、A7H、96H） 50
8.5.3. GPIOx 中断使能寄存器 Px_INT_EN (偏移：D0H、D5H、DAH、DFH、E4H) 50
8.5.4. GPIOx 中断触发方式选择寄存器 Px_INT_SELy (偏移：详见寄存器列表) 50
8.5.5. GPIOx 中断状态寄存器 Px_INT_REG(偏移：D4H、D9H、DEH、E3H、E8H) 51
8.6. GPIO 操作说明 51
8.6.1. 输入/输出 IO 配置 51
8.6.2. 中断触发模式配置 52
8.6.3. 清除中断标志 52

9. 中断服务程序单元（ISR） 53

9.1. 概述 53
9.2. 中断服务单元特性 53
9.3. 中断源 53
9.4. 中断优先级 54
9.5. 中断服务程序单元寄存器 55
9.5.1. 中断使能寄存器 IEN0(偏移：A8H) 55
9.5.2. 中断使能寄存器 IEN1(偏移：B8H) 56
9.5.3. 中断使能寄存器 IEN2(偏移：9AH) 56
9.5.4. 中断使能寄存器 IEN3(偏移：C9H) 57
9.5.5. 中断使能寄存器 IEN4(偏移：D1H) 57
9.5.6. 中断优先级寄存器 IP0(偏移：A9H) 58
9.5.7. 中断优先级寄存器 IP1(偏移：B9H) 58

10. 模拟/数字转换器（ADC） 59

10.1. 概述 59

Copyright © 2022 武汉瑞纳捷半导体有限公司

10.2. ADC 主要特性 59
10.3. ADC 引脚描述 60
10.4. ADC 功能描述 60
10.4.1. ADC 开/关控制 60
10.4.2. 通道选择 60
10.4.3. 转换模式 60
10.4.4. ADC 校准及异常处理 61
10.5. ADC 寄存器 61
10.5.1. 配置寄存器 ADC_CSR(偏移：03H) 61
10.5.2. 配置寄存器 ADC_CR1(偏移：04H) 62
10.5.3. 配置寄存器 ADC_CR2(偏移：05H) 63
10.5.4. 配置寄存器 ADC_CR3(偏移：06H) 63
10.5.5. 数据高位寄存器 ADC_DRH(偏移：07H) 64
10.5.6. 数据低位寄存器 ADC_DRL(偏移：08H) 64
10.5.7. 状态寄存器 ADC_STS(偏移：09H) 64
10.5.8. 中断屏蔽寄存器 ADC_MSK(偏移：0AH) 65
10.5.9. VREF 模块控制寄存器 VREF_CTRL(偏移：CAH) 65
10.6. ADC 操作说明 65
10.6.1. 单次模式配置 65
10.6.2. 连续模式配置 66
10.6.3. 采样触发配置 66

11. 比较器（COMP） 68

11.1. 概述 68
11.2. COMP 主要特性 68
11.3. COMP 引脚描述 68
11.4. 比较器寄存器 68
11.4.1. 控制寄存器 COMP_CTRL0(偏移：1EH) 68
11.4.2. 控制寄存器 COMP_CTRL1(偏移：1FH) 69
11.4.3. 状态/控制寄存器 IOCFG_CTRL6(偏移：FCH) 69
11.5. 比较器操作说明 70
11.5.1. COMP1 配置 70
11.5.2. COMP2 配置 70

12. 基本定时器（TIMER0/1） 71

12.1. 概述 71
12.2. 基本定时器主要特性 71
12.3. 基本定时器工作模式 71
12.4. 基本定时器寄存器 72
12.4.1. 定时器控制寄存器 TCON(偏移：88H) 72
12.4.2. 定时器模式寄存器 TMOD(偏移：89H) 72
12.4.3. TIMER0 定时寄存器低字节 TL0(偏移：8AH) 72
12.4.4. TIMER1 定时寄存器低字节 TL1(偏移：8BH) 73
12.4.5. TIMER0 定时寄存器高字节 TH0(偏移：8CH) 73
12.4.6. TIMER1 定时寄存器高字节 TH1(偏移：8DH) 73
12.5. 基本定时器操作说明 73
12.5.1. TIMER0 的操作 73
12.5.2. TIMER1 的操作 73

13. 通用定时器（TIMER2） 74

13.1. 概述 74
13.2. 通用定时器主要特性 74
13.3. 分频器 74
13.4. 定时器模式 74

Copyright © 2022 武汉瑞纳捷半导体有限公司

13.5. 门控定时模式 74
13.6. 定时器重载功能 74
13.7. 定时器比较输出功能 75
13.7.1. 比较模式 75
13.7.2. PWM 模式 75
13.8. 定时器 2 中断 76
13.9. 定时器 2 寄存器 76
13.9.1. TIMER2 控制寄存器 T2CON(偏移：C8H) 76
13.9.2. TIMER2 比较模式选择寄存器 CCEN(偏移：C1H) 77
13.9.3. TIMER2 比较输出极性选择寄存器 CCL3(偏移：C6H) 77
13.9.4. TIMER2 比较寄存器低字节 CCL1(偏移：C2H) 78
13.9.5. TIMER2 比较寄存器高字节 CCH1(偏移：C3H) 78
13.9.6. TIMER2 比较寄存器低字节 CCL2(偏移：C4H) 78
13.9.7. TIMER2 比较寄存器高字节 CCH2(偏移：C5H) 78
13.9.8. TIMER2 比较寄存器低字节 CRCL(偏移：CAH) 78
13.9.9. TIMER2 比较寄存器高字节 CRCH(偏移：CBH) 78
13.9.10. TIMER2 计数寄存器低字节 TL2(偏移：CCH) 78
13.9.11. TIMER2 计数寄存器高字节 TH2(偏移：0DH) 78
13.9.12. TIMER2 中断标志状态寄存器 IRCON(偏移：C0H) 78
13.10. 定时器 2 操作说明 79
13.10.1. 定时 79
13.10.2. 比较 79

14. 高级定时器（TIMER3/4） 81

14.1. 概述 81
14.2. 主要特性 81
14.3. 功能描述 81
14.3.1. 定时器模式 81
14.3.2. 门控定时模式 81
14.3.3. 重载模式 82
14.3.4. 比较模式 82
14.3.5. 捕获模式 85
14.3.6. 中央对齐与边沿模式 87
14.4. 寄存器描述 88
14.4.1. 状态控制寄存器 TIM_IRCON(偏移：00h) 89
14.4.2. 中断控制寄存器 TIM_IEN1 (偏移：01h) 89
14.4.3. 控制寄存器 TIM_T2CON(偏移：02h) 90
14.4.4. 模式使能寄存器 TIM_CCEN(偏移：03h) 92
14.4.5. 比较寄存器 TIM_CCL1 (偏移：04h) 93
14.4.6. 比较寄存器 TIM_CCH1 (偏移：05h) 93
14.4.7. 比较寄存器 TIM_CCL2(偏移：06h) 93
14.4.8. 比较寄存器 TIM_CCH2(偏移：07h) 93
14.4.9. 死区时间寄存器 TIM_CCL3 (偏移：08h) 93
14.4.10. 比较寄存器 TIM_CRCL (偏移：0Ah) 93
14.4.11. 比较寄存器 TIM_CRCH (偏移：0Bh) 93
14.4.12. 计数寄存器 TIM_TL2(偏移：0Ch) 93
14.4.13. 计数寄存器 TIM_TH2(偏移：0Dh) 93
14.4.14. 刹车控制寄存器 TIM_BOCEN(偏移：0Eh) 94
14.4.15. 比较模式 1 数据寄存器 TIM_ID(偏移：0Fh) 94
14.4.16. 刹车输出控制寄存器 TIM_BREAK_OIS(偏移：10h) 95
14.4.17. 比较输出极性选择寄存器 TIM_OUT_P(偏移：11h) 95
14.4.18. 比较输出通道使能寄存器 TIM_OUT_EN(偏移：12h) 96
14.4.19. CC2/CC1 输入滤波寄存器 TIM_SAMPCC12(偏移：13h) 97

Copyright © 2022 武汉瑞纳捷半导体有限公司

14.4.20. CRC 输入滤波寄存器 TIM_SAMPCRC(偏移：14h) 98
14.4.21. 比较/捕获中断控制寄存器 TIM_IESCON(偏移：15h) 98
14.4.22. 捕获 0 状态寄存器 TIM_EDGES(偏移：16h) 99
14.4.23. 捕获 0 沿事件选择寄存器 TIM_EDGESEL(偏移：17h) 99
14.4.24. 预分频寄存器低 8 位 TIM_PSCL(偏移：18h) 100
14.4.25. 预分频寄存器高 8 位 TIM_PSCH(偏移：19h) 100
14.4.26. ADC 触发寄存器低 8 位 TIM_TRGOL(偏移：1Ah) 100
14.4.27. ADC 触发寄存器高 8 位 TIM_TRGOH(偏移：1Bh) 100
14.4.28. 控制寄存器 0 TIM_CON0(偏移：1Ch) 100
14.4.29. 刹车源控制寄存器 TIM_BRKC(偏移：1Dh) 101
14.4.30. 事件发生寄存器 TIM_EGR(偏移：1Eh) 102
14.4.31. 捕获寄存器 TIM_CAP_CRCL (偏移：1Fh) 103
14.4.32. 捕获寄存器 TIM_CAP_CRCH (偏移：20h) 103
14.4.33. 捕获寄存器 TIM_CAP_CCL1 (偏移：21h) 103
14.4.34. 捕获寄存器 TIM_CAP_CCH1 (偏移：22h) 103
14.4.35. 捕获寄存器 TIM_CAP_CCL2(偏移：23h) 103
14.4.36. 捕获寄存器 TIM_CAP_CCH2(偏移：24h) 104
14.4.37. 比较模式 0 通道 0 沿事件选择寄存器 TIM_COM_INT（地址 0x20A9） 104
14.5. 使用流程 104
14.5.1. 定时 104
14.5.2. 比较 105
14.5.3. 捕获 106

15. 低功耗定时器（LPTIMER） 107

15.1. 概述 107
15.2. 主要特性 107
15.3. 寄存器描述 107
15.3.1. 自动重载寄存器低字节 LPTIM_TL0(偏移：00h) 107
15.3.2. 自动重载寄存器高字节 LPTIM_TH0(偏移：01h) 107
15.3.3. 控制/状态寄存器 LPTIM_T0CON(偏移：02h) 107
15.3.4. 模式选择寄存器 LPTIM_T0MOD(偏移：03h) 108
15.4. 使用说明 108
15.4.1. 使用流程 108

16. 实时时钟（RTC） 110

16.1. 概述 110
16.2. RTC 主要特性 110
16.3. RTC 功能描述 110
16.4. RTC 寄存器 110
16.4.1. 小时配置寄存器 RTC_HOUR (偏移：01H) 111
16.4.2. 分钟配置寄存器 RTC_MINUTE (偏移：02H) 111
16.4.3. 秒配置寄存器 RTC_SECOND (偏移：03H) 111
16.4.4. 年配置寄存器 RTC_YEAR (偏移：05H) 111
16.4.5. 月配置寄存器 RTC_MONTH(偏移：06H) 111
16.4.6. 日期配置寄存器 RTC_DATE (偏移：07H) 111
16.4.7. 世纪配置寄存器 RTC_CENTURY (偏移：08H) 112
16.4.8. 闹铃小时配置寄存器 RTC_ALARM_HOUR (偏移：09H) 112
16.4.9. 闹铃分钟配置寄存器 RTC_ ALARM_MINUTE (偏移：0AH) 112
16.4.10. 闹铃秒配置寄存器 RTC_ ALARM_ SECOND (偏移：0BH) 112
16.4.11. 闹铃年配置寄存器 RTC_ ALARM_ YEAR (偏移：0DH) 112
16.4.12. 闹铃月份配置寄存器 RTC_ ALARM_ MONTH (偏移：0EH) 112
16.4.13. 闹铃日期配置寄存器 RTC_ ALARM_ DATE (偏移：0FH) 112
16.4.14. 秒计数校准寄存器 RTC_SECOND_CNT_x(x=3~0)(偏移：10H~13H) 113

Copyright © 2022 武汉瑞纳捷半导体有限公司

16.4.15. TICK 中断控制寄存器 RTC_TICK_CTRL (偏移：14H) 113
16.4.16. ALARM 中断控制寄存器 RTC_ALARM_CTRL(偏移：15H) 113
16.4.17. 重载使能寄存器 RTC_RELOAD(偏移：16H) 114
16.4.18. 开始工作使能寄存器 RTC_START(偏移：17H) 114
16.4.19. 中断标志清除寄存器 RTC_CLR_INT (偏移：18H) 114
16.4.20. 计数使能寄存器 RTC_CNT_EN (偏移：1AH) 114
16.5. RTC 操作说明 115

17. 看门狗（WDT） 116

17.1. 概述 116
17.2. WDT 主要特性 116
17.3. 看门狗复位 116
17.4. 看门狗中断 116
17.5. 看门狗开始工作 116
17.6. 看门狗计数器 116
17.7. 看门狗寄存器 116
17.7.1. WDTREL 加载值寄存器(偏移：86H) 117
17.7.2. WDT_LOCK_0 加锁寄存器(偏移：AEH) 117
17.7.3. WDT_LOCK_1 加锁寄存器(偏移：AFH) 117
17.8. 看门狗操作说明 117

18. AES/SM4/DES 加解密 119

18.1. 概述 119
18.2. 主要特性 119
18.3. 加解密算法寄存器 119
18.3.1. SM4_DES_CON 控制寄存器(偏移：00H) 120
18.3.2. SM4_DES_STS 状态寄存器(偏移：01H) 120
18.3.3. SM4_DES_MSK 中断控制寄存器(偏移：02H) 121
18.3.4. SM4_DES_DAT0_X 数据寄存器 32 位(偏移：03H) 121
18.3.5. SM4_DES_DAT1_X 数据寄存器 32 位(偏移：07H) 121
18.3.6. SM4_DES_DAT2_X 数据寄存器 32 位(偏移：0BH) 121
18.3.7. SM4_DES_DAT3_X 数据寄存器 32 位(偏移：0FH) 121
18.3.8. SM4_DES_KEY0_X 密钥寄存器 32 位(偏移：13H) 122
18.3.9. SM4_DES_KEY1_X 密钥寄存器 32 位(偏移：17H) 122
18.3.10. SM4_DES_KEY2_X 密钥寄存器 32 位(偏移：1BH) 122
18.3.11. SM4_DES_KEY3_X 密钥寄存器 32 位(偏移：1FH) 122
18.3.12. SM4_DES_KEY4_X 密钥寄存器 32 位(偏移：23H) 123
18.3.13. SM4_DES_KEY5_X 密钥寄存器 32 位(偏移：27H) 123
18.4. 加解密算法操作说明 123
18.4.1. 初始化流程 123
18.4.2. 算法加解密流程 123

19. 乘除法器（MDU） 125

19.1. 概述 125
19.2. 主要特性 125
19.3. MDU 计算操作描述 125
19.3.1. 16bit*16bit 乘法运算 125
19.3.2. 16bit/16bit 除法运算 125
19.3.3. 32bit/16bit 乘法运算 125
19.3.4. 移位操作运算 125
19.4. MDU 寄存器 125
19.4.1. 运算控制寄存器 ARCON（SFR 地址：EFH） 126
19.5. MDU 计算操作说明 126
19.5.1. 16bit*16bit 乘法运算 126

Copyright © 2022 武汉瑞纳捷半导体有限公司

19.5.2. 16bit/16bit 除法运算 126
19.5.3. 32bit/16bit 除法运算 127
19.5.4. 移位运算 127

20. 随机数发生器（RNG） 128

20.1. 概述 128
20.2. 随机数寄存器 128
20.2.1. RNG 控制寄存器 DTRNGCON（偏移：00H） 128
20.2.2. RNG 数据寄存器 DTRNGDAT（偏移：01H） 128
20.2.3. RNG 状态寄存器 DTRNGSTS（偏移：02H） 128
20.3. RNG 操作流程 128

21. CRC8 计算单元 129

21.1. 概述 129
21.2. 主要特性 129
21.3. CRC8 功能描述 129
21.4. CRC8 计算单元寄存器 129
21.4.1. 数据寄存器 CRC8_DATA（偏移：00H） 130
21.4.2. 控制寄存器 CRC8_CTRL（偏移：01H） 130
21.4.3. 初始值配置寄存器 CRC8_INIT（偏移：02H） 130
21.5. CRC8 计算单元操作说明 130

22. I2C 接口 131

22.1. 概述 131
22.2. 主要特性 131
22.3. 寄存器描述 131
22.2.1. I2C 主机时钟分频寄存器 I2C_CLK_DIV（偏移：00H） 131
22.2.2. I2C 控制寄存器 I2C_CR（偏移：01H） 131
22.2.3. I2C 中断使能寄存器 I2C_INT_EN（偏移：02H） 132
22.2.4. I2C 状态寄存器器 I2C_SR1（偏移：03H） 133
22.2.5. I2C 状态寄存器 I2C_SR2（偏移：04H） 134
22.2.6. I2C 数据寄存器 I2C_DR（偏移：05H） 135
22.2.7. I2C 从设备地址寄存器 I2C_SLAVE_ADDR1（偏移：06H） 135
22.2.8. I2C 从设备地址寄存器 I2C_SLAVE_ADDR2（偏移：07H） 136
22.2.9. I2C TIME OUT 计数寄存器 I2C_TIME_OUT_CNT（偏移：1CH） 136
22.3. I2C 功能描述 136
22.3.1. 模式选择 136
22.3.2. I2C 从模式 136
22.3.3. I2C 主模式 137
22.4. 使用说明 138
22.4.1. 主机接收 138
22.4.2. 主机复合接收 139
22.4.3. 主机发送 140
22.4.4. 从机发送模式 140
22.4.5. 从机接收模式 141

23. UART 142

23.1. 概述 142
23.2. 主要特性 142
23.3. UART 功能描述 142
23.4. UART 寄存器 143
23.4.1. 控制状态寄存器 SxCON 144
23.4.2. 数据缓冲寄存器 SxBUF 144
23.4.3. 重装寄存器低字节 SxRELL 145

Copyright © 2022 武汉瑞纳捷半导体有限公司

23.4.4. 重装寄存器高字节 SxRELH 145
23.4.5. 特殊功能寄存器 PxCON 145
23.5. UART 操作说明 145
23.5.1. 串口的发送 145
23.5.2. 串口的接收 145
23.5.3. 多机通信 145

24. LPUART 147

24.1. 概述 147
24.2. 主要特性 147
24.3. 寄存器描述 147
24.3.1. 控制寄存器 LPUART_CON (偏移：00h) 147
24.3.2. 状态寄存器 LPUART_STS(偏移：01h) 148
24.3.3. 发送数据寄存器 LPUART_TXDAT (偏移：02h) 148
24.3.4. 接收数据寄存器 LPUART_RXDAT (偏移：03h) 148
24.3.5. 波特率配置寄存器低字节 LPUART_DIV70(偏移：04h) 148
24.3.6. 波特率配置寄存器次低字节 LPUART_DIV158(偏移：05h) 148
24.3.7. 波特率配置寄存器次高字节 LPUART_DIV2316(偏移：06h) 149
24.3.8. 波特率配置寄存器高字节 LPUART_DIV3124(偏移：07h) 149
24.4. 使用流程 149
24.4.1. 串口的发送 149
24.4.2. 串口的接收 149

25. 串行外设接口（SPI） 150

25.1. 概述 150
25.2. 主要特性 150
25.3. SPI 功能描述 150
25.3.1. SPI 的管脚描述 150
25.3.2. SPI 的时钟相位和极性 150
25.4. SPI 寄存器 151
25.4.1. 控制寄存器 SPIxCON(偏移：00H) 152
25.4.2. 数据寄存器 SPIxDAT(偏移：01H) 152
25.4.3. 状态寄存器 SPIxSTS(偏移：02H) 153
25.4.4. 中断屏蔽寄存器 SPIxMSK(偏移：03H) 153
25.4.5. 数据长度寄存器 SPIxLEN(偏移：04H) 154
25.4.6. 一帧数据里接收数据长度寄存器 SPIxRXLEN(偏移：05H) 154
25.4.7. 一帧数据里发送数据长度寄存器 SPIxTXLEN(偏移：06H) 154
25.4.8. 主机发送数据间隔时间低字节寄存器 SPIx_M_WAIT_L (偏移：08H) 154
25.4.9. 主机发送数据间隔时间高字节寄存器 SPIx_M_WAIT_H (偏移：09H) 154
25.4.10. FAST 接收数据写入 SRAM 首地址低 8 位寄存器 SPI_FAST_RX_ADDRL(地址：0x20C1) 154
25.4.11. FAST 接收数据写入 SRAM 首地址高 4 位寄存器 SPI_FAST_RX_ADDRH(地址：0x20C2) 155
25.4.12. FAST 发送数据读出 SRAM 首地址低 8 位寄存器 SPI_FAST_TX_ADDRL(地址：0x20C3) 155
25.4.13. FAST 发送数据读出 SRAM 首地址高 4 位寄存器 SPI_FAST_TX_ADDRH(地址：0x20C4) 155
25.4.14. FAST 模式 SPI 组选择寄存器 SPI_FAST_SEL(地址：0x20C5) 155
25.5. SPI 操作流程 155
25.5.1. SPI 主机配置 155
25.5.2. SPI FAST 主机发送配置 155
25.5.3. SPI FAST 主机接收配置 156
25.5.4. SPI 从机配置 156
25.5.5. SPI FAST 从机接收配置 156
25.5.6. SPI FAST 从机发送配置 157

26. 版本号 158

26.1. 版本号寄存器 158

Copyright © 2022 武汉瑞纳捷半导体有限公司

26.1.1. 版本寄存器 VERSION0（偏移：1AH） 158
26.1.2. 版本寄存器 VERSION1（偏移：1BH） 158
26.1.3. 版本寄存器 VERSION2（偏移：1CH） 158
26.1.4. 版本寄存器 VERSION3（偏移：1DH） 158

27. 编程接口 159

附录 1：寄存器比特位属性及说明 160

附录 2：专业术语缩写及说明 161

附录 3：版本修订 162

Copyright © 2022 武汉瑞纳捷半导体有限公司

1. 系统概述

RJM8L151S 系列是一款 8051 增强型单片机。芯片内置 256B 内部数据存储器，8KB 的 SRAM，64KB 的 FLASH，具备快速中断响应，高效低功耗设计，多种休眠模式。片内集成多种安全密码模块，包括 SM4 / DES / 3DES/AES 安全算法，支持真随机数发生器，集成 CRC8 校验模块。芯片提供了多种外围接口：SPI、I2C、UART、LPUART、GPIO。

1.1. 功能框图

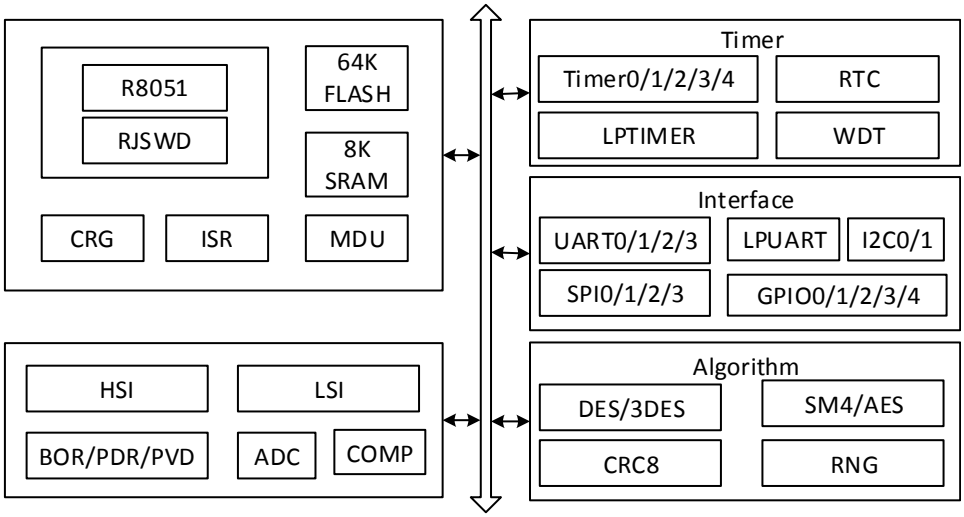


图 1-1 芯片功能框图

1.2. 电源框图

RJM8L151S 系列供电方案如下：外部电源通过 VDD 引脚供电，供电电压在 2.0V~5.0V 之间。

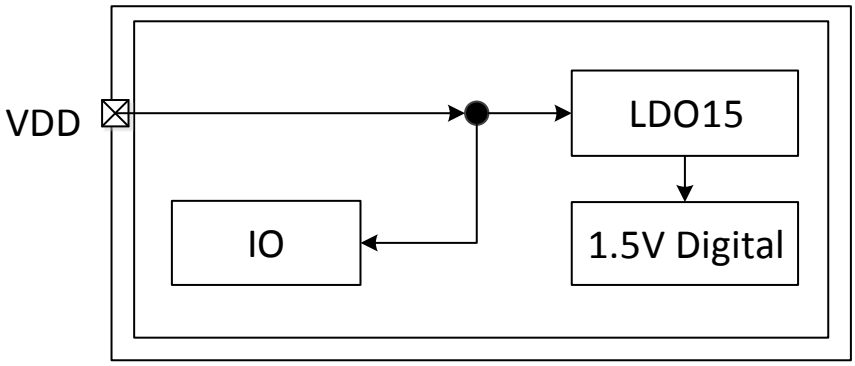


图 1-2 电源框图

Copyright © 2022 武汉瑞纳捷半导体有限公司

2. 处理器

2.1. 概述

CPU 是一颗具有超高性能 8 位架构微控制器。支持紧凑指令译码器和指令执行机，支持多种低功耗模式，适合超低功耗 MCU 应用的场景。

2.2. 主要特性

- 8 位 ALU（加法、减法、逻辑运算和位操作）
- 紧凑指令译码器和指令执行机
- 主频 16M
- 支持 2 个 DPTR

2.3. 内部数据存储器

2.3.1. 介绍

CPU 有 256 字节的内部数据存储器，包含 128 字节的特殊功能寄存器。

地址低于 80H 的 128 字节数据可以直接寻址，也可以间接寻址。间接寻址高于 7FH 的地址访问高 128 字节内部数据内存，而直接寻址高于 7FH 的地址访问 SFR 空间。

低 128 字节包括工作寄存器组（00H…1FH）位寻址空间（20H…2FH）。最低 32 字节共 4 个组，每个组由 8 个寄存器组成（R0-R7），通过 PSW 的两个位来选择使用哪一组寄存器。往上 16 字节为位寻址空间，可通过 00H-7FH 地址访问。

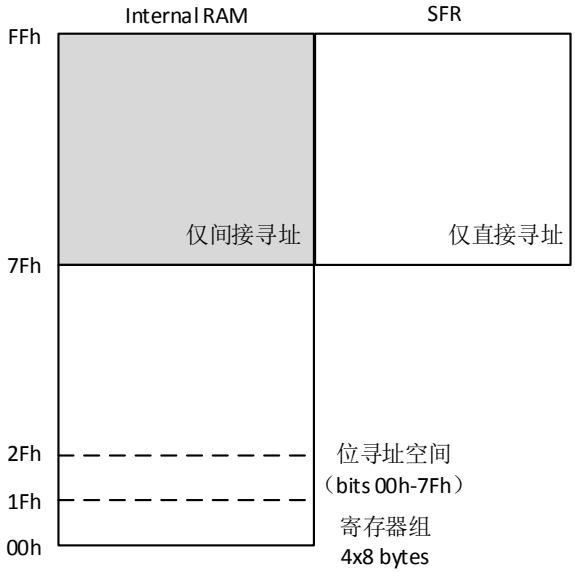


图 2-1 内存映射

2.3.2. 特殊功能寄存器

表 2-1 特殊功能寄存器位置

0xF8	-	-	-	-	-	-	-	-
0xF0	B	-	-	-	-	-	-	SRST
0xE8	-	MD0	MD1	MD2	MD3	MD4	MD5	ARCON
0xE0	ACC	-	-	-	-	-	-	-
0xD8	ADCO	-	-	-	-	-	-	-
0xD0	PSW	IEN4	-	-	-	-	-	-
0xC8	T2CON	IEN3	CRCL	CRCH	TL2	TH2	-	-
0xC0	IRCON	CCEN	CCL1	CCH1	CCL2	CCH2	CCL3	-
0xB8	IP/IEN	IP1	S0RELH	S1RELH	-	-	-	-
0xB0	P3	-	-	-	-	-	-	-
0xA8	IEN0	IP0	S0RELL	-	-	-	WDT_LOCK_0	WDT_LOCK_1
0xA0	P2	-	-	-	P0_OEN	P1_OEN	P2_OEN	P3_OEN
0x98	S0CON	S0BUF	IEN2	S1CON	S1BUF	S1RELL	-	-
0x90	P1	P4	DPS	DPC	PAGESEL	D_PAGESEL	P4_OEN	-
0x88	TCON	TMOD	TL0	TL1	TH0	TH1	CKCON	P1CON
0x80	P0	SP	DPL	DPH	DPL1	DPH1	WDTREL	PCON

Copyright © 2022 武汉瑞纳捷半导体有限公司

3. FLASH 程序存储器和数据

3.1. 概述

RJM8L151S 系列集成了 64KB 的 FLASH 存储器，用于保存代码和数据。由一组寄存器来控制，用户可以使用这些寄存器来编程或擦除存储器的内容、读保护或者配置特定的低功耗模式。

3.2. 术语

- 扇区（sector）

一个扇区指可由一个简单编程操作编程或擦除的一组字节，其操作是标准的编程和擦除操作。每个扇区大小为 128 字节。

3.3. FLASH 主要特性

- 多达 64KB 的 FLASH 程序存储器，不同的器件容量有所不同。
- 编程模式
  - DMA 编程
  - 在每次编程之前进行擦除操作
- 保护特性
- 使能存储器保护，芯片上电启动后进入保护模式，保护等级分为三级（高级别保护优先）。
- 在系统设置为低功耗等待（LOW_POWER_WAIT）和低功耗运行（LOW_POWER_RUN）模式下，存储器可配置为低功耗模式。

3.4. 存储器组织结构

FLASH 存储器组织结构映射

Copyright © 2022 武汉瑞纳捷半导体有限公司

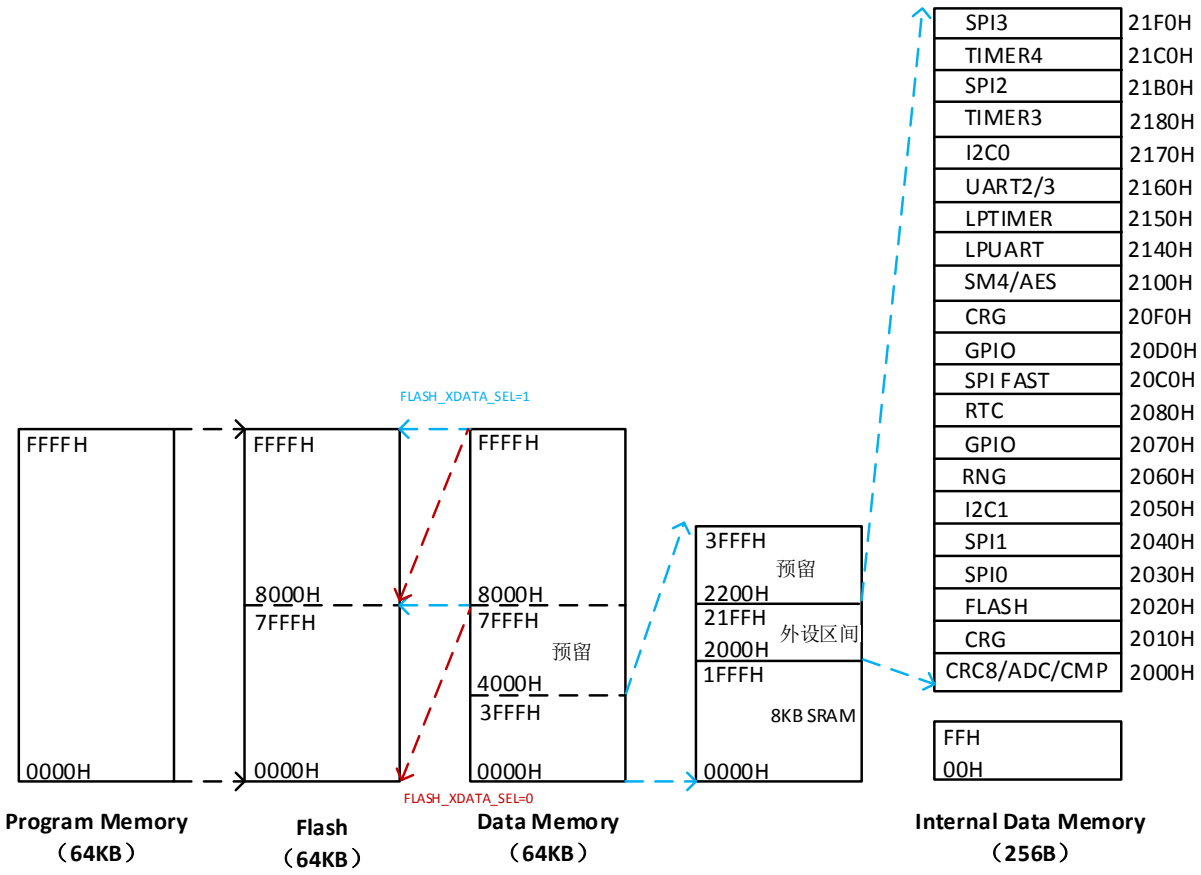


图 3-1 存储器地址映射

3.4.1. 数据区（XDATA）

数据（XDATA）区域可用于存储用户具体项目所需的数据。

3.4.2. 主程序区（CODE）

主程序区是指在 FLASH 程序存储器中用于存储应用代码的区域。

3.5. 存储器保护

使能存储器保护功能后（PROTECT_MOD），芯片上电启动后会进入保护模式，根据保护数据来使能保护级别，分三级（高级别保护优先）：

一级：正常运行模式，擦、写 FLASH 前需先解锁；

二级：可以连接仿真器和在 BOOT_RAM 模式运行，但是禁止仿真器和在 BOOT_RAM 模式读 FLASH；

三级：禁止仿真器连接和在 BOOT_RAM 模式运行；

Memory 保护机制如下：在 Memory 保护模式下，FLASH 数据不可读；若想通过改变保护权限来读取 FLASH 数据，则在改写 Memory 保护权限位时，硬件会擦除主 FLASH 区（main 区），再擦除 NVR 区（PROTECT_MOD 所在的 NVR 区），这样确保在 Memory 保护模式下，FLASH 数据不被读出。

存储器保护位 PROTECT_MOD 需要使用提供的写函数写入，在下次上电时生效。

Copyright © 2022 武汉瑞纳捷半导体有限公司

数据（XDATA）区域可用于存储用户具体项目所需的数据。

3.6. FLASH 寄存器

寄存器基地址：0x2000

偏移	名称	描述
0x21	FLASH_CTRL	FLASH 控制寄存器
0x29	FLASH_DMA_CTRL0	FLASH DMA 源地址高字节寄存器
0x2A	FLASH_DMA_CTRL1	FLASH DMA 源地址低字节寄存器
0x2B	FLASH_DMA_CTRL2	FLASH DMA 目的地址高字节寄存器
0x2C	FLASH_DMA_CTRL3	FLASH DMA 目的地址低字节寄存器
0x2E	FLASH_DMA_CTRL4	FLASH DMA 传输长度
0x12	SCREMAP	SRAM 重映射控制寄存器
0xBC	FLASH_LOCK_0	FLASH 解锁控制寄存器 0
0xBD	FLASH_LOCK_1	FLASH 解锁控制寄存器 1
0xBE	FLASH_LOCK_2	FLASH 解锁控制寄存器 2
0xBF	FLASH_LOCK_3	FLASH 解锁控制寄存器 3

3.6.1. 控制寄存器 FLASH_CTRL (偏移：21H)

比特	名称	属性	默认值	功能描述
7	RSV	-	-	保留
6	DMA_CODE_TO_DATA	RW	0x0	DMA 搬运 Flash 数据到 DATA 区 SRAM 使能位 0：禁止 1：使能 硬件清 0；
5	DMA_PRO_DATA	RW	0x0	DMA 编程 XDATA 区 Flash 使能位 0：禁止 1：使能 软件置 1，硬件清 0；
4	PROG_SECT_EN	RW	0x0	Sector 编程使能位 0：禁止 1：使能 软件置 1，硬件清 0；
3	DMA_AREA_SEL	RW	0x0	DMA 操作选择位 0：选择 XDATA 区 1：选择 CODE 区
2	DMA_PROG_EN	RW	0x0	DMA 编程使能位 0：禁止 1：使能

Copyright © 2022 武汉瑞纳捷半导体有限公司

				软件置 1，硬件清 0
1	DMA_ERASE_EN	RW	0x0	DMA 擦使能位 0：禁止 1：使能 软件置 1，硬件清 0
0	SECT_ERASE_EN	RW	0x0	sector 擦除使能位 0：禁止 1：使能 软件置 1，硬件清 0

3.6.2. DMA 源地址高字节寄存器 FLASH_DMA_CTRL0(偏移：29H)

比特	名称	属性	默认值	功能描述
7:0	FLASH_DMA_CTRL0	RW	0x00	源地址高 8 位

3.6.3. DMA 源地址低字节寄存器 FLASH_DMA_CTRL1(偏移：2AH)

比特	名称	属性	默认值	功能描述
7:0	FLASH_DMA_CTRL1	RW	0x00	源地址低 8 位

3.6.4. DMA 目的地址高字节寄存器 FLASH_DMA_CTRL2(偏移：2BH)

比特	名称	属性	默认值	功能描述
7:0	FLASH_DMA_CTRL2	RW	0x00	目的地址高 8 位

3.6.5. DMA 目的地址低字节寄存器 FLASH_DMA_CTRL3(偏移：2CH)

比特	名称	属性	默认值	功能描述
7:0	FLASH_DMA_CTRL3	RW	0x00	目的地址低 8 位

3.6.6. DMA 传输长度 FLASH_DMA_CTRL4(偏移：2EH)

比特	名称	属性	默认值	功能描述
7	RSV			
6:0	FLASH_DMA_CTRL4	RW	0x00	DMA 传输长度

3.6.7. FLASH 解锁控制寄存器 FLASH_LOCK_0（偏移：BCH）

比特	名称	属性	默认值	功能描述
7:0	LOCK0_EN	WO	0x00	FLASH 解锁寄存器 0x67：解锁 其他：不解锁 必须 LOCK0_EN、LOCK1_EN、LOCK2_EN 和 LOCK3_EN 同时有效才能解锁

3.6.8. FLASH 解锁控制寄存器 FLASH_LOCK_1（偏移：BDH）

比特	名称	属性	默认值	功能描述
7:0	LOCK1_EN	WO	0x00	FLASH 解锁寄存器

Copyright © 2022 武汉瑞纳捷半导体有限公司

				0x45：解锁 其他：不解锁 必须 LOCK0_EN、LOCK1_EN、LOCK2_EN 和 LOCK3_EN 同时有效才能解锁

3.6.9. FLASH 解锁控制寄存器 FLASH_LOCK_2(偏移：BEH)

比特	名称	属性	默认值	功能描述
7：0	LOCK2_EN	WO	0x00	FLASH 解锁寄存器 0xAB：解锁 其他：不解锁 必须 LOCK0_EN、LOCK1_EN、LOCK2_EN 和 LOCK3_EN 同时有效才能解锁

3.6.10. FLASH 解锁控制寄存器 FLASH_LOCK_3(偏移：BFH)

比特	名称	属性	默认值	功能描述
7：0	LOCK3_EN	WO	0x00	FLASH 解锁寄存器 0x89：解锁 其他：不解锁 必须 LOCK0_EN、LOCK1_EN、LOCK2_EN 和 LOCK3_EN 同时有效才能解锁

3.6.11. SCREMAP 控制寄存器 SCRREMAP（偏移：12H）

比特	名称	属性	复位值	描述
7:3	RSV	-	-	保留
2	SOFT_RST_EN	WO	0x1	系统软复位使能寄存器 写 1：复位所有模块； 硬件自动清成 1
1	FLASH_XDATA_SEL	RW	0x1	FLASH 映射到 XDATA 区选择： 1：把 FLASH 后 32KB（0x8000-0xFFFF）映射到 XDATA（0x8000-0xFFFF）空间； 0：把 FLASH 前 32KB（0x0000-0x7FFF）映射到 XDATA（0x8000-0xFFFF）空间
0	SCREMAP	RW	0x0	SRAM 映射位 写 1：将 8K SRAM 映射到 Code 区（0x0000~0x1FFF）；

3.7. FLASH 操作说明

FLASH 操作采用 DMA 方式编程操作，在编程之前需要先解锁，然后将要编程的区域擦除，最后进行编程操作。

3.7.1. 解锁操作

Copyright © 2022 武汉瑞纳捷半导体有限公司

1）配置 FLASH_LOCK_0 为 0x67；

2）配置 FLASH_LOCK_1 为 0x45；

3）配置 FLASH_LOCK_2 为 0xAB；

4）配置 FLASH_LOCK_3 为 0x89；

注：必须 FLASH_LOCK_0、FLASH_LOCK_1、FLASH_LOCK_2 和 FLASH_LOCK_3 同时有效时才能解锁。

3.7.2. 擦除操作

擦除操作会擦除目标地址所在的一个 Sector，共 128Byte。

1）配置 DMA 目的地址寄存器 FLASH_DMA_CTRL2 和 FLASH_DMA_CTRL3；

2）配置 FLASH_CTRL 寄存器为 **0x23**（擦除 XDATA 区）；

3）延时 10 个 NOP 后将 FLASH_CTRL 清零。

3.7.3. 编程操作

1）配置 DMA 目的地址寄存器 FLASH_DMA_CTRL2 和 FLASH_DMA_CTRL3。

2）配置 DMA 源地址寄存器 FLASH_DMA_CTRL0 和 FLASH_DMA_CTRL1。

3）配置 FLASH_DMA_CTRL4 寄存器，确定编程长度（最长 128Byte）。

4）配置 FLASH_CTRL 寄存器为 **0x34**（编程 XDATA 区）

5）延时 10 个 NOP 后将 FLASH_CTRL 清零。

3.8. Memory 保护配置

Memory 保护操作是基于 FLASH 操作流程，对相应地址设置相应值。配置对应保护位为固定值，则 Memory 保护生效；将对应保护位擦除，则 Memory 保护失效。

