SPC1169 技术参考手册

前言

这份参考手册面向应用开发人员，其内容涵盖如何使用 SPC1169 微控制器、存储器和外设的完整信息。有关设备的机械和电气特性，请参阅 SPC1169 数据手册。

SPINTROL

目录

1 存储器和总线结构 74

1.1 系统架构 74

1.2 时钟树 75

1.3 存储空间地址映射 76

1.3.1 嵌入式 Flash 存储器 78

1.3.2 片上 SRAM 78

1.3.3 引导 ROM 78

1.4 启动引导配置 81

2 电源 82

2.1 电源供电 82

2.2 电源管理单元 83

2.3 电源产生单元 84

2.3.1 VCAP12 LDO 84

2.4 低压过压/欠压检测 85

2.5 系统唤醒 87

2.6 寄存器 88

2.6.1 POWER 寄存器列表 88

2.6.2 POWER 寄存器 89

3 时钟和复位 94

3.1 复位框架 94

3.2 引脚复位 94

3.3 上电复位 95

3.4 时钟架构 95

3.5 高频振荡器 96

3.6 内部晶振和外部时钟 96

3.7 PLL 98

3.7.1 输入参考和分频 98

3.7.2 输出分频 99

3.7.3 环路分频 99

3.7.4 PLL 失锁检测 99

3.8 时钟安全 100

3.9 寄存器 101

SPINTROL

©2025 旋智电子科技（上海）有限公司

3.9.1 CLOCK 寄存器列表 101
3.9.2 CLOCK 寄存器 102

4 通用输入输出（GPIO） 117

4.1 概述 117
4.2 引脚复用（PINMUX） 118
4.2.1 消抖控制（Deglitch） 119
4.2.2 I/O 可选功能 119
4.2.3 环回路径（Loopback） 121
4.2.4 调试模式 121
4.3 GPIO 控制模块 122
4.3.1 方向和电平控制 122
4.3.2 边沿触发中断 123
4.3.3 电平触发中断 125
4.4 寄存器 127
4.4.1 PINMUX 寄存器列表 127
4.4.2 PINMUX 寄存器 129
4.4.3 GPIO 寄存器列表 173
4.4.4 GPIO 寄存器 175

5 中断和事件 193

5.1 嵌套向量中断控制器（NVIC） 193
5.1.1 NVIC 特性 193
5.1.2 中断和异常向量 193
5.1.3 NMI 中断 195
5.1.4 NVIC 中断配置 196
5.2 SysTick 定时器 196

6 通用定时器（TIMER） 197

6.1 概述 197
6.2 功能描述 197
6.2.1 通用计时模式 198
6.2.2 门控定时模式 198
6.2.3 事件计数模式 198
6.2.4 事件捕获模式 199
6.3 初始化流程 199
6.4 寄存器 200
6.4.1 TIMER 寄存器列表 200

SPINTROL

6.4.2 TIMER 寄存器 201

7 看门狗定时器（WDT） 206

7.1 概述 206

7.2 功能描述 206

7.3 初始化流程 207

7.4 寄存器 208

7.4.1 WDT 寄存器列表 208

7.4.2 WDT 寄存器 209

8 脉宽调制模块（PWM） 215

8.1 概述 215

8.2 时基产生（TB）子模块 217

8.3 计数比较（CC）子模块 221

8.4 行为限定（AQ）子模块 222

8.5 死区控制（DB）子模块 224

8.6 信号封锁（TZ）子模块 227

8.7 数字比较（DC）子模块 229

8.8 事件触发（ET）子模块 232

8.9 寄存器的全局重置 233

8.10 寄存器同时写入 234

8.11 寄存器 235

8.11.1 PWM 寄存器列表 235

8.11.2 PWM 寄存器 237

8.11.3 PWMCFG 寄存器列表 319

8.11.4 PWMCFG 寄存器 320

9 增强型捕获模块（ECAP） 332

9.1 概述 332

9.2 功能描述 332

9.3 捕获模式和辅助 PWM 模式 333

9.4 捕获模式详述 334

9.4.1 事件分频器 334

9.4.2 边沿选择和排序编号 335

9.4.3 双引脚捕获 335

9.4.4 单轮捕获和持续捕获控制 335

9.4.5 32 位计数器和相位控制 336

9.4.6 CAP0-CAP3 寄存器 336

©2025 旋智电子科技（上海）有限公司

9.4.7 中断控制 337
9.5 APWM 模式详述 338
9.6 ECAP 的应用 339
9.6.1 上升沿触发的绝对时间戳捕获 339
9.6.2 由绝对时间戳捕获周期和占空 341
9.6.3 通过计数器复位捕获相对时间差 342
9.6.4 相对事件模式捕获周期和占空 343
9.6.5 基于计数器复位的双引脚捕获 344
9.7 寄存器 345
9.7.1 ECAP 寄存器列表 345
9.7.2 ECAP 寄存器 346
10 13 位模数转换器（ADC） 363
10.1 概述 363
10.2 ADC 架构 364
10.3 基于控制通道的工作原理 364
10.3.1 SOC 触发源 365
10.3.2 输入选择 365
10.3.3 采样和转换窗口 366
10.3.4 结果平均 367
10.3.5 偏移和增益校准 367
10.3.6 时序图 368
10.4 ADC 转换优先级 368
10.5 EOC 及中断 372
10.6 后处理单元 372
10.6.1 参考比较 372
10.6.2 窗口比较和过零检测 373
10.6.3 采样延迟的测量 373
10.7 预期码值的计算 373
10.8 安全特性 374
10.8.1 输入浮空检测 374
10.8.2 输入短路检测 376
10.9 寄存器 377
10.9.1 ADC 寄存器列表 377
10.9.2 ADC 寄存器 379
11 13 位检测模数转换器（ADC） 417

SPINTROL

11.1 概述 417
11.2 架构及操作 417
11.3 寄存器 419

12 单端可编程增益放大器（SPGA） 420
12.1 概述 420
12.2 架构 420
12.3 输入选择 420
12.4 增益选择 421
12.5 寄存器 421

13 差分可编程增益放大器（DPGA） 422
13.1 概述 422
13.2 输入及输入电压范围 422
13.3 应用 422
13.4 偏置校准 424
13.5 寄存器 425
13.5.1 PGA 寄存器列表 425
13.5.2 PGA 寄存器 426

14 模拟比较器（COMP） 429
14.1 概述 429
14.2 比较器架构 429
14.3 比较器输入选择 430
14.4 数字滤波器 430
14.5 寄存器 431
14.5.1 COMP 寄存器列表 431
14.5.2 COMP 寄存器 432

15 相位比较器（PHCOMP） 449
15.1 概述 449
15.2 相位比较器架构 449
15.3 后处理过程 450
15.4 寄存器 453
15.4.1 PHCOMP 寄存器列表 453
15.4.2 PHCOMP 寄存器 454

16 数模转换器（DAC） 464
16.1 概述 464

©2025 旋智电子科技（上海）有限公司

16.2 DAC 架构 464
16.3 斜坡发生器 465
16.4 寄存器 467

17 差分转单端缓冲器（D2S） 468
17.1 概述 468
17.2 功能描述 468

18 温度传感器 469
18.1 温度传感器概述 469
18.2 传递函数 469
18.3 寄存器 470

19 通用异步收发器（UART） 471
19.1 概述 471
19.2 UART 特征 472
19.3 UART 端口信号 473
19.3.1 UART 设备连接 473
19.3.2 LIN 设备连接 474
19.4 UART 操作 474
19.4.1 复位 475
19.4.2 FIFO 操作 475
19.4.3 接收超时和删除尾字节 476
19.4.4 波特率配置 476
19.4.5 自动波特率检测 477
19.4.6 串行红外异步接口 478
19.5 LIN 操作 479
19.5.1 LIN 协议 480
19.5.2 帧的结构 480
19.5.3 受保护 ID 481
19.5.4 校验和段 482
19.6 中断 482
19.6.1 接受中断 482
19.6.2 接收超时中断 482
19.6.3 波特率自动计数器锁定接收中断 483
19.6.4 接受 FIFO 中断 483
19.6.5 发送 FIFO 中断 483
19.6.6 LIN 错误中断 483

SPINTROL

19.6.7 DMA 中断 483
19.7 寄存器 484
19.7.1 UART 寄存器列表 484
19.7.2 UART 寄存器 485
20 串行外设接口（SPI） 507
20.1 概述 507
20.2 特性 507
20.3 SPI 端口信号 509
20.4 SPI 功能描述 509
20.4.1 波特率产生 509
20.4.2 FIFO 操作 509
20.4.3 SPI 中断 511
20.5 Motorola SPI 操作 512
20.5.1 设备连接 512
20.5.2 协议描述 513
20.6 TI SSP 操作 515
20.6.1 设备连接 515
20.6.2 协议描述 515
20.7 可编程串行协议 PSP 517
20.7.1 设备连接 517
20.7.2 协议描述 517
20.8 寄存器 520
20.8.1 SPI 寄存器列表 520
20.8.2 SPI 寄存器 521
21 内部集成电路总线（I2C） 537
21.1 概述 537
21.2 特性 537
21.3 I2C 端口信号描述 537
21.4 I2C 功能描述 537
21.4.1 START 和 STOP 信号的产生 538
21.4.2 混合格式传输 539
21.5 I2C 协议 539
21.5.1 START 和 STOP 信号 539
21.5.2 地址格式 539
21.5.3 收发协议 540

©2025 旋智电子科技（上海）有限公司

21.6 寄存器 542
21.6.1 I2C 寄存器列表 542
21.6.2 I2C 寄存器 543
22 控制器局域网络（CAN） 568
22.1 概述 568
22.2 CAN 端口信号 570
22.2.1 CAN 设备连接 570
22.3 CAN 帧格式 570
22.3.1 ISO 经典 CAN 570
22.3.2 ISO FDCAN 571
22.3.3 BOSCH CAN2.0B 571
22.3.4 NONISO FDCAN 572
22.3.5 仲裁阶段和数据阶段 572
22.4 CAN 错误状态 572
22.4.1 错误记录 573
22.4.2 主动错误状态 573
22.4.3 被动错误状态 573
22.4.4 总线关闭恢复 573
22.5 CAN 操作 574
22.5.1 速率配置 574
22.5.2 监控模式 575
22.5.3 限制模式 575
22.5.4 测试模式 575
22.6 信箱控制 577
22.6.1 信箱信息 577
22.6.2 非自动应答远程帧时的报文发送 578
22.6.3 自动应答远程帧时的报文发送 578
22.6.4 未对其他节点发出远程帧数据请求时的报文接收 579
22.6.5 对其他节点发出远程帧数据请求时的报文接收 580
22.7 寄存器 581
22.7.1 CAN 寄存器列表 581
22.7.2 CAN 寄存器 582
22.7.3 CANMBOX 寄存器列表 603
22.7.4 CANMBOX 寄存器 604
23 循环冗余校验（CRC） 609

SPINTROL

23.1 概述 609
23.2 主要特性 609
23.3 CRC 工作流程 610
23.4 寄存器 611
23.4.1 CRC 寄存器列表 611
23.4.2 CRC 寄存器 612

24 Flash 存储器 619

24.1 特性 619
24.2 Flash 架构 619
24.3 读操作 621
24.3.1 XIP 读接口读操作 621
24.3.2 Flash 控制器接口读操作 621
24.4 编程和擦除操作 622
24.4.1 使能 Flash 控制器 622
24.4.2 Flash 存储器编程操作 623
24.4.3 Flash 存储器擦除操作 624
24.5 保护机制 626
24.5.1 分区保护 626
24.5.2 写保护 626
24.6 Flash 时序设置 626
24.7 配置字（Configuration words） 627
24.8 寄存器 628
24.8.1 Flash 寄存器列表 628
24.8.2 Flash 寄存器 629

25 系统控制（SYSTEM） 638

25.1 概述 638
25.2 64 位产品识别号 638
25.3 系统监控寄存器 638
25.4 GPIO 唤醒控制寄存器 638
25.5 存储器 ECC 错误地址寄存器 638
25.6 NMI 事件控制寄存器 639
25.7 复位事件控制寄存器 639
25.8 系统信息寄存器 639
25.9 启动信息寄存器 639
25.10 寄存器 640

©2025 旋智电子科技（上海）有限公司

25.10.1 SYSTEM 寄存器列表 640
25.10.2 SYSTEM 寄存器 642

26 直接存储器访问控制器（DMAC） 703

26.1 概述 703
26.2 DMA 特性 703
26.3 DMA 工作过程 704
26.3.1 基本定义 704
26.3.2 硬件握手接口 705
26.3.3 Memory-to-memory transfer operation 705
26.3.4 内存到外设和外设到内存传输 706
26.3.5 外设到外设的传输 709
26.4 AHB 传输错误处理 710
26.5 中断处理 710
26.5.1 中断丢失和阻塞的风险缓解 711
26.6 寄存器 713
26.6.1 DMACH 寄存器列表 713
26.6.2 DMACH 寄存器 714
26.6.3 DMAC 寄存器列表 720
26.6.4 DMAC 寄存器 721

SPINTROL

图片列表

图 1-1：SPC1169 系统框图 74

图 1-2：SPC1169 时钟树 75

图 1-3：SPC1169 存储空间地址映射 76

图 1-4：引导 ROM 地址空间映射 79

图 2-1：电源系统总览 82

图 2-2：电源模式转换 83

图 2-3：上电时序 84

图 2-4：VCAP12 LDO 84

图 2-5：3.3V 电压域失调探测框图 85

图 2-6：1.2V 电压域电压失调探测框图 85

图 2-7：DVDD33 阈值 86

图 2-8：VCAP12 阈值 87

图 3-1：复位架构 94

图 3-2：上电复位时序图 95

图 3-3：时钟树 96

图 3-4：内部晶振模式 97

图 3-5：外部时钟 97

图 3-6：PLL 结构 98

图 3-7：PLL 失锁检测时序图示例（$f_{PFD} > f_{DIV}$） 99

图 3-8：时钟交叉检测 100

©2025 旋智电子科技（上海）有限公司

图 4-1：I/O 控制框图 117
图 4-2：PINMUX 框图 118
图 4-3：输入消抖 119
图 4-4：COMP_MONx 源 120
图 4-5：GPIO 框图 122
图 4-6：GPIO 边沿检测 123
图 4-7：GPIO 边沿触发中断 124
图 4-8：GPIO 电平检测 125
图 4-9：电平触发中断 125
图 6-1：通用定时器框图 197
图 6-2：通用定时器在通用定时模式下的波形（TMRLOAD=100） 198
图 6-3：通用定时器在门控定时模式下的波形（TMRLOAD=100） 198
图 6-4：通用定时器在事件计数模式下的波形（TMRLOAD=100） 198
图 6-5：通用定时器在事件捕获模式下的波形（TMRLOAD=100） 199
图 6-6：通用定时器运行流程图 199
图 7-1：看门狗工作流程 206
图 8-1：PWM 系统总览 215
图 8-2：PWM 内部结构 216
图 8-3：PWM 时基产生子模块信号链 218
图 8-4：时基计数波形 219
图 8-5：时基相位同步波形 220

SPINTROL

图 8-6：计数比较子模块结构 221

图 8-7：计数比较子模块波形 222

图 8-8：行为限定子模块输入输出 222

图 8-9：上下计数时的 PWM 双边沿对称波形 224

图 8-10：上下计数时的 PWM 双边沿非对称波形 224

图 8-11：死区控制子模块结构 225

图 8-12：死区控制典型波形 226

图 8-13：双边沿延迟死区控制波形 226

图 8-14：信号封锁子模块结构 227

图 8-15：信号封锁中断产生 229

图 8-16：数字比较子模块结构 230

图 8-17：事件过滤逻辑 230

图 8-18：消隐窗口时序图 231

图 8-19：事件处理 232

图 8-20：事件触发子模块结构 233

图 8-21：影子寄存器全局读入 234

图 9-1：ECAP 基本工作方式 333

图 9-2：捕获模式和 APWM 模式 333

图 9-3：捕获模式功能框图 334

图 9-4：事件分频控制 334

图 9-5：事件预分频波形 335

©2025 旋智电子科技（上海）有限公司

图 9-6：单轮捕获和持续捕获逻辑 336

图 9-7：计数器及同步逻辑 336

图 9-8：ECAP 模块中断 337

图 9-9：APWM 模式下 PRD 和 CMD 的有效值和影子值 338

图 9-10：APWM 模式波形（极性为高电平有效） 339

图 9-11：上升沿检测和绝对时间戳捕获 340

图 9-12：通过绝对事件戳捕获计算周期和占空 341

图 9-13：通过计数器复位捕获周期 342

图 9-14：通过计数器复位捕获周期与占空 343

图 9-15：基于计数器复位的双引脚捕获 344

图 10-1：ADC 框图 364

图 10-2：ADC 通道 SOC 产生 365

图 10-3：模拟输入引脚模型 367

图 10-4：ADC 结果平均及偏移和增益校准 367

图 10-5：ADC 时序示例 368

图 10-6：ADC SOC 轮询仲裁示例 369

图 10-7：ADC 带优先组的 SOC 仲裁示例 371

图 10-8：ADC 带优先组的 SOC 仲裁示例 372

图 10-9：ADC 后处理单元框图 372

图 10-10：输入引脚悬空的检测（通过节点放电的故障注入） 375

图 10-11：输入引脚悬空的检测（通过节点充电的故障注入） 375

SPINTROL

图 11-1：检测 ADC 框图 417
图 12-1：SPGA 框图 420
图 13-1：DPGA 应用框图 423
图 13-2：FPGA 输入范围的要求 423
图 13-3：DPGA 偏置校准的路径 424
图 14-1：比较器框图 429
图 14-2：数字滤波器 430
图 15-1：相位比较器框图 449
图 15-2：相位比较器后处理过程 450
图 15-3：数字滤波器的行为 451
图 15-4：相位比较器中断 452
图 16-1：DAC 框图 464
图 16-2：DAC 斜坡发生器框图 466
图 16-3：斜坡发生器波形 467
图 17-1：差分转单端缓冲器应用框图 468
图 18-1：温度传感器传递函数 469
图 19-1：UART 框图 471
图 19-2：UART 设备连接图 473
图 19-3：LIN 设备连接图 474
图 19-4：UART 帧格式 474
图 19-5：UART 数据 NRZ 编码 475

©2025 旋智电子科技（上海）有限公司

图 19-6：自动波特率侦测 477
图 19-7：红外发送和接收 478
图 19-8：XMODE 示例 479
图 19-9：LIN 总线拓扑图 480
图 19-10：LIN 字节字段 480
图 19-11：LIN 帧的结构 480
图 19-12：LIIN 传输 481
图 19-13：LIN 保护的 ID 字节字段 481
图 20-1：SPI 框图 507
图 20-2：SPI 框图 508
图 20-3：FIFO 压缩和非压缩模式 510
图 20-4：SPI 主从链接 512
图 20-5：SPI 协议（PHS=0） 513
图 20-6：SPI 协议（PHS=1） 514
图 20-7：SPI 帧背靠背传输 515
图 20-8：SSP 协议（单次传输） 516
图 20-9：SSP 协议（背靠背传输） 516
图 20-10：PSP 协议（单次传输） 518
图 20-11：PSP 协议（连续传输） 519
图 21-1：I2C 总线数据传输 538
图 21-2：START 和 STOP 信号 539

SPINTROL

图 21-3：7 地址格式格式 540
图 21-4：10 地址格式 540
图 21-5：主机-发送协议 541
图 21-6：主机-接收协议 541
图 22-1：CAN 框图 569
图 22-2：CAN 设备连接 570
图 22-3：ISO 经典 CAN 帧格式 571
图 22-4：ISO FDCAN 帧格式 571
图 22-5：BOSCH CAN2.0B 帧格式 571
图 22-6：NONISO FDCAN 帧格式 572
图 22-7：仲裁阶段和数据阶段 572
图 22-8：CAN 节点错误状态转换 573
图 22-9：位时序 574
图 22-10：发送器延时补偿 575
图 22-11：监控模式 575
图 22-12：内部环回模式 576
图 22-13：外部环回模式 576
图 23-1：CRC 模块框图 609
图 24-1：Flash 存储器访问接口 619
图 24-2：Flash 模块逻辑结构 620
图 24-3：Flash 存储器读操作流程 622

©2025 旋智电子科技（上海）有限公司

图 24-4：Flash 存储器编程操作流程 623

图 24-5：Flash 存储器扇区擦除操作流程 624

图 24-6：Flash 擦除流程图 625

图 26-1：DMA 控制器框图 704

图 26-2：DMA 内存传输的分解 705

图 26-3：DMA 非内存传输的分解 706

图 26-4：DMA 中断框图 710

图 26-5：DMA 中断服务程序伪代码 711

图 26-6：DMA 多通道中断事件 711

SPINTROL

表格列表

表 1-1：外设模块基地址 77

表 1-2：引导 ROM 硬件库入口表 79

表 1-3：启动模式 81

表 2-1：POWER 模块基地址 88

表 2-2：POWER 寄存器列表 88

表 2-3：LDO 控制寄存器（LDOCTL）位段定义 89

表 2-4：Power Control Register (LDOCTL) 描述 89

表 2-5：3.3V BOD 控制寄存器（BOD33CTL）位段定义 90

表 2-6：3.3V BOD 控制寄存器（BOD33CTL）位段描述 90

表 2-7：1.2V BOD 控制寄存器（BOD12CTL）位段定义 91

表 2-8：1.2V BOD 控制寄存器（BOD12CTL）位段描述 91

表 2-9：电源模式命令寄存器（PWRMODECMD）位段定义 92

表 2-10：电源模式命令寄存器（PWRMODECMD）位段定义 93

表 2-11：POWER 模块写使能寄存器（PWRREGKEY）位段定义 93

表 2-12：POWER 模块写使能寄存器（PWRREGKEY）位段描述 93

表 3-1：输入分频比设置 98

表 3-2：输出时钟频率和分频比 99

表 3-3：CLOCK Module 基地址 101

表 3-4：CLOCK Register Map 101

表 3-5：时钟状态寄存器（CLKSTS）位段定义 102

©2025 旋智电子科技（上海）有限公司

表 3-6：时钟状态寄存器（CLKSTS）位段描述 102
表 3-7：RCO 控制寄存器（RCOCTL）位段定义 103
表 3-8：RCO 控制寄存器（RCOCTL）位段描述 103
表 3-9：XO 控制寄存器（XOCTL）位段定义 104
表 3-10：XO 控制寄存器（XOCTL）位段描述 104
表 3-11：PLL 控制寄存器 0（PLLCTL0）位段定义 105
表 3-12：PLL 控制寄存器（PLLCTL0）位段描述 106
表 3-13：PLL 控制寄存器 1（PLLCTL1）位段定义 106
表 3-14：PLL 控制寄存器 1（PLLCTL1）位段描述 107
表 3-15：时钟检测控制寄存器（CLKDETCTL）位段定义 107
表 3-16：时钟检测控制寄存器（CLKDETCTL）位段描述 107
表 3-17：时钟检测阈值寄存器（CLKDETCTH）位段定义 109
表 3-18：时钟检测阈值寄存器（CLKDETCTH）位段描述 109
表 3-19：时钟检测计数值寄存器（CLKDETCNT）位段定义 109
表 3-20：时钟检测计数值寄存器（CLKDETCNT）位段描述 109
表 3-21：时钟封锁事件使能寄存器（CLKTZE）位段定义 110
表 3-22：时钟封锁事件使能寄存器（CLKTZE）位段描述 110
表 3-23：系统时钟 0 控制寄存器（SYSCLK0CTL）位段定义 110
表 3-24：系统时钟 0 控制寄存器（SYSCLK0CTL）位段定义 110
表 3-25：消抖时钟控制寄存器（DGCLKCTL）位段定义 111
表 3-26：消抖时钟控制寄存器（DGCLKCTL）位段描述 111

SPINTROL

表 3-27：WDT0 时钟控制寄存器（WDT0CLKCTL）位段定义 112
表 3-28：WDT0 时钟控制寄存器（WDT0CLKCTL）位段描述 112
表 3-29：WDT1 时钟控制寄存器（WDT1CLKCTL）位段定义 112
表 3-30：WDT1 时钟控制寄存器（WDT1CLKCTL）位段描述 113
表 3-31：外设时钟使能寄存器（PERICLKEN）位段定义 113
表 3-32：外设时钟使能寄存器（PERICLKEN）位段描述 113
表 3-33：外设时钟状态寄存器（PERICLKSTS）位段定义 114
表 3-34：外设时钟状态寄存器（PERICLKSTS）位段描述 115
表 3-35：CLOCK 模块写使能寄存器（CLKREGKEY）位段定义 116
表 3-36：CLOCK 模块写使能寄存器（CLKREGKEY）位段描述 116
表 4-1：I/O 引脚功能通道定义 119
表 4-2：调试接口引脚功能 121
表 4-3：PINMUX 模块基地址 127
表 4-4：PINMUX 寄存器列表 127
表 4-5：GPIO0 引脚控制寄存器（GPIO0）位段定义 129
表 4-6：GPIO0 引脚控制寄存器（GPIO0）位段描述 129
表 4-7：GPIO1 引脚控制寄存器（GPIO1）位段定义 130
表 4-8：GPIO1 引脚控制寄存器（GPIO1）位段描述 130
表 4-9：GPIO2 引脚控制寄存器（GPIO2）位段定义 131
表 4-10：GPIO2 引脚控制寄存器（GPIO2）位段描述 131
表 4-11：GPIO3 引脚控制寄存器（GPIO3）位段 Layout 132

©2025 旋智电子科技（上海）有限公司

表 4-12：GPIO3 引脚控制寄存器（GPIO3）位段描述 132
表 4-13：GPIO4 引脚控制寄存器（GPIO4）位段定义 133
表 4-14：GPIO4 引脚控制寄存器（GPIO4）位段描述 133
表 4-15：GPIO5 引脚控制寄存器（GPIO5）位段定义 134
表 4-16：GPIO5 引脚控制寄存器（GPIO5）位段描述 134
表 4-17：GPIO6 引脚控制寄存器（GPIO6）位段定义 135
表 4-18：GPIO6 引脚控制寄存器（GPIO6）位段描述 135
表 4-19：GPIO7 引脚控制寄存器（GPIO7）位段定义 136
表 4-20：GPIO7 引脚控制寄存器（GPIO7）位段描述 136
表 4-21：GPIO8 引脚控制寄存器（GPIO8）位段定义 137
表 4-22：GPIO8 引脚控制寄存器（GPIO8）位段描述 137
表 4-23：GPIO9 引脚控制寄存器（GPIO9）位段定义 138
表 4-24：GPIO9 引脚控制寄存器（GPIO9）位段描述 138
表 4-25：GPIO10 引脚控制寄存器（GPIO10）位段定义 139
表 4-26：GPIO10 引脚控制寄存器（GPIO10）位段描述 139
表 4-27：GPIO11 引脚控制寄存器（GPIO11）位段定义 140
表 4-28：GPIO11 引脚控制寄存器（GPIO11）位段描述 140
表 4-29：GPIO12 引脚控制寄存器（GPIO12）位段定义 141
表 4-30：GPIO12 引脚控制寄存器（GPIO12）位段描述 141
表 4-31：GPIO13 引脚控制寄存器（GPIO13）位段定义 142
表 4-32：GPIO13 引脚控制寄存器（GPIO13）位段描述 142

©2025 旋智电子科技（上海）有限公司

表 4-33：GPIO14 引脚控制寄存器（GPIO14）位段定义 143
表 4-34：GPIO14 引脚控制寄存器（GPIO14）位段描述 143
表 4-35：GPIO15 引脚控制寄存器（GPIO15）位段定义 144
表 4-36：GPIO15 引脚控制寄存器（GPIO15）位段描述 144
表 4-37：GPIO16 引脚控制寄存器（GPIO16）位段定义 145
表 4-38：GPIO16 引脚控制寄存器（GPIO16）位段描述 145
表 4-39：GPIO17 引脚控制寄存器（GPIO17）位段定义 146
表 4-40：GPIO17 引脚控制寄存器（GPIO17）位段描述 146
表 4-41：GPIO18 引脚控制寄存器（GPIO18）位段定义 147
表 4-42：GPIO18 引脚控制寄存器（GPIO18）位段描述 147
表 4-43：GPIO19 引脚控制寄存器（GPIO19）位段定义 148
表 4-44：GPIO19 引脚控制寄存器（GPIO19）位段描述 148
表 4-45：GPIO20 引脚控制寄存器（GPIO20）位段定义 149
表 4-46：GPIO20 引脚控制寄存器（GPIO20）位段描述 149
表 4-47：GPIO21 引脚控制寄存器（GPIO21）位段定义 150
表 4-48：GPIO21 引脚控制寄存器（GPIO21）位段描述 150
表 4-49：GPIO22 引脚控制寄存器（GPIO22）位段定义 151
表 4-50：GPIO22 引脚控制寄存器（GPIO22）位段描述 151
表 4-51：GPIO23 引脚控制寄存器（GPIO23）位段定义 152
表 4-52：GPIO23 引脚控制寄存器（GPIO23）位段描述 152
表 4-53：GPIO24 引脚控制寄存器（GPIO24）位段定义 153

©2025 旋智电子科技（上海）有限公司

表 4-54：GPIO24 引脚控制寄存器（GPIO24）位段描述 153
表 4-55：GPIO25 引脚控制寄存器（GPIO25）位段定义 154
表 4-56：GPIO25 引脚控制寄存器（GPIO25）位段描述 154
表 4-57：GPIO26 引脚控制寄存器（GPIO26）位段定义 155
表 4-58：GPIO26 引脚控制寄存器（GPIO26）位段描述 155
表 4-59：GPIO27 引脚控制寄存器（GPIO27）位段定义 156
表 4-60：GPIO27 引脚控制寄存器（GPIO27）位段描述 156
表 4-61：GPIO28 引脚控制寄存器（GPIO28）位段定义 157
表 4-62：GPIO28 引脚控制寄存器（GPIO28）位段描述 157
表 4-63：GPIO29 引脚控制寄存器（GPIO29）位段定义 158
表 4-64：GPIO29 引脚控制寄存器（GPIO29）位段描述 158
表 4-65：GPIO30 引脚控制寄存器（GPIO30）位段定义 159
表 4-66：GPIO30 引脚控制寄存器（GPIO30）位段描述 159
表 4-67：GPIO31 引脚控制寄存器（GPIO31）位段定义 160
表 4-68：GPIO31 引脚控制寄存器（GPIO31）位段描述 160
表 4-69：GPIO32 引脚控制寄存器（GPIO32）位段定义 161
表 4-70：GPIO32 引脚控制寄存器（GPIO32）位段描述 161
表 4-71：GPIO33 引脚控制寄存器（GPIO33）位段定义 162
表 4-72：GPIO33 引脚控制寄存器（GPIO33）位段描述 162
表 4-73：GPIO34 引脚控制寄存器（GPIO34）位段定义 163
表 4-74：GPIO34 引脚控制寄存器（GPIO34）位段描述 163

©2025 旋智电子科技（上海）有限公司

表 4-75：GPIO35 引脚控制寄存器（GPIO35）位段定义 164
表 4-76：GPIO35 引脚控制寄存器（GPIO35）位段描述 164
表 4-77：GPIO36 引脚控制寄存器（GPIO36）位段定义 165
表 4-78：GPIO36 引脚控制寄存器（GPIO36）位段描述 165
表 4-79：GPIO37 引脚控制寄存器（GPIO37）位段定义 166
表 4-80：GPIO37 引脚控制寄存器（GPIO37）位段描述 166
表 4-81：GPIO38 引脚控制寄存器（GPIO38）位段定义 167
表 4-82：GPIO38 引脚控制寄存器（GPIO38）位段描述 167
表 4-83：GPIO39 引脚控制寄存器（GPIO39）位段定义 168
表 4-84：GPIO39 引脚控制寄存器（GPIO39）位段描述 168
表 4-85：GPIO40 引脚控制寄存器（GPIO40）位段定义 169
表 4-86：GPIO40 引脚控制寄存器（GPIO40）位段描述 169
表 4-87：比较器输出监测控制寄存器（COMPMONCTL）位段定义 170
表 4-88：比较器输出监测控制寄存器（COMPMONCTL）位段描述 170
表 4-89：PINMUX 模块写使能寄存器（PINMUXREGKEY）位段定义 172
表 4-90：PINMUX 模块写使能寄存器（PINMUXREGKEY）位段描述 172
表 4-91：GPIO 模块基地址 173
表 4-92：GPIO 寄存器列表 173
表 4-93：GPIO 引脚电平寄存器（GPLR[n] ([0:1])）位段定义 175
表 4-94：GPIO 引脚电平寄存器（GPLR[n] ([0:1])）位段描述 175
表 4-95：GPIO 按位输出置高（GSLR[n] ([0:1])）位段定义 175

©2025 旋智电子科技（上海）有限公司

表 4-96：GPIO 按位输出置高（GSLR[n] ([0:1])）位段描述 175
表 4-97：GPIO 按位输出置低寄存器（GCLR[n] ([0:1])）位段定义 176
表 4-98：GPIO 按位输出置低寄存器（GCLR[n] ([0:1])）位段描述 176
表 4-99：GPIO 引脚方向寄存器（GPDR[n] ([0:1])）位段定义 176
表 4-100：GPIO 引脚方向寄存器（GPDR[n] ([0:1])）位段描述 176
表 4-101：GPIO 按位设置输出方向寄存器（GSDR[n] ([0:1])）位段定义 177
表 4-102：GPIO 按位设置输出方向寄存器（GSDR[n] ([0:1])）位段描述 177
表 4-103：GPIO 按位设置输入方向寄存器（GCDR[n] ([0:1])）位段定义 177
表 4-104：GPIO 按位设置输入方向寄存器（GCDR[n] ([0:1])）位段描述 177
表 4-105：GPIO 上升沿检测使能寄存器（GRER[n] ([0:1])）位段定义 178
表 4-106：GPIO 上升沿检测使能寄存器（GRER[n] ([0:1])）位段描述 178
表 4-107：GPIO 上升沿检测按位使能寄存器（GSRE[n] ([0:1])）位段定义 178
表 4-108：GPIO 上升沿检测按位使能寄存器（GSRE[n] ([0:1])）位段描述 178
表 4-109：GPIO 上升沿检测按位关闭寄存器（GCRE[n] ([0:1])）位段定义 179
表 4-110：GPIO 上升沿检测按位关闭寄存器（GCRE[n] ([0:1])）位段描述 179
表 4-111：GPIO 下降沿检测使能寄存器（GFER[n] ([0:1])）位段定义 179
表 4-112：GPIO 下降沿检测使能寄存器（GFER[n] ([0:1])）位段描述 179
表 4-113：GPIO 下降沿检测按位使能寄存器（GSFE[n] ([0:1])）位段定义 180
表 4-114：GPIO 下降沿检测按位使能寄存器（GSFE[n] ([0:1])）位段描述 180
表 4-115：GPIO 下降沿检测按位关闭寄存器（GCFE[n] ([0:1])）位段定义 180
表 4-116：GPIO 下降沿检测按位关闭寄存器（GCFE[n] ([0:1])）位段描述 180

SPINTROL

表 4-117：GPIO 上升沿事件标志寄存器（GRERAWIF[n] ([0:1])）位段定义 181
表 4-118：GPIO 上升沿事件标志寄存器（GRERAWIF[n] ([0:1])）位段描述 181
表 4-119：GPIO 下降沿事件标志寄存器（GFERAWIF[n] ([0:1])）位段定义 181
表 4-120：GPIO 下降沿事件标志寄存器（GFERAWIF[n] ([0:1])）位段描述 181
表 4-121：GPIO 上升沿中断标志寄存器（GREIF[n] ([0:1])）位段定义 182
表 4-122：GPIO 上升沿中断标志寄存器（GREIF[n] ([0:1])）位段描述 182
表 4-123：GPIO 下降沿中断标志寄存器（GFEIF[n] ([0:1])）位段定义 182
表 4-124：GPIO 下降沿中断标志寄存器（GFEIF[n] ([0:1])）位段描述 182
表 4-125：GPIO 上升沿中断使能寄存器（GREIE[n] ([0:1])）位段定义 183
表 4-126：GPIO 上升沿中断使能寄存器（GREIE[n] ([0:1])）位段描述 183
表 4-127：GPIO 下降沿中断使能寄存器（GFEIE[n] ([0:1])）位段定义 183
表 4-128：GPIO 下降沿中断使能寄存器（GFEIE[n] ([0:1])）位段描述 183
表 4-129：GPIO 上升沿中断清除寄存器（GREIC[n] ([0:1])）位段定义 184
表 4-130：GPIO 上升沿中断清除寄存器（GREIC[n] ([0:1])）位段定义 184
表 4-131：GPIO 下降沿中断清除寄存器（GFEIC[n] ([0:1])）位段定义 184
表 4-132：GPIO 下降沿中断清除寄存器（GFEIC[n] ([0:1])）位段描述 184
表 4-133：GPIO 上升沿事件强制寄存器（GREIFRC[n] ([0:1])）位段定义 185
表 4-134：GPIO 上升沿事件强制寄存器（GREIFRC[n] ([0:1])）位段描述 185
表 4-135：GPIO 下降沿事件强制寄存器（GFEIFRC[n] ([0:1])）位段定义 185
表 4-136：GPIO 下降沿事件强制寄存器（GFEIFRC[n] ([0:1])）位段描述 185
表 4-137：GPIO 电平检测使能寄存器（GLDE[n] ([0:1])）位段定义 186

©2025 旋智电子科技（上海）有限公司

表 4-138：GPIO 电平检测使能寄存器（GLDE[n] ([0:1])）位段描述 186
表 4-139：GPIO 电平检测按位使能寄存器（GSLDE[n] ([0:1])）位段定义 186
表 4-140：GPIO 电平检测按位使能寄存器（GSLDE[n] ([0:1])）位段描述 186
表 4-141：GPIO 电平检测按位关闭寄存器（GCLDE[n] ([0:1])）位段定义 187
表 4-142：GPIO 电平检测按位关闭寄存器（GCLDE[n] ([0:1])）位段描述 187
表 4-143：GPIO 电平检测极性寄存器（GLPOL[n] ([0:1])）位段定义 187
表 4-144：GPIO 电平检测极性寄存器（GLPOL[n] ([0:1])）位段定义 187
表 4-145：GPIO 电平事件标志寄存器（GLRAWIF[n] ([0:1])）位段定义 188
表 4-146：GPIO 电平事件标志寄存器（GLRAWIF[n] ([0:1])）位段描述 188
表 4-147：GPIO 电平中断标志寄存器（GLIF[n] ([0:1])）位段定义 188
表 4-148：GPIO 电平中断标志寄存器（GLIF[n] ([0:1])）位段描述 188
表 4-149：GPIO 电平中断使能寄存器（GLIE[n] ([0:1])）位段定义 189
表 4-150：GPIO 电平中断使能寄存器（GLIE[n] ([0:1])）位段描述 189
表 4-151：GPIO 电平中断清除寄存器（GLIC[n] ([0:1])）位段定义 189
表 4-152：GPIO 电平中断清除寄存器（GLIC[n] ([0:1])）位段描述 189
表 4-153：GPIO 电平事件强制寄存器（GLIFRC[n] ([0:1])）位段定义 190
表 4-154：GPIO 电平事件强制寄存器（GLIFRC[n] ([0:1])）位段描述 190
表 4-155：GPIO 全局事件标志（GGRAWIF）位段定义 190
表 4-156：GPIO 全局事件标志寄存器（GGRAWIF）位段描述 190
表 4-157：GPIO 全局中断标志寄存器（GGIF）位段定义 191
表 4-158：GPIO 全局中断标志寄存器（GGIF）位段描述 191

©2025 旋智电子科技（上海）有限公司

表 4-159：GPIO 全局中断清除寄存器（GGIC）位段定义 192
表 4-160：GPIO 全局中断清除（GGIC）位段描述 192
表 4-161：GPIO 模块写使能寄存器（GPIOREGKEY）位段定义 192
表 4-162：GPIO 模块写使能寄存器（GPIOREGKEY）位段描述 192
表 5-1：SPC1169 中断向量表 193
表 6-1：Timer 模块基地址 200
表 6-2：TIMER 寄存器列表 200
表 6-3：定时器加载值寄存器（TMRLOAD）位段定义 201
表 6-4：定时器加载值寄存器（TMRLOAD）位段描述 201
表 6-5：定时器计数器值寄存器（TMRCNT）位段定义 201
表 6-6：定时器计数器值寄存器（TMRCNT）位段描述 201
表 6-7：定时器控制寄存器（TMRCTL）位段定义 202
表 6-8：定时器控制寄存器（TMRCTL）位段描述 202
表 6-9：定时器中断原始事件标志寄存器（TMRRAWIF）位段定义 203
表 6-10：定时器中断原始事件标志寄存器（TMRRAWIF）位段描述 203
表 6-11：定时器中断标志寄存器（TMRIF）位段定义 203
表 6-12：定时器中断标志寄存器（TMRIF）位段描述 204
表 6-13：定时器中断清除寄存器（TMRIC）位段定义 204
表 6-14：定时器中断清除寄存器（TMRIC）位段描述 204
表 6-15：定时器中断使能寄存器（TMRIE）位段定义 204
表 6-16：定时器中断使能寄存器（TMRIE）位段描述 205

©2025 旋智电子科技（上海）有限公司

表 6-17：定时器模块写使能寄存器（TMRREGGKEY）位段定义 205
表 6-18：定时器模块写使能寄存器（TMRREGGKEY）位段描述 205
表 7-1：WDT 模块基地址 208
表 7-2：WDT 寄存器列表 208
表 7-3：看门狗定时器加载值寄存器（WDTLOAD）位段定义 209
表 7-4：看门狗定时器加载值寄存器（WDTLOAD）位段描述 209
表 7-5：看门狗定时器计数值寄存器（WDTCNT）位段定义 209
表 7-6：看门狗定时器计数值寄存器（WDTCNT）位段描述 209
表 7-7：看门狗定时器控制寄存器（WDTCTL）位段定义 210
表 7-8：看门狗定时器控制寄存器（WDTCTL）位段描述 210
表 7-9：看门狗定时器中断原始事件标志寄存器（WDTRAWIF）位段定义 210
表 7-10：看门狗定时器中断原始事件标志寄存器（WDTRAWIF）位段描述 211
表 7-11：看门狗定时器中断标志寄存器（WDTIF）位段定义 211
表 7-12：看门狗定时器中断标志寄存器（WDTIF）位段描述 211
表 7-13：看门狗定时器中断清除寄存器（WDTIC）位段定义 211
表 7-14：看门狗定时器中断清除寄存器（WDTIC）位段描述 212
表 7-15：看门狗定时器中断使能寄存器（WDTIE）位段定义 212
表 7-16：看门狗定时器中断使能寄存器（WDTIE）位段描述 212
表 7-17：看门狗定时器服务窗口低阈值寄存器（WDTTHL）位段定义 213
表 7-18：看门狗定时器服务窗口低阈值寄存器（WDTTHL）位段描述 213
表 7-19：看门狗定时器服务窗口高阈值寄存器（WDTTHL）位段定义 213

©2025 旋智电子科技（上海）有限公司

表 7-20：看门狗定时器服务窗口高阈值寄存器（WDTTHL）位段描述 213

表 7-21：看门狗定时器模块写使能寄存器（WDTREGKEY）位段定义 214

表 7-22：看门狗定时器模块写使能寄存器（WDTREGKEY）位段描述 214

表 8-1：向上计数时的行为限定事件优先级 223

表 8-2：向下计数时的行为限定事件优先级 223

表 8-3：上下计数时的行为限定事件优先级 223

表 8-4：典型的死区控制工作模式 225

表 8-5：影响 PWM 最终输出波形的事件优先级 229

表 8-6：PWM 模块基地址 235

表 8-7：PWM 寄存器列表 235

表 8-8：影子状态寄存器（SHADOWSTS）位段定义 237

表 8-9：影子状态寄存器（SHADOWSTS）位段描述 237

表 8-10：全局影子值重置有效值控制寄存器 0（GLDCTL0）位段定义 238

表 8-11：全局影子值重置有效值控制寄存器 0（GLDCTL0）位段描述 240

表 8-12：全局影子值重置有效值控制寄存器 1（GLDCTL1）位段定义 241

表 8-13：全局影子值重置有效值控制寄存器 1（GLDCTL1）位段描述 241

表 8-14：全局影子值重置有效值控制选择寄存器（GLDSEL）位段定义 241

表 8-15：全局影子值重置有效值控制选择寄存器（GLDSEL）位段描述 242

表 8-16：PWM 链接控制寄存器（PWMLINK）位段定义 243

表 8-17：PWM 链接控制寄存器（PWMLINK）位段描述 243

表 8-18：时基控制寄存器（TBCTL）位段定义 245

©2025 旋智电子科技（上海）有限公司

表 8-19：时基控制寄存器（TBCTL）位段描述 245
表 8-20：时基周期寄存器（TBPRD）位段定义 247
表 8-21：时基周期寄存器（TBPRD）位段描述 247
表 8-22：时基周期有效值寄存器（TBPRDA）位段定义 247
表 8-23：时基周期有效值寄存器（TBPRDA）位段描述 249
表 8-24：时基相位寄存器（TBPHS）位段定义 249
表 8-25：时基相位寄存器（TBPHS）位段描述 249
表 8-26：时基计数值寄存器（TBCNT）位段定义 250
表 8-27：时基计数值寄存器（TBCNT）位段描述 250
表 8-28：时基标志寄存器（TBFLG）位段定义 250
表 8-29：时基标志寄存器（TBFLG）位段描述 250
表 8-30：时基标志清除寄存器（TBFLGCLR）位段定义 251
表 8-31：时基标志清除寄存器（TBFLGCLR）位段描述 252
表 8-32：计数比较控制寄存器（CMPCTL）位段定义 252
表 8-33：计数比较控制寄存器（CMPCTL）位段描述 253
表 8-34：计数比较阈值 A 寄存器（CMPA）位段定义 254
表 8-35：计数比较阈值 A 寄存器（CMPA）位段描述 256
表 8-36：计数比较阈值 A 有效值寄存器（CMPAA）位段描述 256
表 8-37：计数比较阈值 A 有效值寄存器（CMPAA）位段描述 256
表 8-38：计数比较阈值 B 寄存器（CMPB）位段定义 256
表 8-39：计数比较阈值 B 寄存器（CMPB）位段描述 257

©2025 旋智电子科技（上海）有限公司

表 8-40：计数比较阈值 B 寄存器（CMPBA）位段定义 257
表 8-41：计数比较阈值 B 寄存器（CMPBA）位段描述 257
表 8-42：计数比较阈值 C 寄存器（CMPC）位段定义 257
表 8-43：计数比较阈值 C 寄存器（CMPC）位段描述 258
表 8-44：计数比较阈值 C 有效值寄存器（CMPCA）位段定义 258
表 8-45：计数比较阈值 C 有效值寄存器（CMPCA）位段描述 258
表 8-46：计数比较阈值 D 寄存器（CMPD）位段定义 258
表 8-47：计数比较阈值 D 寄存器（CMPD）位段描述 258
表 8-48：计数比较阈值 D 有效值寄存器（CMPDA）位段定义 259
表 8-49：计数比较阈值 D 有效值寄存器（CMPDA）位段描述 259
表 8-50：行为限定控制寄存器（AQCTL）位段定义 259
表 8-51：行为限定控制寄存器（AQCTL）位段描述 259
表 8-52：行为限定 A 路输出控制寄存器（AQCTLA）位段定义 261
表 8-53：行为限定 A 路输出控制寄存器（AQCTLA）位段描述 261
表 8-54：行为限定 A 路输出控制有效值寄存器（AQCTLAA）位段定义 263
表 8-55：行为限定 A 路输出控制有效值寄存器（AQCTLAA）位段描述 263
表 8-56：行为限定 B 路输出控制寄存器（AQCTLB）位段定义 265
表 8-57：行为限定 B 路输出控制寄存器（AQCTLB）位段描述 265
表 8-58：行为限定 B 路输出控制有效值寄存器（AQCTLBA）位段定义 267
表 8-59：行为限定 B 路输出控制有效值寄存器（AQCTLBA）位段描述 267
表 8-60：行为限定软件强制寄存器（AQSFRC）位段定义 269

©2025 旋智电子科技（上海）有限公司

表 8-61：行为限定软件强制寄存器（AQSFRC）位段描述 269
表 8-62：行为限定软件持续强制寄存器（AQCSFRC）位段定义 270
表 8-63：行为限定软件持续强制寄存器（AQCSFRC）位段描述 270
表 8-64：行为限定软件持续强制有效值寄存器（AQCSFRCA）位段定义 270
表 8-65：行为限定软件持续强制有效值寄存器（AQCSFRCA）位段描述 271
表 8-66：死区产生控制寄存器（DBCTL）位段定义 271
表 8-67：死区产生控制寄存器（DBCTL）位段描述 271
表 8-68：死区产生控制有效值寄存器（DBCTLA）位段定义 273
表 8-69：死区产生控制有效值寄存器（DBCTLA）位段描述 275
表 8-70：死区产生上升沿延迟量寄存器（DBRED）位段定义 276
表 8-71：死区产生上升沿延迟量寄存器（DBRED）位段描述 276
表 8-72：死区产生上升沿延迟量有效值寄存器（DBREDA）位段定义 276
表 8-73：死区产生上升沿延迟量有效值寄存器（DBREDA）位段描述 276
表 8-74：死区产生下降沿延迟量寄存器（DBFED）位段定义 277
表 8-75：死区产生下降沿延迟量寄存器（DBFED）位段描述 277
表 8-76：死区产生下降沿延迟量有效值寄存器（DBFEDA）位段定义 277
表 8-77：死区产生下降沿延迟量有效值寄存器（DBFEDA）位段描述 277
表 8-78：封锁事件选择寄存器（TZSEL）位段定义 278
表 8-79：封锁事件选择寄存器（TZSEL）位段描述 278
表 8-80：封锁标志寄存器（TZFLG）位段定义 280
表 8-81：封锁标志寄存器（TZFLG）位段描述 280

©2025 旋智电子科技（上海）有限公司

表 8-82：封锁状态清除寄存器（TZSTSCLR）位段定义 282
表 8-83：封锁状态清除寄存器（TZSTSCLR）位段描述 282
表 8-84：数字比较封锁事件选择寄存器（TZDCSEL）位段定义 284
表 8-85：数字比较封锁事件选择寄存器（TZDCSEL）位段描述 284
表 8-86：封锁 A 路输出控制寄存器（TZACTL）位段定义 286
表 8-87：封锁 A 路输出控制寄存器（TZACTL）位段描述 286
表 8-88：封锁 B 路输出控制寄存器（TZBCTL）位段定义 287
表 8-89：封锁 B 路输出控制寄存器（TZBCTL）位段描述 287
表 8-90：封锁中断标志寄存器（TZIF）位段定义 288
表 8-91：封锁中断标志寄存器（TZIF）位段描述 290
表 8-92：封锁中断清除寄存器（TZIC）位段定义 291
表 8-93：封锁中断清除寄存器（TZIC）位段描述 291
表 8-94：封锁中断使能寄存器（TZIE）位段定义 292
表 8-95：封锁中断使能寄存器（TZIE）位段描述 292
表 8-96：封锁中断强制寄存器（TZFRC）位段定义 293
表 8-97：封锁中断强制寄存器（TZFRC）位段描述 293
表 8-98：数字比较 DCAL 事件选择寄存器（DCALTRIPSEL）位段定义 295
表 8-99：数字比较 DCAL 事件选择寄存器（DCALTRIPSEL）位段描述 295
表 8-100：数字比较 DCAH 事件选择寄存器（DCAHTRIPSEL）位段定义 296
表 8-101：数字比较 DCAH 事件选择寄存器（DCAHTRIPSEL）位段描述 296
表 8-102：数字比较 DCBL 事件选择寄存器（DCBLTRIPSEL）位段定义 297