二级保护位对应地址为 **0xFF01H**，设置值位 **0x5A**；

三级保护位对应地址为 **0xFF02H**，设置值为 **0xA5**；

注：在使用保护功能时，不能重复在 NVR 区写入特征值。

3.8.1. 保护操作

1）配置 DMA 目的地址寄存器 FLASH_DMA_CTRL2 和 FLASH_DMA_CTRL3；

2）配置 DMA 源地址寄存器 FLASH_DMA_CTRL0 和 FLASH_DMA_CTRL1。

3）配置 FLASH_DMA_CTRL4，确定编程长度 1 字节。

4）使能 DMA 编程，配置 FLASH_CTRL 为 0xB4（编程 NVR 区）；

5）延时 10 个 NOP 后将 FLASH_CTRL 清零。

Copyright © 2022 武汉瑞纳捷半导体有限公司

4. 电源检测控制

本节适用所有型号的 RJM8L151S 设备。

4.1. 电源

芯片需要一个 2.0V 到 5.0V 电压供应（VDD）。

4.2. 上电复位（POR）/掉电复位(PDR)

芯片内置有 PDR 模块，兼具上电复位与掉电复位功能，操作允许在 1.25V。在上电过程中，当 VDD 低于规定的阈值 VPOR 时，设备保持复位模式，不需要外部复位。POR 的特性始终启用，POR 的阈值为 1.25V。在掉电过程中，当 VDD 低于规定的阈值 VPDR 时，设备保持复位模式，不需要外部复位。

4.3. 欠压复位（BOR）

芯片内置有 BOR 模块，当供电电压 VDD 低于 BOR 设置阈值时，使芯片处于复位。BOR 有 5 个不同阈值设置值可以选择，分为 1.7V，1.9V，2.3V，2.6V，2.8V。当芯片上电或复位后 BOR 自动开启，阈值为 BOR_CTRL 控制寄存器的阈值设置值 1.7V。同时若 NVR 区的阈值使能标记为 0x5A，则将 NVR 区的阈值设置值加载至 BOR_CTRL 控制寄存器并使能生效。

NVR 区的 BOR 阈值设置值和使能标记写入后在下次上电时生效，写入流程详见 FLASH 编程操作。

启用 BOR 时需确保芯片供电电压大于阈值，否则 MCU 无法运行。

4.4. 可编程电压检测（PVD）

该设备具有一个嵌入式可编程电压检测器（PVD），监控 VDD 电源，并将其与 Vpvd 阈值进行比较。

当 VDD 低于 Vpvd 阈值时或者当 VDD 高于 Vpvd 阈值时，会产生一个指示位。通过指示位来确定电压是否超过设定阈值（SCRSTEN 寄存器的 bit7）。软件通过 SCRSTEN 寄存器的 bit4 来设置使能启用电压检测 PVD。

PVD 在 1.8V 和 3.0V 之间有 7 个不同档位电平，通过 PVD_CFG_REG 寄存器进行选择。

4.5. 电压检测控制寄存器

寄存器基地址：0x2000

偏移	名称	描述
0x16	SCRSTEN	电压检测及复位使能寄存器

Copyright © 2022 武汉瑞纳捷半导体有限公司

0xF6	PVD_CFG_REG	电压检测档位选择寄存器
0xCB	PVD_INT_REG	电压检测中断控制寄存器
0x2F	BOR_CTRL	BOR 控制寄存器

4.5.1. 电压检测及复位使能寄存器 SCRSTEN(偏移：16H)

比特	名称	属性	复位值	描述
7	PVD_IN	RW	0x0	可编程电压监测指示位 0：VCC<Vth 1：VCC>Vth
6:5	RSV	-	-	保留
4	PVD_EN_REG	RW	0x0	可编程电压检测使能 0：禁止 1：使能
3	LPTIMER_CKEN	RW	0x0	LPTIMER 模块时钟使能 0：不使能； 1：使能；
2	LPUART_CKEN	RW	0x0	LPUART 模块时钟使能 0：不使能； 1：使能；
1	WDT_RST_EN	RW	0x0	WDT RST 是否复位整个芯片 0：禁止 1：复位
0	RSV	-	-	保留

4.5.2. 电压检测档位选择寄存器 PVD_CFG_REG(偏移：F6H)

比特	名称	属性	复位值	描述
7：3	RSV	-	-	保留
2:0	PVD_CFG	RW	0x0	PVD 比较阈值档位表 Vth： 000： 1.8V 001： 2.0V 010： 2.2V 011： 2.4V 100： 2.6V 101： 2.8V 11x： 3.0V 当 VCC>Vth，SCRSTEN 的 PVD_IN 位读出值 1

4.5.3. PVD 中断控制寄存器 PVD_INT_REG（偏移：CBH）

比特	名称	属性	复位值	描述
7：5	RSV	-	-	保留

Copyright © 2022 武汉瑞纳捷半导体有限公司

4	PVD_INT_FLAG	RW	0x0	PVD 中断标志位： 1：有 PVD 中断； 0：无 PVD 中断； 软件写 1 清 0；
3	PVD_INT_MASK	RW	0x1	PVD 中断屏蔽位： 1：屏蔽 PVD 中断； 0：打开 PVD 中断；
2	PVD_INT_POLAR	RW	0x0	PVD 中断极性选择： 1：当 VDD 大于 PVD 设置阈值 Vth 时产生中断； 0：当 VDD 小于 PVD 设置阈值 Vth 时产生中断；
1：0	RSV	-	-	保留

4.5.4. BOR 控制寄存器 BOR_CTRL（偏移：2FH）

比特	名称	属性	复位值	描述
7：4	RSV	-	-	保留
3:1	BOR_CFG	RW	0x0	BOR 阈值设置值： 0：1.7V； 1：1.9V； 2：2.3V； 3：2.6V； 其他：2.8V；
0	BOR_EN	RW	0x1	BOR 使能位: 1：使能； 0：不使能； ***注：进 HALT 之前需要先把 BOR 不使能***

注：BOR 配置分为 NVR2 区和寄存器：1.NVR2 区第三字节低三位配置阈值和 NVR2 的第四字节是否为 5A 控制该阈值的使能。2.寄存器（BOR_CTRL），芯片上电时，BOR 配置生效，用户可以通过配置 BOR_CTRL 寄存器调整 BOR 配置。

4.6. 电压检测操作说明

4.6.1. PVD 初始化流程

1) 配置 SCRSTEN 寄存器的 bit4 为 1，使能可编程电压检测功能。
2) 配置 PVD_CFG_REG 的 bit3-bit5，选择电压检测档位。
3) 配置 PVD_INT_REG 寄存器中的 bit2 和 bit3，选择 PVD 中断极性和开启 PVD 中断；
4) PVD 产生中断后，进去到对应的中断处理函数进行处理；

4.6.2. BOR 初始化流程

在写 MEMORY 保护功能位 0xFF01、0xFF02 时，进行 BOR 设置，0xFF03 的

Copyright © 2022 武汉瑞纳捷半导体有限公司

低三位为 BOR 阈值设置值，只有当 FF04 为 5A 时，BOR 的阈值设置值会变为 0x FF03 的低三位设置值；之后可通过配置 BOR_CTRL 寄存器写 BOR 使能和 BOR 阈值。

Copyright © 2022 武汉瑞纳捷半导体有限公司

5. 低功耗模式

5.1. 概述

芯片支持 5 种低功耗模式，以便用户在低功耗、唤醒时间以及唤醒源之间选择最优的解决方案。在系统复位撤销以后芯片默认处于正常工作模式，正常工作状态下系统时钟为内部 16MHz 二分频高速时钟。

- **HALT 模式：**系统所有时钟停止，系统不掉电，RAM 处于保持状态。HALT 模式下，可以配置多个外设作为唤醒源。唤醒过程中不涉及重新上下电，唤醒后 CPU 从上次进入 HALT 低功耗处接着运行。
- **ACTIVE_HALT 模式：**CPU 和外设时钟停止，但低速时钟 LSI 或 LSE 开启，除 HALT 模式下列出的唤醒源外，RTC、LPTIMER、LPUART 也可将系统从 ACTIVE_HALT 模式唤醒到正常工作模式。
- **WAIT 模式：**除 CPU 时钟停止外，其他模块均正常工作，系统工作在高速时钟 HSI 或 HSE 或 HSO 下。外设中断可将系统从 WAIT 模式唤醒到正常工作模式。与 HALT 模式相比，WAIT 模式唤醒时间非常短，唤醒源有效后，只需 3 个系统时钟周期可切换到正常工作模式。
- **LOW_POWER_WAIT 模式：**与 WAIT 模式相比，LOW_POWER_WAIT 模式下系统工作在低速时钟 LSI 或 LSE、或高速时钟 HSI 的 16 分频下。
- **LOW_POWER_RUN 模式：**使能并选择低频时钟 LSI 或 LSE 作为系统时钟，同时关闭高频时钟，系统进入该模式。

5.2. 低功耗特性

低功耗模式	唤醒时间	功耗
HALT	10us	0.6uA
ACTIVE_HALT	10us	0.85uA
LOW_POWER_WAIT	1.5ms	80uA
WAIT	3.5us	0.55mA
LOW_POWER_RUN	-	85uA

5.3. HALT

5.3.1. HALT 模式唤醒源

唤醒源	唤醒方式
PVD	PVD 输出，唤醒极性可选
I2C0/1	SDA 或 SCL 为低
SPI0/1/2/3	SPI 做从机时 CS 被选中
UART0/1/2/3	RX 为低

Copyright © 2022 武汉瑞纳捷半导体有限公司

GPIO0/1/2/3/4	GPIO 输入为高
COMP1/2	比较器 COMP1/COMP2，唤醒极性可选

在 HALT 模式下，若唤醒源为 GPIO，配置 CRG_PORT_SELx（x=0~4）选择唤醒源的端口（例：选择 GPIO10 为 HALT 唤醒源，则配置 CRG_PORT_SEL1 的 bit0 为 1）。其他唤醒源默认关闭，通过配置 CRG_WKUP_SELx（x=0~2）寄存器打开相应唤醒源。

5.3.2. 进入 HALT 模式

系统进入 HALT 模式，配置 SCSYS 寄存器的 bit2，则系统进入此模式，此时 CPU 停止运行。（详见“5.8 HALT/ACTIVE_HALT 操作说明”）

5.3.3. HALT 模式唤醒

在设置完唤醒源及系统进入 HALT 模式后，系统等待有效唤醒源信号，当系统等到有效唤醒信号后，系统退出 HALT 模式，此时 CPU 从上次停止运行处开始运行。

5.3.4. HALT 模式寄存器

低功耗模式基地址：0x2000

地址	名称	描述
0xF1	CRG_PORT_SEL0	HALT 模式 GPIO 唤醒源选择寄存器 0
0xF2	CRG_PORT_SEL1	HALT 模式 GPIO 唤醒源选择寄存器 1
0xF3	CRG_PORT_SEL2	HALT 模式 GPIO 唤醒源选择寄存器 2
0xF4	CRG_PORT_SEL3	HALT 模式 GPIO 唤醒源选择寄存器 3
0xF5	CRG_PORT_SEL4	HALT 模式 GPIO 唤醒源选择寄存器 4
0xC6	CRG_WKUP_SEL0	HALT 模式唤醒源选择寄存器 0
0xC7	CRG_WKUP_SEL1	HALT 模式唤醒源选择寄存器 1
0xCC	CRG_WKUP_SEL2	HALT 模式唤醒源选择寄存器 2

5.3.4.1. HALT 模式 GPIO 唤醒源选择寄存器 CRG_PORT_SEL0（偏移：F1H）

比特	名称	属性	复位值	描述
7:0	CRG_PORT_SEL0	RW	0x00	相应 bit 位为 1 即选择 GPIO0 中相对应 bit 作为 HALT 模式唤醒源，高电平唤醒；

5.3.4.2. HALT 模式 GPIO 唤醒源选择寄存器 CRG_PORT_SEL1（偏移：F2H）

比特	名称	属性	复位值	描述
7:0	CRG_PORT_SEL1	RW	0x00	相应 bit 位为 1 即选择 GPIO1 中相对应 bit 作为 HALT 模式唤醒源，高电平唤醒；

5.3.4.3. HALT 模式 GPIO 唤醒源选择寄存器 CRG_PORT_SEL2（偏移：F3H）

比特	名称	属性	复位值	描述
7:0	CRG_PORT_SEL2	RW	0x00	相应 bit 位为 1 即选择 GPIO2 中相对应 bit 作为 HALT 模式唤醒源，高电平唤醒；

5.3.4.4. HALT 模式 GPIO 唤醒源选择寄存器 CRG_PORT_SEL3（偏移：F4H）

比特	名称	属性	复位值	描述

Copyright © 2022 武汉瑞纳捷半导体有限公司

7:0	CRG_PORT_SEL3	RW	0x00	相应 bit 位为 1 即选择 GPIO3 中相对应 bit 作为 HALT 模式唤醒源，高电平唤醒；

5.3.4.5. HALT 模式 GPIO 唤醒源选择寄存器 CRG_PORT_SEL4（偏移：F5H）

比特	名称	属性	复位值	描述
7:0	CRG_PORT_SEL4	RW	0x00	相应 bit 位为 1 即选择 GPIO4 中相对应 bit 作为 HALT 模式唤醒源，高电平唤醒；

5.3.4.6. HALT 模式唤醒源选择寄存器 CRG_WKUP_SEL0（偏移：C6H）

比特	名称	属性	复位值	描述
7	UART2_WKUP_EN	RW	0x0	相应 bit 位为 1 即使能相应唤醒源；
6	UART1_WKUP_EN	RW	0x0	相应 bit 位为 1 即使能相应唤醒源；
5	UART0_WKUP_EN	RW	0x0	相应 bit 位为 1 即使能相应唤醒源；
4	SPI3_WKUP_EN	RW	0x0	相应 bit 位为 1 即使能相应唤醒源；
3	SPI2_WKUP_EN	RW	0x0	相应 bit 位为 1 即使能相应唤醒源；
2	SPI1_WKUP_EN	RW	0x0	相应 bit 位为 1 即使能相应唤醒源；
1	SPI0_WKUP_EN	RW	0x0	相应 bit 位为 1 即使能相应唤醒源；
0	RSV	-	-	保留

5.3.4.7. HALT 模式唤醒源选择寄存器 CRG_WKUP_SEL1（偏移：C7H）

比特	名称	属性	复位值	描述
7	PVD_WKUP_POLAR	RW	0x0	PVD 唤醒极性选择，1：高电平唤醒；0：低电平唤醒；
6	PVD_WKUP_EN	RW	0x0	相应 bit 位为 1 即使能相应唤醒源；
5:4	RSV	-	-	保留
3	RTC_WKUP_EN	RW	0x0	相应 bit 位为 1 即使能相应唤醒源；
2	I2C1_WKUP_EN	RW	0x0	相应 bit 位为 1 即使能相应唤醒源；
1	I2C0_WKUP_EN	RW	0x0	相应 bit 位为 1 即使能相应唤醒源；
0	UART3_WKUP_EN	RW	0x0	相应 bit 位为 1 即使能相应唤醒源；

5.3.4.8. HALT 模式唤醒源选择寄存器 CRG_WKUP_SEL2（偏移：CCH）

比特	名称	属性	复位值	描述
7	RSV	-	-	保留
6	BOR_WKUP_EN	RW	0x1	在系统从 HALT/ACTIVE_HALT 模式唤醒时，是否开启 BOR： 1：开启； 0：不开启；
5	LPTIMER_WKUP_EN	RW	0x0	LPTIMER 唤醒使能选择，相应 bit 位为 1 即使能相应唤醒源；
4	LPUART_WKUP_EN	RW	0x0	LPUART 唤醒使能选择，相应 bit 位为 1 即使能相应唤醒源；

Copyright © 2022 武汉瑞纳捷半导体有限公司

3	COMP1_WKUP_POLAR	RW	0x0	比较器 1 唤醒极性选择，1：高电平唤醒；0：低电平唤醒；
2	COMP1_WKUP_EN	RW	0x0	比较器 1 唤醒使能选择，相应 bit 位为 1 即使能相应唤醒源；
1	COMP2_WKUP_POLAR	RW	0x0	比较器 2 唤醒极性选择，1：高电平唤醒；0：低电平唤醒；
0	COMP2_WKUP_EN	RW	0x0	比较器 2 唤醒使能选择，相应 bit 位为 1 即使能相应唤醒源；

5.4. ACTIVE_HALT

5.4.1. ACTIVE_HALT 模式唤醒源

唤醒源	唤醒方式
PVD	PVD 输出，唤醒极性可选
I2C0/1	SDA 或 SCL 为低
SPI0/1/2/3	SPI 做从机时 CS 被选中
UART0/1/2/3	RX 为低
GPIO0/1/2/3/4	GPIO 输入为高
COMP1/2	比较器 COMP1/COMP2，唤醒极性可选
RTC	RTC 中断
LPTIMER	LPTIMER 计数到配置值，或者相应的外部事件计数到指定个数
LPUART	在 9600 波特率下接收到 1byte 数据

在 ACTIVE_HALT 模式下，若唤醒源为 GPIO，配置的流程与 HALT 模式下相同，不过 ACTIVE_HALT 模式下多了 3 种唤醒源。

5.4.2. 进入 ACTIVE_HALT 模式

系统进入 ACTIVE_HALT 模式，先打开低速时钟，配置 SCSYS 寄存器的 bit2，则系统进入此模式，此时 CPU 停止运行。（详见“5.8 HALT/ACTIVE_HALT 操作说明”）

5.4.3. ACTIVE_HALT 模式唤醒

在设置完唤醒源及系统进入 ACTIVE_HALT 模式后，系统等待有效唤醒源信号，当系统等到有效唤醒信号后，系统退出 ACTIVE_HALT 模式，此时 CPU 从上次停止运行处开始运行。

5.5. WAIT

5.5.1. WAIT 模式唤醒源

唤醒源	唤醒方式
PVD	PVD_IN
GPIO0/1/2/3/4	GPIO 模块中断

Copyright © 2022 武汉瑞纳捷半导体有限公司

TIMER0/1/2/3/4	TIMER 模块中断
UART0/1/2/3/	UART 模块中断
SPI0/1/2/3	SPI 模块中断
I2C0/1	I2C 模块中断
COMP1/2	比较器 1/2 中断
WDT	WDT 模块中断
ADC	ADC 模块中断
RTC	RTC 模块中断
LPUART	LPUART 模块中断
LPTIMER	LPTIMER 模块中断

在 WAIT 模式下，唤醒源为外设模块的中断信号，则在配置唤醒源时注意使能该模块中断。

5.5.2. 进入 WAIT 模式

系统进入 WAIT 模式之前，选择唤醒源并正常配置作为唤醒源的功能模块及其中断。配置寄存器 LP_CTRL1 的 bit6，关闭 CPU 时钟，则系统进入此模式，此时 CPU 停止运行，外设正常工作。

5.5.3. WAIT 模式唤醒

在设置完唤醒源及系统进入 WAIT 模式后，系统等待外设中断信号，当系统等到外设中断后，系统退出 WAIT 模式，此时 CPU 从上次停止运行处开始运行。

5.6. LOW_POWER_WAIT

5.6.1. LOW_POWER_WAIT 模式唤醒源

唤醒源	唤醒方式
PVD	PVD_IN
GPIO0/1/2/3/4	GPIO 模块中断
TIMER0/1/2/3/4	TIMER 模块中断
UART0/1/2/3/	UART 模块中断
SPI0/1/2/3	SPI 模块中断
I2C0/1	I2C 模块中断
COMP1/2	比较器 1/2 中断
WDT	WDT 模块中断
ADC	ADC 模块中断
RTC	RTC 模块中断
LPUART	LPUART 模块中断
LPTIMER	LPTIMER 模块中断

在 LOW_POWER_WAIT 模式下，唤醒源为外设模块的中断信号，则在配置唤醒源时注意使能该模块中断。

Copyright © 2022 武汉瑞纳捷半导体有限公司

5.6.2. 进入 LOW_POWER_WAIT 模式

系统进入 LOW_POWER_WAIT 模式之前，先选择唤醒源并正常配置作为唤醒源的功能模块及其中断；使能并选择低频时钟作为系统时钟，同时关闭高频时钟；配置寄存器 LP_CTRL1 的 bit6，关闭 CPU 时钟，则系统进入此模式，此时 CPU 停止运行，外设正常工作。

5.6.3. LOW_POWER_WAIT 模式唤醒

在设置完唤醒源及系统进入 LOW_POWER_WAIT 模式后，系统等待外设中断信号，当系统等到外设中断后，系统退出 LOW_POWER_WAIT 模式，此时 CPU 从上次停止运行处开始运行。

5.7. LOW_POWER_RUN

5.7.1. 进入 LOW_POWER_RUN 模式

使能并选择低频时钟 LSI 或 LSE 作为系统时钟，同时关闭高频时钟，系统进入该模式。

5.8. HALT/ACTIVE_HALT 操作说明

5.8.1. HALT/ACTIVE_HALT 休眠与唤醒配置

1) 关总中断和模块中断；
2) 将系统时钟切换为内部高速时钟；
3) ACTIVE_HALT 下需打开低速时钟，HALT 下需关闭低速时钟；
4) 配置 CRG_WKUP_SEL2 的 BOR_WKUP_EN 位，禁止系统 HALT/ACTIVE_HALT 模式唤醒时开启 BOR；
5) 配置 SCRSTEN 的 PVD_EN_REG 位，关闭 PVD；
6) 配置 BOR_CTRL 的 BOR_EN 位，关闭 BOR；
7) 配置 SCSYS 的 HALT 位，进入低功耗；
8) 等待唤醒后，先开 PVD，而后开 BOR；
9) 系统时钟回切；
10) 开模块中断与总中断；

Copyright © 2022 武汉瑞纳捷半导体有限公司

6. 复位（RST）

RJM8L151S 系统复位将所有寄存器至它们的复位状态，共有 5 个复位源：

- NRST 引脚产生的外部复位
- 上/掉电复位
- 欠压复位
- 看门狗复位
- 模块软件复位
- 系统软复位

6.1. 复位电路

复位引脚 NRST 上给最小为 500ns 的低电平脉冲即可产生一个外部复位。

6.2. 内部复位源

每个内部复位源在复位控制器中都有与之对应的控制位，这些控制位通过软件写 1，来使能该内部复位源的复位功能。

6.2.1. 上/掉电复位（POR/PDR）

内置的 PDR 模块，兼具上电复位与掉电复位功能，操作允许在 1.25V。在上电或掉电过程中，当 VDD 低于 VPOR 或 VPDR 时，设备保持复位模式，不需要外部复位。

6.2.2. 欠压复位（BOR）

当供电电压小于 BOR 设定的阈值电压时，系统会产生复位。

6.2.3. 看门狗复位

详见看门狗章节。

6.2.4. 模块软件复位

芯片外设模块可以通过配置复位寄存器（RSTCON0）开启相关外设的软复位功能。

6.2.5. 系统软复位

通过配置 SCREMAP 控制寄存器的 SOFT_RST_EN 开启系统软复位功能。

6.3. 复位寄存器

系统寄存器基地址：0x2000

地址	名称	描述
0x11	SCU	复位查询寄存器
0x16	SCRSTEN	电压检测及复位使能寄存器

Copyright © 2022 武汉瑞纳捷半导体有限公司

0x18	RSTCON0	复位控制寄存器

6.3.1. 复位查询寄存器 SCU(偏移：11H)

比特	名称	属性	复位值	描述
7:6	RSV	-	-	保留
5	BOR_RST_FLAG	RW	0x0	BOR 复位标志： 1：系统被 BOR 复位； 软件写 1 清 0；
4	SOFT_RST_FLAG	RW	0x0	软复位复位标志： 1：系统被软复位复位； 软件写 1 清 0；
3	RSV	-	-	保留
2	WDT_RST_FLAG	RW	0x0	WDT 复位标志： 1：系统被 WDT 复位； 软件写 1 清 0；
1:0	RSV	-	-	保留

6.3.2. 电压检测及复位使能寄存器 SCRSTEN(偏移：16H)

比特	名称	属性	复位值	描述
7	PVD_IN	RW	0x0	可编程电压监测指示位 0：VCC<Vth 1：VCC>Vth
6:5	RSV	-	-	保留
4	PVD_EN_REG	RW	0x0	可编程电压检测使能 0：禁止 1：使能
3	LPTIMER_CKEN	RW	0x0	LPTIMER 模块时钟使能 0：不使能； 1：使能；
2	LPUART_CKEN	RW	0x0	LPUART 模块时钟使能 0：不使能； 1：使能；
1	WDT_RST_EN	RW	0x0	WDT RST 是否复位整个芯片 0：禁止 1：复位
0	RSV	-	-	保留

6.3.3. 复位控制寄存器 RSTCON0(偏移：18H)

比特	名称	属性	复位值	描述
7	SPI3_SRST_REQ	RW	0x1	SPI3 模块软复位使能位 0：禁止

Copyright © 2022 武汉瑞纳捷半导体有限公司

				1：使能
6	SPI2_SRST_REQ	RW	0x1	SPI2 模块软复位使能位 0：禁止 1：使能
5	SPI1_SRST_REQ	RW	0x1	SPI1 模块软复位使能位 0：禁止 1：使能
4	SPI0_SRST_REQ	RW	0x1	SPI0 模块软复位使能位 0：禁止 1：使能
3	RSV	-	-	保留
2	DES_SRST_REQ	RW	0x1	DES/SM4 模块软复位使能位 0：禁止 1：使能
1	ADC_SRST_REQ	RW	0x1	ADC 模块软复位使能位 0：禁止 1：使能
0	RSV	-	-	保留

Copyright © 2022 武汉瑞纳捷半导体有限公司

7. 时钟控制

时钟控制器功能强大，目的在于使用户在获取最好性能的同时，能保证消耗功率最低。用户可以独立管理各个时钟源，并将它们分配到 CPU 或各个外设。

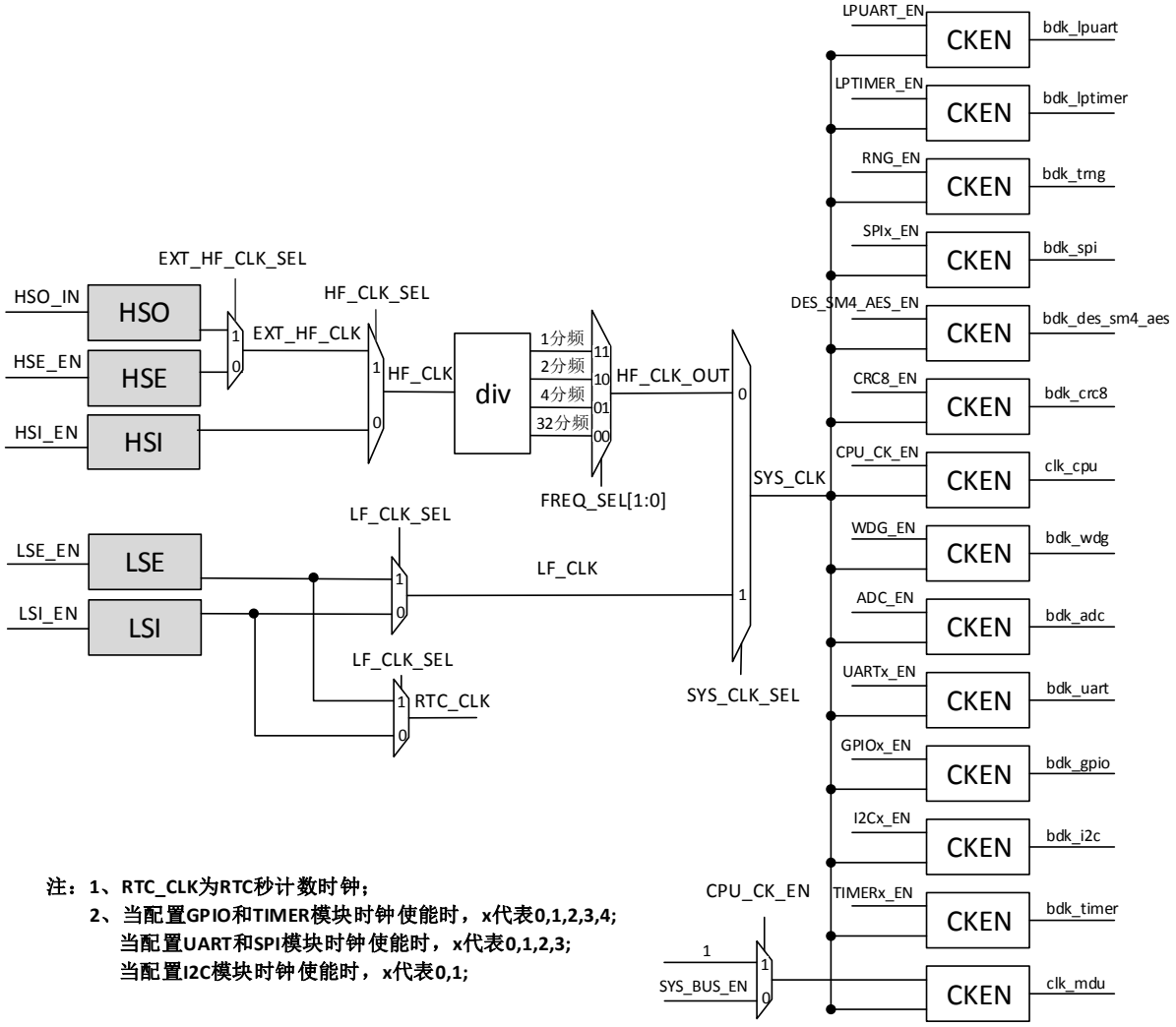


图 7-1 时钟模块框图

7.1. 主时钟源

系统时钟有 5 个时钟源可选：

- 外部高速时钟 HSE：4~24MHz 无源晶振
- 外部高速时钟 HSO：1~16MHz 有源晶振
- 内部高速时钟 HSI：16MHz RC 振荡器
- 外部低速时钟 LSE：32.768KHz 无源晶振
- 内部低速时钟 LSI：32KHz RC 振荡器

Copyright © 2022 武汉瑞纳捷半导体有限公司

7.1.1. HSE

HSE 由外部无源晶振产生，允许时钟频率范围为 4~24MHz，可通过设置系统时钟选择寄存器 SCSYS 的 bit3，时钟控制寄存器 SCCM2 的 bit7 来打开或关闭。其优点在于能够为系统时钟提供精确的占空比为 50%的时钟信号。

7.1.2. HSO

HSO 由外部有源晶振产生，允许时钟频率范围为 1~16MHz，作系统时钟时需要将相应的 P07 管脚复用为 HSO_IN 功能，同时通过设置系统时钟选择寄存器 SCSYS 的 bit3，时钟控制寄存器 SCCM1 的 bit7 来选择。

7.1.3. HSI

高速内部时钟信号由内部 RC 振荡器产生。

HSI 由 16MHz 内部 RC 振荡器产生，可通过设置系统时钟选择寄存器 SCSYS 的 bit1、bit3 来开启或关闭。HSI 可为系统时钟提供时钟信号。

7.1.4. LSE

LSE 由 32.768KHz 外部无源晶振产生，可通过 OSC 控制寄存器 LP_CTRL1 的 bit1 来开启或关闭。LSE 可为系统时钟或外设 RTC 提供准确的时钟信号。

7.1.5. LSI

LSI 由 CLKHz 内部 RC 振荡器产生，可通过 LSI 时钟控制寄存器 LSI_CTRL 的 bit7 来开启或关闭。LSI 可为系统时钟或外设 RTC 提供时钟信号。

7.2. 系统时钟源选择

7.2.1. 系统启动

为了系统快速启动，复位后时钟默认使用 HSI 的 2 分频作为系统时钟。如有需要，用户可将系统时钟切换到另外的时钟源。

7.2.2. 系统时钟切换

用户根据需要可以手动切换系统时钟，可通过配置系统时钟选择寄存器 SCSYS 的第 0,1,3,6,7 比特位，OSC 控制寄存器 LP_CTRL1 的第 1,3 比特位，时钟控制寄存器 SCCM1、SCCM2、LSI_CTRL 的 bit7 来选择系统时钟源：HSO、HSE、HSI、LSE、LSI。

注：时钟切换时，必须先开启并使能要切换的时钟，再关闭切换前的时钟。

7.3. RTC 时钟选择

RTC 时钟可通过配置寄存器 LSI_CTRL 的 bit7、LP_CTRL1 的 bit1 与 bit3 来选择时钟源：LSI 或 LSE。

7.4. 外设时钟选择

Copyright © 2022 武汉瑞纳捷半导体有限公司

除 RTC 设备外，其他外设的工作时钟，可通过寄存器 SCCM0，SCCM1，SCCM2 中对应控制位来开启或关闭。

7.5. 时钟控制寄存器

时钟控制寄存器基地址：0x2000

地址	名称	描述
0x15	SCSYS	系统时钟选择寄存器
0x10	LP_CTRL1	OSC 控制寄存器
0x13	SCCM0	时钟控制寄存器 0
0x14	SCCM1	时钟控制寄存器 1
0x19	SCCM2	时钟控制寄存器 2
0Xc0	LSI_CTRL	LSI 时钟控制寄存器

7.5.1. 系统时钟选择寄存器 SCSYS（偏移：15H）

比特	名称	属性	复位值	描述
7：6	FREQ_SEL	RW	0x2	高频工作时钟 HF_CLK_OUT 选择，分频源头为 HF_CLK 00: 32 分频 01: 4 分频 10: 2 分频 11: 1 分频 ***注：默认系统时钟是 16MHz 的 2 分频（8MHz）***
5:4	RSV	-	-	保留
3	HF_CLK_SEL	RW	0x0	HF_CLK 源头选择位 1：选择 EXT_HF_CLK（选中前通过 SCCM2 寄存器的 bit7 配使能）； 0：选择 HSI（选中前通过 HSI_EN 配使能）；
2	HALT	RW	0x0	HALT 模式选择位 0：无影响 1：进入 HALT 模式
1	HSI_EN	RW	0x0	HSI 时钟使能控制位 0：使能 1：禁止
0	SYS_CLK_SEL	RW	0x0	系统时钟 SYS_CLK 选择位 0：HF_CLK_OUT 1：LF_CLK（选中后需通过配置 LP_CTRL1 寄存器选择 LSI 或 LSE）

7.5.2. OSC 控制寄存器 LP_CTRL 1(偏移：10H)

Copyright © 2022 武汉瑞纳捷半导体有限公司

比特	名称	属性	复位值	描述
7	SYS_BUS_EN	RW	0x0	系统总线时钟使能位 0：禁止 1：使能
6	CPU_CK_EN	RW	0x1	CPU 时钟使能位 0：禁止 1：使能
5	FLASH_DPSDB	RW	0x0	Flash Standby 使能位 0：Flash 处于正常状态 1：Flash 处于低功耗状态
4	FLASH_CE	RW	0x0	Flash 使能位 0：Flash 使能 1：Flash 不使能
3	LF_CLK_SEL	RW	0x0	LF_CLK 低频时钟来源选择位 0：LSI； 1：LSE
2	RSV			
1	LSE_EN	RW	0x0	LSE 时钟使能控制位 0：禁止 1：使能；
0	RSV	-	-	保留

7.5.3. 时钟控制寄存器 SCCM0（偏移：13H）

比特	名称	属性	复位值	描述
7	WDG_EN	RW	0x0	WDG 时钟使能控制位 0：禁止 1：使能
6	I2C1_EN	RW	0x0	I2C1 时钟使能控制位 0：禁止 1：使能
5	I2C0_EN	RW	0x0	I2C0 时钟使能控制位 0：禁止 1：使能
4	SPI3_EN	RW	0x0	SPI3 时钟使能控制 0：禁止 1：使能
3	SPI2_EN	RW	0x0	SPI2 时钟使能控制 0：禁止 1：使能
2	SPI1_EN	RW	0x0	SPI1 时钟使能控制

Copyright © 2022 武汉瑞纳捷半导体有限公司

				0：禁止 1：使能
1	SPI0_EN	RW	0x0	SPI0 时钟使能控制 0：禁止 1：使能
0	RSV	-	-	保留

7.5.4. 时钟控制寄存器 SCCM1（偏移：14H）

比特	名称	属性	复位值	描述
7	EXT_HF_CLK_SEL	RW	0x0	EXT_HF_CLK 时钟源选择： 0：选择 HSE 时钟； 1：选择 HSO 时钟；
6	RNG_EN	RW	0x0	RNG 时钟使能控制 0：禁止 1：使能
5	GPIO_CKEN	RW	0x0	GPIO 时钟使能位 0：禁止 1：使能
4	TIMER4_EN	RW	0x0	TIMER4 时钟使能控制 0：禁止 1：使能
3	TIMER3_EN	RW	0x0	TIMER3 时钟使能控制 0：禁止 1：使能
2	TIMER2_EN	RW	0x0	TIMER2 时钟使能控制 0：禁止 1：使能
1	TIMER1_EN	RW	0x0	TIMER1 时钟使能控制 0：禁止 1：使能
0	TIMER0_EN	RW	0x0	TIMER0 时钟使能控制 0：禁止 1：使能

7.5.5. 时钟控制寄存器 SCCM2（偏移：19H）

比特	名称	属性	复位值	描述
7	HSE_EN	RW	0x0	HSE 时钟使能控制位 0：禁止； 1：使能
6	ADC_EN	RW	0x0	ADC 时钟使能控制位

Copyright © 2022 武汉瑞纳捷半导体有限公司

				0：禁止 1：使能
5	DES_SM4_AES_EN	RW	0x0	DES/SM4/AES 时钟使能控制位 0：禁止 1：使能
4	CRC_EN	RW	0x0	CRC 时钟使能控制位 0：禁止 1：使能
3	UART 3_EN	RW	0x0	UART 3 时钟使能控制位 0：禁止 1：使能
2	UART 2_EN	RW	0x0	UART 2 时钟使能控制位 0：禁止 1：使能
1	UART 1_EN	RW	0x0	UART 1 时钟使能控制位 0：禁止 1：使能
0	UART0_EN	RW	0x0	UART 0 时钟使能控制位 0：禁止 1：使能

7.5.6. LSI 时钟控制寄存器 LSI_CTRL（偏移：C0H）

比特	名称	属性	复位值	描述
7	LSI_EN	RW	0x0	内部 LSI 时钟振荡器使能控制 0：使能 1：关闭
6:2	LSI_TRIM	RW	0x0F	内部 LSI 时钟振荡器校准寄存器 ***注：中心值为 0x0F，根据 FT 实测情况来调整。当实测值高于 32K 时，这个 trim 值就往下降一档，直至实测值很接近 32K。***
1:0	LSI_TRIM_CURR	RW	0x0	内部 LSI 时钟振荡器校准时电流控制寄存器 ***注：实际使用时就用默认值。***

Copyright © 2022 武汉瑞纳捷半导体有限公司

8. 通用输入输出口（GPIO）及端口复用

8.1. 概述

芯片有 5 组 GPIO 通用输入/输出接口：GPIO0、GPIO1、GPIO2、GPIO3 和 GPIO4。GPIO 引脚多达 40 个，具体视封装而定，每个引脚都可以被独立编程作为数字输入或数字输出口。另外部分端口还可以作为模拟输入、外部中断、片上外设的输入/输出等复用功能。在同一时刻仅有一个复用功能可以映射到引脚上。复用功能可通过相关寄存器进行控制。此外，每个端口都可作为外部中断信号输入。

每个端口都分配有一个输出/输入数据寄存器，一个数据方向寄存器。

8.2. GPIO 主要特性

- GPIO 端口支持浮空输入、上拉输入、模拟输入、推挽输出；
- 片上外设的 IO 功能复用；
- 外部中断可以单独使能和关闭（支持边沿触发和电平触发）；
- 管脚输出驱动能力： 4.5mA(3.3V)/6mA(5V)。

8.3. I/O 的配置和使用

8.3.1. 普通输入/输出 I/O

每个端口都有一个数据方向控制寄存器 Px_OEN(x=0~4)，控制选择数据方向。一个数据输入/输出寄存器 Px，用于输入/输出数字信号。

8.3.2. 作为外部中断源

每个端口都可以作为外部中断源。

在配置中断时，需要将端口复用配置为 GPIO 口，数据方向配置为输入，触发方式配置寄存器 Px_INT_SELy（x=0~4，y=0~2），选择边沿触发/电平触发，配置中断使能寄存器 Px_INT_EN 选择相应位的中断，此外还需配置中断服务程序单元（ISR），具体参照相关章节。

中断标志通过寄存器 Px_INT_REG 来清除。

8.4. I/O 管脚复用寄存器

芯片每个 GPIO 端口都有相应的复用功能，详见管脚复用控制寄存器。

管脚复用寄存器基地址：0x2000

偏移	名称	描述
0x70	P0_PU	GPIO 0 管脚上拉寄存器
0x71	P1_PU	GPIO 1 管脚上拉寄存器

Copyright © 2022 武汉瑞纳捷半导体有限公司

0x72	P2_PU	GPIO 2 管脚上拉寄存器
0x73	P3_PU	GPIO 3 管脚上拉寄存器
0x74	P4_PU	GPIO 4 管脚上拉寄存器
0x7F	P0_IEN	GPIO 0 管脚输入使能寄存器
0xCD	P1_IEN	GPIO 1 管脚输入使能寄存器
0xCE	P2_IEN	GPIO 2 管脚输入使能寄存器
0xCF	P3_IEN	GPIO 3 管脚输入使能寄存器
0xFF	P4_IEN	GPIO 4 管脚输入使能寄存器
0xF0	IOCFG_CTRL0	管脚复用控制寄存器 0
0xF7	IOCFG_CTRL1	管脚复用控制寄存器 1
0xF8	IOCFG_CTRL2	管脚复用控制寄存器 2
0xF9	IOCFG_CTRL3	管脚复用控制寄存器 3
0xFA	IOCFG_CTRL4	管脚复用控制寄存器 4
0xFB	IOCFG_CTRL5	管脚复用控制寄存器 5
0xFC	IOCFG_CTRL6	管脚复用控制寄存器 6
0xFD	IOCFG_CTRL7	管脚复用控制寄存器 7
0xFE	IOCFG_CTRL8	管脚复用控制寄存器 8
0xC8	AD_SEL1	管脚复用为模拟功能或数字功能选择寄存器 1
0xC9	AD_SEL2	管脚复用为模拟功能或数字功能选择寄存器 2

8.4.1. GPIOx 管脚上拉寄存器 Px_PU（偏移：70H、71H、72H、73H 和 74H）

比特	名称	属性	复位值	描述
7:0	PAD_UP	RW	0x0	GPIOx 对应 PAD 内部上拉使能位 0：不使能 1：使能

注：此处 x 为 0,1,2,3,4, 下同。

8.4.2. GPIOx 管脚输入使能控制寄存器 Px_IEN（偏移：7FH、CDH、CEH、CFH 和 FFH）

比特	名称	属性	复位值	描述
7:0	PAD_IEN	RW	0xFF	GPIOx 对应 PAD 输入使能选择位 0：不使能 PAD 输入功能 1：使能 PAD 输入功能

8.4.3. 管脚复用模拟功能或数字功能选择寄存器 AD_SEL1（偏移：C8H）

比特	名称	属性	复位值	描述
7	ADC_IO_EN[7]	RW	0x0	1：P23 管脚复用为 ADC_IN7 0：P23 管脚复用为数字功能（由 IOCFG_CTRLx 来选定）
6	ADC_IO_EN[6]	RW	0x0	1：P24 管脚复用为 ADC_IN6

Copyright © 2022 武汉瑞纳捷半导体有限公司

				0：P24 管脚复用为数字功能（由 IOCFG_CTRLx 来选定）
5	ADC_IO_EN[5]	RW	0x0	1：P25 管脚复用为 ADC_IN5 0：P25 管脚复用为数字功能（由 IOCFG_CTRLx 来选定）
4	ADC_IO_EN[4]	RW	0x0	1：P26 管脚复用为 ADC_IN4 0：P26 管脚复用为数字功能（由 IOCFG_CTRLx 来选定）
3	ADC_IO_EN[3]	RW	0x0	1：P27 管脚复用为 ADC_IN3 0：P27 管脚复用为数字功能（由 IOCFG_CTRLx 来选定）
2	ADC_IO_EN[2]	RW	0x0	1：P44 管脚复用为 ADC_IN2 0：P44 管脚复用为数字功能（由 IOCFG_CTRLx 来选定）
1	ADC_IO_EN[1]	RW	0x0	1：P45 管脚复用为 ADC_IN1 0：P45 管脚复用为数字功能（由 IOCFG_CTRLx 来选定）
0	RSV	RW	0x0	必须保留为 0

8.4.4. 管脚复用模拟功能或数字功能选择寄存器 AD_SEL2（偏移：C9H）

比特	名称	属性	复位值	描述
7	RSV	RW	0x0	
6:5	ADC_TIM_TRI_SEL	RW	0x0	ADC 外部触发 TIMER 通道源头选择： 00：TIMER2； 01：TIMER3； 10：TIMER4； 11：LPTIMER；
4	CMP_IO_EN[2]	RW	0x0	1：P32 管脚复用为 COMP2_IN_P 0：P32 管脚复用为数字功能（由 IOCFG_CTRLx 来选定）
3	CMP_IO_EN[1]	RW	0x0	1：P31 管脚复用为 COMP1_IN_P 0：P31 管脚复用为数字功能（由 IOCFG_CTRLx 来选定）
2	CMP_IO_EN[0]	RW	0x0	1：P30 管脚复用为 COMP2_IN_N 0：P30 管脚复用为数字功能（由 IOCFG_CTRLx 来选定）
1	ADC_IO_EN[9]	RW	0x0	1：P20 管脚复用为 ADC_REFN 0：P20 管脚复用为数字功能（由 IOCFG_CTRLx 来选定）
0	ADC_IO_EN[8]	RW	0x0	1：P21 管脚复用为 ADC_REFP

Copyright © 2022 武汉瑞纳捷半导体有限公司

				0：P21 管脚复用为数字功能（由 IOCFG_CTRLx 来选定）

8.4.5. 管脚复用控制寄存器 IOCFG_CTRL0（偏移：F0H）

比特	名称	属性	复位值	描述
7:6	IOCFG_CTRL0[7:6]	RW	0x0	00：GPIO03 01：UART0_TX 10：LF_CLK_OUT 11：TIM2_CC2（TIMER2 的比较通道 2）
5:4	IOCFG_CTRL0[5:4]	RW	0x0	00：GPIO02 01：IIC0_SDA 10：REV（保留） 11：TIM2_CC1（TIMER2 的比较通道 1）
3:2	IOCFG_CTRL0[3:2]	RW	0x0	00：RJ_SCK 01：REV（保留） 10：GPIO01 11：TIM2_CC0（TIMER2 的比较通道 0）
1:0	IOCFG_CTRL0[1:0]	RW	0x0	00：RJ_SDA 01：REV（保留） 10：GPIO00 11：TIM2_ERT（TIMER2 的外部重载触发）

8.4.6. 管脚复用控制寄存器 IOCFG_CTRL1（偏移：F7H）

比特	名称	属性	复位值	描述
7	IOCFG_CTRL1[7]	RW	0x0	0：GPIO10 1：UART2_RX/LPUART_RX
6	IOCFG_CTRL1[6]	RW	0x0	0：GPIO06 1：IIC0_SCL
5:4	IOCFG_CTRL1[5:4]	RW	0x0	00：GPIO07 01：HSO_IN 10：REV（保留） 11：REV（保留）
3:2	IOCFG_CTRL1[3:2]	RW	0x0	00：GPIO05 01：REV（保留） 10：REV（保留） 11：REV（保留）
1:0	IOCFG_CTRL1[1:0]	RW	0x0	00：GPIO04 01：UART0_RX 10：HF_CLK_OUT 11：REV（保留）

Copyright © 2022 武汉瑞纳捷半导体有限公司

8.4.7. 管脚复用控制寄存器 IOCFG_CTRL2（偏移：F8H）

比特	名称	属性	复位值	描述
7:6	IOCFG_CTRL2[7:6]	RW	0x0	00：GPIO14 01：SPI2_MOSI_M 10：SPI2_MOSI_S 11：TIM3_CC1（TIMER3 的比较通道 1）
5:4	IOCFG_CTRL2[5:4]	RW	0x0	00：GPIO13 01：SPI2_MISO_S 10：SPI2_MISO_M 11：TIM3_CC0N（TIMER3 的比较互补通道 0）
3:2	IOCFG_CTRL2[3:2]	RW	0x0	00：GPIO12 01：SPI2_CLK_M 10：SPI2_CLK_S 11：TIM3_CC0（TIMER3 的比较通道 0）
1:0	IOCFG_CTRL2[1:0]	RW	0x0	00：GPIO11 01：UART2_TX 10：LPUART_TX 11：TIM3_CAP0（TIMER3 的捕获通道 0）

8.4.8. 管脚复用控制寄存器 IOCFG_CTRL3（偏移：F9H）

比特	名称	属性	复位值	描述
7:6	IOCFG_CTRL3[7:6]	RW	0x0	00：GPIO40 01：IIC0_SDA 10：REV（保留） 11：TIM3_ERT（TIMER3 的外部重载触发）
5:4	IOCFG_CTRL3[5:4]	RW	0x0	00：GPIO17 01：IIC0_SCL 10：REV（保留） 11：TIM3_CC2N（TIMER3 的比较互补通道 2）
3	IOCFG_CTRL3[3]	RW	0x0	0：GPIO41 1：TIM3_ETR（TIMER3 的外部事件触发）
2	IOCFG_CTRL3[2]	RW	0x0	0：GPIO16 1：TIM3_CC2（TIMER3 的比较通道 2）
1:0	IOCFG_CTRL3[1:0]	RW	0x0	00：GPIO15 01：SPI2_CS 10：REV（保留） 11：TIM3_CC1N（TIMER3 的比较互补通道 1）

8.4.9. 管脚复用控制寄存器 IOCFG_CTRL4（偏移：FAH）

比特	名称	属性	复位值	描述

Copyright © 2022 武汉瑞纳捷半导体有限公司

7	IOCFG_CTRL4[7]	RW	0x0	0：GPIO46 1：UART1_TX
6	IOCFG_CTRL4[6]	RW	0x0	0：GPIO23 1：SPI0_CS
5:4	IOCFG_CTRL4[5:4]	RW	0x0	00：GPIO22 01：SPI0_MOSI_M 10：SPI0_MOSI_S 11：ADC_ETR
3:2	IOCFG_CTRL4[3:2]	RW	0x0	00：GPIO21 01：SPI0_MISO_M 10：SPI0_MISO_S 11：TIM4_CAP2（TIMER4 的捕获通道 2 输入）
1:0	IOCFG_CTRL4[1:0]	RW	0x0	00：GPIO20 01：SPI0_CLK_M 10：SPI0_CLK_S 11：TIM4_CAP1（TIMER4 的捕获通道 1 输入）

8.4.10. 管脚复用控制寄存器 IOCFG_CTRL5（偏移：FBH）

比特	名称	属性	复位值	描述
7:6	IOCFG_CTRL5[7:6]	RW	0x0	00：GPIO27 01：SPI1_CS 10：REV（保留） 11：TIM4_CC1N（TIMER4 的比较互补通道 1）
5:4	IOCFG_CTRL5[5:4]	RW	0x0	00：GPIO26 01：SPI1_MOSI_M 10：SPI1_MOSI_S 11：TIM4_CC1（TIMER4 的比较通道 1）
3:2	IOCFG_CTRL5[3:2]	RW	0x0	00：GPIO25 01：SPI1_MISO_M 10：SPI1_MISO_S 11：TIM4_CC0N（TIMER4 的比较互补通道 0）
1:0	IOCFG_CTRL5[1:0]	RW	0x0	00：GPIO24 01：SPI1_CLK_M 10：SPI1_CLK_S 11：TIM4_CC0（TIMER4 的比较通道 0）

8.4.11. 管脚复用控制寄存器 IOCFG_CTRL6（偏移：FCH）

比特	名称	属性	复位值	描述
7:4	IOCFG_CTRL6[7:6]	RW	0x0	REV（保留）
3:2	IOCFG_CTRL6[3:2]	RW	0x0	00：GPIO45

Copyright © 2022 武汉瑞纳捷半导体有限公司

				01：UART1_RX 10：REV（保留） 11：TIM4_CC2N（TIMER4 的比较互补通道 2）
1:0	IOCFG_CTRL6[1:0]	RW	0x0	00：GPIO44 01：REV（保留） 10：REV（保留） 11：TIM4_CC2（TIMER4 的比较通道 2）

8.4.12. 管脚复用控制寄存器 IOCFG_CTRL7（偏移：FDH）

比特	名称	属性	复位值	描述
7:6	IOCFG_CTRL7[7:6]	RW	0x0	00：GPIO33 01：UART3_TX 10：REV（保留） 11：TIM3_BKIN（TIMER3 的刹车输入）
5:4	IOCFG_CTRL7[5:4]	RW	0x0	00：GPIO32 01：UART3_RX 10：REV（保留） 11：TIM4_ETR（TIMER4 的外部事件触发）
3:2	IOCFG_CTRL7[3:2]	RW	0x0	00：GPIO31 01：IIC1_SDA 10：REV（保留） 11：TIM4_ERT（TIMER4 的外部重载触发）
1:0	IOCFG_CTRL7[1:0]	RW	0x0	00：GPIO30 01：IIC1_SCL 10：REV（保留） 11：TIM2_ETR（TIMER2 的外部事件触发）

8.4.13. 管脚复用控制寄存器 IOCFG_CTRL8（偏移：FEH）

比特	名称	属性	复位值	描述
7	IOCFG_CTRL8[7]	RW	0x0	0：GPIO42/GPIO43 1：TIM3_CAP1（TIMER3 捕获通道 1） /TIM3_CAP2（TIMER3 捕获通道 2）
6	IOCFG_CTRL8[6]	RW	0x0	0：GPIO37 1：SPI3_CS
5:4	IOCFG_CTRL8[5:4]	RW	0x0	00：GPIO36 01：SPI3_MOSI_M 10：SPI3_MOSI_S 11：LPTIM_ETR（LPTIMER 的外部事件触发）
3:2	IOCFG_CTRL8[3:2]	RW	0x0	00：GPIO35 01：SPI3_MISO_M

Copyright © 2022 武汉瑞纳捷半导体有限公司

				10：SPI3_MISO_S 11：TIM4_CAP0（TIMER4 的捕获通道 0 输入）
1:0	IOCFG_CTRL8[1:0]	RW	0x0	00：GPIO34 01：SPI3_CLK_M 10：SPI3_CLK_S 11：TIM4_BKIN（TIMER4 的刹车输入）

8.4.14. GPIO 功能复用表

Copyright © 2022 武汉瑞纳捷半导体有限公司

TSSOP20	QFN32	LQFP48	引脚名称	复用功能寄存器配置						
				模拟/数字功能选择（AD_SEL）：1：管脚模拟功能（AF）；0：管脚数字功能（DF0-DF3）；						
				数字功能复用控制(DF_SEL)：0：DF0；1：DF1；2：DF2；3：DF3；						
				DF0	DF1	DF2	DF3	AF	AD_SEL	DF_SEL
-	-	1	P10	GPIO10	UART2_RX /LPUART_RX					IOCFG_CTRL1[7]
-	-	2	P11	GPIO11	UART2_TX	LPUART_TX	TIM3_CAP0			IOCFG_CTRL2[1:0]
-	2	3	P12	GPIO12	SPI2_CLK_M	SPI2_CLK_S	TIM3_CC0 比较通道 0			IOCFG_CTRL2[3:2]
-	3	4	P13	GPIO13	SPI2_MISO_S	SPI2_MISO_M	TIM3_CC0N 比较互补通道 0			IOCFG_CTRL2[5:4]
-	4	5	P14	GPIO14	SPI2_MOSI_M	SPI2_MOSI_S	TIM3_CC1 比较通道 1			IOCFG_CTRL2[7:6]
-	5	6	P15	GPIO15	SPI2_CS		TIM3_CC1N 比较互补通道 1			IOCFG_CTRL3[1:0]
-	6	7	P16	GPIO16	TIM3_CC2 比较通道 2					IOCFG_CTRL3[2]
-	7	8	P17	GPIO17	IIC0_SCL		TIM3_CC2N 比较互补通道 2			IOCFG_CTRL3[5:4]
-	-	9	P40	GPIO40	IIC0_SDA		TIM3_ERT 外部重载触发			IOCFG_CTRL3[7:6]
-	-	10	P41	GPIO41	TIM3_ETR 外部事件触发					IOCFG_CTRL3[3]
-	-	11	NC							
-	-	12	P42	GPIO42	TIM3_CAP1					IOCFG_CTRL8[7]
-	-	13	P43	GPIO43	TIM3_CAP2					IOCFG_CTRL8[7]
11	8	14	P20	GPIO20	SPI0_CLK_M	SPI0_CLK_S	TIM4_CAP1	ADC_REFN	ADC_IO_EN[9]	IOCFG_CTRL4[1:0]
12	9	15	P21	GPIO21	SPI0_MISO_M	SPI0_MISO_S	TIM4_CAP2	ADC_REFP	ADC_IO_EN[8]	IOCFG_CTRL4[3:2]
-	-	16	P22	GPIO22	SPI0_MOSI_M	SPI0_MOSI_S	ADC_ETR 外部触发通道			IOCFG_CTRL4[5:4]
13	10	17	P23	GPIO23	SPI0_CS			ADC_IN7	ADC_IO_EN[7]	IOCFG_CTRL4[6]
14	11	18	P24	GPIO24	SPI1_CLK_M	SPI1_CLK_S	TIM4_CC0 比较通道 0	ADC_IN6	ADC_IO_EN[6]	IOCFG_CTRL5[1:0]
15	12	19	P25	GPIO25	SPI1_MISO_M	SPI1_MISO_S	TIM4_CC0N 比较互补通道 0	ADC_IN5	ADC_IO_EN[5]	IOCFG_CTRL5[3:2]
16	13	20	P26	GPIO26	SPI1_MOSI_M	SPI1_MOSI_S	TIM4_CC1 比较通道 1	ADC_IN4	ADC_IO_EN[4]	IOCFG_CTRL5[5:4]
17	14	21	P27	GPIO27	SPI1_CS		TIM4_CC1N 比较互补通道 1	ADC_IN3	ADC_IO_EN[3]	IOCFG_CTRL5[7:6]
18	15	22	P44	GPIO44			TIM4_CC2 比较通道 2	ADC_IN2	ADC_IO_EN[2]	IOCFG_CTRL6[1:0]
19	16	23	P45	GPIO45	UART1_RX		TIM4_CC2N 比较互补通道 2	ADC_IN1	ADC_IO_EN[1]	IOCFG_CTRL6[3:2]
-	17	24	P46	GPIO46	UART1_TX					IOCFG_CTRL4[7]
-	-	25	P47	GPIO47						
-	-	26	P30	GPIO30	IIC1_SCL		TIM2_ETR 外部事件触发	COMP2_IN_N	CMP_IO_EN[0]	IOCFG_CTRL7[1:0]
-	-	27	P31	GPIO31	IIC1_SDA		TIM4_ERT 外部重载触发	COMP1_IN_P	CMP_IO_EN[1]	IOCFG_CTRL7[3:2]
-	-	28	P32	GPIO32	UART3_RX		TIM4_ETR 外部事件触发	COMP2_IN_P	CMP_IO_EN[2]	IOCFG_CTRL7[5:4]

Copyright © 2022 武汉瑞纳捷半导体有限公司

-	-	29	P33	GPIO33	UART3_TX		TIM3_BKIN 刹车输入			IOCFG_CTRL7[7:6]
20	18	30	P34	GPIO34	SPI3_CLK_M	SPI3_CLK_S	TIM4_BKIN 刹车输入			IOCFG_CTRL8[1:0]
-	19	31	P35	GPIO35	SPI3_MISO_M	SPI3_MISO_S	TIM4_CAP0			IOCFG_CTRL8[3:2]
-	20	32	P36	GPIO36	SPI3_MOSI_M	SPI3_MOSI_S	LPTIM_ETR 外部事件触发			IOCFG_CTRL8[5:4]
-	21	33	P37	GPIO37	SPI3_CS					IOCFG_CTRL8[6]
-	22	34	HSE_I							
-	23	35	HSE_O							
-	24	36	LSE_O							
-	25	37	LSE_I							
2	26	38	P00	RJ_SDA		GPIO00	TIM2_ERT 外部重载触发			IOCFG_CTRL0[1:0]
3	27	39	P01	RJ_SCK		GPIO01	TIM2_CC0 比较通道 0			IOCFG_CTRL0[3:2]
4	28	40	P02	GPIO02	IIC0_SDA		TIM2_CC1 比较通道 1			IOCFG_CTRL0[5:4]
5	29	41	P03	GPIO03	UART0_TX	LF_CLK_OUT	TIM2_CC2 比较通道 2			IOCFG_CTRL0[7:6]
6		42	P04	GPIO04	UART0_RX	HF_CLK_OUT				IOCFG_CTRL1[1:0]
7		43	P05	GPIO05						IOCFG_CTRL1[3:2]
8		44	P06	GPIO06	IIC0_SCL					IOCFG_CTRL1[6]
-	30	45	P07	GPIO07	HSO_IN					IOCFG_CTRL1[5:4]
9	31	46	RST							
1	32	47	VSS							
10	1	48	VDD							

注：

1) ***I2C0/1 接口分别对应不同组 IO 引脚，但同一 I2C 接口的主/从模式下共用同一组 IO 引脚。同时 I2C0 接口可以复用至两组不同 IO 引脚，I2C1 接口仅可以复用至一组固定的 IO 引脚。***
2) ***SPI0/1/2/3 接口分别对应不同组 IO 引脚，但同一 SPI 接口的主/从模式下共用同一组 IO 引脚，却对应不同的复用功能。***
3) ***UART0/1/2/3 接口分别对应不同组 IO 引脚，其中 UART2 与 LPUART 共用同一组 IO 引脚，而且 UART2_RX 与 LPUART_RX 对应同一 IO 的相同复用功能，UART2_TX 与 LPUART_TX 对应同一 IO 的不同复用功能。***
4) ***TIMER3 的输入捕获通道 1/2：TIM3_CAP1、TIM3_CAP2，分别对应 GPIO42、GPIO43 引脚的复用功能，却由 IOCFG_CTRL8[7]进行共同控制。***
5) ***"LF_CLK_OUT"即低频时钟输出，通过使能并选择低频时钟源 LSE 或 LSI，同时将 P03 功能复用为 DF2 即可输出。"HF_CLK_OUT"即高频时钟输出，通过使能并选择高频时钟源 HSE 或 HSI 或 HSO，同时将 P04 功能复用为 DF2 而后经固定 16 分频后即可输出。***

8.5. GPIO 寄存器

GPIO 特殊功能寄存器

偏移	名称	描述
0x80	P0	GPIO0 数据寄存器

Copyright © 2022 武汉瑞纳捷半导体有限公司

0x90	P1	GPIO1 数据寄存器
0xA0	P2	GPIO2 数据寄存器
0xB0	P3	GPIO3 数据寄存器
0x91	P4	GPIO4 数据寄存器
0xA4	P0_OEN	GPIO0 输出使能寄存器
0xA5	P1_OEN	GPIO1 输出使能寄存器
0xA6	P2_OEN	GPIO2 输出使能寄存器
0xA7	P3_OEN	GPIO3 输出使能寄存器
0x96	P4_OEN	GPIO4 输出使能寄存器

GPIO 外部中断寄存器基地址：0x2000

偏移	名称	描述
0xD0	P0_INT_EN	GPIO0 中断使能寄存器
0xD1	P0_INT_SEL0	GPIO0 口中断触发方式选择寄存器低字节
0xD2	P0_INT_SEL1	GPIO0 口中断触发方式选择寄存器中间字节
0xD3	P0_INT_SEL2	GPIO0 口中断触发方式选择寄存器高字节
0xD4	P0_INT_REG	GPIO0 中断状态寄存器
0xD5	P1_INT_EN	GPIO1 中断使能寄存器
0xD6	P1_INT_SEL0	GPIO1 口中断触发方式选择寄存器低字节
0xD7	P1_INT_SEL1	GPIO1 口中断触发方式选择寄存器中间字节
0xD8	P1_INT_SEL2	GPIO1 口中断触发方式选择寄存器高字节
0xD9	P1_INT_REG	GPIO1 中断状态寄存器
0xDA	P2_INT_EN	GPIO2 中断使能寄存器
0xDB	P2_INT_SEL0	GPIO2 口中断触发方式选择寄存器低字节
0xDC	P2_INT_SEL1	GPIO2 口中断触发方式选择寄存器中间字节
0xDD	P2_INT_SEL2	GPIO2 口中断触发方式选择寄存器高字节
0xDE	P2_INT_REG	GPIO2 中断状态寄存器
0xDF	P3_INT_EN	GPIO3 中断使能寄存器
0xE0	P3_INT_SEL0	GPIO3 口中断触发方式选择寄存器低字节
0xE1	P3_INT_SEL1	GPIO3 口中断触发方式选择寄存器中间字节
0xE2	P3_INT_SEL2	GPIO3 口中断触发方式选择寄存器高字节
0xE3	P3_INT_REG	GPIO3 中断状态寄存器
0xE4	P4_INT_EN	GPIO4 中断使能寄存器
0xE5	P4_INT_SEL0	GPIO4 口中断触发方式选择寄存器低字节
0xE6	P4_INT_SEL1	GPIO4 口中断触发方式选择寄存器中间字节
0xE7	P4_INT_SEL2	GPIO4 口中断触发方式选择寄存器高字节
0xE8	P4_INT_REG	GPIO4 中断状态寄存器

8.5.1. GPIOx 数据寄存器 Px（偏移：80H、90H、A0H、B0H、91H）

Copyright © 2022 武汉瑞纳捷半导体有限公司

比特	名称	属性	复位值	描述
7:0	Px	RW	0xFF	数据寄存器 例如：GPIO1.5 输出高电平，则 P1=0x20； GPIO1.5 输入，则直接读取 P1 寄存器，

注：此处 x 为 0,1,2,3,4，下同。

8.5.2. GPIOx 输出使能寄存器 Px_OEN（偏移：A4H、A5H、A6H、A7H、96H）

比特	名称	属性	复位值	描述
7:0	Px_OEN	RW	0x00	配置 GPIOx 口的输出使能 例如：将 GPIO2.3 配置成输出，则 P2_OEN=0x08；

8.5.3. GPIOx 中断使能寄存器 Px_INT_EN (偏移：D0H、D5H、DAH、DFH、E4H)

比特	名称	属性	复位值	描述
7:0	Px_INT_EN	RW	0x00	配置 GPIOx 口的中断使能 例如，将 GPIO3.4 配置成中断使能，则 P3_INT_EN=0x10；

8.5.4. GPIOx 中断触发方式选择寄存器 Px_INT_SELy (偏移：详见寄存器列表)

注：x=0~4，y=0~2，下同

比特	名称	属性	复位值	描述
7:5	Px_INT_SEL2	RW	0x00	GPIOx.7 的中断触发方式选择： 3'b000：高电平中断 3'b001：低电平中断 3'b010：上升沿中断 3'b011：下降沿中断 3'b100：双沿中断
4:2	Px_INT_SEL2	RW	0x00	GPIOx.6 的中断触发方式选择： 3'b000：高电平中断 3'b001：低电平中断 3'b010：上升沿中断 3'b011：下降沿中断 3'b100：双沿中断
1:0 7	Px_INT_SEL2 Px_INT_SEL1	RW	0x00	GPIOx.5 的中断触发方式选择： 3'b000：高电平中断 3'b001：低电平中断 3'b010：上升沿中断 3'b011：下降沿中断 3'b100：双沿中断
6:4	Px_INT_SEL1	RW	0x00	GPIOx.4 的中断触发方式选择： 3'b000：高电平中断 3'b001：低电平中断

Copyright © 2022 武汉瑞纳捷半导体有限公司

				3'b010：上升沿中断 3'b011：下降沿中断 3'b100：双沿中断
3:1	Px_INT_SEL1	RW	0x00	GPIOx.3 的中断触发方式选择： 3'b000：高电平中断 3'b001：低电平中断 3'b010：上升沿中断 3'b011：下降沿中断 3'b100：双沿中断
0 7:6	Px_INT_SEL1 Px_INT_SEL0	RW	0x00	GPIOx.2 的中断触发方式选择： 3'b000：高电平中断 3'b001：低电平中断 3'b010：上升沿中断 3'b011：下降沿中断 3'b100：双沿中断
5:3	Px_INT_SEL0	RW	0x00	GPIOx.1 的中断触发方式选择： 3'b000：高电平中断 3'b001：低电平中断 3'b010：上升沿中断 3'b011：下降沿中断 3'b100：双沿中断
2:0	Px_INT_SEL0	RW	0x00	GPIOx.0 的中断触发方式选择： 3'b000：高电平中断 3'b001：低电平中断 3'b010：上升沿中断 3'b011：下降沿中断 3'b100：双沿中断

注：{ Px_INT_SEL2，Px_INT_SEL1，Px_INT_SEL0 }，3 个 8bit 的寄存器按高低字节顺序拼接用于配置 GPIO 中断触发方式。

8.5.5. GPIOx 中断状态寄存器 Px_INT_REG(偏移：D4H、D9H、DEH、E3H、E8H)

比特	名称	属性	复位值	描述
7:0	Px_INT_REG	RW	0x00	GPIOx 中断状态寄存器 0：无中断 1：产生中断 硬件置 1，软件写 1 清 0 例如：清除 GPIO4.5 口的中断状态 P4_INT_REG=0x20；

8.6. GPIO 操作说明

8.6.1. 输入/输出 IO 配置

Copyright © 2022 武汉瑞纳捷半导体有限公司

1）配置 SCCM1 寄存器，使能 GPIO 的时钟；

2）配置 IOCFG_CTRLx 寄存器，将引脚配置为 GPIO 功能；

3）配置 Px_OEN/Px_IEN 寄存器，选择 GPIO 方向（输入/输出）；

4）设置 Px 的任意位来控制对应输出引脚的电平；

5） 读取 Px 来获取输入引脚电平。

8.6.2. 中断触发模式配置

1）配置 SCCM1 寄存器，使能 GPIO 的时钟；

2）配置 IOCFG_CTRLx 寄存器，将引脚配置为 GPIO 功能；

3）设置 Px_OEN/Px_IEN，将 GPIO 口设置为输入；

4）配置寄存器 Px_INT_SEL，选择边沿/电平触发类型；

5）配置寄存器 Px_INT_EN 使能相应位中断；

6）配置使能 GPIOx 中断和开总中断（EAL）。

8.6.3. 清除中断标志

中断处理函数通过写 Px_INT_REG 来清除中断状态。如果在清除寄存器的同时有新的边沿触发中断产生，这个新的中断将会保持有效直到下一次清除。

注：读取中断状态操作应该在对 Px_INT_EN 相应位清 0 之前进行，因为对 Px_INT_EN 相应位写 1 将清除相应中断状态

Copyright © 2022 武汉瑞纳捷半导体有限公司

9. 中断服务程序单元（ISR）

9.1. 概述

中断服务程序单元（ISR）是 8051 的一个重要组成部分。它与 CPU 处理器内核紧密耦合，实现低中断延迟以及对新到中断的有效处理，外部中断信号连接到 ISR，ISR 将对这些中断进行优先级排序。

每个中断源都有自己的请求标志，它位于中断源的设备中。没有中断请求标志是直接位于 ISR。所有中断请求是通过相应高电平输入到 ISR。通过相应的使能标志位（在“IEN0”、“IEN1”和“IEN2 ”“IEN3”“IEN4 ”SFR 寄存器中），每个中断源都可以单独启用或禁用。此外，所有的中断可以在全局范围内启用或禁用的“EAL”的标志位（在“IEN0“SFR 寄存器中）。

所有中断源被分成 6 个中断组。每个中断组可以有 1 到 4 个中断优先级分配。中断优先级的定义是由位于“IP0”和“IP1”SFR 寄存器的标志决定。

9.2. 中断服务单元特性

- 支持 30 路向量中断
- 4 级可编程中断优先级
- 支持可嵌套中断
- 中断可屏蔽

9.3. 中断源

表 9- 中断源

中断号	中断源	备注	使能位	中断入口地址
Int0	PVD	PVD 中断	IEN0.0	03H
Int1	TIMER0	定时器 0 中断	IEN0.1	0BH
Int2	GPIO1	GPIO1 模块中断	IEN0.2	13H
Int3	TIMER1	定时器 1 中断	IEN0.3	1BH
Int4	UART0	串口 0 模块中断	IEN0.4	23H
Int5	TIMER2	定时器 2 中断	IEN0.5	2BH
Int6	GPIO3	GPIO3 模块中断	IEN1.0	33H
Int7	GPIO2	GPIO2 模块中断	IEN1.1	3BH
Int8	SPI0	SPI0 模块中断	IEN1.2	43H
Int9	LPTIMER	LPTIMER 模块中断	IEN1.3	4BH
Int10	GPIO4	GPIO4 模块中断	IEN1.4	53H
Int11	SM4/DES/AES	SM4/DES/AES 加解密模块中断	IEN1.5	5BH
Int12	UART1	串口 1 模块中断	IEN2.0	63H

Copyright © 2022 武汉瑞纳捷半导体有限公司

Int13	Reserved	Reserved	IEN2.1	6BH
Int14	SPI1	SPI1 模块中断	IEN2.2	73H
Int15	SPI2	SPI2 模块中断	IEN2.3	7BH
Int16	SPI3	SPI3 模块中断	IEN2.4	83H
Int17	I2C0	I2C0 模块中断	IEN2.5	8BH
Int18	RTC	RTC 模块中断	IEN3.0	93H
Int19	WDT	看门狗模块中断	IEN3.1	9BH
Int20	GPIO0	GPIO0 模块中断	IEN3.2	A3H
Int21	ADC	ADC 模块中断	IEN3.3	ABH
Int22	I2C1	I2C1 模块中断	IEN3.4	B3H
Int23	LPUART	LPUART 模块中断	IEN3.5	BBH
Int24	TIMER3	定时器 3 中断	IEN4.0	C3H
Int25	TIMER4	定时器 4 中断	IEN4.1	CBH
Int26	UART2	串口 2 模块中断	IEN4.2	D3H
Int27	UART3	串口 3 模块中断	IEN4.3	DBH
Int28	COMP1	比较器 1 模块中断	IEN4.4	E3H
Int29	COMP2	比较器 2 模块中断	IEN4.5	EBH

9.4. 中断优先级

表9-2 中断优先级组

中断组	组间自然优先级	最高优先级	较高优先级	中间优先级	次低优先级	最低优先级
Group0	最高优先级	Int0	Int6	Int12	Int18	Int24
Group1	↑	Int1	Int7	Int13	Int19	Int25
Group2		Int2	Int8	Int14	Int20	Int26
Group3		Int3	Int9	Int15	Int21	Int27
Group4		Int4	Int10	Int16	Int22	Int28
Group5	最低优先级	Int5	Int11	Int17	Int23	Int29

表 9-3 可配置组间中断优先级

可配置优先级组	组间优先级	IP1.x bit	IP0.x bit
Level3	最高优先级	1	1
Level2	↑	1	0
Level1		0	1
Level0	最低优先级	0	0

注：1. 上表中 x 的值为 0, 1, 2, 3, 4, 5，分别代表 Group0，Group1，Group2，Group3，Group4，Group5；
例如，将 Group2 的中断优先级设为最高，则 IP1=0x04, IP0=0x04；

Copyright © 2022 武汉瑞纳捷半导体有限公司

2. 当请求多个中断时，所有优先级类型都考虑在内。最重要的是由“IP0”和“IP1”寄存器决定的优先级，然后组间自然优先，最后在每个组组内的优先级；

3. 优先级高的中断可以打断优先级低的中断，反之则不行，同一时间最多有 4 个中断服务。

9.5. 中断服务程序单元寄存器

ISR 特殊功能寄存器

偏移	名称	描述
0xA8	IEN0	中断使能寄存器 0
0xB8	IEN1	中断使能寄存器 1
0x9A	IEN2	中断使能寄存器 2
0xC9	IEN3	中断使能寄存器 3
0xD1	IEN4	中断使能寄存器 4
0xA9	IP0	中断优先级寄存器 0
0xB9	IP1	中断优先级寄存器 1

9.5.1. 中断使能寄存器 IEN0(偏移：A8H)

比特	名称	属性	复位值	描述
7	EAL	RW	0x0	总中断使能位 0：总中断不使能 1：开总中断（使用模块的中断都必须将这一位置 1）
6	WDTREL_EN	RW	0x0	看门狗计数值装载使能位 0：禁止 1：使能
5	ET2	RW	0x0	定时器 2 中断使能位 0：禁止 1：使能
4	ES0	RW	0x0	串行端口 0 中断使能位 0：禁止 1：使能
3	ET1	RW	0x0	定时器 1 溢出中断使能位 0：禁止 1：使能
2	EX1	RW	0x0	GPIO1 中断使能位 0：禁止 1：使能
1	ET0	RW	0x0	定时器 0 溢出使能位 0：禁止 1：使能

Copyright © 2022 武汉瑞纳捷半导体有限公司

0	EX0	RW	0x0	PVD 中断使能位 0：禁止 1：使能

9.5.2. 中断使能寄存器 IEN1(偏移：B8H)

比特	名称	属性	复位值	描述
7	EXEN2	RW	0x0	定时器 2 外部重载中断使能位（捕获模式 2 中断） 0：禁止 1：使能
6	SWDT	RW	0x0	看门狗定时器开始/刷新标志位
5	EX6	RW	0x0	SM4/DES/3DES/AES 中断使能位 0：禁止 1：使能
4	EX5	RW	0x0	GPIO4 中断使能位 0：禁止 1：使能
3	EX4	RW	0x0	LPTIMER 中断使能位 0：禁止 1：使能
2	EX3	RW	0x0	SPI0 中断使能位 0：禁止 1：使能
1	EX2	RW	0x0	GPIO2 中断使能位 0：禁止 1：使能
0	EX7	RW	0x0	GPIO3 中断使能位 0：禁止 1：使能

9.5.3. 中断使能寄存器 IEN2(偏移：9AH)

比特	名称	属性	复位值	描述
7：6	RSV	-	-	保留
5	I2C0_EN	RW	0x0	I2C0 中断使能位 0：禁止 1：使能
4	SPI3_EN	RW	0x0	SPI3 中断使能位 0：禁止 1：使能
3	SPI2_EN	RW	0x0	SPI2 中断使能位 0：禁止 1：使能

Copyright © 2022 武汉瑞纳捷半导体有限公司

2	SPI1_EN	RW	0x0	SPI1 中断使能位 0：禁止 1：使能
1	RSV			
0	ES1	RW	0x0	UART1 中断使能位 0：禁止 1：使能

9.5.4. 中断使能寄存器 IEN3(偏移：C9H)

比特	名称	属性	复位值	描述
7：6	RSV	-	-	保留
5	LPUART_EN	RW	0x0	LPUART 中断使能位 0：禁止 1：使能
4	I2C1_EN	RW	0x0	I2C1 中断使能位 0：禁止 1：使能
3	ADC_EN	RW	0x0	ADC 中断使能位 0：禁止 1：使能
2	GPIO0_EN	RW	0x0	GPIO0 中断使能位 0：禁止 1：使能
1	WDT_EN	RW	0x0	WDT 中断使能位 0：禁止 1：使能
0	RTC_EN	RW	0x0	RTC 中断使能位 0：禁止 1：使能

9.5.5. 中断使能寄存器 IEN4(偏移：D1H)

比特	名称	属性	复位值	描述
7：6	RSV	-	-	保留
5	COMP2_EN	RW	0x0	比较器 2 中断使能位 0：禁止 1：使能
4	COMP1_EN	RW	0x0	比较器 1 中断使能位 0：禁止 1：使能
3	UART3_EN	RW	0x0	串口 3 中断使能位 0：禁止

Copyright © 2022 武汉瑞纳捷半导体有限公司

				1：使能
2	UART2_EN	RW	0x0	串口 2 中断使能位 0：禁止 1：使能
1	TIMER4_EN	RW	0x0	定时器 4 中断使能位 0：禁止 1：使能
0	TIMER3_EN	RW	0x0	定时器 3 中断使能位 0：禁止 1：使能

9.5.6. 中断优先级寄存器 IP0(偏移：A9H)

比特	名称	属性	复位值	描述
7	RSV	-	-	保留
6	WDTS	RW	0x0	看门狗定时器状态标志位 当看门狗定时器复位发生，由硬件将此位置 1
5	IP0.5	RW	0x0	中断优先级配置 与 IP1 相应的位配合指定中断优先级组各自的优先级。 详见表 9-3 及其注释
4	IP0.4	RW	0x0	
3	IP0.3	RW	0x0	
2	IP0.2	RW	0x0	
1	IP0.1	RW	0x0	
0	IP0.0	RW	0x0	

9.5.7. 中断优先级寄存器 IP1(偏移：B9H)

比特	名称	属性	复位值	描述
7:6	RSV	-	-	保留
5	IP1.5	RW	0x0	中断优先级配置 与 IP0 相应的位配合指定中断优先级组各自的优先级。 详见表 9-3 及其注释
4	IP1.4	RW	0x0	
3	IP1.3	RW	0x0	
2	IP1.2	RW	0x0	
1	IP1.1	RW	0x0	
0	IP1.0	RW	0x0	

Copyright © 2022 武汉瑞纳捷半导体有限公司

10.模拟/数字转换器（ADC）

10.1. 概述

该设备内置了一个高速的 12 位逐次比较模拟数字转换器。提供 7 个多功能的输入通道（具体请参考 ADC 引脚描述说明）。A/D 转换的各个通道可以执行单次和有缓存功能的连续转换模式。ADC 模块的结构框图如图 10-1 所示。