©2025 旋智电子科技（上海）有限公司

表 8-103：数字比较 DCBL 事件选择寄存器（DCBLTRIPSEL）位段描述 298
表 8-104：数字比较 DCBH 事件选择寄存器（DCBHTRIPSEL）位段定义 299
表 8-105：数字比较 DCBH 事件选择寄存器（DCBHTRIPSEL）位段描述 299
表 8-106：数字比较 A 控制寄存器（DCACTL）位段定义 300
表 8-107：数字比较 A 控制寄存器（DCACTL）位段描述 300
表 8-108：数字比较 B 控制寄存器（DCBCTL）位段定义 301
表 8-109：数字比较 B 控制寄存器（DCBCTL）位段描述 301
表 8-110：数字比较消隐滤波控制寄存器（DCFCTL）位段定义 302
表 8-111：数字比较消隐滤波控制寄存器（DCFCTL）位段描述 302
表 8-112：数字比较消隐窗口偏移量寄存器（DCFOFFSET）位段定义 304
表 8-113：数字比较消隐窗口偏移量寄存器（DCFOFFSET）位段描述 304
表 8-114：数字比较消隐窗口偏移计数值寄存器（DCFOFFSETCNT）位段定义 305
表 8-115：数字比较消隐窗口偏移计数值寄存器（DCFOFFSETCNT）位段描述 306
表 8-116：数字比较消隐窗口大小寄存器（DCFWINDOW）位段定义 306
表 8-117：数字比较消隐窗口大小寄存器（DCFWINDOW）位段描述 306
表 8-118：数字比较消隐窗口大小计数寄存器（DCFWINDOWCNT）位段定义 307
表 8-119：数字比较消隐窗口大小计数寄存器（DCFWINDOWCNT）位段描述 307
表 8-120：数字比较时基计数捕获控制寄存器（DCCAPCTL）位段定义 307
表 8-121：数字比较时基计数捕获控制寄存器（DCCAPCTL）位段描述 307
表 8-122：数字比较时基计数捕获值寄存器（DCCAP）位段定义 309
表 8-123：数字比较时基计数捕获值寄存器（DCCAP）位段描述 309

SPINTROL

表 8-124：事件触发控制寄存器（ETCTL）位段定义 309
表 8-125：事件触发控制寄存器（ETCTL）位段描述 310
表 8-126：事件触发预分频寄存器（ETPS）位段定义 312
表 8-127：事件触发预分频寄存器（ETPS）位段描述 312
表 8-128：事件触发标志寄存器（ETFLG）位段定义 314
表 8-129：事件触发标志寄存器（ETFLG）位段描述 314
表 8-130：事件触发标志清除寄存器（ETCLR）位段描述 315
表 8-131：事件触发标志清除寄存器（ETCLR）位段描述 315
表 8-132：事件触发软件强制寄存器（ETFRC）位段定义 317
表 8-133：事件触发软件强制寄存器（ETFRC）位段描述 317
表 8-134：PWM 模块写使能寄存器（PWMREGKEY）位段定义 318
表 8-135：PWM 模块写使能寄存器（PWMREGKEY）位段描述 318
表 8-136：PWMCFG 模块基地址 319
表 8-137：PWMCFG 寄存器列表 319
表 8-138：TZ0 信号源控制寄存器（TZ0SRCCTL）位段定义 320
表 8-139：TZ0 信号源控制寄存器（TZ0SRCCTL）位段描述 320
表 8-140：TZ1 信号源控制寄存器（TZ1SRCCTL）位段定义 320
表 8-141：TZ1 信号源控制寄存器（TZ1SRCCTL）位段描述 320
表 8-142：TZ2 信号源控制寄存器（TZ2SRCCTL）位段定义 321
表 8-143：TZ2 信号源控制寄存器（TZ2SRCCTL）位段描述 321
表 8-144：TZ3 信号源控制寄存器（TZ3SRCCTL）位段定义 321

©2025 旋智电子科技（上海）有限公司

表 8-145：TZ3 信号源控制寄存器（TZ3SRCCTL）位段描述 322
表 8-146：TZ4 信号源控制寄存器（TZ4SRCCTL）位段定义 322
表 8-147：TZ4 信号源控制寄存器（TZ4SRCCTL）位段描述 322
表 8-148：全局 PWM 软件强制同步寄存器（FRCSYNC）位段定义 322
表 8-149：全局 PWM 软件强制同步寄存器（FRCSYNC）位段描述 323
表 8-150：GPIO 触发 PWMSYNCI 信号源控制寄存器（GPIOSYNCICTL）位段定义 323
表 8-151：GPIO 触发 PWMSYNCI 信号源控制寄存器（GPIOSYNCICTL）位段描述 323
表 8-152：GPIO 触发 PWMSYNCI 使能寄存器（GPIOSYNCIEN）位段定义 324
表 8-153：GPIO 触发 PWMSYNCI 使能寄存器（GPIOSYNCIEN）位段描述 324
表 8-154：TIMER0 触发 PWMSYNCI 使能寄存器（TMR0SYNCIEN）位段定义 325
表 8-155：TIMER0 触发 PWMSYNCI 使能寄存器（TMR0SYNCIEN）位段描述 325
表 8-156：TIMER1 触发 PWMSYNCI 使能寄存器（TMR1SYNCIEN）位段定义 325
表 8-157：TIMER1 触发 PWMSYNCI 使能寄存器（TMR1SYNCIEN）位段描述 326
表 8-158：TIMER2 触发 PWMSYNCI 使能寄存器（TMR2SYNCIEN）位段定义 326
表 8-159：TIMER2 触发 PWMSYNCI 使能寄存器（TMR2SYNCIEN）位段描述 326
表 8-160：PWMSYNCO 输出控制寄存器（SYNCOCTL）位段定义 327
表 8-161：PWMSYNCO 输出控制寄存器（SYNCOCTL）位段描述 327
表 8-162：PWMSOCA 输出控制寄存器（SOCAOCTL）位段定义 328
表 8-163：PWMSOCA 输出控制寄存器（SOCAOCTL）位段描述 328
表 8-164：PWMSOCB 输出控制寄存器（SOCBOCTL）位段定义 329
表 8-165：PWMSOCB 输出控制寄存器（SOCBOCTL）位段描述 329

©2025 旋智电子科技（上海）有限公司

表 8-166：PWMSOCC 输出控制寄存器（SOCCOCTL）位段定义 330
表 8-167：PWMSOCC 输出控制寄存器（SOCCOCTL）位段描述 330
表 8-168：PWMCFG 模块写使能寄存器（PWMCFG）位段定义 331
表 8-169：PWMCFG 模块写使能寄存器（PWMCFG）位段描述 331
表 9-1：ECAP 模块基地址 345
表 9-2：ECAP 寄存器列表 345
表 9-3：捕获信号输入控制寄存器（CAPSRCCTL）位段定义 346
表 9-4：捕获信号输入控制寄存器（CAPSRCCTL）位段描述 346
表 9-5：同步信号输入控制寄存器（CAPSYNCICTL）位段定义 346
表 9-6：同步信号输入控制寄存器（CAPSYNCICTL）位段描述 347
表 9-7：时间戳计数寄存器（CAPTSCNT）位段定义 347
表 9-8：时间戳计数寄存器（CAPTSCNT）位段描述 347
表 9-9：时间戳计数相位寄存器（CAPCNTPHS）位段定义 347
表 9-10：时间戳计数相位寄存器（CAPCNTPHS）位段描述 348
表 9-11：捕获值 0 寄存器（CAP0）位段定义 348
表 9-12：捕获值 0 寄存器（CAP0）位段描述 348
表 9-13：捕获值 1 寄存器（CAP1）位段定义 348
表 9-14：捕获值 1 寄存器（CAP1）位段描述 349
表 9-15：捕获值 2 寄存器（CAP2）位段定义 349
表 9-16：捕获值 2 寄存器（CAP2）位段描述 349
表 9-17：捕获值 3 寄存器（CAP3）位段定义 350

©2025 旋智电子科技（上海）有限公司

表 9-18：捕获值 3 寄存器（CAP3）位段描述 350
表 9-19：捕获控制寄存器（CAPCTL）位段定义 350
表 9-20：捕获控制寄存器（CAPCTL）位段描述 350
表 9-21：捕获中断原始事件标志寄存器（CAPRAWIF）位段定义 354
表 9-22：捕获中断原始事件标志寄存器（CAPRAWIF）位段定义 354
表 9-23：捕获中断标志寄存器（CAPIF）位段定义 355
表 9-24：捕获中断标志寄存器（CAPIF）位段描述 355
表 9-25：捕获中断使能寄存器（CAPIE）位段定义 356
表 9-26：捕获中断使能寄存器（CAPIE）位段描述 357
表 9-27：捕获中断清除寄存器（CAPIC）位段定义 358
表 9-28：捕获中断清除寄存器（CAPIC）位段描述 358
表 9-29：捕获中断强制寄存器（CAPIFRC）位段定义 359
表 9-30：捕获中断强制寄存器（CAPIFRC）位段描述 359
表 9-31：ECAP 模块写使能寄存器（CAPREGKEY）位段定义 362
表 9-32：ECAP 模块写使能寄存器（CAPREGKEY）位段描述 362
表 10-1：ADC 输入选择 366
表 10-2：ADC 模块基地址 377
表 10-3：ADC 寄存器列表 377
表 10-4：ADC 中断标志寄存器（ADCIF）位段定义 379
表 10-5：ADC 中断标志寄存器（ADCIF）位段描述 379
表 10-6：ADC 中断清除寄存器（ADCIC）位段定义 381

SPINTROL

表 10-7：ADC 中断清除寄存器（ADCIC）位段描述 381
表 10-8：ADC 中断溢出标志寄存器（ADCIOVF）位段定义 383
表 10-9：ADC 中断溢出标志寄存器（ADCIOVF）位段描述 383
表 10-10：ADC 中断溢出清除寄存器（ADCIOVFC）位段定义 384
表 10-11：ADC 中断溢出清除寄存器（ADCIOVFC）位段描述 385
表 10-12：ADC 中断使能寄存器（ADCIE）位段定义 386
表 10-13：ADC 中断使能寄存器（ADCIE）位段描述 387
表 10-14：ADC SOC 优先级控制寄存器（ADCSOCPRICTL）位段定义 389
表 10-15：ADC SOC 优先级控制寄存器（ADCSOCPRICTL）位段描述 389
表 10-16：ADC SOC 请求标志寄存器（ADCSOCFLG）位段定义 390
表 10-17：ADC SOC 请求标志寄存器（ADCSOCFLG）位段描述 390
表 10-18：ADC SOC 请求强制寄存器（ADCSOCFRC）位段定义 392
表 10-19：ADC SOC 请求强制寄存器（ADCSOCFRC）位段描述 392
表 10-20：ADC SOC 请求溢出标志寄存器（ADCSOCOVF）位段定义 393
表 10-21：ADC SOC 请求溢出标志寄存器（ADCSOCOVF）位段描述 394
表 10-22：ADC SOC 请求溢出标志寄存器（ADCSOCOVFC）位段定义 396
表 10-23：ADC SOC 请求溢出标志寄存器（ADCSOCOVFC）位段描述 396
表 10-24：ADC EOC 中断事件触发 SOC 使能寄存器（ADCINTSOCEN）位段定义 397
表 10-25：ADC EOC 中断事件触发 SOC 使能寄存器（ADCINTSOCEN）位段描述 398
表 10-26：ADC EOC 中断事件触发 SOC 选择寄存器（ADCINTSOCSEL）位段定义 399
表 10-27：ADC EOC 中断事件触发 SOC 选择寄存器（ADCINTSOCSEL）位段描述 400

©2025 旋智电子科技（上海）有限公司

表 10-28：ADC 外部 SOC 输入控制寄存器（ADCEXTSOCCTL）位段定义 400
表 10-29：ADC 外部 SOC 输入控制寄存器（ADCEXTSOCCTL）位段描述 400
表 10-30：ADC 通道 n ([0:15]) 控制寄存器（ADCCHCTL）位段定义 401
表 10-31：ADC 通道 n ([0:15]) 控制寄存器（ADCCHCTL）位段描述 401
表 10-32：ADC 通道 n ([0:7]) 偏移校准寄存器（ADCOFFSET）位段定义 403
表 10-33：ADC 通道 n ([0:7]) 偏移校准寄存器（ADCOFFSET）位段描述 403
表 10-34：ADC 通道 n ([0:7]) 增益校准寄存器（ADCGAIN）位段定义 403
表 10-35：ADC 通道 n ([0:7]) 增益校准寄存器（ADCGAIN）位段描述 403
表 10-36：ADC 采样转换核心电路偏移校准寄存器（SHOFFSET）位段定义 404
表 10-37：ADC 采样转换核心电路偏移校准寄存器（SHOFFSET）位段描述 404
表 10-38：ADC 采样转换核心电路增益校准寄存器（SHGAIN）位段定义 404
表 10-39：ADC 采样转换核心电路增益校准寄存器（SHGAIN）位段描述 404
表 10-40：ADC 采样转换 EOC 标志寄存器（SHFLG）位段定义 405
表 10-41：ADC 采样转换 EOC 标志寄存器（SHFLG）位段描述 405
表 10-42：ADC 采样转换 EOC 标志寄存器（SHFLGCLR）位段定义 405
表 10-43：ADC 采样转换 EOC 标志寄存器（SHFLGCLR）位段描述 405
表 10-44：ADC 控制寄存器（ADCCTL）位段定义 406
表 10-45：ADC 控制寄存器（ADCCTL）位段描述 406
表 10-46：ADC 采样转换原始码值（SHRAWCODE）位段定义 407
表 10-47：ADC 采样转换原始码值（SHRAWCODE）位段描述 407
表 10-48：ADC 通道 n ([0:15])结果寄存器（ADCRESULT）位段定义 408

©2025 旋智电子科技（上海）有限公司

表 10-49：ADC 通道 n ([0:15])结果寄存器（ADCRESULT）位段描述 408
表 10-50：后处理单元 n ([0:2])比较结果寄存器（PPURESULT）位段定义 408
表 10-51：后处理单元 n ([0:2])比较结果寄存器（PPURESULT）位段描述 408
表 10-52：后处理单元 n ([0:2]) SOC 响应延时寄存器（PPUSOCDLY）位段定义 409
表 10-53：后处理单元 n ([0:2]) SOC 响应延时寄存器（PPUSOCDLY）位段描述 409
表 10-54：后处理单元 n ([0:2]) 中断原始事件标志寄存器（PPURAWIF）位段定义..... 409
表 10-55：后处理单元 n ([0:2]) 中断原始事件标志寄存器（PPURAWIF）位段描述..... 409
表 10-56：后处理单元 n ([0:2]) 中断标志寄存器（PPUIF）位段定义..... 410
表 10-57：后处理单元 n ([0:2]) 中断标志寄存器（PPUIF）位段描述..... 410
表 10-58：后处理单元 n ([0:2]) 中断清除寄存器（PPUIC）位段定义 411
表 10-59：后处理单元 n ([0:2]) 中断清除寄存器（PPUIC）位段描述 411
表 10-60：后处理单元 n ([0:2]) 中断使能寄存器（PPUIE）位段定义 411
表 10-61：后处理单元 n ([0:2]) 中断使能寄存器（PPUIE）位段描述 412
表 10-62：后处理单元 n ([0:2]) 封锁事件使能寄存器（PPUTZE）位段定义..... 412
表 10-63：后处理单元 n ([0:2]) 封锁事件使能寄存器（PPUTZE）位段描述..... 412
表 10-64：后处理单元 n ([0:2]) 控制寄存器（PPUCTL）位段定义..... 413
表 10-65：后处理单元 n ([0:2]) 控制寄存器（PPUCTL）位段描述..... 413
表 10-66：后处理单元 n ([0:2]) 比较参考寄存器（PPUREF）位段定义 413
表 10-67：后处理单元 n ([0:2]) 比较参考寄存器（PPUREF）位段描述 414
表 10-68：后处理单元 n ([0:2]) 下限阈值寄存器（PPUTHL）位段定义 414
表 10-69：后处理单元 n ([0:2]) 下限阈值寄存器（PPUTHL）位段描述 414

©2025 旋智电子科技（上海）有限公司

表 10-70：后处理单元 n ([0:2]) 上限阈值寄存器（PPUTHH）位段定义 414
表 10-71：后处理单元 n ([0:2]) 上限阈值寄存器（PPUTHH）位段描述 414
表 10-72：温度传感器控制寄存器（TSENSCTL）位段定义 416
表 10-73：温度传感器控制寄存器（TSENSCTL）位段描述 416
表 10-74：ADC 模块写使能寄存器（ADCREGKEY）位段定义 416
表 10-75：ADC 模块写使能寄存器（ADCREGKEY）位段描述 416
表 11-1：检测 ADC 检测的对象 418
表 11-2：计算温度所需的参数 419
表 12-1：SPGA 输入选择 420
表 12-2：SPGA 增益选择 421
表 12-3：PGA 单端模式不同增益的输入/输出电压范围 421
表 13-1：增益及差分输入/输出范围 422
表 13-2：PGA Module 基地址 425
表 13-3：PGA Register Map 425
表 13-4：DPGA 控制寄存器（DPGACTL）位段定义 426
表 13-5：DPGA 控制寄存器（DPGACTL）位段描述 426
表 13-6：SPGA 控制寄存器（SPGACTL）位段定义 427
表 13-7：SPGA 控制寄存器（SPGACTL）位段描述 427
表 13-8：PGA 模块写使能寄存器（PGAREGKEY）位段定义 428
表 13-9：PGA 模块写使能寄存器（PGAREGKEY）位段描述 428
表 14-1：比较器输入选择 430

SPINTROL

表 14-2：COMP 模块基地址 431
表 14-3：COMP 寄存器列表 431
表 14-4：比较器滤波输出寄存器（COMPFLTOUT）位段定义 432
表 14-5：比较器滤波输出寄存器（COMPFLTOUT）位段描述 432
表 14-6：比较器标志寄存器（COMPFLG）位段定义 432
表 14-7：比较器标志寄存器（COMPFLG）位段描述 433
表 14-8：比较器标志清除寄存器（COMPFLGCLR）位段定义 433
表 14-9：比较器标志清除寄存器（COMPFLGCLR）位段描述 433
表 14-10：比较器控制寄存器（COMPCTL）位段定义 434
表 14-11：比较器控制寄存器（COMPCTL）位段描述 434
表 14-12：COMPL 控制寄存器（COMPLCTL）位段定义 435
表 14-13：COMPL 控制寄存器（COMPLCTL）位段描述 435
表 14-14：COMPH 控制寄存器（COMPHCTL）位段定义 436
表 14-15：COMPH 控制寄存器（COMPHCTL）位段描述 436
表 14-16：DAC0 控制寄存器（DAC0CTL）位段定义 437
表 14-17：DAC0 控制寄存器（DAC0CTL）位段描述 438
表 14-18：DAC0 码值寄存器（DAC0CODE）位段定义 439
表 14-19：DAC0 码值寄存器（DAC0CODE）位段描述 439
表 14-20：DAC0 有效码值寄存器（DAC0CODEA）位段定义 439
表 14-21：DAC0 有效码值寄存器（DAC0CODEA）位段描述 439
表 14-22：RAMP0 延时影子寄存器（RAMP0DLY）位段定义 440

©2025 旋智电子科技（上海）有限公司

表 14-23：RAMP0 延时影子寄存器（RAMP0DLY）位段描述 440
表 14-24：RAMP0 有效延时寄存器（RAMP0DLYA）位段定义 440
表 14-25：RAMP0 有效延时寄存器（RAMP0DLYA）位段描述 440
表 14-26：RAMP0 递减步长影子寄存器（RAMP0DEC）位段定义 441
表 14-27：RAMP0 递减步长影子寄存器（RAMP0DEC）位段描述 441
表 14-28：RAMP0 有效递减步长寄存器（RAMP0DECA）位段定义 441
表 14-29：RAMP0 有效递减步长寄存器（RAMP0DECA）位段描述 441
表 14-30：RAMP0 计数数值重载影子寄存器（RAMP0MAX）位段定义 442
表 14-31：RAMP0 计数数值重载影子寄存器（RAMP0MAX）位段描述 442
表 14-32：RAMP0 有效计数数值重载寄存器（RAMP0MAXA）位段定义 442
表 14-33：RAMP0 有效计数数值重载寄存器（RAMP0MAXA）位段描述 442
表 14-34：RAMP0 计数寄存器（RAMP0CNT）位段定义 443
表 14-35：RAMP0 计数寄存器（RAMP0CNT）位段描述 443
表 14-36：DAC1 控制寄存器（DAC1CTL）位段定义 443
表 14-37：DAC1 控制寄存器（DAC1CTL）位段描述 443
表 14-38：DAC1 码值寄存器（DAC1CODE）位段定义 444
表 14-39：DAC1 码值寄存器（DAC1CODE）位段描述 445
表 14-40：DAC1 有效码值寄存器（DAC1CODEA）位段定义 445
表 14-41：DAC1 有效码值寄存器（DAC1CODEA）位段描述 445
表 14-42:： DAC2 控制寄存器（DAC2CTL）位段定义 445
表 14-43：DAC2 控制寄存器（DAC2CTL）位段描述 446

©2025 旋智电子科技（上海）有限公司

表 14-44：DAC2 码值寄存器（DAC2CODE）位段定义 446
表 14-45：DAC2 码值寄存器（DAC2CODE）位段描述 446
表 14-46：DAC2 有效码值寄存器（DAC2CODEA）位段定义 447
表 14-47：DAC2 有效码值寄存器（DAC2CODEA）位段描述 447
表 14-48：DAC 缓冲器控制寄存器（DACBUFCTL）位段定义 447
表 14-49：DAC 缓冲器控制寄存器（DACBUFCTL）位段描述 448
表 14-50：COMP 模块写使能寄存器（COMPREGKEY）位段定义 448
表 14-51：COMP 模块写使能寄存器（COMPREGKEY）位段描述 448
表 15-1：PHCOMP 模块基地址 453
表 15-2：PHCOMP 寄存器列表 453
表 15-3：PHCOMP 状态寄存器（PHCOMPSTS）位段定义 454
表 15-4：PHCOMP 状态寄存器（PHCOMPSTS）位段描述 454
表 15-5：PHCOMP 中断原始标志寄存器（PHCOMPRAWIF）位段定义 455
表 15-6：PHCOMP 中断原始标志寄存器（PHCOMPRAWIF）位段描述 455
表 15-7：PHCOMP 中断标志寄存器（PHCOMPIF）位段定义 456
表 15-8：PHCOMP 中断标志寄存器（PHCOMPIF）位段描述 456
表 15-9：PHCOMP 中断清除寄存器（PHCOMPIC）位段定义 457
表 15-10：PHCOMP 中断清除寄存器（PHCOMPIC）位段描述 457
表 15-11：PHCOMP 中断使能寄存器（PHCOMPIE）位段定义 458
表 15-12：PHCOMP 中断使能寄存器（PHCOMPIE）位段描述 458
表 15-13：PHCOMP 控制寄存器（PHCOMPCTL）位段定义 459

©2025 旋智电子科技（上海）有限公司

表 15-14：PHCOMP 控制寄存器（PHCOMPCTL）位段描述 459
表 15-15：PHCOMP 消影寄存器（PHCOMPBLANK）位段定义 461
表 15-16：PHCOMP 消影寄存器（PHCOMPBLANK）位段 461
表 15-17：PHCOMP 滤波寄存器（PHCOMPFILTER）位段定义 461
表 15-18：PHCOMP 滤波寄存器（PHCOMPFILTER）位段描述 462
表 15-19：PHCOMP 有限状态机复位寄存器（PHCOMPRST）位段定义 462
表 15-20：PHCOMP 有限状态机复位寄存器（PHCOMPRST）位段描述 462
表 15-21：PHCOMP 模块写使能寄存器（PHCOMPREGKEY）位段定义 463
表 15-22：PHCOMP 模块写使能寄存器（PHCOMPREGKEY）位段描述 463
表 16-1：DAC 缓冲器输入选择 465
表 19-1：UART 端口信号 473
表 19-2：UART 模块基地址 484
表 19-3：UART 寄存器列表 484
表 19-4：UART 数据寄存器（UARTDAT）位段定义 485
表 19-5：UART 数据寄存器（UARTDAT）位段描述 485
表 19-6：UART 控制寄存器（UARTCTL）位段定义 485
表 19-7：LIN ID 筛选标准寄存器（LINIDFILT）位段描述 491
表 19-8：UART/LIN 线路空闲阈值寄存器（BUSIDLETH）位段定义 491
表 19-9：UART/LIN 线路空闲阈值寄存器（BUSIDLETH）位段描述 491
表 19-10：UART 波特计数寄存器（UARTBDCNT）位段定义 491
表 19-11：UART 波特计数寄存器（UARTBDCNT）位段描述 492

©2025 旋智电子科技（上海）有限公司

表 19-12：UART 自动波特率计数寄存器（UARTABCNT）位段定义 492
表 19-13：UART 自动波特率计数寄存器（UARTABCNT）位段描述 492
表 19-14：UART 接收超时阈值寄存器（UARTTOTH）位段定义 492
表 19-15：UART 接收超时阈值寄存器（UARTTOTH）位段描述 493
表 19-16：UART 接收 FIFO 当前长度寄存器（UARTRXLVL）位段定义 493
表 19-17：UART 接收 FIFO 当前长度寄存器（UARTRXLVL）位段描述 493
表 19-18：UART 发送 FIFO 当前长度寄存器（UARTTXLVL）位段定义 493
表 19-19：UART 发送 FIFO 当前长度寄存器（UARTTXLVL）位段描述 494
表 19-20：UART 接收 FIFO 长度阈值寄存器（UARTRXTH）位段定义 494
表 19-21：UART 接收 FIFO 长度阈值寄存器（UARTRXTH）位段描述 494
表 19-22：UART 发送 FIFO 长度阈值寄存器（UARTTXTH）位段定义 494
表 19-23：UART 红外控制寄存器（UARTSIRCTL）位段定义 495
表 19-24：UART 红外控制寄存器（UARTSIRCTL）位段描述 495
表 19-25：UART 状态寄存器（UARTSTS）位段定义 496
表 19-26：UART 状态寄存器（UARTSTS）位段描述 496
表 19-27：UART 中断原始事件标志寄存器（UARTRAWIF）位段定义 497
表 19-28：UART 中断原始事件标志寄存器（UARTRAWIF）位段描述 497
表 19-29：UART 中断标志寄存器（UARTIF）位段定义 499
表 19-30：UART 中断标志寄存器（UARTIF）位段描述 499
表 19-31：UART 中断使能寄存器（UARTIE）位段描述 503
表 19-32：UART 接收信号消抖控制寄存器（UARTDGCTL）位段定义 506

©2025 旋智电子科技（上海）有限公司

表 19-33：UART 接收信号消抖控制寄存器（UARTDGCTL）位段描述 506
表 19-34：UART 模块写使能寄存器（UARTREGKEY）位段定义 506
表 19-35：UART 模块写使能寄存器（UARTREGKEY）位段描述 506
表 20-1：SPI 端口信号 509
表 20-2：SPI 模块基地址 520
表 20-3：SPI 寄存器列表 520
表 20-4：SPI 数据寄存器（SPIDATA）位段定义 521
表 20-5：SPI 数据寄存器（SPIDATA）位段描述 521
表 20-6：SPI 控制寄存器 0（SPICTL0）位段定义 521
表 20-7：SPI 控制寄存器 0（SPICTL0）位段描述 521
表 20-8：SPI 控制寄存器 1（SPICTL1）位段定义 523
表 20-9：SPI 控制寄存器 1（SPICTL1）位段描述 524
表 20-10：SPI 状态寄存器（SPISTS）位段定义 524
表 20-11：SPI 状态寄存器（SPISTS）位段描述 524
表 20-12：SPI 接收 FIFO 当前长度寄存器（SPIRFLVL）位段定义 526
表 20-13：SPI 接收 FIFO 当前长度寄存器（SPIRFLVL）位段描述 526
表 20-14：SPI 发送 FIFO 当前长度寄存器（SPITFLVL）位段定义 527
表 20-15：SPI 发送 FIFO 当前长度寄存器（SPITFLVL）位段描述 527
表 20-16：SPI 接收 FIFO 阈值寄存器（SPIRFTH）位段定义 527
表 20-17：SPI 接收 FIFO 阈值寄存器（SPIRFTH）位段描述 527
表 20-18：SPI 发送 FIFO 阈值寄存器（SPITFTH）位段定义 528

©2025 旋智电子科技（上海）有限公司

表 20-19：SPI 发送 FIFO 阈值寄存器（SPITFTH）位段描述 528
表 20-20：SPI 中断原始标志寄存器（SPIRAWIF）位段定义 528
表 20-21：SPI 中断原始标志寄存器（SPIRAWIF）位段描述 528
表 20-22：SPI 中断标志寄存器（SPIIF）位段定义 529
表 20-23：SPI 中断标志寄存器（SPIIF）位段描述 529
表 20-24：SPI 中断清除寄存器（SPIIC）位段定义 530
表 20-25：SPI 中断清除寄存器（SPIIC）位段描述 531
表 20-26：SPI 中断使能寄存器（SPIIE）位段定义 531
表 20-27：SPI 中断使能寄存器（SPIIE）位段描述 532
表 20-28：SPI 超时寄存器（SPITO）位段定义 532
表 20-29：SPI 超时寄存器（SPITO）位段描述 533
表 20-30：SPI 可编程串行协议寄存器（SPIPSP）位段定义 533
表 20-31：SPI 可编程串行协议寄存器（SPIPSP）位段描述 533
表 20-32：SPI 测试控制寄存器（SPITCTL）位段定义 534
表 20-33：SPI 测试控制寄存器（SPITCTL）位段描述 534
表 20-34：SPI 波特率计数寄存器（SPIBDCNT）位段定义 535
表 20-35：SPI 波特率计数寄存器（SPIBDCNT）位段描述 535
表 20-36：SPI 模块写使能寄存器（SPIREGKEY）位段定义 535
表 20-37：SPI 模块写使能寄存器（SPIREGKEY）位段描述 536
表 21-1：I2C 端口信号描述 537
表 21-2：I2C 模块基地址 542

©2025 旋智电子科技（上海）有限公司

表 21-3：I2C 寄存器列表 542
表 21-4：I2C 控制寄存器（I2CCTL）位段定义 543
表 21-5：I2C 控制寄存器（I2CCTL）位段描述 543
表 21-6：I2C 主机寻址寄存器（I2CMADDR）位段定义 544
表 21-7：I2C 主机寻址寄存器（I2CMADDR）位段描述 545
表 21-8：I2C 从机地址寄存器（I2CSADDR）位段定义 545
表 21-9：I2C 从机地址寄存器（I2CSADDR）位段描述 546
表 21-10：I2C 数据和命令寄存器（I2CDATACMD）位段定义 546
表 21-11：I2C Data and Command Register (I2CDATACMD) 描述 546
表 21-12：I2C 标准速度模式时钟低电平计数寄存器（I2CSSLCNT）位段定义 547
表 21-13：I2C 标准速度模式时钟低电平计数寄存器（I2CSSLCNT）位段描述 547
表 21-14：I2C 标准速度模式时钟高电平计数寄存器（I2CSSHCNT）位段定义 547
表 21-15：I2C 标准速度模式时钟高电平计数寄存器（I2CSSHCNT）位段 548
表 21-16：I2C 快速度模式时钟低电平计数寄存器（I2CFSLCNT）位段定义 548
表 21-17：I2C 快速度模式时钟低电平计数寄存器（I2CFSLCNT）位段描述 548
表 21-18：I2C 快速度模式时钟高电平计数寄存器（I2CFSHCNT）位段定义 549
表 21-19：I2C 快速度模式时钟高电平计数寄存器（I2CFSHCNT）位段描述 549
表 21-20：I2C 高速度模式时钟低电平计数寄存器（I2CHSLCNT）位段定义 549
表 21-21：I2C 高速度模式时钟低电平计数寄存器（I2CHSLCNT）位段描述 550
表 21-22：I2C 高速度模式时钟高电平计数寄存器（I2CHSHCNT）位段定义 550
表 21-23：I2C 高速度模式时钟高电平计数寄存器（I2CHSHCNT）位段描述 550

©2025 旋智电子科技（上海）有限公司

表 21-24：I2C 快速模式尖峰抑制阈值寄存器（I2CFSSPKLEN）位段定义 551
表 21-25：I2C 快速模式尖峰抑制阈值寄存器（I2CFSSPKLEN）位段描述 551
表 21-26：I2C 高速模式尖峰抑制阈值寄存器（I2CHSSPKLEN）位段定义 551
表 21-27：I2C 高速模式尖峰抑制阈值寄存器（I2CHSSPKLEN）位段描述 552
表 21-28：I2C 数据建立时间寄存器（I2CSDASETUP）位段定义 552
表 21-29：I2C 数据建立时间寄存器（I2CSDASETUP）位段描述 552
表 21-30：I2C 数据保持时间寄存器（I2CSDAHOLD）位段定义 553
表 21-31：I2C 数据保持时间寄存器（I2CSDAHOLD）位段描述 553
表 21-32：I2C 发送 FIFO 当前长度寄存器（I2CTXLVL）位段定义 553
表 21-33：I2C 发送 FIFO 当前长度寄存器（I2CTXLVL）位段描述 554
表 21-34：I2C 接收 FIFO 当前长度寄存器（I2CRXLVL）位段定义 554
表 21-35：I2C 接收 FIFO 当前长度寄存器（I2CRXLVL）位段描述 554
表 21-36：I2C 发送 FIFO 长度阈值寄存器（I2CTXTH）位段定义 554
表 21-37：I2C 发送 FIFO 长度阈值寄存器（I2CTXTH）位段描述 555
表 21-38：I2C 接收 FIFO 长度阈值寄存器（I2CRXTH）位段定义 555
表 21-39：I2C 接收 FIFO 长度阈值寄存器（I2CRXTH）位段描述 555
表 21-40：I2C 状态寄存器（I2CSTS）位段定义 555
表 21-41：I2C 状态寄存器（I2CSTS）位段描述 556
表 21-42：I2C 发送中止源寄存器（I2CTXABRTSRC）位段定义 557
表 21-43：I2C 发送中止源寄存器（I2CTXABRTSRC）位段描述 557
表 21-44：I2C 中断原始事件标志寄存器（I2CRAWIF）位段定义 560

©2025 旋智电子科技（上海）有限公司

表 21-45：I2C 中断原始事件标志寄存器（I2CRAWIF）位段描述 560
表 21-46：I2C 中断标志寄存器（I2CIF）位段定义 561
表 21-47：I2C 中断标志寄存器（I2CIF）位段描述 562
表 21-48：I2C 中断使能寄存器（I2CIE）位段定义 563
表 21-49：I2C 中断使能寄存器（I2CIE）位段描述 563
表 21-50：I2C 中断清除寄存器（I2CIC）位段定义 564
表 21-51：I2C 中断清除寄存器（I2CIC）位段描述 565
表 21-52：I2C 模块写使能寄存器（I2CREGKEY）位段定义 566
表 21-53：I2C 模块写使能寄存器（I2CREGKEY）位段描述 566
表 22-1：CAN 端口信号 570
表 22-2：信箱控制位 577
表 22-3：用于普通报文发送的信箱控制位设置 578
表 22-4：用于自动应答远程帧时的报文发送信箱控制位设置 578
表 22-5：用于普通报文接收的信箱控制位设置 579
表 22-6：用于接收其他节点应答远程帧的报文的信箱控制位设置 580
表 22-7：CAN 模块基地址 581
表 22-8：CAN 寄存器列表 581
表 22-9：CAN 控制寄存器（CANCTL）位段定义 582
表 22-10：CAN 控制寄存器（CANCTL）位段描述 582
表 22-11：CAN 标称位时间量子寄存器（CANNBTQ）位段定义 584
表 22-12：CAN 标称位时间量子寄存器（CANNBTQ）位段描述 584

SPINTROL

表 22-13：CAN 数据位时间量子寄存器（CANDBTQ）位段定义 585
表 22-14：CAN 数据位时间量子寄存器（CANDBTQ）位段描述 585
表 22-15：CAN 发送器延时补偿寄存器（CANTDC）位段定义 586
表 22-16：CAN 发送器延时补偿寄存器（CANTDC）位段描述 586
表 22-17：CAN 总线关闭恢复延时寄存器（CANBRDLY）位段定义 586
表 22-18：CAN 总线关闭恢复延时寄存器（CANBRDLY）位段描述 586
表 22-19：CAN 状态（CANSTS）位段定义 587
表 22-20：CAN 状态（CANSTS）位段描述 587
表 22-21：CAN 最新错误码寄存器（CANLEC）位段定义 588
表 22-22：CAN 最新错误码寄存器（CANLEC）位段描述 588
表 22-23：CAN 杂项中断使能寄存器（CANMIE）位段定义 589
表 22-24：CAN 杂项中断使能寄存器（CANMIE）位段描述 589
表 22-25：CAN 杂项中断原始事件标志寄存器（CANMRAWIF）位段定义 590
表 22-26：CAN 杂项中断原始事件标志寄存器（CANMRAWIF）位段描述 590
表 22-27：CAN 杂项中断标志寄存器（CANMIF）位段定义 591
表 22-28：CAN 杂项中断标志寄存器（CANMIF）位段描述 592
表 22-29：CAN 杂项中断清除寄存器（CANMIC）位段定义 593
表 22-30：CAN 杂项中断清除寄存器（CANMIC）位段描述 593
表 22-31：CAN 信箱组传输完成中断标志寄存器（CANTGIF）位段定义 594
表 22-32：CAN 信箱组传输完成中断标志寄存器（CANTGIF）位段描述 595
表 22-33：CAN 信箱组传输完成中断清除寄存器（CANTGIC）位段定义 596

©2025 旋智电子科技（上海）有限公司

表 22-34：CAN 信箱组传输完成中断清除寄存器（CANTGIC）位段描述 596
表 22-35：CAN 内存错误地址寄存器（CANMEMERRADDR）位段定义 597
表 22-36：CAN 内存错误地址寄存器（CANMEMERRADDR）位段描述 597
表 22-37：CAN 测试控制寄存器（CANTEST）位段定义 597
表 22-38：CAN 测试控制寄存器（CANTEST）位段描述 598
表 22-39：CAN 错误计数控制寄存器（CANECCTL）位段定义 598
表 22-40：CAN 错误计数控制寄存器（CANECCTL）位段描述 599
表 22-41：CAN 发送请求寄存器 n ([0:1])（CANTXREQ）位段定义 599
表 22-42：CAN 发送请求寄存器 n ([0:1])（CANTXREQ）位段描述 599
表 22-43：CAN 新报文标志寄存器 n ([0:1])（CANMSGNEW）位段定义 599
表 22-44：CAN 新报文标志寄存器 n ([0:1])（CANMSGNEW）位段描述 600
表 22-45：CAN 信箱传输完成中断标志寄存器 n ([0:1])（CANMSGIF）位段定义 600
表 22-46：CAN 信箱传输完成中断标志寄存器 n ([0:1])（CANMSGIF）位段描述 600
表 22-47：CAN 信箱使能寄存器 n ([0:1])（CANMBEN）位段定义 600
表 22-48：CAN 信箱使能寄存器 n ([0:1])（CANMBEN）位段描述 601
表 22-49：CAN 时间戳控制寄存器（CANTSCTL）位段定义 601
表 22-50：CAN 时间戳控制寄存器（CANTSCTL）位段描述 601
表 22-51：CAN 时间戳计数寄存器（CANTSCNT）位段定义 601
表 22-52：CAN 时间戳计数寄存器（CANTSCNT）位段描述 602
表 22-53：CAN 模块写使能寄存器（CANREGKEY）位段定义 602
表 22-54：CAN 模块写使能寄存器（CANREGKEY）位段描述 602

©2025 旋智电子科技（上海）有限公司

表 22-55：CANMBOX 模块基地址 603
表 22-56：CANMBOX 寄存器列表 603
表 22-57：CAN 信箱报文数据字 n ([0:15])寄存器（CANMBOXFDW）位段定义 604
表 22-58：CAN 信箱报文数据字 n ([0:15])寄存器（CANMBOXFDW）位段描述 604
表 22-59：CAN 信箱报文标识符掩码寄存器（CANMBOXFMSK）位段定义 604
表 22-60：CAN 信箱报文标识符掩码寄存器（CANMBOXFMSK）位段描述 605
表 22-61：CAN 信箱报文标识符寄存器（CANMBOXFID）位段定义 605
表 22-62：CAN 信箱报文标识符寄存器（CANMBOXFID）位段描述 605
表 22-63：CAN 信箱报文时间戳寄存器（CANMBOXMTS）位段定义 606
表 22-64：CAN 信箱报文时间戳寄存器（CANMBOXMTS）位段描述 606
表 22-65：CAN 信箱报文控制寄存器（CANMBOXMCTL）位段定义 606
表 22-66：CAN 信箱报文控制寄存器（CANMBOXMCTL）位段描述 606
表 23-1：CRC 模块基地址 611
表 23-2：CRC 寄存器列表 611
表 23-3：CRC 控制寄存器（CRCCTL）位段定义 612
表 23-4：CRC 控制寄存器（CRCCTL）位段描述 612
表 23-5：CRC 数据流长度寄存器（CRCSTRLEN）位段定义 613
表 23-6：CRC 数据流长度寄存器（CRCSTRLEN）位段描述 613
表 23-7：CRC 初值种子寄存器（CRCSEED）位段定义 614
表 23-8：CRC 初值种子寄存器（CRCSEED）位段描述 614
表 23-9：CRC 数据流输入寄存器（CRCSTRIN）位段定义 614

©2025 旋智电子科技（上海）有限公司

表 23-10：CRC 数据流输入寄存器（CRCSTRIN）位段描述 614
表 23-11：CRC 结果寄存器（CRCRESULT）位段定义 615
表 23-12：CRC 结果寄存器（CRCRESULT）位段描述 615
表 23-13：CRC 中断原始事件标志寄存器（CRCRAWIF）位段定义 615
表 23-14：CRC 中断原始事件标志寄存器（CRCRAWIF）位段描述 615
表 23-15：CRC 中断标志寄存器（CRCIF）位段定义 616
表 23-16：CRC 中断标志寄存器（CRCIF）位段描述 616
表 23-17：CRC 中断清除寄存器（CRCIC）位段定义 616
表 23-18：CRC 中断清除寄存器（CRCIC）位段描述 617
表 23-19：CRC 中断使能寄存器（CRCIE）位段定义 617
表 23-20：CRC 中断使能寄存器（CRCIE）位段描述 617
表 23-21：CRC 模块写使能寄存器（CRCREGKEY）位段定义 618
表 23-22：CRC 模块写使能寄存器（CRCREGKEY）位段描述 618
表 24-1：Flash 模块构成 620
表 24-2：Flash 存储器读访问操作延迟 621
表 24-3：配置字的构成及其描述 627
表 24-4：Flash 模块基地址 628
表 24-5：FLASHC 寄存器列表 628
表 24-6：Flash 控制寄存器（FLASHCTL）位段定义 629
表 24-7：Flash 控制寄存器（FLASHCTL）位段描述 629
表 24-8：Flash 地址寄存器（FLASHADDR）位段定义 630

©2025 旋智电子科技（上海）有限公司

表 24-9：Flash 地址寄存器（FLASHADDR）位段描述 630
表 24-10：Flash 输入数据低 32 位寄存器（FLASHDINL）位段定义 630
表 24-11：Flash 输入数据低 32 位寄存器（FLASHDINL）位段描述 630
表 24-12：Flash 输入数据高 32 位寄存器（FLASHDINH）位段定义 631
表 24-13：Flash 输入数据高 32 位寄存器（FLASHDINH）位段描述 631
表 24-14：Flash 输入数据 ECC 字节寄存器（FLASHDINE）位段定义 631
表 24-15：Flash 输入数据 ECC 字节寄存器（FLASHDINE）位段描述 631
表 24-16：Flash 数据输入模式寄存器（FLASHDINP）位段定义 632
表 24-17：Flash 数据输入模式寄存器（FLASHDINP）位段描述 632
表 24-18：Flash 输出数据低 32 位寄存器（FLASHDOUTL）位段定义 632
表 24-19：Flash 输出数据低 32 位寄存器（FLASHDOUTL）位段描述 632
表 24-20：Flash 输出数据高 32 位寄存器（FLASHDOUTH）位段定义 633
表 24-21：Flash 输出数据高 32 位寄存器（FLASHDOUTH）位段描述 633
表 24-22：Flash 输出数据 ECC 字节寄存器（FLASHDOUTE）位段定义 633
表 24-23：Flash 输出数据 ECC 字节寄存器（FLASHDOUTE）位段描述 633
表 24-24：Flash 数据输出模式寄存器（FLASHDOUTP）位段定义 634
表 24-25：Flash 数据输出模式寄存器（FLASHDOUTP）位段描述 634
表 24-26：Flash 写保护寄存器（FLASHWP）位段定义 634
表 24-27：Flash 写保护寄存器（FLASHWP）位段描述 634
表 24-28：Flash 控制器写使能寄存器（FLASHCREGKEY）位段定义 637
表 24-29：Flash 控制器写使能寄存器（FLASHCREGKEY）位段描述 637

©2025 旋智电子科技（上海）有限公司

表 25-1：SYSTEM 模块及地址 640
表 25-2：SYSTEM 寄存器列表 640
表 25-3：产品识别号寄存器 0（CID0）位段定义 642
表 25-4：产品识别号寄存器 0（CID0）位段描述 642
表 25-5：产品识别号寄存器 1（CID1）位段描述 642
表 25-6：产品识别号寄存器 1（CID1）位段描述 642
表 25-7：唯一设备识别号寄存器 0（UID0）位段定义 643
表 25-8：唯一设备识别号寄存器 0（UID0）位段描述 643
表 25-9：唯一设备识别号寄存器 1（UID1）位段定义 643
表 25-10：唯一设备识别号寄存器 1（UID1）位段描述 643
表 25-11：随机数寄存器 0（RND0）位段定义 644
表 25-12：随机数寄存器 0（RND0）位段描述 644
表 25-13：随机数寄存器 1（RND1）位段定义 644
表 25-14：随机数寄存器 1（RND1）位段描述 644
表 25-15：版本信息寄存器 0（REV0）位段定义 645
表 25-16：版本信息寄存器 0（REV0）位段描述 645
表 25-17：版本信息寄存器 1（REV1）位段定义 645
表 25-18：版本信息寄存器 1（REV1）位段描述 645
表 25-19：系统监控寄存器（SYSMONVTL）位段定义 646
表 25-20：系统监控寄存器（SYSMONVTL）位段描述 646
表 25-21：VBAT 监测阈值寄存器（VBATTH）位段定义 647

©2025 旋智电子科技（上海）有限公司

表 25-22：VBAT 监测阈值寄存器（VBATTH）位段描述 647
表 25-23：VCP 监测阈值寄存器（VCPTH）位段定义 648
表 25-24：VCP 监测阈值寄存器（VCPTH）位段描述 648
表 25-25：VMON 监测阈值寄存器（VMONTH）位段定义 648
表 25-26：VMON 监测阈值寄存器（VMONTH）位段描述 648
表 25-27：VDD5 监测阈值寄存器（VDD5TH）位段定义 649
表 25-28：VDD5 监测阈值寄存器（VDD5TH）位段描述 649
表 25-29：POWER Bandgap 监测阈值寄存器（VBGTH）位段定义 649
表 25-30：POWER Bandgap 监测阈值寄存器（VBGTH）位段描述 649
表 25-31：POWER 模块温度监测阈值寄存器（TPMUTH）位段定义 650
表 25-32：POWER 模块温度监测阈值寄存器（TPMUTH）位段描述 650
表 25-33：LIN PHY 温度监测阈值寄存器（TLINTH）位段定义 650
表 25-34：LIN PHY 温度监测阈值寄存器（TLINTH）位段描述 650
表 25-35：MCU 温度监测阈值寄存器（TMCUTH）位段定义 651
表 25-36：MCU 温度监测阈值寄存器（TMCUTH）位段描述 651
表 25-37：VDD33 监测阈值寄存器（VDD33TH）位段定义 651
表 25-38：VDD33 监测阈值寄存器（VDD33TH）位段描述 651
表 25-39：VDD12 监测阈值寄存器（VDD12TH）位段定义 652
表 25-40：VDD12 监测阈值寄存器（VDD12TH）位段描述 652
表 25-41：ADC 参考电压监测阈值寄存器（VADCREFTH）位段定义 652
表 25-42：ADC 参考电压监测阈值寄存器（VADCREFTH）位段描述 652

©2025 旋智电子科技（上海）有限公司

表 25-43：VBAT 监测码值寄存器（VBATCODE）位段定义 653
表 25-44：VBAT 监测码值寄存器（VBATCODE）位段描述 653
表 25-45：VCP 监测码值寄存器（VCPCODE）位段定义 653
表 25-46：VCP 监测码值寄存器（VCPCODE）位段描述 653
表 25-47：VMON 监测码值寄存器（VCPCODE）位段定义 654
表 25-48：VMON 监测码值寄存器（VCPCODE）位段描述 654
表 25-49：VDD5 监测码值寄存器（VDD5CODE）位段定义 654
表 25-50：VDD5 监测码值寄存器（VDD5CODE）位段描述 654
表 25-51：POWER Bandgap 监测码值寄存器（VBGCODE）位段定义 655
表 25-52：POWER Bandgap 监测码值寄存器（VBGCODE）位段描述 655
表 25-53：POWER 模块温度监测码值寄存器（TPMUCODE）位段定义 655
表 25-54：POWER 模块温度监测码值寄存器（TPMUCODE）位段描述 655
表 25-55：LIN PHY 温度监测码值寄存器（TLINCODE）位段定义 656
表 25-56：LIN PHY 温度监测码值寄存器（TLINCODE）位段描述 656
表 25-57：MCU 温度监测码值寄存器（TMCUCODE）位段定义 656
表 25-58：MCU 温度监测码值寄存器（TMCUCODE）位段描述 656
表 25-59：VDD33 监测码值寄存器（VDD33CODE）位段定义 657
表 25-60：VDD33 监测码值寄存器（VDD33CODE）位段描述 657
表 25-61：VDD12 监测码值寄存器（VDD12CODE）位段定义 657
表 25-62：VDD12 监测码值寄存器（VDD12CODE）位段描述 657
表 25-63：13 位 ADC 参考电压监测码值寄存器（VADCREFCODE）位段定义 658

©2025 旋智电子科技（上海）有限公司

表 25-64：13 位 ADC 参考电压监测码值寄存器（VADCREFCODE）位段描述 658
表 25-65：GPIO 唤醒控制寄存器（GPIOWKUPCTL）位段定义 658
表 25-66：GPIO 唤醒控制寄存器（GPIOWKUPCTL）位段描述 659
表 25-67：GPIO NMI 中断控制寄存器（GPIONMICTL）位段定义 659
表 25-68：GPIO NMI 中断控制寄存器（GPIONMICTL）位段描述 660
表 25-69：NMI 中断原始标志寄存器（NMRAWIF）位段定义 660
表 25-70：NMI 中断原始标志寄存器（NMRAWIF）位段描述 660
表 25-71：不可屏蔽中断标志寄存器（NMIF）位段定义 662
表 25-72：不可屏蔽中断标志寄存器（NMIF）位段描述 663
表 25-73：不可屏蔽中断清除寄存器（NMIC）位段定义 666
表 25-74 不可屏蔽中断清除寄存器（NMIC）位段描述 666
表 25-75：不可屏蔽中断使能寄存器（NMIE）位段定义 669
表 25-76：不可屏蔽中断使能寄存器（NMIE）位段描述 670
表 25-77：内存 ECC 错误原始标志寄存器（MEMRAWIF）位段定义 672
表 25-78：内存 ECC 错误原始标志寄存器（MEMRAWIF）位段描述 672
表 25-79：内存 ECC 错误标志寄存器（MEMIF）位段定义 673
表 25-80：内存 ECC 错误标志寄存器（MEMIF）位段描述 673
表 25-81：内存 ECC 错误清除寄存器（MEMIC）位段定义 674
表 25-82：内存 ECC 错误清除寄存器（MEMIC）位段描述 674
表 25-83：内存 ECC 错误使能寄存器（MEMIE）位段定义 675
表 25-84：内存 ECC 错误使能寄存器（MEMIE）位段描述 675

©2025 旋智电子科技（上海）有限公司

表 25-85：复位事件标志寄存器（RSTEVTFLG）位段定义 675
表 25-86：复位事件标志寄存器（RSTEVTFLG）位段描述 676
表 25-87：复位事件清除寄存器（RSTEVTCLR）位段定义 677
表 25-88：复位事件清除寄存器（RSTEVTCLR）位段描述 677
表 25-89：复位事件使能寄存器（RSTEVTEN）位段定义 679
表 25-90：复位事件使能寄存器（RSTEVTEN）位段描述 679
表 25-91：系统信息寄存器（SYSINFO）位段定义 680
表 25-92：系统信息寄存器（SYSINFO）位段描述 680
表 25-93：ROM 访问 ECC 错误地址寄存器（ROMERRADDR）位段定义 681
表 25-94：ROM 访问 ECC 错误地址寄存器（ROMERRADDR）位段描述 681
表 25-95：Flash 访问 ECC 错误地址寄存器（FLASHERRADDR）位段定义 682
表 25-96：Flash 访问 ECC 错误地址寄存器（FLASHERRADDR）位段描述 682
表 25-97：RAM0 访问 ECC 错误地址寄存器（RAM0ERRADDR）位段定义 682
表 25-98：RAM0 访问 ECC 错误地址寄存器（RAM0ERRADDR）位段描述 682
表 25-99：RAM1 访问 ECC 错误地址寄存器（RAM1ERRADDR）位段定义 683
表 25-100：RAM1 访问 ECC 错误地址寄存器（RAM1ERRADDR）位段描述 683
表 25-101：调试接口控制寄存器（DBGIFCTL）位段定义 683
表 25-102：调试接口控制寄存器（DBGIFCTL）位段描述 683
表 25-103：引导信息寄存器（BOOTINFO）位段定义 684
表 25-104：引导信息寄存器（BOOTINFO）位段描述 684
表 25-105：AHB 多比特 ECC 错误事件寄存器（AHBMBEFLG）位段定义 684

©2025 旋智电子科技（上海）有限公司

表 25-106：AHB 多比特 ECC 错误事件寄存器（AHBMBEFLG）位段描述 685
表 25-107：AHB 多比特 ECC 错误事件清除寄存器（AHBMBECLR）位段定义 687
表 25-108：AHB 多比特 ECC 错误事件清除寄存器（AHBMBECLR）位段定义 687
表 25-109：AHB 单比特 ECC 错误事件标志寄存器（AHBSBEFLG）位段定义 690
表 25-110：AHB 单比特 ECC 错误事件标志寄存器（AHBSBEFLG）位段描述 690
表 25-111：AHB 单比特 ECC 错误事件清除寄存器（AHBSBECLR）位段定义 693
表 25-112：AHB 单比特 ECC 错误事件清除寄存器（AHBSBECLR）位段描述 693
表 25-113：外设复位事件标志寄存器（PERIRSTFLG）位段定义 696
表 25-114：外设复位事件标志寄存器（PERIRSTFLG）位段描述 696
表 25-115：外设复位事件清除寄存器（PERIRSTCLR）位段定义 698
表 25-116：Peripheral Reset Event Clear Register (PERIRSTCLR) 描述 698
表 25-117：上电永久寄存器 0（NVREG0）位段定义 699
表 25-118：上电永久寄存器 0（NVREG0）位段描述 700
表 25-119：上电永久寄存器 1（NVREG1）位段定义 700
表 25-120：上电永久寄存器 1（NVREG1）位段描述 700
表 25-121：上电永久寄存器 2（NVREG2）位段定义 700
表 25-122：上电永久寄存器 2（NVREG2）位段描述 701
表 25-123：SYSTEM 模块写使能寄存器（SYSREGKEY）位段定义 702
表 25-124：SYSTEM 模块写使能寄存器（SYSREGKEY）位段描述 702
表 26-1：DMAC 硬件握手接口分配 705
表 26-2：DMACH 模块基地址 713

©2025 旋智电子科技（上海）有限公司

表 26-3：DMACH 寄存器列表 713
表 26-4：DMA 通道源地址寄存器（DMACHSA）位段定义 714
表 26-5：DMA 通道源地址寄存器（DMACHSA）位段描述 714
表 26-6：DMA 通道源地址寄存器（DMACHDA）位段定义 714
表 26-7：DMA 通道源地址寄存器（DMACHDA）位段描述 714
表 26-8：DMA 通道控制寄存器 0（DMACHCTL0）位段定义 715
表 26-9：DMA 通道控制寄存器 0（DMACHCTL0）位段描述 715
表 26-10：DMA 通道控制寄存器 1（DMACHCTL1）位段定义 717
表 26-11：DMA 通道控制寄存器 1（DMACHCTL1）位段描述 717
表 26-12：DMA 通道配置寄存器 0（DMACHCFG0）位段定义 717
表 26-13：DMA 通道配置寄存器 0（DMACHCFG0）位段描述 718
表 26-14：DMA 通道配置寄存器 1（DMACHCFG1）位段定义 719
表 26-15：DMA 通道配置寄存器 1（DMACHCFG1）位段描述 719
表 26-16：DMAC 模块基地址 720
表 26-17：DMAC 寄存器列表 720
表 26-18：DMA 传输完成中断原始事件标志寄存器（DMATCRAWIF）位段定义 721
表 26-19：DMA 传输完成中断原始事件标志寄存器（DMATCRAWIF）位段描述 721
表 26-20：DMA 块传输完成中断原始事件标志寄存器（DMABTCRAWIF）位段定义 722
表 26-21：DMA 块传输完成中断原始事件标志寄存器（DMABTCRAWIF）位段描述 722
表 26-22：DMA 源端传输完成中断原始事件标志寄存器（DMASTCRAWIF）位段定义 723
表 26-23：DMA 源端传输完成中断原始事件标志寄存器（DMASTCRAWIF）位段描述 723

©2025 旋智电子科技（上海）有限公司

表 26-24：DMA 目的端传输完成中断原始事件标志寄存器（DMADTCRAWIF）位段定义 724
表 26-25：DMA 目的端传输完成中断原始事件标志寄存器（DMADTCRAWIF）位段描述 724
表 26-26：DMA 总线访问错误原始标志寄存器（DMAERRRAWIF）位段定义 725
表 26-27：DMA 总线访问错误原始标志寄存器（DMAERRRAWIF）位段描述 725
表 26-28：DMA 传输完成中断标志寄存器（DMATCIF）位段定义 726
表 26-29：DMA 传输完成中断标志寄存器（DMATCIF）位段描述 726
表 26-30：DMA 块传输完成中断标志寄存器（DMABTCIF）位段定义 727
表 26-31：DMA 块传输完成中断标志寄存器（DMABTCIF）位段描述 727
表 26-32：DMA 源端传输完成中断标志寄存器（DMASTCIF）位段定义 728
表 26-33：DMA 源端传输完成中断标志寄存器（DMASTCIF）位段描述 729
表 26-34：DMA 目的端传输完成中断标志寄存器（DMADTCIF）位段定义 730
表 26-35：DMA 目的端传输完成中断标志寄存器（DMADTCIF）位段描述 730
表 26-36：DMA 总线访问错误中断标志寄存器（DMAERRIF）位段定义 731
表 26-37：DMA 总线访问错误中断标志寄存器（DMAERRIF）位段描述 731
表 26-38：DMA 传输完成中断使能寄存器（DMATCIE）位段定义 732
表 26-39：DMA 传输完成中断使能寄存器（DMATCIE）位段描述 732
表 26-40：DMA 块传输完成中断使能寄存器（DMABTCIE）位段定义 734
表 26-41：DMA 块传输完成中断使能寄存器（DMABTCIE）位段描述 734
表 26-42：DMA 源端传输完成中断使能寄存器（DMASTCIE）位段定义 735
表 26-43：DMA 源端传输完成中断使能寄存器（DMASTCIE）位段描述 736
表 26-44：DMA 目的端传输完成中断使能寄存器（DMADTCIE）位段定义 737

©2025 旋智电子科技（上海）有限公司

表 26-45：DMA 目的端传输完成中断使能寄存器（DMADTCIE）位段描述 737
表 26-46：DMA 总线访问错误中断使能寄存器（DMAERRIE）位段定义 739
表 26-47：DMA 总线访问错误中断使能寄存器（DMAERRIE）位段描述 739
表 26-48：DMA 传输完成中断清除寄存器（DMATCIC）位段定义 741
表 26-49：DMA 传输完成中断清除寄存器（DMATCIC）位段描述 741
表 26-50：DMA 块传输完成中断清除寄存器（DMABTCIC）位段定义 742
表 26-51：DMA 块传输完成中断清除寄存器（DMABTCIC）位段描述 742
表 26-52：DMA 源端传输完成中断清除寄存器（DMASTCIC）位段定义 743
表 26-53：DMA 源端传输完成中断清除寄存器（DMASTCIC）位段描述 743
表 26-54：DMA 目的端传输完成中断清除寄存器（DMADTCIC）位段定义 744
表 26-55：DMA 目的端传输完成中断清除寄存器（DMADTCIC）位段描述 745
表 26-56：DMA 总线访问错误中断清除寄存器（DMAERRIC）位段定义 746
表 26-57：DMA 总线访问错误中断清除寄存器（DMAERRIC）位段描述 746
表 26-58：DMA 组合中断标志寄存器（DMAIF）位段定义 747
表 26-59：DMA 组合中断标志寄存器（DMAIF）位段描述 747
表 26-60：DMA 源端软件事务请求寄存器（DMASSREQ）位段定义 748
表 26-61：DMA 源端软件事务请求寄存器（DMASSREQ）位段描述 748
表 26-62：DMA 目的端软件事务请求寄存器（DMADSREQ）位段定义 750
表 26-63：DMA 目的端软件事务请求寄存器（DMADSREQ）位段描述 750
表 26-64：DMA 源端软件事务单次请求寄存器（DMASSSREQ）位段定义 751
表 26-65：DMA 源端软件事务单次请求寄存器（DMASSSREQ）位段描述 752

©2025 旋智电子科技（上海）有限公司

表 26-66：DMA 目的端软件事务单次请求寄存器（DMADSSREQ）位段定义 753

表 26-67：DMA 目的端软件事务单次请求寄存器（DMADSSREQ）位段描述 753

表 26-68：DMA 源端软件事务末次请求寄存器（DMASSLREQ）位段定义 755

表 26-69：DMA 源端软件事务末次请求寄存器（DMASSLREQ）位段描述 755

表 26-70：DMA 目的端软件事务末次请求寄存器（DMADSLREQ）位段定义 758

表 26-71：DMA 目的端软件事务末次请求寄存器（DMADSLREQ）位段描述 758

表 26-72：DMA 使能寄存器（DMAEN）位段定义 759

表 26-73：DMA 使能寄存器（DMAEN）位段描述 760

表 26-74：DMA 通道使能寄存器（DMACHEN）位段定义 760

表 26-75：DMA 通道使能寄存器（DMACHEN）位段描述 760

SPINTROL

©2025 旋智电子科技（上海）有限公司

历史版本

版本	日期	作者	状态	变更
A/0	2023-07-10	C. Chai	已过期	首次发布。
A/1	2023-07-14	C. Chai	已过期	1. 修改章节 10.8.1。 2. 修改表 24-3。 3. 修改章节 15.3。 4. 修改章节 20.2。 修改图 20-4，图 20-5，图 20-6，图 20-7，图 20-8，图 20-9， 图 20-10， 5. 图 20-11 的编号。 6. 重新添加题注：图 18-1。 7. 修改章节 18.3 表格引用。 8. 修改章节 19.1 图引用。 9. 修改章节 19.4 中图 19-4，图 19-5，图 19-6，图 19-7，图 19-8，的引用。 10. 修改章节 19.5 中图 19-9，图 19-10，图 19-11，图 19-12，图 19-13 的引用。 11. 修改章节 20.1 中图 20-1 的引用。 12. 修改章节 20.5.1 中图 20-4 的编号。 13. 修改章节 20.5.2 中图 20-6 的编号。 修改章节 20.7.2 中 图 20-10， 14. 图 20-11 的编号。 15. 修改章节 22.5.1 中图 22-9 的引用。 16. 修改表 19-12 的描述。 17. 修改表 14-42~表 14-51 的引用。 修改表 20-1~ 18. 表 20-37 的编号。
A/2	2023-08-20	C. Chai	已过期	1. 更新图表引用。 2. 修改章节 1.3.2。 3. 修改章节 19.5.2。 4. 修改表 20-29。 5. 修改表 4-1。 6. 修改表 4-10、表 4-16、表 4-74、表 4-76。 7. 修改表 5-1。 8. 更新章节 8.7。 9. 更新章节 4.4.2。
A/3	2023-11-08	C.Chai	已过期	1. 更新章节 4.4.4 表格标题。 2. 修改表 20-6、表 20-7、表 19-8、表 4-2。 3. 修改章节 20.4。 4. 修改图 8-1。

SPINTROL

版本	日期	作者	状态	变更
A/4	2023-12-05	C.Chai	已过期	1. 修改章节 2.2。 2. 修改章节 11.2。
A/5	2024-03-08	Jiali Zhou	已过期	1. 修改图 4-1：I/O 控制框图、章节 4.2 2. 修改章节 19 3. 删除 EEPROM 算法 4. 修改图 8-14：信号封锁子模块结构 5. 修改章节 4.3.2、4.3.3
				1. 修改表 25-76 中错误的寄存器描述。 2. 修改表 25-79 MEMIF 错误描述。
C/0	2025-02-08	Jiali Zhou	已发布	1. ADCSTS 改为 SHFLG。 2. 删除寄存器 BOOTINFO 中无效位段。 3. 修改表 25-2、表 25-80 中 MEMIF 描述。 4. 修改寄存器 UARTIF，UARTIC，UARTIE。 5. 修改 PWMCFG 寄存器。 6. 修改章节 4.2.3。 7. 修改表 6-2。

SPINTROL

©2025 旋智电子科技（上海）有限公司

术语或缩写

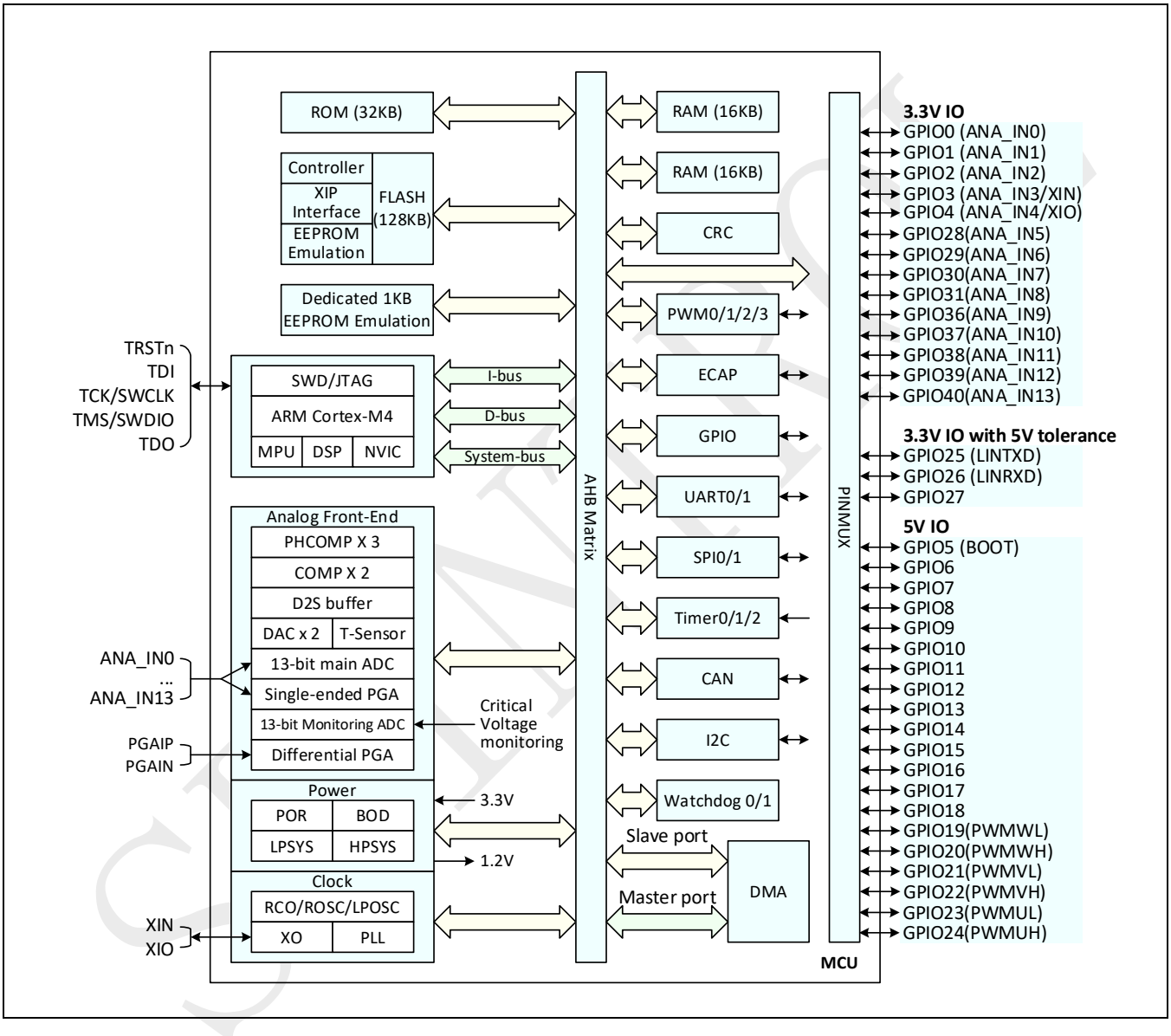
术语或缩写	描述
MCU	Microcontroller Unit（微控制器单元）
AHB	Advanced High Performance Bus（先进高性能总线）
DMA	Direct Memory Access（直接存储器访问）
XIP	Execution In Place（就地执行）
SWD	Serial Wire Debug（串行线调试）
PLL	Phase Locked Loop（锁相环）
BOD	Brownout Detector（过压/欠压检测器）
NMI	Non Maskable Interrupt（不可屏蔽中断）
ISP	In System Programming（在系统编程）
PFD	Phase Frequency Detector（鉴频鉴相器）
NVIC	Nested Vectored Interrupt Controller（嵌套向量中断控制器）
UART	Universal Asynchronous Receiver-Transmitter（通用异步收发器）
LIN	Local Interconnect Network（局域互联网络）
CAN	Controller Area Network（控制器局域网络）
ADC	Analog-to-Digital Converter（模数转换器）
DAC	Digital-to-Analog Converter（数模转换器）
PGA	Programmable-Gain Amplifier（可编程增益放大器）
CRC	Cyclic Redundancy Check（循环冗余校验）
AES	Advanced Encryption Standard（高级加密标准）
SAR ADC	Successive Approximation A/D Converter（逐次逼近型模数转换器）

SPINTROL

1 存储器和总线结构

1.1 系统架构

图 1-1：SPC1169 系统框图



如图 1-1 所示，在 SPC1169 芯片中，主系统由微控制子系统和外设子系统组成，使用低延迟的 AHB 总线架构互连。

微控制子系统包括：

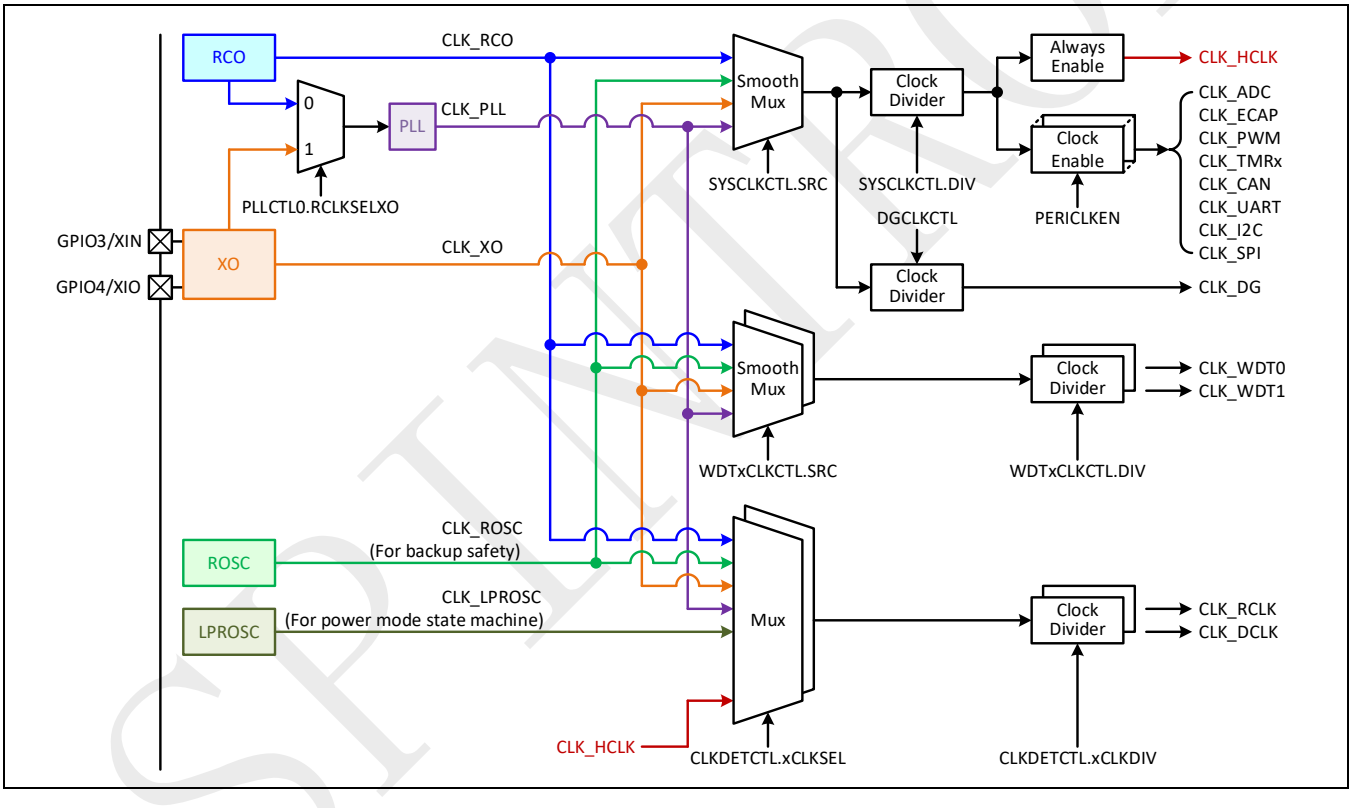
- 电源管理单元
  - LDO（低压差稳压器）产生 1.2V 电源；
  - 进入睡眠模式和唤醒处理；

©2025 旋智电子科技（上海）有限公司

- 4 个主设备
  - Cortex-M4 I 总线、D 总线、S 总线和 DMA 主设备端口；
- 23 个从设备
  - ROM；
  - 两个 RAM；
  - Flash 控制器和 XIP 接口；
  - 电源和时钟；
  - AHB 外设：AFE、PWM、ECAP、WDT0/1、TIMER0/1/2、GPIO、SPI0/1、UART0/1、I2C、CAN、CRC、DMA；

1.2 时钟树

图 1-2：SPC1169 时钟树



如图 1-2 所示，SPC1169 有 5 个根时钟：

- RCO：经过调校为 32MHz；
- ROSC：自由运行的环形振荡器，额定频率为 32MHz，作为 RCO 缺失时的备用安全时钟；
- XO：带有片上有源谐振器的晶振或外部时钟输入（通过 XIN 输入）；
- PLL：提供最高 100MHz 的灵活片上时钟，参考时钟可以选择 RCO 或 XO 时钟；
- LPROSC：用于电源模式状态机；

平滑 MUX 提供无故障切换，为每个功能模块生成专用时钟，具有独立的时钟门控和分频器。

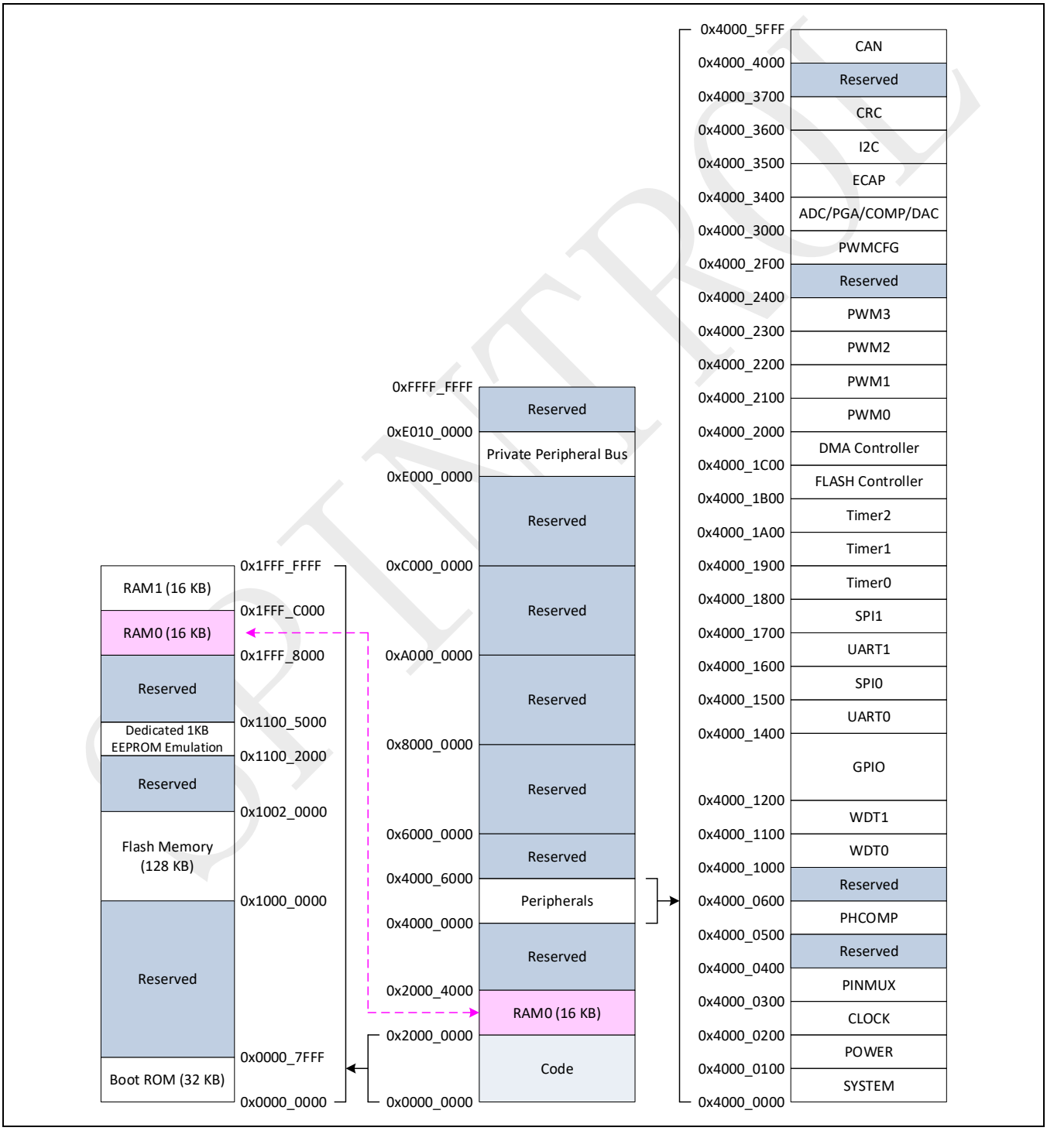
SPINTROL

1.3 存储空间地址映射

SPC1169 将 ROM、RAM、寄存器和 IO 端口分配在同一个线性 4GB 地址空间内。数据字节在存储单元中采用小端模式存放。

未分配给任何存储单元或者外设寄存器的地址视作“未使用的”空间。对这类地址的访问将产生总线错误。SPC1169 的存储器映射如图 1-3 所示。

图 1-3：SPC1169 存储空间地址映射



SPINTROL

©2025 旋智电子科技（上海）有限公司

注意： SRAM0 可以映射到地址范围 0x1FFF 8000 ~ 0x1FFF BFFF（可由 I 总线和 D 总线访问）或 0x2000 0000 ~ 0x2000 3FFF（可由 S 总线访问）。

表 1-1 给出了 SPC1169 各个外设模块的基地址。

表 1-1：外设模块基地址

外设模块	基地址	寄存器列表
SYSTEM	0x4000 0000	章节 25.10.1
POWER	0x4000 0100	章节 2.6.1
CLOCK	0x4000 0200	章节 3.9.1
PINMUX	0x4100 0300	章节 4.4.3
GPIO	0x4000 1200	章节 4.4.1
TIMER0	0x4000 1800	章节 6.4.1
TIMER1	0x4000 1900	章节 6.4.1
TIMER2	0x4000 1A00	章节 6.4.1
WDT0	0x4000 1000	章节 7.4.1
WDT1	0x4000 1100	章节 7.4.1
PWM0	0x4000 2000	章节 8.11.1
PWM1	0x4000 2100	章节 8.11.1
PWM2	0x4000 2200	章节 8.11.1
PWM3	0x4000 2300	章节 8.11.1
PWMCFG	0x4000 2F00	章节 8.11.3
ECAP	0x4000 3400	章节 9.7.1
ADC	0x4000 3000	章节 10.9.1
PGA	0x4000 3000	章节 13.5.1
COMP	0x4000 3000	章节 14.5.1
PHCOMP	0x4000 0500	章节 15.4.1
UART0	0x4000 1400	章节 19.7.1
UART1	0x4000 1600	章节 19.7.1
SPI0	0x4000 1500	章节 20.8.1
SPI1	0x4000 1700	章节 20.8.1
FLASH	0x4000 1B00	章节 24.8.1
DMAC	0x4000 1C00	章节 26.6.3
I2C	0x4000 3500	章节 21.6.1
CRC	0x4000 3600	章节 23.4.1
CAN	0x4000 5400	章节 22.7.1

©2025 旋智电子科技（上海）有限公司

1.3.1 嵌入式 Flash 存储器

SPC1169 集成了一个嵌入式闪存，可以提供最大 128kB 空间用于存储程序代码和数据。如图 1-1 所示，可通过控制器接口访问 Flash，还支持通过特别设计的就地执行（XIP）可口读取 Flash 存储器的内容。因此，在 SPC1169 上电启动后，Flash 存储器中的代码可以直接就地执行，也可以在运行之前将其加载到 RAM 存储器中，再在 RAM 中执行。

为了保证数据的安全性，嵌入式 Flash 支持 ECC 功能，能够纠正 1 Bit 错误并检测 2 Bit 错误（SECDED）。当使能该功能时，Flash ECC 发生多 Bit 错误时会向 CPU 发出不可屏蔽中断（NMI）中断，而发生单 Bit 错误时会向 CPU 发出可屏蔽中断，发生位错误的地址将记录在 FLASHERRADDR 寄存器中，以供诊断使用。

Flash 有 32 个可编程扇区，每个扇区有 4096 字节，可以一次编程 64 位（双字）数据，其擦除方式有 4KB 扇区擦除或整片擦除。

1.3.2 片上 SRAM

SPC1169 带有两块 SRAM 存储器，分别命名为 RAM0 和 RAM1，两者都可以以字节、半字（16 位）或全字（32 位）方式访问，能实现 CPU 零等待访问的功能。

16KB 的 RAM1 可以由 CPU 的 I 总线和 D 总线通过地址范围 0x1FFF C000 ~ 0x1FFF FFFF 进行访问。

RAM0 的大小可达 16KB，可以由 CPU 的 I 总线和 D 总线通过地址范围 0x1FFF 8000 ~ 0x1FFF BFFF 进行访问，或者通过 CPU 的 S 总线通过地址范围 0x2000 0000 ~ 0x2000 3FFF 进行访问。不同的子系列产品有不同的 RAM0 大小，可能为 0KB、4KB（映射到 0x1FFF B000 ~ 0x1FFF BFFF 或 0x2000 3000 ~ 0x2000 3FFF）、8KB（映射到 0x1FFF A000 ~ 0x1FFF BFFF 或 0x2000 2000 ~ 0x2000 3FFF）、12KB（映射到 0x1FFF 9000 ~ 0x1FFF BFFF 或 0x2000 1000 ~ 0x2000 3FFF）或完整的 16KB。

为了保证数据的安全性，RAM0 和 RAM1 都支持 SECDED ECC。当使能该事件时，RAM ECC 发生多位错误时会向 CPU 发出不可屏蔽中断，而发生单位错误时会向 CPU 发出可屏蔽中断，发生位错误的地址将记录在 RAM0ERRADDR 和 RAM1ERRADDR 寄存器中，以供诊断使用。

1.3.3 引导 ROM

SPC1169 内置有 SECDED ECC 支持的 32KB ROM，默认情况下，ROM ECC 发生多比特位错误时系统将复位，并且发生比特位错误的地址将记录在 ROMERRADDR 寄存器中，供诊断使用。当使能 ECC 校验错误触发中断功能时，ROM ECC 发生多比特位错误可以触发一个不可屏蔽中断到 CPU，而 ROM ECC 发生单比特位错误可以触发一个可屏蔽中断。

为了方便使用，与电源管理相关的功能、FLASH 驱动库、进入睡眠模式、进入停止模式等的函数实现被存储在 ROM 中作为硬件库函数，这些函数的入口地址被当做一个表放置在 ROM 偏移为 0x1000 的位置，通过入口表，用户可以轻松调用指定的函数来安全地使系统进入睡眠或停止状态、处理 FLASH 等功能。引导 ROM 的内存映射如图 1-4 所示，硬件库入口表的详细信息列在表 1-2 中。

©2025 旋智电子科技（上海）有限公司

图 1-4：引导 ROM 地址空间映射

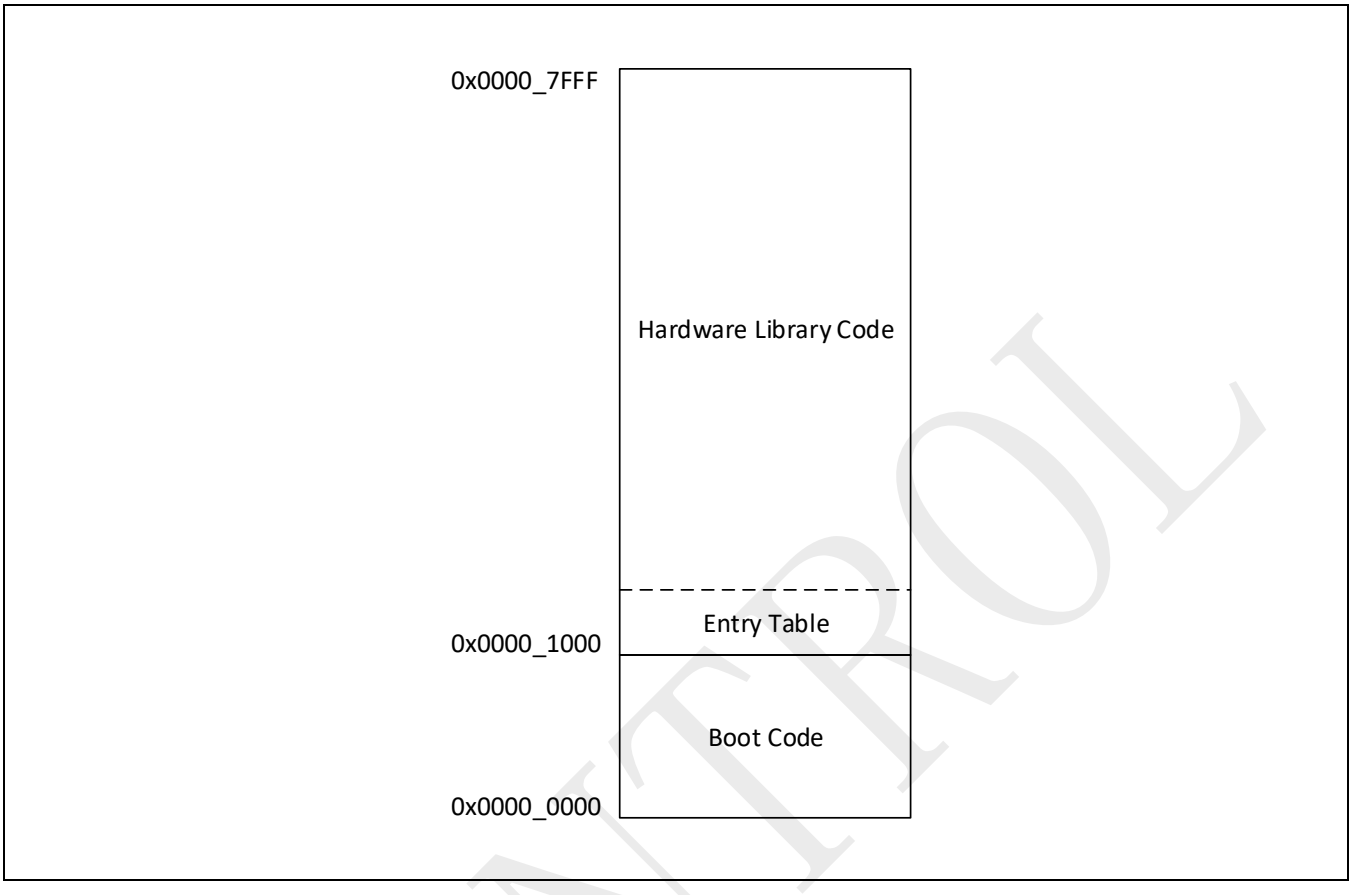


表 1-2：引导 ROM 硬件库入口表

函数名称	地址	描述
FLASHC_Init	0x0000 1000	初始化 Flash 配置
FLASHC_Read	0x0000 1004	读取 Flash 中一个字长度的数据
FLASHC_VerifyRead	0x0000 1008	在擦除或者写入数据后读取相关数据检查擦除或者写入后的数据是否正确。 这个检查的动作，若发生错误也并不会触发 ECC 错误事件。
FLASHC_ProgramDWord	0x0000 100C	写入双字数据
FLASHC_Program	0x0000 1010	写入单字数据
FLASHC_EraseSector	0x0000 1014	擦除一个扇区
FLASHC_EraseChip	0x0000 1018	擦除整个 Flash
FLASHC_VerifyErase	0x0000 101C	擦除动作之后，检查数据是否全为 1，校验失败时不会触发 ECC 错误事件。
FLASHC_PowerUp	0x0000 1020	Flash 上电
FLASHC_PowerDown	0x0000 1024	Flash 下电
FLASHC_WriteProtect	0x0000 1028	使能指定 Flash 区域的保护功能
FLASHC_WriteUnProtect	0x0000 102C	失能指定 Flash 区域的保护功能
FLASHC_EnableCache	0x0000 1030	使能 Flash XIP 的 64 位宽的缓存

©2025 旋智电子科技（上海）有限公司

函数名称	地址	描述
FLASHC_DisableCache	0x0000 1034	失能 Flash XIP 的 64 位宽的缓存
FLASHC_SetTiming	0x0000 1038	设置 Flash 时序参数
-	0x0000 103C	Reserved
-	0x0000 1040	Reserved
-	0x0000 1044	Reserved
-	0x0000 1048	Reserved
-	0x0000 104C	Reserved
-	0x0000 1050	Reserved
-	0x0000 1054	Reserved
-	0x0000 1058	Reserved
-	0x0000 105C	Reserved
SYSTEM_Sleep	0x0000 1060	整个系统进入睡眠模式
SYSTEM_Stop	0x0000 1064	整个系统进入停止模式
MCU_Sleep	0x0000 1068	仅 MCU 进入睡眠模式
MCU_Stop	0x0000 106C	仅 MCU 进入停止模式
SYSTEM_CalculateMemCRC	0x0000 1070	获取指定内存区域数据的 CRC 值
SYSTEM_CheckCriticalRegister	0x0000 1074	检查关键寄存器的复位值
-	0x0000 1078	Reserved
Sin	0x0000 107C	Reserved
Cos	0x0000 1080	Reserved
Sin2	0x0000 1084	Reserved
Cos2	0x0000 1088	Reserved
Atan2	0x0000 108C	Reserved
AtanF32	0x0000 1090	Reserved
FastSqrt	0x0000 1094	Reserved
Sin512	0x0000 1098	Reserved
Cos512	0x0000 109C	Reserved
Atan512	0x0000 10A0	Reserved
ThetaRadWrap	0x0000 10A4	Reserved
SAT0	0x0000 10A8	Reserved
SVPWM	0x0000 10AC	Reserved
ClarkeF32	0x0000 10B0	Reserved
InvParkF32	0x0000 10B4	Reserved
ParkF32	0x0000 10B8	Reserved
BubbleSortForCurrent	0x0000 10BC	Reserved
Motor_1PhaseCurrentReconstructInit	0x0000 10C0	Reserved
Motor_1PhaseCurrentReconstruct	0x0000 10C4	Reserved
Motor_1PhaseCurrentSamplingShift	0x0000 10C8	Reserved

©2025 旋智电子科技（上海）有限公司

1.4 启动引导配置

在 SPC1169 中，可通过 BOOT 引脚（GPIO5）选择两种不同的启动模式：

- 从 Flash 启动（BOOT 引脚为低电平）：在复位后，引导加载程序跳转到 Flash 中，并从地址 0x1000 0000 开始运行。
- ISP 模式（BOOT 引脚为高电平）：引导加载程序通过 UART 或 LIN 接口重新烧录程序到 Flash。在此过程中，如果使用 UART 接口，GPIO10 被配置为 UART_TXD，GPIO11 被配置为 UART_RXD。

表 1-3：启动模式

BOOT（GPIO5）	启动模式
1	ISP 模式 （如果使用 UART 接口，GPIO10/GPIO11 将被配置成 UART_TXD/UART_RXD 功能）
0	从 Flash 正常启动 **注意：当 TRSTn 为高，调试接口将处于使能状态。**

注意： ISP 模式只能在上电复位（由重新上电或者 XRSTn 引脚被低电平保持足够时间而触发）后 BOOT（GPIO5）为高电平时进入。对于其他事件触发的复位（如看门狗超时或内存多比特位 ECC 错误），系统将进入正常启动模式，并且不论 BOOT（GPIO5）引脚的状态如何，都将从 Flash 中启动。

SPINTROL

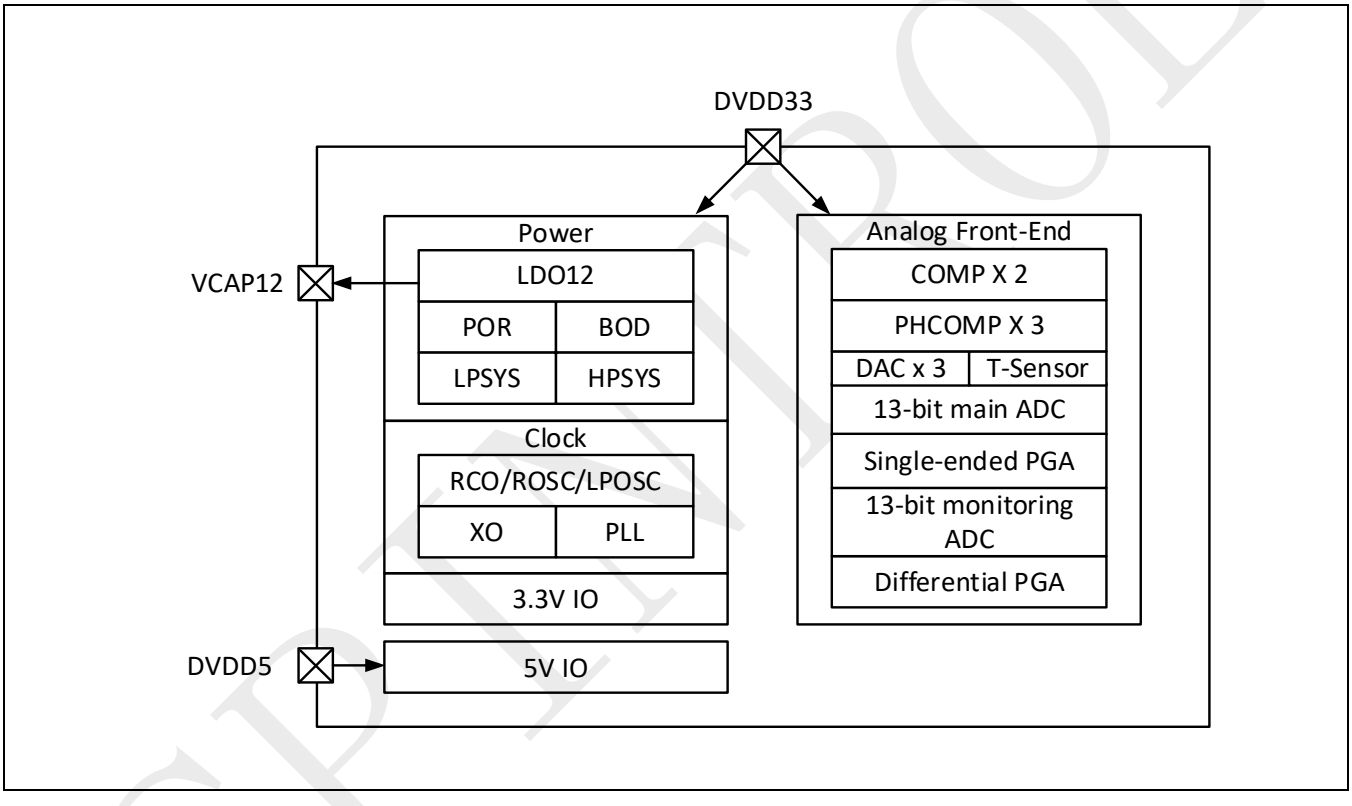
2 电源

2.1 电源供电

如图 2-1 所示，SPC1169 需要有两个电源供电（DVDD33 and DVDD5）。DVDD5（5V）仅被用于 5V IO。

DVDD33（3.3V）被用于内部电路和 3.3V IO。由 DVDD33 供电的 MCU 内部稳压器产生 1.2V 给 MCU 供电。它需要用于外部去耦的分离引脚 VCAP12。

图 2-1：电源系统总览

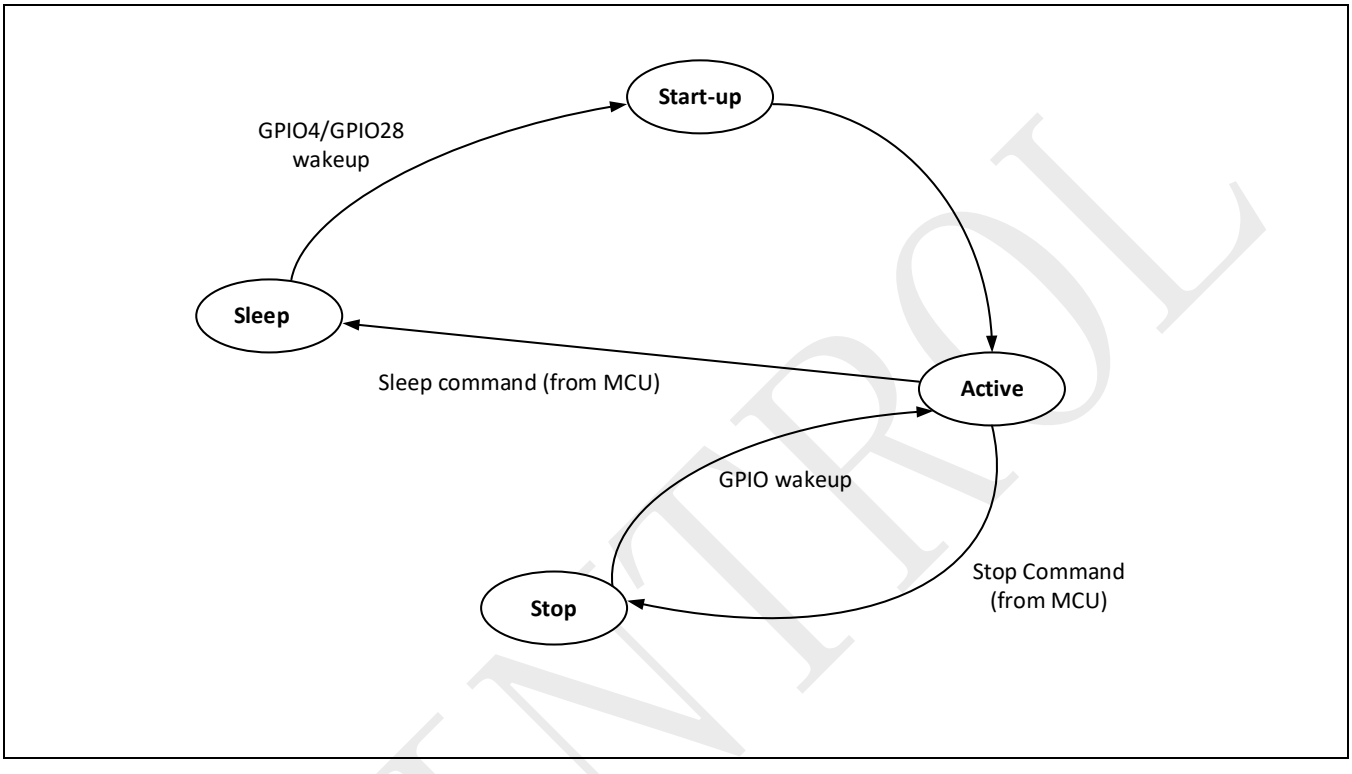


©2025 旋智电子科技（上海）有限公司

2.2 电源管理单元

电源管理单元（PMU）负责为嵌入式 MCU（VCAP12）所需的所有电压供应。PMU 具有明确定义的状态机，用于控制电源模式的转换，如图 2-2 所示。

图 2-2：电源模式转换



SPC1169 提供的启动过程和 3 种电源模式的定义为：

活动模式：

稳压器（VCAP12）由芯片内部具有全电流驱动能力的 LDO 供电；所有时钟根据用户在寄存器控制位中的设置而启用。

停止模式：

除了用于电源模式状态机的 128 KHz 时钟外，所有时钟都停止；VCAP12 由具有有限电流驱动能力的低功耗 LDO 供电；VCAP12 可以配置为低于 1.2V 以进一步降低静态电流；所有寄存器和内存保持内容不变。系统可以根据配置由任意 GPIO 引脚上的任意电平唤醒。当退出停止模式时，CPU 将继续执行下一条指令。

睡眠模式：

稳压器（VCAP12）完全关闭。它可以通过 GPIO 唤醒，整个系统将进行冷启动。

注意：
1. 在发送停止命令之前，用户应安全地禁用未使用的模拟模块。
2. 在停止模式下，泄漏电流在高温下显著增加，并且可能超过低功耗 LDO 的负载能力。在发出停止命令之前，用户需要确保结温低于 100°C。