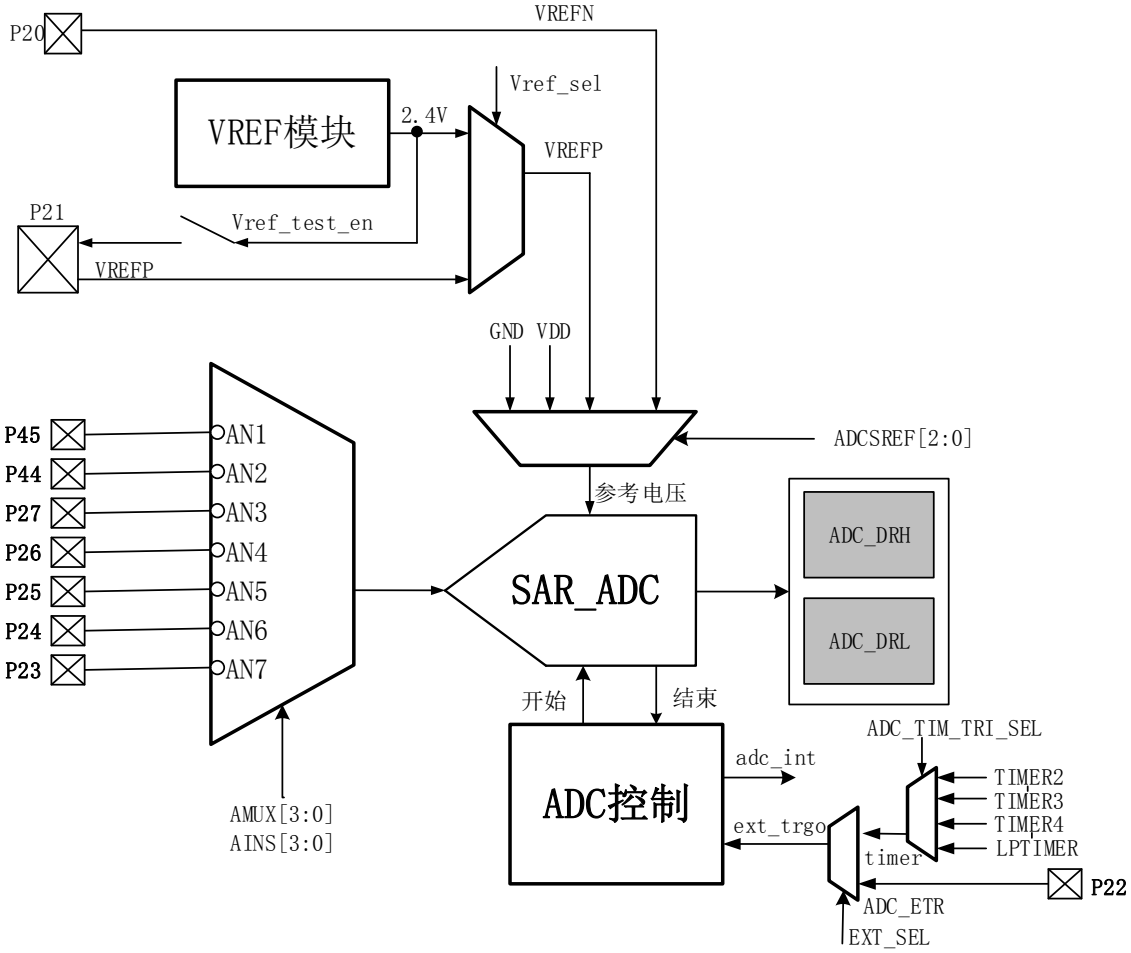


图 10-1 ADC 结构框图

10.2. ADC 主要特性

- 12 位分辨率，提供 7 个功能通道；
- 转换速率高达 1MSPS（16M 时钟下）；
- 单端输入和差分对输入方式；
- 单次转换和连续转换模式；
- 触发转换源：外部 IO 输入信号、TIMER2/3/4 和 LPTIMER 定时；
- 外部触发转换模式：上升沿触发、下降沿触发、双沿触发；
- VREF 模块，支持内部 2.4V 参考电压，也可输出至片外供其他器件使用；

Copyright © 2022 武汉瑞纳捷半导体有限公司

- 转换结束时可以产生中断；
- 内置校准单元。

注：

1. 当芯片 VDD 供电高于 2.4V 时，VREF 输出为 2.4V；当 VDD 供电低于 2.4V 时，VREF 输出则跟随 VDD 值，且较 VDD 小几十 mV。若 VREF 输出至片外供其他器件使用，驱动能力可能不足，需外挂跟随电路。

2. 使用 ADC 时需将管脚复用模拟功能或数字功能选择寄存器 AD_SEL1、AD_SEL2 对应位置位。

10.3. ADC 引脚描述

表 10-1 ADC 引脚

引脚名称	ADC引脚名称	信号类型	注释	
VDD	VDD	输入，模拟电源	模拟电源供电端，默认 ADC 参考电压高端	
VSS	GND	输入，模拟电源地	模拟电源地端，默认 ADC 参考电压低端	
P20	ADC_REFN	输入，模拟参考负极	ADC 使用的低端/负极参考电压	
P21	ADC_REFP	输入，模拟参考正极	ADC 使用的高端/正极参考电压	
P23	ADC_IN7	模拟输入信号 7	采集信号端 7	单端模式 ADC 采集时，每次只使用 1 个通道。 差分模式 ADC 采集时，通道 2 和 3、4 和 5、6 和 7 组成差分对。（具体参考 ADC 寄存器描述）
P24	ADC_IN6	模拟输入信号 6	采集信号端 6	
P25	ADC_IN5	模拟输入信号 5	采集信号端 5	
P26	ADC_IN4	模拟输入信号 4	采集信号端 4	
P27	ADC_IN3	模拟输入信号 3	采集信号端 3	
P44	ADC_IN2	模拟输入信号 2	采集信号端 2	
P45	ADC_IN1	模拟输入信号 1	采集信号端 1	

10.4. ADC 功能描述

10.4.1. ADC 开/关控制

通过寄存器 ADC_CR1 的 bit0 来开启 ADC。如果长时间不使用 ADC，可以通过清零这个位来关闭 ADC。

当 ADC 模块上电后，所选通道的 I/O 口输出模块是被禁止的。因此，在使用 ADC 模块时要选择合适的 ADC 转换通道。

10.4.2. 通道选择

ADC 有 7 个外部输入通道，实际外部通道的数量取决于 MCU 封装大小。单端时可通过配置 ADC_CSR 的 bit[3:0]选择模拟转换通道。差分时可通过配置 ADC_CR2 的 bit[6:3]选择差分对。

10.4.3. 转换模式

ADC 支持单次模式，带缓存的连续模式。模式的选择可通过 ADC_CR1 寄存器的 bit1 来选择转换模式。

Copyright © 2022 武汉瑞纳捷半导体有限公司

单次模式：在单次转换模式中，每次通过 ADC_CR1 的 bit4 开始单次转换（转换完成后自动清零）。当 ADC_STS 寄存器的 bit0 置 1（如果使能 ADC_MSK 寄存器的 bit0，则会产生 1 个中断），转换后的数据存储在 ADC_DRH 和 ADC_DRL 寄存器中。

带缓存的连续模式：在连续转换模式中，每次通过 ADC_CR1 的 bit2 开始连续转换（转换完成后自动清零）。每完成 1 次转换数据存在 FIFO 中，转化 8 次，当 ADC_STS 寄存器的 bit7 置 1，FIFO 被填满。ADC_STS 寄存器的 bit0 置 1（如果使能 ADC_MSK 寄存器的 bit0，则会产生 1 个中断），然后读取 FIFO 数据 8 次(即读 ADC_DRH 和 ADC_DRL 寄存器 8 次)。

注：当发生转换错误及数据溢出时，可通过 ADC_STS 寄存器了解相关信息，及通过 ADC_CR3 寄存器清除该错误标志（具体参考寄存器描述）。

10.4.4. ADC 校准及异常处理

ADC 自带校准功能，可通过 ADC_CR1 寄存器的 bit3 开启校准功能。通过 ADC_STS 寄存器的 bit3、bit5，了解校准是否有错误及是否完成。当发生校准错误时及完成时，通过 ADC_CR3 寄存器的 bit0、bit1 置位清除状态。

10.5. ADC 寄存器

ADC 寄存器基地址：0x2000

偏移	名称	描述
0x03	ADC_CSR	ADC 配置寄存器
0x04	ADC_CR1	ADC 配置寄存器 1
0x05	ADC_CR2	ADC 配置寄存器 2
0x06	ADC_CR3	ADC 配置寄存器 3
0x07	ADC_DRH	ADC 数据高位寄存器（低 4bit 有效）
0x08	ADC_DRL	ADC 数据低位寄存器
0x09	ADC_STS	ADC 状态寄存器
0x0A	ADC_MSK	ADC 中断屏蔽寄存器
0xCA	VREF_CTRL	VREF 模块控制寄存器

10.5.1. 配置寄存器 ADC_CSR(偏移：03H)

比特	名称	属性	复位值	描述
7	EXT_SEL	RW	0x0	外部信号触发 ADC 采样源头选择 1：芯片管脚（P22 管脚 ADC_ETR 功能）； 0：TIMER 计时触发；
6:4	ADCSREF[2:0]	RW	0x0	ADC 参考电压选择 000：VREF+ = VDD VREF- = GND 001：保留 010：VREF+ = VREFP VREF- = GND

Copyright © 2022 武汉瑞纳捷半导体有限公司

				011：VREF+ = VREFP VREF- = GND 100：VREF+ = VDD VREF- = VREFN 101：保留 110：VREF+ = VREFP VREF- = VREFN 111：VREF+ = VREFP VREF- = VREFN
3:0	AMUX[3:0]	RW	0x0	模拟转换通道选择 0000：保留 0001：ADC_IN1 0010：ADC_IN2 0011：ADC_IN3 0100：ADC_IN4 0101：ADC_IN5 0110：ADC_IN6 0111：ADC_IN7 1xxx：保留

10.5.2. 配置寄存器 ADC_CR1(偏移：04H)

比特	名称	属性	复位值	描述
7:6	EXT_TRIG_MODE	RW	0x0	外部信号触发 ADC 采样模式选择： 00：上升沿； 01：下降沿； 其他：双沿；
5	EXT_TRIG_EN	RW	0x0	外部信号触发 ADC 采样功能使能位： 1：使能； 0：不使能；
4	SING_CONV_EN	RW	0x0	单次转换使能位 0：禁止 1：使能 硬件清 0
3	CAL_EN	RW	0x0	校准使能位 0：禁止 1：使能 硬件清 0
2	CONT_CONV_EN	RW	0x0	连续转换使能位 0：禁止 1：使能 硬件清 0
1	CONV_MODE	RW	0x0	转换模式选择 0：单次模式

Copyright © 2022 武汉瑞纳捷半导体有限公司

				1：连续模式
0	ADC_EN	RW	0x0	ADC 使能位 0：禁止 1：使能

10.5.3. 配置寄存器 ADC_CR2(偏移：05H)

比特	名称	属性	复位值	描述
7	ADCCR	RW	0x0	ADC 转换速率 0：最大时钟频率为 16MHz 1：最大时钟频率为 4MHz 时钟频率≤4MHz 时，将该位置 1 可降低功耗
6	AINS[3]	RW	0x0	AIN[6],AIN[7]输入对配置 0：AIN[6],AIN[7]是独立的单端输入 1：AIN[6],AIN[7]是+和-的差分对
5	AINS[2]	RW	0x0	AIN[4],AIN[5] 输入对配置 0：AIN[4],AIN[5]是独立的单端输入 1：AIN[4],AIN[5]是+和-的差分对
4	AINS[1]	RW	0x0	AIN[2],AIN[3] 输入对配置 0：AIN[2],AIN[3]是独立的单端输入 1：AIN[2],AIN[3]是+和-的差分对
3	AINS[0]	RW	0x0	AIN[0],AIN[1] 输入对配置 0：AIN[1]是独立的单端输入 1：保留
2:1	ADCSHT	RW	0x0	ADC 采样保持时间选择 00：3.5 个 CLKADC 时钟周期 01：4.5 个 CLKADC 时钟周期 10：6.5 个 CLKADC 时钟周期 11：10.5 个 CLKADC 时钟周期
0	CONV_CAL	RW	0x0	带校准值转换选择位 0：不带校准值转换 1：带校准值转换

10.5.4. 配置寄存器 ADC_CR3(偏移：06H)

比特	名称	属性	复位值	描述
7:4	RSV	-	-	保留
3	CLR_FIFO	WO	0x0	FIFO 数据清除位 写 1 清空 FIFO 数据
2	CLR_CONV_ERR	WO	0x0	转换错误标志清除位 写 1 清 0
1	CLR_CAL_ERR	WO	0x0	校准错误标志清除位 写 1 清 0

Copyright © 2022 武汉瑞纳捷半导体有限公司

0	CLR_EOCAL	WO	0x0	校准完成标志清除位 写 1 清 0

10.5.5. 数据高位寄存器 ADC_DRH(偏移：07H)

比特	名称	属性	复位值	描述
7:4	RSV	-	-	保留
3:0	DH	RO	0x0	转换结果的高 4 位

10.5.6. 数据低位寄存器 ADC_DRL(偏移：08H)

比特	名称	属性	复位值	描述
7:0	DL	RO	0x00	转换结果的低 8 位 先读高 4 位，再读低 8 位

10.5.7. 状态寄存器 ADC_STS(偏移：09H)

比特	名称	属性	复位值	描述
7	FIFO_FULL	RO	0x0	FIFO 满标志位 0：非满 1：满
6	BUSY	RO	0x0	ADC 状态位 0：ADC 空闲 1：ADC 忙
5	EOCAL	RO	0x0	校准完成标志位 0：未完成 1：完成
4	CONV_ERR	RO	0x0	转换错误标志位 0：无错误 1：有错误
3	CAL_ERR	RO	0x0	校准错误标志位 0：无错误 1：有错误
2	EMPTY	RO	0x0	FIFO 空标志位 0：非空 1：空
1	OVR	RW	0x0	数据溢出标志位 0：无溢出 1：有溢出 软件写 1 清标志位。
0	EOC	RW	0x0	转换完成标志位 0：未完成 1：完成

Copyright © 2022 武汉瑞纳捷半导体有限公司

				软件写 1 清标志位。

10.5.8. 中断屏蔽寄存器 ADC_MSK(偏移：0AH)

比特	名称	属性	复位值	描述
7:4	RSV	-	-	保留
3	FIFO_HALF_FULL	RO	0x0	FIFO 半满标志位 0：未半满 1：半满
2	ADC_MSK2	RW	0x1	FIFO 半满中断屏蔽位 0：不屏蔽 1：屏蔽
1	ADC_MSK1	RW	0x1	FIFO 空中断屏蔽位 0：不屏蔽 1：屏蔽
0	ADC_MSK0	RW	0x1	转换完成中断屏蔽位 0：不屏蔽 1：屏蔽

10.5.9. VREF 模块控制寄存器 VREF_CTRL(偏移：CAH)

比特	名称	属性	复位值	描述
7	Vref_sel	RW	0x0	选择 ADC VREFP 来源： 1：来自管脚（P21）； 0：来自 VREF 模块输出 2.4V；
6	Vref_test_en	RW	0x0	VREF 模块 2.4V 参考电压输出到管脚使能： 1：使能； 0：不使能；
5	Vref_en	RW	0x0	VREF 模块使能： 1：使能； 0：不使能；
4:0	Vref_trim	RW	0x10	VREF 校准信号

10.6. ADC 操作说明

10.6.1. 单次模式配置

1) 配置 SCCM2 寄存器，使能 ADC 的时钟，同时失能模块软复位；
2) 将管脚模拟功能复用控制寄存器 ANA_ADC 的通道对应位置位；
3) 配置 ADC_CSR，选择参考电压以及转换通道；
4) 配置 ADC_CR1 的 CONV_MODE 位为 0，选择单次模式；
5) 配置 ADC_CR2，选择带校准值转换、设置采样保持时间、设置选定的转换通道模式、选择转换速率；

Copyright © 2022 武汉瑞纳捷半导体有限公司

6) 配置 ADC_MSK,选择是否使能中断；
7) 配置 ADC_CR3 的 CLR_FIFO 位，清空 FIFO；
8) 置位 ADC_CR1 的 ADC_EN 位，启动 ADC；
9) 配置 ADC_CR1 的 CAL_EN 位，启动软件校准使能；
10) 查询 ADC_STS 的 EOCAL 位，等待校准完成；
11) 置位 ADC_CR1 的 CONV_EN 位，启动转换；
12) 查询方式：软件查询 ADC_STS 的 EOC 位，判断转换完成，清除状态，读取转换结果；
13) 中断方式：转换完成后产生中断，中断服务函数内判断转换完成，清除 ADC_STS 的 EOC 位，读取转换结果；
14) 异常处理：查询状态，判断异常后，根据异常结果做如下处理：
15) 清除校准异常；清除转换完成异常；清除溢出标志；清除 fifo。

10.6.2. 连续模式配置

1) 配置 SCCM2 寄存器，使能 ADC 的时钟，同时失能模块软复位；
2) 将管脚模拟功能复用控制寄存器 ANA_ADC 的通道对应位置位；
3) 配置 ADC_CSR，选择参考电压以及转换通道；
4) 配置 ADC_CR1 的 CONV_MODE 位为 1，选择连续模式；
5) 配置 ADC_CR2，选择带校准值转换、设置采样保持时间、设置选定的转换通道模式、选择转换速率；
6) 配置 ADC_MSK,选择是否使能中断；
7) 配置 ADC_CR3 的 CLR_FIFO 位，清空 FIFO；
8) 置位 ADC_CR1 的 ADC_EN 位，启动 ADC；
9) 配置 ADC_CR1 的 CAL_EN 位，启动软件校准使能；
10) 查询 ADC_STS 的 EOCAL 位，等待校准完成；
11) 置位 ADC_CR1 的 CONV_EN 位，启动转换；
12) 每完成一次转换，将转换数据存在 FIFO 中，连续转换 8 次 FIFO 被填满，ADC_STS 的 EOC 为硬件置位
13) 查询模式：软件查询 ADC_STS 的 EOC 位，判断转换完成，读取 fifo 数据 8 次，清除状态；
14) 中断模式：进入中断后，查询 ADC_STS 的 EOC 位，判断转换完成，读取 fifo 数据 8 次，清除状态；
15) 异常处理，查询状态判断异常后，根据异常结果做如下处理：
16) 清除校准异常；清除转换完成异常；清除溢出标志；清除 fifo。

10.6.3. 采样触发配置

1) 若配置 ADC_CSR 的 EXT_SEL=1 选择管脚触发，则需将 P04 端口复用为

Copyright © 2022 武汉瑞纳捷半导体有限公司

ADC_ETR 外部触发功能；若配置 ADC_CSR 的 EXT_SEL=0 则选计数器触发；

2) 配置 ADC_CR1 的 EXT_TRIG_MODE 位，选择外部触发转换模式；
3) 配置 ADC_CR1 的 EXT_TRIG_EN 位，使能外部触发 ADC 转换；

Copyright © 2022 武汉瑞纳捷半导体有限公司

11.比较器（COMP）

11.1. 概述

该设备内置了 2 路比较器 COMP1 和 COMP2，它们共享相同的偏移电流。

11.2. COMP 主要特性

- COMP1:反相输入端是固定阈值，为内部参考电压（电压值为 1V）；正相输入端为外部 IO 输入
- COMP2:反相输入端阈值可选，可为内部参考电压或外部 IO 输入参考电压；正相输入端为外部 IO 输入
- 两路比较器都可产生中断
- 两路比较器输出极性可控
- COMP2 的反相输入端阈值可选，内部参考电压有 4 个档位
- COMP2 的输出可重定向为 TIMER4 的刹车输入或者 TIMER3 的重载模式 1 下的外部重载信号和 TIMER2 的外部重载信号
- COMP2 的工作模式可选：快速模式下，比较器翻转速度快，功耗高
  低速模式下，比较器翻转速度慢，功耗低

11.3. COMP 引脚描述

表 11-1 COMP 引脚

引脚名称	ADC引脚名称	信号类型	注释
P30	COMP2_IN_N	输入，模拟反相输入端	比较器 2 外部反相输入通道
P31	COMP1_IN_P	输入，模拟正相输入端	比较器 1 外部正相输入通道
P32	COMP2_IN_P	输入，模拟正相输入端	比较器 2 外部正相输入通道

11.4. 比较器寄存器

COMP 寄存器基地址：0x2000

偏移	名称	描述
0x1E	COMP_CTRL0	控制寄存器 0
0x1F	COMP_CTRL1	控制寄存器 1
0xFC	IOCFG_CTRL6	状态/控制寄存器

11.4.1. 控制寄存器 COMP_CTRL0(偏移：1EH)

比特	名称	属性	复位值	描述
7	MC	RW	0x0	比较器 2 的工作模式控制位 0：低速模式

Copyright © 2022 武汉瑞纳捷半导体有限公司

				1：快速模式
6:4	CM[2:0]	RW	0x0	比较器 2 的反相参考电压选择位： 000：1.0V 001：0.9V 010：0.6V 011：0.3V 100：IO 输入电压
3	ENBUFFER	RW	0x0	缓冲器模块使能位 0：禁止 1：使能
2	ENBIAS	RW	0x0	偏移电流模块使能位 0：禁止 1：使能
1	ENCOMP2	RW	0x0	比较器 2 的使能位 0：禁止 1：使能
0	ENCOMP1	RW	0x0	比较器 1 的使能位 0：禁止 1：使能

11.4.2. 控制寄存器 COMP_CTRL1(偏移：1FH)

比特	名称	属性	复位值	描述
7	COMP_POL1	RW	0x0	比较器 1 的输出极性选择 0：输出取反 1：输出不变
6:5	OUTSEL[1:0]	RW	0x0	比较器 2 输出选择 00：保留 01：比较器 2 的输出接 TIMER4 的刹车输入 10：比较器 2 的输出接 TIMER3 的重载模式 1 下的外部重载信号 11：比较器 2 的输出接 TIMER2 的外部重载信号
4	COMP_POL2	RW	0x0	比较器 2 的输出极性选择 0：输出取反 1：输出不变
3:0	-	-	-	保留

11.4.3. 状态/控制寄存器 IOCFG_CTRL6(偏移：FCH)

比特	名称	属性	复位值	描述
7	COMP_MSK[1]	RW	0x1	比较器 2 中断屏蔽使能位 0：不屏蔽 1：屏蔽
6	COMP_MSK[0]	RW	0x1	比较器 1 中断屏蔽使能位 0：不屏蔽

Copyright © 2022 武汉瑞纳捷半导体有限公司

				1：屏蔽
5	COMP_STS[1]	RW	0x0	比较器 2 事件标志 输出端发生事件，该位置 1，通过软件写 0 清。可根据屏蔽位选择是否产生中断。 0：未检测到事件 1：检测到事件
4	COMP_STS[0]	RW	0x0	比较器 1 事件标志 输出端发生事件，该位置 1，通过软件写 0 清。可根据屏蔽位选择是否产生中断。 0：未检测到事件 1：检测到事件
3:0	-	-	-	低 4 位不是比较器的功能位，不在此描述

11.5. 比较器操作说明

11.5.1. COMP1 配置

1) 配置 AD_SEL2 的 CMP_IO_EN[1]位，将 P31 管脚复用为 COMP1_IN_P；
2) 配置寄存器 COMP_CTRL0 的 ENBIAS 位，使能偏移电流；
3) 配置寄存器 COMP_CTRL1 的 COMP_POL1 位设置输出极性；
4) 配置寄存器 IOCFG_CTRL6 的 COMP_MSK[0]位，设置是否产生中断；
5) 配置 COMP_CTRL0 的 ENCOMP1 位，使能比较器 1。

11.5.2. COMP2 配置

1) 配置 AD_SEL2 的 CMP_IO_EN[2] 位，将 P32 管脚复用为 COMP2_IN_P；
2) 若反相参考电压选择为外部 IO 输入，则需配置 AD_SEL2 的 CMP_IO_EN[0]位，将 P30 管脚复用为 COMP2_IN_N；
3) 配置寄存器 COMP_CTRL0 的 MC、CM、ENBIAS 和 ENBUFFER 位，设置工作模式，缓冲器和偏移电流使能，选择反相参考电压；
4) 配置 COMP_CTRL1 的 OUTSEL、COMP_POL2、EDGE_SEL2 位，设置输出选择、输出极性；
5) 配置寄存器 IOCFG_CTRL6 的 COMP_MSK[1]位，设置是否产生中断；
6) 配置 COMP_CTRL0 的 ENCOMP2 位，使能比较器

Copyright © 2022 武汉瑞纳捷半导体有限公司

12.基本定时器（TIMER0/1）

12.1. 概述

芯片有 2 个独立的、向上计数的基本定时器：TIMER0 和 TIMER1。每个定时器都可以用来作为时基发生器，具有溢出中断功能。

12.2. 基本定时器主要特性

- 最大 16 位递增定时器；
- 计数分频固定为系统时钟的 12 分频；
- TIMER0 有 4 种工作模式：13 位定时器、16 位定时器、8 位自动重载定时器、拆成两个 8 位定时器；
- TIMER1 有 3 种工作模式：13 位定时器、16 位定时器、8 位自动重载定时器；
- 从初值计数至溢出时产生中断。

12.3. 基本定时器工作模式

基本定时器工作模式是可配置的。通过 TMOD 寄存器 bit[1:0]来设置定时器 0 的工作模式，通过 TMOD 寄存器 bit[5:4]来设置定时器 1 的工作模式。

表 12-1 基本定时器工作模式（表中 x=0 或 1）

模式	M1	M0	功能
模式 0	0	0	13 位定时，TLx 低 5 位和 THx 构成 13 位定时器
模式 1	0	1	16 位定时，TLx 和 THx 构成 16 位定时器
模式 2	1	0	8 位重载定时，溢出后将 THx 的值重载到 TLx 重新计数
模式 3	1	1	TIMER0 拆成两个 8 位定时器(会借用 TIMER1 的控制位、标志位和中断线）

定时是指当接入周期变化的计数脉冲信号，定时器从某个初值 x（可通过 TL0 和 TH0 寄存器进行配置）开始加 1 计数，到计满回零的瞬间产生溢出中断，表示定时时间到。定时时间计算如下：

$$t = (M - x) * T$$

式中，t 为定时时间；

M 为定时器的模；

x 为计数初值；

T 为计数脉冲的周期。

模 M 是指定时器从 0 开始计数到溢出时的最大计数值，即$M = 2^n$。

例如，8 位定时器 $M = 2^8 = 256$

Copyright © 2022 武汉瑞纳捷半导体有限公司

13 位定时器 $M = 2^{13} = 8192$

16 位定时器 $M = 2^{16} = 65536$

计数脉冲周期： $T = \frac{1}{fosc} * 12$ （f_{osc}为系统时钟频率）。

12.4. 基本定时器寄存器

TIMER0/1 特殊功能寄存器

偏移	名称	描述
0x88	TCON	定时器控制寄存器
0x89	TMOD	定时器模式寄存器
0x8A	TL0	TIMER0 定时寄存器低字节
0x8B	TL1	TIMER1 定时寄存器低字节
0x8C	TH0	TIMER0 定时寄存器高字节
0x8D	TH1	TIMER1 定时寄存器高字节

12.4.1. 定时器控制寄存器 TCON(偏移：88H)

比特	名称	属性	复位值	描述
7	TF1	RW	0x0	定时器 1 溢出标志位 计数溢出时硬件置 1，软件写 1 清 0
6	TR1	RW	0x0	定时器 1 运行控制位 0：禁止 1：运行
5	TF0	RW	0x0	定时器 0 溢出控制位 计数溢出时硬件置 1，软件写 1 清 0
4	TR0	RW	0x0	定时器 0 运行控制位 0：禁止 1：运行
3：0	RSV	-	-	保留

12.4.2. 定时器模式寄存器 TMOD(偏移：89H)

比特	名称	属性	复位值	描述
7:6	RSV	-	-	保留
5	TIMER1_M1	RW	0x0	TIMER1 工作模式选择位 详见“12.3 基本定时器工作模式”章节
4	TIMER1_M0	RW	0x0	
3:2	RSV	-	-	保留
1	TIMER0_M1	RW	0x0	TIMER0 工作模式选择位 详见“12.3 基本定时器工作模式”章节
0	TIMER0_M0	RW	0x0	

12.4.3. TIMER0 定时寄存器低字节 TL0(偏移：8AH)

比特	名称	属性	复位值	描述

Copyright © 2022 武汉瑞纳捷半导体有限公司

7:0	TL0	RW	0x00	TIMER0 计数寄存器低字节

12.4.4. TIMER1 定时寄存器低字节 TL1(偏移：8BH)

比特	名称	属性	复位值	描述
7:0	TL1	RW	0x00	TIMER1 计数寄存器低字节

12.4.5. TIMER0 定时寄存器高字节 TH0(偏移：8CH)

比特	名称	属性	复位值	描述
7:0	TH0	RW	0x00	TIMER0 计数寄存器高字节

12.4.6. TIMER1 定时寄存器高字节 TH1(偏移：8DH)

比特	名称	属性	复位值	描述
7:0	TH1	RW	0x00	TIMER1 计数寄存器高字节

12.5. 基本定时器操作说明

12.5.1. TIMER0 的操作

1) 配置 SCCM1 寄存器，使能 TIMER0 的时钟；
2) 配置 TMOD[1:0]，设置定时器工作模式；
3) 设置 TH0，TL0 初值；
4) 清除状态标记 TCON 的 TF0 位；
5) 设置 TCON 的 TR0 位，启动 TIMER0；
6) 使能 TIEMR0 中断和总中断；
7) 溢出后，TCON 的 TF0 被置位，触发中断，需要软件清除 TF0 标志位。

12.5.2. TIMER1 的操作

1) 配置 SCCM1 寄存器，使能 TIMER1 的时钟；
2) 配置 TMOD[5:4]，设置定时器工作模式（不支持模式 3）；
3) 设置 TH1，TL1 初值；
4) 清除状态标记 TCON 的 TF1 位；
5) 设置 TCON 的 TR1 位，启动 TIMER1；
6) 使能 TIEMR1 中断和总中断；
7) 溢出后，TCON 的 TF1 被置位，触发中断，需要软件清除 TF1 标志位。

Copyright © 2022 武汉瑞纳捷半导体有限公司

13.通用定时器（TIMER2）

13.1. 概述

芯片有 1 个带有可编程分频器、向上计数的 16 位通用定时器：TIMER2。适用于多个场合，包括基本的定时，测量输入信号脉冲长度，产生输出波形（输出比较，PWM）。

13.2. 通用定时器主要特性

- 16 位向上计数和自动重载计数器
- 2 种工作模式：
  - — 基本定时模式
  - — 门控定时模式
- 可更改的计数分频（系统时钟的 12 分频/24 分频）
- 3 个独立通道
  - — 输出比较
  - — PWM 生成（无互补输出）
- 如下事件发生时可产生中断
  - — 从初值向上计数至溢出
  - — 外部重载
  - — 输出比较

13.3. 分频器

该通用定时器，计数时钟分频可以通过寄存器 T2CON 的 bit7 来选择，即 T2CON[7]=0 为 12 分频，T2CON[7]=1 为 24 分频。

13.4. 定时器模式

TIMER2 定时器模式，可通过配置寄存器 T2CON 的 bit[1:0]来选择该模式，即 T2CON[1]=0，T2CON[0]=1。

13.5. 门控定时模式

TIMER2 门控定时模式，可通过配置寄存器 T2CON 的 bit[1:0]来选择该模式，即 T2CON[1]=1，T2CON[0]=1。该模式下当外部信号为高时，则计数增加。外部信号为低时，则计数增加停止。（TIMER2 是 TIM2_ETR）

13.6. 定时器重载功能

TIMER2 重载功能有 2 种模式，可通过配置寄存器 T2CON 的 bit[4:3]来选择重载

Copyright © 2022 武汉瑞纳捷半导体有限公司

模式，即 T2CON[4]=1，T2CON[3]=0 选择模式 0；T2CON[4]=1，T2CON[3]=1 选择模式 1。

重载模式 0：计数溢出后产生重载信号（自动重载）。

重载模式 1：输入引脚信号有下降沿，则产生重载信号。（TIMER2 是 TIM2_ERT）

重载功能下，重载值为寄存器 CRCH,CRCL 的值。

13.7. 定时器比较输出功能

13.7.1. 比较模式

TIMER2 比较输出功能有 2 种模式，可通过配置寄存器 T2CON 的 bit2 来选择比较模式，即 T2CON[2]=0 选择模式 0；T2CON[2]=1 选择模式 1。

在 2 种比较模式中有 3 个比较输出通道：即 CRC,CC1,CC2。3 个通道下对应 3 个端口和 3 个比较器，可通过寄存器 CCEN 的 bit[1:0]选择 CRC；CCEN 的 bit[3:2]选择 CC1；CCEN 的 bit[5:4]选择 CC2（具体配置参考寄存器描述）。

设定比较值时：CRC 通道对应寄存器 CRCH,CRCL ；CC1 通道对应寄存器 CCH1,CCL1 ；CC2 通道对应寄存器 CCH2, CCL2。

表 13-1 TIMER2 比较通道对应输出引脚

引脚名称	输出引脚名称	信号类型	注释(在使用时注意端口复用)
P01	TIM2_CC0	输出	与通道 CRC 相应的比较输出引脚
P02	TIM2_CC1	输出	与通道 CC1 相应的比较输出引脚
P03	TIM2_CC2	输出	与通道 CC2 相应的比较输出引脚

比较模式 0：当 TIMER2 的计数值等于比较设定值时，相应比较器通道的比较输出信号变化。具体可由寄存器 CCL3[i](i=0/1/2)控制，即

当 CCL3[i](i=0/1/2)=0 时，相应通道比较输出由低变高，

当 CCL3[i](i=0/1/2)=1 时，相应通道比较输出由高变低；

TIMER2 计数值溢出时，相应比较器通道的输出信号变为与比较器的值相等时输出信号相反的极性。

注：i=0 对应 CRC 通道；i=1 对应 CC1 通道；i=2 对应 CC2 通道

比较模式 1：当 TIMER2 的计数值等于比较器的值时，相应比较器通道的比较输出信号变化（具体由寄存器 P1 的第 0,1,2 位控制，即当计数值等于设定的比较值时，输出信号被激活，输出引脚输出 P1[i](i=0/1/2)的数值(需要使能 P1 口的时钟)；TIMER2 计数值溢出时，相应比较器通道的输出信号不会变化。

13.7.2. PWM 模式

脉冲宽度调制基于定时器的比较输出功能，可以通过设定比较值来改变占空比。

Copyright © 2022 武汉瑞纳捷半导体有限公司

在 T2CON[2]=0 选择比较模式 0，来选择 PWM 模式，能够独立地设定每个输出通道产生 PWM，其通过 CCEN[i][j]=10(i=1,3,5；j=0,2,4)来选择 1 路 PWM 也可选择多路。

PWM 的极性可以通过软件在寄存器 CCL3[i](i=0/1/2)中设置，其计数值和比较值始终进行比较用以确定是否触发比较输出信号的产生。

通过设置通道的比较值来产生不同占空比的 PWM。（CRC 通道对应寄存器 CRCH,CRCL；CC1 通道对应寄存器 CCH1 ,CCL1；CC2 通道对应寄存器 CCH2, CCL2）。

13.8. 定时器 2 中断

中断的产生需使能相应的中断源。

如下事件发生时可产生中断：

— 更新中断：计数向上溢出

— 外部重载

— 输出比较中断

13.9. 定时器 2 寄存器

TIMER2 特殊功能寄存器

偏移	名称	描述
0xC8	T2CON	TIMER2 控制寄存器
0xC1	CCEN	TIMER2 比较模式选择寄存器
0xC6	CCL3	TIMER2 比较输出极性选择寄存器
0xC2	CCL1	TIMER2 比较寄存器低字节
0xC3	CCH1	TIMER2 比较寄存器高字节
0xC4	CCL2	TIMER2 比较寄存器低字节
0xC5	CCH2	TIMER2 比较寄存器高字节
0xCA	CRCL	TIMER2 比较寄存器低字节
0xCB	CRCH	TIMER2 比较寄存器高字节
0xCC	TL2	TIMER2 计数寄存器低字节
0xCD	TH2	TIMER2 计数寄存器高字节
0xC0	IRCON	TIMER2 中断标志状态寄存器

13.9.1. TIMER2 控制寄存器 T2CON(偏移：C8H)

比特	名称	属性	复位值	描述
7	T2PS	RW	0x0	预分频选择 0：系统时钟的 12 分频 1：系统时钟的 24 分频
6：5	RSV	-	-	保留

Copyright © 2022 武汉瑞纳捷半导体有限公司

4	T2R1	RW	0x0	重载模式选择 0x：重载禁止 10：模式 0 11：模式 1
3	T2R0	RW	0x0	
2	T2CM	RW	0x0	比较模式选择 0：模式 0 1：模式 1
1	T2I1	RW	0x0	输入选择 00：TIMER2 停止 01：基本定时模式 10：RSV 11：门控定时模式
0	T2I0	RW	0x0	

13.9.2. TIMER2 比较模式选择寄存器 CCEN(偏移：C1H)

比特	名称	属性	复位值	描述		
7:6	-	-	-	保留		
5:4	COCAH2 COCAL2	RW	0x0	比较模式下，CC2 通道		
				COCAH2	COCAL2	描述
				0	0	比较禁止
				0	1	-
				1	0	通道比较输出使能
				1	1	-
3:2	COCAH1 COCAL1	RW	0x0	比较模式下，CC1 通道		
				COCAH1	COCAL1	描述
				0	0	比较禁止
				0	1	-
				1	0	通道比较输出使能
				1	1	-
1:0	COCAH0 COCAL0	RW	0x0	比较模式下，CRC 通道		
				COCAH0	COCAL0	描述
				0	0	比较禁止
				0	1	-
				1	0	通道比较输出使能
				1	1	-

13.9.3. TIMER2 比较输出极性选择寄存器 CCL3(偏移：C6H)

比特	名称	属性	复位值	描述
7	-	-	-	保留
2	CCL3[2]	RW	0x0	通道 CC2 的输出极性选择 0：计数值等于比较值时，信号由低变高 1：计数值等于比较值时，信号由高变低

Copyright © 2022 武汉瑞纳捷半导体有限公司

1	CCL3[1]	RW	0x0	通道 CC1 的输出极性选择 0：计数值等于比较值时，信号由低变高 1：计数值等于比较值时，信号由高变低
0	CCL3[0]	RW	0x0	通道 CRC 的输出极性选择 0：计数值等于比较值时，信号由低变高 1：计数值等于比较值时，信号由高变低

13.9.4. TIMER2 比较寄存器低字节 CCL1(偏移：C2H)

比特	名称	属性	复位值	描述
7:0	CCL1	RW	0x00	TIMER2 的通道 CC1 比较寄存器低字节

13.9.5. TIMER2 比较寄存器高字节 CCH1(偏移：C3H)

比特	名称	属性	复位值	描述
7:0	CCH1	RW	0x00	TIMER2 的通道 CC1 比较寄存器高字节

13.9.6. TIMER2 比较寄存器低字节 CCL2(偏移：C4H)

比特	名称	属性	复位值	描述
7:0	CCL2	RW	0x00	TIMER2 的通道 CC2 比较寄存器低字节

13.9.7. TIMER2 比较寄存器高字节 CCH2(偏移：C5H)

比特	名称	属性	复位值	描述
7:0	CCH2	RW	0x00	TIMER2 的通道 CC2 比较寄存器高字节

13.9.8. TIMER2 比较寄存器低字节 CRCL(偏移：CAH)

比特	名称	属性	复位值	描述
7:0	CRCL	RW	0x00	TIMER2 的通道 CRC 比较寄存器高字节

13.9.9. TIMER2 比较寄存器高字节 CRCH(偏移：CBH)

比特	名称	属性	复位值	描述
7:0	CRCH	RW	0x00	TIMER2 的通道 CRC 比较寄存器高字节

13.9.10. TIMER2 计数寄存器低字节 TL2(偏移：CCH)

比特	名称	属性	复位值	描述
7:0	TL2	RW	0x00	TIMER2 计数初值设置低字节

13.9.11. TIMER2 计数寄存器高字节 TH2(偏移：0DH)

比特	名称	属性	复位值	描述
7:0	TH2	RW	0x00	TIMER2 计数初值设置高字节

13.9.12. TIMER2 中断标志状态寄存器 IRCON(偏移：C0H)

比特	名称	属性	复位值	描述
7	EXF2	RW	0x0	TIMER2 外部重载标志位 外部重载发生时硬件置 1，软件写 1 清 0

Copyright © 2022 武汉瑞纳捷半导体有限公司

6	TF2	RW	0x0	TIMER2 溢出标志位 0：未溢出 1：溢出 硬件置 1，软件写 1 清 0
5：0	RSV	-	-	保留

13.10. 定时器 2 操作说明

13.10.1. 定时

a. 基本定时配置

1) 配置 SCCM1 寄存器，使能 TIMER2 的时钟；
2) 配置 T2CON 寄存器，设置 TIMER2 为基本定时模式，禁止重载功能，设置预分频数；
3) 配置 TH2，TL2 寄存器，设置定时器初值；
4) 使能 TIEMR2 中断和总中断；
5) 溢出后，IRCON 的 TF2 位被置位，并触发中断，需要软件清除 TF2 标志位。

b. 重载基本定时配置

1) 配置 SCCM1 寄存器，使能 TIMER2 的时钟；
2) 配置 T2CON 寄存器，设置 TIMER2 为基本定时模式，设置预分频数，使能重载模式，若为重载模式 1，则还需将 P00 端口复用为 TIM2_ERT；
3) 配置 TH2，TL2 寄存器，设置定时器初值；
4) 使能 TIEMR2 中断和总中断；
5) 溢出后，IRCON 的 TF2 位被置位，并触发中断，需要软件清除 TF2 标志位。

c. 门控定时配置

1) 将 P30 端口复用为 TIM2_ETR；
2) 配置 SCCM1 寄存器，使能 TIMER2 的时钟；
3) 配置 T2CON 寄存器，设置 TIMER2 工作模式为门控定时器模式，禁止重载，设置预分频数；
4) 配置 TH2，TL2 寄存器，设置定时器初值；
5) 使能 TIEMR2 中断和总中断；
6) 溢出后，IRCON 的 TF2 位被置位，并触发中断，需要软件清除 TF2 标志位

13.10.2. 比较

a. 比较模式 0

1) 配置 SCCM1 寄存器，使能 TIMER2 的时钟；
2) 配置 T2CON 寄存器，设置 TIMER2 工作模式为定时器模式，禁止重载功能，预分频数 12，设置为比较模式 0；