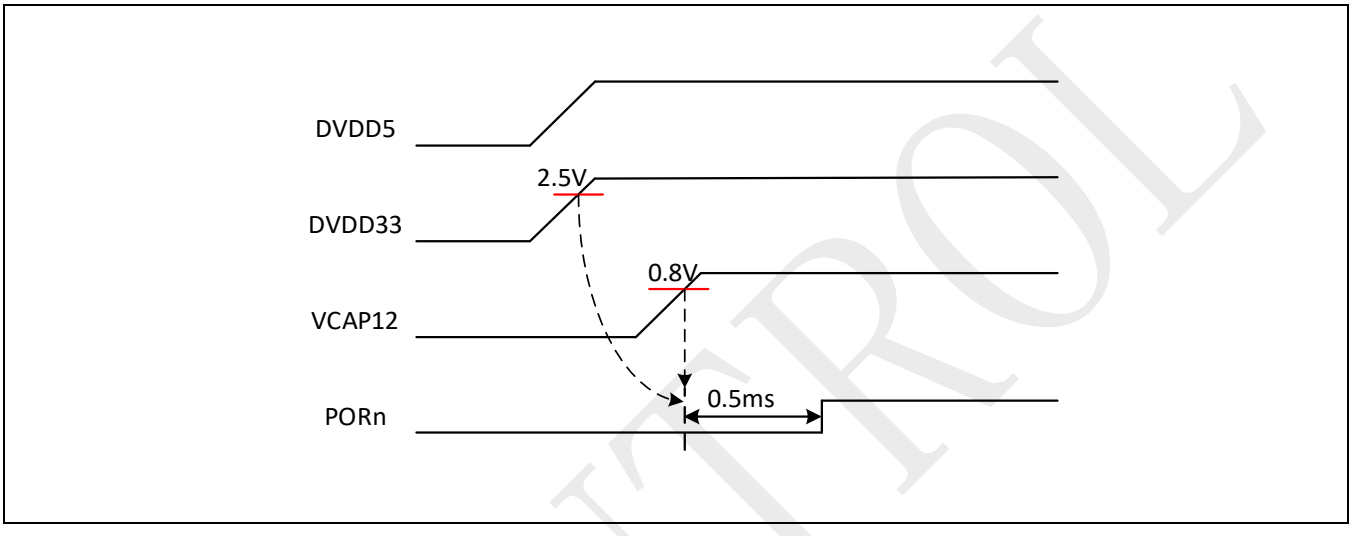
©2025 旋智电子科技（上海）有限公司

2.3 电源产生单元

SPC1169 的上电顺序如图 2-3 所示。当 DVDD33 升至 2.5V 时，VCAP12 升至 0.8V，经过 0.5ms，MCU 核心逻辑的 PORn 被释放，SPC1169 进入主动模式。

DVDD5 和 DVDD33 之间没有任何通电顺序。因为 DVDD5 只用于 5V IO，它没有参与系统上电序列。

图 2-3：上电时序



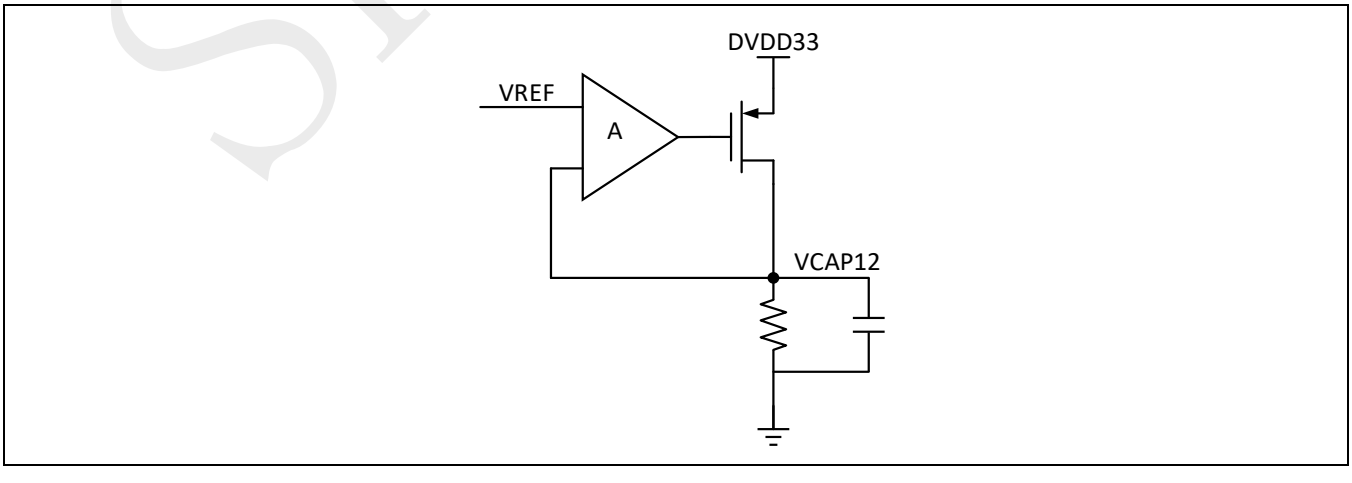
2.3.1 VCAP12 LDO

VCAP12 LDO 是为嵌入式 MCU 提供电源的模块。SPC1169 实现了以下 VCAP12 LDO 模块的特性：

- 1.2V LDO，具有 40mA 的驱动能力

VCAP12 LDO 的原理图如图 2-4 所示。

图 2-4：VCAP12 LDO



©2025 旋智电子科技（上海）有限公司

2.4 低压过压/欠压检测

过压/欠压检测通过低电压探测器（BOD）实现，BOD 可以通过将电压与预设的阈值进行比较来监测各种电源电平。本节详细描述了适用于 DVDD33 和 VCAP12 的低压 BOD。

在 NMIF 寄存器中，可用 VDD33OV/VDD12OV 和 VDD33UV/VDD12UV 标志指示 3.3V/1.2V 域是高于还是低于预设的阈值。如果使能，VDD33OV/VDD12OV 和 VDD33UV/VDD12UV 事件可以生成中断。

BOD33CTL 和 BOD12CTL 寄存器确定如何设置过压和欠压阈值，NMIE 寄存器使能 3.3V/1.2V 域过压和欠压事件触发 NMI 中断，RSTEVTEN 寄存器使能在 3.3V/1.2V 域过压和欠压事件时进行复位的功能。

图 2-5 和图 2-6 显示了 3.3V 和 1.2V 域的低压探测器结构。

图 2-5：3.3V 电压域失调探测框图

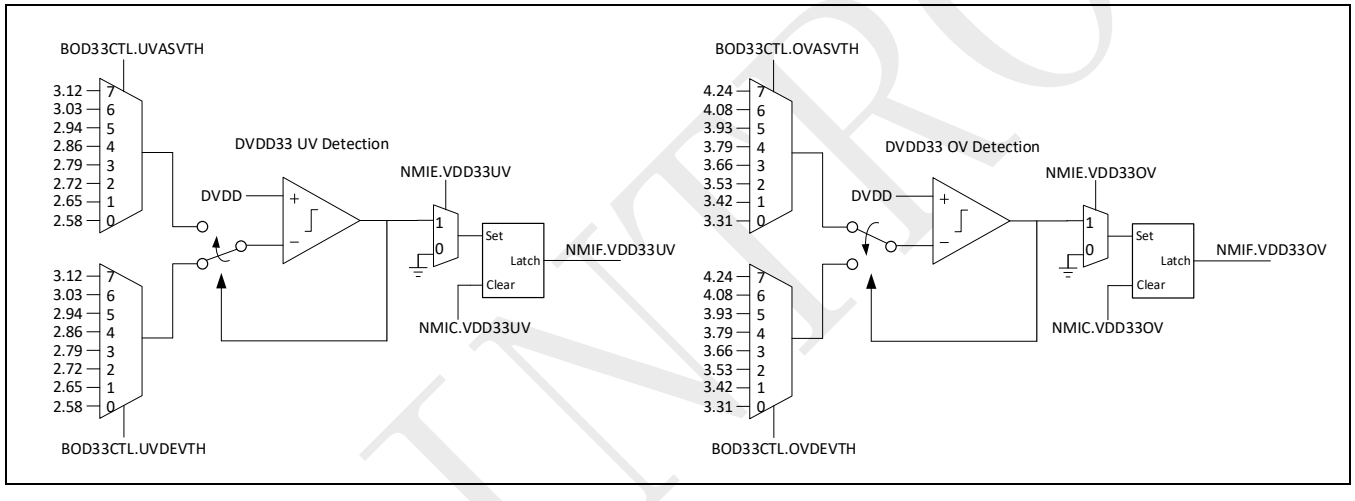
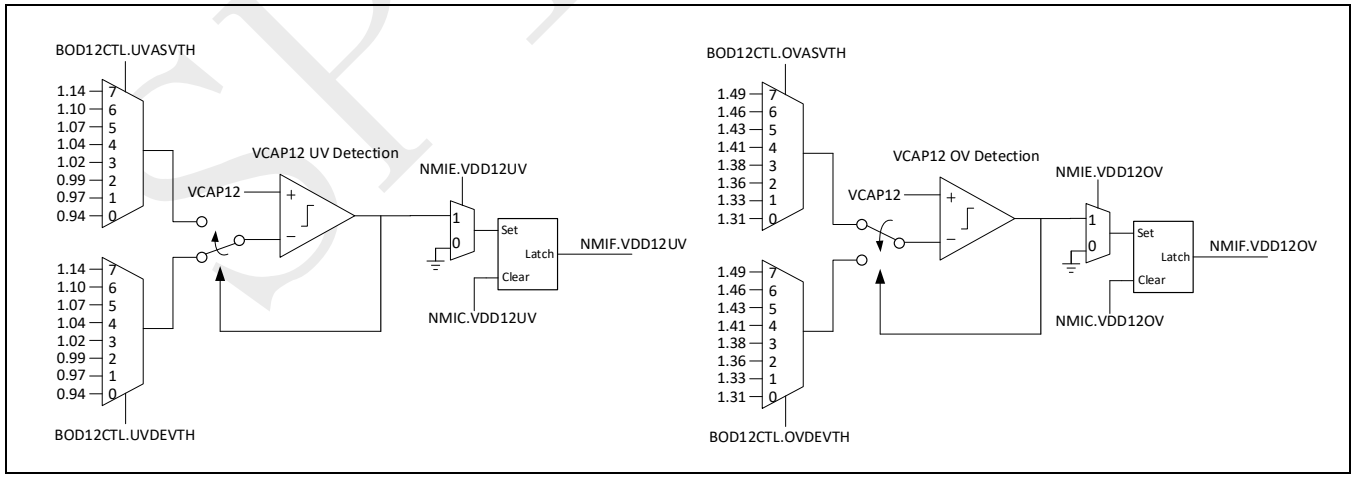


图 2-6：1.2V 电压域电压失调探测框图

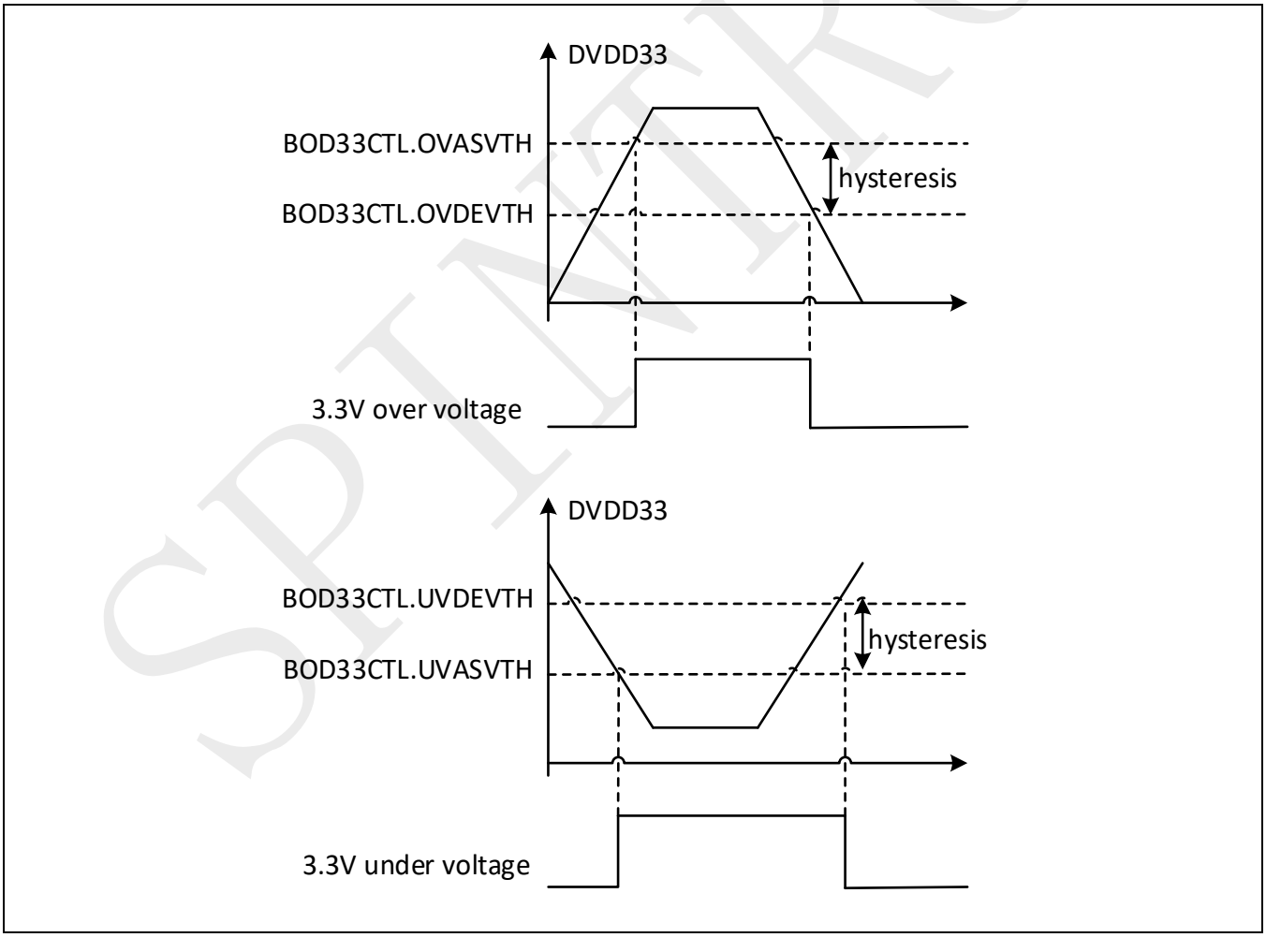


注意： 为了确保 BOD 的正常操作，必须满足以下条件：

1. 在过压检测中：BOD33CTL 寄存器中的过渡阈值 OVASVTH（过压事件触发阈值）和 BOD12CTL 寄存器中的过渡阈值 OVASVTH 必须大于 BOD33CTL 寄存器中的过渡阈值 OVDEVTH（过压事件消失阈值）和 BOD12CTL 寄存器中的过渡阈值 OVDEVTH。
2. 在欠压检测中：BOD33CTL 寄存器中的过渡阈值 UVASVTH（欠压事件触发阈值）和 BOD12CTL 寄存器中的过渡阈值 UVASVTH 必须小于 BOD33CTL 寄存器中的过渡阈值 UVDEVTH（欠压事件消失阈值）和 BOD12CTL 寄存器中的过渡阈值 UVDEVTH。

当 DVDD33 上升超过过压阈值或下降低于欠压阈值时，会产生 3.3V 域过压或欠压中断，如图 2-7 所示。

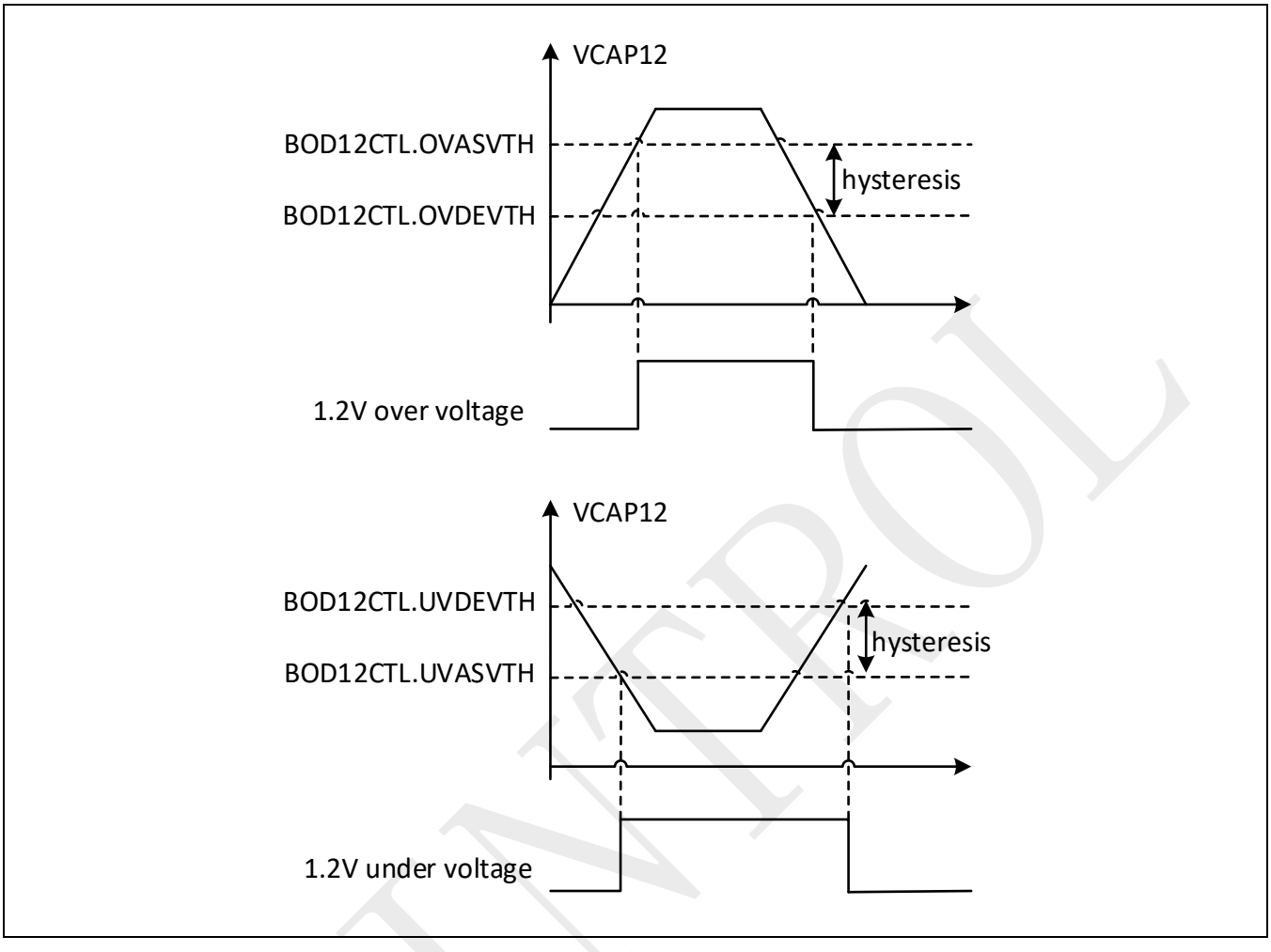
图 2-7：DVDD33 阈值



当 VCAP12 上升超过过压阈值或下降低于欠压阈值时，会产生 1.2V 域过压或欠压中断，如图 2-8 所示。

©2025 旋智电子科技（上海）有限公司

图 2-8：VCAP12 阈值



2.5 系统唤醒

- 停止模式：
  - GPIO 引脚的任何边缘变化都可以用作停止模式下的唤醒源。GPIO 选择如表 25-66 所示。
- 睡眠模式：
  - 只有 GPIO4 和 GPIO28 引脚可以用作睡眠模式下的唤醒源。寄存器的详细描述如表 25-66 所示。

SPINTROL

2.6 寄存器

2.6.1 POWER 寄存器列表

表 2-1：POWER 模块基地址

外设模块	基地址
POWER	0x4000 0100

表 2-2：POWER 寄存器列表

Register	Offset	描述	Reset Value
LDOCTL*	0x0	LDO 控制寄存器	0x00000007
BOD33CTL*	0x4	3.3V BOD 控制寄存器	0x06070100
BOD12CTL*	0x8	1.2V BOD 控制寄存器	0x06070100
PWRMODECMD*	0xC	电源模式命令寄存器	0x00000000
PWRREGKEY	0x10	POWER 模块写使能寄存器	0x1ACCE551

注意： 由*标记的寄存器仅当 PWRREGKEY=0x1ACCE551 时才可以改写。

SPINTROL

©2025 旋智电子科技（上海）有限公司

2.6.2 POWER 寄存器

表 2-3：LDO 控制寄存器（LDOCTL）位段定义

LDOCTL (Power Control Register) Offset：0x0 Default：0x00000007 Access：POWER->LDOCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				VOUT			

表 2-4：Power Control Register (LDOCTL) 描述

位段	位段名	属性	复位值	描述
31:4	RESERVED_31_4	RO	0x0	保留
3:0	VOUT	RW	0x7	1.2V LDO 输出电压 当本字段低于某个值（通常小于 4）时，输出电压将将维持在某最低值； 当本字段高于某个值（通常高于 11）时，输出电压将维持在某最高值； 具体的阈值随芯片而不同。 0000：1.06V 0001：1.08V 0010：1.10V 0011：1.12V 0100：1.14V 0101：1.16V 0110：1.18V 0111：1.20V 1000：1.22V 1001：1.24V 1010：1.26V 1011：1.28V 1100：1.30V 1101：1.32V 1110：1.34V 1111：1.36V

SPINTROL

表 2-5：3.3V BOD 控制寄存器（BOD33CTL）位段定义

BOD33CTL (3.3V BOD Control Register) Offset：0x4 Default：0x06070100 Access：POWER->BOD33CTL							
31	30	29	28	27	26	25	24
RESERVED					OVDEVTH		
23	22	21	20	19	18	17	16
RESERVED					OVASVTH		
15	14	13	12	11	10	9	8
RESERVED					UVDEVTH		
7	6	5	4	3	2	1	0
RESERVED					UVASVTH		

表 2-6：3.3V BOD 控制寄存器（BOD33CTL）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	OVDEVTH	RW	0x6	3.3V 电源下降时撤销 VDD33OV 的阈值 000：3.31V 001：3.42V 010：3.53V 011：3.66V 100：3.79V 101：3.93V 110：4.08V 111：4.24V
23:19	RESERVED_23_19	RO	0x0	保留
18:16	OVASVTH	RW	0x7	3.3V 电源上升时触发 VDD33OV 的阈值 000：3.31V 001：3.42V 010：3.53V 011：3.66V 100：3.79V 101：3.93V 110：4.08V 111：4.24V
15:11	RESERVED_15_11	RO	0x0	保留
10:8	UVDEVTH	RW	0x1	3.3V 电源上升时撤销 VDD33UV 的阈值 000：2.58V 001：2.65V 010：2.72V 011：2.79V 100：2.86V 101：2.94V

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				110：3.03V 111：3.12V
7:3	RESERVED_7_3	RO	0x0	保留
2:0	UVASVTH	RW	0x0	3.3V 电源下降时触发 VDD33UV 的阈值 000：2.58V 001：2.65V 010：2.72V 011：2.79V 100：2.86V 101：2.94V 110：3.03V 111：3.12V

表 2-7：1.2V BOD 控制寄存器（BOD12CTL）位段定义

BOD12CTL (1.2V BOD Control Register) Offset：0x8 Default：0x06070100 Access：POWER->BOD12CTL							
31	30	29	28	27	26	25	24
RESERVED					OVDEVTH		
23	22	21	20	19	18	17	16
RESERVED					OVASVTH		
15	14	13	12	11	10	9	8
RESERVED					UVDEVTH		
7	6	5	4	3	2	1	0
RESERVED					UVASVTH		

表 2-8：1.2V BOD 控制寄存器（BOD12CTL）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	OVDEVTH	RW	0x6	1.2V 电源下降时撤销 VDD12OV 的阈值 000：1.31V 001：1.33V 010：1.36V 011：1.38V 100：1.41V 101：1.43V 110：1.46V 111：1.49V
23:19	RESERVED_23_19	RO	0x0	保留
18:16	OVASVTH	RW	0x7	1.2V 电源上升时触发 VDD12OV 的阈值 000：1.31V 001：1.33V

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				010：1.36V 011：1.38V 100：1.41V 101：1.43V 110：1.46V 111：1.49V
15:11	RESERVED_15_11	RO	0x0	保留
10:8	UVDEVTH	RW	0x1	1.2V 电源上升时撤销 VDD12UV 的阈值 000：0.94V 001：0.97V 010：0.99V 011：1.02V 100：1.04V 101：1.07V 110：1.10V 111：1.14V
7:3	RESERVED_7_3	RO	0x0	保留
2:0	UVASVTH	RW	0x0	1.2V 电源下降时触发 VDD12UV 的阈值 000：0.94V 001：0.97V 010：0.99V 011：1.02V 100：1.04V 101：1.07V 110：1.10V 111：1.14V

表 2-9：电源模式命令寄存器（PWRMODECMD）位段定义

PWRMODECMD (Power Mode Command Register) Offset：0xC Default：0x00000000 Access：POWER->PWRMODECMD							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

©2025 旋智电子科技（上海）有限公司

表 2-10：电源模式命令寄存器（PWRMODECMD）位段定义

位段	位段名	属性	复位值	描述
31:0	KEY	W1S	0x0	本寄存器仅当 PWRREGKEY=0xACCE55CD 时才有写权限。 写 0xA5A51EE9 进入睡眠模式；写 0x5A5A5709 进入停机模式。回读值始终为 0

表 2-11：POWER 模块写使能寄存器（PWRREGKEY）位段定义

PWRREGKEY (Power Register Write-Allow Key Register) Offset：0x10 Default：0x1ACCE551 Access：POWER->PWRREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

表 2-12：POWER 模块写使能寄存器（PWRREGKEY）位段描述

位段	位段名	属性	复位值	描述
31:0	KEY	RW	0x1ACCE551	写入 0xACCE55CD 解锁 PWRMODECMD 寄存器； 写入 0x1ACCE551 解锁其它受保护的 POWER 寄存器。

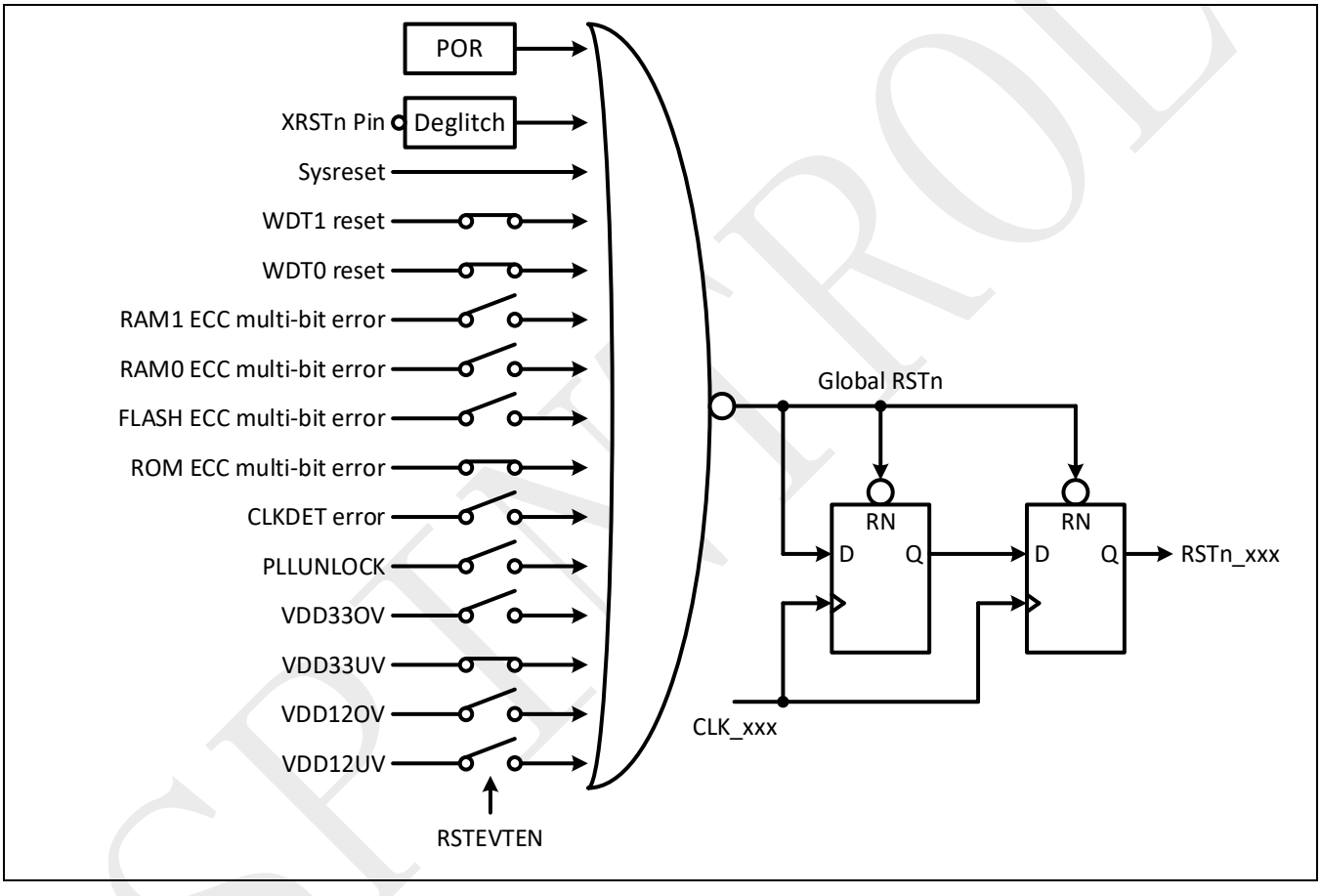
SPINTROL

3 时钟和复位

3.1 复位框架

SPC1169 的全局主动低电平复位信号来自上电复位（POR）模块、XRSTn 引脚和功能逻辑复位信号的或非（NOR）逻辑。在每个外设的时钟域中，该信号经过异步触发和同步释放的处理，以确保时序完整性，如图 3-1 所示。功能复位源可以是内存错误、电压监测器事件、时钟错误事件、看门狗计时器和 Cortex-M4 的 SYSRESET 请求等。

图 3-1：复位架构



作为诊断功能，RSTEVTFLG 寄存器记录功能复位事件，可以通过 RSTEVTCLR 寄存器清除对应位。此外，功能复位源可以通过 RSTEVTEN 寄存器进行编程设置。具体细节请参考表 25-89 和表 25-90。

3.2 引脚复位

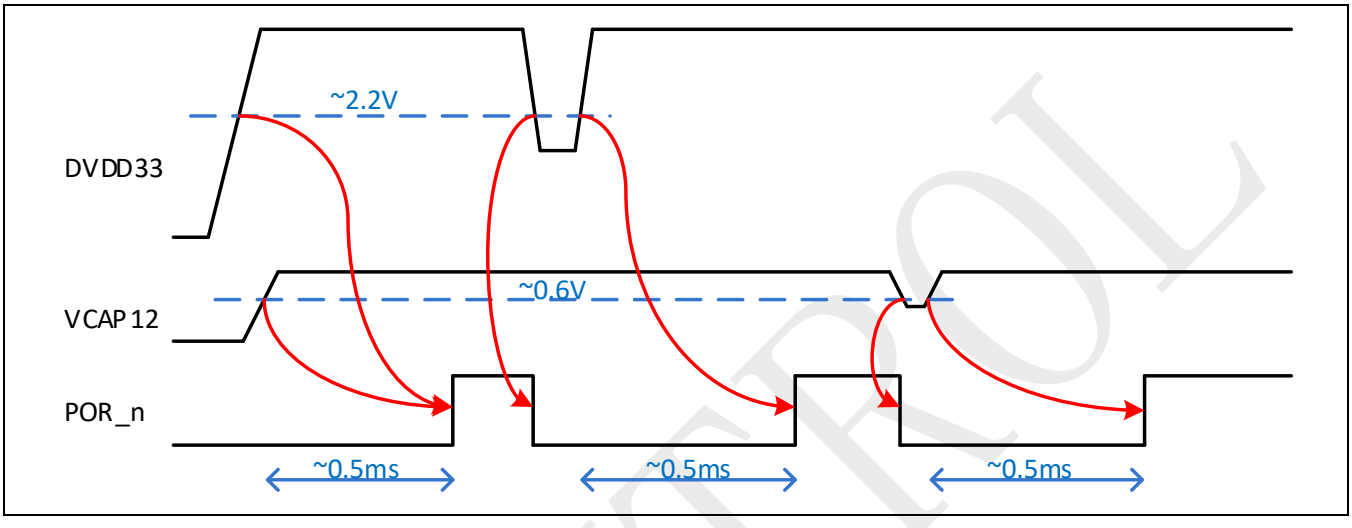
SPC1169 的 XRSTn 引脚为所有功能模块提供全局复位，芯片内部将它上拉，以使芯片处于正常操作模式。原始的 XRSTn 信号经过毛刺滤波器处理，为了触发有效的复位，XRSTn 引脚应该保持低电平的时间应长于通常的 500 微秒。此外，如果 XRSTn 引脚保持低电平超过通常的 5 秒，将进一步触发对所有模拟模块的复位，包括 3.3V 逻辑，在这种情况下，它类似于上电复位。

©2025 旋智电子科技（上海）有限公司

3.3 上电复位

SPC1169 内置的 POR 电路确保了所有上电复位序列的要求，使芯片易于使用。在上电后约 0.5 毫秒内，它将被激活。全局有效低电平复位信号 POR_n 会在 DVDD33 和 VCAP12 的电压水平分别低于约 2.2V 和 0.6V 时变为低电平，如图 3-2 所示。

图 3-2：上电复位时序图



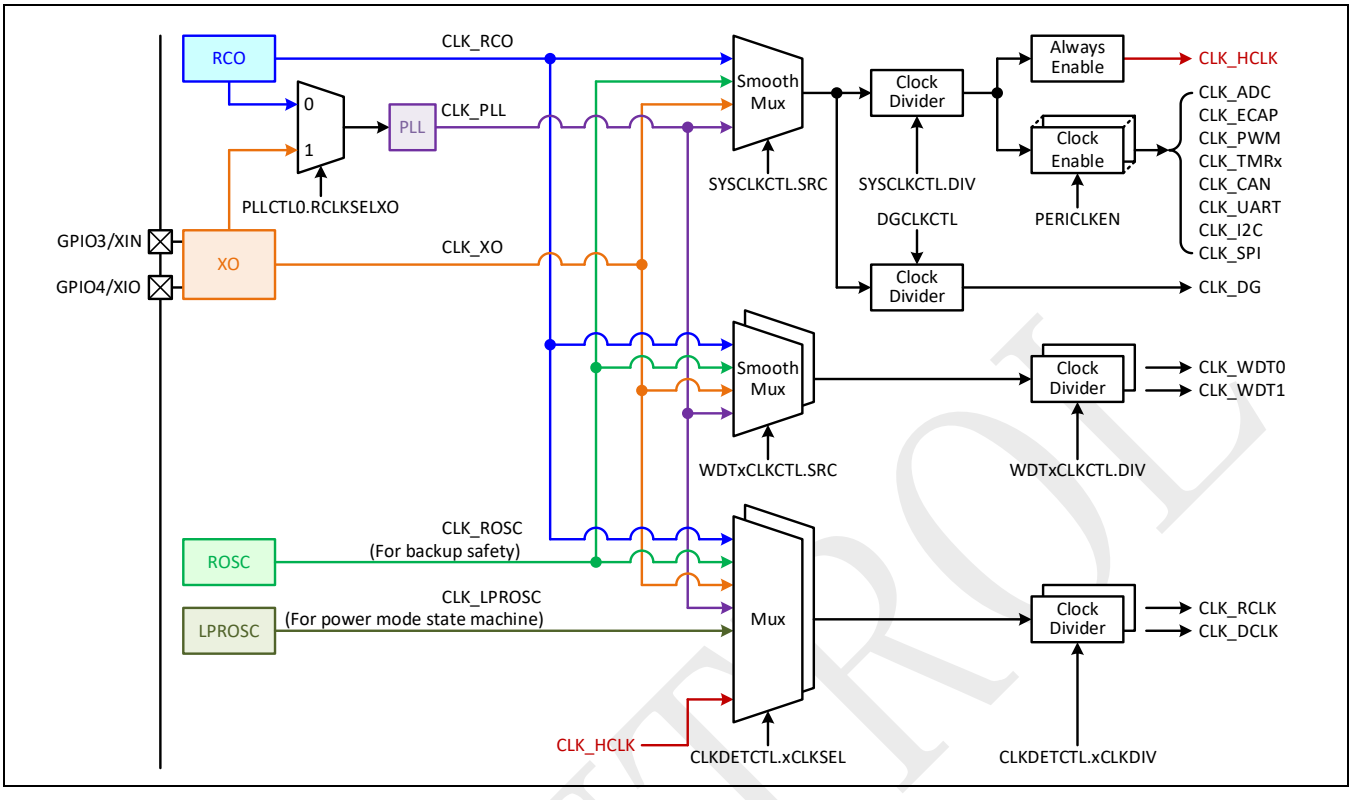
3.4 时钟架构

SPC1169 可以通过内部 RC 振荡器、环形振荡器或 PLL 进行时钟驱动。它还可以通过外部振荡器或连接到芯片上振荡器电路的晶振来提供时钟。PLL 支持 25MHz 到 100MHz 的高频时钟，而 PLL 的输入频率可以从 4MHz 到 56MHz。结合可编程分频器和门控模块，SPC1169 实现了灵活的时钟树，如图 3-3 所示。

注意：
1. 在使用外设模块之前，用户应首先启用相应的时钟。
2. 时钟信号可以引出到 GPIO2 进行观察。详细信息请参考表 4-1。

SPINTROL

图 3-3：时钟树



3.5 高频振荡器

SPC1169 上实现了两个零引脚振荡器，其中一个经过调整，使其在温度变化时频率变化最小，并且中心频率为 32MHz，另一个是自由运行的环形振荡器，其典型频率为 32MHz，主要用于时钟安全功能。

3.6 内部晶振和外部时钟

晶体振荡器可以使用片上和外部模式，对于内部模式，需要连接一个频率范围为 4MHz 到 32MHz 的晶体，如图 3-4 所示。

为了限定晶体内的功耗以提高频率长期稳定性，晶振电路内部包含一个幅度控制逻辑。图 3-4 显示了晶体振荡器的结构，它由一个与晶体一起工作的振荡器和一个幅度控制模块组成，该模块能够在振荡过程中感知 XIN 和 XOUT 引脚之间的振幅，并将其调整到预设值。

对于内部晶振模式，有两种方法可用于控制 XIN 和 XOUT 引脚之间的振荡幅度：

第一种方法称为开环幅度控制，通过设置寄存器 XOCTL.FRCCODE 来调节，最大可提供的电流为 2mA（此时需要设置 XOCTL.FRCCODEEN 为 1）。在这种模式下，用户需要选择适当的供电电流以实现安全的振荡。

第二种方法称为闭环幅度控制。通过寄存器 XOCTL.AMP 设置所需的振荡幅度（可选择 0.32V 至 1.33V 的幅度电压），幅度控制模块将感知 XIN 和 XOUT 之间的振幅，并将其与预设值进行比较。如果振荡幅度小于或大于设定幅度，逻辑引擎将增加偏置电流，直到它在 XOCTL.AMP 定义的振幅电压级别内达到预设振幅。为了调整振幅，需要等待一定的时钟周期，以确保控制回路的

©2025 旋智电子科技（上海）有限公司

稳定性。可以通过寄存器 XOCTL.ACTLPRD 调整此时钟周期，默认推荐值为 2，即 8192 个时钟周期，此值应与晶体的 Q 值大致相同。在启用 XO 之后，会在 XIN 引脚注入 69.6 微秒的伪随机噪声。然后可以通过选择寄存器 XOCTL.INITCODE 中的 4 个 MSB 来设置 DAC 电流，推荐将其值设置为最大值以实现安全启动。如果用户在幅度校准后大致知道最终的控制代码，可以将 XOCTL.OPCODE 设置为接近最终值的值，该值定义了在启动后一个更新时钟周期内的 4 个 MSB 代码，以减少校准时间。校准完成后，如果由于某种原因振幅超出了预设范围，将重新开始校准过程。为避免误触发校准过程，可以将寄存器 XOCTL.ACTLRELAX 设置为 1，这将增加幅度校准的触发窗口。

对于外部模式，如果 XOCTL.BYPASS=1，则晶振时钟输出将按照 XIN 输入进行，如图 3-5 所示。

图 3-4：内部晶振模式

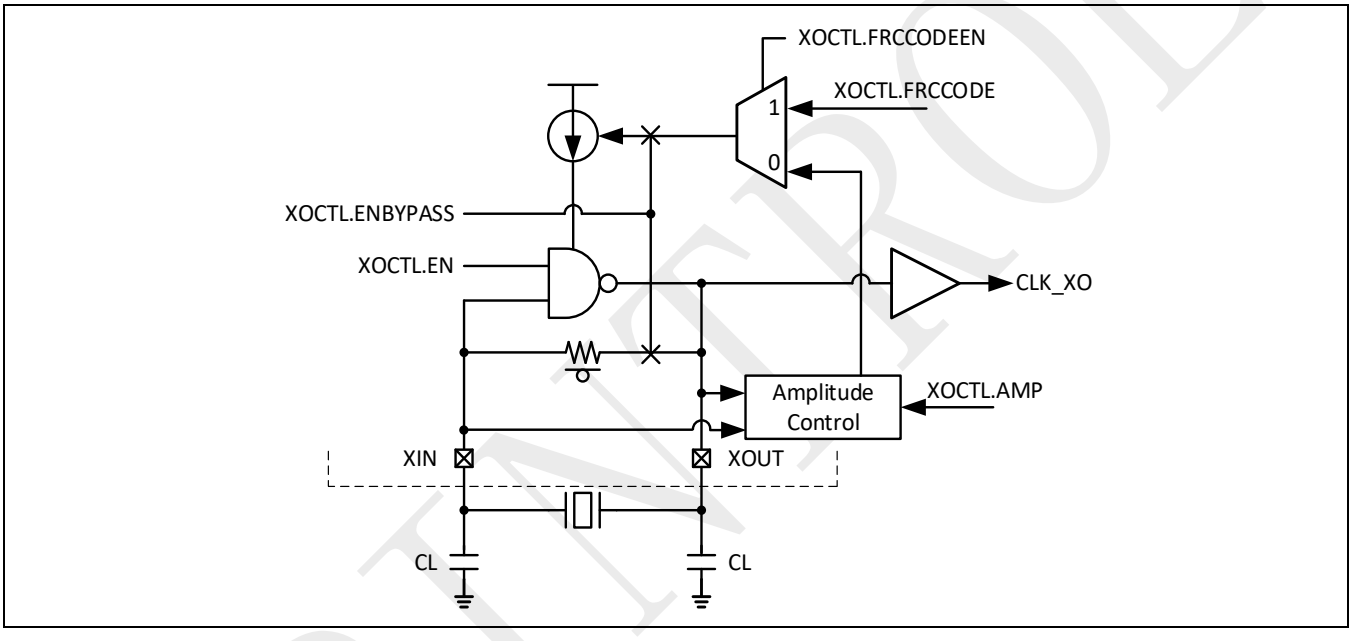
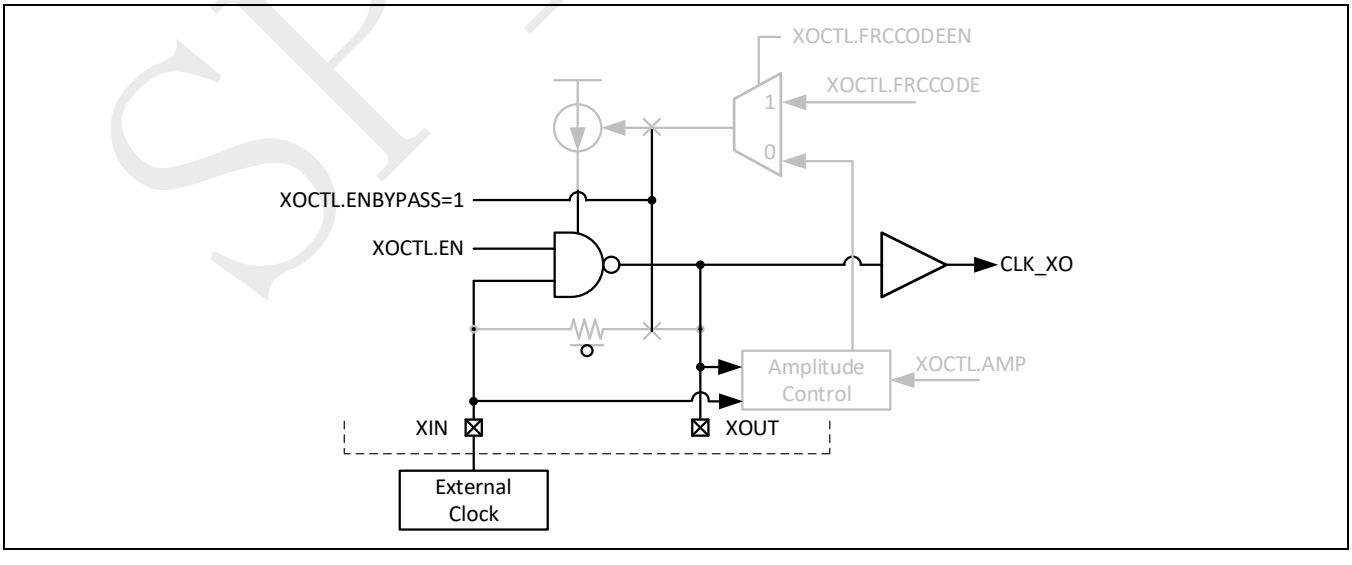


图 3-5：外部时钟



SPINTROL

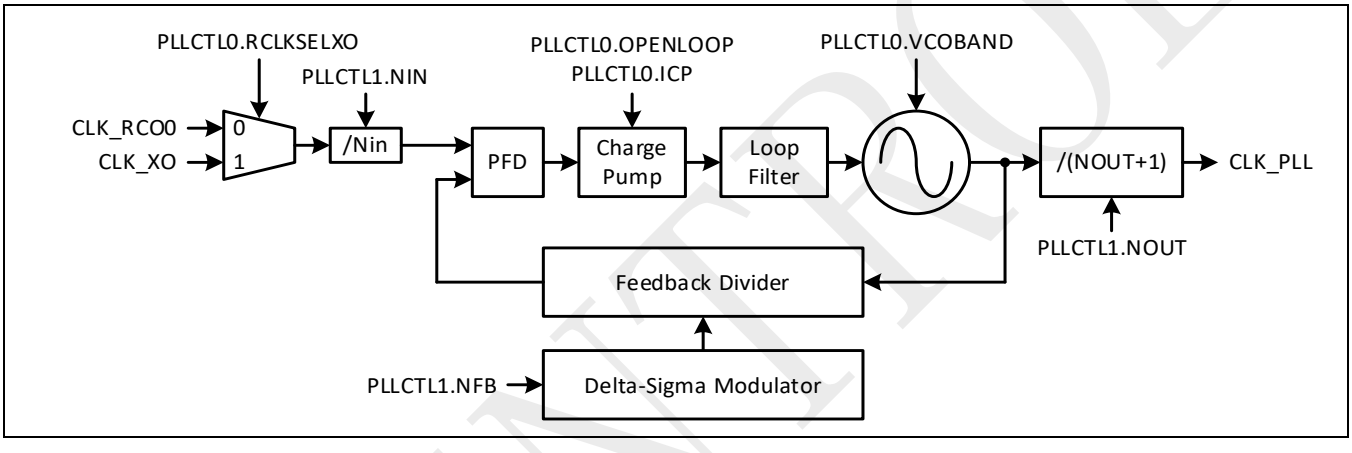
注意： 当将 CLK_XO 用作 PLL 的参考时，外部时钟频率应在 4MHz 到 56MHz 范围内。

3.7 PLL

SPC1169 中的锁相环（PLL）的结构如图 3-6 所示，它可以从 RCO 或外部晶振接收时钟作为输入参考。当锁相环被锁定时，VCO 的频率将在 400MHz 到 600MHz 之间，然后，通过可编程分频器将其分频，提供通常为 25MHz 到 100MHz 的时钟，输出频率可以通过以下公式计算：

$$f_{PLL} = \frac{f_{REF}}{NIN} \times \frac{NFB}{65536} \times \frac{1}{NOUT + 1}$$

图 3-6：PLL 结构



3.7.1 输入参考和分频

PLL 的输入参考时钟（f_{REF}）为 4MHz 到 56MHz，最终送至 PFD（鉴频鉴相器）的输入（f_{PFD}）为从 4MHz 到 8MHz。因此，输入分频比（f_{REF}/f_{PFD}）应当如表 3-1 所示配置。此处的 NIN 是用户配置 PLL 时需要计算的第一个参数。

表 3-1：输入分频比设置

输入频率（f_{REF}）	输入分频比（NIN = f_{REF} / f_{PFD}）
4~8	1
8~16	2
16~24	3
24~32	4
32~40	5
40~48	6
48~56	7

©2025 旋智电子科技（上海）有限公司

3.7.2 输出分频

VCO 的调频范围为 400MHz 到 600MHz，结合输出分频，最终时钟 CLK_PLL 的频率如表 3-2 所示，此处的 NOUT 是用户需要计算的第二个参数。

表 3-2：输出时钟频率和分频比

输出频率（f_{PLL}）			分频比（f_{VCO}/f_{PLL} = NOUT + 1）
75MHz	~	100MHz	6
50MHz	~	75MHz	8
37.5MHz	~	50MHz	12
25MHz	~	37.5MHz	16

3.7.3 环路分频

根据从表 3-1 和表 3-2 得出的 NIN 和 NOUT 的准则，用户可以计算出第三个寄存器控制项：

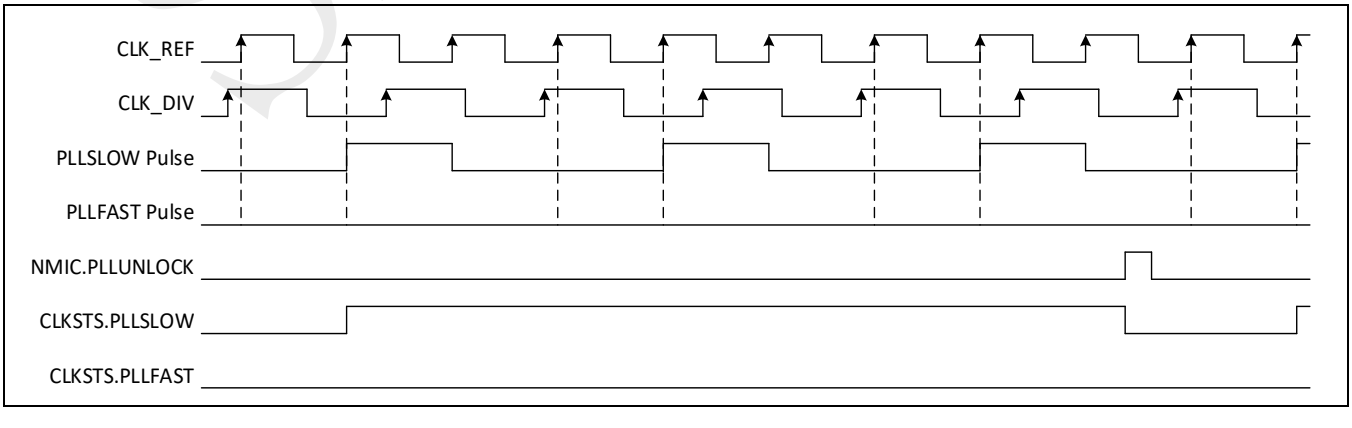
$$NFB = \frac{f_{PLL} \times (NOUT + 1)}{f_{REF} \,/\, NIN} \times 65536$$

3.7.4 PLL 失锁检测

SPC1169 的 PLL 在 PFD 中内嵌了环路失锁检测电路，以提供环路状态侦测信息。具体原理为：对于 PFD 的两个输入，如果在某一个时钟相邻上升沿之间，另一个时钟出现了两个及以上的上升沿，则产生 PLLFAST 或 PLLSLOW 脉冲，并更新 CLKSTS 寄存器。用户可以通过对 NMIC.PLLUNLOCK 写 1 来清除这些标志位。图 3-7 给出了 PFD 输入的参考时钟快于分频器输出时钟并使得 CLKSTS.PLLSLOW 置位的例子。

在 PLL 初始建立或者环路分频有很大改变的时候，f_{PFD} 和 f_{DIV} 有较大差别，导致有不可避免的 PLLFAST 或者 PLLSLOW 事件被锁存到 CLKSTS 寄存器中。为了避免误报，使用时需要等待约 10us 以待环路锁定，随后对 NMIC.PLLUNLOCK 写 1 以清除建立过程中可能误报的失锁，然后才能根据需要通过 NMIE.PLLUNLOCK 来使能 PLL 失锁事件中断。一旦 PLL 锁定，将不应当再出现 CLKSTS.PLLFAST 或 CLKSTS.PLLSLOW 事件。

图 3-7：PLL 失锁检测时序图示例（$f_{PFD} > f_{DIV}$）

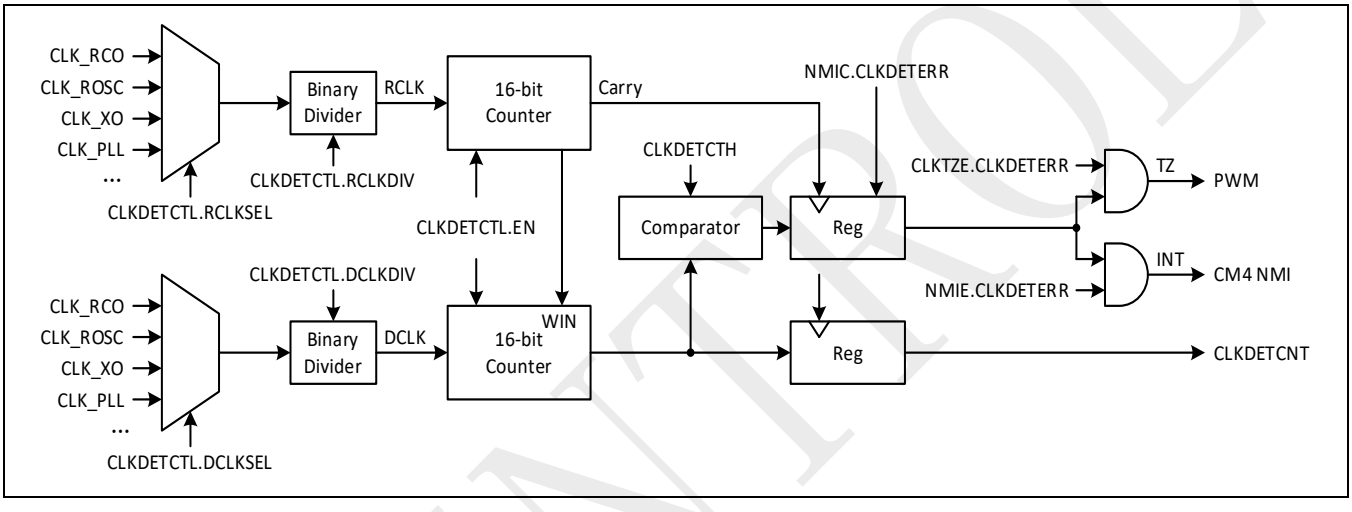


SPINTROL

3.8 时钟安全

为了时钟安全，SPC1169 提供了时钟交叉检测模块。如图 3-8 所示，参考时钟和待检测时钟可以独立选择，随后各自经过一个二进制分频器得到 RCLK 和 DCLK。RCLK 用于产生观测窗口，在该窗口内 DCLK 保持计数。待观测窗口关闭后，DCLK 的最终计数值被锁存到 CLKDETCNT，并与 CLKDETCTH 寄存器中的高低边界阈值进行比较，若计数越界则产生 NMRAWIF.CLKDETERR 标志位。计数值越界还可以根据 NMIE 和 CLKTZE 寄存器相应位的配置来触发 CPU 中断或者 PWM 信号封锁事件。DCLK 可以通过配置 GPIO2.MUXSEL=1 后，在 GPIO2 引脚观测。有关详细信息，请参考表 4-1。

图 3-8：时钟交叉检测



SPINTROL

©2025 旋智电子科技（上海）有限公司

3.9 寄存器

3.9.1 CLOCK 寄存器列表

表 3-3：CLOCK Module 基地址

外设模块	基地址
CLOCK	0x4000 0200

表 3-4：CLOCK Register Map

寄存器	偏移地址	描述	复位值
CLKSTS	0x00	时钟状态寄存器	0x00000003
RCOCTL*	0x04	RCO 控制寄存器	0x00000001
XOCTL*	0x08	晶振控制寄存器	0x00007F58
PLLCTL0*	0x0C	PLL 控制寄存器 0	0x00000270
PLLCTL1*	0x10	PLL 控制寄存器 1	0x2C4B0000
CLKDETCTL*	0x14	时钟检测控制寄存器	0x00000000
CLKDETCTH*	0x18	时钟检测阈值寄存器	0x00000000
CLKDETCNT	0x1C	时钟检测计数值寄存器	0x00000000
CLKTZE*	0x20	时钟状态寄存器	0x00000000
SYSCLKCTL*	0x24	系统时钟控制寄存器	0x00000000
DGCLKCTL*	0x28	消抖时钟控制寄存器	0x00000200
WDT0CLKCTL*	0x2C	WDT0 时钟控制寄存器	0x00000000
WDT1CLKCTL*	0x30	WDT1 时钟控制寄存器	0x01000000
PERICLKEN*	0x34	外设时钟使能寄存器	0x00000000
PERICLKSTS	0x38	外设时钟状态寄存器	0x0000F000
CLKREGKEY	0x3C	CLOCK 模块写使能寄存器	0x1ACCE551

注意： 由*标记的寄存器仅当 PWRREGKEY=0x1ACCE551 时才可以改写。

©2025 旋智电子科技（上海）有限公司

3.9.2 CLOCK 寄存器

表 3-5：时钟状态寄存器（CLKSTS）位段定义

CLKSTS (Clock Status Register) Offset：0x0 Default：0x00000003 Access：CLOCK->CLKSTS							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							CLKDETERR
7	6	5	4	3	2	1	0
PLLSLOW	PLLFAST	VCOFREQVLD	VCOFREQ	PLLRDY	XORDY	ROSCRDY	RCORDY

表 3-6：时钟状态寄存器（CLKSTS）位段描述

位段	位段名	属性	复位值	描述
31:9	RESERVED_31_9	RO	0x0	保留
8	CLKDETERR	RO	0x0	CLKDET 检测到时钟错误 0：事件未发生 1：事件发生
7	PLLSLOW	RO	0x0	PLL 失锁且频率偏慢标志，可能的原因包括： – f_{REF}/NIN*NFB/65536 高于 VCO 频率上限 – VCO 未能起振 – 环路分频器异常，导致分频器没有输出或者输出频率低于预期 0：事件未发生 1：事件发生
6	PLLFAST	RO	0x0	PLL 失锁且频率偏慢标志，可能的原因包括： – f_{REF}/NIN*NFB/65536 高于 VCO 频率上限 – 参考时钟丢失 – 环路分频器异常，导致分频器输出频率高于预期 0：事件未发生 1：事件发生
5	VCOFREQVLD	RO	0x0	VCOFREQ 位段读数有效标志 0：VCOFREQ 位段读数无效 1：VCOFREQ 位段读数有效
4	VCOFREQ	RO	0x0	VCO 频率标志位（用于校准）

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				0：VCO 频率偏慢 1：VCO 频率偏快
3	PLLRDY	RO	0x0	PLL 时钟就绪标志 0：未就绪 1：就绪
2	XORDY	RO	0x0	XO 时钟就绪标志 0：未就绪 1：就绪
1	ROSCRDY	RO	0x1	ROSC 时钟就绪标志 0：未就绪 1：就绪
0	RCORDY	RO	0x1	RCO 时钟就绪标志 0：未就绪 1：就绪

表 3-7：RCO 控制寄存器（RCOCTL）位段定义

RCOCTL (RCO Control Register) Offset：0x4 Default：0x00000001 Access：CLOCK->RCOCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						RESERVED	EN

表 3-8：RCO 控制寄存器（RCOCTL）位段描述

位段	位段名	属性	复位值	描述
31:2	RESERVED_31_2	RO	0x0	保留
1	RESERVED_1	RW	0x0	保留
0	EN	RW	0x1	RCO 使能 0：关闭 1：使能

SPINTROL

表 3-9：XO 控制寄存器（XOCTL）位段定义

XOCTL (Crystal Oscillator Control Register) Offset：0x8 Default：0x00007F58
Access：CLOCK->XOCTL

31	30	29	28	27	26	25	24
CURRENT	FEEDPIN	BYPASS	NINJLEN		FRCCODEEN	FRCCODE	
23	22	21	20	19	18	17	16
FRCCODE							
15	14	13	12	11	10	9	8
OPCODE				INITCODE			
7	6	5	4	3	2	1	0
ACTLPRD			ACTLRELAX	AMP			EN

表 3-10：XO 控制寄存器（XOCTL）位段描述

位段	位段名	属性	复位值	描述
31	CURRENT	RW	0x0	电流偏置档位 0：正常电流 1：两倍电流
30	FEEDPIN	RW	0x0	外部时钟引脚选择（此时不使用 XIO 及 XIN） 0：时钟由 XIN（GPIO3）引脚输入，XIO 保持悬空 1：时钟由 XIO（GPIO4）引脚输入，XIN 可用作其他
29	BYPASS	RW	0x0	外部时钟直通模式使能 0：内部晶振模式 1：外部时钟直通模式
28:27	RESERVED_28_27	RW	0x0	保留。
26	FRCCODEEN	RW	0x0	幅度强制控制使能 0：幅度通过校准电路调整偏置电流闭环控制 1：幅度由 FRCCODE 位段指定偏置电流直接控制
25:16	FRCCODE	RW	0x0	强制的偏置电流控制字
15:12	OPCODE	RW	0x7	预估的起振后偏置电流控制字高 4 位
11:8	INITCODE	RW	0xF	启动时偏置电流控制字高 4 位

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
7:5	ACTLPRD	RW	0x2	幅度控制电流调整更新周期 0：每 2048 个时钟周期调整一次 1：每 4096 个时钟周期调整一次 2：每 8192 个时钟周期调整一次 3：每 16384 个时钟周期调整一次 4：每 32768 个时钟周期调整一次 5：每 65536 个时钟周期调整一次 6：每 131072 个时钟周期调整一次 7：每 262144 个时钟周期调整一次
4	ACTLRELAX	RW	0x1	幅度控制阈值窗口 0：常规阈值窗口 1：放宽阈值窗口
3:1	AMP	RW	0x4	闭环控制锁定时的振荡幅度 0：0.32 V 1：0.48 V 2：0.64 V 3：0.77 V 4：0.92 V 5：1.06 V 6：1.19 V 7：1.33 V
0	EN	RW	0x0	晶振时钟使能 0：关闭 1：使能

表 3-11：PLL 控制寄存器 0（PLLCTL0）位段定义

PLLCTL0 (PLL Control Register 0)　Offset：0xC　Default：0x00000270
Access：CLOCK->PLLCTL0

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED					VCOBAND		ICP
7	6	5	4	3	2	1	0
ICP				RCLKSELXO	FCALWIN	FCALEN	EN

SPINTROL

表 3-12：PLL 控制寄存器（PLLCTL0）位段描述

位段	位段名	属性	复位值	描述
31:11	RESERVED_31_11	RO	0x0	保留
10:9	VCOBAND	RW	0x1	PLL VCO 频段选择 0：大约 350 ~ 450MHz 1：大约 450 ~ 550MHz 2：大约 550 ~ 650MHz 3：大约 600 ~ 700MHz
8:4	ICP	RW	0x7	PLL 电荷泵电流控制 本位段可以简单地设定为固定值 7。 若要得到更好的环路稳定性，可以设置为 0.96*NFB/(11-VCOBAND)，此处 NFB 是以小数形式表示的 PLL 环路反馈分频比。
3	RCLKSELXO	RW	0x0	PLL 参考时钟 0：CLK_RCO（RC 振荡器） 1：CLK_XO（晶振时钟）
2	FCALWIN	RW	0x0	PLL 频率校准时间窗。 在每个校准周期内，保持本位段为 1 直到 CLKSTS.VCOFREQVLD=1，然后再设定为 0。 0：时间窗关闭 1：时间窗开启
1	FCALEN	RW	0x0	PLL 频率校准使能 0：关闭 1：使能
0	EN	RW	0x0	PLL 使能 0：关闭 1：使能

表 3-13：PLL 控制寄存器 1（PLLCTL1）位段定义

PLLCTL1 (PLL Control Register 1)　Offset：0x10　Default：0x2C4B0000
Access：CLOCK->PLLCTL1

31	30	29	28	27	26	25	24
RESERVED	NOUT				NIN		
23	22	21	20	19	18	17	16
NFB							
15	14	13	12	11	10	9	8
NFB							
7	6	5	4	3	2	1	0
NFB							

©2025 旋智电子科技（上海）有限公司

表 3-14：PLL 控制寄存器 1（PLLCTL1）位段描述

位段	位段名	属性	复位值	描述
31	RESERVED_31	RO	0x0	保留
30:27	NOUT	RW	0x5	设置 VCO 到 PLL 输出的分频比（f_{VCO}/f_{PLL}）为 NOUT+1
26:24	NIN	RW	0x4	设置输入参考时钟的分频比（f_{REF}/f_{PFD}）为 NIN 1：若 f_{REF} 在[4MHz, 8MHz]区间，则设置为 1 2：若 f_{REF} 在(8MHz, 16MHz]区间，则设置为 2 3：若 f_{REF} 在(16MHz, 24MHz]区间，则设置为 3 4：若 f_{REF} 在(24MHz, 32MHz]区间，则设置为 4 5：若 f_{REF} 在(32MHz, 40MHz]区间，则设置为 5 6：若 f_{REF} 在(40MHz, 48MHz]区间，则设置为 6 7：若 f_{REF} 在(48MHz, 56MHz]区间，则设置为 7
23:0	NFB	RW	0x4B0000	设置环路反馈的分频比（f_{VCO}/f_{PFD}）为 NFB/65536

表 3-15：时钟检测控制寄存器（CLKDETCTL）位段定义

CLKDETCTL (Clock Detection Control Register)　Offset：0x14　Default：0x00000000
Access：CLOCK->CLKDETCTL

31	30	29	28	27	26	25	24
EN	RESERVED				DCLKDIV		
23	22	21	20	19	18	17	16
RESERVED					DCLKSEL		
15	14	13	12	11	10	9	8
RESERVED					RCLKDIV		
7	6	5	4	3	2	1	0
RESERVED					RCLKSEL		

表 3-16：时钟检测控制寄存器（CLKDETCTL）位段描述

位段	位段名	属性	复位值	描述
31	EN	RW	0x0	时钟检测使能 关闭时钟检测时，之前的错误事件自动清零 0：关闭 1：使能
30:27	RESERVED_30_27	RO	0x0	保留
26:24	DCLKDIV	RW	0x0	待检测时钟的分频比 0：1 分频（不分频） 1：2 分频 2：4 分频

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				3：8 分频 4：16 分频 5：32 分频 6：64 分频 7：128 分频
23:19	RESERVED_23_19	RO	0x0	保留
18:16	DCLKSEL	RW	0x0	待检测时钟的选择 000：CLK_RCO 001：CLK_ROSC 010：CLK_XO 011：CLK_PLL 100:CLK_LPROSC 101:在诊断模式下通过 wPower 的 TZ0 脚的高压时钟 110：在诊断模式下通过 wPower 的 TZ1 脚的高压时钟 111:AHB 时钟，也即 CPU 时钟
15:11	RESERVED_15_11	RO	0x0	保留
10:8	RCLKDIV	RW	0x0	待检测时钟的分频比 0：1 分频（不分频） 1：2 分频 2：4 分频 3：8 分频 4：16 分频 5：32 分频 6：64 分频 7：128 分频
7:3	RESERVED_7_3	RO	0x0	保留
2:0	RCLKSEL	RW	0x0	参考时钟的选择 000：CLK_RCO 001：CLK_ROSC 010：CLK_XO 011：CLK_PLL 100:CLK_LPROSC 101:在诊断模式下通过 wPower 的 TZ0 脚的高压时钟 110：在诊断模式下通过 wPower 的 TZ1 脚的高压时钟 111:AHB 时钟，也即 CPU 时钟

©2025 旋智电子科技（上海）有限公司

表 3-17：时钟检测阈值寄存器（CLKDETCTH）位段定义

CLKDETCTH (Clock Detection Counter Threshold Register) Offset：0x18 Default：0x00000000 Access：CLOCK->CLKDETCTH							
31	30	29	28	27	26	25	24
HI							
23	22	21	20	19	18	17	16
HI							
15	14	13	12	11	10	9	8
LO							
7	6	5	4	3	2	1	0
LO							

表 3-18：时钟检测阈值寄存器（CLKDETCTH）位段描述

位段	位段名	属性	复位值	描述
31:16	HI	RW	0x0	时钟检测计数值上边界 计数值高于上边界时，可以触发 PWM 封锁和 CPU 不可屏蔽的中断
15:0	LO	RW	0x0	时钟检测计数值下边界 计数值低于下边界时，可以触发 PWM 封锁和 CPU 不可屏蔽的中断

表 3-19：时钟检测计数值寄存器（CLKDETCNT）位段定义

CLKDETCNT (Clock Detection Counter Register) Offset：0x1C Default：0x00000000 Access：CLOCK->CLKDETCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 3-20：时钟检测计数值寄存器（CLKDETCNT）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RO	0x0	时钟检测最终计数值

©2025 旋智电子科技（上海）有限公司

表 3-21：时钟封锁事件使能寄存器（CLKTZE）位段定义

CLKTZE (Clock Trip-zone Event Enable Register) Offset：0x20 Default：0x00000000 Access：CLOCK->CLKTZE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						CLKDETERR	PLLUNLOCK

表 3-22：时钟封锁事件使能寄存器（CLKTZE）位段描述

位段	位段名	属性	复位值	描述
31:2	RESERVED_31_2	RO	0x0	保留
1	CLKDETERR	RW	0x0	CLKDET 错误封锁事件使能 0：CLKDET 错误不触发 PWM 封锁 1：CLKDET 错误触发 PWM 封锁
0	PLLUNLOCK	RW	0x0	PLL 失锁封锁事件使能 0：PLL 失锁不触发 PWM 封锁 1：PLL 失锁触发 PWM 封锁

表 3-23：系统时钟 0 控制寄存器（SYSCLK0CTL）位段定义

SYSCLKCTL (System Clock Control Register) Offset：0x24 Default：0x00000000 Access：CLOCK->SYSCLKCTL							
31	30	29	28	27	26	25	24
RESERVED						SRC	
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
DIV							

表 3-24：系统时钟 0 控制寄存器（SYSCLK0CTL）位段定义

位段	位段名	属性	复位值	描述
31:26	RESERVED_31_26	RO	0x0	保留

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
25:24	SRC	RW	0x0	时钟源选择 0：CLK_RCO 1：CLK_ROSC 2：CLK_XO 3：CLK_PLL
23:8	RESERVED_23_8	RO	0x0	保留
7:0	DIV	RW	0x0	设置从时钟源的分频比为 DIV+1

表 3-25：消抖时钟控制寄存器（DGCLKCTL）位段定义

DGCLKCTL (Deglitch Clock Control Register) Offset：0x28 Default：0x00000200 Access：CLOCK->DGCLKCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED					PDIV		
7	6	5	4	3	2	1	0
DIV							

表 3-26：消抖时钟控制寄存器（DGCLKCTL）位段描述

位段	位段名	属性	复位值	描述
31:11	RESERVED_31_11	RO	0x0	保留
10:8	PDIV	RW	0x2	设置时钟源（由 SYSCLK0CTL.SRC 指定）的预分频系数 000:1 分频 001:2 分频 010:4 分频 011:8 分频 100:16 分频 101:32 分频 110:64 分频 111:128 分频
7:0	DIV	RW	0x0	设置总的分频比为 2^{PDIV} x (DIV+1)

SPINTROL

表 3-27：WDT0 时钟控制寄存器（WDT0CLKCTL）位段定义

WDT0CLKCTL (WDT0 Clock Control Register) Offset：0x2C Default：0x00000000 Access：CLOCK->WDT0CLKCTL							
31	30	29	28	27	26	25	24
RESERVED						SRC	
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
DIV							

表 3-28：WDT0 时钟控制寄存器（WDT0CLKCTL）位段描述

位段	位段名	属性	复位值	描述
31:26	RESERVED_31_26	RO	0x0	保留
25:24	SRC	RW	0x0	时钟源选择 0：CLK_RCO 1：CLK_ROSC 2：CLK_XO 3：CLK_PLL
23:8	RESERVED_23_8	RO	0x0	保留
7:0	DIV	RW	0x0	设置从时钟源（系统时钟 1）的分频比为 (DIV+1)

表 3-29：WDT1 时钟控制寄存器（WDT1CLKCTL）位段定义

WDT1CLKCTL (WDT1 Clock Control Register) Offset：0x30 Default：0x01000000 Access：CLOCK->WDT1CLKCTL							
31	30	29	28	27	26	25	24
RESERVED						SRC	
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
DIV							

©2025 旋智电子科技（上海）有限公司

表 3-30：WDT1 时钟控制寄存器（WDT1CLKCTL）位段描述

位段	位段名	属性	复位值	描述
31:26	RESERVED_31_26	RO	0x0	保留
25:24	SRC	RW	0x1	时钟源选择 0：CLK_RCO 1：CLK_ROSC 2：CLK_XO 3：CLK_PLL
23:8	RESERVED_23_8	RO	0x0	保留
7:0	DIV	RW	0x0	设置从时钟源（系统时钟 1）的分频比为 (DIV+1)

表 3-31：外设时钟使能寄存器（PERICLKEN）位段定义

PERICLKEN (Peripheral Clock Enable Register)　Offset：0x34　Default：0x00000000
Access：CLOCK->PERICLKEN

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED				CANCLK	I2CCLK	SPI1CLK	SPI0CLK
7	6	5	4	3	2	1	0
UART1CLK	UART0CLK	TIMER2CLK	TIMER1CLK	TIMER0CLK	ECAPCLK	PWMCLK	ADCCLK

表 3-32：外设时钟使能寄存器（PERICLKEN）位段描述

位段	位段名	属性	复位值	描述
31:12	RESERVED_31_12	RO	0x0	保留
11	CANCLK	RW	0x0	CAN 模块时钟使能 0：关闭 1：使能
10	I2CCLK	RW	0x0	I2C 模块时钟使能 0：关闭 1：使能
9	SPI1CLK	RW	0x0	SPI1 模块时钟使能 0：关闭 1：使能
8	SPI0CLK	RW	0x0	SPI0 模块时钟使能 0：关闭 1：使能

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
7	UART1CLK	RW	0x0	UART1 模块时钟使能 0：关闭 1：使能
6	UART0CLK	RW	0x0	UART0 模块时钟使能 0：关闭 1：使能
5	TIMER2CLK	RW	0x0	TIMER2 模块时钟使能 0：关闭 1：使能
4	TIMER1CLK	RW	0x0	TIMER1 模块时钟使能 0：关闭 1：使能
3	TIMER0CLK	RW	0x0	TIMER0 模块时钟使能 0：关闭 1：使能
2	ECAPCLK	RW	0x0	ECAP 模块时钟使能 0：关闭 1：使能
1	PWMCLK	RW	0x0	PWM 模块时钟使能 0：关闭 1：使能
0	ADCCLK	RW	0x0	ADC 模块时钟使能 0：关闭 1：使能

表 3-33：外设时钟状态寄存器（PERICLKSTS）位段定义

PERICLKSTS (Peripheral Clock Status Register) Offset：0x38 Default：0x0000F000 Access：CLOCK->PERICLKSTS							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 WDT1	14 WDT0	13 DG	12 AHB	11 CAN	10 I2C	9 SPI1	8 SPI0
7 UART1	6 UART0	5 TIMER2	4 TIMER1	3 TIMER0	2 ECAP	1 PWM	0 ADC

©2025 旋智电子科技（上海）有限公司

表 3-34：外设时钟状态寄存器（PERICLKSTS）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	WDT1	RO	0x1	WDT1 时钟状态 0：关闭 1：使能
14	WDT0	RO	0x1	WDT0 时钟状态 0：关闭 1：使能
13	DG	RO	0x1	消抖时钟状态 0：关闭 1：使能
12	AHB	RO	0x1	总线和 CPU 时钟状态 0：关闭 1：使能
11	CAN	RO	0x0	CAN 时钟状态 0：关闭 1：使能
10	I2C	RO	0x0	I2C 时钟状态 0：关闭 1：使能
9	SPI1	RO	0x0	SPI1 时钟状态 0：关闭 1：使能
8	SPI0	RO	0x0	SPI0 时钟状态 0：关闭 1：使能
7	UART1	RO	0x0	UART1 时钟状态 0：关闭 1：使能
6	UART0	RO	0x0	UART0 时钟状态 0：关闭 1：使能
5	TIMER2	RO	0x0	TIMER2 时钟状态 0：关闭 1：使能

SPINTROL

位段	位段名	属性	复位值	描述
4	TIMER1	RO	0x0	TIMER1 时钟状态 0：关闭 1：使能
3	TIMER0	RO	0x0	TIMER0 时钟状态 0：关闭 1：使能
2	ECAP	RO	0x0	ECAP 时钟状态 0：关闭 1：使能
1	PWM	RO	0x0	PWM 时钟状态 0：关闭 1：使能
0	ADC	RO	0x0	ADC 时钟状态 0：关闭 1：使能

表 3-35：CLOCK 模块写使能寄存器（CLKREGKEY）位段定义

CLKREGKEY (Clock Register Write-Allow Key Register) Offset：0x3C Default：0x1ACCE551 Access：CLOCK->CLKREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

表 3-36：CLOCK 模块写使能寄存器（CLKREGKEY）位段描述

位段	位段名	属性	复位值	描述
31:0	KEY	RW	0x1ACCE551	写入 0x1ACCE551 解锁受保护的 CLOCK 寄存器

©2025 旋智电子科技（上海）有限公司

4 通用输入输出（GPIO）

4.1 概述

在 SPC1169 中，I/O 引脚可以分为以下几类：

- 3.3V I/O，与模拟输入功能复用
  - GPIO0~GPIO4、GPIO28~GPIO31：可用作模拟输入通道 0~8；
  - GPIO36~GPIO39：内部引脚，应用于 U 相、V 相、W 相和公共节点的 BEMF 电压感测默认功能；
  - GPIO40：内部引脚，其默认功能为测量分压后的 VBAT 电压；
- 3.3V I/O，兼容 5V 输入
  - GPIO25、GPIO26：内部引脚，与 LIN 收发器相连（GPIO25 应用于 UART1 的 TXD，GPIO26 应用于 RXD）；
  - GPIO27：内部引脚，与 MON 模块相连；

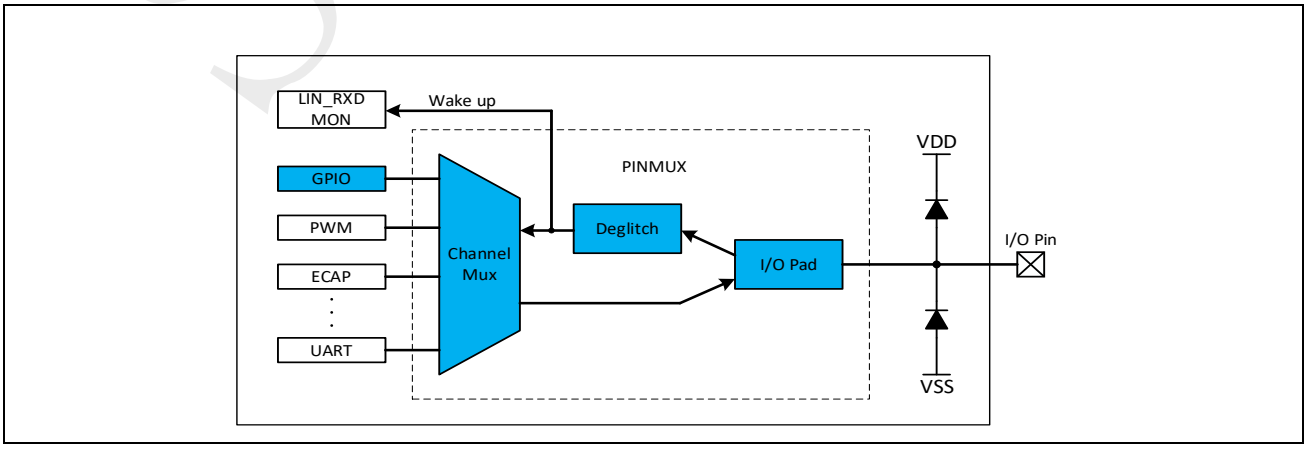
  其中，GPIO26、GPIO27 不经过 PINMUX；
- 5V I/O
  - GPIO5~GPIO18、GPIO32~GPIO35：与各种功能复用；
  - GPIO19~GPIO24：中断引脚，与预驱动器模块相连，应配置为 PWM 输出；

本章介绍了如何对这些 I/O 引脚进行多路复用，使它们可以配置为通用 I/O 或任意一种备用功能之一。如图 4-1 所示，I/O 控制由两部分组成：

- PINMUX 模块
  - I/O Pad：上拉/下拉控制、输入/输出控制和输出驱动强度控制；
  - Deglitch：输入去抖控制；
  - MUX：引脚多路复用；
- GPIO 模块
  - GPIO 功能 IP 控制；

每个 I/O 引脚可以独立地进行编程和使用。如图 4-2 所示，从 I/O 到 VDD 有一个二极管，从 VSS 到 I/O 也有一个二极管，用于 ESD 保护。

图 4-1：I/O 控制框图



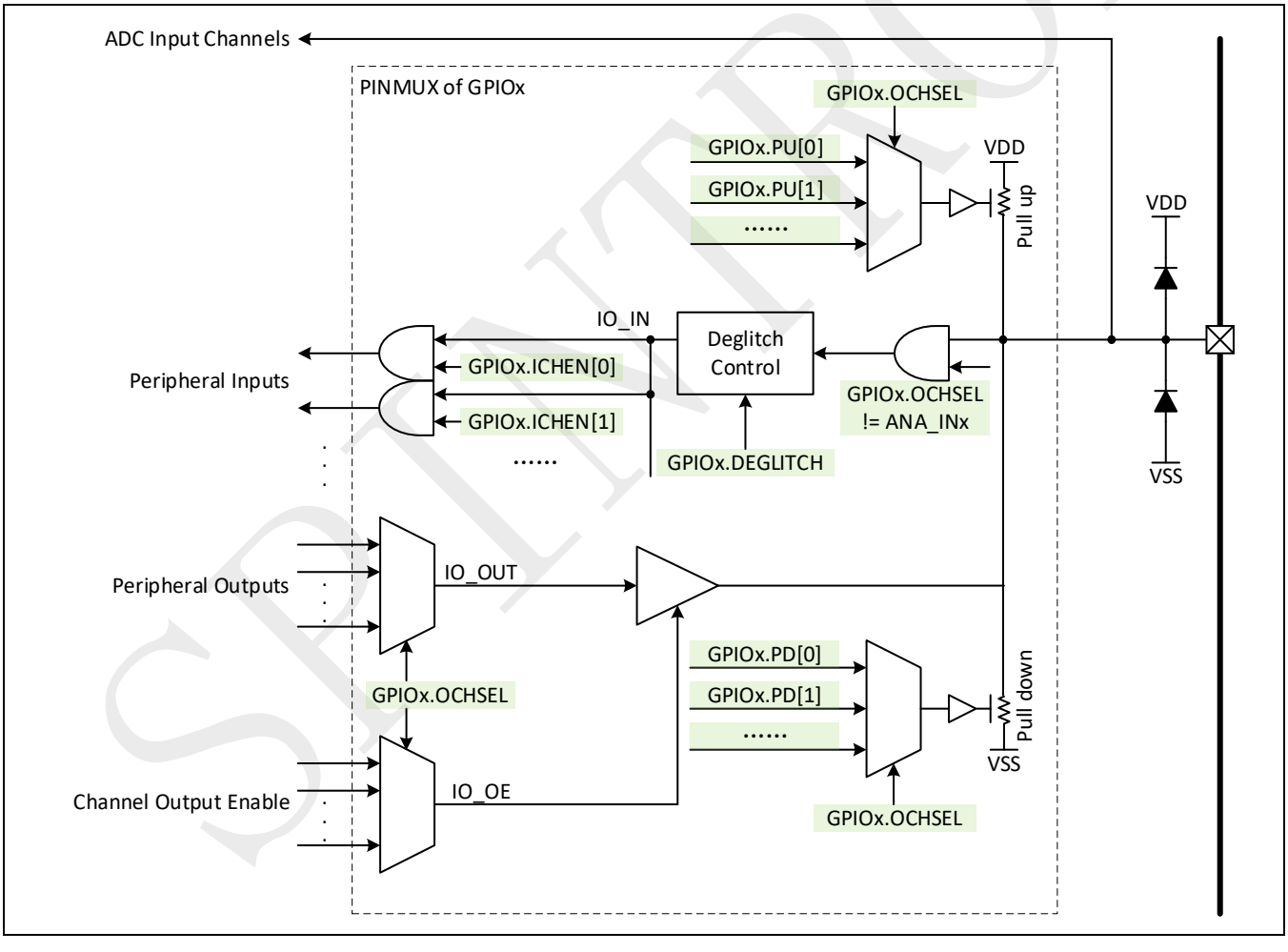
4.2 引脚复用（PINMUX）

PINMUX 模块如图 4-2 所示，每个 I/O 引脚都使用多路复用器来支持最多 4 个功能通道。功能通道可以是输入或输出。如果将 I/O 引脚用作输入，则输出缓冲将被禁用并设置为高阻态。只有当 I/O 用作模拟输入时，输入缓冲才会被禁用，以避免潜在的电流泄漏。对于所有其他情况，输入缓冲始终处于启用状态。输入信号也可以通过滤波器传递，以抑制可能的毛刺。

输出功能和上拉/下拉控制的多路复用是互斥的，并由相应的 GPIOx.OCHSEL 寄存器字段选择。例如，如果 GPIO4.OCHSEL=3，则 GPIO4 引脚的上拉和下拉控制由 GPIO4.PU[3]和 GPIO4.PD[3]确定，并且输出功能设置为通道 3。

每个通道的输入功能的多路复用是非互斥的，每个通道都有自己的使能控制，通过 GPIOx.ICHEN[y]来确定是否独立使用 GPIOx 输入作为通道 y 的功能输入。

图 4-2：PINMUX 框图



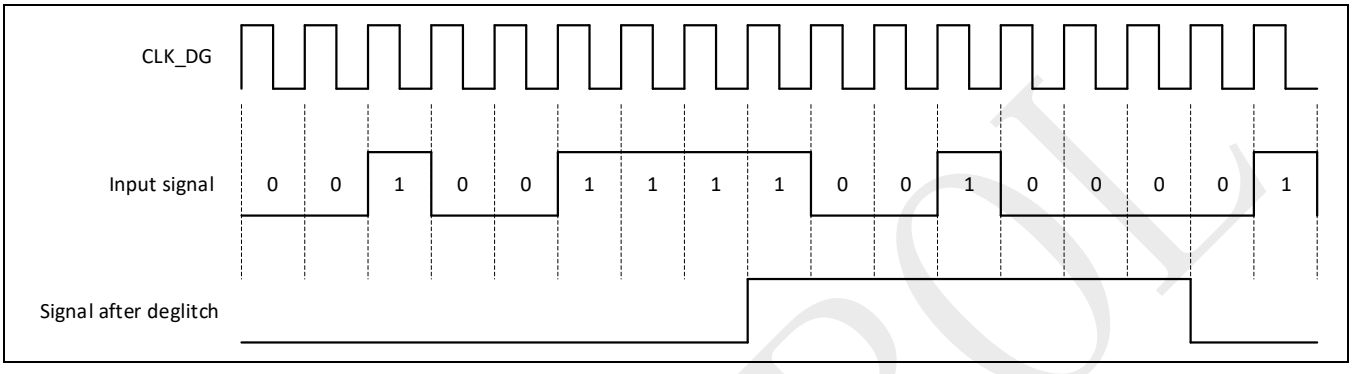
注意： 当 GPIO3，GPIO4 工作在 XIN 与 XOUT 时，外部晶振和内部时钟电路直连，不存在图中的输入输出关系。

©2025 旋智电子科技（上海）有限公司

4.2.1 消抖控制（Deglitch）

通过设置寄存器 GPIOx.DEGLITCH 可以启用输入消抖功能，图 4-3 显示了输入去抖动的原理。当输入信号切换到新值并保持连续 3 个 CLK_DG 周期时，输出将更新为新值，否则输出将保持原始值。

图 4-3：输入消抖



4.2.2 I/O 可选功能

I/O 引脚各个通道对应的功能如表 4-1 所示。

表 4-1：I/O 引脚功能通道定义

引脚名	I/O 类型	通道（红色部分为默认通道）			
		0	1	2	3
PIN_GPIO0	3.3V IO	GPIO0	ANA_IN0	PWMSOCAO	PWMSOCCO
PIN_GPIO1	3.3V IO	GPIO1	ANA_IN1	PWMSOCBO	ECAP_APWMO
PIN_GPIO2	3.3V IO	GPIO2	ANA_IN2	CLK_DCLK	-
PIN_GPIO3	3.3V IO	GPIO3	ANA_IN3	XIN	COMP_MON0
PIN_GPIO4	3.3V IO	GPIO4	ANA_IN4	XIO	COMP_MON1
PIN_GPIO5	5V IO	GPIO5	-	PWMSOCO	ECAP_APWMO
PIN_GPIO6	5V IO	GPIO6	SPI0_SCLK	UART1_TXD	COMP_MON2
PIN_GPIO7	5V IO	GPIO7	SPI0_SFRM	UART1_RXD	COMP_MON3
PIN_GPIO8	5V IO	GPIO8	SPI0_MOSI	CAN_TXD	COMP_MON4
PIN_GPIO9	5V IO	GPIO9	SPI0_MISO	CAN_RXD	COMP_MON5
PIN_GPIO10	5V IO	GPIO10	UART0_TXD	PWM0A	I2C_SCL
PIN_GPIO11	5V IO	GPIO11	UART0_RXD	PWM0B	I2C_SDA
PIN_GPIO12	5V IO	GPIO12	SPI1_SCLK	PWM1A	PWM3A
PIN_GPIO13	5V IO	GPIO13	SPI1_SFRM	PWM1B	PWM3B
PIN_GPIO14	5V IO	GPIO14	SPI1_MOSI	PWM2A	I2C_SCL
PIN_GPIO15	5V IO	GPIO15	SPI1_MISO	PWM2B	I2C_SDA
PIN_GPIO16	5V IO	GPIO16	UART1_TXD	CAN_TXD	SPI1_SFRM
PIN_GPIO17	5V IO	GPIO17	UART1_RXD	CAN_RXD	SPI1_SCLK

©2025 旋智电子科技（上海）有限公司

引脚名	I/O 类型	通道（红色部分为默认通道）			
		0	1	2	3
PIN_GPIO18	5V IO	GPIO18	PWMSOCO	PWMSYNCO	ECAP_APWMO
PIN_GPIO19	5V IO	GPIO19	PWM2B	PWM0A	ECAP_APWMO
PIN_GPIO20	5V IO	GPIO20	PWM2A	PWM1A	ECAP_APWMO
PIN_GPIO21	5V IO	GPIO21	PWM1B	PWM2A	ECAP_APWMO
PIN_GPIO22	5V IO	GPIO22	PWM1A	PWM0B	ECAP_APWMO
PIN_GPIO23	5V IO	GPIO23	PWM0B	PWM1B	ECAP_APWMO
PIN_GPIO24	5V IO	GPIO24	PWM0A	PWM2B	ECAP_APWMO
PIN_GPIO25	3.3V/5V IO	GPIO25	UART1_TXD	CAN_TXD	I2C_SCL
PIN_GPIO26	3.3V/5V IO	GPIO26	UART1_RXD	CAN_RXD	I2C_SDA
PIN_GPIO27	3.3V/5V IO	GPIO27	-	-	-
PIN_GPIO28	3.3V IO	GPIO28	ANA_IN5	SPI1_SCLK	PWMSOCAO
PIN_GPIO29	3.3V IO	GPIO29	ANA_IN6	SPI1_SFRM	PWMSOCBO
PIN_GPIO30	3.3V IO	GPIO30	ANA_IN7	SPI1_MOSI	PWMSOCCO
PIN_GPIO31	3.3V IO	GPIO31	ANA_IN8	SPI1_MISO	PWMSYNCO
PIN_GPIO32	5V IO	GPIO32	UART1_TXD	I2C_SCL	COMP_MON6
PIN_GPIO33	5V IO	GPIO33	UART1_RXD	I2C_SDA	COMP_MON7
PIN_GPIO34	5V IO	GPIO34	CAN_TXD	I2C_SCL	-
PIN_GPIO35	5V IO	GPIO35	CAN_RXD	I2C_SDA	-
PIN_GPIO36	3.3V IO	GPIO36	ANA_IN9(PHU)	SPI1_SCLK	UART1_TXD
PIN_GPIO37	3.3V IO	GPIO37	ANA_IN10(PHV)	SPI1_SFRM	UART1_RXD
PIN_GPIO38	3.3V IO	GPIO38	ANA_IN11(PHW)	SPI1_MOSI	CAN_TXD
PIN_GPIO39	3.3V IO	GPIO39	ANA_IN12(PHC)	SPI1_MISO	CAN_RXD
PIN_GPIO40	3.3V IO	GPIO40	ANA_ IN13(VBAT)	-	-

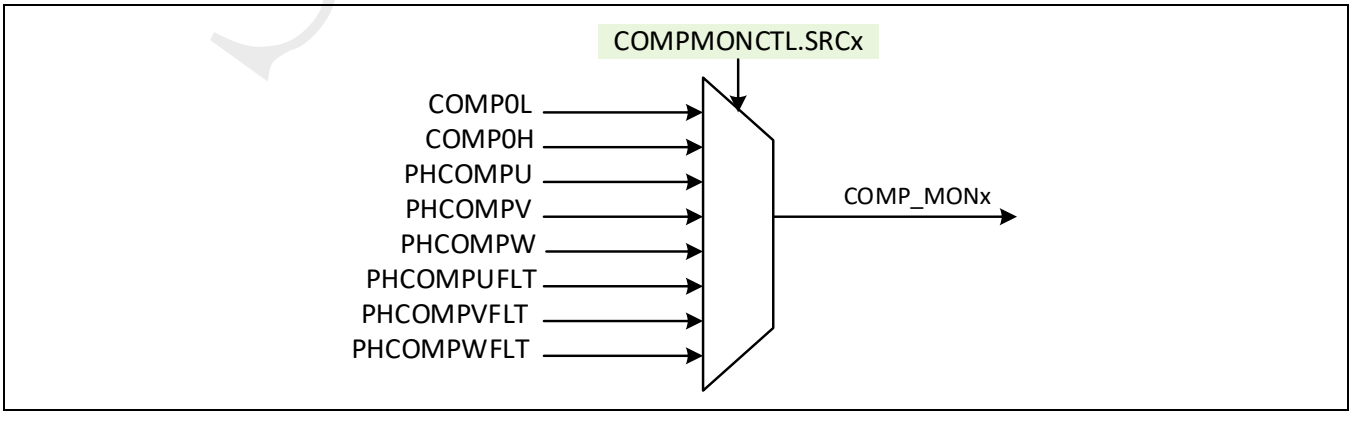
[1] 所有的 GPIO 引脚可以通过通道 0（GPIO 功能通道）配置为 ECAP 输入。

[2] 在 SPC1169 中，GPIO19~GPIO27 和 GPIO36~GPIO40 如图 1-1 所示，没有连接到外部引脚，保留用于其它用途。

[3] 注意：GPIO5 为 BOOT 引脚，在芯片启动阶段不可被配置为其他功能，否则将导致 GPIO5 出现恒定电平，影响 ISP 模式或 Flash 启动模式的进入。

COMP_MONx 的通道功能总结如图 4-4 所示。

图 4-4：COMP_MONx 源



©2025 旋智电子科技（上海）有限公司

4.2.3 环回路径（Loopback）

为了在 I/O 端口提供回环测试功能，针对具有通信数据功能的端口（包括 SPIx_MISO、SPIx_MOSI、UARTx_RXD、UARTx_TXD、CAN_RXD、CAN_TXD），提供了回环选项，通过相应 GPIOx.LPBK 寄存器位段控制。使能该路径后，由 GPIOx.OCHSEL 选择的输出通道不仅会输出到 I/O，还会环回到输入，可以由 GPIOx.ICHEN[y]使能用作通道 y 的输入。

例如，如果 PINMUX->GPIO10 的 ICHEN[1]和 LPBK 位均设置为 1，则从 UART0_TXD 输出到 GPIO10 的信号将回环到 UART0_RXD。

4.2.4 调试模式

当 TRSTn 引脚被拉低时，GPIO15~GPIO18 引脚将作为由软件控制的功能使用，当 TRSTn 引脚被拉高时，它们将用于调试接口功能，具体定义请参考表 4-2。

表 4-2：调试接口引脚功能

引脚	调试接口信号	
	SYSTEM->DBGIFCTL=0 （SWD 模式）	SYSTEM->DBGIFCTL=1 （JTAG 模式）
GPIO15	软件控制功能通道	TDI
GPIO16	SWV [1]	TDO
GPIO17	SWD	TMS
GPIO18	SWCK	TCK

[1] 用户可以通过代码"CoreDebug->DEMCR &= ~(1U << 24);"在 SWD 模式下禁用 SWV 输出功能，从而可以将 GPIO16 用作正常的功能通道，该代码将清除寄存器 0xE000EDFC 中位 24 的值。

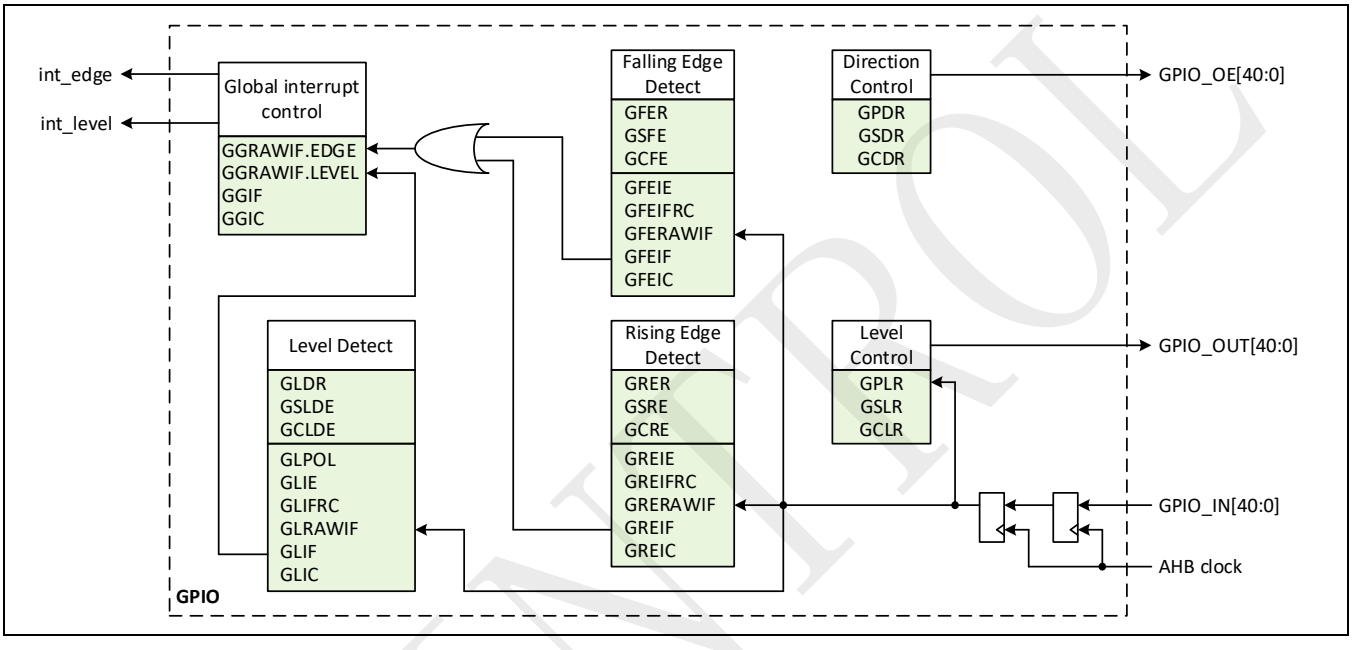
SPINTROL

4.3 GPIO 控制模块

如图 4-5 所示，GPIO 控制模块包括电平控制、方向控制、边沿检测和电平检测。

寄存器中的每一位对应一个 GPIO 引脚，为了方便使用，提供了按位控制寄存器，用户可以通过向“GS-”寄存器位写入 1 来设置控制位，或者通过向“GC-”寄存器位写入 1 来清除控制位。向这些按位控制寄存器写入 0 不产生任何效果。

图 4-5：GPIO 框图



4.3.1 方向和电平控制

默认情况下，所有 GPIO 引脚均被设置为输入模式。通过 GPDR[x]寄存器，可以配置 GPIO 引脚的方向。用户可以将 GSDR 位写为 1，将这些引脚设置为输出模式，或将 GCDR 位写为 1，将这些引脚设置为输入模式，GPDR 寄存器将相应地更新。

当配置为输出引脚时，可以通过 GPLR[x]或 GSLR[x]/GCLR[x]寄存器设置输出电平。无论向 GPLR[x]或 GPDR[x]寄存器写入了什么值，从 GPLR[x]寄存器读取的值始终反映 GPIO 端口的状态，即 I/O 引脚的电平。

请注意，为了避免逻辑不稳定性，应用了跨域输入重新采样。因此，如果通过写入 GPLR[x]寄存器更改输出引脚的电平，从 GPLR[x]寄存器的读取将在 2 个 AHB 时钟周期后才更新为已写入的值。因此，如果想通过软件在输出引脚上生成波形，建议使用 GSLR[x]寄存器位写为高电平，将 GCLR[x]寄存器位写为低电平的方式进行操作。若直接使用读-修改-写 GPLR[x]寄存器的编码方式，在引脚上将观察不到预期的工作情形，除非在写操作之后插入手动延迟（例如 NOP 指令）。

©2025 旋智电子科技（上海）有限公司

4.3.2 边沿触发中断

如图 4-6 所示，使用 AHB 时钟可以在重新采样的引脚输入上检测到上升沿事件和下降沿事件。当设置 GRER[x]或 GFER[x]时，启用此功能。GSRE[x]/GSFE[x]和 GCRE[x]/GCFE[x]是位控制寄存器，也可用于设置或清除 GREF[x]/GFER[x]寄存器位。

每当在重新采样的引脚输入上检测到上升沿事件或下降沿事件时，它将始终被锁存到 GRERAWIF[x]或 GFERAWIF[x]寄存器中的相应位。但是，除非通过设置 GREIE[x]或 GFEIE[x]中的相应位启用，否则中断不会触发和锁存为 GREIF[x]或 GFEIF[x]中断标志。将 GREIC[x]或 GFEIC[x]位写为 1 将清除 GRERAWIF[x]/GFERAWIF[x]中的锁定事件标志和 GREIF[x]或 GFEIF[x]中的中断标志。

为了便于调试，还可以通过将 1 写入 GREIFRC[x]和 GFEIFRC[x]寄存器位来软件强制上升沿和下降沿事件。

GGRAWIF 寄存器中的 REDGE 位是全局上升沿事件标志，它是所有 GRERAWIF 位的逻辑或。类似地，GGRAWIF 寄存器中的 FEDGE 位是全局下降沿事件标志，它是所有 GFERAWIF 位的逻辑或。REDGE 和 FEDGE 被 OR 运算以生成 GGRAWIF 寄存器中的 EDGE 位。因此，无论在任何引脚输入上检测到上升沿，GGRAWIF 的 REDGE 位和 EDGE 位都会被锁存为 1。无论在任何引脚输入上检测到下降沿，GGRAWIF 的 FEDGE 位和 EDGE 位都会被锁存为 1。