Copyright © 2022 武汉瑞纳捷半导体有限公司

3) 配置 IOCFG_CTRLx 寄存器，将相应管脚配置为比较/互补输出功能；
4) 设置计数周期 TH2，TL2；
5) 配置 CRC 通道比较值 CRCL，CRCH 寄存器；
6) 配置 CC1 通道比较值 CCL1，CCH1 寄存器；
7) 配置 CC2 通道比较值 CCL2，CCH2 寄存器；
8) 写 CCL3 寄存器，设置输出极性；
9) 写 CCEN 寄存器，使能比较模式；
10) 使能 TIMER2 中断和总中断；
11) 当 TIMER2 的计数值与设定的比较值相等时，对应通道输出设定的电平；
12) 溢出后，IRCON 的 TF2 位被置位，并触发溢出中断，需要软件清除 TF2 标志位并重装计数周期 TH2，TL2。

b. 比较模式 1

1) 配置 SCCM1 寄存器，使能 TIMER2 的时钟；
2) 配置 T2CON 寄存器，设置 TIMER2 工作模式为定时器模式，禁止重载功能，预分频数 12，设置为比较模式 1；
3) 配置 IOCFG_CTRLx 寄存器，将相应管脚配置为比较/互补输出功能；
4) 设置计数周期 TH2，TL2；
5) 配置 CRC 通道比较值 CRCL，CRCH 寄存器；
6) 配置 CC1 通道比较值 CCL1，CCH1 寄存器；
7) 配置 CC2 通道比较值 CCL2，CCH2 寄存器；
8) 配置 CCEN 寄存器，使能比较模式；
9) 使能 TIMER2 中断和总中断；
10) 当 TIMER2 的计数值与设定的比较值相等时，对应的比较器输出为 P1 口 P10、P11、P12 的电平值；
11) 溢出后，IRCON 的 TF2 位被置位，并触发溢出中断，需要软件清除 TF2 标志位并重装计数周期 TH2，TL2，但比较的输出不受影响。

Copyright © 2022 武汉瑞纳捷半导体有限公司

14.高级定时器（TIMER3/4）

14.1. 概述

芯片有 2 个 16 位高级定时器：TIMER3、TIMER4。适用于多个用途，包括基本的定时、测量输入信号脉冲宽度、产生输出波形（输出比较，PWM），以及对应不同事件（溢出、捕获、比较、刹车）的中断。可对刹车的输入信号进行滤波。

14.2. 主要特性

- 16 位的向上、向下和向上/向下交替计数自动重载计数器
- 多达 3 个独立通道，可以配置成：
  - — 输入捕获
  - — 输出比较
  - — PWM 生成（边沿对齐和中央对齐）
- 支持互补输出，并且死区时间可编程
- 刹车输入信号可以将定时器输出信号置于已知状态
- 刹车输入信号可进行数字滤波
- 产生中断的事件包括：
  - — 溢出：从 0 向上计数至重载值溢出、从重载值向下计数至 0 溢出
  - — 外部重载
  - — 捕获输入
  - — 输出比较
  - — 刹车信号输入

注：定时器比较输出周期内的高、低电平存在一个计数值的时间差

14.3. 功能描述

TIMER3 和 TIMER4 功能相同，下面以 TIMER3 为例进行说明。

14.3.1. 定时器模式

通过设置“TIM_T2CON”寄存器的标志位“T2I0”=1 和“T2I1”=0 来选定该模式。在该模式下，计数脉冲来源于输入的“CLK”。

TIM3 的增加，是基于预分频配置器的值。预分频器的配置值是由“TIM_PSCL”和“TIM_PSCH”寄存器来选定的，可配置成 1~65536 分频，每 1~65536 个时钟周期，定时器加 1。

14.3.2. 门控定时模式

Copyright © 2022 武汉瑞纳捷半导体有限公司

通过设置“TIM_T2CON”寄存器的标志位“T2I0”=1 和“T2I1”=1 来选定该模式。

在该模式下，TIM3 每 1~65536 个时钟周期增加，但是 TIM3 是由外部信号“TIM3_ETR”控制的。当“TIM3_ETR”=0 时，TIM3 计数保持；当“TIM3_ETR”=1 时，TIM3 计数增加。（TIMER3 是 TIM3_ETR，TIMER4 是 TIM4_ETR）

14.3.3. 重载模式

通过设置“TIM_T2CON”寄存器的标志位“T2R0”和“T2R1”来选择重载模式。

在下面两种情况下，16 位的重载将被执行，TIM3 重载值为“TIMER_TH2，TIMER_TL2”寄存器的值。

1）重载模式 0：TIM3 的溢出产生重载信号(自动重载)。

2）重载模式 1：输入引脚“TIM3_ERT”的下降沿、上升沿或者双沿（沿事件通过配置“TIM_CCEN”寄存器的标志位“EXF2_SEL”来选择）产生重载信号（TIMER3 是 TIM3_ERT，TIMER4 是 TIM4_ERT）。

14.3.4. 比较模式

比较单元包含 3 个寄存器：“CC1”、“CC2”、“CRC”。在比较模式中，每一个寄存器都可以配置。寄存器的值会与 TIMER 的计数值作比较。

“TIM3_CC0、TIM3_CC0N”是与“CRC”相对应的比较器的互补输出管脚；

“TIM3_CC1、TIM3_CC1N”是与“CC1”相对应的比较器的互补输出管脚；

“TIM3_CC2、TIM3_CC2N”是与“CC2”相对应的比较器的互补输出管脚。

有两种比较模式，可以通过“TIM_T2CON”寄存器中“T2CM”来选择。

在计数值等于比较寄存器的值时，可产生中断。

表 14-1 TIMER3 和 TIMER4 比较通道对应输出引脚

引脚名称	输出引脚名称	信号类型	注释(在使用时注意端口复用)
P12	TIM3_CC0	输出	与 TIMER3 的比较通道 CRC 相应的比较输出引脚
P13	TIM3_CC0N	输出	与 TIMER3 的比较通道 CRC 相应的比较互补输出引脚
P14	TIM3_CC1	输出	与 TIMER3 的比较通道 CC1 相应的比较输出引脚
P15	TIM3_CC1N	输出	与 TIMER3 的比较通道 CC1 相应的比较互补输出引脚
P16	TIM3_CC2	输出	与 TIMER3 的比较通道 CC2 相应的比较输出引脚
P17	TIM3_CC2N	输出	与 TIMER3 的比较通道 CC2 相应的比较互补输出引脚
P24	TIM4_CC0	输出	与 TIMER4 的比较通道 CRC 相应的比较输出引脚
P25	TIM4_CC0N	输出	与 TIMER4 的比较通道 CRC 相应的比较互补输出引脚
P26	TIM4_CC1	输出	与 TIMER4 的比较通道 CC1 相应的比较输出引脚
P27	TIM4_CC1N	输出	与 TIMER4 的比较通道 CC1 相应的比较互补输出引脚
P44	TIM4_CC2	输出	与 TIMER4 的比较通道 CC2 相应的比较输出引脚
P45	TIM4_CC2N	输出	与 TIMER4 的比较通道 CC2 相应的比较互补输出引脚

Copyright © 2022 武汉瑞纳捷半导体有限公司

a. 比较模式0

通过设置“TIM_T2CON”寄存器中“T2CM”＝0 来选择该模式。

在模式 0 中，当 TIM3 的计数值小于比较寄存器的值时，比较器的输出处于激活状态；否则，比较器输出处于非激活状态。

TIM3 有 3 组比较输出，每一组都是两路，每一路都有独立的输出使能信号和输出极性选择信号，可配置为互补输出，并且能够管理输出的瞬时关断和接通时间。

这段时间通常被称为死区，用户应该根据连接的输出器件和它们的特性(电平转换的延时、电源开关的延时等)来调整死区时间。

同时设置“TIM_OUT_EN”的 OUT_EN_[i]和 OUT_EN_N[i]（i＝0/1/2），并使能对应的互补输出使能，将插入死区，死区时间通过寄存器配置。

以互补输出都是高有效为例：OCiREF 为参考电压，下降沿为比较条件满足，上升沿为计数溢出。

- COMP0i 输出信号在计数值小于比较值时为高；计数值在大于等于比较值时为低。在计数溢出时，延迟死区时间后拉高。
- COMPN0i 输出信号在计数值小于比较值时为低；计数值在大于等于比较值时，延时死区时间后拉高。在计数溢出时输出拉低。

如果延迟大于当前有效的输出宽度(COMP0i 或者 COMPN0i)，则相应的通道不会产生脉冲。

带死区的互补输出：(OUT_P=1，OUT_P_N=1，OUT_EN=1，OUT_EN_N=1)

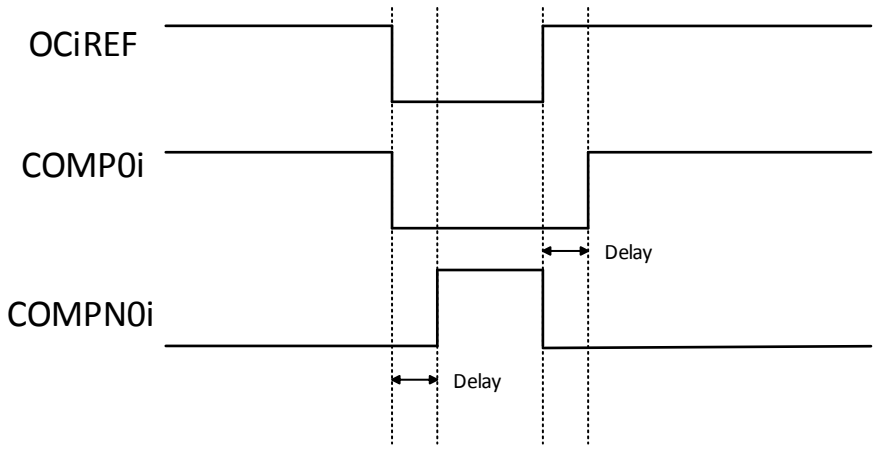


图 13-1 带死区的互补输出

Copyright © 2022 武汉瑞纳捷半导体有限公司

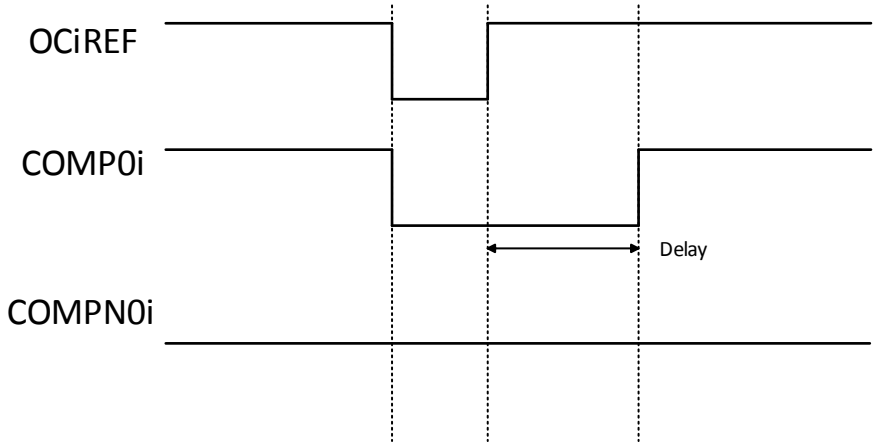


图 13-2 死区时间大于负脉冲

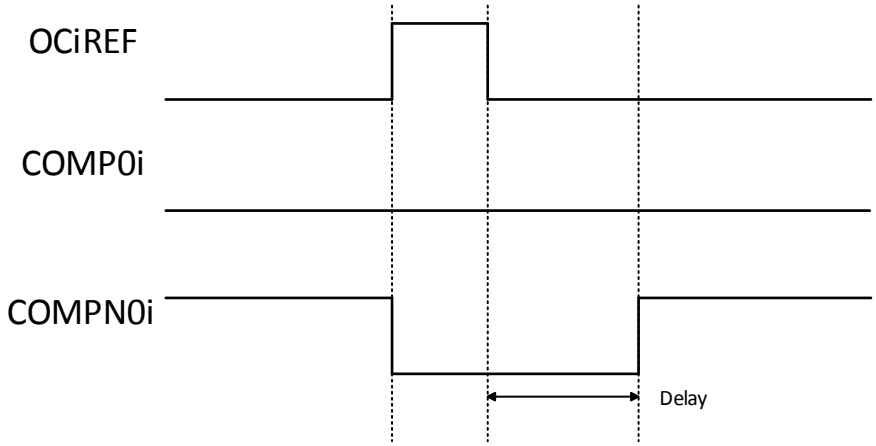


图 13-3 死区时间大于正脉冲

b. 比较模式1

通过设置“TIM_T2CON”寄存器中“T2CM”=1 来选择该模式。

该模式没有互补输出和死区插入的功能。

在比较模式 1 中，输出信号的转变是由软件决定的，TIMER 的溢出不会导致输出的变化。

在这种模式下，输出信号的转变可以被控制。

在比较模式 1 中，数据先被写入到“SHADOW 寄存器”，当比较信号被激活时，数据被写入输出寄存器。此时比较器的输出值为“写入 TIM_ID 数据的低 3 位”。

除了 a、b 两种功能之外，还具有刹车的功能。刹车可以打断比较的输出，使输出处于已知的状态。

系统复位后，刹车功能被禁止。设置 TIM_BOCEN 寄存器中的 BK_EN 位可以使能刹车功能，刹车输入信号的极性可以通过配置同一个寄存器中的 BK_POLA 位选择，配置 SAMPLE_FILTER，可以进行数字滤波，滤波宽度见寄存器描述。BK_EN、

Copyright © 2022 武汉瑞纳捷半导体有限公司

SAMPLE_FILTER 和 BK_POLA 可以被同时修改。（TIMER3 的刹车输入管脚为 TIM3_BKIN，TIMER4 为 TIM4_BKIN）

当发生刹车时，有如下动作：

- 每一个输出通道的输出值由 TIM_BREAK_OIS 寄存器的 BREAK_OIS 位设定的电平决定。
- 当使用互补输出时：
  - 输出首先被置于无效状态(取决于极性)。
  - 在死区之后根据 BREAK_OIS 和 BREAK_OIS_N 位指示的电平驱动输出端口，COMP 和 COMPN 不能被同时驱动到有效的电平。
- 如果设置了 TIM_IEN1 中的 BK_MSK=0，当刹车状态标志(TIMER3_IRCON 中的 BK_FLAG 位)为 1 时则产生一个中断。
- 如果设置了 TIMER3_BOCEN 中的 AUTO_EN 位，在计数溢出时，刹车状态标志会被自动清零。否则，刹车状态标志始终保持为 1 直到软件清零。

注：刹车输入为电平有效。当刹车输入有效时，设置 AUTO_EN 或软件清零，刹车状态标志不能被清除。详细配置见寄存器描述。

14.3.5. 捕获模式

三个 16 位的 CCU 寄存器都可以配置为在捕获模式下工作。

该模式下，实际的时间/计数器内容，在外部事件（模式 0）或者软件的写操作（模式 1）时被保存到 CCU 寄存器中。

a. 捕获模式 0

在模式 0 中，当出现以下情况时，TIM3 的内容会被捕获：

- 检测到输入“TIM3_CAP1”的上升沿、下降沿或者双沿，基于 EDGESEL[3:2](对应“CC1”寄存器)(TIM3_CAP1 是 TIMER3 的管脚，TIM4_CAP1 是 TIMER4 的管脚)
- 检测到输入“TIM3_CAP2”的上升沿、下降沿或者双沿，基于 EDGESEL[5:4](对应“CC2”寄存器)(TIM3_CAP2 是 TIMER3 的管脚，TIM4_CAP2 是 TIMER4 的管脚)
- 检测到输入“TIM3_CAP0”的上升沿、下降沿或者双沿，基于 EDGESEL[1:0]](对应“CRC”寄存器)(TIM3_CAP0 是 TIMER3 的管脚，TIM4_CAP0 是 TIMER4 的管脚)

TIM3 的内容会被锁存到合适的捕获寄存器中。在该模式下，可选择产生中断请

Copyright © 2022 武汉瑞纳捷半导体有限公司

求。

表 14-2 TIMER3 和 TIMER4 捕获模式 0 通道对应输入引脚

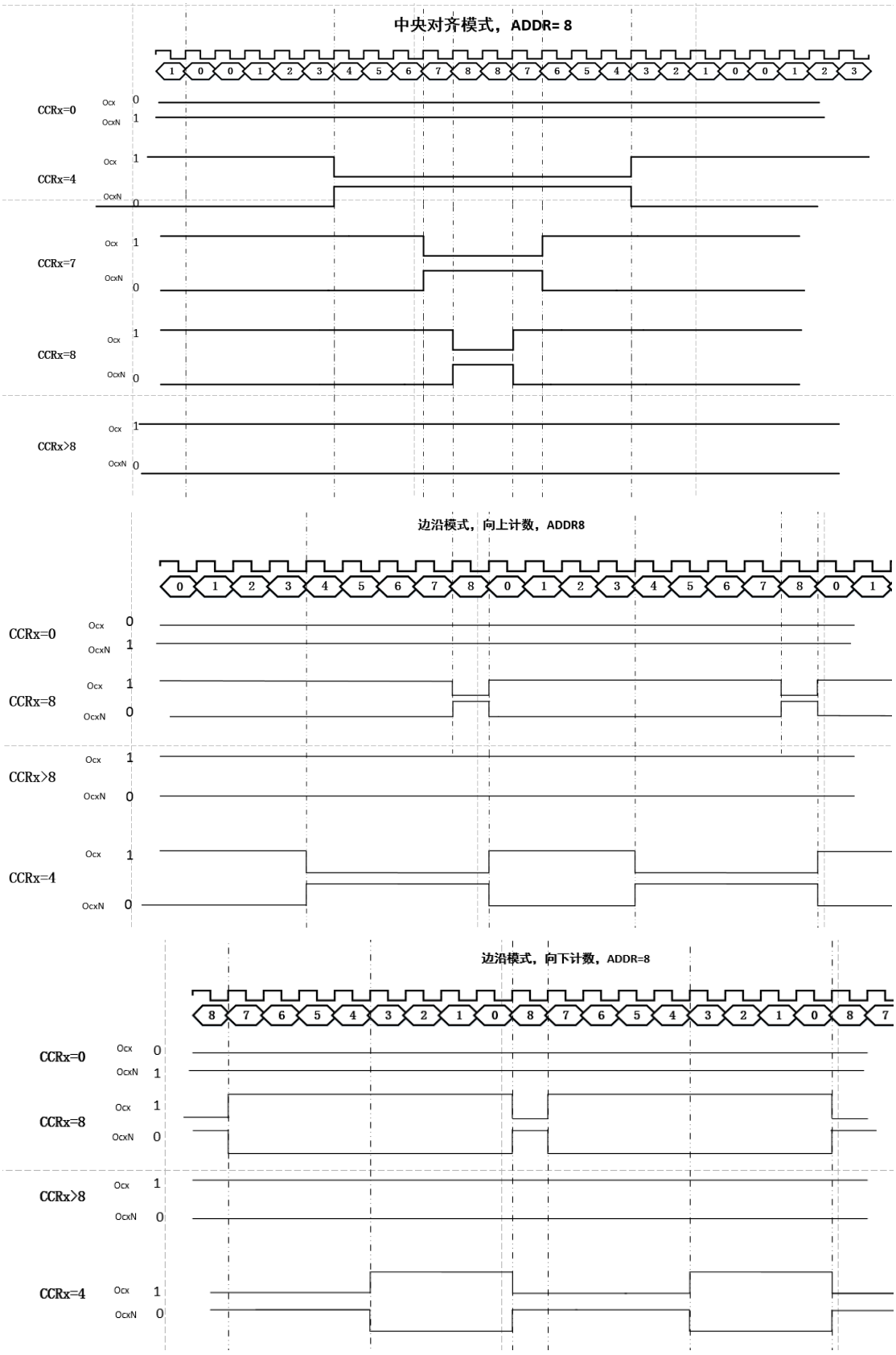
引脚名称	输出引脚名称	信号类型	注释(在使用时注意端口复用)
P11	TIM3_CAP0	输入	与 TIMER3 的 CRC 通道相应的通道 0 输入引脚
P42	TIM3_CAP1	输入	与 TIMER3 的 CC1 通道相应的通道 1 输入引脚
P43	TIM3_CAP2	输入	与 TIMER3 的 CC2 通道相应的通道 2 输入引脚
P35	TIM4_CAP0	输入	与 TIMER4 的 CRC 通道相应的通道 0 输入引脚
P20	TIM4_CAP1	输入	与 TIMER4 的 CC1 通道相应的通道 1 输入引脚
P21	TIM4_CAP2	输入	与 TIMER4 的 CC2 通道相应的通道 2 输入引脚

b. 捕获模式 1

在模式 1 中，TIM3 的捕获是由对任何专用捕获寄存器的写入引起的，写入捕获寄存器的值对此功能不会产生影响，TIM3 的计数值会被锁存到合适的捕获寄存器中。在该模式下，可选择产生中断请求。

Copyright © 2022 武汉瑞纳捷半导体有限公司

14.3.6. 中央对齐与边沿模式



14.4. 寄存器描述

TIM3 寄存器基地址：0x2180

偏移	名称	描述
0x00	TIM_IRCON	TIM 状态控制寄存器
0x01	TIM_IEN1	TIM 中断控制寄存器
0x02	TIM_T2CON	TIM 控制寄存器
0x03	TIM_CCEN	TIM 模式使能寄存器
0x04	TIM_CCL1	TIM 比较寄存器低字节
0x05	TIM_CCH1	TIM 比较寄存器高字节
0x06	TIM_CCL2	TIM 比较寄存器低字节
0x07	TIM_CCH2	TIM 比较寄存器高字节
0x08	TIM_CCL3	TIM 死区时间寄存器
0x0A	TIM_CRCL	TIM 比较寄存器低字节
0x0B	TIM_CRCH	TIM 比较寄存器高字节
0x0C	TIM_TL2	TIM 计数寄存器低字节
0x0D	TIM_TH2	TIM 计数寄存器高字节
0x0E	TIM_BOCEN	TIM 刹车控制寄存器
0x0F	TIM_ID	TIM 比较模式 1 数据寄存器
0x10	TIM_BREAK_OIS	TIM 刹车输出控制寄存器
0x11	TIM_OUT_P	TIM 比较输出极性选择寄存器
0x12	TIM_OUT_EN	TIM 比较输出通道使能寄存器
0x13	TIM_SAMPCC12	CC2/CC1 输入滤波寄存器
0x14	TIM_SAMPCRC	CRC 输入滤波寄存器
0x15	TIM_IESCON	比较/捕获中断控制寄存器
0x16	TIM_EDGES	捕获 0 状态寄存器
0x17	TIM_EDGESEL	捕获 0 沿事件选择寄存器
0x18	TIM_PSCL	预分频寄存器低 8 位
0x19	TIM_PSCH	预分频寄存器高 8 位

Copyright © 2022 武汉瑞纳捷半导体有限公司

0x1A	TIM_TRGOL	ADC 触发寄存器低 8 位
0x1B	TIM_TRGOH	ADC 触发寄存器高 8 位
0x1C	TIM_CON0	控制寄存器 0
0x1D	TIM_BRKC	刹车源控制寄存器
0x1E	TIM_EGR	事件发生寄存器
0x1F	TIM_CAP_CRCL	TIM 通道 0 捕获寄存器低字节
0x20	TIM_CAP_CRCH	TIM 通道 0 捕获寄存器高字节
0x21	TIM_CAP_CCL1	TIM 通道 1 捕获寄存器低字节
0x22	TIM_CAP_CCH1	TIM 通道 1 捕获寄存器高字节
0x23	TIM_CAP_CCL2	TIM 通道 2 捕获寄存器低字节
0x24	TIM_CAP_CCH2	TIM 通道 2 捕获寄存器高字节
0x20A9	TIM_COM_INT	TIM 比较模式 0 通道 0 沿事件选择寄存器

14.4.1. 状态控制寄存器 TIM_IRCON(偏移：00h)

比特	名称	属性	复位值	描述
7	EXF2	RW	0x0	外部重载标志位 外部重载发生时硬件置 1；软件写 1 清 0
6	TF2_UP	RW	0x0	向上溢出标志 硬件置 1；软件写 1 清 0
5	TF2_DOWN	RW	0x0	向下溢出标志 硬件置 1；软件写 1 清 0
4	BK_FLAG	RW	0x0	刹车标志位 硬件置 1；溢出自动清 0 或软件写 1 清 0
3	UP_INT_FLAG	RW	0x0	更新中断事件标志位 硬件置 1；软件写 1 清 0
2	COMPL_EN2	RW	0x0	通道 2 的死区使能 0：关闭 1：打开
1	COMPL_EN1	RW	0x0	通道 1 的死区使能 0：关闭 1：打开
0	COMPL_EN0	RW	0x0	通道 0 的死区使能 0：关闭 1：打开

14.4.2. 中断控制寄存器 TIM_IEN1 (偏移：01h)

Copyright © 2022 武汉瑞纳捷半导体有限公司

比特	名称	属性	复位值	描述
7	EXEN2	RW	0x0	外部重载中断使能位 0：禁止 1：使能
6	UP_IE	RW	0x0	更新中断使能位 0：禁止 1：使能
5	TF2_UP_IE	RW	0x0	向上溢出中断使能位 0：禁止 1：使能
4	TF2_DOWN_IE	RW	0x0	向下溢出中断使能位 0：禁止 1：使能
3	BK_IE	RW	0x0	刹车中断使能位 0：禁止 1：使能
2:0	RSV	-	-	保留

14.4.3. 控制寄存器 TIM_T2CON(偏移：02h)

比特	名称	属性	复位值	描述
7	PWM_M2	RW	0	比较输出通道 2 PWM 输出模式选择寄存器 0：PWM 模式 1 ● 边沿计数模式下：向上计数时，若计数值小于比较值，正向比较输出为有效电平，否则为无效电平；向下计数时，若计数值大于比较值，正向比较输出为无效电平，否则为有效电平。 ● 中央对齐计数模式下：向上计数时， 若计数值小于比较值，正向比较输出为有效电平，否则为无效电平；向下计数时，若计数值大于比较值，正向比较输出为无效电平，否则为有效电平。 1：PWM 模式 2 ● 边沿计数模式下：向上计数时，若计数值小于比较值，正向比较输出为无效电平，否则为有效电平；向下计数时，若计数值大于比较值，正向比较输出为有效电平，否则为无效电平。 ● 中央对齐计数模式下：向上计数时，若计数值小于比较值，正向比较输出为无效电平，否则为有效电平；向下计数时，若计数值大于

Copyright © 2022 武汉瑞纳捷半导体有限公司

				比较值，正向比较输出为有效电平，否则为无效电平。
6	PWM_M1	RW	0	比较输出通道 1 PWM 输出模式选择寄存器 0：PWM 模式 1 ● 边沿计数模式下：向上计数时，若计数值小于比较值，正向比较输出为有效电平，否则为无效电平；向下计数时，若计数值大于比较值，正向比较输出为无效电平，否则为有效电平。 ● 中央对齐计数模式下：向上计数时， 若计数值小于比较值，正向比较输出为有效电平，否则为无效电平；向下计数时，若计数值大于比较值，正向比较输出为无效电平，否则为有效电平。 1：PWM 模式 2 ● 边沿计数模式下：向上计数时，若计数值小于比较值，正向比较输出为无效电平，否则为有效电平；向下计数时，若计数值大于比较值，正向比较输出为有效电平，否则为无效电平。 ● 中央对齐计数模式下：向上计数时，若计数值小于比较值，正向比较输出为无效电平，否则为有效电平；向下计数时，若计数值大于比较值，正向比较输出为有效电平，否则为无效电平。
5	PWM_M0	RW	0	比较输出通道 0 PWM 输出模式选择寄存器 0：PWM 模式 1 ● 边沿计数模式下：向上计数时，若计数值小于比较值，正向比较输出为有效电平，否则为无效电平；向下计数时，若计数值大于比较值，正向比较输出为无效电平，否则为有效电平。 ● 中央对齐计数模式下：向上计数时， 若计数值小于比较值，正向比较输出为有效电平，否则为无效电平；向下计数时，若计数值大于比较值，正向比较输出为无效电平，否则为有效电平。 1：PWM 模式 2 ● 边沿计数模式下：向上计数时，若计数值小于比较值，正向比较输出为无效电平，否则为有效电平；向下计数时，若计数值大于比较值，正向比较输出为有效电平，否则为无效电平。

Copyright © 2022 武汉瑞纳捷半导体有限公司

				● 中央对齐计数模式下：向上计数时，若计数值小于比较值，正向比较输出为无效电平，否则为有效电平；向下计数时，若计数值大于比较值，正向比较输出为有效电平，否则为无效电平。 ***注：有效电平为高，无效电平为低***
4	T2R1	RW	0x0	重载模式选择 0x：重载禁止 10：模式 0 11：模式 1
3	T2R0	RW	0x0	
2	T2CM	RW	0x0	比较模式选择 0：模式 0 1：模式 1
1	T2I1	RW	0x0	输入选择 00：TIMER2 停止 01：定时器 10：RSV 11：门控定时器
0	T2I0	RW	0x0	

14.4.4. 模式使能寄存器 TIM_CCEN(偏移：03h)

比特	名称	属性	复位值	描述
7:6	EXF2_SEL	RW	0x0	外部重载触发方式选择 00：下降沿触发 01：上升沿触发 1x：双沿触发
5:4	COCAH2 COCAL2	RW	0x0	比较模式下，CC2 寄存器 COCAH2 / COCAL2 / 描述 0 / 0 / 比较禁止 0 / 1 / 保留 1 / 0 / 比较使能 1 / 1 / 保留
3:2	COCAH1 COCAL1	RW	0x0	比较模式下，CC1 寄存器 COCAH1 / COCAL1 / 描述 0 / 0 / 比较禁止 0 / 1 / 保留 1 / 0 / 比较使能 1 / 1 / 保留
1:0	COCAH0 COCAL0	RW	0x0	比较模式下，CRC 寄存器 COCAH0 / COCAL0 / 描述 0 / 0 / 比较禁止 0 / 1 / 保留

Copyright © 2022 武汉瑞纳捷半导体有限公司

				1	0	比较使能
				1	1	保留

14.4.5. 比较寄存器 TIM_CCL1 (偏移：04h)

比特	名称	属性	复位值	描述
7:0	CCL1	RW	0x01	TIM 的通道 1 比较寄存器低字节

14.4.6. 比较寄存器 TIM_CCH1 (偏移：05h)

比特	名称	属性	复位值	描述
7:0	CCL1	RW	0x00	TIM 的通道 1 比较寄存器高字节

14.4.7. 比较寄存器 TIM_CCL2(偏移：06h)

比特	名称	属性	复位值	描述
7:0	CCL2	RW	0x01	TIM 的通道 2 比较寄存器低字节

14.4.8. 比较寄存器 TIM_CCH2(偏移：07h)

比特	名称	属性	复位值	描述
7:0	CCH2	RW	0x00	TIM 的通道 2 比较寄存器高字节

14.4.9. 死区时间寄存器 TIM_CCL3 (偏移：08h)

比特	名称	属性	复位值	描述
7:0	CCL3	RW	0x01	TIM 的死区时间寄存器低字节（若死区使能，该寄存器不能为 0）

14.4.10. 比较寄存器 TIM_CRCL (偏移：0Ah)

比特	名称	属性	复位值	描述
7:0	CRCL	RW	0x01	TIM 的通道 0 比较寄存器低字节

14.4.11. 比较寄存器 TIM_CRCH (偏移：0Bh)

比特	名称	属性	复位值	描述
7:0	CRCH	RW	0x00	TIM 的通道 0 比较寄存器高字节

14.4.12. 计数寄存器 TIM_TL2(偏移：0Ch)

比特	名称	属性	复位值	描述
7:0	TL2_S	RW	0x00	TIM 的计数寄存器低字节

14.4.13. 计数寄存器 TIM_TH2(偏移：0Dh)

Copyright © 2022 武汉瑞纳捷半导体有限公司

比特	名称	属性	复位值	描述
7:0	TH2_S	RW	0x00	TIM 的计数寄存器高字节

14.4.14. 刹车控制寄存器 TIM_BOCEN(偏移：0Eh)

比特	名称	属性	复位值	描述
7:6	RSV	-	-	保留
5	BK_POLA	RW	0x0	刹车输入极性选择位 0：低电平 1：高电平
4	BK_EN	RW	0x0	刹车使能位 0：禁止 1：使能
3:0	SAMPLE_FILTER	RW	0x0	引脚刹车信号滤波 0000：无滤波 0001：信号宽度大于 2 个时钟周期 0010：信号宽度大于 4 个时钟周期 0011：信号宽度大于 8 个时钟周期 0100：信号宽度大于 12 个时钟周期 0101：信号宽度大于 16 个时钟周期 0110：信号宽度大于 24 个时钟周期 0111：信号宽度大于 32 个时钟周期 1000：信号宽度大于 48 个时钟周期 1001：信号宽度大于 64 个时钟周期 1010：信号宽度大于 80 个时钟周期 1011：信号宽度大于 96 个时钟周期 1100：信号宽度大于 128 个时钟周期 1101：信号宽度大于 160 个时钟周期 1110：信号宽度大于 192 个时钟周期 1111：信号宽度大于 256 个时钟周期

14.4.15. 比较模式 1 数据寄存器 TIM_ID(偏移：0Fh)

比特	名称	属性	复位值	描述
7	-	-	-	保留
6	COM2_FLAG	RW	0x0	CC2 比较标志位 硬件置 1，软件写 1 清 0
5	COM1_FLAG	RW	0x0	CC1 比较标志位 硬件置 1，软件写 1 清 0
4	COM0_FLAG	RW	0x0	CRC 比较标志位 硬件置 1，软件写 1 清 0
3	RSV	-	-	保留

Copyright © 2022 武汉瑞纳捷半导体有限公司

2	CC2_ID	-	-	比较模式 1 通道 2 输出电平选择位 0：低电平 1：高电平
1	CC1_ID	-	-	比较模式 1 通道 1 输出电平选择位 0：低电平 1：高电平
0	CRC_ID	-	-	比较模式 1 通道 0 输出电平选择位 0：低电平 1：高电平

14.4.16. 刹车输出控制寄存器 TIM_BREAK_OIS(偏移：10h)

比特	名称	属性	复位值	描述
7	RSV	-	-	保留
6	BREAK_OIS_2	RW	0x0	比较输出 2 空闲状态位 0：发生刹车，在死区时间后输出为 0 1：发生刹车，在死区时间后输出为 1
5	BREAK_OIS_1	RW	0x0	比较输出 1 空闲状态位 0：发生刹车，在死区时间后输出为 0 1：发生刹车，在死区时间后输出为 1
4	BREAK_OIS_0	RW	0x0	比较输出 0 空闲状态位 0：发生刹车，在死区时间后输出为 0 1：发生刹车，在死区时间后输出为 1
3	RSV	-	-	保留
2	BREAK_OIS_N2	RW	0x0	比较互补输出 2 空闲状态位 0：发生刹车，在死区时间后输出为 0 1：发生刹车，在死区时间后输出为 1
1	BREAK_OIS_N1	RW	0x0	比较互补输出 1 空闲状态位 0：发生刹车，在死区时间后输出为 0 1：发生刹车，在死区时间后输出为 1
0	BREAK_OIS_N0	RW	0x0	比较互补输出 0 空闲状态位 0：发生刹车，在死区时间后输出为 0 1：发生刹车，在死区时间后输出为 1

14.4.17. 比较输出极性选择寄存器 TIM_OUT_P(偏移：11h)

比特	名称	属性	复位值	描述
7	RSV	-	-	保留
6	OUT_P_2	RW	0x0	比较输出 2 极性选择位 若比较输出 2 使能或刹车 0：内部输出电平保持

Copyright © 2022 武汉瑞纳捷半导体有限公司

				1：内部输出电平取反 若比较输出 2 不使能且不刹车 0：输出低电平 1：输出高电平
5	OUT_P_1	RW	0x0	比较输出 1 极性选择位 若比较输出 1 使能或刹车 0：内部输出电平保持 1：内部输出电平取反 若比较输出 1 不使能且不刹车 0：输出低电平 1：输出高电平
4	OUT_P_0	RW	0x0	比较输出 0 极性选择位 若比较输出 0 使能或刹车 0：内部输出电平保持 1：内部输出电平取反 若比较输出 0 不使能且不刹车 0：输出低电平 1：输出高电平
3	RSV	-	-	保留
2	OUT_P_N2	RW	0x0``	比较互补输出 2 极性选择位 若比较互补输出 2 使能或刹车 0：内部输出电平保持 1：内部输出电平取反 若比较互补输出 2 不使能且不刹车 0：输出低电平 1：输出高电平
1	OUT_P_N1	RW	0x0	比较互补输出 1 极性选择位 若比较互补输出 1 使能或刹车 0：内部输出电平保持 1：内部输出电平取反 若比较互补输出 1 不使能且不刹车 0：输出低电平 1：输出高电平
0	OUT_P_N0	RW	0x0	比较互补输出 0 极性选择位 若比较互补输出 0 使能或刹车 0：内部输出电平保持 1：内部输出电平取反 若比较互补输出 0 不使能且不刹车 0：输出低电平 1：输出高电平

14.4.18. 比较输出通道使能寄存器 TIM_OUT_EN(偏移：12h)

比特	名称	属性	复位值	描述

7	RSV	-	-	保留
6	OUT_EN_2	RW	0x0	比较输出 2 使能位 0：禁止 1：使能
5	OUT_EN_1	RW	0x0	比较输出 1 使能位 0：禁止 1：使能
4	OUT_EN_0	RW	0x0	比较输出 0 使能位 0：禁止 1：使能
3	RSV	-	-	保留
2	OUT_EN_N2	RW	0x0	比较互补输出 2 使能位 0：禁止 1：使能
1	OUT_EN_N1	RW	0x0	比较互补输出 1 使能位 0：禁止 1：使能
0	OUT_EN_N0	RW	0x0	比较互补输出 0 使能位 0：禁止 1：使能

14.4.19. CC2/CC1 输入滤波寄存器 TIM_SAMPCC12(偏移：13h)

比特	名称	属性	复位值	描述
7：4	SAMP_FILTER_CC2	RW	0x0	CC2 通道捕获输入信号滤波 0000：无滤波 0001：信号宽度大于 2 个时钟周期 0010：信号宽度大于 4 个时钟周期 0011：信号宽度大于 8 个时钟周期 0100：信号宽度大于 12 个时钟周期 0101：信号宽度大于 16 个时钟周期 0110：信号宽度大于 24 个时钟周期 0111：信号宽度大于 32 个时钟周期 1000：信号宽度大于 48 个时钟周期 1001：信号宽度大于 64 个时钟周期 1010：信号宽度大于 80 个时钟周期 1011：信号宽度大于 96 个时钟周期 1100：信号宽度大于 128 个时钟周期 1101：信号宽度大于 160 个时钟周期 1110：信号宽度大于 192 个时钟周期 1111：信号宽度大于 256 个时钟周期
3：0	SAMP_FILTER_CC1	RW	0x0	CC1 通道捕获输入信号滤波 0000：无滤波 0001：信号宽度大于 2 个时钟周期

Copyright © 2022 武汉瑞纳捷半导体有限公司

				0010：信号宽度大于 4 个时钟周期 0011：信号宽度大于 8 个时钟周期 0100：信号宽度大于 12 个时钟周期 0101：信号宽度大于 16 个时钟周期 0110：信号宽度大于 24 个时钟周期 0111：信号宽度大于 32 个时钟周期 1000：信号宽度大于 48 个时钟周期 1001：信号宽度大于 64 个时钟周期 1010：信号宽度大于 80 个时钟周期 1011：信号宽度大于 96 个时钟周期 1100：信号宽度大于 128 个时钟周期 1101：信号宽度大于 160 个时钟周期 1110：信号宽度大于 192 个时钟周期 1111：信号宽度大于 256 个时钟周期

14.4.20. CRC 输入滤波寄存器 TIM_SAMPCRC(偏移：14h)

比特	名称	属性	复位值	描述
7:4	RSV	-	-	保留
3:0	SAMP_FILTER_CRC	RW	0x0	CRC 通道捕获输入信号滤波 0000：无滤波 0001：信号宽度大于 2 个时钟周期 0010：信号宽度大于 4 个时钟周期 0011：信号宽度大于 8 个时钟周期 0100：信号宽度大于 12 个时钟周期 0101：信号宽度大于 16 个时钟周期 0110：信号宽度大于 24 个时钟周期 0111：信号宽度大于 32 个时钟周期 1000：信号宽度大于 48 个时钟周期 1001：信号宽度大于 64 个时钟周期 1010：信号宽度大于 80 个时钟周期 1011：信号宽度大于 96 个时钟周期 1100：信号宽度大于 128 个时钟周期 1101：信号宽度大于 160 个时钟周期 1110：信号宽度大于 192 个时钟周期 1111：信号宽度大于 255 个时钟周期

14.4.21. 比较/捕获中断控制寄存器 TIM_IESCON(偏移：15h)

比特	名称	属性	复位值	描述
7	RSV	-	-	保留
6	COM2_IE	RW	0x0	CC2 比较中断使能位 0：禁止 1：使能

Copyright © 2022 武汉瑞纳捷半导体有限公司

5	COM1_IE	RW	0x0	CC1 比较中断使能位 0：禁止 1：使能
4	COM0_IE	RW	0x0	CRC 比较中断使能位 0：禁止 1：使能
3	RSV	-	-	保留
2	CAP_CRC_IE	RW	0x0	CRC 捕获中断使能位 0：禁止 1：使能
1	CAP_CC2_IE	RW	0x0	CC2 捕获中断使能位 0：禁止 1：使能
0	CAP_CC1_IE	RW	0x0	CC1 捕获中断使能位 0：禁止 1：使能

14.4.22. 捕获 0 状态寄存器 TIM_EDGES(偏移：16h)

比特	名称	属性	复位值	描述
7	RSV	-	-	保留
6	CC2_STATE	RO	0x0	CC2沿事件标志位 0：下降沿 1：上升沿
5	CC1_STATE	RO	0x0	CC1沿事件标志位 0：下降沿 1：上升沿
4	CRC_STATE	RO	0x0	CRC沿事件标志位 0：下降沿 1：上升沿
3	RSV	-	-	保留
2	CAP_CRC	RW	0x0	CRC 捕获标志位(硬件置 1，软件写 1 清 0)
1	CAP_CC2	RW	0x0	CC2 捕获标志位(硬件置 1，软件写 1 清 0)
0	CAP_CC1	RW	0x0	CC1 捕获标志位(硬件置 1，软件写 1 清 0)

14.4.23. 捕获 0 沿事件选择寄存器 TIM_EDGESEL(偏移：17h)

比特	名称	属性	复位值	描述
7：6	RSV	-	-	保留

Copyright © 2022 武汉瑞纳捷半导体有限公司

5：4	EDGESEL[5:4]	RW	0x0	CC2 捕获 0 触发沿事件 00：禁止 01：上升沿 10：下降沿 11：双沿
3：2	EDGESEL[3:2]	RW	0x0	CC1 捕获 0 触发沿事件 00：禁止 01：上升沿 10：下降沿 11：双沿
1：0	EDGESEL[1:0]	RW	0x0	CRC 捕获 0 触发沿事件 00：禁止 01：上升沿 10：下降沿 11：双沿

14.4.24. 预分频寄存器低 8 位 TIM_PSCL(偏移：18h)

比特	名称	属性	复位值	描述
7：0	PSCL	RW	0x00	预分频寄存器低 8 位

14.4.25. 预分频寄存器高 8 位 TIM_PSCH(偏移：19h)

比特	名称	属性	复位值	描述
7：0	PSCH	RW	0x00	预分频寄存器高 8 位

14.4.26. ADC 触发寄存器低 8 位 TIM_TRGOL(偏移：1Ah)

比特	名称	属性	复位值	描述
7：0	TRGOL	RW	0x00	ADC 触发寄存器低 8 位

14.4.27. ADC 触发寄存器高 8 位 TIM_TRGOH(偏移：1Bh)

比特	名称	属性	复位值	描述
7：0	TRGPH	RW	0x00	ADC 触发寄存器高 8 位

14.4.28. 控制寄存器 0 TIM_CON0(偏移：1Ch)

比特	名称	属性	复位值	描述

Copyright © 2022 武汉瑞纳捷半导体有限公司

7	UPDIS	RW	0x0	上溢无效寄存器： 0：当计数器上溢时产生更新事件。 1：当计数器上溢时不产生更新事件。
6	DOWNDIS	RW	0x0	下溢无效寄存器： 0：当计数器下溢时产生更新事件。 1：当计数器下溢时不产生更新事件。
5	URC	RW	0x0	更新源选择寄存器，软件通过该位选择更新事件的源： 0：下述任一事件产生一个更新事件，如果更新中断使能，则同时产生一个更新中断： --计数器上溢/下溢 --设置 TIM_EGR 寄存器的 UG 位 1：下述任一事件产生一个更新事件，如果更新中断使能，只有计数器上溢/下溢时产生一个更新中断： --计数器上溢/下溢 --设置 TIM_EGR 寄存器的 UG 位
4	RSV	-	-	保留
3：2	CNTMODE	RW	0x0	计数器模式控制 00：边沿对齐模式，计数器依据方向位 CNTDIR 的配置向上或向下计数。 其他：中央对齐模式 1，计数器交替向上向下计数， ***注：当计数器正在计数时，不允许从边沿对齐模式转换到中央对齐模式。***
1	CNTDIR	RW	0x0	计数器计数方向 0：TIMER 向上计数 1：TIMER 向下计数 ***注：当计数器配置为中央对齐模式时，该位只读***
0	ARPLE	RW	0x0	计数开始位 0：计数未开始 1：计数开始

14.4.29. 刹车源控制寄存器 TIM_BRKC(偏移：1Dh)

比特	名称	属性	复位值	描述
7	RSV	-	-	保留
6	IOSS	RW	0x0	空闲模式（MOE=0）下“关闭状态”（OUT_EN=0 或 OUT_EN_N=0）输出控制，该位用于当 MOE=0 且通道设置为输出时。 0：当定时器 CCx 通道的正向或互补通道有一路关闭，关闭的一路输出高阻，没关闭的一路输出正常的正向通道波形。

Copyright © 2022 武汉瑞纳捷半导体有限公司

				1：当定时器 CCx 通道的正向或互补通道有一路关闭，关闭的一路输出空闲电平（空闲电平受输出极性以及 PWM 模式控制），没关闭的一路输出正常的正向通道波形。
5	ROSS	RW	0x00	运行模式（MOE=1）下“关闭状态”（OUT_EN=0 或 OUT_EN_N=0）输出控制，该位用于当 MOE=1 且通道互补输出时。 0：当定时器 CCx 通道的正向或互补通道有一路关闭，关闭的一路输出高阻，没关闭的一路输出正常的正向通道波形。 1：当定时器 CCx 通道的正向或互补通道有一路关闭，关闭的一路输出空闲电平（空闲电平受输出极性以及 PWM 模式控制），没关闭的一路输出正常的正向通道波形。
4	EXTBRKE	RW	0x0	外部刹车功能使能寄存器： 0：禁止刹车输入 1：开启刹车输入 ***注 ：此位只控制外部输入的刹车信号，对寄存器配置的刹车事件 BG 无效。***
3	CMPBRKC	RW	0x0	比较器刹车功能使能寄存器： 0：禁止刹车输入 1：开启刹车输入 ***注：此位只控制外部输入的刹车信号，对寄存器配置的刹车事件 BG 无效。***
2	CMPSEL	RW	0x0	比较器刹车输入选择： 0：选择比较器 1 作为刹车源 1：选择比较器 2 作为刹车源
1	AOE	RW	0x0	自动输出使能： 0：MOE 只能被软件置 1 1：MOE 能被软件置 1 或在刹车信号无效后的更新时间自动置 1.
0	MOE	RW	0x0	主输出使能寄存器，一旦刹车输入有效，该位被硬件异步清 0，根据 AOE 的设置，可由软件置 1,或自动置 1，它仅对配置为输出通道有效。 0：使定时器处于空闲状态。定时器的输出受 IOSS、OIS0、OIS0N、CC0EN、CC0NEN、CC0P 和 CC0NP 位控制。 1：使定时器处于运行状态。定时器的输出受 ROSS、CC0EN、CC0NEN、CC0P 和 CC0NP 位的控制。

14.4.30. 事件发生寄存器 TIM_EGR(偏移：1Eh)

比特	名称	属性	复位值	描述

7	UG	W	0x0	产生更新事件，该位由软件置 1，由硬件自动清 0： 0：无动作； 1：初始化计数器、初始化预分频计数器、产生一个更新事件： ● 若在中央对齐模式或向上计数模式下，计数器被清 0； ● 若在向下计数模式下，计数器初始化为 TIM0_ARR 的值； ● 初始化预分频计数器为 0。
6:4	RSV	-	-	保留
3	WR_CAP_CC2	W	0x0	写 CC2 寄存器捕获
2	WR_CAP_CC1	W	0x0	写 CC1 寄存器捕获
1	WR_CAP_CRC	W	0x0	写 CRC 寄存器捕获
0	BG	W	0x0	产生刹车事件，该位由软件置 1，硬件自动清 0： 0：无动作； 1：产生一个刹车事件。此时 MOE=0，BIF=1， 若打开刹车中断使能，则产生刹车中断。

14.4.31. 捕获寄存器 TIM_CAP_CRCL (偏移：1Fh)

比特	名称	属性	复位值	描述
7:0	CAP_CRCL	RW	0x00	TIM 的通道 0 捕获寄存器低字节

14.4.32. 捕获寄存器 TIM_CAP_CRCH (偏移：20h)

比特	名称	属性	复位值	描述
7:0	CAP_CRCH	RW	0xFF	TIM 的通道 0 捕获寄存器高字节

14.4.33. 捕获寄存器 TIM_CAP_CCL1 (偏移：21h)

比特	名称	属性	复位值	描述
7:0	CAP_CCL1	RW	0x00	TIM 的通道 1 捕获寄存器低字节

14.4.34. 捕获寄存器 TIM_CAP_CCH1 (偏移：22h)

比特	名称	属性	复位值	描述
7:0	CAP_CCH1	RW	0xFF	TIM 的通道 1 捕获寄存器高字节

14.4.35. 捕获寄存器 TIM_CAP_CCL2(偏移：23h)

比特	名称	属性	复位值	描述

Copyright © 2022 武汉瑞纳捷半导体有限公司

7:0	CCL2	RW	0x00	TIM 的通道 2 捕获寄存器低字节

14.4.36. 捕获寄存器 TIM_CAP_CCH2(偏移：24h)

比特	名称	属性	复位值	描述
7:0	CCH2	RW	0xFF	TIM 的通道 2 捕获寄存器高字节

14.4.37. 比较模式 0 通道 0 沿事件选择寄存器 TIM_COM_INT（地址 0x20A9）

比特	名称	属性	复位值	描述
7	TIM3_COM_INT	WO	0x0	比较模式 0 通道 0 中断沿选择位： 0：上升沿 1：下降沿
6	TIM4_COM_INT	WO	0x0	比较模式 0 通道 0 中断沿选择位： 0：上升沿 1：下降沿
5：0	RSV	-	-	保留

14.5. 使用流程

14.5.1. 定时

a. 定时器

1) 写 TIM_CON0 的 CNTDIR，选择计数方向以及选择边沿计数模式。写 TIM_T2CON 的 T2I0、T2I1 位选择定时器。写 TIM3_PSCL 和 TIM3_PSCH 配置时钟分频；
2) 设置 TIM_TL2、TIM_TH2 初值，根据选择的 CNTDIR 写 TIMER_IEN1 对应的溢出中断使能位。(向上计数会循环从 0x00 记到设定值产生溢出中断。向下计数会循环从设定值计到 0x00 产生溢出中断)；
3) 配置 TIM_EGR 的 UG 位产生软更新，之后写 TIM_CON0 的 ARPLE 计数开始；
4) 溢出后，TIM_IRCON 的溢出标志位被置位，需要软件清除溢出标志位；
5) 可采用轮询或中断方式处理定时器事件。

b. 重载定时器

1) 写 TIM_CON0 的 CNTDIR，选择计数方向以及选择边沿计数模式。写 TIM_T2CON 的 T2I0、T2I1 位选择定时器并选择重载模式。写 TIM3_PSCL 和 TIM3_PSCH 配置时钟分频。
2) 设置 TIM_TL2、TIM_TH2 初值，根据选择的 CNTDIR 写 TIMER_IEN1 对应的溢出中断使能位。(向上计数会循环从 0x00 记到设定值产生溢出中断。向下计数会循环从设定值计到 0x00 产生溢出中断)；
3) 配置 TIM_EGR 的 UG 位产生软更新，之后写 TIM_CON0 的 ARPLE 计数开始；

Copyright © 2022 武汉瑞纳捷半导体有限公司

4) 溢出后，TIM_IRCON 的溢出标志位被置位，需要软件清除溢出标志位；
5) 可采用轮询或中断方式处理定时器事件。

c. 门控定时器

1) 写 TIM_CON0 的 CNTDIR，选择计数方向以及选择边沿计数模式。写 TIM_T2CON 的 T2I0、T2I1 位选择门控定时器并选择重载模式 0。写 TIM3_PSCL 和 TIM3_PSCH 配置时钟分频；
2) 设置 TIM_TL2、TIM_TH2 初值，根据选择的 CNTDIR 写 TIMER_IEN1 对应的溢出中断使能位。(向上计数会循环从 0x00 记到设定值产生溢出中断。向下计数会循环从设定值计到 0x00 产生溢出中断)；
3) 设置 IO 复用为 TIM3_ETR；
4) 配置 TIM_EGR 的 UG 位产生软更新，之后写 TIM_CON0 的 ARPLE 计数开始；
5) TIM3_ETR 给高电平，TIM 工作；TIM3_ETR 给低电平，TIM 停止， TIM_TL2、TIM_TH2 的值保持。
6) 可采用轮询或中断方式处理定时器事件。

14.5.2. 比较

a. 比较模式 0

1) 将 IO 复用为比较互补输出 TIM3_CCx 与 TIM3_CCxN（x=0~2）；
2) 写 TIM_CON0 的 CNTDIR，选择计数方向以及选择边沿计数模式（若为中央对齐模式则计数方向只读且不可使用重载）。写 TIM_T2CON 的 T2I0、T2I1 位选择定时器、重载模式 0、PWM 模式、比较模式。写 TIM3_PSCL 和 TIM3_PSCH 配置时钟分频；
3) 写 TIM_TL2、TIM_TH2，设置 TIM_CRCL、TIM_CRCH、TIM_CCL1、TIM_CCH1、TIM_CCL2 、TIM_CCH2（如果配置死区可在此一起配置并打开死区使能）；
4) 写 TIM_OUT_EN、TIM_CCEN，设置比较输出通道与比较使能。写 TIMER_OUT_P，设置输出极性。可选择打开 TIM_IEN1，设置溢出中断使能。写 TIM_IESCON，打开比较中断使能；
5) 配置 TIM_EGR 的 UG 位产生软更新，写 TIM_BRKC 的 MOE 总输出使能位，写 TIM_CON0 的 ARPLE 计数开始。当计数值与设定值相等时，对应的比较输出设定的电平，触发比较中断，需要软件清除比较中断标志；
6) 可采用轮询或中断方式处理定时器事件。

b. 比较模式 1

1) 写 TIM_T2CON，设置比较、重载工作模式；
2) 设置 TIM_TL2、TIM_TH2、TIM_CRCL、TIM_CRCH、TIM_CCL1、TIM_CCH1、TIM_CCL2 、TIM_CCH2；
3) 写 TIM_CCEN，使能比较模式，写 TIM_ID；

Copyright © 2022 武汉瑞纳捷半导体有限公司

4) 当 TIM 的值与设定值相等时，对应的比较器输出为 TIM_ID 的低三位电平值；
5) 溢出后，TIM_IRCON 的 TF2 位被置位，并触发溢出中断，需要软件清除标志位，但比较的输出不受影响；
6) 可采用轮询或中断方式处理定时器事件。

14.5.3. 捕获

a. 捕获模式 0

1) 写 TIM_CON0 的 CNTDIR，选择计数方向以及选择边沿计数模式。写 TIM_T2CON 的 T2I0、T2I1 位选择定时器并选择重载模式 0。写 TIM3_PSCL 和 TIM3_PSCH 配置时钟分频。
2) 设置 TIM_TL2、TIM_TH2 初值；
3) 将相应 IO 复用为 TIM3_CAPx(x=0~8)并指定为捕获通道 0~2 的输入管脚。写 TIM_EDGESEL 选择上升沿、下降沿或者是双沿捕获。写 TIM_IESCON 使能捕获中断；
4) 配置 TIM_EGR 的 UG 位产生软更新，之后写 TIM_CON0 的 ARPLE 计数开始；
5) 当检测到选定的 TIM3_CAPx(x=0~8)满足条件时，TIM 的值保存到相应的捕获寄存器中。
6) 可采用轮询或中断方式处理定时器事件。

b. 捕获模式 1

1) 写 TIM_CON0 的 CNTDIR，选择计数方向以及选择边沿计数模式。写 TIM_T2CON 的 T2I0、T2I1 位选择定时器并选择重载模式 0。写 TIM3_PSCL 和 TIM3_PSCH 配置时钟分频。
2) 写 TIM_TL2、TIM_TH2；
3) 写 TIM_IESCON 使能捕获中断；
4) 配置 TIM_EGR 的 UG 位产生软更新，之后写 TIM_CON0 的 ARPLE 计数开始；
5) 当对捕获寄存器 TIMER_EGR 的 WR_CAP_CC2 或 CC1 或 CRC 进行写操作，产生捕获，TIM 的值保存到相应的捕获寄存器中。
6) 可采用轮询或中断方式处理定时器事件。

Copyright © 2022 武汉瑞纳捷半导体有限公司

15.低功耗定时器（LPTIMER）

15.1. 概述

芯片有 1 个 16 位的低功耗定时/计数器：LPTIMER。作定时器使用时，每个低速时钟周期，计数器加 1；作计数器使用时，检测到外部管脚的沿事件，计数器加 1。

15.2. 主要特性

- 16 位递增自动重载定时/计数器；
- 工作时钟为低速时钟 LSE 或 LSI；
- 作为计数器使用时，计数事件可选上升沿、下降沿或者双沿；
- 具有 ACTIVE HALT 模式下唤醒功能；
- 中断在 UEV（Update Event）事件时产生：计数器往上从 0 计数到自动重载值，重新从 0 开始并产生一个计数溢出事件。

注：使用事件计数时，需将 P36 复用为 LPTIM_ETR，且计数值无法读取

15.3. 寄存器描述

LPTIMER 寄存器基地址：0x2150

偏移	名称	描述
0x00	LPTIM_TL0	自动重载寄存器低字节
0x01	LPTIM_TH0	自动重载寄存器高字节
0x02	LPTIM_T0CON	控制/状态寄存器
0x03	LPTIM_T0MOD	模式选择寄存器

15.3.1. 自动重载寄存器低字节 LPTIM_TL0(偏移：00h)

比特	名称	属性	复位值	描述
7:0	TL0	RW	0x00	自动重载寄存器 计数器达到的最大值低字节，通过软件配置

15.3.2. 自动重载寄存器高字节 LPTIM_TH0(偏移：01h)

比特	名称	属性	复位值	描述
7:0	TH0	RW	0x00	自动重载寄存器 计数器达到的最大值高字节，通过软件配置

15.3.3. 控制/状态寄存器 LPTIM_T0CON(偏移：02h)

比特	名称	属性	复位值	描述
7	TF0	RW	0x0	溢出标志位 0：未溢出 1：溢出

Copyright © 2022 武汉瑞纳捷半导体有限公司

				硬件置 1，软件写 1 清 0
6:3	RSV	-	-	保留
2	TR0	RW	0x0	停/启控制位 0：停止 1：启动
1	RSV	-	-	保留
0	INT_EN	RW	0x0	中断使能位 0：禁止 1：使能

15.3.4. 模式选择寄存器 LPTIM_T0MOD(偏移：03h)

比特	名称	属性	复位值	描述
7:3	RSV	-	-	保留
2:1	EDGE_SEL	RW	0x0	计数事件选择 00：下降沿 01：上升沿 1x：双沿
0	CT0	RW	0x0	定时/计数模式选择位 0：定时 1：计数

15.4. 使用说明

计数：指对外部事件进行计数。每来一个外部事件输入，计数器就加 1，当计数器计到设定值产生中断请求。

定时：指当接入周期变化的计数脉冲信号，计数器从 0 开始加 1 计数，计到设定的计数值产生中断，表示定时时间到。定时时间计算如下：

$$t = M * T$$

式中，t 为定时时间；M 为设置的计数值；T 为计数脉冲的周期。

计数脉冲周期：$T = \frac{1}{f}$（f为低功耗定时器工作时钟）

15.4.1. 使用流程

a. 定时器配置

1) 写时钟控制寄存器 SCCM3，使能 LPTIMER 的时钟；
2) 写 LPTIME_T0MOD，设置为定时模式；
3) 设置 LPTIME_TH0、LPTIME_TL0 自动重载值；
4) 设置 LPTIME_T0CON.TR0 = 1，启动 LPTIM；
5) 使能 LPTIMER 中断与总中断；
6) 溢出后，LPTIME_T0CON 的 TF0 被置位，并触发中断，需要软件清除标志。

b. 事件计数配置

1) 写时钟控制寄存器 SCCM3，使能 LPTIMER 的时钟
2) 将 P06 端口复用为 LPTIM_ETR；

Copyright © 2022 武汉瑞纳捷半导体有限公司

3) 写 LPTIME_T0MOD，设置为计时模式并选择沿事件；
4) 设置 LPTIME_TH0、LPTIME_TL0 自动重载值；
5) 设置 LPTIME_T0CON.TR0 = 1，启动 LPTIM；
6) 使能 LPTIMER 中断与总中断；
7) 溢出后，LPTIME_T0CON 的 TF0 被置位，并触发中断，需要软件清除标志。

Copyright © 2022 武汉瑞纳捷半导体有限公司

16.实时时钟（RTC）

16.1. 概述

芯片有一个独立的实时时钟（RTC），它提供了一个时钟和可编程的闹钟。RTC 还包括一个用于唤醒低功耗的单元。

RTC 包含了对世纪/年/月/日/小时（24 小时制）/分/秒的二进制编码，对 28 天、29 天（闰年）、30 天和 31 天的月份自动执行调整。

16.2. RTC 主要特性

- 时钟源可为 LSE 或 LSI；
- 日历功能：世纪、年、月、日、时、分、秒，支持月份天数自动调整；
- 周期定时功能：支持 1s、1min、1h、1d 产生周期 TICK 中断；
- 闹钟功能：当日历时间与设置的闹钟时间匹配时，产生 ALARM 中断。

16.3. RTC 功能描述

该 RTC 单元功能的实现由一组可读写的 8 位寄存器控制（具体参照 RTC 寄存器描述）。时钟选择具体参照寄存器 LP_CTRL 1（具体参照第 7 章 时钟控制）。

16.4. RTC 寄存器

RTC 寄存器基地址：0x2080

偏移	名称	描述
0x01	RTC_HOUR	RTC 小时配置寄存器
0x02	RTC_MINUTE	RTC 分钟配置寄存器
0x03	RTC_SECOND	RTC 秒配置寄存器
0x05	RTC_YEAR	RTC 年配置寄存器
0x06	RTC_MONTH	RTC 月份配置寄存器
0x07	RTC_DATE	RTC 日期配置寄存器
0x08	RTC_CENTURY	RTC 世纪配置寄存器
0x09	RTC_ALARM_HOUR	RTC 闹铃小时配置寄存器
0x0A	RTC_ALARM_MINUTE	RTC 闹铃分钟配置寄存器
0x0B	RTC_ALARM_SECOND	RTC 闹铃秒配置寄存器
0x0D	RTC_ALARM_YEAR	RTC 闹铃年配置寄存器
0x0E	RTC_ALARM_MONTH	RTC 闹铃月份配置寄存器
0x0F	RTC_ALARM_DATE	RTC 闹铃日期配置寄存器
0x10	RTC_SECOND_CNT_3	RTC 秒计数校准寄存器高字节
0x11	RTC_SECOND_CNT_2	RTC 秒计数校准寄存器次高字节

Copyright © 2022 武汉瑞纳捷半导体有限公司

0x12	RTC_SECOND_CNT_1	RTC 秒计数校准寄存器次低字节
0x13	RTC_SECOND_CNT_0	RTC 秒计数校准寄存器低字节
0x14	RTC_TICK_CTRL	RTC Tick 中断控制寄存器
0x15	RTC_ALARM_CTRL	RTC Alarm 中断控制寄存器
0x16	RTC_RELOAD	RTC 重载使能寄存器
0x17	RTC_START	RTC 开始工作使能寄存器
0x18	RTC_CLR_INT	RTC 中断标志清除寄存器
0x1A	RTC_CNT_EN	RTC 计数使能寄存器

16.4.1. 小时配置寄存器 RTC_HOUR (偏移：01H)

比特	名称	属性	复位值	描述
7:0	RTC_HOUR	RW	0x00	Hour 配置寄存器（范围为 0~59） W：配置初始 hour； R：RTC 计数 hour 值；

16.4.2. 分钟配置寄存器 RTC_MINUTE (偏移：02H)

比特	名称	属性	复位值	描述
7:0	RTC_MIN	RW	0x00	Minute 配置寄存器（范围为 0~59） W：配置初始 Minute； R：RTC 计数 Minute 值；

16.4.3. 秒配置寄存器 RTC_SECOND (偏移：03H)

比特	名称	属性	复位值	描述
7:0	RTC_SEC	RW	0x00	Second 配置寄存器（范围为 0~59） W：配置初始 Second； R：RTC 计数 Second 值；

16.4.4. 年配置寄存器 RTC_YEAR (偏移：05H)

比特	名称	属性	复位值	描述
7:0	RTC_YEAR	RW	0x00	Year 配置寄存器（范围为 0~99） W：配置初始 Year； R：RTC 计数 Year 值；

16.4.5. 月配置寄存器 RTC_MONTH(偏移：06H)

比特	名称	属性	复位值	描述
7:0	RTC_MON	RW	0x00	Month 配置寄存器（范围为 0~59） W：配置初始 Month； R：RTC 计数 Month 值；

16.4.6. 日期配置寄存器 RTC_DATE (偏移：07H)

比特	名称	属性	复位值	描述
7:0	RTC_DATE	RW	0x00	Month 配置寄存器（范围为 0~59）

Copyright © 2022 武汉瑞纳捷半导体有限公司

				W：配置初始 Month； R：RTC 计数 Month 值；

16.4.7. 世纪配置寄存器 RTC_CENTURY (偏移：08H)

比特	名称	属性	复位值	描述
7:0	RTC_CENTURY	RW	0x00	Century 配置寄存器（范围为 0~59） W：配置初始 Century； R：RTC 计数 Century 值；

16.4.8. 闹铃小时配置寄存器 RTC_ALARM_HOUR (偏移：09H)

比特	名称	属性	复位值	描述
7:0	ALARM_HOUR	RW	0x00	闹铃 Hour 配置寄存器 W：配置初始闹铃 Hour； R：读取闹铃 Hour 值；

16.4.9. 闹铃分钟配置寄存器 RTC_ ALARM_MINUTE (偏移：0AH)

比特	名称	属性	复位值	描述
7:0	ALARM_MINUTE	RW	0x00	闹铃 Minute 配置寄存器 W：配置初始闹铃 Minute； R：读取闹铃 Minute 值；

16.4.10. 闹铃秒配置寄存器 RTC_ ALARM_ SECOND (偏移：0BH)

比特	名称	属性	复位值	描述
7:0	ALARM_SECOND	RW	0x00	闹铃 Second 配置寄存器 W：配置初始闹铃 Second； R：读取闹铃 Second 值；

16.4.11. 闹铃年配置寄存器 RTC_ ALARM_ YEAR (偏移：0DH)

比特	名称	属性	复位值	描述
7:0	ALARM_YEAR	RW	0x00	闹铃 Year 配置寄存器 W：配置初始闹铃 Year； R：读取闹铃 Year 值；

16.4.12. 闹铃月份配置寄存器 RTC_ ALARM_ MONTH (偏移：0EH)

比特	名称	属性	复位值	描述
7:0	ALARM_MONTH	RW	0x00	闹铃 Month 配置寄存器 W：配置初始闹铃 Month； R：读取闹铃 Month 值；

16.4.13. 闹铃日期配置寄存器 RTC_ ALARM_ DATE (偏移：0FH)

比特	名称	属性	复位值	描述
7:0	ALARM_DATE	RW	0x00	闹铃 Date 配置寄存器

Copyright © 2022 武汉瑞纳捷半导体有限公司

				W：配置初始闹铃 Date； R：读取闹铃 Date 值；

16.4.14. 秒计数校准寄存器 RTC_SECOND_CNT_x(x=3~0)(偏移：10H~13H)

比特	名称	属性	复位值	描述
31:0	SEC_CNT	RW	0x8000	秒计数校准寄存器 当采用 LSI 或 LSE 作为 RTC 计数时钟时，均需要配置此寄存器。配置值为使用时钟计数 1 秒需要的 cycle 个数 W：配置秒计数初始值； R：读取秒计数初始值；

注：该寄存器由 Sec_cnt_3、Sec_cnt_2、Sec_cnt_1、Sec_cnt_0 依次拼接而成，其中 Sec_cnt_3 为最高字节，Sec_cnt_0 为最低字节。

16.4.15. TICK 中断控制寄存器 RTC_TICK_CTRL (偏移：14H)

比特	名称	属性	复位值	描述
7:3	RSV	-	-	保留
2:1	TICK_SEL	RW	0x0	Tick 中断源头选择 00：Second 01：Minute 10：Hour 11：Date
0	TICK_INT_EN	RW	0x0	Tick 中断使能位 0：不使能 1：使能

16.4.16. ALARM 中断控制寄存器 RTC_ALARM_CTRL(偏移：15H)

比特	名称	属性	复位值	描述
7	RSV	-	-	保留
6	ALARM_YEAR_SEL	RW	0x0	闹铃中断选择位： 0：闹铃中断不选中 Year 1：闹铃中断选中 Year
5	ALARM_MON_SEL	RW	0x0	闹铃中断选择位： 0：闹铃中断不选中 Month 1：闹铃中断选中 Month
4	ALARM_DATE_SEL	RW	0x0	闹铃中断选择位： 0：闹铃中断不选中 Date 1：闹铃中断选中 Date
3	ALARM_HOUR_SEL	RW	0x0	闹铃中断选择位： 0：闹铃中断不选中 Hour 1：闹铃中断选中 Hour

Copyright © 2022 武汉瑞纳捷半导体有限公司

2	ALARM_MIN_SEL	RW	0x0	闹铃中断选择位： 0：闹铃中断不选中 Minute 1：闹铃中断选中 Minute
1	ALARM_SEC_SEL	RW	0x0	闹铃中断选择位： 0：闹铃中断不选中 Second 1：闹铃中断选中 Second
0	ALARM_INT_EN	RW	0x0	闹铃中断使能位 0：不使能 1：使能

注：使能闹铃中断后，若 RTC 当前各寄存器计数值与 Alarm_Ctrl[6:1]选中中断的寄存器的值全部相等时，产生中断。

16.4.17. 重载使能寄存器 RTC_RELOAD(偏移：16H)

比特	名称	属性	复位值	描述
7:1	RSV	-	-	保留
0	RTC_RELOAD	RW	0x0	RTC 重载使能位 0：不重载 RTC 寄存器值； 1：重载 RTC 寄存器值；

注：此处重新加载的寄存器为：Hour、Minute、Second、Year、Month、Date、Century。

16.4.18. 开始工作使能寄存器 RTC_START(偏移：17H)

比特	名称	属性	复位值	描述
7:1	RSV	-	-	保留
0	RTC_START	RW	0x0	RTC 工作使能位 0：不使能； 1：使能；

16.4.19. 中断标志清除寄存器 RTC_CLR_INT (偏移：18H)

比特	名称	属性	复位值	描述
7:2	RSV	-	-	保留
1	CLR_ALARM_INT	RW	0x0	清 ALARM 中断位： W：1：清除闹钟中断； R：0：此时无闹钟中断； 1：此时有闹钟中断；
0	CLR_TICK_INT	RW	0x0	清 TICK 中断位： W：1：清除 Tick 中断； R：0：此时有 Tick 中断； 1：此时有 Tick 中断；

16.4.20. 计数使能寄存器 RTC_CNT_EN (偏移：1AH)

比特	名称	属性	复位值	描述

Copyright © 2022 武汉瑞纳捷半导体有限公司

7:1	RSV	-	-	保留
0	RTC_CNT_EN	RW	0x0	RTC 计数使能寄存器： 1：使能 RTC 计数； 0：不使能 RTC 计数；

16.5. RTC 操作说明

1) 使能外部 32.768K 或者内部 32K，配置 LP_CTRL1 的 LF_32K_SEL 位，选择低速时钟源；
2) 初始化 HOUR、MINUTE、SECOND、YEAR、MONTH、DATE、CENTURY 等寄存器；
3) 根据工作时钟频率，配置 SEC_CNT 寄存器；SEC_CNT 为工作时钟计数 1 S 时对应的计数值；
4) 根据场景配置 TICK 中断对应的寄存器 TICK_CTRL；
5) 根据场景配置 ALARM 中断对应的寄存器 ALARM_HOUR、ALARM_MINUTE、ALARM_SECOND、 ALARM_YEAR、ALARM_MONTH、ALARM_DATE、ALARM_CTRL；
6) 分别配置 RTC_CNT_EN 和 RTC_START 寄存器，启动 RTC 正常工作；

注：改变 HOUR、MINUTE、SECOND、YEAR、MONTH、DATE、CENTURY 等寄存器值，必须再次配置 RTC_RELOAD 和 RTC_START 寄存器。

Copyright © 2022 武汉瑞纳捷半导体有限公司

17.看门狗（WDT）

17.1. 概述

看门狗定时器（WDT）采用 15 位的递增计数器，从可编程的加载值计数到溢出值。当计数器计数到溢出值时，如果看门狗动作设为复位，则看门狗模块输出复位信号，复位系统；如果看门狗动作设为中断，则触发看门狗中断，如果在中断中未及时进行喂狗操作（重新设置加载值），则产生复位信号复位系统。用户可以通过设置看门狗使能位来停止/启动计数器。

17.2. WDT 主要特性

- 15 位的递增计数器；
- 可选择的计数时钟周期；
- 可编程装载值；
- 看门狗中断和复位请求都有相应的屏蔽位，默认是关闭（即不能产生中断和复位请求）。

17.3. 看门狗复位

看门狗复位功能时可配置寄存器 SCRSTEN 的 bit1，当看门狗计数器计数到加载值溢出时，则会产生一个系统复位。

17.4. 看门狗中断

看门狗中断时可配置寄存器 IEN3 的 bit1，使能看门狗中断，同时使能总中断。

17.5. 看门狗开始工作

看门狗工作使能可配置寄存器 IEN1 的 bit6，使能看门狗定时器开始工作标志。

17.6. 看门狗计数器

看门狗计数器计数时钟有 24 或 3072 个时钟周期计数递增一次，可通过配置寄存器 WDTREL 的 bit7 选择。

看门狗计数器为 15 位计数器，可配置寄存器 WDTREL 的 bit[6:0]设置计数加载值的高 7 位，低 8 位默认为 0x00。

17.7. 看门狗寄存器

WDT 特殊功能寄存器

偏移	名称	描述

Copyright © 2022 武汉瑞纳捷半导体有限公司

0x86	WDTREL	加载值寄存器
0xAE	WDT_LOCK_0	加锁寄存器 0
0xAF	WDT_LOCK_1	加锁寄存器 1

17.7.1. WDTREL 加载值寄存器(偏移：86H)

比特	名称	属性	复位值	描述
7	DIV	WO	0x0	计数时钟周期选择位： 0：看门狗计数器每 24 个时钟周期递增一次 1：看门狗计数器每 3072（128*24）个时钟周期递增一次
6:0	LOAD	WO	0x00	7 位寄存器，用来存入加载值的高 7 位。

注：实际 15 位计数器计数值的高 7 位为 LOAD 值，低 8 位值为 0x00。

若 LOAD[6:0]的 7 位寄存器值对应的十进制数据为 n，看门狗复位时间（时间单位：秒）的计算公式如下：

- 当 DIV 配置为 0 时，则看门狗复位时间为：

$$\frac{(128-n)*256*24}{16000000}$$

- 当 DIV 配置为 1 时，则看门狗复位时间为：

$$\frac{(128-n)*256*3072}{16000000}$$

17.7.2. WDT_LOCK_0 加锁寄存器(偏移：AEH)

比特	名称	属性	复位值	描述
7:0	WDT_LOCK_0	WO	0x00	加锁寄存器 0 0xCD：解锁 其他：加锁 软件解锁，硬件自动清零

17.7.3. WDT_LOCK_1 加锁寄存器(偏移：AFH)

比特	名称	属性	复位值	描述
7:0	WDT_LOCK_1	WO	0x00	加锁寄存器 1 0xEF：解锁 其他：加锁 软件解锁，硬件自动清零

注：1.配置时需要同时配置 WDT_LOCK_0 和 WDT_LOCK_1 为解锁值才能解锁；

2.配置解锁后要立即配置 WDG 相关寄存器，且每次配置 WDTREL 和 IEN1 前均需先解锁一次。

17.8. 看门狗操作说明

1) 配置 SCCM0，使能看门狗时钟；

Copyright © 2022 武汉瑞纳捷半导体有限公司

2) 配置 SCRSTEN 使能看门狗复位芯片；
3) 配置使能看门狗中断 IEN3.WDT_EN；
4) 开启总中断；
5) 配置 WDT_LOCK_0 和 WDT_LOCK_1 为解锁值；
6) 配置加载寄存器 WDTREL，设置计数时钟周期以及加载值；
7) 配置 WDT_LOCK_0 和 WDT_LOCK_1 为解锁值；

开启看门狗（IEN0.WDT）并激活看门狗定时器（IEN1.SWDT）。

Copyright © 2022 武汉瑞纳捷半导体有限公司

18.AES/SM4/DES 加解密

18.1. 概述

芯片内置有 DES、SM4、AES128 协处理器，支持 DES、3DES、SM4、AES128 算法，实现数据的加解密。

18.2. 主要特性

- DES 符合 FIPS 46-3 的加解密标准
- DES 支持 64bit 密钥，3DES 支持 192bit 密钥
- DES 支持 ECB(Electronic Code Book)模式
- AES 支持 128bit 密钥
- SM4 支持 128bit 密钥

18.3. 加解密算法寄存器

加解密算法寄存器基地址：0x2100

偏移	名称	描述
0x00	SM4_DES_CON	控制寄存器
0x01	SM4_DES_STS	状态寄存器
0x02	SM4_DES_MSK	中断控制寄存器
0x03	SM4_DES_DAT0_0	数据寄存器
0x04	SM4_DES_DAT0_1	数据寄存器
0x05	SM4_DES_DAT0_2	数据寄存器
0x06	SM4_DES_DAT0_3	数据寄存器
0x07	SM4_DES_DAT1_0	数据寄存器
0x08	SM4_DES_DAT1_1	数据寄存器
0x09	SM4_DES_DAT1_2	数据寄存器
0x0A	SM4_DES_DAT1_3	数据寄存器
0x0B	SM4_DES_DAT2_0	数据寄存器
0x0C	SM4_DES_DAT2_1	数据寄存器
0x0D	SM4_DES_DAT2_2	数据寄存器
0x0E	SM4_DES_DAT2_3	数据寄存器
0x0F	SM4_DES_DAT3_0	数据寄存器
0x10	SM4_DES_DAT3_1	数据寄存器
0x11	SM4_DES_DAT3_2	数据寄存器
0x12	SM4_DES_DAT3_3	数据寄存器
0x13	SM4_DES_KEY0_0	密钥寄存器

Copyright © 2022 武汉瑞纳捷半导体有限公司

0x14	SM4_DES_KEY0_1	密钥寄存器
0x15	SM4_DES_KEY0_2	密钥寄存器
0x16	SM4_DES_KEY0_3	密钥寄存器
0x17	SM4_DES_KEY1_0	密钥寄存器
0x18	SM4_DES_KEY1_1	密钥寄存器
0x19	SM4_DES_KEY1_2	密钥寄存器
0x1A	SM4_DES_KEY1_3	密钥寄存器
0x1B	SM4_DES_KEY2_0	密钥寄存器
0x1C	SM4_DES_KEY2_1	密钥寄存器
0x1D	SM4_DES_KEY2_2	密钥寄存器
0x1E	SM4_DES_KEY2_3	密钥寄存器
0x1F	SM4_DES_KEY3_0	密钥寄存器
0x20	SM4_DES_KEY3_1	密钥寄存器
0x21	SM4_DES_KEY3_2	密钥寄存器
0x22	SM4_DES_KEY3_3	密钥寄存器
0x23	SM4_DES_KEY4_0	密钥寄存器
0x24	SM4_DES_KEY4_1	密钥寄存器
0x25	SM4_DES_KEY4_2	密钥寄存器
0x26	SM4_DES_KEY4_3	密钥寄存器
0x27	SM4_DES_KEY5_0	密钥寄存器
0x28	SM4_DES_KEY5_1	密钥寄存器
0x29	SM4_DES_KEY5_2	密钥寄存器
0x2A	SM4_DES_KEY5_3	密钥寄存器

18.3.1. SM4_DES_CON 控制寄存器(偏移：00H)

比特	名称	属性	复位值	描述
7:4	RSV	-	-	保留
3:2	MODE	R/W	0x0	算法选择： 00：AES128 01：DES 10：3DES 11：SM4
1	ENDE_REG	R/W	0x0	运算方式控制位 1：加密 0：解密
0	START	R/W	0x0	启动控制位 0：停止 1：启动。软件置 1，加解密结束时硬件自动清零

18.3.2. SM4_DES_STS 状态寄存器(偏移：01H)

Copyright © 2022 武汉瑞纳捷半导体有限公司

比特	名称	属性	复位值	描述
7:1	RSV	-	-	保留
0	STS	R/W	0x0	加解密完成标志 0：未完成 1：运算完成，硬件置 1，软件写清零。

18.3.3. SM4_DES_MSK 中断控制寄存器(偏移：02H)

比特	名称	属性	复位值	描述
7:1	RSV	-	-	保留
0	MSK	R/W	0x0	屏蔽加解密中断控制位 1：屏蔽 0：不屏蔽

18.3.4. SM4_DES_DAT0_X 数据寄存器 32 位(偏移：03H)

比特	名称	属性	复位值	描述
31:24	SM4_DES_DAT0_3	R/W	0x00	1、SM4/AES128/DES/3DES 输入数据寄存器 2、加密/解密输出的数据
23:16	SM4_DES_DAT0_2	R/W	0x00	
15:8	SM4_DES_DAT0_1	R/W	0x00	
7:0	SM4_DES_DAT0_0	R/W	0x00	

18.3.5. SM4_DES_DAT1_X 数据寄存器 32 位(偏移：07H)

比特	名称	属性	复位值	描述
31:24	SM4_DES_DAT1_3	R/W	0x00	1、SM4/AES128/DES/3DES 输入数据寄存器 2、加密/解密输出的数据
23:16	SM4_DES_DAT1_2	R/W	0x00	
15:8	SM4_DES_DAT1_1	R/W	0x00	
7:0	SM4_DES_DAT1_0	R/W	0x00	