图 4-6：GPIO 边沿检测

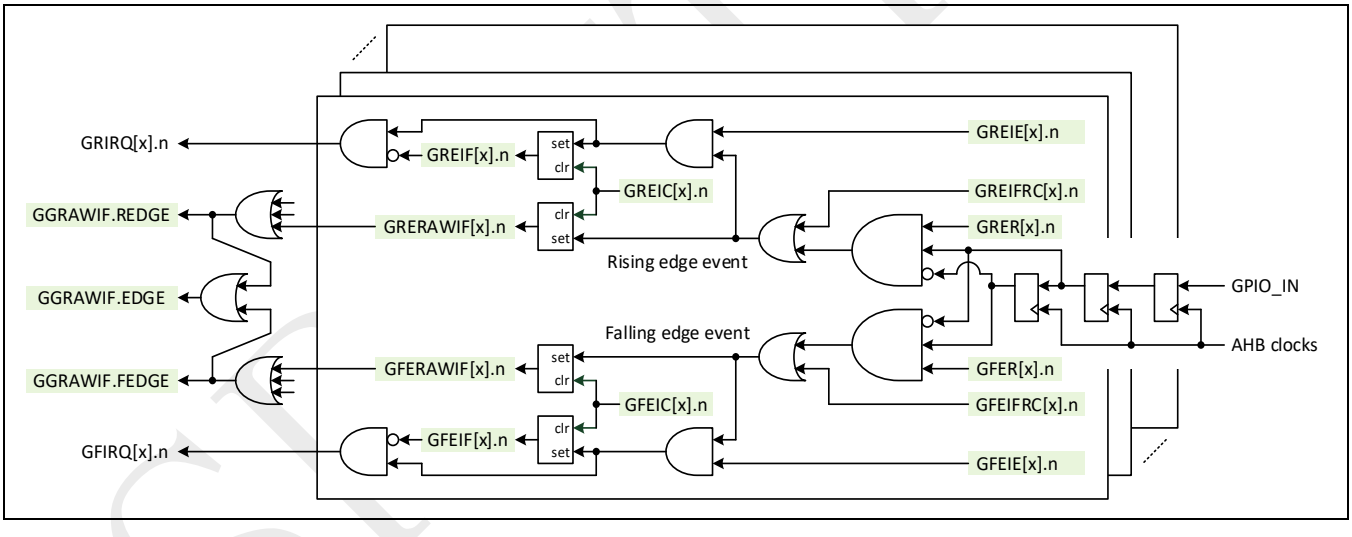
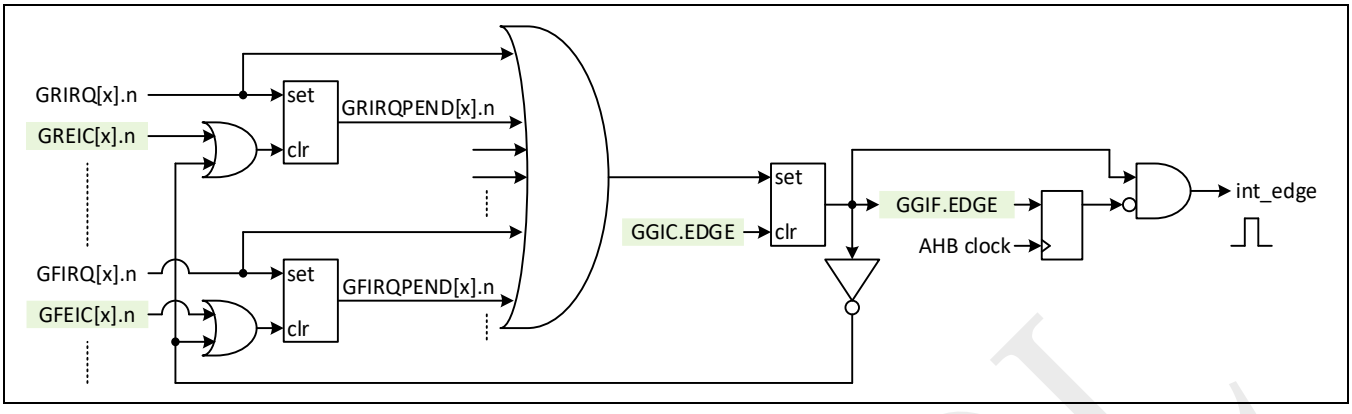


图 4-6 中命名为 GRIRQ[x].n 和 GFIRQ[x].n 的信号是有效的上升沿触发和下降沿触发中断请求，它们能触发上升沿和下降沿中断事件，但需要注意的是，若 GREIF[x]或 GFEIF[x]寄存器中的值为 1 时，将无法触发 GRIRQ[x].n 和 GFIRQ[x].n 的中断请求，所以中断处理完成之后，请记住清 GREIF[x]或 GFEIF[x]寄存器。

如图 4-7 所示，所有有效的中断请求被 OR 在一起生成最终的边沿触发中断，并锁存到全局中断标志寄存器 GGIF 的 EDGE 位，可以通过向 GGIC 寄存器的 EDGE 位写入 1 来清除锁定的标志位。

©2025 旋智电子科技（上海）有限公司

图 4-7：GPIO 边沿触发中断



假如 PIN x 先触发边沿中断请求，中断标志寄存器 GGIF.EDGE 置 1，并触发中断，随后 PIN y、PIN z 也同时触发边沿中断请求，此时 PIN y、PIN z 的边沿中断请求都将会被挂起。

若 PIN x 中断服务程序执行完毕，并往中断清除寄存器 GGIC.EDGE 写 1，此时 GGIF.EDGE 标志位被清 0。随后在时钟的下一拍，将会发生两个动作：

1. PIN y、PIN z 挂起的请求再次将中断标志寄存器 GGIF.EDGE 标志位置 1，并触发中断；
2. PIN x 中断清除动作（往中断清除寄存器 GGIC.EDGE 写 1），将经由反相器清除 PIN y、PIN z 挂起的中断请求。

总结：当一个中断被响应时，不管发生多少个其他中断，后续能被响应的中断只有一个。

如上所述，当同时有两个中断请求时，只有一个中断请求会被响应，此时若想处理所有请求，建议按照以下步骤实现 ISR 程序：

- 首先读取 GREIF 或 GFEIF 寄存器来确定要处理的请求；
- 根据请求进行处理；
- 根据处理的请求清除 GREIF 或 GFEIF 寄存器；
- 清除 GGIF.EDGE 位（否则，GPIO 边沿触发中断生成将被阻塞）。

©2025 旋智电子科技（上海）有限公司

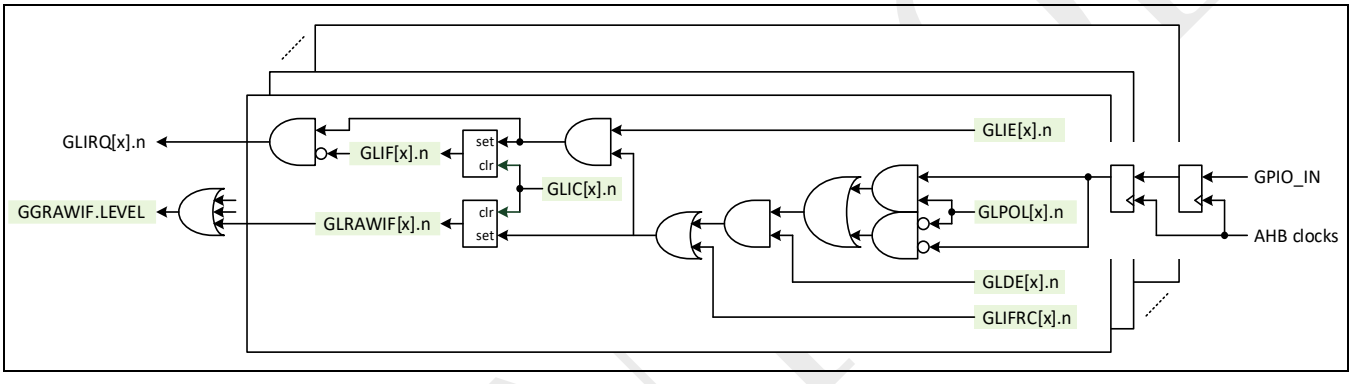
4.3.3 电平触发中断

如图 4-8 所示，在用 AHB 时钟对于输入重新采样后，可以进一步检测信号的电平事件。该特性通过 GLDE[x]寄存器使能。GSLDE[x]和 GCLDE[x]是按位控制的寄存器，可以用于对 GLDE[x]置位或者清零。

每当采样后的输入信号为 GLPOL[x]寄存器对应位段指定的电平时，被判定为有效的电平事件，并被锁存到 GLRAWIF[x]寄存器的相应标志位。但只有通过 GLIE[x]的对应位段使能后，这些事件才会触发中断。通过对 GLIC[x]位段写 1，可以清除 GLRAWIF[x]锁存的事件标志和 GLIF[x]锁存的中断标志。

为了便于调试，电平事件也可以通过向 GLIFRC[x]寄存器位写入 1 的方式强制触发。

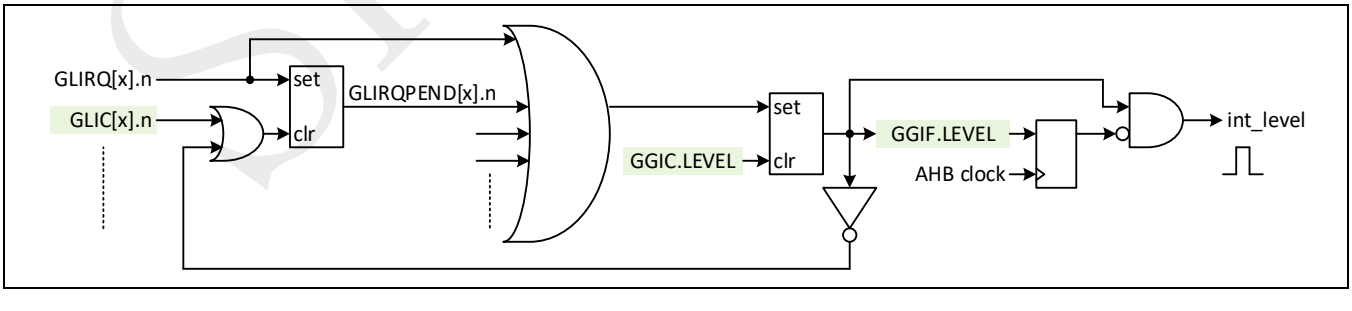
图 4-8：GPIO 电平检测



GGRAWIF 寄存器中的 LEVELx 位段为端口 x 的全局电平事件标志。它是 GLRAWIF[x]寄存器按位的逻辑或。因此，每当端口 x 上的任意引脚检测到有效电平时，GGRAWIF 寄存器的 LEVELx 标志位都会锁存 1。

图 4-8 中名为 GLIRQ[x].n 的信号是有效的电平触发中断请求，它们是已使能的电平事件，而相应的 GLIF[x]寄存器中的中断标志位仍为 0。也就是说，只有新的事件才能触发中断。

图 4-9：电平触发中断



如图 4-9 所示，所有有效的中断请求被逻辑 OR 在一起，生成最终的电平触发中断传递给 CPU，并锁存到全局中断标志寄存器 GGIF 的 LEVEL 位。通过向 GGIC 寄存器的 LEVEL 位写入 1，可以清除锁存的标志。

©2025 旋智电子科技（上海）有限公司

假如 PIN x 先触发电平中断请求，中断标志寄存器 GGIF.LEVEL 置 1，并触发中断，随后 PIN y、PIN z 也同时触发电平中断请求，此时 PIN y，PIN z 的电平中断请求都将会被挂起。

若 PIN x 中断服务程序执行完毕，并往中断清除寄存器 GGIC.LEVEL 写 1，此时 GGIF.LEVEL 标志位被清 0。随后在时钟的下一拍，将会发生两个动作：

1. PIN y、PIN z 挂起的请求再次将中断标志寄存器 GGIF.LEVEL 标志位置 1，并触发中断；
2. PIN x 中断清除动作（往中断清除寄存器 GGIC.LEVEL 写 1），将经由反相器清除 PIN y、PIN z 挂起的中断请求。

总结：当一个中断被响应时，不管发生多少个其他中断，后续能被响应的中断只有一个。

如上所述，当同时有两个中断请求时，只有一个中断请求会被响应，此时若想处理所有请求，建议按照以下步骤实现电平触发的 ISR 程序：

- 首先读取 GLIF 寄存器来确定要处理的请求；
- 根据请求进行相应的处理；
- 根据处理的请求清除 GLIF 寄存器；
- 清除 GGIF.LEVEL 位（否则，GPIO 电平触发中断生成将被阻塞）。

SPINTROL

©2025 旋智电子科技（上海）有限公司

4.4 寄存器

4.4.1 PINMUX 寄存器列表

表 4-3：PINMUX 模块基地址

外设模块	基地址
PINMUX	0x4000 0300

表 4-4：PINMUX 寄存器列表

寄存器	偏移地址	描述	复位值
GPIO0*	0x00	GPIO0 引脚控制寄存器	0x00010001
GPIO1*	0x04	GPIO1 引脚控制寄存器	0x00010001
GPIO2*	0x08	GPIO2 引脚控制寄存器	0x00010001
GPIO3*	0x0C	GPIO3 引脚控制寄存器	0x00010001
GPIO4*	0x10	GPIO4 引脚控制寄存器	0x00010001
GPIO5*	0x14	GPIO5 引脚控制寄存器	0x00010100
GPIO6*	0x18	GPIO6 引脚控制寄存器	0x05000100
GPIO7*	0x1C	GPIO7 引脚控制寄存器	0x05000100
GPIO8*	0x20	GPIO8 引脚控制寄存器	0x05000100
GPIO9*	0x24	GPIO9 引脚控制寄存器	0x05000100
GPIO10*	0x28	GPIO10 引脚控制寄存器	0x0B000100
GPIO11*	0x2C	GPIO11 引脚控制寄存器	0x0B000100
GPIO12*	0x30	GPIO12 引脚控制寄存器	0x00000100
GPIO13*	0x34	GPIO13 引脚控制寄存器	0x00000100
GPIO14*	0x38	GPIO14 引脚控制寄存器	0x09000100
GPIO15*	0x3C	GPIO15 引脚控制寄存器	0x09000100
GPIO16*	0x40	GPIO16 引脚控制寄存器	0x07000100
GPIO17*	0x44	GPIO17 引脚控制寄存器	0x07000100
GPIO18*	0x48	GPIO18 引脚控制寄存器	0x00010100
GPIO19*	0x4C	GPIO19 引脚控制寄存器	0x00000100
GPIO20*	0x50	GPIO20 引脚控制寄存器	0x00000100
GPIO21*	0x54	GPIO21 引脚控制寄存器	0x00000100
GPIO22*	0x58	GPIO22 引脚控制寄存器	0x00000100
GPIO23*	0x5C	GPIO23 引脚控制寄存器	0x00000100
GPIO24*	0x60	GPIO24 引脚控制寄存器	0x00000100
GPIO25*	0x64	GPIO25 引脚控制寄存器	0x0F000100
GPIO26*	0x68	GPIO26 引脚控制寄存器	0x0F000100
GPIO27*	0x6C	GPIO27 引脚控制寄存器	0x00010100
GPIO28*	0x70	GPIO28 引脚控制寄存器	0x00010001
GPIO29*	0x74	GPIO29 引脚控制寄存器	0x00010001
GPIO30*	0x78	GPIO30 引脚控制寄存器	0x00010001

©2025 旋智电子科技（上海）有限公司

寄存器	偏移地址	描述	复位值
GPIO31*	0x7C	GPIO31 引脚控制寄存器	0x00010001
GPIO32*	0x80	GPIO32 引脚控制寄存器	0x07000100
GPIO33*	0x84	GPIO33 引脚控制寄存器	0x07000100
GPIO34*	0x88	GPIO34 引脚控制寄存器	0x07000100
GPIO35*	0x8C	GPIO35 引脚控制寄存器	0x07000100
GPIO36*	0x90	GPIO36 引脚控制寄存器	0x09000001
GPIO37*	0x94	GPIO37 引脚控制寄存器	0x09000001
GPIO38*	0x98	GPIO38 引脚控制寄存器	0x09000001
GPIO39*	0x9C	GPIO39 引脚控制寄存器	0x09000001
GPIO40*	0xA0	GPIO40 引脚控制寄存器	0x00010001
COMPMONCTL*	0xA4	比较器输出监测控制寄存器	0x76543210
PINMUXREGKEY	0xA8	PINMUX 模块写使能寄存器	0x1ACCE551

注意： 由*标记的寄存器仅当 PWRREGKEY=0x1ACCE551 时才可以改写。

SPINTROL

©2025 旋智电子科技（上海）有限公司

4.4.2 PINMUX 寄存器

表 4-5：GPIO0 引脚控制寄存器（GPIO0）位段定义

GPIO0 (GPIO0 Pin Control Register) Offset：0x0 Default：0x00010001 Access：PINMUX->GPIO0							
31	30	29	28	27	26	25	24
RESERVED						PU	
23	22	21	20	19	18	17	16
RESERVED						PD	
15	14	13	12	11	10	9	8
RESERVED							ICHEN
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED			OCHSEL	

表 4-6：GPIO0 引脚控制寄存器（GPIO0）位段描述

位段	位段名	属性	复位值	描述
31:26	RESERVED_31_26	RO	0x0	保留
25:24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:18	RESERVED_23_18	RO	0x0	保留
17:16	PD	RW	0x1	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:9	RESERVED_15_9	RO	0x0	保留
8	ICHEN	RW	0x0	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7:6	STRENGTH	RW	0x0	输出驱动力 0：5mA 1：10mA 2：15mA 3：20mA
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:2	RESERVED_4_2	RO	0x0	保留
1:0	OCHSEL	RW	0x1	输出通道选择 00：GPIO0 01：ANA_IN0 10：PWM SOCA 输出监测 11：PWM SOCC 输出监测

©2025 旋智电子科技（上海）有限公司

表 4-7：GPIO1 引脚控制寄存器（GPIO1）位段定义

GPIO1 (GPIO1 Pin Control Register) Offset：0x4 Default：0x00010001 Access：PINMUX->GPIO1							
31	30	29	28	27	26	25	24
RESERVED						PU	
23	22	21	20	19	18	17	16
RESERVED						PD	
15	14	13	12	11	10	9	8
RESERVED							ICHEN
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED			OCHSEL	

表 4-8：GPIO1 引脚控制寄存器（GPIO1）位段描述

位段	位段名	属性	复位值	描述
31:26	RESERVED_31_26	RO	0x0	保留
25:24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:18	RESERVED_23_18	RO	0x0	保留
17:16	PD	RW	0x1	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:9	RESERVED_15_9	RO	0x0	保留
8	ICHEN	RW	0x0	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7:6	STRENGTH	RW	0x0	输出驱动力 0：5mA 1：10mA 2：15mA 3：20mA
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:2	RESERVED_4_2	RO	0x0	保留
1:0	OCHSEL	RW	0x1	输出通道选择 00：GPIO1 01：ANA_IN1 10：PWM SOCB 输出监测 11:ECAP APWM 模式输出

©2025 旋智电子科技（上海）有限公司

表 4-9：GPIO2 引脚控制寄存器（GPIO2）位段定义

GPIO2 (GPIO2 Pin Control Register) Offset：0x8 Default：0x00010001 Access：PINMUX->GPIO2							
31	30	29	28	27	26	25	24
RESERVED						PU	
23	22	21	20	19	18	17	16
RESERVED						PD	
15	14	13	12	11	10	9	8
RESERVED							ICHEN
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED			OCHSEL	

表 4-10：GPIO2 引脚控制寄存器（GPIO2）位段描述

位段	位段名	属性	复位值	描述
31:26	RESERVED_31_26	RO	0x0	保留
25:24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:18	RESERVED_23_18	RO	0x0	保留
17:16	PD	RW	0x1	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:9	RESERVED_15_9	RO	0x0	保留
8	ICHEN	RW	0x0	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7:6	STRENGTH	RW	0x0	输出驱动力 0：5mA 1：10mA 2：15mA 3：20mA
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:2	RESERVED_4_2	RO	0x0	保留
1:0	OCHSEL	RW	0x1	输出通道选择 00：GPIO2 01：ANA_IN2 10：CLK_DCLK 11：保留

©2025 旋智电子科技（上海）有限公司

表 4-11：GPIO3 引脚控制寄存器（GPIO3）位段 Layout

GPIO3 (GPIO3 Pin Control Register) Offset：0xC Default：0x00010001 Access：PINMUX->GPIO3							
31	30	29	28	27	26	25	24
RESERVED						PU	
23	22	21	20	19	18	17	16
RESERVED						PD	
15	14	13	12	11	10	9	8
RESERVED							ICHEN
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED			OCHSEL	

表 4-12：GPIO3 引脚控制寄存器（GPIO3）位段描述

位段	位段名	属性	复位值	描述
31:26	RESERVED_31_26	RO	0x0	保留
25:24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:18	RESERVED_23_18	RO	0x0	保留
17:16	PD	RW	0x1	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:9	RESERVED_15_9	RO	0x0	保留
8	ICHEN	RW	0x0	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7:6	STRENGTH	RW	0x0	输出驱动力 0：5mA 1：10mA 2：15mA 3：20mA
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:2	RESERVED_4_2	RO	0x0	保留
1:0	OCHSEL	RW	0x1	输出通道选择 00：GPIO3 01：ANA_IN3 10：XIN 11：COMP_MON0

©2025 旋智电子科技（上海）有限公司

表 4-13：GPIO4 引脚控制寄存器（GPIO4）位段定义

GPIO4 (GPIO4 Pin Control Register) Offset：0x10 Default：0x00010001 Access：PINMUX->GPIO4							
31	30	29	28	27	26	25	24
RESERVED						PU	
23	22	21	20	19	18	17	16
RESERVED						PD	
15	14	13	12	11	10	9	8
RESERVED							ICHEN
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED			OCHSEL	

表 4-14：GPIO4 引脚控制寄存器（GPIO4）位段描述

位段	位段名	属性	复位值	描述
31:26	RESERVED_31_26	RO	0x0	保留
25:24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:18	RESERVED_23_18	RO	0x0	保留
17:16	PD	RW	0x1	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:9	RESERVED_15_9	RO	0x0	保留
8	ICHEN	RW	0x0	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7:6	STRENGTH	RW	0x0	输出驱动力 0：5mA 1：10mA 2：15mA 3：20mA
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:2	RESERVED_4_2	RO	0x0	保留
1:0	OCHSEL	RW	0x1	输出通道选择 00：GPIO4 01：ANA_IN4 10：XOUT 11：COMP_MON1

SPINTROL

表 4-15：GPIO5 引脚控制寄存器（GPIO5）位段定义

GPIO5 (GPIO5 Pin Control Register) Offset：0x14 Default：0x00010100 Access： PINMUX->GPIO5							
31	30	29	28	27	26	25	24
RESERVED							PU
23	22	21	20	19	18	17	16
RESERVED							PD
15	14	13	12	11	10	9	8
RESERVED							ICHEN
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED			OCHSEL	

表 4-16：GPIO5 引脚控制寄存器（GPIO5）位段描述

位段	位段名	属性	复位值	描述
31:25	RESERVED_31_25	RO	0x0	保留
24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:17	RESERVED_23_17	RO	0x0	保留
16	PD	RW	0x1	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:9	RESERVED_15_9	RO	0x0	保留
8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:2	RESERVED_4_2	RO	0x0	保留
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO5 01：保留 10：PWM SOC 输出监测 它是 PWM_SOCAO, PWM_SOCBO 和 PWM_SOCCO 的逻辑或 11：ECAP APWM 模式输出

©2025 旋智电子科技（上海）有限公司

表 4-17：GPIO6 引脚控制寄存器（GPIO6）位段定义

GPIO6 (GPIO6 Pin Control Register) Offset：0x18 Default：0x05000100 Access：PINMUX->GPIO6							
31	30	29 RESERVED	28	27	26	25 PU	24
23	22	21 RESERVED	20	19	18	17 PD	16
15	14	13 RESERVED	12	11	10	9 ICHEN	8
7 RESERVED	6 STRENGTH	5 DEGLITCH	4 RESERVED	3	2 LOOPBACK	1 OCHSEL	0

表 4-18：GPIO6 引脚控制寄存器（GPIO6）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	PU	RW	0x5	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:19	RESERVED_23_19	RO	0x0	保留
18:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:11	RESERVED_15_11	RO	0x0	保留
10:8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 UART1 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO6 01：SPI0_SCLK 10：UART1_TXD 11：COMP_MON2

©2025 旋智电子科技（上海）有限公司

表 4-19：GPIO7 引脚控制寄存器（GPIO7）位段定义

GPIO7 (GPIO7 Pin Control Register) Offset：0x1C Default：0x05000100 Access：PINMUX->GPIO7							
31	30	29 RESERVED	28	27	26	25 PU	24
23	22	21 RESERVED	20	19	18	17 PD	16
15	14	13 RESERVED	12	11	10	9 ICHEN	8
7 RESERVED	6 STRENGTH	5 DEGLITCH	4 RESERVED	3	2 LOOPBACK	1 OCHSEL	0

表 4-20：GPIO7 引脚控制寄存器（GPIO7）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	PU	RW	0x5	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:19	RESERVED_23_19	RO	0x0	保留
18:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:11	RESERVED_15_11	RO	0x0	保留
10:8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 UART1 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO7 01：SPI0_SFRM 10：UART1_RXD 11：COMP_MON3

©2025 旋智电子科技（上海）有限公司

表 4-21：GPIO8 引脚控制寄存器（GPIO8）位段定义

GPIO8 (GPIO8 Pin Control Register) Offset：0x20 Default：0x05000100 Access：PINMUX->GPIO8							
31	30	29 RESERVED	28	27	26	25 PU	24
23	22	21 RESERVED	20	19	18	17 PD	16
15	14	13 RESERVED	12	11	10	9 ICHEN	8
7 RESERVED	6 STRENGTH	5 DEGLITCH	4 RESERVED	3	2 LOOPBACK	1 OCHSEL	0

表 4-22：GPIO8 引脚控制寄存器（GPIO8）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	PU	RW	0x5	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:19	RESERVED_23_19	RO	0x0	保留
18:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:11	RESERVED_15_11	RO	0x0	保留
10:8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 SPI0 和 CAN 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO8 01：SPI0_MOSI 10：CAN_TXD 11：COMP_MON4

©2025 旋智电子科技（上海）有限公司

表 4-23：GPIO9 引脚控制寄存器（GPIO9）位段定义

GPIO9 (GPIO9 Pin Control Register) Offset：0x24 Default：0x05000100 Access：PINMUX->GPIO9							
31	30	29 RESERVED	28	27	26	25 PU	24
23	22	21 RESERVED	20	19	18	17 PD	16
15	14	13 RESERVED	12	11	10	9 ICHEN	8
7 RESERVED	6 STRENGTH	5 DEGLITCH	4 RESERVED	3	2 LOOPBACK	1 OCHSEL	0

表 4-24：GPIO9 引脚控制寄存器（GPIO9）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	PU	RW	0x5	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:19	RESERVED_23_19	RO	0x0	保留
18:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:11	RESERVED_15_11	RO	0x0	保留
10:8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 SPI0 和 CAN 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO9 01：SPI0_MISO 10：CAN_RXD 11：COMP_MON5

©2025 旋智电子科技（上海）有限公司

表 4-25：GPIO10 引脚控制寄存器（GPIO10）位段定义

GPIO10 (GPIO10 Pin Control Register) Offset：0x28 Default：0x0B000100 Access：PINMUX->GPIO10							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

表 4-26：GPIO10 引脚控制寄存器（GPIO10）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:24	PU	RW	0xB	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:20	RESERVED_23_20	RO	0x0	保留
19:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:12	RESERVED_15_12	RO	0x0	保留
11:8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 UART0 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO10 01：UART0_TXD 10：PWM0A 11：I2C_SCL

©2025 旋智电子科技（上海）有限公司

表 4-27：GPIO11 引脚控制寄存器（GPIO11）位段定义

GPIO11 (GPIO11 Pin Control Register) Offset：0x2C Default：0x0B000100 Access：PINMUX->GPIO11							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

表 4-28：GPIO11 引脚控制寄存器（GPIO11）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:24	PU	RW	0xB	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:20	RESERVED_23_20	RO	0x0	保留
19:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:12	RESERVED_15_12	RO	0x0	保留
11:8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 UART0 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO11 01：UART0_RXD 10：PWM0B 11：I2C_SDA

©2025 旋智电子科技（上海）有限公司

表 4-29：GPIO12 引脚控制寄存器（GPIO12）位段定义

GPIO12 (GPIO12 Pin Control Register) Offset：0x30 Default：0x00000100 Access：PINMUX->GPIO12							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED						ICHEN	
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED			OCHSEL	

表 4-30：GPIO12 引脚控制寄存器（GPIO12）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:20	RESERVED_23_20	RO	0x0	保留
19:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:10	RESERVED_15_10	RO	0x0	保留
9:8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:2	RESERVED_4_2	RO	0x0	保留
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO12 01：SPI1_SCLK 10：PWM1A 11：PWM3A

©2025 旋智电子科技（上海）有限公司

表 4-31：GPIO13 引脚控制寄存器（GPIO13）位段定义

GPIO13 (GPIO13 Pin Control Register) Offset：0x34 Default：0x00000100 Access：PINMUX->GPIO13							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED						ICHEN	
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED			OCHSEL	

表 4-32：GPIO13 引脚控制寄存器（GPIO13）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:20	RESERVED_23_20	RO	0x0	保留
19:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:10	RESERVED_15_10	RO	0x0	保留
9:8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:2	RESERVED_4_2	RO	0x0	保留
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO13 01：SPI1_SFRM 10：PWM1B 11：PWM3B

©2025 旋智电子科技（上海）有限公司

表 4-33：GPIO14 引脚控制寄存器（GPIO14）位段定义

GPIO14 (GPIO14 Pin Control Register) Offset：0x38 Default：0x09000100 Access：PINMUX->GPIO14							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

表 4-34：GPIO14 引脚控制寄存器（GPIO14）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:24	PU	RW	0x9	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:20	RESERVED_23_20	RO	0x0	保留
19:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:12	RESERVED_15_12	RO	0x0	保留
11:8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 SPI1 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO14 01：SPI1_MOSI 10：PWM2A 11：I2C_SCL

©2025 旋智电子科技（上海）有限公司

表 4-35：GPIO15 引脚控制寄存器（GPIO15）位段定义

GPIO15 (GPIO15 Pin Control Register) Offset：0x3C Default：0x09000100 Access： PINMUX->GPIO15							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

表 4-36：GPIO15 引脚控制寄存器（GPIO15）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:24	PU	RW	0x9	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:20	RESERVED_23_20	RO	0x0	保留
19:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:12	RESERVED_15_12	RO	0x0	保留
11:8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 SPI1 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO15 01：SPI1_MISO 10：PWM2B 11：I2C_SDA

©2025 旋智电子科技（上海）有限公司

表 4-37：GPIO16 引脚控制寄存器（GPIO16）位段定义

GPIO16 (GPIO16 Pin Control Register) Offset：0x40 Default：0x07000100 Access：PINMUX->GPIO16							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

表 4-38：GPIO16 引脚控制寄存器（GPIO16）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:24	PU	RW	0x7	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:20	RESERVED_23_20	RO	0x0	保留
19:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:12	RESERVED_15_12	RO	0x0	保留
11:8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 UART1 和 CAN 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO16 01：UART1_TXD 10：CAN_TXD 11：SPI1_SFRM

©2025 旋智电子科技（上海）有限公司

表 4-39：GPIO17 引脚控制寄存器（GPIO17）位段定义

GPIO17 (GPIO17 Pin Control Register) Offset：0x44 Default：0x07000100 Access： PINMUX->GPIO17							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

表 4-40：GPIO17 引脚控制寄存器（GPIO17）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:24	PU	RW	0x7	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:20	RESERVED_23_20	RO	0x0	保留
19:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:12	RESERVED_15_12	RO	0x0	保留
11:8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 UART1 和 CAN 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO17 01：UART1_RXD 10：CAN_RXD 11：SPI1_SCLK

©2025 旋智电子科技（上海）有限公司

表 4-41：GPIO18 引脚控制寄存器（GPIO18）位段定义

GPIO18 (GPIO18 Pin Control Register) Offset：0x48 Default：0x00010100 Access：PINMUX->GPIO18							
31	30	29	28	27	26	25	24
RESERVED							PU
23	22	21	20	19	18	17	16
RESERVED							PD
15	14	13	12	11	10	9	8
RESERVED							ICHEN
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED			OCHSEL	

表 4-42：GPIO18 引脚控制寄存器（GPIO18）位段描述

位段	位段名	属性	复位值	描述
31:25	RESERVED_31_25	RO	0x0	保留
24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:17	RESERVED_23_17	RO	0x0	保留
16	PD	RW	0x1	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:9	RESERVED_15_9	RO	0x0	保留
8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:2	RESERVED_4_2	RO	0x0	保留
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO18 01：PWM_SOC 输出监测 它是 PWM_SOCAO, PWM_SOCBO 和 PWM_SOCCO 的逻辑或 10：PWM_SYNC 输出监测 11：ECAP APWM 模式输出

©2025 旋智电子科技（上海）有限公司

表 4-43：GPIO19 引脚控制寄存器（GPIO19）位段定义

GPIO19 (GPIO19 Pin Control Register) Offset：0x4C Default：0x00000100 Access：PINMUX->GPIO19							
31	30	29	28	27	26	25	24
RESERVED					PU		
23	22	21	20	19	18	17	16
RESERVED					PD		
15	14	13	12	11	10	9	8
RESERVED							ICHEN
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED			OCHSEL	

表 4-44：GPIO19 引脚控制寄存器（GPIO19）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:19	RESERVED_23_19	RO	0x0	保留
18:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:9	RESERVED_15_9	RO	0x0	保留
8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:2	RESERVED_4_2	RO	0x0	保留
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO19 01：PWM2B 10：PWM0A 11：ECAP APWM 模式输出

©2025 旋智电子科技（上海）有限公司

表 4-45：GPIO20 引脚控制寄存器（GPIO20）位段定义

GPIO20 (GPIO20 Pin Control Register) Offset：0x50 Default：0x00000100 Access：PINMUX->GPIO20							
31	30	29	28	27	26	25	24
RESERVED					PU		
23	22	21	20	19	18	17	16
RESERVED					PD		
15	14	13	12	11	10	9	8
RESERVED							ICHEN
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED			OCHSEL	

表 4-46：GPIO20 引脚控制寄存器（GPIO20）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:19	RESERVED_23_19	RO	0x0	保留
18:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:9	RESERVED_15_9	RO	0x0	保留
8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:2	RESERVED_4_2	RO	0x0	保留
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO20 01：PWM2A 10：PWM1A 11：ECAP APWM 模式输出

©2025 旋智电子科技（上海）有限公司

表 4-47：GPIO21 引脚控制寄存器（GPIO21）位段定义

GPIO21 (GPIO21 Pin Control Register) Offset：0x54 Default：0x00000100
Access：PINMUX->GPIO21

31	30	29	28	27	26	25	24
RESERVED					PU		
23	22	21	20	19	18	17	16
RESERVED					PD		
15	14	13	12	11	10	9	8
RESERVED							ICHEN
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED			OCHSEL	

表 4-48：GPIO21 引脚控制寄存器（GPIO21）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:19	RESERVED_23_19	RO	0x0	保留
18:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:9	RESERVED_15_9	RO	0x0	保留
8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:2	RESERVED_4_2	RO	0x0	保留
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO21 01：PWM1B 10：PWM2A 11：ECAP APWM 模式输出

©2025 旋智电子科技（上海）有限公司

表 4-49：GPIO22 引脚控制寄存器（GPIO22）位段定义

GPIO22 (GPIO22 Pin Control Register) Offset：0x58 Default：0x00000100 Access：PINMUX->GPIO22							
31	30	29	28	27	26	25	24
RESERVED					PU		
23	22	21	20	19	18	17	16
RESERVED					PD		
15	14	13	12	11	10	9	8
RESERVED							ICHEN
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED			OCHSEL	

表 4-50：GPIO22 引脚控制寄存器（GPIO22）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:19	RESERVED_23_19	RO	0x0	保留
18:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:9	RESERVED_15_9	RO	0x0	保留
8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:2	RESERVED_4_2	RO	0x0	保留
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO22 01：PWM1A 10：PWM0B 11：ECAP APWM 模式输出

SPINTROL

表 4-51：GPIO23 引脚控制寄存器（GPIO23）位段定义

GPIO23 (GPIO23 Pin Control Register) Offset：0x5C Default：0x00000100 Access：PINMUX->GPIO23							
31	30	29	28	27	26	25	24
RESERVED					PU		
23	22	21	20	19	18	17	16
RESERVED					PD		
15	14	13	12	11	10	9	8
RESERVED							ICHEN
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED			OCHSEL	

表 4-52：GPIO23 引脚控制寄存器（GPIO23）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:19	RESERVED_23_19	RO	0x0	保留
18:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:9	RESERVED_15_9	RO	0x0	保留
8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:2	RESERVED_4_2	RO	0x0	保留
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO23 01：PWM0B 10：PWM1B 11：ECAP APWM 模式输出

©2025 旋智电子科技（上海）有限公司

表 4-53：GPIO24 引脚控制寄存器（GPIO24）位段定义

GPIO24 (GPIO24 Pin Control Register) Offset：0x60 Default：0x00000100 Access：PINMUX->GPIO24							
31	30	29	28	27	26	25	24
RESERVED					PU		
23	22	21	20	19	18	17	16
RESERVED					PD		
15	14	13	12	11	10	9	8
RESERVED							ICHEN
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED			OCHSEL	

表 4-54：GPIO24 引脚控制寄存器（GPIO24）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:19	RESERVED_23_19	RO	0x0	保留
18:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:9	RESERVED_15_9	RO	0x0	保留
8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:2	RESERVED_4_2	RO	0x0	保留
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO24 01：PWM0A 10：PWM2B 11：ECAP APWM 模式输出

©2025 旋智电子科技（上海）有限公司

表 4-55：GPIO25 引脚控制寄存器（GPIO25）位段定义

GPIO25 (GPIO25 Pin Control Register) Offset：0x64 Default：0x0F000100 Access：PINMUX->GPIO25							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

表 4-56：GPIO25 引脚控制寄存器（GPIO25）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:24	PU	RW	0xF	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:20	RESERVED_23_20	RO	0x0	保留
19:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:12	RESERVED_15_12	RO	0x0	保留
11:8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7:6	STRENGTH	RW	0x0	输出驱动力 00：5mA 01：10mA 10：15mA 11：20mA
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 UART1 和 CAN 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO25 01：UART1_TXD 10：CAN_TXD 11：I2C_SCL

©2025 旋智电子科技（上海）有限公司

表 4-57：GPIO26 引脚控制寄存器（GPIO26）位段定义

GPIO26 (GPIO26 Pin Control Register) Offset：0x68 Default：0x0F000100 Access：PINMUX->GPIO26							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

表 4-58：GPIO26 引脚控制寄存器（GPIO26）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:24	PU	RW	0xF	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:20	RESERVED_23_20	RO	0x0	保留
19:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:12	RESERVED_15_12	RO	0x0	保留
11:8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7:6	STRENGTH	RW	0x0	输出驱动力 00：5mA 01：10mA 10：15mA 11：20mA
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 UART1 和 CAN 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO26 01：UART1_RXD 10：CAN_RXD 11：I2C_SDA

SPINTROL

表 4-59：GPIO27 引脚控制寄存器（GPIO27）位段定义

GPIO27 (GPIO27 Pin Control Register) Offset：0x6C Default：0x00010100 Access：PINMUX->GPIO27							
31	30	29	28	27	26	25	24
RESERVED							PU
23	22	21	20	19	18	17	16
RESERVED							PD
15	14	13	12	11	10	9	8
RESERVED							ICHEN
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED				

表 4-60：GPIO27 引脚控制寄存器（GPIO27）位段描述

位段	位段名	属性	复位值	描述
31:25	RESERVED_31_25	RO	0x0	保留
24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:17	RESERVED_23_17	RO	0x0	保留
16	PD	RW	0x1	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:9	RESERVED_15_9	RO	0x0	保留
8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7:6	STRENGTH	RW	0x0	输出驱动力 00：5mA 01：10mA 10：15mA 11：20mA
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:0	RESERVED_4_0	RO	0x0	保留

©2025 旋智电子科技（上海）有限公司

表 4-61：GPIO28 引脚控制寄存器（GPIO28）位段定义

GPIO28 (GPIO28 Pin Control Register) Offset：0x70 Default：0x00010001 Access：PINMUX->GPIO28							
31	30	29 RESERVED	28	27	26	25 PU	24
23	22	21 RESERVED	20	19	18	17 PD	16
15	14	13 RESERVED	12	11	10	9 ICHEN	8
7 STRENGTH	6	5 DEGLITCH	4	3 RESERVED	2	1 OCHSEL	0

表 4-62：GPIO28 引脚控制寄存器（GPIO28）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:19	RESERVED_23_19	RO	0x0	保留
18:16	PD	RW	0x1	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:11	RESERVED_15_11	RO	0x0	保留
10:8	ICHEN	RW	0x0	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7:6	STRENGTH	RW	0x0	输出驱动力 00：5mA 01：10mA 10：15mA 11：20mA
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:2	RESERVED_4_2	RO	0x0	保留
1:0	OCHSEL	RW	0x1	输出通道选择 00：GPIO28 01：ANA_IN5 10：SPI1_SCLK 11：PWM SOCA 输出监测

©2025 旋智电子科技（上海）有限公司

表 4-63：GPIO29 引脚控制寄存器（GPIO29）位段定义

GPIO29 (GPIO29 Pin Control Register) Offset：0x74 Default：0x00010001 Access：PINMUX->GPIO29							
31	30	29 RESERVED	28	27	26	25 PU	24
23	22	21 RESERVED	20	19	18	17 PD	16
15	14	13 RESERVED	12	11	10	9 ICHEN	8
7 STRENGTH	6	5 DEGLITCH	4	3 RESERVED	2	1 OCHSEL	0

表 4-64：GPIO29 引脚控制寄存器（GPIO29）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:19	RESERVED_23_19	RO	0x0	保留
18:16	PD	RW	0x1	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:11	RESERVED_15_11	RO	0x0	保留
10:8	ICHEN	RW	0x0	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7:6	STRENGTH	RW	0x0	输出驱动力 00：5mA 01：10mA 10：15mA 11：20mA
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:2	RESERVED_4_2	RO	0x0	保留
1:0	OCHSEL	RW	0x1	输出通道选择 00：GPIO29 01：ANA_IN6 10：SPI1_SFRM 11：PWM SOCB 输出监测

©2025 旋智电子科技（上海）有限公司

表 4-65：GPIO30 引脚控制寄存器（GPIO30）位段定义

GPIO30 (GPIO30 Pin Control Register) Offset：0x78 Default：0x00010001 Access：PINMUX->GPIO30							
31	30	29 RESERVED	28	27	26	25 PU	24
23	22	21 RESERVED	20	19	18	17 PD	16
15	14	13 RESERVED	12	11	10	9 ICHEN	8
7 STRENGTH	6	5 DEGLITCH	4 RESERVED	3	2 LOOPBACK	1 OCHSEL	0

表 4-66：GPIO30 引脚控制寄存器（GPIO30）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:19	RESERVED_23_19	RO	0x0	保留
18:16	PD	RW	0x1	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:11	RESERVED_15_11	RO	0x0	保留
10:8	ICHEN	RW	0x0	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7:6	STRENGTH	RW	0x0	输出驱动力 00：5mA 01：10mA 10：15mA 11：20mA
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 SPI1 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x1	输出通道选择 00：GPIO30 01：ANA_IN7 10：SPI1_MOSI 11：PWM SOCC 输出监测

©2025 旋智电子科技（上海）有限公司

表 4-67：GPIO31 引脚控制寄存器（GPIO31）位段定义

GPIO31 (GPIO31 Pin Control Register) Offset：0x7C Default：0x00010001 Access：PINMUX->GPIO31							
31	30	29	28	27	26	25	24
		RESERVED				PU	
23	22	21	20	19	18	17	16
		RESERVED				PD	
15	14	13	12	11	10	9	8
		RESERVED				ICHEN	
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

表 4-68：GPIO31 引脚控制寄存器（GPIO31）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:19	RESERVED_23_19	RO	0x0	保留
18:16	PD	RW	0x1	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:11	RESERVED_15_11	RO	0x0	保留
10:8	ICHEN	RW	0x0	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7:6	STRENGTH	RW	0x0	输出驱动力 00：5mA 01：10mA 10：15mA 11：20mA
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 SPI1 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x1	输出通道选择 00：GPIO31 01：ANA_IN8 10：SPI1_MISO 11：PWM_SYNC 输出监测

©2025 旋智电子科技（上海）有限公司

表 4-69：GPIO32 引脚控制寄存器（GPIO32）位段定义

GPIO32 (GPIO32 Pin Control Register) Offset：0x80 Default：0x07000100 Access：PINMUX->GPIO32							
31	30	29 RESERVED	28	27	26	25 PU	24
23	22	21 RESERVED	20	19	18	17 PD	16
15	14	13 RESERVED	12	11	10	9 ICHEN	8
7 RESERVED	6 STRENGTH	5 DEGLITCH	4 RESERVED	3	2 LOOPBACK	1 OCHSEL	0

表 4-70：GPIO32 引脚控制寄存器（GPIO32）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	PU	RW	0x7	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:19	RESERVED_23_19	RO	0x0	保留
18:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:11	RESERVED_15_11	RO	0x0	保留
10:8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 UART1 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO32 01：UART1_TXD 10：I2C_SCL 11：COMP_MON6

©2025 旋智电子科技（上海）有限公司

表 4-71：GPIO33 引脚控制寄存器（GPIO33）位段定义

GPIO33 (GPIO33 Pin Control Register) Offset：0x84 Default：0x07000100 Access：PINMUX->GPIO33							
31	30	29 RESERVED	28	27	26	25 PU	24
23	22	21 RESERVED	20	19	18	17 PD	16
15	14	13 RESERVED	12	11	10	9 ICHEN	8
7 RESERVED	6 STRENGTH	5 DEGLITCH	4 RESERVED	3	2 LOOPBACK	1 OCHSEL	0

表 4-72：GPIO33 引脚控制寄存器（GPIO33）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	PU	RW	0x7	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:19	RESERVED_23_19	RO	0x0	保留
18:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:11	RESERVED_15_11	RO	0x0	保留
10:8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 UART1 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO33 01：UART1_RXD 10：I2C_SDA 11：COMP_MON7

©2025 旋智电子科技（上海）有限公司

表 4-73：GPIO34 引脚控制寄存器（GPIO34）位段定义

GPIO34 (GPIO34 Pin Control Register) Offset：0x88 Default：0x07000100 Access：PINMUX->GPIO34							
31	30	29 RESERVED	28	27	26	25 PU	24
23	22	21 RESERVED	20	19	18	17 PD	16
15	14	13 RESERVED	12	11	10	9 ICHEN	8
7 RESERVED	6 STRENGTH	5 DEGLITCH	4 RESERVED	3	2 LOOPBACK	1 OCHSEL	0

表 4-74：GPIO34 引脚控制寄存器（GPIO34）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	PU	RW	0x7	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:19	RESERVED_23_19	RO	0x0	保留
18:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:11	RESERVED_15_11	RO	0x0	保留
10:8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 CAN 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO34 01：CAN_TXD 10：I2C_SCL 11：保留

©2025 旋智电子科技（上海）有限公司

表 4-75：GPIO35 引脚控制寄存器（GPIO35）位段定义

GPIO35 (GPIO35 Pin Control Register) Offset：0x8C Default：0x07000100 Access：PINMUX->GPIO35							
31	30	29 RESERVED	28	27	26	25 PU	24
23	22	21 RESERVED	20	19	18	17 PD	16
15	14	13 RESERVED	12	11	10	9 ICHEN	8
7 RESERVED	6 STRENGTH	5 DEGLITCH	4 RESERVED	3	2 LOOPBACK	1 OCHSEL	0

表 4-76：GPIO35 引脚控制寄存器（GPIO35）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26:24	PU	RW	0x7	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:19	RESERVED_23_19	RO	0x0	保留
18:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:11	RESERVED_15_11	RO	0x0	保留
10:8	ICHEN	RW	0x1	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7	RESERVED_7	RO	0x0	保留
6	STRENGTH	RW	0x0	输出驱动力（详细数据请见数据手册） 0：弱驱动能力（大概为满驱动能力的 1/10） 1：满驱动能力
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 CAN 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x0	输出通道选择 00：GPIO35 01：CAN_RXD 10：I2C_SDA 11：保留

©2025 旋智电子科技（上海）有限公司

表 4-77：GPIO36 引脚控制寄存器（GPIO36）位段定义

GPIO36 (GPIO36 Pin Control Register) Offset：0x90 Default：0x09000001 Access：PINMUX->GPIO36							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

表 4-78：GPIO36 引脚控制寄存器（GPIO36）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:24	PU	RW	0x9	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:20	RESERVED_23_20	RO	0x0	保留
19:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:12	RESERVED_15_12	RO	0x0	保留
11:8	ICHEN	RW	0x0	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7:6	STRENGTH	RW	0x0	输出驱动力 00：5mA 01：10mA 10：15mA 11：20mA
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 UART1 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x1	输出通道选择 00：GPIO36 01：ANA_IN9_PHU 10：SPI1_SCLK 11：UART1_TXD

©2025 旋智电子科技（上海）有限公司

表 4-79：GPIO37 引脚控制寄存器（GPIO37）位段定义

GPIO37 (GPIO37 Pin Control Register) Offset：0x94 Default：0x09000001 Access：PINMUX->GPIO37							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

表 4-80：GPIO37 引脚控制寄存器（GPIO37）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:24	PU	RW	0x9	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:20	RESERVED_23_20	RO	0x0	保留
19:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:12	RESERVED_15_12	RO	0x0	保留
11:8	ICHEN	RW	0x0	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7:6	STRENGTH	RW	0x0	输出驱动力 00：5mA 01：10mA 10：15mA 11：20mA
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 UART1 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x1	输出通道选择 00：GPIO37 01：ANA_IN10_PHV 10：SPI1_SFRM 11：UART1_RXD

©2025 旋智电子科技（上海）有限公司

表 4-81：GPIO38 引脚控制寄存器（GPIO38）位段定义

GPIO38 (GPIO38 Pin Control Register) Offset：0x98 Default：0x09000001 Access：PINMUX->GPIO38							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

表 4-82：GPIO38 引脚控制寄存器（GPIO38）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:24	PU	RW	0x9	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:20	RESERVED_23_20	RO	0x0	保留
19:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:12	RESERVED_15_12	RO	0x0	保留
11:8	ICHEN	RW	0x0	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7:6	STRENGTH	RW	0x0	输出驱动力 00：5mA 01：10mA 10：15mA 11：20mA
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 SPI1 和 CAN 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x1	输出通道选择 00：GPIO38 01：ANA_IN11_PHW 10：SPI1_MOSI 11：CAN_TXD

©2025 旋智电子科技（上海）有限公司

表 4-83：GPIO39 引脚控制寄存器（GPIO39）位段定义

GPIO39 (GPIO39 Pin Control Register) Offset：0x9C Default：0x09000001 Access：PINMUX->GPIO39							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

表 4-84：GPIO39 引脚控制寄存器（GPIO39）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:24	PU	RW	0x9	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:20	RESERVED_23_20	RO	0x0	保留
19:16	PD	RW	0x0	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:12	RESERVED_15_12	RO	0x0	保留
11:8	ICHEN	RW	0x0	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7:6	STRENGTH	RW	0x0	输出驱动力 00：5mA 01：10mA 10：15mA 11：20mA
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:3	RESERVED_4_3	RO	0x0	保留
2	LOOPBACK	RW	0x0	将 SPI1 和 CAN 的输出环回到输入通道 0：输出到引脚 1：输出直接环回到输入
1:0	OCHSEL	RW	0x1	输出通道选择 00：GPIO39 01：ANA_IN12_PHC 10：SPI1_MISO 11：CAN_RXD