18.3.6. SM4_DES_DAT2_X 数据寄存器 32 位(偏移：0BH)

比特	名称	属性	复位值	描述
31:24	SM4_DES_DAT0_3	R/W	0x00	1、SM4/AES128 输入数据寄存器 2、加密/解密输出的数据
23:16	SM4_DES_DAT0_2	R/W	0x00	
15:8	SM4_DES_DAT0_1	R/W	0x00	
7:0	SM4_DES_DAT0_0	R/W	0x00	

18.3.7. SM4_DES_DAT3_X 数据寄存器 32 位(偏移：0FH)

Copyright © 2022 武汉瑞纳捷半导体有限公司

比特	名称	属性	复位值	描述
31:24	SM4_DES_DAT0_3	R/W	0x00	1、SM4/AES128 输入数据寄存器 2、加密/解密输出的数据
23:16	SM4_DES_DAT0_2	R/W	0x00	
15:8	SM4_DES_DAT0_1	R/W	0x00	
7:0	SM4_DES_DAT0_0	R/W	0x00	

18.3.8. SM4_DES_KEY0_X 密钥寄存器 32 位(偏移：13H)

比特	名称	属性	复位值	描述
31:24	SM4_DES_KEY0_3	R/W	0x00	1、SM4/AES128/DES/3DES 密钥输入数据寄存器
23:16	SM4_DES_KEY0_2	R/W	0x00	
15:8	SM4_DES_KEY0_1	R/W	0x00	
7:0	SM4_DES_KEY0_0	R/W	0x00	

18.3.9. SM4_DES_KEY1_X 密钥寄存器 32 位(偏移：17H)

比特	名称	属性	复位值	描述
31:24	SM4_DES_KEY1_3	R/W	0x00	1、SM4/AES128/DES/3DES 密钥输入数据寄存器
23:16	SM4_DES_KEY1_2	R/W	0x00	
15:8	SM4_DES_KEY1_1	R/W	0x00	
7:0	SM4_DES_KEY1_0	R/W	0x00	

18.3.10. SM4_DES_KEY2_X 密钥寄存器 32 位(偏移：1BH)

比特	名称	属性	复位值	描述
31:24	SM4_DES_KEY2_3	R/W	0x00	1、SM4/AES128/3DES 密钥输入数据寄存器
23:16	SM4_DES_KEY2_2	R/W	0x00	
15:8	SM4_DES_KEY2_1	R/W	0x00	
7:0	SM4_DES_KEY2_0	R/W	0x00	

18.3.11. SM4_DES_KEY3_X 密钥寄存器 32 位(偏移：1FH)

比特	名称	属性	复位值	描述
31:24	SM4_DES_KEY3_3	R/W	0x00	1、SM4/AES128/3DES 密钥输入数据寄存器

Copyright © 2022 武汉瑞纳捷半导体有限公司

23:16	SM4_DES_KEY3_2	R/W	0x00	
15:8	SM4_DES_KEY3_1	R/W	0x00	
7:0	SM4_DES_KEY3_0	R/W	0x00	

18.3.12. SM4_DES_KEY4_X 密钥寄存器 32 位(偏移：23H)

比特	名称	属性	复位值	描述
31:24	SM4_DES_KEY4_3	R/W	0x00	1、3DES 密钥输入数据寄存器
23:16	SM4_DES_KEY4_2	R/W	0x00	
15:8	SM4_DES_KEY4_1	R/W	0x00	
7:0	SM4_DES_KEY4_0	R/W	0x00	

18.3.13. SM4_DES_KEY5_X 密钥寄存器 32 位(偏移：27H)

比特	名称	属性	复位值	描述
31:24	SM4_DES_KEY5_3	R/W	0x00	1、3DES 密钥输入数据寄存器
23:16	SM4_DES_KEY5_2	R/W	0x00	
15:8	SM4_DES_KEY5_1	R/W	0x00	
7:0	SM4_DES_KEY5_0	R/W	0x00	

18.4. 加解密算法操作说明

18.4.1. 初始化流程

1) 配置 SCCM2 寄存器，使能加解密算法时钟
2) 使能加解密算法中断和总中断
3) 配置 SM4_DES_MSK 寄存器，选择是否屏蔽 DES 和 SM4 中断，

若采用中断方式，则不屏蔽中断，若采用查询方式则屏蔽中断且第 2 步可以跳过。

18.4.2. 算法加解密流程

1) 写入待加密或解密的数据到寄存器 SM4_DES_DAT0_X 至 SM4_DES_DAT3_X（X=0,1,2,3。下同）
   - DES/3DES 对应 SM4_DES_DAT0_X ~ SM4_DES_DAT1_X；
   - SM4/AES 对应 SM4_DES_DAT0_X ~ SM4_DES_DAT3_X；
2) 写入相应的密钥数据到寄存器 SM4_DES_KEY0_X 至 SM4_DES_KEY5_X
   - DES 对应 SM4_DES_KEY0_X ~ SM4_DES_KEY1_X；

Copyright © 2022 武汉瑞纳捷半导体有限公司

- 3DES 对应 SM4_DES_KEY0_X ~ SM4_DES_KEY5_X;
- SM4/AES 对应 SM4_DES_KEY0_X ~ SM4_DES_KEY3_X；

3) 写寄存器 SM4_DES_CON 的第 0,1,2,3 比特位配置加解密模式
4) 查询 SM4_DES_STS 寄存器，等待加解密完成（若采用中断方式，则在加解密完成后会产生中断）。
5) 查到 SM4_DES_STS 寄存器标志位（或进入中断）后，清除标志位。然后读数据寄存器 SM4_DES_DAT0_X 至 SM4_DES_DAT3_X
   - DES/3DES 对应 SM4_DES_DAT0_X~SM4_DES_DAT1_X；
   - SM4/AES 对应 SM4_DES_DAT0_X ~ SM4_DES_DAT3_X；

注：若要改变 SM4_DES_MSK 中断的值，建议在写入密钥数据之前执行。

需加解数据和密钥的写入与读取，都是高字节数据在后，低字节数据在前，依次填入。例如：0x11,0x12,0x13,0x14。

SM4_DES_DAT0_0=0x14，SM4_DES_DAT0_1=0x13，

SM4_DES_DAT0_2=0x12，SM4_DES_DAT0_3=0x11。

Copyright © 2022 武汉瑞纳捷半导体有限公司

19.乘除法器（MDU）

19.1. 概述

芯片集成一个算术协处理器，使高性能的 8051 执行额外的扩展算术运算。

19.2. 主要特性

- 支持 32 位除法
- 支持 16 位乘法
- 移位和标准化操作

19.3. MDU 计算操作描述

19.3.1. 16bit*16bit 乘法运算

将乘数写入寄存器 MD0，MD1（高位在后，低位在前，下同），将被乘数写入寄存器 MD4，MD5。等待计算出结果，读取寄存器 MD0/MD1/MD2/MD3（MD0 为低字节，MD3 为高字节）。

19.3.2. 16bit/16bit 除法运算

将被除数写入寄存器 MD0，MD1；将除数写入寄存器 MD4，MD5。等待计算出结果，商存储在寄存器 MD0（低 8 位），MD1（高 8 位）；余数存储在寄存器 MD4（低 8 位）、MD5（高 8 位）。

19.3.3. 32bit/16bit 除法运算

将被除数写入寄存器 MD0，MD1，MD2，MD3；将除数写入寄存器 MD4，MD5。等待计算出结果，商存储在寄存器 MD0（低字节），MD1，MD2，MD3（最高字节）；余数存储在寄存器 MD4（低 8 位）、MD5（高 8 位）。

19.3.4. 移位操作运算

将需要移位的数据写入寄存器 MD0，MD1，MD2，MD3（高位在后，低位在前）；配置 ARCON 寄存器的第 5 比特位选择左移或者右移；配置 ARCON 的 0,1,2,3,4 比特位确定移动位数。等待计算结果，读取寄存器 MD0（最低字节），MD1，MD2，MD3（最高字节）。

例：16bit*16bit 以 0x1234*0x5678=0x6260060 为例

乘数写入寄存器 MD0=0x34，MD1=0x12；被乘数写入寄存器 MD4=0x78，MD5=0x56

等待结果，读取寄存器 MD0=0x60，MD1=0x00，MD2=0x26，MD3=0x06。

19.4. MDU 寄存器

Copyright © 2022 武汉瑞纳捷半导体有限公司

MDU 特殊功能寄存器

寄存器	地址	描述	复位值
MD0	E9H	乘法/乘法数据寄存器	00H
MD1	EAH		00H
MD2	EBH		00H
MD3	ECH		00H
MD4	EDH		00H
MD5	EEH		00H
ARCON	EFH	运算控制寄存器	00H

19.4.1. 运算控制寄存器 ARCON（SFR 地址：EFH）

比特	名称	属性	功能
7	MDEF	R/W	MDU 错误标志
6	MDOV	R/W	MDU 溢出标志
5	SLR	R/W	位移方向 0：左移操作 1：右移操作
4	SC.4	R/W	移位计数器： 当这些 bit 位都设置为“0”时，就选择了标准化操作。标准化之后，这些位就包含了执行了标准移位的数量。当至少有一个 bit 位设为“1”时，就选择了移位操作。执行移位的数量由写入这些 bit 位的值来决定。
3	SC.3	R/W	
2	SC.2	R/W	
1	SC.1	R/W	
0	SC.0	R/W	

19.5. MDU 计算操作说明

19.5.1. 16bit*16bit 乘法运算

1) 写入乘数 MD0（低 8 位）和 MD1（高 8 位）；
2) 写入被乘数 MD4（低 8 位）和 MD5（高 8 位）；
3) 软件延时约 20us（系统 16M 时钟下）；
4) 读出计算结果，即读取：MD0（最低字节）、MD1、MD2 和 MD3（最高字节）；

19.5.2. 16bit/16bit 除法运算

1) 写入除数 MD0（低 8 位）和 MD1（高 8 位）；
2) 写入被除数 MD4（低 8 位）和 MD5（高 8 位）；
3) 软件延时约 20us（系统 16M 时钟下）；

Copyright © 2022 武汉瑞纳捷半导体有限公司

4) 读出计算结果，即读取商：MD0（低 8 位），MD1（高 8 位）
余数：MD4（低 8 位），MD5（高 8 位）

19.5.3. 32bit/16bit 除法运算

1) 写入除数 MD0（最低字节）、MD1、MD2 和 MD3（最高字节）；
2) 写入被除数 MD4（低 8 位）和 MD5（高 8 位）；
3) 软件延时约 100us（系统 16M 时钟下）；
4) 读出计算结果，即读取商：MD0(最低字节)、MD1、MD2 和 MD3(最高字节)
余数：MD4（低 8 位），MD5（高 8 位）

19.5.4. 移位运算

1) 写入移位操作数 MD0（最低字节）、MD1、MD2 和 MD3（最高字节）；
2) 配置 ARCON 寄存器，第五位 SLR，选择左移或右移操作；
3) 配置 ARCON 寄存器，SC0~SC4，选择移动位数；
4) 软件延时约 100us（系统 16M 时钟下）；
5) 读出计算结果，即读取：MD0（最低字节）、MD1、MD2 和 MD3（最高字节）；

Copyright © 2022 武汉瑞纳捷半导体有限公司

20.随机数发生器（RNG）

20.1. 概述

芯片内嵌 1 个 8 位随机数发生器（RNG），用以满足某些应用中的安全交易流程。

20.2. 随机数寄存器

RNG 基址：0x2068

偏移	寄存器	描述
00H	DTRNGCON	RNG 控制寄存器
01H	DTRNGSTS	RNG 状态寄存器
02H	DTRNGDAT	RNG 数据寄存器

20.2.1. RNG 控制寄存器 DTRNGCON（偏移：00H）

比特	名称	属性	功能
7:2	-	-	保留
1	MODEL_SEL	R/W	0：选择 RNG 工作于 model0； 1：选择 RNG 工作于 model1；
0	TRNG_START	R/W	1：关闭 RNG； 0：启动 RNG；

20.2.2. RNG 数据寄存器 DTRNGDAT（偏移：01H）

比特	名称	属性	功能
7:0	TRNG_DATA	R/W	数据寄存器

20.2.3. RNG 状态寄存器 DTRNGSTS（偏移：02H）

比特	名称	属性	功能
7:1	-	-	保留
0	TRNG_STS	R/W	0：随机数生产未开始或未完成 1：随机数生成完成

20.3. RNG 操作流程

1) 配置 RNG 时钟（SCCM1. RNG_EN）使能；
2) 配置 RNG 模式（DTRNGCON.MODEL_SEL）选择；
3) 开启 RNG 模块（DTRNGCON.TRNG_START）；
4) 判断完成标志（DTRNGSTS.TRNG_STS），并读取数据（DTRNGDAT）。

Copyright © 2022 武汉瑞纳捷半导体有限公司

21.CRC8 计算单元

21.1. 概述

CRC（Cyclic Redundancy Check）即循环冗余校验码，是根据固定的生成多项式得到任一 8 位的 CRC 计算结果，用于校验数据的完整性。

21.2. 主要特性

- 使用 CRC-8 多项式：0x07
  - x^8+x^2+x+1
- 一个 8 位的数据输入/输出寄存器
- CRC 计算时间：X 个时钟周期

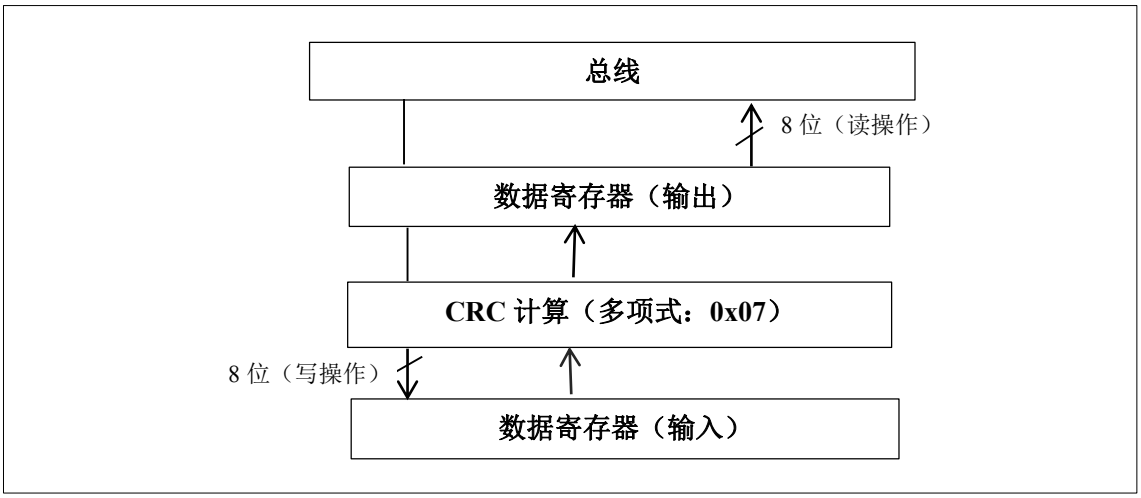


图 20-1　CRC 计算单元框图

21.3. CRC8 功能描述

CRC 计算单元含有 1 个 8 位数据寄存器：

- 对该寄存器进行写操作时，作为输入寄存器，可以输入要进行 CRC 计算的新数据。
- 对该寄存器进行读操作时，返回上一次 CRC 计算的结果。

21.4. CRC8 计算单元寄存器

CRC 寄存器基址：0x2000

偏移	名称	描述
0x00	CRC8_DATA	CRC8 数据寄存器
0x01	CRC8_CTRL	CRC8 控制寄存器
0x02	CRC8_INIT	CRC8 初始值配置寄存器

Copyright © 2022 武汉瑞纳捷半导体有限公司

21.4.1. 数据寄存器 CRC8_DATA（偏移：00H）

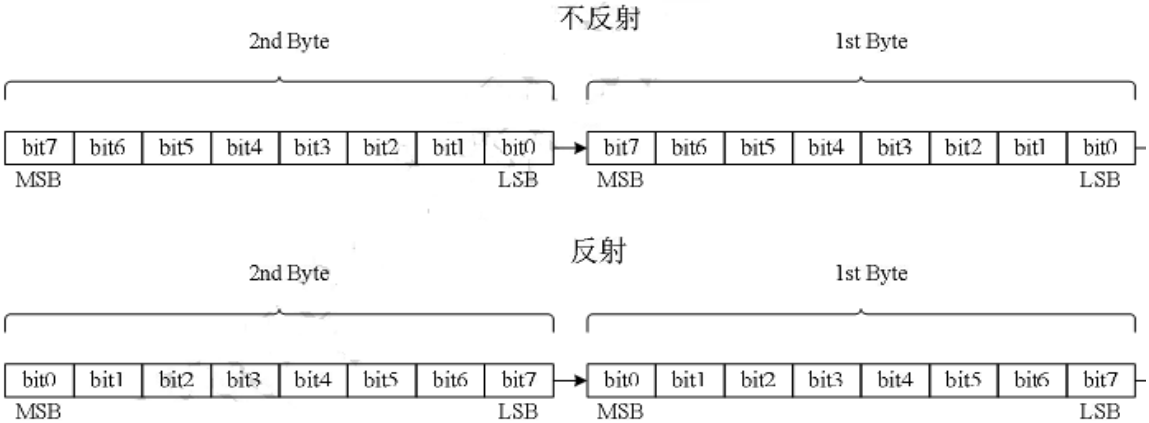
比特	名称	属性	复位值	描述
7:0	DATA	RW	0xFF	写：写入需要进行 CRC 校验计算的数据，如需要校验的数据不止一个 byte 需按顺序逐次写入 读：读出 8 位 CRC 计算结果

注：读取完计算结果后，如需再次计算 CRC，必须重新初始化 CRC8_INIT

21.4.2. 控制寄存器 CRC8_CTRL（偏移：01H）

比特	名称	属性	复位值	描述
7:2	RSV	-	-	保留
1	RSLT_REV	RW	0x0	CRC 计算结果是否进行高低位倒序 0：不倒序 1：倒序
0	DATA_REV	RW	0x0	CRC 计算数据是否进行高低位倒序 0：不倒序 1：倒序

输入数据与输出结果反射说明如下：



21.4.3. 初始值配置寄存器 CRC8_INIT（偏移：02H）

比特	名称	属性	复位值	描述
7:0	CRC_INIT	RW	0xFF	8 位 CRC 初始值

21.5. CRC8 计算单元操作说明

1) 设置 8 位初始值 CRC8_INIT；
2) 设置 CRC8_CTRL，选择输入数据和结输出结果是否反射；
3) 向 CRC8_DATA 中写入 8 位 CRC 计算数据，如需要校验的数据不止一个 byte 需按顺序逐次写入，直至输入完成；
4) 读 CRC8_DATA，将一次返回 CRC 计算结果。

Copyright © 2022 武汉瑞纳捷半导体有限公司

22.I2C 接口

22.1. 概述

芯片有 2 路 I2C 接口：I2C0 与 I2C1，通过数据引脚 SDA 和时钟引脚 SCL 与 I2C 外设进行通信。本模块支持主模式和从模式。

22.2. 主要特性

- I2C 主/从设备功能；
- 可编程的 I2C 从设备地址；
- 可编程的 NACK/ACK 回复；
- 主模式支持 100Kbps、400Kbps 两种速率；
- 从模式支持 100Kbps 速率
- 支持 7bit 设备地址；
- 独立的发送 FIFO（TXFIFO）和接收 FIFO（RXFIFO），深度均为 4B；
- 不支持主机仲裁模式；

22.3. 寄存器描述

寄存器基地址：（I2C0：0x2170；I2C1：0x2050）

偏移	名称	描述
0x00	I2C_CLK_DIV	I2C时钟分频寄存器
0x01	I2C_CR	I2C控制寄存器
0x02	I2C_INT_EN	I2C中断使能寄存器
0x03	I2C_SR1	I2C状态寄存器1
0x04	I2C_SR2	I2C状态寄存器2
0x05	I2C_DR	I2C数据寄存器
0x06	I2C_SLAVE_ADDR1	I2C从设备地址寄存器1
0x07	I2C_SLAVE_ADDR2	I2C从设备地址寄存器2
0x08	I2C_TIMEOUT_CNT	I2C TIMEOUT计数寄存器

22.2.1. I2C 主机时钟分频寄存器 I2C_CLK_DIV（偏移：00H）

比特	名称	属性	复位值	描述
7:0	I2C_CLK_DIV	RW	0x80	I2C 时钟分频值，只在主机模式时设置。 Fscl = （Fpclk）/（4*（I2C_CLK_DIV +1）） ***注：Fpclk 为 AHB 时钟频率，和系统时钟频率一致；*** 2、I2C_CLK_DIV 的值必须大于 1

22.2.2. I2C 控制寄存器 I2C_CR（偏移：01H）

比特	名称	属性	复位值	描述
7	TIMER_INIT_I2C	RW	0x0	发生TIMEOUT事件后，初始化I2C选择位 0：不初始化I2C 1：初始化I2C； 初始化状态机和FIFO
6	TIMEOUT_EN	RW	0x0	TIMEOUT使能位 0：不使能 1：使能； 软件置1，TIMEOUT后硬件清0
5	CLEAR_FIFO	RW	0x0	FIFO清除控制位 0：不清空FIFO 1：清空FIFO
4	TACK	RW	0x0	传输应答位 0：接收一字节后，在应答周期产生ACK； 1：接收一字节后，在应答周期产生NACK； ***注:TACK必须在应答周期前写入。***
3	STOP	RW	0x0	结束条件产生位 0：发送完当前字节不产生结束条件； 1：主设备在发送完当前字节后，将产生结束条件。产生结束条件后，硬件自动清0
2	START	RW	0x0	起始条件产生位 0：主模式下不产生起始条件 1：主模式下产生起始条件 ***注：空闲时刻和NACK/ACK应答后才可产生起始条件，起始条件产生后，硬件自动清0，I2C_SR的STARTF位置1。***
1	TX_RX_SEL	RW	0x0	发送接收选择位 0：主设备作为接收器 1：主设备作为发送器 ***注：主设备默认START条件后的第一个字节为发送***
0	MASTER	RW	0x0	主从设备选择位 0：从模式 1：主模式

22.2.3. I2C 中断使能寄存器 I2C_INT_EN（偏移：02H）

Copyright © 2022 武汉瑞纳捷半导体有限公司

比特	名称	属性	复位值	描述
7	SCL_TIMEOUT_INT_EN	RW	0x0	SCL超时中断使能 0：SCL超时中断不使能； 1：SCL超时中断使能；
6	SDA_TIMEOUT_INT_EN	RW	0x0	SDA超时中断使能 0：SDA超时中断不使能； 1：SDA超时中断使能；
5	RX_ADDR2_INT_EN	RW	0x0	ADDR2地址匹配中断使能 0：RX_ADDR2中断不使能； 1：RX_ADDR2中断使能；
4	RX_ADDR1_INT_EN	RW	0x0	ADDR1地址匹配中断使能 0：RX_ADDR1中断不使能； 1：RX_ADDR1中断使能；
3	MTF_INT_EN	RW	0x0	字节传输完成中断使能 0：MTF中断不使能； 1：MTF中断使能；
2	OVR_INT_EN	RW	0x0	读溢出/写下溢中断使能 0：OVR中断不使能； 1：OVR中断使能；
1	TXFIFO_NEMPTY_INT_EN	RW	0x0	TXFIFO非空中断使能选择位 0：TXFIFO非空中断不使能； 1：TXFIFO非空中断使能；
0	RXFIFO_NEMPTY_INT_EN	RW	0x0	RXFIFO非空中断使能选择位 0：RXFIFO非空中断不使能； 1：RXFIFO非空中断使能；

22.2.4. I2C 状态寄存器器 I2C_SR1（偏移：03H）

比特	名称	属性	复位值	描述
7	RSV	-	-	保留
6	TXFIFO_FULL	RO	0x0	TXFIFO满标志 0：未发生TXFIFO满 1：发生TXFIFO满
5	RXFIFO_FULL	RO	0x0	RXFIFO满标志 0：未发生RXFIFO满 1：发生RXFIFO满

Copyright © 2022 武汉瑞纳捷半导体有限公司

4	BUS_BUSY	RO	0x0	总线忙碌状态位 0：总线上无数据通信（检测到总线上的结束条件，此位清0） 1：总线上正在进行数据通信（检测到总线上的起始条件，此位置1）
3	SRW	RO	0x0	从机读状态指示位 0：作为从设备接收器 1：作为从设备发送器 当地址匹配后，SRW指示地址字节中的RW位，该位仅在如下条件有效：一个完整的传输已经发生，没有其他传输被初始化；并且I2C被配置为从模式，且从地址匹配。当接收到停止条件或一个新的起始条件，该位自动清除
2	STOPF	RW	0x0	主机STOP条件检测位 1：检测到停止位 0：未检测到停止位 检测到start信号和写1清0
1	STARTF	RW	0x0	主机起始条件发送状态位 1：起始条件已发送 0：起始条件未发送 写1清0 ***注：I2C_DR寄存器有数据将自动启动发送时序并清除该位，主机模式下，SDA方向不由硬件切换并处于接收状态时STARTF也将被清除。***
0	RACK	RO	0x0	应答接收状态位 0：最近的发送应答周期接收到ACK 1：最近的发送应答周期接收到NACK 只有START条件将清除RACK位 ***注：主机发送模式下收到NACK停止发送，等待软件处理事件***

22.2.5. I2C 状态寄存器 I2C_SR2（偏移：04H）

比特	名称	属性	复位值	描述
7	SDA_TIMEOUT	RO	0x0	SDA TIMEOUT状态位 0：SDA未发生超时溢出； 1：SDA发生超时溢出
6	SCL_TIMEOUT	RO	0x0	SCL TIMEOUT状态位 0：SCL未发生超时溢出；

Copyright © 2022 武汉瑞纳捷半导体有限公司

				1：SCL发生超时溢出
5	RX_ADDR2	RW	0x0	从设备地址2匹配状态位 0：设备地址2和接收到的地址不相等 1：设备地址2和接收到的地址相等 写1清0 ***注：当地址匹配时，SRW位表示了地址字节的RW位。***
4	RX_ADDR1	RW	0x0	从设备地址1匹配状态位 0：设备地址1和接收到的地址不相等 1：设备地址1和接收到的地址相等 写1清0 ***注：当地址匹配时，SRW位表示了地址字节的RW位。***
3	MTF	RW	0x0	字节传输完成状态位 0：字节传输未完成 1：字节传输完成 当一个字节数据（包括地址）正在传输时，该位为0；在一个字节传输完后，在第9个SCL时钟下降沿（应答周期）MTF被置为1。写1清0
2	OVER	RW	0x0	读溢出/写下溢状态位 0：FIFO未发生溢出 1：FIFO发生溢出 硬件置1，软件写1清0,FIFO溢出不影响FIFO中以写入的数据，溢出数据丢失
1	TXFIFO_EMPTY	RO	0x1	TXFIFO空标志 0：未发生TXFIFO空 1：发生TXFIFO空
0	RXFIFO_NEMPTY	RO	0x0	RXFIFO非空标志 0：未发生RXFIFO非空 1：发生RXFIFO非空

22.2.6. I2C 数据寄存器 I2C_DR（偏移：05H）

比特	名称	属性	复位值	描述
7:0	I2C_DR	RW	0x0	I2C数据寄存器

注：写 I2C_DR 寄存器为往发送 FIFO 中写数据，读 I2C_DR 寄存器为读接收 FIFO。

22.2.7. I2C 从设备地址寄存器 I2C_SLAVE_ADDR1（偏移：06H）

比特	名称	属性	复位值	描述

Copyright © 2022 武汉瑞纳捷半导体有限公司

7:1	ADDR1[7:1]	RW	0x0	ADDR1地址的7~1位
0	RSV	-	-	保留

22.2.8. I2C 从设备地址寄存器 I2C_SLAVE_ADDR2（偏移：07H）

比特	名称	属性	复位值	描述
7:1	ADDR2[7:1]	RW	0x0	ADDR2地址的7~1位
0	ADDR2_EN	RW	0x0	0：SLAVE_ADDR2地址匹配不使能 1：SLAVE_ADDR2地址匹配使能

22.2.9. I2C TIME OUT 计数寄存器 I2C_TIME_OUT_CNT（偏移：1CH）

比特	名称	属性	复位值	描述
7:0	TIMEOUT_CNT	RW	0x20	TIMEOUT周期为： TIMEOUT_CNT*256个系统时钟

22.3. I2C 功能描述

22.3.1. 模式选择

该模块默认为从模式，当软件配置 I2C_CR 寄存器 MASTER 位为 1 后设备变成主模式。主模式时，写 I2C 寄存器 START 位，I2C 接口启动数据传输并产生时钟信号，并可以发出 STOP 信号停止传输。从模式时，I2C 接口能识别它自己的地址（7 位）。数据和地址按 8 位/字节进行传输，高位在前。跟在起始条件后的是地址。地址只在主模式发送。在一个字节传输的 8 个时钟后的第 9 个时钟期间，接收器必须回送一个应答位（ACK）给发送器。参考下图：

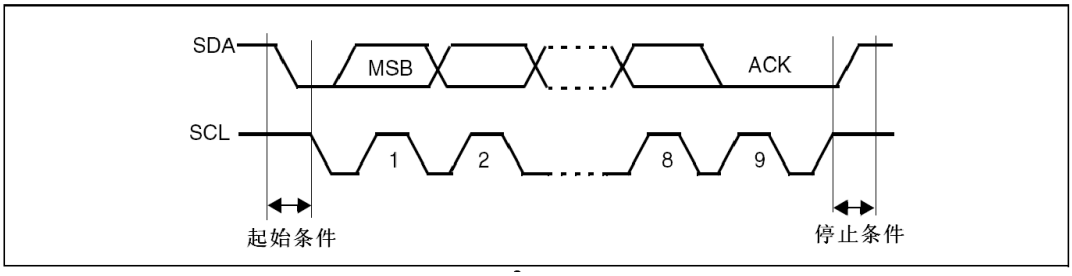


图19-1 I2C字节格式

22.3.2. I2C 从模式

一旦检测到起始条件，在 SDA 线上接收到的地址被送到移位寄存器。然后与芯片自己的设备地址相比较，如果地址不匹配 I2C 将其忽略并等待另一个起始条件。如果地址匹配，则此控制器会检测当前操作是发送还是接收（SRW 寄存器），I2C 接口进行如下操作：

从发送器：

- 发送器将字节从 FIFO 加载到内部移位寄存器发送到 SDA 线上，加载数据完成

后如果此时 FIFO 中无数据将产生发送 FIFO 空标志，写入数据到 FIFO 中可以清除该状态。

- 如果加载数据时，FIFO 中无数据，将产生 OVER 溢出状态，状态需要写 1 清 0。

从接收器：

- 在接收到数据后，从接收器将通过内部移位寄存器从 SDA 线接收到的字节存储到 FIFO 中，如果 FIFO 之前无数据，FIFO 接收数据后并产生 FIFO 非空标志，软件需要读出数据寄存器的值来清除 FIFO 非空标志。
- I2C 接口在接收到每个字节后都产生一个应答脉冲。
- 如果在接收新数据前数据 FIFO 数据已满，即 FIFO 满状态仍然为 1，接收完当前字节后将产生 OVER 溢出状态，状态需要写 1 清 0。

从机时序图：

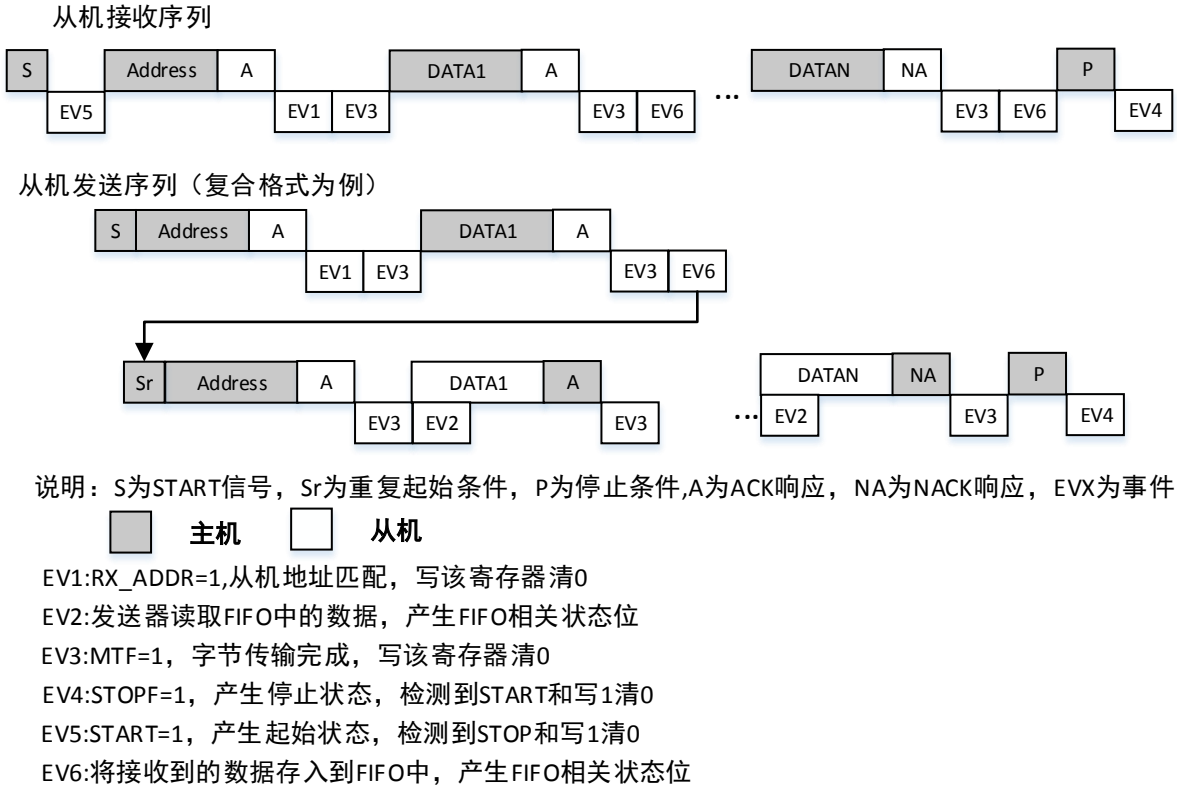


图 19-2 I2C 从机时序说明图

22.3.3. I2C 主模式

在主模式时，I2C 接口启动数据传输并产生时钟信号。串行数据传输总是以起始条件开始并以停止条件结束。当软件配置 I2C_CR 寄存器 MASTER 位为 1 后设备变成主模式。

以下是主模式的操作顺序：

- 配置时钟控制寄存器。
- 配置数据寄存器（从设备的地址和读写控制位）。
- 配置控制寄存器的 MASTER 位为 1，产生起始条件。

Copyright © 2022 武汉瑞纳捷半导体有限公司

主发送器：

发送了地址后，主设备通过内部移位寄存器将字节从数据寄存器load到SDA线上，并产生 FIFO 状态，如若产生发送 FIFO 空状态，主机需要写寄存器消除它。

收到应答脉冲后确认新的数据已经发送到数据寄存器。如果在下一个数据发送结束之前新数据仍然没有被写进数据寄存器，即发送 FIFO 为空，这时 I2C 接口保持 SCL 为低以等待新的数据被写进数据寄存器。

当发送完成倒数第二个字节的时候配置 STOP 信号，等待最后一个字节传输完成后主设备发出 STOP 信号产生停止条件。

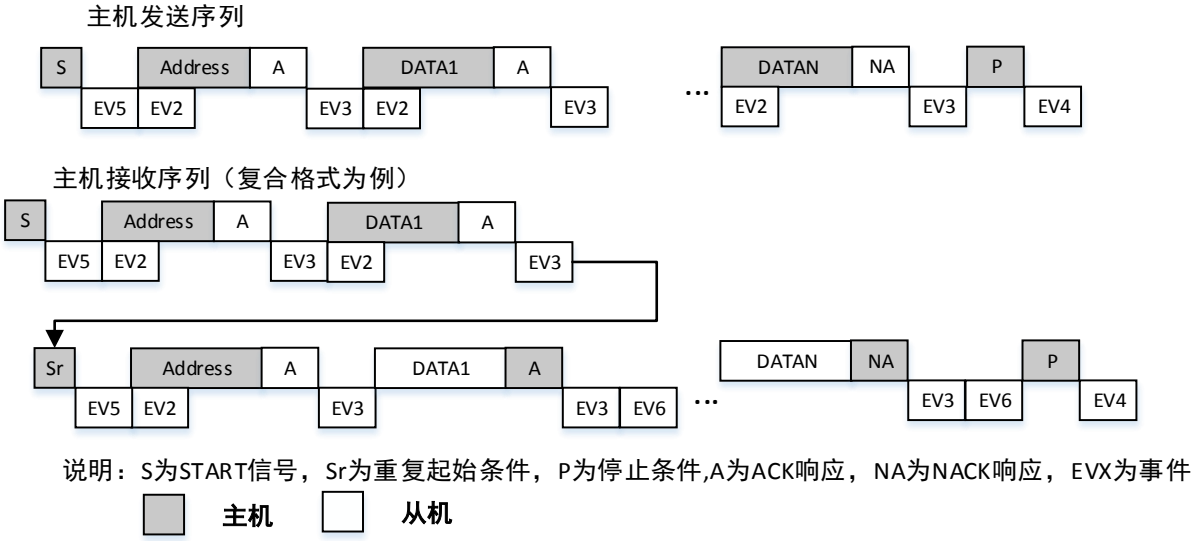
主接收器：

发送了地址后，I2C 接口从 SDA 线接收数据字节，并通过内部移位寄存器存到到 FIFO 中，产生数据 FIFO 状态，如若产生接收 FIFO 非空状态，主机需要读寄存器消除它。

如果在接收新数据前数据 FIFO 产生满状态，这时 I2C 接口保持 SCL 为低以等待数据寄存器的值被读出。

当接收完成倒数第二个字节的时候配置 STOP 和 NACK 信号，等待最后一个字节传输完成后主设备发出 NACK 应答和 STOP 信号产生停止条件。

主机时序图：



EV2:发送器读取FIFO中的数据，产生FIFO相关状态位
EV3:MTF=1，字节传输完成，写该寄存器清0
EV4:STOPF=1，产生停止状态，检测到START和写1清0
EV5:START=1，产生起始状态，检测到STOP和写1清0
EV6:将接收到的数据存入到FIFO中，产生FIFO相关状态位

图19-3 I2C主机时序说明图

22.4. 使用说明

22.4.1. 主机接收

1) 配置 SCCM0 寄存器，使能 I2C 的时钟；

2) 将对应管脚分别复用为 IIC_SDA、IIC_SCL；
3) 配置 I2C_CLK_DIV 时钟分频，同时配置 I2C_CR 的 MASTER 位，设为主模式。
4) 配置 I2C_CR 的 TACK 传输应答位为 ACK，为接收字节确认作准备；
5) 检测 I2C_SR1 中 BUS_BUSY 总线忙状态位，等待总线空闲；
6) 配置 I2C_CR 的 START 位为 1，等待 I2C_SR1 的 STARTF 状态而后清除；
7) 写 I2C_DR 寄存器，发送地址+读控制位（1-读 0-写）组成的数据，等待 I2C_SR2 的 TXFIFO_EMPTY 状态，清除 I2C_SR2 的 MTF 状态并判断 I2C_SR2 的 RACK 是否收到 ACK，以确认是否继续通信；
8) 清除 FIFO，配置 I2C_CR 的 TX_RX_SEL 位，设置主设备作为接收器。
9) 等待 I2C_SR2 的 RXFIFO_NEMPTY 位为非空，读取 I2C_DR，清除 I2C_SR2 的 MTF 状态，如此循环接收直至比目标长度少 1 个字节；
10) 配置 I2C_CR 的 TACK 传输应答位为 NACK，为接收完最后 1 字节的确认作准备；
11) 等待 I2C_SR2 的 RXFIFO_NEMPTY 位为非空，读取 I2C_DR，清除 I2C_SR2 的 MTF 状态，如此接收完最后一个字节；
12) 配置 I2C_CR 的 STOP 位为 1，等待 I2C_SR1 的 STOPF 状态而后清除，结束主机接收。

22.4.2. 主机复合接收

1) 配置 SCCM0 寄存器，使能 I2C 的时钟；
2) 将对应管脚分别复用为 IIC_SDA、IIC_SCL；
3) 配置 I2C_CLK_DIV 时钟分频，同时配置 I2C_CR 的 MASTER 位，设为主模式。
4) 配置 I2C_CR 的 TACK 传输应答位为 ACK，为接收字节确认作准备；
5) 检测 I2C_SR1 中 BUS_BUSY 总线忙状态位，等待总线空闲；
6) 配置 I2C_CR 的 START 位为 1，等待 I2C_SR1 的 STARTF 状态而后清除；
7) 清除 FIFO，配置 I2C_CR 的 TX_RX_SEL 位，设置主设备作为发送器；
8) 写 I2C_DR 寄存器，发送地址+写控制位（1-读 0-写）组成的数据，等待 I2C_SR2 的 TXFIFO_EMPTY 状态，清除 I2C_SR2 的 MTF 状态并判断 I2C_SR2 的 RACK 是否收到 ACK，以确认是否继续通信；
9) 清除 FIFO，写 I2C_DR 寄存器发送数据，等待 I2C_SR2 的 TXFIFO_EMPTY 状态，清除 I2C_SR2 的 MTF 状态并判断 I2C_SR2 的 RACK 是否收到 ACK，以确认是否继续通信；
10) 检测 I2C_SR1 中 BUS_BUSY 总线忙状态位，等待总线空闲；
11) 配置 I2C_CR 的 START 位为 1，等待 I2C_SR1 的 STARTF 状态而后清除；
12) 写 I2C_DR 寄存器，发送地址+读控制位（1-读 0-写）组成的数据，等待 I2C_SR2 的 TXFIFO_EMPTY 状态，清除 I2C_SR2 的 MTF 状态并判断 I2C_SR2 的 RACK