©2025 旋智电子科技（上海）有限公司

表 4-85：GPIO40 引脚控制寄存器（GPIO40）位段定义

GPIO40 (GPIO40 Pin Control Register) Offset：0xA0 Default：0x00010001 Access：PINMUX->GPIO40							
31	30	29	28	27	26	25	24
RESERVED						PU	
23	22	21	20	19	18	17	16
RESERVED						PD	
15	14	13	12	11	10	9	8
RESERVED							ICHEN
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED				OCHSEL

表 4-86：GPIO40 引脚控制寄存器（GPIO40）位段描述

位段	位段名	属性	复位值	描述
31:26	RESERVED_31_26	RO	0x0	保留
25:24	PU	RW	0x0	上拉使能 若 PU[n]=1 则使能通道 n 的上拉
23:18	RESERVED_23_18	RO	0x0	保留
17:16	PD	RW	0x1	下拉使能 若 PD[n]=1 则使能通道 n 的下拉
15:9	RESERVED_15_9	RO	0x0	保留
8	ICHEN	RW	0x0	输入通道使能 若 ICHEN[n]=1 则使能通道 n 的输入
7:6	STRENGTH	RW	0x0	输出驱动力 00：5mA 01：10mA 10：15mA 11：20mA
5	DEGLITCH	RW	0x0	输入消抖 0：关闭 1：使能
4:1	RESERVED_4_1	RO	0x0	保留
0	OCHSEL	RW	0x1	输出通道选择 0：GPIO40 1：ANA_IN13

SPINTROL

表 4-87：比较器输出监测控制寄存器（COMPMONCTL）位段定义

COMPMONCTL (Comparator Output Monitor Control Register) Offset：0xA4 Default：0x76543210 Access：PINMUX->COMPMONCTL							
31 RESERVED	30	29 SRC7	28	27 RESERVED	26	25 SRC6	24
23 RESERVED	22	21 SRC5	20	19 RESERVED	18	17 SRC4	16
15 RESERVED	14	13 SRC3	12	11 RESERVED	10	9 SRC2	8
7 RESERVED	6	5 SRC1	4	3 RESERVED	2	1 SRC0	0

表 4-88：比较器输出监测控制寄存器（COMPMONCTL）位段描述

位段	位段名	属性	复位值	描述
31	RESERVED_31	RO	0x0	保留
30:28	SRC7	RW	0x7	COMP 输出监测 7 源选择 000：COMP0L 输出 001：COMP0H 输出 010：PHCOMPU 原始输出 011：PHCOMPV 原始输出 100：PHCOMPW 原始输出 101：PHCOMPU 滤波后的输出 110：PHCOMPV 滤波后的输出 111：PHCOMPW 滤波后的输出
27	RESERVED_27	RO	0x0	保留
26:24	SRC6	RW	0x6	COMP 输出监测 6 源选择 000：COMP0L 输出 001：COMP0H 输出 010：PHCOMPU 原始输出 011：PHCOMPV 原始输出 100：PHCOMPW 原始输出 101：PHCOMPU 滤波后的输出 110：PHCOMPV 滤波后的输出 111：PHCOMPW 滤波后的输出
23	RESERVED_23	RO	0x0	保留
22:20	SRC5	RW	0x5	COMP 输出监测 5 源选择 000：COMP0L 输出 001：COMP0H 输出 010：PHCOMPU 原始输出 011：PHCOMPV 原始输出 100：PHCOMPW 原始输出 101：PHCOMPU 滤波后的输出

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				110：PHCOMPV 滤波后的输出 111：PHCOMPW 滤波后的输出
19	RESERVED_19	RO	0x0	保留
18:16	SRC4	RW	0x4	COMP 输出监测 4 源选择 000：COMP0L 输出 001：COMP0H 输出 010：PHCOMPU 原始输出 011：PHCOMPV 原始输出 100：PHCOMPW 原始输出 101：PHCOMPU 滤波后的输出 110：PHCOMPV 滤波后的输出 111：PHCOMPW 滤波后的输出
15	RESERVED_15	RO	0x0	保留
14:12	SRC3	RW	0x3	COMP 输出监测 3 源选择 000：COMP0L 输出 001：COMP0H 输出 010：PHCOMPU 原始输出 011：PHCOMPV 原始输出 100：PHCOMPW 原始输出 101：PHCOMPU 滤波后的输出 110：PHCOMPV 滤波后的输出 111：PHCOMPW 滤波后的输出
11	RESERVED_11	RO	0x0	保留
10:8	SRC2	RW	0x2	COMP 输出监测 2 源选择 000：COMP0L 输出 001：COMP0H 输出 010：PHCOMPU 原始输出 011：PHCOMPV 原始输出 100：PHCOMPW 原始输出 101：PHCOMPU 滤波后的输出 110：PHCOMPV 滤波后的输出 111：PHCOMPW 滤波后的输出
7	RESERVED_7	RO	0x0	保留
6:4	SRC1	RW	0x1	COMP 输出监测 1 源选择 000：COMP0L 输出 001：COMP0H 输出 010：PHCOMPU 原始输出 011：PHCOMPV 原始输出 100：PHCOMPW 原始输出 101：PHCOMPU 滤波后的输出 110：PHCOMPV 滤波后的输出 111：PHCOMPW 滤波后的输出
3	RESERVED_3	RO	0x0	保留

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
2:0	SRC0	RW	0x0	COMP 输出监测 0 源选择 000：COMP0L 输出 001：COMP0H 输出 010：PHCOMPU 原始输出 011：PHCOMPV 原始输出 100：PHCOMPW 原始输出 101：PHCOMPU 滤波后的输出 110：PHCOMPV 滤波后的输出 111：PHCOMPW 滤波后的输出

表 4-89：PINMUX 模块写使能寄存器（PINMUXREGKEY）位段定义

PINMUXREGKEY (PINMUX Register Write-Allow Key Register) Offset：0xA8 Default：0x1ACCE551 Access：PINMUX->PINMUXREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

表 4-90：PINMUX 模块写使能寄存器（PINMUXREGKEY）位段描述

位段	位段名	属性	复位值	描述
31:0	KEY	RW	0x1ACCE551	写入 0x1ACCE551 解锁受保护的 PINMUX 寄存器

©2025 旋智电子科技（上海）有限公司

4.4.3 GPIO 寄存器列表

表 4-91：GPIO 模块基地址

外设模块	基地址
GPIO	0x4000 1200

表 4-92：GPIO 寄存器列表

寄存器	偏移地址	描述	复位值
GPLR*	0x000	GPIO 引脚电平寄存器 GPLR[n] ([0:1])	0x00000000
GSLR*	0x010	GPIO 按位输出置高寄存器 GSLR[n] ([0:1])	0x00000000
GCLR*	0x020	GPIO 按位输出置低寄存器 GCLR[n] ([0:1])	0x00000000
GPDR*	0x030	GPIO 引脚方向控制寄存器 GPDR[n] ([0:1])	0x00000000
GSDR*	0x040	GPIO 按位设置输出方向寄存器 GSDR[n] ([0:1])	0x00000000
GCDR*	0x050	GPIO 按位设置输入方向寄存器 GCDR[n] ([0:1])	0x00000000
GRER*	0x060	GPIO 上升沿检测使能寄存器 GRER[n] ([0:1])	0x00000000
GSRE*	0x070	GPIO 上升沿检测按位使能寄存器 GSRE[n] ([0:1])	0x00000000
GCRE*	0x080	GPIO 上升沿检测按位关闭寄存器 GCRE[n] ([0:1])	0x00000000
GFER*	0x090	GPIO 下降沿检测使能寄存器 GFER[n] ([0:1])	0x00000000
GSFE*	0x0A0	GPIO 下降沿检测按位使能寄存器 GSFE[n] ([0:1])	0x00000000
GCFE*	0x0B0	GPIO 下降沿检测按位关闭寄存器 GCFE[n] ([0:1])	0x00000000
GRERAWIF	0x0C0	GPIO 上升沿事件标志寄存器 GRERAWIF[n] ([0:1])	0x00000000
GFERAWIF	0x0D0	GPIO 下降沿事件标志寄存器 GFERAWIF[n] ([0:1])	0x00000000
GREIF	0x0E0	GPIO 上升沿中断标志寄存器 GREIF[n] ([0:1])	0x00000000
GFEIF	0x0F0	GPIO 下降沿中断标志寄存器 GFEIF[n] ([0:1])	0x00000000
GREIE*	0x100	GPIO 上升沿中断使能寄存器 GREIE[n] ([0:1])	0x00000000
GFEIE*	0x110	GPIO 下降沿中断使能寄存器 GFEIE[n] ([0:1])	0x00000000
GREIC	0x120	GPIO 上升沿中断清除寄存器 GREIC[n] ([0:1])	0x00000000
GFEIC	0x130	GPIO 下降沿中断清除寄存器 GFEIC[n] ([0:1])	0x00000000
GREIFRC*	0x140	GPIO 上升沿事件强制寄存器 GREIFRC[n] ([0:1])	0x00000000
GFEIFRC*	0x150	GPIO 下降沿事件强制寄存器 GFEIFRC[n] ([0:1])	0x00000000
GLDE*	0x160	GPIO 电平检测使能寄存器 GLDE[n] ([0:1])	0x00000000
GSLDE*	0x170	GPIO 电平检测按位使能寄存器 GSLDE[n] ([0:1])	0x00000000
GCLDE*	0x180	GPIO 电平检测按位关闭寄存器 GCLDE[n] ([0:1])	0x00000000
GLPOL*	0x190	GPIO 电平检测极性控制寄存器 GLPOL[n] ([0:1])	0x00000000
GLRAWIF	0x1A0	GPIO 电平事件标志寄存器 GLRAWIF[n] ([0:1])	0x00000000
GLIF	0x1B0	GPIO 电平中断标志寄存器 GLIF[n] ([0:1])	0x00000000
GLIE*	0x1C0	GPIO 电平中断使能寄存器 GLIE[n] ([0:1])	0x00000000
GLIC	0x1D0	GPIO 电平中断清除寄存器 GLIC[n] ([0:1])	0x00000000
GLIFRC*	0x1E0	GPIO 电平事件强制寄存器 GLIFRC[n] ([0:1])	0x00000000
GGRAWIF	0x1F0	GPIO 全局事件标志寄存器 GGRAWIF	0x00000000
GGIF	0x1F4	GPIO 全局中断标志寄存器 GGIF	0x00000000

©2025 旋智电子科技（上海）有限公司

寄存器	偏移地址	描述	复位值
GGIC	0x1F8	GPIO 全局中断清除寄存器 GGIC	0x00000000
GPIOREGKEY	0x1FC	GPIO 模块写使能寄存器 GPIOREGKEY	0x1ACCE551

注意： 由*标记的寄存器仅当 PWRREGKEY=0x1ACCE551 时才可以改写。

SPINTROL

©2025 旋智电子科技（上海）有限公司

4.4.4 GPIO 寄存器

表 4-93：GPIO 引脚电平寄存器（GPLR[n] ([0:1])）位段定义

GPLR (GPIO Pin Level Register n ([0:1])) Access：GPIO->GPLR Offset：0x000 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-94：GPIO 引脚电平寄存器（GPLR[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	GPIO 各引脚电平 0：写 0 设置对应引脚输出低，读 0 表示当前电平为低 1：写 1 设置对应引脚输出高，读 1 表示当前电平为高

表 4-95：GPIO 按位输出置高（GSLR[n] ([0:1])）位段定义

GSLR (GPIO Pin Output Set Register n ([0:1])) Access：GPIO->GSLR Offset：0x010 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-96：GPIO 按位输出置高（GSLR[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	W1S	0x0	设置 GPIO 各引脚输出高 0：写 0 无效，读总是 0 1：写 1 设置对应引脚输出高，位段自动清零

SPINTROL

表 4-97：GPIO 按位输出置低寄存器（GCLR[n] ([0:1])）位段定义

GCLR (GPIO Pin Output Clear Register n ([0:1])) Access：GPIO->GCLR Offset：0x020 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-98：GPIO 按位输出置低寄存器（GCLR[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	W1C	0x0	设置 GPIO 各引脚输出低 0：写 0 无效，读总是 0 1：写 1 设置对应引脚输出低，位段自动清零

表 4-99：GPIO 引脚方向寄存器（GPDR[n] ([0:1])）位段定义

GPDR (GPIO Pin Direction Register n ([0:1])) Access：GPIO->GPDR Offset：0x030 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-100：GPIO 引脚方向寄存器（GPDR[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	设置 GPIO 各引脚方向 0：设置对应引脚为输入方向 1：设置对应引脚为输出方向

©2025 旋智电子科技（上海）有限公司

表 4-101：GPIO 按位设置输出方向寄存器（GSDR[n] ([0:1])）位段定义

GSDR (GPIO Pin Bitwise Set Direction Register n ([0:1])) Access：GPIO->GSDR Offset：0x040 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-102：GPIO 按位设置输出方向寄存器（GSDR[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	W1S	0x0	设置 GPIO 各引脚为输出（GPDR[n]位段置 1） 0：写 0 无效，读总是 0 1：写 1 设置对应引脚为输出，位段自动清零

表 4-103：GPIO 按位设置输入方向寄存器（GCDR[n] ([0:1])）位段定义

GCDR (GPIO Pin Bitwise Clear Direction Register n ([0:1])) Access：GPIO->GCDR Offset：0x050 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-104：GPIO 按位设置输入方向寄存器（GCDR[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	W1C	0x0	设置 GPIO 各引脚为输入（GPDR[n]位段置 0） 0：写 0 无效，读总是 0 1：写 1 设置对应引脚为输入，位段自动清零

SPINTROL

表 4-105：GPIO 上升沿检测使能寄存器（GRER[n] ([0:1])）位段定义

GRER (GPIO Pin Rising Edge Detect Enable Register n ([0:1])) Access：GPIO->GRER Offset：0x060 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-106：GPIO 上升沿检测使能寄存器（GRER[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	使能 GPIO 各引脚上升沿检测 0：关闭对应引脚上升沿检测 1：使能对应引脚上升沿检测

表 4-107：GPIO 上升沿检测按位使能寄存器（GSRE[n] ([0:1])）位段定义

GSRE (GPIO Pin Bitwise Set Rising Edge Detect Enable Register n ([0:1])) Access：GPIO->GSRE Offset：0x070 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-108：GPIO 上升沿检测按位使能寄存器（GSRE[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	W1S	0x0	按位使能 GPIO 上升沿检测（GRER[n]位段置 1） 0：写 0 无效，读总是 0 1：写 1 使能对应引脚上升沿检测，位段自动清零

©2025 旋智电子科技（上海）有限公司

表 4-109：GPIO 上升沿检测按位关闭寄存器（GCRE[n] ([0:1])）位段定义

GCRE (GPIO Pin Bitwise Clear Rising Edge Detect Enable Register n ([0:1]))							
Access：GPIO->GCRE		Offset：0x080		Default：0x00000000			
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-110：GPIO 上升沿检测按位关闭寄存器（GCRE[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	W1C	0x0	按位关闭 GPIO 上升沿检测（GRER[n]位段置 0） 0：写 0 无效，读总是 0 1：写 1 关闭对应引脚上升沿检测，位段自动清零

表 4-111：GPIO 下降沿检测使能寄存器（GFER[n] ([0:1])）位段定义

GFER (GPIO Pin Falling Edge Detect Enable Register n ([0:1]))							
Access：GPIO->GFER		Offset：0x090		Default：0x00000000			
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-112：GPIO 下降沿检测使能寄存器（GFER[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	使能 GPIO 各引脚下降沿检测 0：关闭对应引脚下降沿检测 1：使能对应引脚下降沿检测

SPINTROL

表 4-113：GPIO 下降沿检测按位使能寄存器（GSFE[n] ([0:1])）位段定义

GSFE (GPIO Pin Bitwise Set Falling Edge Detect Enable Register n ([0:1])) Access：GPIO->GSFE Offset：0x0A0 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-114：GPIO 下降沿检测按位使能寄存器（GSFE[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	W1S	0x0	按位使能 GPIO 下降沿检测（GFER[n]位段置 1） 0：写 0 无效，读总是 0 1：写 1 使能对应引脚下降沿检测，位段自动清零

表 4-115：GPIO 下降沿检测按位关闭寄存器（GCFE[n] ([0:1])）位段定义

GCFE (GPIO Pin Bitwise Clear Falling Edge Detect Enable Register n ([0:1])) Access：GPIO->GCFE Offset：0x0B0 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-116：GPIO 下降沿检测按位关闭寄存器（GCFE[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	W1C	0x0	按位关闭 GPIO 下降沿检测（GFER[n]位段置 0） 0：写 0 无效，读总是 0 1：写 1 关闭对应引脚下降沿检测，位段自动清零

©2025 旋智电子科技（上海）有限公司

表 4-117：GPIO 上升沿事件标志寄存器（GRERAWIF[n] ([0:1])）位段定义

GRERAWIF (GPIO Pin Rising Edge Detect Interrupt Raw Flag Register n ([0:1])) Access：GPIO->GRERAWIF Offset：0x0C0 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-118：GPIO 上升沿事件标志寄存器（GRERAWIF[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	锁存的 GPIO 上升沿事件标志 0：GPIO 对应引脚上未检测到上升沿事件 1：GPIO 对应引脚上检测到上升沿事件

表 4-119：GPIO 下降沿事件标志寄存器（GFERAWIF[n] ([0:1])）位段定义

GFERAWIF (GPIO Pin Falling Edge Detect Interrupt Raw Flag Register n ([0:1])) Access：GPIO->GFERAWIF Offset：0x0D0 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-120：GPIO 下降沿事件标志寄存器（GFERAWIF[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	锁存的 GPIO 下降沿事件标志 0：GPIO 对应引脚上未检测到下降沿事件 1：GPIO 对应引脚上检测到下降沿事件

SPINTROL

表 4-121：GPIO 上升沿中断标志寄存器（GREIF[n] ([0:1])）位段定义

GREIF (GPIO Pin Rising Edge Triggered Interrupt Flag Register n ([0:1])) Access：GPIO->GREIF Offset：0x0E0 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-122：GPIO 上升沿中断标志寄存器（GREIF[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	锁存的 GPIO 上升沿中断标志 0：GPIO 对应引脚不曾触发上升沿中断 1：GPIO 对应引脚曾触发上升沿中断

表 4-123：GPIO 下降沿中断标志寄存器（GFEIF[n] ([0:1])）位段定义

GFEIF (GPIO Pin Falling Edge Triggered Interrupt Flag Register n ([0:1])) Access：GPIO->GFEIF Offset：0x0F0 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-124：GPIO 下降沿中断标志寄存器（GFEIF[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	锁存的 GPIO 下降沿中断标志 0：GPIO 对应引脚不曾触发下降沿中断 1：GPIO 对应引脚曾触发下降沿中断

©2025 旋智电子科技（上海）有限公司

表 4-125：GPIO 上升沿中断使能寄存器（GREIE[n] ([0:1])）位段定义

GREIE (GPIO Pin Rising Edge Triggered Interrupt Enable Register n ([0:1])) Access：GPIO->GREIE Offset：0x100 Default：0x00000000							
31	30	29	28	27	26	25	24
			VAL				
23	22	21	20	19	18	17	16
			VAL				
15	14	13	12	11	10	9	8
			VAL				
7	6	5	4	3	2	1	0
			VAL				

表 4-126：GPIO 上升沿中断使能寄存器（GREIE[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	GPIO 各引脚上升沿中断使能 0：关闭对应引脚上升沿触发中断 1：使能对应引脚上升沿触发中断

表 4-127：GPIO 下降沿中断使能寄存器（GFEIE[n] ([0:1])）位段定义

GFEIE (GPIO Pin Falling Edge Triggered Interrupt Enable Register n ([0:1])) Access：GPIO->GFEIE Offset：0x110 Default：0x00000000							
31	30	29	28	27	26	25	24
			VAL				
23	22	21	20	19	18	17	16
			VAL				
15	14	13	12	11	10	9	8
			VAL				
7	6	5	4	3	2	1	0
			VAL				

表 4-128：GPIO 下降沿中断使能寄存器（GFEIE[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	GPIO 各引脚下降沿中断使能 0：关闭对应引脚下降沿触发中断 1：使能对应引脚下降沿触发中断

SPINTROL

表 4-129：GPIO 上升沿中断清除寄存器（GREIC[n] ([0:1])）位段定义

GREIC (GPIO Pin Rising Edge Triggered Interrupt Clear Register n ([0:1])) Access：GPIO->GREIC Offset：0x120 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-130：GPIO 上升沿中断清除寄存器（GREIC[n] ([0:1])）位段定义

位段	位段名	属性	复位值	描述
31:0	VAL	W1C	0x0	清除锁存的 GPIO 各引脚上升沿事件和中断标志 0：写 0 无效，读总是 0 1：写 1 清除对应的 GRERAWIF[n]和 GREIF[n]位段 本位段自动清零

表 4-131：GPIO 下降沿中断清除寄存器（GFEIC[n] ([0:1])）位段定义

GFEIC (GPIO Pin Falling Edge Triggered Interrupt Clear Register n ([0:1])) Access：GPIO->GFEIC Offset：0x130 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-132：GPIO 下降沿中断清除寄存器（GFEIC[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	W1C	0x0	清除锁存的 GPIO 各引脚下降沿事件和中断标志 0：写 0 无效，读总是 0 1：写 1 清除对应的 GFERAWIF[n]和 GFEIF[n]位段 本位段自动清零

©2025 旋智电子科技（上海）有限公司

表 4-133：GPIO 上升沿事件强制寄存器（GREIFRC[n] ([0:1])）位段定义

GREIFRC (GPIO Pin Rising Edge Event Force Register n ([0:1])) Access：GPIO->GREIFRC Offset：0x140 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-134：GPIO 上升沿事件强制寄存器（GREIFRC[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	W1S	0x0	强制 GPIO 各引脚上升沿事件 0：写 0 无效，读总是 0 1：写 1 强制对应引脚上升沿事件，位段自动清零

表 4-135：GPIO 下降沿事件强制寄存器（GFEIFRC[n] ([0:1])）位段定义

GFEIFRC (GPIO Pin Falling Edge Event Force Register n ([0:1])) Access：GPIO->GFEIFRC Offset：0x150 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-136：GPIO 下降沿事件强制寄存器（GFEIFRC[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	W1S	0x0	强制 GPIO 各引脚下降沿事件 0：写 0 无效，读总是 0 1：写 1 强制对应引脚下降沿事件，位段自动清零

SPINTROL

表 4-137：GPIO 电平检测使能寄存器（GLDE[n] ([0:1])）位段定义

GLDE (GPIO Pin Level Detect Enable Register n ([0:1])) Access：GPIO->GLDE Offset：0x160 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-138：GPIO 电平检测使能寄存器（GLDE[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	GPIO 各引脚电平检测使能 0：关闭对应引脚的电平检测 1：使能对应引脚的电平检测

表 4-139：GPIO 电平检测按位使能寄存器（GSLDE[n] ([0:1])）位段定义

GSLDE (GPIO Pin Bitwise Set Level Detect Enable Register n ([0:1])) Access：GPIO->GSLDE Offset：0x170 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-140：GPIO 电平检测按位使能寄存器（GSLDE[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	W1S	0x0	按位使能各引脚电平检测（GLDE[n]置 1） 0：写 0 无效，读总是 0 1：写 1 使能对应引脚电平检测，位段自动清零

©2025 旋智电子科技（上海）有限公司

表 4-141：GPIO 电平检测按位关闭寄存器（GCLDE[n] ([0:1])）位段定义

GCLDE (GPIO Pin Bitwise Clear Level Detect Enable Register n ([0:1])) Access：GPIO->GCLDE Offset：0x180 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-142：GPIO 电平检测按位关闭寄存器（GCLDE[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	W1C	0x0	按位关闭各引脚电平检测（GLDE[n]置 0） 0：写 0 无效，读总是 0 1：写 1 关闭对应引脚电平检测，位段自动清零

表 4-143：GPIO 电平检测极性寄存器（GLPOL[n] ([0:1])）位段定义

GLPOL (GPIO Pin Level Detect Polarity Register n ([0:1])) Access：GPIO->GLPOL Offset：0x190 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-144：GPIO 电平检测极性寄存器（GLPOL[n] ([0:1])）位段定义

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	GPIO 各引脚电平检测极性 0：检测到低为有效电平 1：检测到高为有效电平

SPINTROL

表 4-145：GPIO 电平事件标志寄存器（GLRAWIF[n] ([0:1])）位段定义

GLRAWIF (GPIO Pin Level Detect Interrupt Raw Flag Register n ([0:1])) Access：GPIO->GLRAWIF Offset：0x1A0 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-146：GPIO 电平事件标志寄存器（GLRAWIF[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	锁存的 GPIO 各引脚电平事件标志 0：对应的引脚上未检测到指定的电平 1：对应的引脚上检测到指定的电平

表 4-147：GPIO 电平中断标志寄存器（GLIF[n] ([0:1])）位段定义

GLIF (GPIO Pin Level-Triggered Interrupt Flag Register n ([0:1])) Access：GPIO->GLIF Offset：0x1B0 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-148：GPIO 电平中断标志寄存器（GLIF[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	锁存的 GPIO 各引脚电平中断标志 0：对应的引脚上不曾触发电平中断 1：对应的引脚上曾触发电平中断

©2025 旋智电子科技（上海）有限公司

表 4-149：GPIO 电平中断使能寄存器（GLIE[n] ([0:1])）位段定义

GLIE (GPIO Pin Level-Triggered Interrupt Enable Register n ([0:1])) Access：GPIO->GLIE Offset：0x1C0 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-150：GPIO 电平中断使能寄存器（GLIE[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	GPIO 各引脚电平中断使能 0：关闭对应引脚触发电平中断 1：使能对应引脚触发电平中断

表 4-151：GPIO 电平中断清除寄存器（GLIC[n] ([0:1])）位段定义

GLIC (GPIO Pin Level-Triggered Interrupt Clear Register n ([0:1])) Access：GPIO->GLIC Offset：0x1D0 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-152：GPIO 电平中断清除寄存器（GLIC[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	W1C	0x0	Clear GPIO level-triggered event and interrupt flag 0：Write a 0 has no effect and always reads back 0 1：Write a 1 clears the GLRAWIF bit and GLIF bit. This bit is self-cleared to 0.

SPINTROL

表 4-153：GPIO 电平事件强制寄存器（GLIFRC[n] ([0:1])）位段定义

GLIFRC (GPIO Pin Level-Triggered Event Force Register n ([0:1])) Access：GPIO->GLIFRC Offset：0x1E0 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 4-154：GPIO 电平事件强制寄存器（GLIFRC[n] ([0:1])）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	W1S	0x0	强制 GPIO 各引脚电平事件 0：写 0 无效，读总是 0 1：写 1 强制对应引脚电平事件，位段自动清零

表 4-155：GPIO 全局事件标志（GGRAWIF）位段定义

GGRAWIF (GPIO Global Interrupt Raw Flag Register) Access：GPIO->GGRAWIF Offset：0x1F0 Default：0x00000000							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				FEDGE	REDGE	EDGE	LEVEL

表 4-156：GPIO 全局事件标志寄存器（GGRAWIF）位段描述

位段	位段名	属性	复位值	描述
31:4	RESERVED_31_4	RO	0x0	保留
3	FEDGE	RO	0x0	GPIO 端口下降沿事件标志 本位段是 GFERAWIF[2]所有位的逻辑或 0：GPIO 端口上未检测到下降沿事件 1：GPIO 端口上检测到下降沿事件

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
2	REDGE	RO	0x0	GPIO 端口上升沿事件标志 本位段是 GRERAWIF[2]所有位的逻辑或 0：GPIO 端口上未检测到上升沿事件 1：GPIO 端口上检测到上升沿事件
1	EDGE	RO	0x0	GPIO 端口边沿事件标志 本位段是 GRERAWIF[2]和 GFERAWIF[2]所有位的逻辑或 0：GPIO 端口上未检测到边沿事件 1：GPIO 端口上检测到边沿事件
0	LEVEL	RO	0x0	GPIO 端口电平事件标志 本位段是 GLRAWIF[2]所有位的逻辑或 0：GPIO 端口上未检测到电平事件 1：GPIO 端口上检测到电平事件

表 4-157：GPIO 全局中断标志寄存器（GGIF）位段定义

GGIF (GPIO Global Interrupt Flag Register) Access：GPIO->GGIF Offset：0x1F4 Default：0x00000000							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						EDGE	LEVEL

表 4-158：GPIO 全局中断标志寄存器（GGIF）位段描述

位段	位段名	属性	复位值	描述
31:2	RESERVED_31_2	RO	0x0	保留
1	EDGE	RO	0x0	GPIO 端口边沿中断标志，本位段被清除前，端口上新的边沿事件将无法触发中断 0：GPIO 端口上无任何引脚触发边沿中断 1：GPIO 端口上有引脚触发边沿中断
0	LEVEL	RO	0x0	GPIO 端口电平中断标志，本位段被清除前，端口上新的电平事件将无法触发中断 0：GPIO 端口上无任何引脚触发电平中断 1：GPIO 端口上有引脚触发电平中断

©2025 旋智电子科技（上海）有限公司

表 4-159：GPIO 全局中断清除寄存器（GGIC）位段定义

GGIC (GPIO Global Interrupt Clear Register) Access：GPIO->GGIC Offset：0x1F8 Default：0x00000000							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						EDGE	LEVEL

表 4-160：GPIO 全局中断清除（GGIC）位段描述

位段	位段名	属性	复位值	描述
31:2	RESERVED_31_2	RO	0x0	保留
1	EDGE	W1C	0x0	GPIO 端口边沿中断清除 0：写 0 无效，读总是 0 1：写 1 清除中断标志，本位段自动清零
0	LEVEL	W1C	0x0	GPIO 端口电平中断清除 0：写 0 无效，读总是 0 1：写 1 清除中断标志，本位段自动清零

表 4-161：GPIO 模块写使能寄存器（GPIOREGKEY）位段定义

GPIOREGKEY (GPIO Register Write-Allow Key Register) Access：GPIO->GPIOREGKEY Offset：0x1FC Default：0x1ACCE551							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

表 4-162：GPIO 模块写使能寄存器（GPIOREGKEY）位段描述

位段	位段名	属性	复位值	描述
31:0	KEY	RW	0x1ACCE551	写入 0x1ACCE551 解锁受保护的 GPIO 寄存器

©2025 旋智电子科技（上海）有限公司

5 中断和事件

5.1 嵌套向量中断控制器（NVIC）

嵌套向量中断控制器（NVIC）是 ARM Cortex-M4 处理器单元的重要组成部分，由于与 CPU 紧密耦合，它提供了最低的中断延迟和对后续中断的高效处理能力。

5.1.1 NVIC 特性

NVIC 支持以下特性：

- 最多支持 54 个向量中断（不包括 16 个 Cortex-M4 中断）
- 每个中断可编程的中断优先级级别为 16 级（使用 4 位中断优先级）。较高级别对应较低的优先级，因此级别 0 是最高的中断优先级。
- 控制系统异常和外设中断
- 支持中断尾链接和延迟到达
- 支持中断惰性堆栈
- 支持非屏蔽中断（NMI）

5.1.2 中断和异常向量

当一个异常事件发生并被 Cortex-M4F 处理时，处理器需要知道异常处理函数的确切起始地址。因此，采用了中断向量表的机制。中断向量表是位于系统储存器的一个由 32 位数据组成的数组，每一个 32 位数据代表一个异常处理函数的起始地址，根据异常向量号乘以 4 来排列。中断向量表的位置由 NVIC 中一个叫做 VTOR（地址 0xE000ED08）的可编程寄存器控制。在复位后，VTOR 的值被复位为 0；因此，在复位后，中断向量表的位置在地址 0。表 5-1 是 SPC1169 的向量表。

表 5-1：SPC1169 中断向量表

异常号	CMSIS 中断号	略缩语	优先级	描述	偏移地址
-	-	MSP	-	初始化主堆栈指针	0x0000_0000
1	-	Reset	-3，最高级	复位	0x0000_0004
2	-14	NMI	-2	不可屏蔽中断	0x0000_0008
3	-13	Hard Fault	-1	所有类型的故障	0x0000_000C
4	-12	MemManage	可配置	存储器管理故障	0x0000_00010
5	-11	Bus Fault	可配置	总线系统故障	0x0000_0014
6	-10	Usage Fault	可配置	用法故障	0x0000_0018
7-10	-	-	-	保留	0x0000_001C – 0x0000_002B

©2025 旋智电子科技（上海）有限公司

异常号	CMSIS 中断号	略缩语	优先级	描述	偏移地址
11	-5	SVCall	-	保留	0x0000_002C
12	-4	Debug Monitor	-	保留	0x0000_0030
13	-	-	-	保留	0x0000_0034
14	-2	PendSV	可配置	通过 SVC 指令调用的系统服务	0x0000_0038
15	-1	SysTick	可配置	调试监视器	0x0000_003C
16	0	WDT1	可配置	WDT1 中断	0x0000_0040
17	1	MEM	可配置	Memory ECC 单比特错误中断	0x0000_0044
18	2	-	-	保留	0x0000_0048
19	3	-	-	保留	0x0000_004C
20	4	GPIO_LEVEL	可配置	GPIO 电平中断	0x0000_0050
21	5	GPIO_EDGE	可配置	GPIO 边沿中断	0x0000_0054
22	6	UART0	可配置	UART0 中断	0x0000_0058
23	7	SPI0	可配置	SPI0 中断	0x0000_005C
24	8	UART1	可配置	UART1 中断	0x0000_0060
25	9	SPI1	可配置	SPI1 中断	0x0000_0064
26	10	CANTRG0	可配置	CAN 信箱组 0 中断	0x0000_0068
27	11	CANTRG1	可配置	CAN 信箱组 1 中断	0x0000_006C
28	12	CANTRG2	可配置	CAN 信箱组 2 中断	0x0000_0070
29	13	CANTRG3	可配置	CAN 信箱组 3 中断	0x0000_0074
30	14	CANTRG4	可配置	CAN 信箱组 4 中断	0x0000_0078
31	15	CANTRG5	可配置	CAN 信箱组 5 中断	0x0000_007C
32	16	CANTRG6	可配置	CAN 信箱组 6 中断	0x0000_0080
33	17	CANTRG7	可配置	CAN 信箱组 7 中断	0x0000_0084
34	18	CANMISC	可配置	CAN 杂项中断	0x0000_0088
35	19	I2C	可配置	I2C 中断	0x0000_008C
36	20	ADCCH0	可配置	ADC0 通道 0 转换完成中断	0x0000_0090
37	21	ADCCH1	可配置	ADC0 通道 1 转换完成中断	0x0000_0094
38	22	ADCCH2	可配置	ADC0 通道 2 转换完成中断	0x0000_0098
39	23	ADCCH3	可配置	ADC0 通道 3 转换完成中断	0x0000_009C
40	24	ADCCH4	可配置	ADC0 通道 4 转换完成中断	0x0000_00A0
41	25	ADCCH5	可配置	ADC0 通道 5 转换完成中断	0x0000_00A4
42	26	ADCCH6	可配置	ADC0 通道 6 转换完成中断	0x0000_00A8
43	27	ADCCH7	可配置	ADC0 通道 7 转换完成中断	0x0000_00AC
44	28	ADCCH8	可配置	ADC0 通道 8 转换完成中断	0x0000_00B0
45	29	ADCCH9	可配置	ADC0 通道 9 转换完成中断	0x0000_00B4
46	30	ADCCH10	可配置	ADC0 通道 10 转换完成中断	0x0000_00B8
47	31	ADCCH11	可配置	ADC0 通道 11 转换完成中断	0x0000_00BC
48	32	ADCCH12	可配置	ADC0 通道 12 转换完成中断	0x0000_00C0
49	33	ADCCH13	可配置	ADC0 通道 13 转换完成中断	0x0000_00C4

©2025 旋智电子科技（上海）有限公司

异常号	CMSIS 中断号	略缩语	优先级	描述	偏移地址
50	34	ADCCH14	可配置	ADC0 通道 14 转换完成中断	0x0000_00C8
51	35	ADCCH15	可配置	ADC0 通道 15 转换完成中断	0x0000_00CC
52	36	ADCPPU0	可配置	ADCPPU0 中断	0x0000_00D0
53	37	ADCPPU1	可配置	ADCPPU1 中断	0x0000_00D4
54	38	ADCPPU2	可配置	ADCPPU2 中断	0x0000_00D8
55	39	PHCOMP	可配置	相位比较器中断	0x0000_00DC
56	40	PWM0	可配置	PWM0 中断	0x0000_00E0
57	41	PWM1	可配置	PWM1 中断	0x0000_00E4
58	42	PWM2	可配置	PWM2 中断	0x0000_00E8
59	43	PWM3	可配置	PWM3 中断	0x0000_00EC
60	44	PWM0TZ	可配置	PWM0 封锁中断	0x0000_00F0
61	45	PWM1TZ	可配置	PWM1 封锁中断	0x0000_00F4
62	46	PWM2TZ	可配置	PWM2 封锁中断	0x0000_00F8
63	47	PWM3TZ	可配置	PWM3 封锁中断	0x0000_00FC
64	48	ECAP	可配置	ECAP 中断	0x0000_0100
65	49	DMAC	可配置	DMAC 中断	0x0000_0104
66	50	TIMER0	可配置	通用定时器 0 中断	0x0000_0108
67	51	TIMER1	可配置	通用定时器 1 中断	0x0000_010C
68	52	TIMER2	可配置	通用定时器 2 中断	0x0000_0110
69	53	CRC	可配置	CRC 中断	0x0000_0114

5.1.3 NMI 中断

可以通过 SYSTEM->NMIE 寄存器启用以下事件作为触发非屏蔽中断的源：

- 电源/监测器事件
  - VCAP12 过压
  - VCAP12 欠压
  - DVDD33 过压
  - DVDD33 欠压
- 时钟事件
  - PLL 解锁
  - 时钟检测器报告错误
- 存储器 ECC 多位错误
- 看门狗定时器 0 超时
- GPIO 引脚触发事件
- 13 位监测 ADC 检测到错误
- AHB 地址和数据的 ECC 多位错误
- JTAG/SWD 调试模式进入

©2025 旋智电子科技（上海）有限公司

- 产品测试模式进入

5.1.4 NVIC 中断配置

NVIC 中断可以通过写入 SETENA 寄存器来启用，也可以通过写入 CLRENA 寄存器来禁用。这样，启用或禁用中断不会影响其他中断启用状态。SETENA/CLRENA 寄存器是 32 位宽的；每个比特表示一个中断输入。

NVIC 中断挂起状态可通过中断设置挂起（SETPEND）和中断清除挂起（CLRPEND）寄存器访问。您可以通过CLRPEND 寄存器取消当前挂起的异常，或者通过SETPEND 寄存器生成软件中断。

每个外部中断都有一个相关的优先级寄存器，其最大宽度为 8 位，最小宽度为 3 位，位宽的具体位数，取决于每个厂商的设计。每个优先级寄存器可以根据优先级组设置进一步划分为抢占优先级和子优先级。SPC1169 实现了 4 位优先级，因此该芯片支持 16 级可编程优先级。

有关 NVIC 配置的更多信息，请参阅 ARM 发布的文档。

5.2 SysTick 定时器

Cortex-M4F 处理器包含了一个集成的系统定时器，叫做 SysTick。该定时器是一个 24 比特位宽的向下计数的计数器。一旦它计数到 0，计数器会从 RELOAD 寄存器中加载重装载值。只有清除 SysTick 控制和状态寄存器的使能位，该定时器才会停止。

当 SysTick 定时器计数从 1 变为 0 的时候，SysTick 控制和状态寄存器的 COUNTFLAG 位段置 1。COUNTFLAG 位可以通过下面的任意一种方式清除：

- 通过处理器读取 SysTick 控制和状态寄存器；
- 通过向 SysTick 当前值寄存器中写入任意值以清除 SysTick 计数器值。

SysTick 计数器可以被用来产生固定间隔的 SysTick 中断。这对操作系统 OS、任务以及资源管理常常是必要的。TICKINT 位段置 1 可以使能 SysTick 中断的产生。

有关 SysTick 定时器配置的更多信息，请参阅 ARM 发布的文档。

©2025 旋智电子科技（上海）有限公司

6 通用定时器（TIMER）

6.1 概述

SPC1169 包括三个 32 位的通用定时器（GPT），具有以下特性：

- 32 位向下计数器，具有专用时钟
- 在计数器达到零时生成中断
- 在计数器达到零时生成 ADCSOC 事件
- 在计数器达到零时生成 PWMSYNCI 事件
- 支持 4 种模式
  - 通用定时器：周期性向下计数器；
  - 门控定时器：由外部输入电平启用的向下计数器；
  - 事件计数器：在外部输入边沿触发的向下计数器；
  - 事件捕获：在外部输入边沿触发时，记录时间戳的向下计数器；

6.2 功能描述

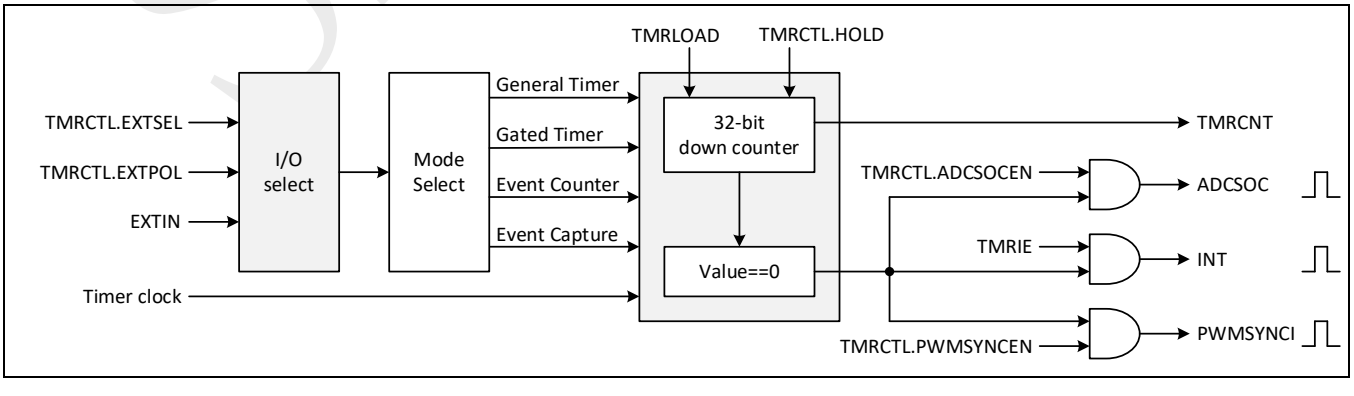
如图 6-1 所示，每一个通用定时器都基于一个 32 位的周期性向下计数器。当使能定时器之后，计数器加载 TMRLOAD 寄存器的值，并开始向下计数。当计数到 0 后，计数器从 TMRLOAD 寄存器的值重新开始向下计数，且有如下动作发生：

- 如果使能了定时器中断，将会有中断产生，并锁存中断标志直到被手动清除
- 如果 TMRCTL.ADCSOCEN 使能，将会产生一个 ADCSOC 事件
- 如果 TMRCTL.PWMSYNCEN 使能，将会产生一个 PWMSYNCI 事件

用于外部信号输入的 I/O 引脚及其有效的电平或者边沿可以通过 TMRCTL.EXTSEL 和 TMRCTL.EXTPOL 寄存器位段控制。详细信息请参见表 6-7 和表 6-8。

在系统复位之后，定时器默认为关闭状态。

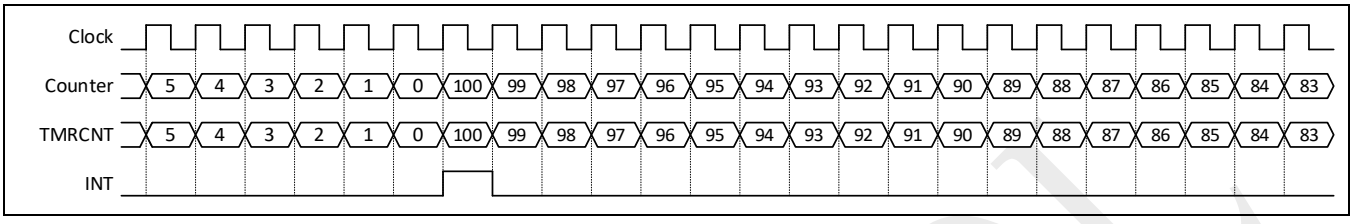
图 6-1：通用定时器框图



6.2.1 通用计时模式

如图 6-2 所示，该模式下 32 位计时器在模块时钟上升沿递减计数，计数到 0 时重新加载初值并继续计数。寄存器 TMRCNT 的回读值始终是实时的计数值。

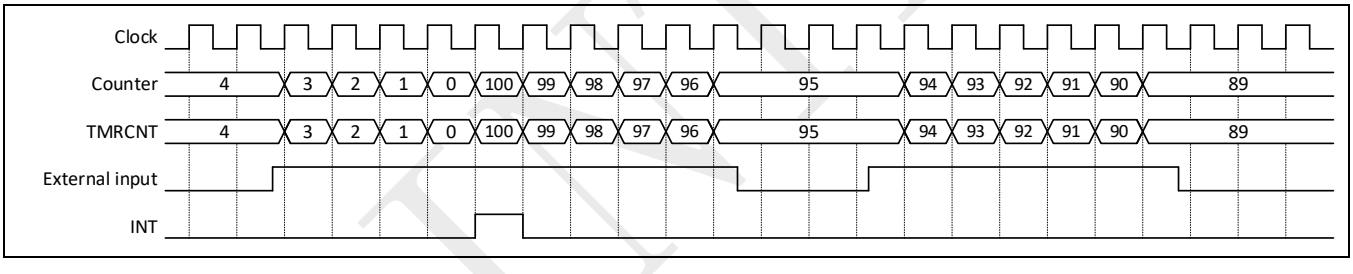
图 6-2：通用定时器在通用定时模式下的波形（TMRLOAD=100）



6.2.2 门控定时模式

如图 6-3 所示，该模式与通用定时模式类似，只是计数器仅在外部引脚输入有效电平时才计数。用于外部输入的 I/O 引脚及有效电平软件可配。寄存器 TMRCNT 的回读值始终是实时的计数值。

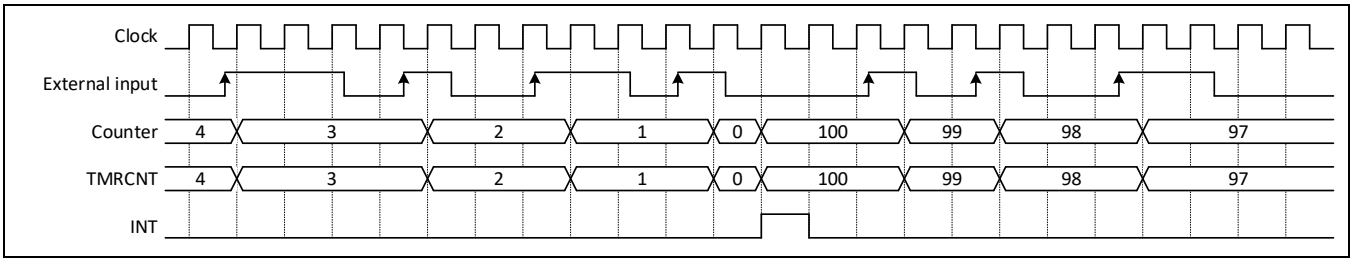
图 6-3：通用定时器在门控定时模式下的波形（TMRLOAD=100）



6.2.3 事件计数模式

如图 6-4 所示，该模式下计数器仅当在外部输入引脚上检测到一个有效边沿的时候才计数。当计数到 0 时，将自动加载初始值，为下一轮计数做准备。因此，每当外部输入引脚上累计检测到的有效边沿数等于 TMRLOAD 时，将产生一个中断。该模式可以帮助用户在某个引脚上检测预定数量的信号沿并通过中断告知。用于外部输入的 I/O 引脚及有效电平软件可配。寄存器 TMRCNT 的回读值始终是实时的计数值。为了能正确检测信号沿，外部输入信号的高低电平保持时间必须大于定时器模块的时钟周期。

图 6-4：通用定时器在事件计数模式下的波形（TMRLOAD=100）

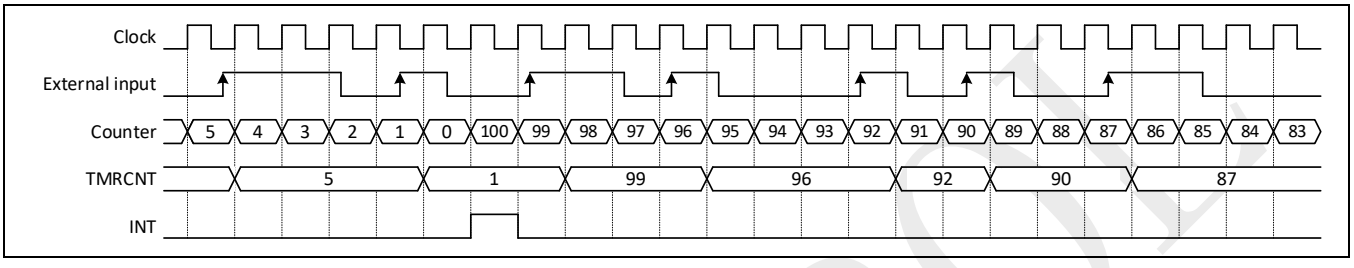


©2025 旋智电子科技（上海）有限公司

6.2.4 事件捕获模式

如图 6-5 所示，该模式下计数器的行为与通用定时模式完全相同。但是，TMRCNT 寄存器的回读值并非实时的计数值，而是检测到外部输入有效边沿时锁存的计数值。用于外部输入的 I/O 引脚及有效电平软件可配。为了能正确检测信号沿，外部输入信号的高低电平保持时间必须大于定时器模块的时钟周期。

图 6-5：通用定时器在事件捕获模式下的波形（TMRLOAD=100）



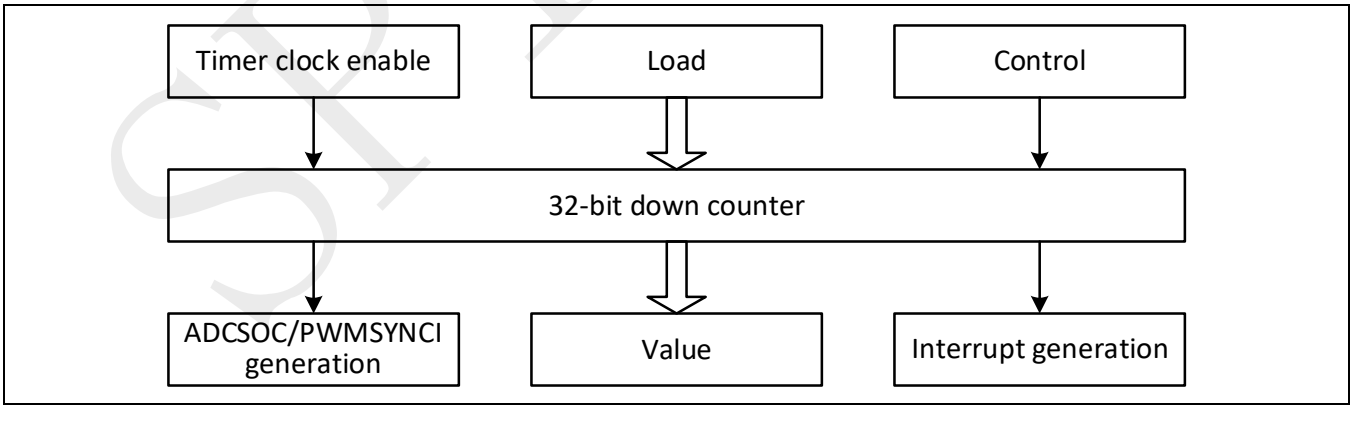
6.3 初始化流程

应当遵照下面的流程来启动定时器：

- 关闭定时器，并清除定时器中断；
- 配置定时器加载值寄存器（TMRLOAD）；
- 配置定时器控制寄存器（TMRCTL）；
- 使能定时器开始计数。

定时器运行运行流程框图如图 6-6 所示。

图 6-6：通用定时器运行流程图



SPINTROL

6.4 寄存器

6.4.1 TIMER 寄存器列表

表 6-1：Timer 模块基地址

外设模块	基地址
TIMER0	0x4000 1800
TIMER1	0x4000 1900
TIMER2	0x4000 1A00

表 6-2：TIMER 寄存器列表

寄存器	偏移地址	描述	复位值
TMRLOAD*	0x00	定时器加载值寄存器	0x00000000
TMRCNT*	0x04	定时器计数值寄存器	0x00000000
TMRCTL*	0x08	定时器控制寄存器	0x00000000
TMRRAWIF	0x0C	定时器中断原始事件标志寄存器	0x00000000
TMRIF	0x10	定时器中断标志寄存器	0x00000000
TMRIC*	0x14	定时器中断清除寄存器	0x00000000
TMRIE*	0x18	定时器中断使能寄存器	0x00000000
TMRREGKEY	0x1C	定时器模块写使能寄存器	0x1ACCE551