Copyright © 2022 武汉瑞纳捷半导体有限公司

是否收到 ACK，以确认是否继续通信；

13) 清除 FIFO，配置 I2C_CR 的 TX_RX_SEL 位，设置主设备作为接收器。
14) 等待 I2C_SR2 的 RXFIFO_NEMPTY 位为非空，读取 I2C_DR，清除 I2C_SR2 的 MTF 状态，如此循环接收直至比目标长度少 1 个字节；
15) 配置 I2C_CR 的 TACK 传输应答位为 NACK，为接收完最后 1 字节的确认作准备；
16) 等待 I2C_SR2 的 RXFIFO_NEMPTY 位为非空，读取 I2C_DR，清除 I2C_SR2 的 MTF 状态，如此接收完最后一个字节；
17) 配置 I2C_CR 的 STOP 位为 1，等待 I2C_SR1 的 STOPF 状态而后清除，结束主机接收。

22.4.3. 主机发送

1) 配置 SCCM0 寄存器，使能 I2C 的时钟；
2) 将对应管脚分别复用为 IIC_SDA、IIC_SCL；
3) 配置 I2C_CLK_DIV 时钟分频，同时配置 I2C_CR 的 MASTER 位，设为主模式。
4) 检测 I2C_SR1 中 BUS_BUSY 总线忙状态位，等待总线空闲；
5) 配置 I2C_CR 的 START 位为 1，等待 I2C_SR1 的 STARTF 状态而后清除；
6) 清除 FIFO，配置 I2C_CR 的 TX_RX_SEL 位，设置主设备作为发送器；
7) 写 I2C_DR 寄存器，发送地址+写控制位（1-读 0-写）组成的数据，等待 I2C_SR2 的 TXFIFO_EMPTY 状态，清除 I2C_SR2 的 MTF 状态并判断 I2C_SR2 的 RACK 是否收到 ACK，以确认是否继续通信；
8) 清除 FIFO，写 I2C_DR 寄存器发送数据，等待 I2C_SR2 的 TXFIFO_EMPTY 状态，清除 I2C_SR2 的 MTF 状态并判断 I2C_SR2 的 RACK，以确认是否继续通信，如此循环发送直至目标长度字节；
9) 配置 I2C_CR 的 STOP 位为 1，等待 I2C_SR1 的 STOPF 状态而后清除，结束主机发送。

22.4.4. 从机发送模式

1) 配置 SCCM0 寄存器，使能 I2C 的时钟；
2) 将对应管脚分别复用为 IIC_SDA、IIC_SCL；
3) 清除状态寄存器状态，清除 FIFO。配置 I2C_CR 的 MASTER 位，设为从模式。
4) 配置 I2C_SLAVE_ADDR1 和 I2C_SLAVE_ADDR2 寄存器；
5) 等待 I2C_SR2 的 RX_ADDR1 或 RX_ADDR2 是否有效；
6) 地址匹配有效，判断 SRW 位是否为 1。为 0 表示从接收，为 1 表示从发送；
7) 判断为发送后，写第一个要发送的数据；
8) 等待 I2C_SR2 的 TXFIFO_EMPTY 为空，写入发送数据；
9) 等待并清除 I2C_SR2 的 MTF，判断 I2C_SR2 的 RACK；

Copyright © 2022 武汉瑞纳捷半导体有限公司

10) 重复 8）-9），直到接收到 STOP 为止。

22.4.5. 从机接收模式

1) 配置 SCCM0 寄存器，使能 I2C 的时钟；
2) 将对应管脚分别复用为 IIC_SDA、IIC_SCL；
3) 清除状态寄存器状态，清除 FIFO。配置 I2C_CR 的 MASTER 位，设为从模式。
4) 配置 I2C_SLAVE_ADDR1 和 I2C_SLAVE_ADDR2 寄存器；
5) 等待 I2C_SR2 的 RX_ADDR1 或 RX_ADDR2 是否有效；
6) 地址匹配有效，判断 SRW 位是否为 0。为 0 表示从接收，为 1 表示从发送；
7) 判断为接收后，配置 I2C_CR 的 TACK 传输应答位为 ACK，为接收字节确认作准备；
8) 等待 I2C_SR2 的 RXFIFO_NEMPTY 为 1，读出接收数据，并清除 I2C_SR2 的 MTF 状态；
9) 重复 8）-9），直到接收到 STOP 为止；
10) I2C 也可以在一个字节接收完成，并发送完成 (即 MTF=1)，配置 I2C_CR 的 TACK 传输应答位为 NACK，则会在下一个字节字节接收完成后发送 NACK 信号给主机。等待 I2C_SR1 的 BUS_BUSY 为空闲后退出。

Copyright © 2022 武汉瑞纳捷半导体有限公司

23.UART

23.1. 概述

芯片有 4 路 UART 接口：UART0、UART1、UART2 和 UART3。UART 是一种通用串行数据总线，用于异步通信。该总线双向通信，可以实现全双工传输。UART 接口可以工作在查询或中断方式下。

23.2. 主要特性

- 16 位的波特率分频因子；
- 两种工作模式：模式 1、模式 3；
- 可编程数据位宽（8/9bit）；
- 模式 3 下支持多机通信；
- 模式 3 下支持校验控制；
- 支持接收与发送中断；

注：串口中断使能会同时开启接收中断和发送中断，不能单独控制

23.3. UART 功能描述

接口通过两个或三个引脚与其他设备连接在一起，任何 UART 双向通信至少需要两个脚：接收数据输入(UART_RX)和发送数据输出(UART_TX)。

UART 模式下串行数据的发送接收帧结构组成如下：

- 总线在发送或接收前应处于空闲状态
- 一个起始位
- 一个数据字节(8 或 9 位)，最低有效位在前
- 一个停止位，由此表明数据帧的结束

UART 有两种工作模式：模式 1 和模式 3。

工作模式如下表所示：

SM0	SM1	工作方式	功能描述
0	1	模式 1	8 位 UART
1	1	模式 3	9 位 UART

波特率计算公式如下：

$$\text{baud rate} = \frac{Fclk}{(\text{SxRELH} * 256 + \text{SxRELL})}$$

模式 1：

在模式 1 下，10 位的发送或接收。

Copyright © 2022 武汉瑞纳捷半导体有限公司

发送时：一个起始位(总是 0)，8 个数据位(首先是 lsb)和 1 个停止位(总是 1)。

接收时：起始位用来同步数据，8 个数据位可通过读 SxBUF 获得，停止位表明通信结束。若多机通信使能位 SxCON 的 SM20=0，停止位会设置 SxCON 的 RB80 标志位。

模式 3：

在模式 3 下，11 位的发送或接收。一个起始位(总是 0)，8 个数据位(首先是 LSB)，一个可编程的第 9 位和 1 个停止位(总是 1)，第 9 位用来控制奇偶性。

发送时：SxCON 的 TB80 位作为第 9 位输出。

接收时：第 9 位影响 SxCON 的 RB80 位

多机通信：

在模式 3 下可以实现多机通信的功能。通过 SxCON 的 SM20 位来使能。接收中断只有在接收到的第 9 位数据为 1 时才能产生，否则，不会产生接收中断。

为了实现多机通信，所有的从机需要把 SxCON 的 SM20 置 1，主处理器发送一个地址字节来识别目标从处理器。 地址字节不同于数据字节，地址字节的第 9 位是 1，而数据字节的第 9 位是 0。当 SxCON 的 SM20 = 1 时，数据字节不会导致从处理器中断，而地址字节则会中断所有从处理器。从机将接收到的地址和自身地址进行软件匹配，被寻址的从处理器会清 SxCON 的 SM20 位，并准备接收后续的数据。没有被寻址的从处理器保留其 SxCON 的 SM20 的设置，并忽略即将传入的数据。

23.4. UART 寄存器

UART0/1 特殊功能寄存器

偏移	名称	描述
0x98	S0CON	UART0 控制状态寄存器
0x99	S0BUF	UART0 数据缓冲寄存器
0xAA	S0RELL	UART0 重装寄存器低字节
0xBA	S0RELH	UART0 重装寄存器高字节
0x87	PCON	UART0 特殊功能寄存器
0x9B	S1CON	UART1 控制状态寄存器
0x9C	S1BUF	UART1 数据缓冲寄存器
0x9D	S1RELL	UART1 重装寄存器低字节
0xBB	S1RELH	UART1 重装寄存器高字节
0x8F	P1CON	UART1 特殊功能寄存器

UART2/3 寄存器基地址：0x2160

偏移	名称	描述
0x00	S2CON	UART2 控制状态寄存器
0x01	S2RELL	UART2 重装寄存器低字节

Copyright © 2022 武汉瑞纳捷半导体有限公司

0x02	S2RELH	UART2 重装寄存器高字节
0x04	P2CON	UART2 特殊功能寄存器
0x05	S2BUF	UART2 数据缓冲寄存器
0x08	S3CON	UART3 控制状态寄存器
0x09	S3RELL	UART3 重装寄存器低字节
0x0A	S3RELH	UART3 重装寄存器高字节
0x0C	P3CON	UART3 特殊功能寄存器
0x0D	S3BUF	UART3 数据缓冲寄存器

23.4.1. 控制状态寄存器 SxCON

注：此处 x=0~3，下同

比特	名称	属性	复位值	描述
7	SM0	RW	0x0	模式选择 00：保留 01：模式 1 10：保留 11：模式 3
6	SM1	RW	0x0	
5	SM20	RW	0x0	多机通信使能位 0：禁止 1：使能
4	REN0	RW	0x0	串行接收使能位 0：禁止 1：使能
3	TB80	RW	0x0	发送的第 9 位数据 该位只在模式 3 中发送数据时使用(奇偶校验或多机通信)，由软件控制
2	RB80	RW	0x0	接收的第 9 位数据 在模式 1 中，如果 SM20 是 0，该位为停止位 在模式 3 中，该位为接收的第 9 位数据
1	TI0	RW	0x0	发送完成标志位 0：未完成 1：完成 硬件置 1，软件写 1 清 0
0	RI0	RW	0x0	接收完成标志位 0：未完成 1：完成 硬件置 1，软件写 1 清 0

23.4.2. 数据缓冲寄存器 SxBUF

比特	名称	属性	复位值	描述

Copyright © 2022 武汉瑞纳捷半导体有限公司

7:0	SxBUF	RW	0x00	数据缓冲

23.4.3. 重装寄存器低字节 SxRELL

比特	名称	属性	复位值	描述
7:0	SxRELL	RW	0x00	重载值低 8 位

23.4.4. 重装寄存器高字节 SxRELH

比特	名称	属性	复位值	描述
7:0	SxRELH	RW	0x00	重载值高 8 位

23.4.5. 特殊功能寄存器 PxCON

比特	名称	属性	复位值	描述
7	RSV	-	-	保留
6	WDT_TM	RW	0x0	看门狗定时器测试模式标志位(只存在于 PCON 寄存器)
5	ISR_TM	RW	0x0	中断服务例程测试模式标志位(只存在于 PCON 寄存器)
4	RSV	-	-	保留
3	RSV	-	-	保留
2	GF0	RW	0x0	通用标志位(只存在于 PCON 寄存器)
1	RSV	-	-	保留
0	RSV	-	-	保留

23.5. UART 操作说明

23.5.1. 串口的发送

1) 设置模式、波特率；
2) 写数据到缓冲寄存器 SxBUF，写入 1Byte 数据待发送；
3) 查询 SxCON 的 TI0 位，为“1”时发送完成；
4) 写 1 清 SxCON 的 TI0 位。

23.5.2. 串口的接收

1) 设置模式、波特率、接收使能位；
2) 查询 SxCON 的 RI0 位为“1”时，接收完成；
3) 读 SxBUF；
4) 写 1 清 SxCON 的 RI0 位。

23.5.3. 多机通信

作为主处理器时：

1) 设置模式 3、波特率、将 SxCON 的 SM20 位置 1；
2) 写从机地址到缓冲器区寄存器 SxBUF；
3) 查询 SxCON 的 TI0 位，为“1”时发送完成；

Copyright © 2022 武汉瑞纳捷半导体有限公司

4) 写 1 清 SxCON 的 TI0 位；
5) 将 SxCON 的 SM20 位清 0；
6) 写发送数据到缓冲器区寄存器 SXBUF，保持 SxCON 的 TB80 位为 0；
7) 重复 3~6 到所有数据发送完成。

作为从处理器时：

1) 设置模式 3、波特率、将 SxCON 的 SM20 置 1，接收使能位；
2) 查询 SxCON 的 RI0 位，为“1”时接收完成；
3) 读 SxBUF，软件匹配从机地址；
4) 写 1 清 SxCON 的 RI0 位；
5) 若从机地址匹配，将 SxCON 的 SM20 位清 0，主、从机正常通信；若从机地址不匹配，则 SxCON 的 SM20 位仍为 1，忽略主机发送的数据。

Copyright © 2022 武汉瑞纳捷半导体有限公司

24.LPUART

24.1. 概述

芯片有 1 路 LPUART 接口。LPUART 是一种低功耗的通用异步收发器，允许以低功耗进行半双工通信。在低速时钟下，允许最高的通信速率为 9600 波特率。切换到高速时钟，可进行全双工和实现更高的波特率通信。

24.2. 主要特性

- 工作频率可选（高速时钟：HSI 或 HSE 或 HSO 作系统时钟的 1/2/4/16 分频、低速时钟：LSI 或 LSE）；
- 高速时钟下支持全双工通信，低速时钟下支持半双工通信；
- 32 位的可编程波特率；
- 可产生接收完成中断；
- 具有 ACTIVE HALT 模式下唤醒功能；
- 奇偶校验可选。
- 波特率计算公式：$\text{baud rate} = \frac{256*\text{Fclk}}{\text{LPUART_DIV}}$（Fclk 是模块时钟）

24.3. 寄存器描述

LPUART 寄存器基地址：0x2140

偏移	名称	描述
0x00	LPUART_CON	LPUART 控制寄存器
0x01	LPUART_STS	LPUART 状态寄存器
0x02	LPUART_TXDAT	LPUART 发送数据寄存器
0x03	LPUART_RXDAT	LPUART 接收数据寄存器
0x04	LPUART_DIV70	LPUART 波特率配置寄存器低字节
0x05	LPUART_DIV158	LPUART 波特率配置寄存器次低字节
0x06	LPUART_DIV2316	LPUART 波特率配置寄存次高字节
0x07	LPUART_DIV3124	LPUART 波特率配置寄存器高字节

24.3.1. 控制寄存器 LPUART_CON (偏移：00h)

比特	名称	属性	复位值	描述
7:5	RSV	-	-	保留
4	PARITY_SEL	RW	0x0	奇偶校验选择 0：奇校验

Copyright © 2022 武汉瑞纳捷半导体有限公司

				1：偶校验
3	PARITY_EN	RW	0x0	奇偶校验使能位 0：禁止 1：使能
2	RX_MSK	RW	0x1	屏蔽接收中断 0：不屏蔽 1：屏蔽
1	CLK_SEL	RW	0x0	时钟选择位 0：低速时钟 LSE 或 32KHz(内部) 1：高速系统时钟
0	TX_EN	RW	0x0	发送使能 0：禁止 1：使能 软件置 1，硬件清 0

24.3.2. 状态寄存器 LPUART_STS(偏移：01h)

比特	名称	属性	复位值	描述
7:3	RSV	-	-	保留
2	PARITY_STS	RW	0x0	校验状态位 0：校验正确 1：校验错误 硬件置位，软件写 1 清 0
1	RX_DONE	RW	0x0	接收完成状态位 0：接收未完成 1：接收完成 硬件置位，软件写 1 清 0
0	TX_DONE	RW	0x0	发送状态完成位 0：发送未完成 1：发送完成 硬件置位，软件写 1 清 0

24.3.3. 发送数据寄存器 LPUART_TXDAT (偏移：02h)

比特	名称	属性	复位值	描述
7:0	TXDAT	RW	0x00	发送的 8 位数据 通过软件配置

24.3.4. 接收数据寄存器 LPUART_RXDAT (偏移：03h)

比特	名称	属性	复位值	描述
7:0	RXDAT	RO	0x00	接收的 8 位数据 只读

24.3.5. 波特率配置寄存器低字节 LPUART_DIV70(偏移：04h)

比特	名称	属性	复位值	描述
7:0	UART_DIV70	RW	0x69	波特率配置寄存器的低 8 位

24.3.6. 波特率配置寄存器次低字节 LPUART_DIV158(偏移：05h)

Copyright © 2022 武汉瑞纳捷半导体有限公司

比特	名称	属性	复位值	描述
7:0	UART_DIV158	RW	0x03	波特率配置寄存器的次低 8 位

24.3.7. 波特率配置寄存器次高字节 LPUART_DIV2316(偏移：06h)

比特	名称	属性	复位值	描述
7:0	UART_DIV2316	RW	0x00	波特率配置寄存器的次高 8 位

24.3.8. 波特率配置寄存器高字节 LPUART_DIV3124(偏移：07h)

比特	名称	属性	复位值	描述
7:0	UART_DIV3124	RW	0x00	波特率配置寄存器的高 8 位

24.4. 使用流程

配置 SCCM3，使能 LPUART 模块时钟。

24.4.1. 串口的发送

1) 设置时钟选择位、是否进行奇偶教校验，波特率配置寄存器；
2) 写数据到缓冲寄存器 SXBUF，写入 1Byte 数据待发送；
3) 使能发送；
4) 查询 UART_STS 的 TX_DONE 位，为“1”时发送完成；
5) 写 1 清 UART_STS 的 TX_DONE 位。

24.4.2. 串口的接收

1) 设置时钟选择位、是否进行奇偶校验，是否屏蔽接收中断，波特率配置寄存器；
2) 查询 UART_STS 的 RX_DONE 位，为“1”时接收完成；
3) 读 UART_RDAT；
4) 写 1 清 UART_STS 的 RX_DONE 位。

Copyright © 2022 武汉瑞纳捷半导体有限公司

25.串行外设接口（SPI）

25.1. 概述

芯片有 4 路 SPI 接口：SPI0、SPI1、SPI2 和 SPI3，用于与 SPI 外设之间进行全双工、全同步、串行通讯。SPI 接口支持主/从模式，可以工作在查询或中断方式下。

25.2. 主要特性

- 可选择主/从模式；
- 可配置的传输速率，支持主时钟的 1、2、4、8、16、32、64、128 分频；
- 主机支持 Mode0/1/2 /3 四种传输协议；
- 从机仅支持 Mode0；
- 支持 SPI 四线传输；
- 支持 8 位的串行传输，高位先出；

25.3. SPI 功能描述

25.3.1. SPI 的管脚描述

MISO：主设备输入/从设备输出管脚。

MOSI：主设备输出/从设备输入管脚。

SCK： 串口时钟，作为主设备的输出，从设备的输入。

SPI_CS：从设备选择。这是一个可选的管脚。它的功能是用来作为“片选管脚”，让主设备可以单独的与特定的从设备通信，避免数据线上的冲突。

25.3.2. SPI 的时钟相位和极性

使用 CPOL 和 CPHA 位，能够组合成四种可能的时序关系。

CPOL(时钟极性)位控制在没有数据传输时时钟的空闲状态电平，此位对主模式和从模式下的设备都有效。如果 CPOL 被清 0，SCK 引脚在空闲状态保持低电平；如果 CPOL 被置 1,SCK 引脚在空闲状态保持高电平。

注：确保 SPI 引脚置成 SPI 空闲状态时的电平，以免在使能或者禁止 SPI 模块的时候在 SPI 时钟引脚上产生一个边沿。

如果 CPHA(时钟相位)被置 1，SCK 时钟的第二个边沿(CPOL 位为 0 时就是下降沿，CPOL 位为 1 时就是上升沿)进行数据位采样，数据在第一个时钟传输周期被锁存。如果 CPHA(时钟相位)被清 0，SCK 时钟的第一个边沿(CPOL 位为 0 时就是上升沿，CPOL 位为 1 时就是下降沿)进行数据位采样，数据在第二个时钟传输周期被锁存。

图 26-1 显示了 SPI 的 4 种 CPHA 和 CPOL 位选择不同的组合主设备与从设备的 SCK、MISO、MOSI 引脚直接连接时，这些管脚上的时序。

Copyright © 2022 武汉瑞纳捷半导体有限公司

注：

1. 在改变 CPOL/CPHA 位之前，必须清除 SPI 使能位将 SPI 禁止。

2. 主设备和从设备必须配置成相同的时序模式。

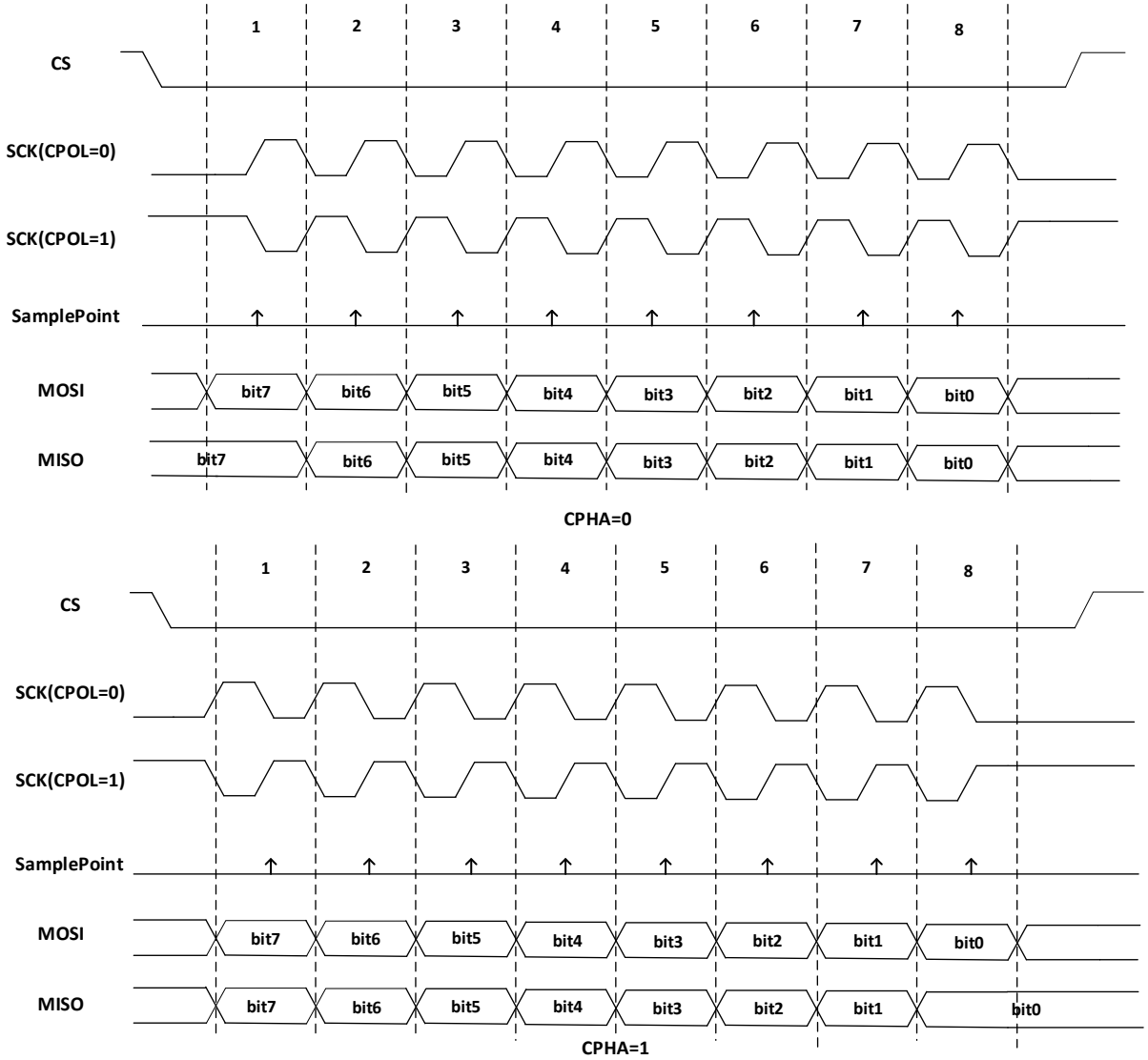


图 26-1 SPI 数据时钟时序图

25.4. SPI 寄存器

SPI0/1/2/3 基地址分别为：0x2030，0x2040，0x21B0，0x21F0

偏移	名称	描述
0x00	SPIxCON	SPI 控制寄存器
0x01	SPIxDAT	SPI 数据寄存器
0x02	SPIxSTS	SPI 状态寄存器
0x03	SPIxMSK	SPI 中断屏蔽寄存器
0x04	SPIxLEN	SPI 数据长度寄存器
0x05	SPIxRXLEN	SPI 一帧数据里接收数据长度寄存器
0x06	SPIxTXLEN	SPI 一帧数据里发送数据长度寄存器

Copyright © 2022 武汉瑞纳捷半导体有限公司

0x08	SPIx_M_WAIT_L	SPI 主机发送数据间隔时间低字节寄存器
0x09	SPIx_M_WAIT_H	SPI 主机发送数据间隔时间高字节寄存器

SPI0/1/2/3 支持 FAST 模式，但在同一时刻只能有一路使用 FAST 操作模式。

SPI FAST MODE 配置寄存器基地址：0x2000

偏移	名称	描述
0xC1	SPI_FAST_RX_ADDRL	FAST 模式下接收数据写入 SRAM 首地址低 8 位
0xC2	SPI_FAST_RX_ADDRH	FAST 模式下接收数据写入 SRAM 首地址高 4 位
0xC3	SPI_FAST_TX_ADDRL	FAST 模式下发送数据读出 SRAM 首地址低 8 位
0xC4	SPI_FAST_TX_ADDRH	FAST 模式下发送数据读出 SRAM 首地址高 4 位
0xC5	SPI_FAST_SEL	支持 FAST 模式的 SPI 组选择寄存器

25.4.1. 控制寄存器 SPIxCON(偏移：00H)

比特	名称	属性	复位值	描述
7	RUN_EN	RW	0x0	SPI 工作使能位 0：禁止 1：使能
6	SPI_MS_SEL	RW	0x0	主/从机选择位 0：主机模式 1：从机模式
5	RSV	-	-	保留
4:2	SPI_RATE	RW	0x0	SPI 主机波特率设置 000 : SPI 系统时钟 2 分频 001 : SPI 系统时钟 4 分频 010 : SPI 系统时钟 8 分频 011 : SPI 系统时钟 16 分频 100 : SPI 系统时钟 32 分频 101 : SPI 系统时钟 64 分频 110 : SPI 系统时钟 128 分频 111 : SPI 系统时钟 1 分频
1	CPHA	RW	0x0	SPI 主机时钟相位控制 0：在时钟 SCK 的奇边沿采样数据 1：在时钟 SCK 的偶边沿采样数据
0	CPOL	RW	0x0	SPI 主机时钟极性控制 0：时钟 SCK 初始电平为低电平 1：时钟 SCK 初始电平为高电平

25.4.2. 数据寄存器 SPIxDAT(偏移：01H)

比特	名称	属性	复位值	描述
7:0	SPIDAT	RW	0x00	发送时，作为发送数据寄存器；

Copyright © 2022 武汉瑞纳捷半导体有限公司

				接收时，作为接收数据寄存器。

25.4.3. 状态寄存器 SPIxSTS(偏移：02H)

比特	名称	属性	复位值	描述
7:5	RSV	-	-	保留
4	FRAME_OVER	RW	0x0	SPI 一帧数据完成标志（从机时，SPI_CS 信号拉高触发此标志；主机时，发送完 RXCNT 个数据后触发此标志） 0：一帧数据未完成 1：一帧数据已完成 该标志硬件置位，软件写 1 清 0
3	RX_RXCNT_FLAG	RW	0x0	SPI 从机接收到 RXCNT 长度数据标志 0：SPI 未接收到 RXCNT 长度数据 1：SPI 已接收到大于或等于 RXCNT 长度数据 该标志硬件置位，软件清 0
2	SPI_CS_FALL	RW	0x0	SPI 从机通信开始标志位 0：SPI 通信未开始 1：SPI 通信开始，SPI_CS 下降沿置 1 该标志硬件置位，软件写 1 清 0
1	TX_RX_SEL	RW	0x0	SPI 发送/接收选择位 0：接收 1：发送 该标志软件置位/清 0；
0	BOVER	RW	0x0	当前字节数据交换结束标志位 0：未结束 1：结束 硬件置位，软件写 1 清 0

25.4.4. 中断屏蔽寄存器 SPIxMSK(偏移：03H)

比特	名称	属性	复位值	描述
7:6	RSV	-	-	保留
5	TXDATA_SEL	RW	0x0	在 FAST 模式下，SPI 做主机发送数据时，SPI_OUT 管脚的数据来源选择 0：来自 SPIDAT； 1：来自指定的 SRAM 区域；
4	FAST_MODE_EN	RW	0x0	SPI 主/从机快速模式使能位 0：不使能； 1：使能 FAST 模式；
3	FRAME_OVER_MASK	RW	0x1	屏蔽“FAST 模式下主/从机一帧数据完成”中断控制位 0：不屏蔽

Copyright © 2022 武汉瑞纳捷半导体有限公司

				1：屏蔽
2	RX_RXCNT_MASK	RW	0x1	屏蔽“FAST 模式下从机接收 RXCNT 字节数据”中断控制位 0：不屏蔽 1：屏蔽
1	FILTER_EN	RW	0x0	滤波使能，使能后 SPI 从机的通信速率不高于 4M 0：关闭 1：开启
0	BOVER_MASK	RW	0x1	屏蔽“当前字节数据交换结束”中断控制位 0：不屏蔽 1：屏蔽

25.4.5. 数据长度寄存器 SPIxLEN(偏移：04H)

比特	名称	属性	复位值	描述
7	RSV	-	-	保留
6:0	SPILEN	RW	0x4	从机：接收数据长度中断阈值控制寄存器； 主机：待发送数据长度；

25.4.6. 一帧数据里接收数据长度寄存器 SPIxRXLEN(偏移：05H)

比特	名称	属性	复位值	描述
7	RSV	-	-	保留
6:0	FRAME_RX_CNT	R	0x0	一帧数据里接收数据长度（在 SPI_CS 下降沿时硬件清 0）

25.4.7. 一帧数据里发送数据长度寄存器 SPIxTXLEN(偏移：06H)

比特	名称	属性	复位值	描述
7	RSV	-	-	保留
6:0	FRAME_TX_CNT	RW	0x7F	W：一帧数据里发送数据长度 R：一帧数据里剩余待发送数据长度 （读 SPIxSTS 寄存器时硬件清 0）

25.4.8. 主机发送数据间隔时间低字节寄存器 SPIx_M_WAIT_L (偏移：08H)

比特	名称	属性	复位值	描述
7:0	SPI_M_WAIT_L	RW	0x04	SPI 主机发送数据间隔时间低 8 位；

25.4.9. 主机发送数据间隔时间高字节寄存器 SPIx_M_WAIT_H (偏移：09H)

比特	名称	属性	复位值	描述
7:0	SPI_M_WAIT_H	RW	0x0	SPI 主机发送数据间隔时间高 8 位；

25.4.10. FAST 接收数据写入 SRAM 首地址低 8 位寄存器 SPI_FAST_RX_ADDRL(地址：0x20C1)

比特	名称	属性	复位值	描述
7:0	SPI_FAST_RX_ADDRL	RW	0x0	接收写入 SRAM 首地址低 8 位；

25.4.11. FAST 接收数据写入 SRAM 首地址高 4 位寄存器 SPI_FAST_RX_ADDRH(地址：0x20C2)

比特	名称	属性	复位值	描述
7:0	SPI_FAST_RX_ADDRH	RW	0x0	接收写入 SRAM 首地址高 4 位

25.4.12. FAST 发送数据读出 SRAM 首地址低 8 位寄存器 SPI_FAST_TX_ADDRL(地址：0x20C3)

比特	名称	属性	复位值	描述
7:0	SPI_FAST_TX_ADDRL	RW	0X0	发送数据读出 SRAM 首地址低 8 位

25.4.13. FAST 发送数据读出 SRAM 首地址高 4 位寄存器 SPI_FAST_TX_ADDRH(地址：0x20C4)

比特	名称	属性	复位值	描述
7:0	SPI_FAST_TX_ADDRH	RW	0x0	发送数据读出 SRAM 首地址高 4 比特

25.4.14. FAST 模式 SPI 组选择寄存器 SPI_FAST_SEL(地址：0x20C5)

比特	名称	属性	复位值	描述
7:2	RSV	RW	0x0	预留
1:0	SPI_FAST_SEL	RW	0x0	支持 FAST 操作的 SPI 组 0：SPI0； 1：SPI1； 2：SPI2； 3：SPI3；

25.5. SPI 操作流程

25.5.1. SPI 主机配置

1) 配置 SCCM0，使能 SPI 模块时钟。配置 RSTCON0，失能 SPI 模块复位；
2) 配置 IOCFG_CTRL 寄存器，将 IO 管脚复用为 SPI 功能；
3) 配置 SPIMSK，选择是否屏蔽字节交换完成中断和滤波功能；
4) 配置 SPICON，设置时钟相位与极性、时钟分频、主机模式并工作使能；
5) 将数据写入 SPIDAT，等待并清除 SPISTS 的 BOVER 状态，而后读取 SPIDAT，若上述字节交换完成中断未屏蔽，同时使能 SPI 模块中断与总中断，则会触发中断进行异步处理；

25.5.2. SPI FAST 主机发送配置

Copyright © 2022 武汉瑞纳捷半导体有限公司

1) 配置 SCCM0，使能 SPI 模块时钟。配置 RSTCON0，失能 SPI 模块复位；
2) 配置 IOCFG_CTRL 寄存器，将 IO 管脚复用为 SPI 功能；
3) 配置 SPICON，设置时钟相位与极性、时钟分频和主机模式；
4) 配置 SPI_FAST_TX_ADDRH 和 SPI_FAST_TX_ADDRL，设置发送数据读出 SRAM 的地址。配置 SPILEN，设置数据长度。配置 SPISTS，设置为发送；
5) 配置 SPIMSK，选择 SRAM 数据源与使能 FAST 模式。配置 SPICON，使能 SPI 模块；
6) 写 SPIDAT 启动 SPI FAST 发送；
7) 等待 SPISTS 的 FRAME_OVER 状态标记而后清除并退出；

25.5.3. SPI FAST 主机接收配置

1) 配置 SCCM0，使能 SPI 模块时钟。配置 RSTCON0，失能 SPI 模块复位；
2) 配置 IOCFG_CTRL 寄存器，将 IO 管脚复用为 SPI 功能；
3) 配置 SPICON，设置时钟相位与极性、时钟分频和主机模式；
4) 配置 SPI_FAST_RX_ADDRH 和 SPI_FAST_RX_ADDRL，设置接收数据写入 SRAM 的地址。配置 SPILEN，设置数据长度。配置 SPISTS，设置为接收；
5) 配置 SPIMSK，选择 SRAM 数据源与使能 FAST 模式。配置 SPICON，使能 SPI 模块；
6) 写 SPIDAT 启动 SPI FAST 发送；
7) 等待 SPISTS 的 FRAME_OVER 状态标记而后清除并退出；

25.5.4. SPI 从机配置

1) 配置 SCCM0，使能 SPI 模块时钟。配置 RSTCON0，失能 SPI 模块复位；
2) 配置 IOCFG_CTRL 寄存器，将 IO 管脚复用为 SPI 功能；
3) 配置 SPICON，设置为从机模式；
4) 配置 SPIMSK，不屏蔽字节交换完成中断；
5) 使能 SPI 模块中断与总中断。
6) 配置 SPICON 的 RUN_EN，工作使能；
7) 配置 SPISTS 的 TX_RX_SEL，设置成接收或发送，同时配合读取或写入 SPIDAT 寄存器完成数据传输。

25.5.5. SPI FAST 从机接收配置

1) 配置 SCCM0，使能 SPI 模块时钟。配置 RSTCON0，失能 SPI 模块复位；
2) 配置 IOCFG_CTRL 寄存器，将 IO 管脚复用为 SPI 功能；
3) 配置 SPICON，设置为从机模式；
4) 配置 SPI_FAST_RX_ADDRH 和 SPI_FAST_RX_ADDRL，设置接收数据写入 SRAM 的地址。配置 SPILEN，设置数据长度。配置 SPISTS，设置为接收；

Copyright © 2022 武汉瑞纳捷半导体有限公司

5) 配置 SPIMSK，选择 SRAM 数据源与使能 FAST 模式并不屏蔽帧数据完成中断与接收 RXCNT 字节数据中断。开启 SPI 模块中断与总中断。配置 SPICON，使能 SPI 模块；
6) 等待中断事件发生并处理。

25.5.6. SPI FAST 从机发送配置

1) 配置 SCCM0，使能 SPI 模块时钟。配置 RSTCON0，失能 SPI 模块复位；
2) 配置 IOCFG_CTRL 寄存器，将 IO 管脚复用为 SPI 功能；
3) 配置 SPICON，设置为从机模式；
4) 配置 SPI_FAST_TX_ADDRH 和 SPI_FAST_TX_ADDRL，设置发送数据读出 SRAM 的地址。配置 SPILEN，设置数据长度。配置 SPISTS，设置为发送；
5) 配置 SPIMSK，选择 SRAM 数据源与使能 FAST 模式并不屏蔽帧数据完成中断。开启 SPI 模块中断与总中断。配置 SPICON，使能 SPI 模块；
6) 等待中断事件发生并处理。

Copyright © 2022 武汉瑞纳捷半导体有限公司

26.版本号

版本号用于标识芯片型号，是一个固定的 32bit 数，本芯片型号为 RJM8L151S，其版本号固定为 0x43313531。

26.1. 版本号寄存器

版本号寄存器基地址：0x2000

地址	名称	描述
0x1A	VERSION0	版本号寄存器 0
0x1B	VERSION1	版本号寄存器 1
0x1C	VERSION2	版本号寄存器 2
0x1D	VERSION3	版本号寄存器 3

26.1.1. 版本寄存器 VERSION0（偏移：1AH）

比特	名称	属性	复位值	描述
7:0	VERSION0	RO	0x31	版本号 0

26.1.2. 版本寄存器 VERSION1（偏移：1BH）

比特	名称	属性	复位值	描述
7:0	VERSION1	RO	0x35	版本号 1

26.1.3. 版本寄存器 VERSION2（偏移：1CH）

比特	名称	属性	复位值	描述
7:0	VERSION2	RO	0x31	版本号 2

26.1.4. 版本寄存器 VERSION3（偏移：1DH）

比特	名称	属性	复位值	描述
7:0	VERSION3	RO	0x43	版本号 3

Copyright © 2022 武汉瑞纳捷半导体有限公司

27.编程接口

芯片实现了两线编程接口：RJ_SWD。RJ_SWD 接口由时钟输入 RJ_SCK 与数据输入/输出 RJ_SDA 组成。在 memory 保护的允许范围内，通过 RJ_SWD 接口可访问片内资源。

RJ_SWD 接口后续将主要被用来实现仿真调试与代码烧录功能，若用户代码中存在对 RJ_SDA 和 RJ_SCK 管脚的复用，则建议设计时预留出复位引脚 RST。

RJM8L151S 系列芯片与烧录器对应引脚的关系如下：

引脚名称		功能
烧录器	RJM8L151S 系列芯片	
NRST	RST	复位脚
RJ_SDA	P00	数据脚
RJ_SCK	P01	时钟脚

Copyright © 2022 武汉瑞纳捷半导体有限公司

附录 1：寄存器比特位属性及说明

比特位属性	说明
RW	Read/Write，软件可以读写这些位
RO	Read Only，软件只可以读这些位
WO	Write Only，软件仅可以写这些位，读这些位时将会返回该位的复位值
W1C	Write 1 Clear，软件可以通过向该位写 1 来清零，向其写 0 将不影响该位
W0C	Write 0 Clear，软件可以通过向该位写 0 来清零，向其写 1 将不影响该位

Copyright © 2022 武汉瑞纳捷半导体有限公司

附录 2：专业术语缩写及说明

缩写	说明
B	Byte，字节
CPU	Central Processing Unit，中央处理器
ESD	Electro-Static discharge，静电放电
NMROM	Normal Mode Read Only Memory，正常模式只读存储器
NVM	Non-Volatile Memory，非易失性存储器
POR	Power On Reset，上电复位
PDR	Power down Reset，掉电复位
BOR	Brown-out Reset，欠压复位
RNG	Random Number Generator，随机数发生器
SFR	Special Function Register，特殊功能寄存器
SN	Chip Serial Number，芯片序列号（唯一标识符）
VR	Voltage Regulator，电压调整器
WDT	Watch Dog TIMER，看门狗计数器

Copyright © 2022 武汉瑞纳捷半导体有限公司

附录 3：版本修订

版本	日期	作者	描述
V1.0	2022.03.25	邓勇	1) 应用手册初版
V1.1	2022.07.27	邓勇	1) 更正封面页公司英文字体格式 2) 更正“15.2.主要特性”描述：“具有 HALT 模式下唤醒功能”HALT 改为 ACTIVE HALT 3) 更正“24.2.主要特性”描述：“具有 HALT 模式下唤醒功能”HALT 改为 ACTIVE HALT 4) 更正“表 14-1 TIMER3 和 TIMER4 比较通道对应输出引脚”描述 5) 更正“19.3.3 32bit/16bit 乘法运算”中的“乘法”为“除法” 6) 更正“8.4.14 GPIO 功能复用表”表格下面描述：内部高速时钟输出管脚由 P03 改为 P04 7) 更正“3.6.7~3.6.10 FLASH 解锁寄存器”的偏移 8) 更正“表 14-1 TIMER3 和 TIMER4 比较通道对应输出引脚”

Copyright © 2022 武汉瑞纳捷半导体有限公司