注意： 由*标记的寄存器仅当 PWRREGKEY=0x1ACCE551 时才可以改写。

©2025 旋智电子科技（上海）有限公司

6.4.2 TIMER 寄存器

表 6-3：定时器加载值寄存器（TMRLOAD）位段定义

TMRLOAD (Timer Load Register) Access：TIMERx->TMRLOAD Offset：0x00 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 6-4：定时器加载值寄存器（TMRLOAD）位段描述

寄存器	偏移地址	描述	复位值	寄存器
31:0	VAL	RW	0x0	计数器超时加载值 向本寄存器写入数值时，也会设置 TMRCNT 寄存器

表 6-5：定时器计数器值寄存器（TMRCNT）位段定义

TMRCNT (Timer Counter Value Register) Access：TIMERx->TMRCNT Offset：0x04 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 6-6：定时器计数器值寄存器（TMRCNT）位段描述

寄存器	偏移地址	描述	复位值	寄存器
31:0	VAL	RW	0x0	向本寄存器写入数值时，会设置当前计数值。 对于事件捕获模式，寄存器回读值为外部引脚输入上检测到有效边沿时锁存的计数值； 对于其他模式，寄存器回读值为当前计数值。

SPINTROL

表 6-7：定时器控制寄存器（TMRCTL）位段定义

TMRCTL (Timer Control Register) Access：TIMERx->TMRCTL Offset：0x08 Default：0x00000000							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED		EXTSEL					
7	6	5	4	3	2	1	0
EXTSEL	EXTPOL	PWMSYNCEN	ADCSOCEN	HOLD	MODE		EN

表 6-8：定时器控制寄存器（TMRCTL）位段描述

寄存器	偏移地址	描述	复位值	寄存器
31:14	RESERVED_31_14	RO	0x0	保留
13:7	EXTSEL	RW	0x0	外部输入源选择（GPIO 编号）
6	EXTPOL	RW	0x0	外部输入极性 0：低有效 门控定时模式：输入为低时使能计数器； 事件计数/捕获模式：输入下降沿触发事件 1：高有效 门控定时模式：输入为高时使能计数器； 事件计数/捕获模式：输入上升沿触发事件
5	PWMSYNCEN	RW	0x0	使能 PWMSYNCI 信号产生 0：不产生 PWMSYNCI 信号 1：当 TMRCNT 向下计数到 0 时产生 PWMSYNCI
4	ADCSOCEN	RW	0x0	使能 ADCSOC 信号产生 0：不产生 ADCSOC 信号 1：当 TMRCNT 向下计数到 0 时产生 ADCSOC
3	HOLD	RW	0x0	计数值保持 0：TMRCTL.EN 为低时复位计数值 1：TMRCTL.EN 为低时保持计数值
2:1	MODE	RW	0x0	工作模式 0：通用定时（周期性向下计数） 1：门控定时（由外部输入电平使能向下计数）

©2025 旋智电子科技（上海）有限公司

寄存器	偏移地址	描述	复位值	寄存器
				2：事件计数（由外部输入沿触发向下计数） 3：事件捕获（锁存外部输入沿时的计数值）
0	EN	RW	0x0	定时器使能 0：关闭定时器 1：使能定时器

表 6-9：定时器中断原始事件标志寄存器（TMRRAWIF）位段定义

TMRRAWIF (Timer Interrupt Raw Flag Register) Access：TIMERx->TMRRAWIF Offset：0x0C Default：0x00000000							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							INT

表 6-10：定时器中断原始事件标志寄存器（TMRRAWIF）位段描述

寄存器	偏移地址	描述	复位值	寄存器
31:1	RESERVED_31_1	RO	0x0	保留
0	INT	RO	0x0	锁存的定时器超时事件标志 本位段在 TMRCNT 向下计数到 0 时被置 1；且只能通过向 TMRIC 寄存器写 1 清除。 0：超时事件未发生 1：超时事件已发生

表 6-11：定时器中断标志寄存器（TMRIF）位段定义

TMRIF (Timer Interrupt Flag Register) Access：TIMERx->TMRIF Offset：0x10 Default：0x00000000							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							INT

©2025 旋智电子科技（上海）有限公司

表 6-12：定时器中断标志寄存器（TMRIF）位段描述

寄存器	偏移地址	描述	复位值	寄存器
31:1	RESERVED_31_1	RO	0x0	保留
0	INT	RO	0x0	锁存的定时器中断标志 0：中断未发生 1：中断已产生，并已经通知 CPU 本位段被清除之前不会产生新中断

表 6-13：定时器中断清除寄存器（TMRIC）位段定义

TMRIC (Timer Interrupt Clear Register)
Access：TIMERx->TMRIC Offset：0x14 Default：0x00000000

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							INT

表 6-14：定时器中断清除寄存器（TMRIC）位段描述

寄存器	偏移地址	描述	复位值	寄存器
31:1	RESERVED_31_1	RO	0x0	保留
0	INT	W1C	0x0	定时器中断清除 0：写 0 无效，读总是 0 1：写 1 清除 TMRRAWIF 和 TMRIF 标志 本位段自动清零

表 6-15：定时器中断使能寄存器（TMRIE）位段定义

TMRIE (Timer Interrupt Enable Register)
Access：TIMERx->TMRIE Offset：0x18 Default：0x00000000

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							INT

©2025 旋智电子科技（上海）有限公司

表 6-16：定时器中断使能寄存器（TMRIE）位段描述

寄存器	偏移地址	描述	复位值	寄存器
31:1	RESERVED_31_1	RO	0x0	保留
0	INT	RW	0x0	定时器中断使能 本位段不影响 TMRRAWIF 以及 PWMSYNCI 和 ADCSOC 的产生 0：不产生中断 1：当 TMRCNT=0 时产生中断

表 6-17：定时器模块写使能寄存器（TMRREGGKEY）位段定义

TMRREGKEY (Timer Register Write-Allow Key Register) Access：TIMERx->TMRREGKEY Offset：0x1C Default：0x1ACCE551							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

表 6-18：定时器模块写使能寄存器（TMRREGGKEY）位段描述

寄存器	偏移地址	描述	复位值	寄存器
31:0	KEY	RW	0x1ACCE551	写入 0x1ACCE551 解锁受保护的 TIMER 寄存器

SPINTROL

7 看门狗定时器（WDT）

7.1 概述

看门狗定时器（WDT）可以在系统运行失败（由于软件错误）的情况下，重新恢复系统的控制，以增加应用的可靠性。WDT 在计时到一个给定的超时数值时，可以产生系统复位或者一个中断。SPC1169 配备了 2 个 32 位看门狗定时器。

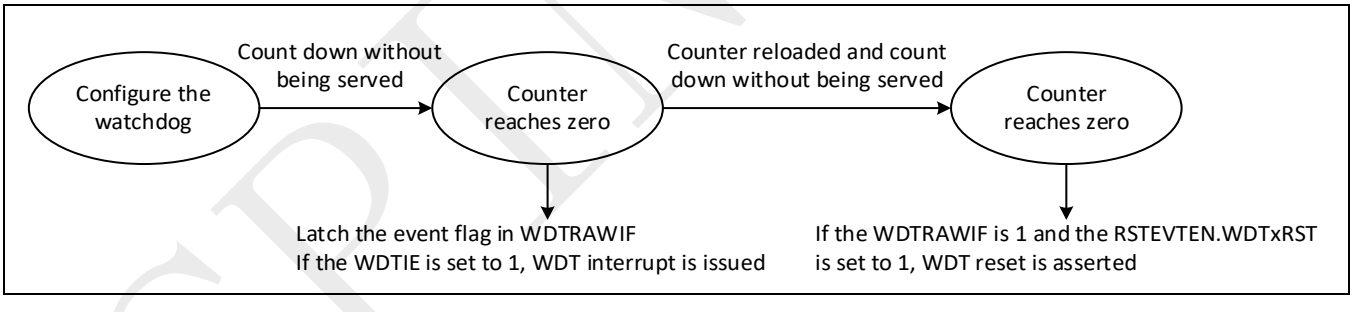
每一个看门狗定时器都有如下特性：

- AHB 总线寄存器接口；
- 专属时钟，通过相应的 WDTxCLKCTL 寄存器配置；
- 32 位向下计数器，在可配置的计数窗口内允许刷新计数值；
- 当发生超时事件时，可配置产生复位或者中断；

7.2 功能描述

如图 7-1 所示，通过置位 WDTCTL.EN 位段来使能 WDT 后，计数器从 WDTLOAD 寄存器预设的超时值开始递减计数。每当计数到 0 时，将重新装载 WDTLOAD 值并再次向下计数。与此同时，超时事件会被锁存到 WDTRAWIF 寄存器中。如果 WDTIE 寄存器为 1，还会产生一个中断，且锁存中断标志到 WDTIF 寄存器。如果使能了复位功能且再次计数到 0 时 WDTRAWIF 标志未被清除，则会产生一个复位请求。

图 7-1：看门狗工作流程



当向 WDTLOAD 写入新的数值时，计数器将会立即从新的数值重新开始计数。当向 WDTIC 写入 1 时，WDTRAWIF 中锁存的超时事件标志和 WDTIF 中锁存的中断标志都会被清除。

仅当计数值处于[WDTTHL, WDTTHH]这个窗口内时，才可以通过向 WDTIC 写 1 来服务（刷新）看门狗定时器，此时计数值将被复位为 WDTLOAD。

可以通过 WDTCTL 寄存器，配置看门狗定时器在 CPU 处于 halted 或 lockup 时是否关闭。

出于安全方面的考虑，为了防止软件意外操作致使看门狗关闭，关键的控制寄存器均有写保护。用户在配置看门狗定时器前，需要先向 WDTREGKEY 寄存器写入 0x1ACCE551，才可以对其他寄存器进行写访问。配置完后，需要向 WDTREGKEY 寄存器写入任何其它数值以重新开启写保护。

注意： WDT0 的超时事件被用作 NMI 中断。

©2025 旋智电子科技（上海）有限公司

7.3 初始化流程

需要按照下面的操作流程来启动看门狗定时器：

- 向 WDTREGKEY 写入 0x1ACCE551，以使能其他寄存器的写访问；
- 通过 WDTLOAD 寄存器设置超时值；
- 通过 WDTTHL 和 WDTTHH 寄存器配置服务窗口；
- 通过 SYSTEM->RSTEVTEN.WDTxRST 位段，配置是否触发复位；
- 通过 WDTIE 配置是否触发中断；
- 通过向 WDTCTL.EN 位段写 1 启动定时器计数；
- 向 WDTREGKEY 写入 0x0（任何不为 0x1ACCE551 的值），禁止对其他寄存器的写访问。

SPINTROL

7.4 寄存器

7.4.1 WDT 寄存器列表

表 7-1：WDT 模块基地址

外设模块	基地址
WDT0	0x4000 1000
WDT1	0x4000 1100

表 7-2：WDT 寄存器列表

寄存器	偏移地址	描述	复位值
WDTLOAD*	0x00	看门狗定时器加载值寄存器	0x03D09000
WDTCNT	0x04	看门狗定时器计数值寄存器	0x03D09000
WDTCTL*	0x08	看门狗定时器控制寄存器	0x00000007
WDTRAWIF	0x0C	看门狗定时器中断原始事件标志寄存器	0x00000000
WDTIF	0x10	看门狗定时器中断标志寄存器	0x00000000
WDTIC	0x14	看门狗定时器中断清除寄存器	0x00000000
WDTIE*	0x18	看门狗定时器中断使能寄存器	0x00000001
WDTTHL*	0x1C	看门狗定时器服务窗口下边界寄存器	0x00000000
WDTTHH*	0x20	看门狗定时器服务窗口上边界寄存器	0xFFFFFFFF
WDTREGKEY	0x24	看门狗定时器模块写使能寄存器	0x00000000

注意： 由*标记的寄存器仅当 PWRREGKEY=0x1ACCE551 时才可以改写。

©2025 旋智电子科技（上海）有限公司

7.4.2 WDT 寄存器

表 7-3：看门狗定时器加载值寄存器（WDTLOAD）位段定义

WDTLOAD (Watchdog Timer Load Register) Access：WDTx->WDTLOAD Offset：0x00 Default：0x03D09000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 7-4：看门狗定时器加载值寄存器（WDTLOAD）位段描述

寄存器	偏移地址	描述	复位值	寄存器
31:0	VAL	RW	0x3D09000	超时加载值（最小有效值为 1） 计数器从该加载值递减。当本寄存器被写入新的数值时，计数器将会立即从新的数值重新开始向下计数。

表 7-5：看门狗定时器计数值寄存器（WDTCNT）位段定义

WDTCNT (Watchdog Timer Current Value Register) Access：WDTx->WDTCNT Offset：0x04 Default：0x03D09000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 7-6：看门狗定时器计数值寄存器（WDTCNT）位段描述

寄存器	偏移地址	描述	复位值	寄存器
31:0	VAL	RO	0x3D09000	递减计数器当前计数值

SPINTROL

表 7-7：看门狗定时器控制寄存器（WDTCTL）位段定义

WDTCTL (Watchdog Timer Control Register) Access：WDTx->WDTCTL Offset：0x08 Default：0x00000007							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED					LOCKUPRUN	HALTEDRUN	EN

表 7-8：看门狗定时器控制寄存器（WDTCTL）位段描述

寄存器	偏移地址	描述	复位值	寄存器
31:3	RESERVED_31_3	RO	0x0	保留
2	LOCKUPRUN	RW	0x1	CPU 处于 lockup 状态时看门狗定时器是否运行 0：关闭看门狗定时器并复位计数值为 WDTLOAD 1：根据 EN 位段决定是否使能看门狗定时器
1	HALTEDRUN	RW	0x1	CPU 处于 halted 状态时看门狗定时器是否运行 0：关闭看门狗定时器并复位计数值为 WDTLOAD 1：根据 EN 位段决定是否使能看门狗定时器
0	EN	RW	0x1	使能看门狗定时器 0：关闭定时器并复位计数值为 WDTLOAD 1：使能定时器

表 7-9：看门狗定时器中断原始事件标志寄存器（WDTRAWIF）位段定义

WDTRAWIF (Watchdog Timer Interrupt Raw Flag Register) Access：WDTx->WDTRAWIF Offset：0x0C Default：0x00000000							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							INT

©2025 旋智电子科技（上海）有限公司

表 7-10：看门狗定时器中断原始事件标志寄存器（WDTRAWIF）位段描述

寄存器	偏移地址	描述	复位值	寄存器
31:1	RESERVED_31_1	RO	0x0	保留
0	INT	RO	0x0	超时（计数到 0）事件标志 0：超时事件未发生 1：超时事件已发生

表 7-11：看门狗定时器中断标志寄存器（WDTIF）位段定义

WDTIF (Watchdog Timer Interrupt Flag Register) Access：WDTx->WDTIF Offset：0x10 Default：0x00000000							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							INT

表 7-12：看门狗定时器中断标志寄存器（WDTIF）位段描述

寄存器	偏移地址	描述	复位值	寄存器
31:1	RESERVED_31_1	RO	0x0	保留
0	INT	RO	0x0	中断标志 0：中断未发生 1：中断已发生

表 7-13：看门狗定时器中断清除寄存器（WDTIC）位段定义

WDTIC (Watchdog Timer Interrupt Clear Register) Access：WDTx->WDTIC Offset：0x14 Default：0x00000000							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							INT

©2025 旋智电子科技（上海）有限公司

表 7-14：看门狗定时器中断清除寄存器（WDTIC）位段描述

寄存器	偏移地址	描述	复位值	寄存器
31:1	RESERVED_31_1	RO	0x0	保留
0	INT	W1C	0x0	Interrupt clear. The WDTCNT will be reset to WDTLOAD if it is within [WDTTHL, WDTTHH] when this bit is written a 1. 0：Write a 0 has no effect and always read back 0. 1：Write a 1 clears both the WDTRAWIF and WDTIF. This bit is self-cleared to 0.

表 7-15：看门狗定时器中断使能寄存器（WDTIE）位段定义

WDTIE (Watchdog Timer Interrupt Enable Register) Access：WDTx->WDTIE Offset：0x18 Default：0x00000001							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							INT

表 7-16：看门狗定时器中断使能寄存器（WDTIE）位段描述

寄存器	偏移地址	描述	复位值	寄存器
31:1	RESERVED_31_1	RO	0x0	保留
0	INT	RW	0x1	中断使能 本位段不影响 WDTRAWIF 的锁存 0：不产生中断 1：计数到 0 时产生中断

©2025 旋智电子科技（上海）有限公司

表 7-17：看门狗定时器服务窗口低阈值寄存器（WDTTHL）位段定义

WDTTHL (Watchdog Windowed-Feed Low Threshold Register) Access：WDTx->WDTTHL Offset：0x1C Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 7-18：看门狗定时器服务窗口低阈值寄存器（WDTTHL）位段描述

寄存器	偏移地址	描述	复位值	寄存器
31:0	VAL	RW	0x0	仅当 WDTCNT 处于[WDTTHL, WDTTHH]窗口内时可以通过向 WDTIC 写 1 来服务看门狗定时器（重新加载计数值为 WDTLOAD）

表 7-19：看门狗定时器服务窗口高阈值寄存器（WDTTHL）位段定义

WDTTHH (Watchdog Windowed-Feed High Threshold Register) Access：WDTx->WDTTHH Offset：0x20 Default：0xFFFFFFFF							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 7-20：看门狗定时器服务窗口高阈值寄存器（WDTTHL）位段描述

寄存器	偏移地址	描述	复位值	寄存器
31:0	VAL	RW	0xFFFFFFFF	仅当 WDTCNT 处于[WDTTHL, WDTTHH]窗口内时可以通过向 WDTIC 写 1 来服务看门狗定时器（重新加载计数值为 WDTLOAD）

©2025 旋智电子科技（上海）有限公司

表 7-21：看门狗定时器模块写使能寄存器（WDTREGKEY）位段定义

WDTREGKEY (Watchdog Register Write-Allow Key Register) Access：WDTx->WDTREGKEY			Offset：0x24		Default：0x00000000		
31	30	29	28	27	26	25	24
			KEY				
23	22	21	20	19	18	17	16
			KEY				
15	14	13	12	11	10	9	8
			KEY				
7	6	5	4	3	2	1	0
			KEY				

表 7-22：看门狗定时器模块写使能寄存器（WDTREGKEY）位段描述

寄存器	偏移地址	描述	复位值	寄存器
31:0	KEY	RW	0x0	写入 0x1ACCE551 解锁受保护的看门狗定时器寄存器

SPINTROL

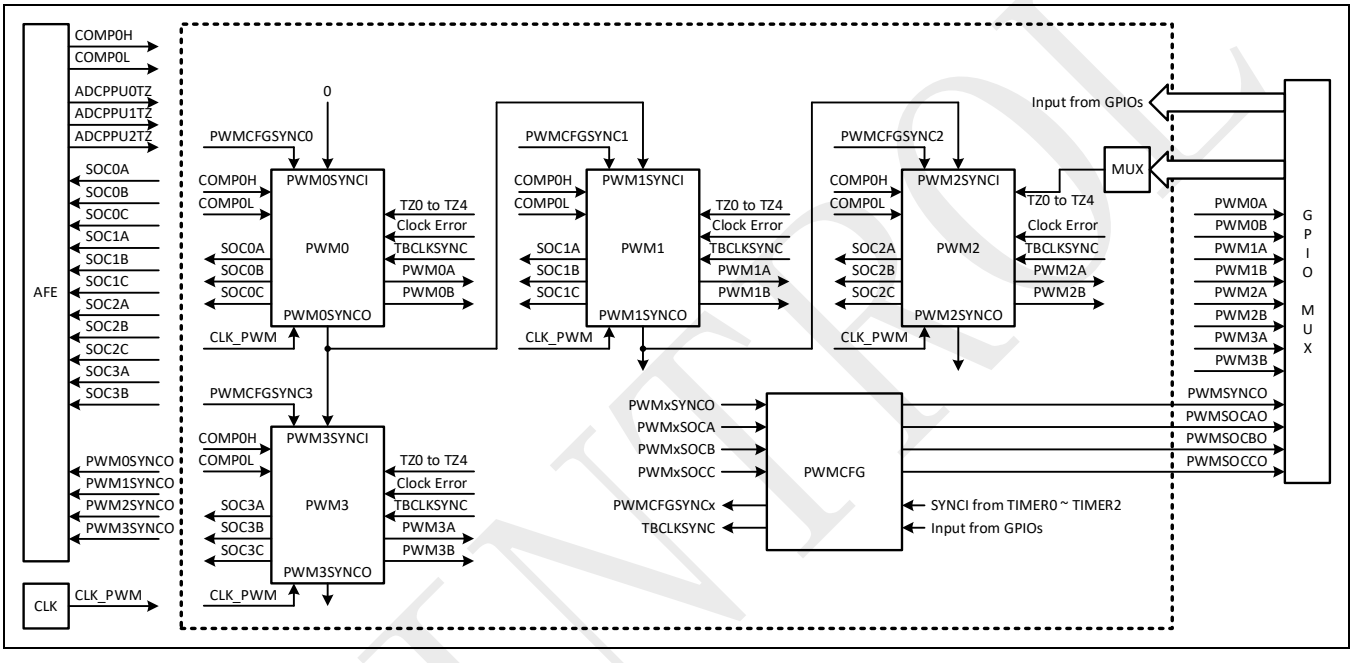
©2025 旋智电子科技（上海）有限公司

8 脉宽调制模块（PWM）

8.1 概述

PWM（脉冲宽度调制）在电力电子系统中扮演着重要的角色，例如数字电机控制、开关电源控制和不间断电源（UPS）控制。SPC1169 提供了 4 个 PWM 模块，每个 PWM 模块都有两个输出：PWMxA 和 PWMxB。它们通过一个时钟同步方案进行链式连接，如图 8-1 所示。

图 8-1：PWM 系统总览



每个 PWM 模块均支持如下特性：

- 周期可配置的 16 位时基计数；
- 软件直接强制改变 PWM 输出；
- 独立可配的相位控制；
- 周期性的相位同步；
- 上升沿延迟和下降沿延迟独立可配置的死区控制；
- 异常事件下可配的周期性或者一次性封锁；
- 封锁时可以配置 PWM 输出为高、低或高阻；
- 数字比较或者封锁信号输入可产生原始封锁事件或者过滤后的封锁事件；
- 所有事件可以用于产生 CPU 中断或者 ADC SOC（开始采样）信号。

SPINTROL

图 8-2：PWM 内部结构

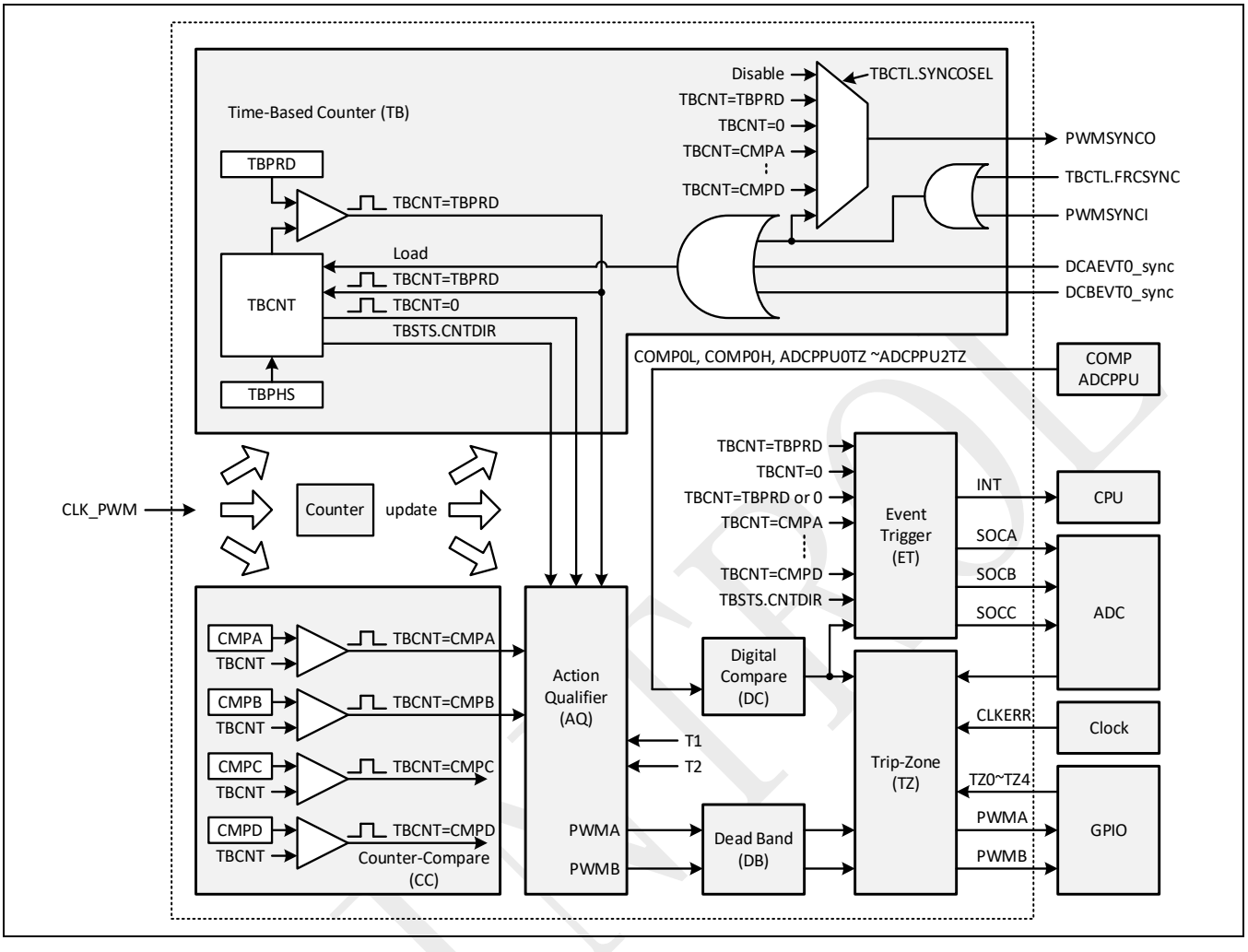


图 8-2 展示了单个 PWM 模块的更多内部细节。PWM 模块的关键信号包括：

- 同步输入/输出（PWMSYNCI 和 PWMSYNCO）
  用于同步多个 PWM 的相位关系。在 SPC1169 中，PWMSYNCI 可以来自 GPIO、定时器或前一个 PWM 的 PWMSYNCO。

注意： 从同步信号输入到同步操作真正完成需要一个 PWM 时钟，因此当 TBCTL 寄存器的 TBPDIV 和 TBDIV 位段同时为 0（即不分频）时，会有一拍时基时钟的延时。用户在设置 TBPHS 寄存器时需要考虑这个因素。
PWMCFG 模块中的 FRCSYNC 等全局同步控制提供了更加灵活便捷的跨 PWM 同步方式。

©2025 旋智电子科技（上海）有限公司

- 封锁信号
  每个 PWM 有 5 个配置为来自 GPIO 的封锁信号（TZ0~TZ4）用于警示外部异常，另外来自 CLKDET 时钟交叉检测模块或者 PLL 的时钟错误事件、CPU 的 lockup 和 halted 事件也可以配置 PWM 封锁信号。

注意： 来自 GPIO 的封锁信号极性可以配置，默认当 GPIO 为高时触发 PWM 封锁。

- 比较器结果
  来自 2 个模拟比较器的结果可以产生数字比较事件。
- ADC SOC 信号（PWMxSOCA、PWMxSOCB 和 PWMxSOCC）
  每个 PWM 有 3 个 ADC SOC 信号。触发 ADC SOC 信号的事件在事件触发（Event Trigger）子模块中配置。
- PWM 输出（PWMxA 和 PWMxB）

  PWM 输出配置为与其他功能复用，如表 4-1 所示。

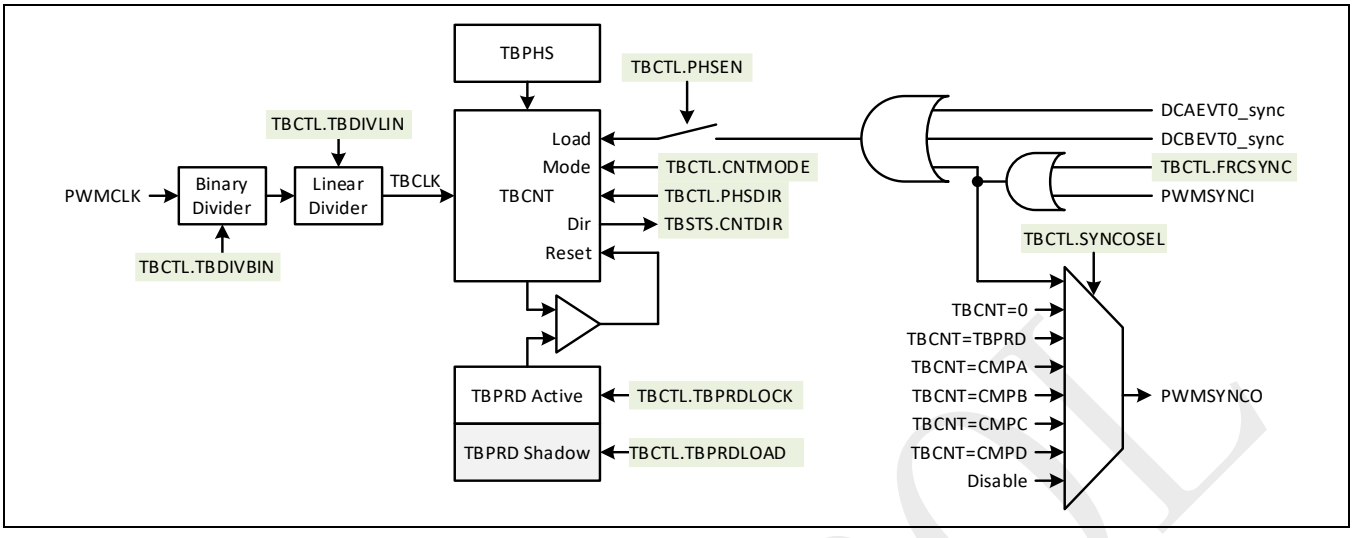
8.2 时基产生（TB）子模块

时基产生子模块负责 PWM 的时序控制，其关键信号如图 8-3 所示。该子模块有如下配置项：

- 设置时基计数时钟相对于 PWM 输入时钟的分频比；
- 设置时基计数器（TBCNT）模式：
  - 向上计数
  - 向下计数
  - 上下计数
- 设置时基计数器（TBCNT）周期；
- 设置相对于其他 PWM 的相位；
- 产生 TBCNT=TBPRD 和 TBCNT=0 事件；
- 设置同步触发事件以及事件发生时计数器采取的操作。

SPINTROL

图 8-3：PWM 时基产生子模块信号链



PWM 计数波形周期由计数模式（TBCTL.CNTMODE）和周期（TBPRD）寄存器共同决定。图 8-4 展示了向上计数、向下计数和上下计数三种情况下波形周期和 TBPRD 寄存器的关系。

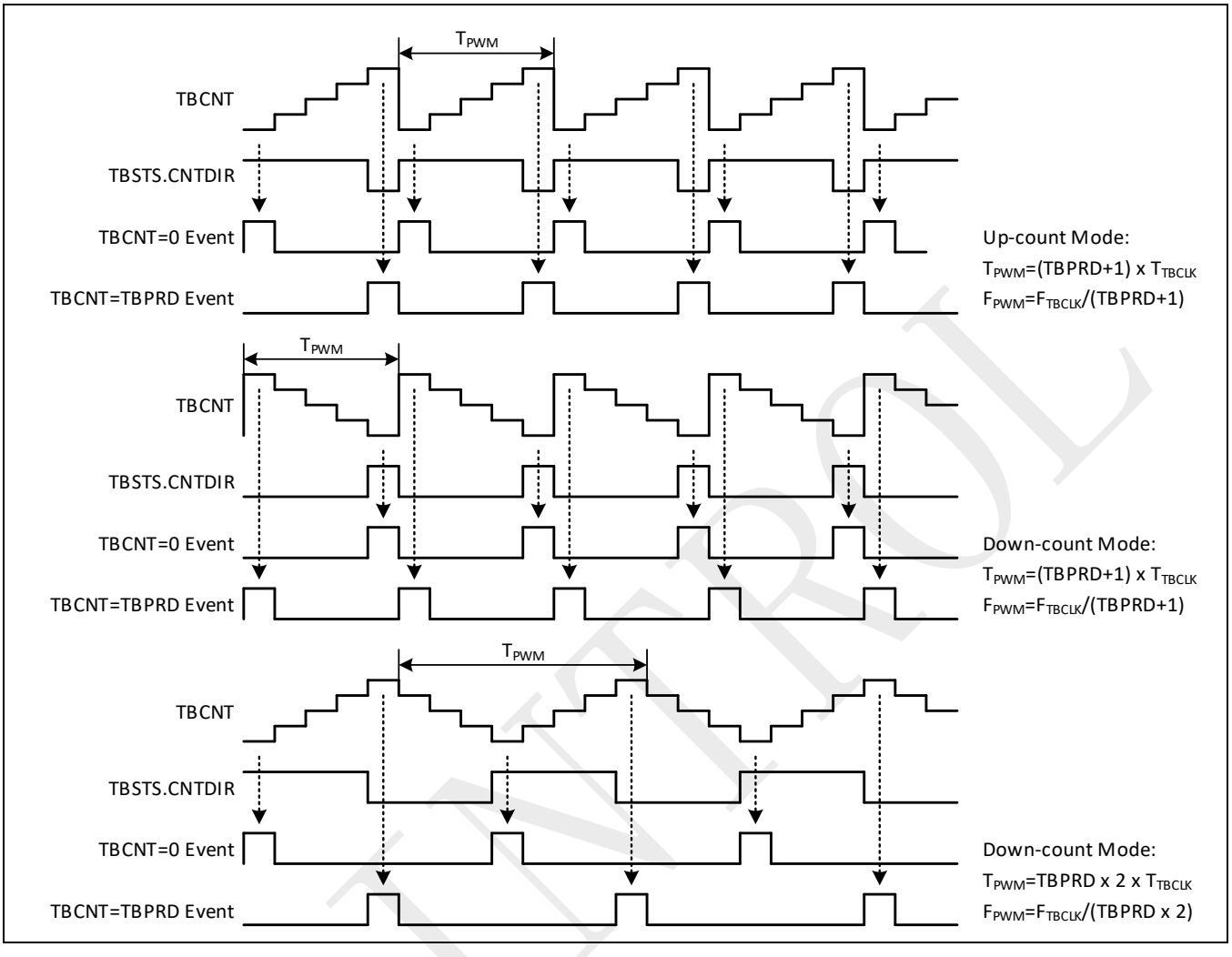
向上计数模式时，时基计数器从 0 开始一直递增计数到 TBPRD，然后复位到 0 重新开始递增计数。

向下计数模式时，时基计数器从 TBPRD 开始一直递减计数到 0，然后复位到 TBPRD 重新开始递减计数。

上下计数模式时，时基计数器从 0 开始递增计数到 TBPRD，然后递减计数回 0，如此周而复始。从 TBSTS 寄存器 CNTDIR 位段读回的值，在从 0 到(TBPRD-1)递增计数时为 1，在从 TBPRD 到 1 递减计数时为 0。

©2025 旋智电子科技（上海）有限公司

图 8-4：时基计数波形

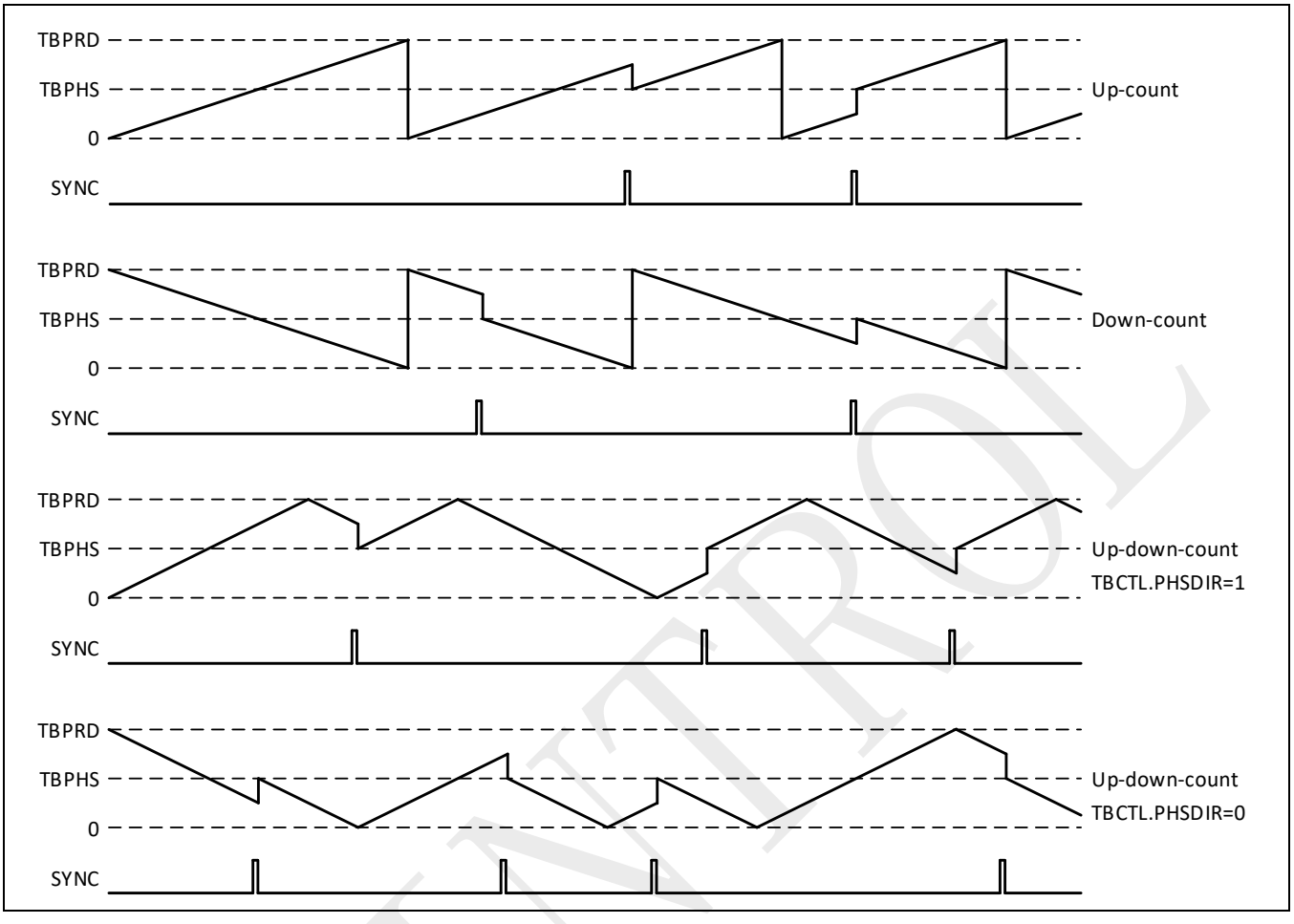


TBPRD 寄存器支持影子模式。在该模式下，对 TBPRD 写操作只会先更新影子寄存器的值而不会立即更新实际值，然后根据 TBCTL.TBPRDLOAD 位段的设置，当 TBCNT=0 或者 TBCNT=TBPRD 时才从影子寄存器中更新。

相位同步功能使得 PWM 之间可以自动实现时基同步，可以通过设置 TBCTL.PHSEN 位段为 1 来开启这个功能。此时，一旦发生同步事件，TBCNT 会被更新为时基相位寄存器（TBPHS）的值。如图 8-5 所示，对于向上计数和向下计数模式，同步前后计数方向保持不变；对于上下计数方式，按照时基控制寄存器（TBCTL）的 PHSDIR 位段来决定同步后的计数方向。此外，如图 8-3 所示，同步事件可以是外部 PWMSYNCI 输入信号、数字比较同步信号 A 和 B、本地软件强制同步和全局软件强制同步。

当 TBCTL.PHSEN = 0 时，PWM 忽略输入的同步事件信号，但这些信号仍然会被送往 PWMSYNCO 输出给后级 PWM 模块用于其同步。

图 8-5：时基相位同步波形



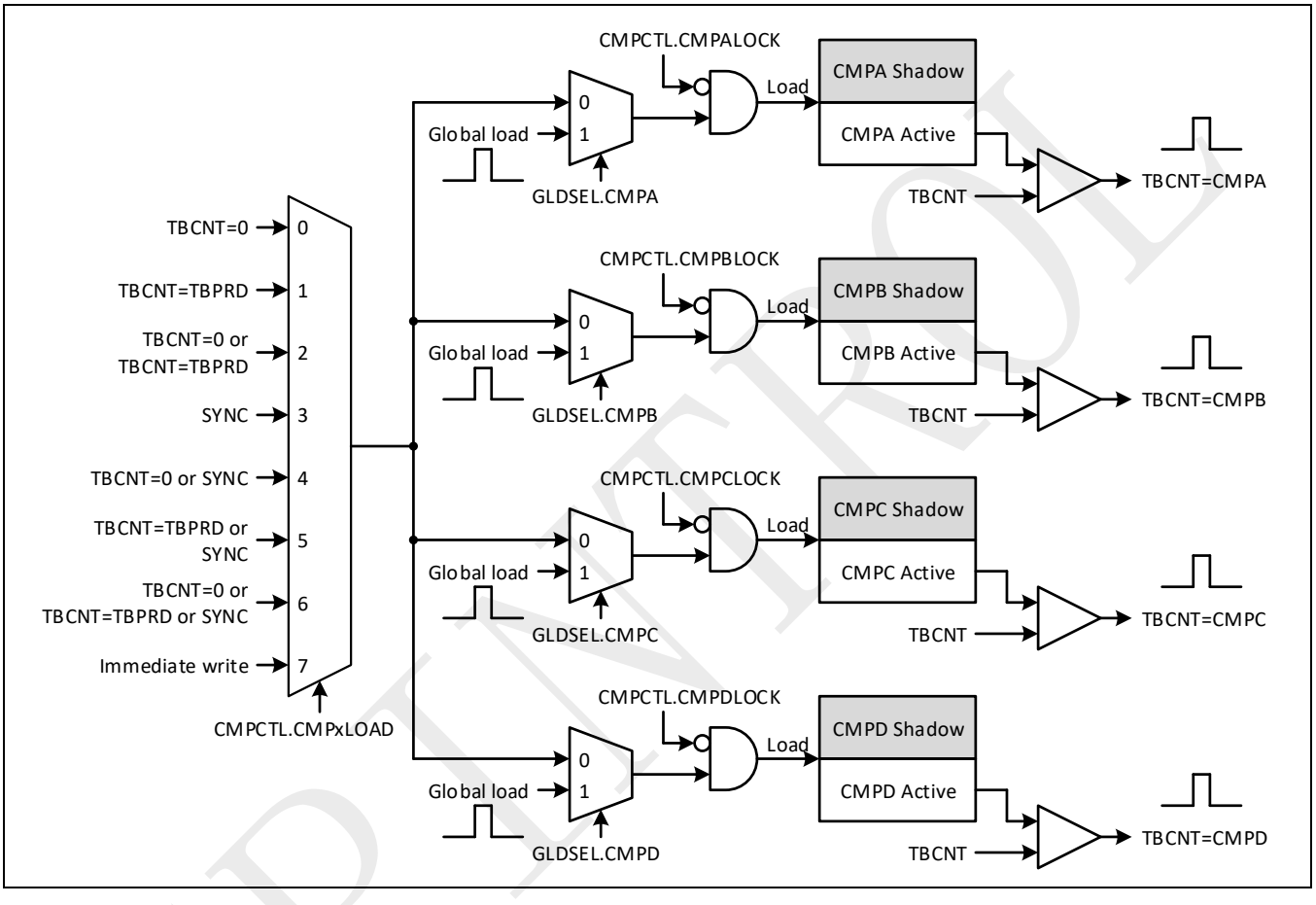
注意： 从同步信号输入到同步操作真正完成需要一个 PWM 时钟，因此当 TBCTL 寄存器的 TBPDIV 和 TBDIV 位段同时为 0（即不分频）时，会有一拍时基时钟的延时。用户在设置 TBPHS 寄存器时需要考虑这个因素。
PWMCFG 模块中的 FRCSYNC 提供了更加灵活便捷的跨 PWM 同步方式。

©2025 旋智电子科技（上海）有限公司

8.3 计数比较（CC）子模块

计数比较子模块持续将 TBCNT 与计数比较寄存器 A（CMPA）、计数比较寄存器 B（CMPB）、计数比较寄存器 C（CMPC）以及计数比较寄存器 D（CMPD）的值进行比对。当 TBCNT 的值与这些寄存器的值相等时，产生相应的事件。

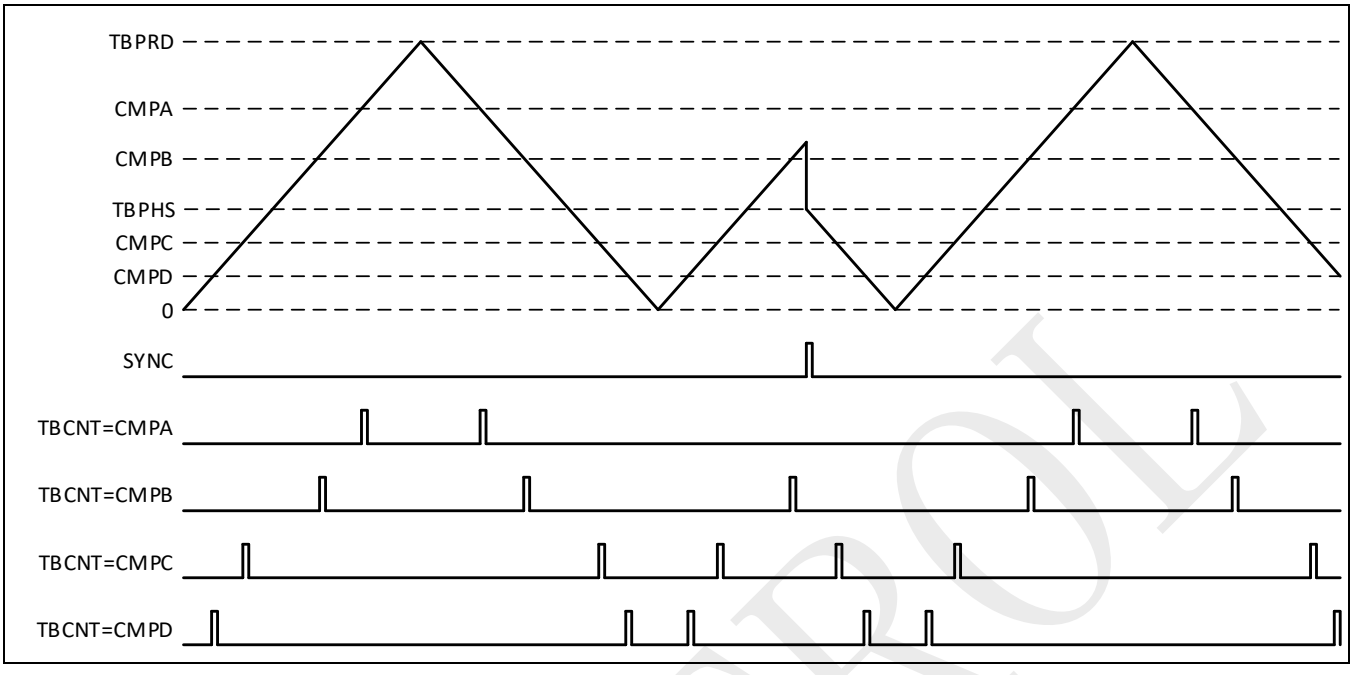
图 8-6：计数比较子模块结构



计数比较子模块的基本结构如图 8-6 所示。计数比较寄存器同样具备影子寄存功能，此时对计数比较寄存器的写操作会先更新影子寄存器的值，实际比较值会根据计数比较控制寄存器（CMPCTL）的 CMPxLOAD 位段定义的时间点更新。也可以按照表 8-14 和表 8-15 所示，通过全局影子到有效值更新时机控制寄存器（GLDSEL）的 CMPx 位段来配置影子功能。图 8-7 给出了计数比较子模块波形示例。

SPINTROL

图 8-7：计数比较子模块波形

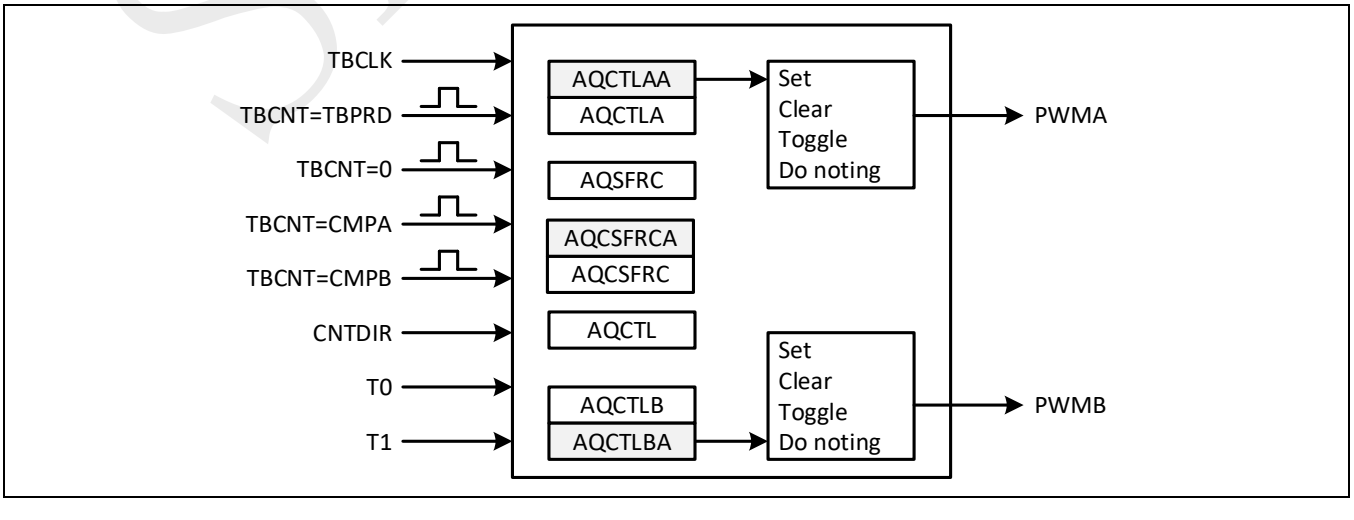


8.4 行为限定（AQ）子模块

行为限定子模块决定了哪些事件会转换为哪种行为并产生相应的 PWMA 和 PWMB 开关波形。如图 8-8 所示，这些事件包括：

- TBCNT=TBPRD
- TBCNT=0
- TBCNT=CMPA
- TBCNT=CMPB
- T0/T1 事件（基于数字比较、封锁和同步输入事件产生）
- 软件强制事件

图 8-8：行为限定子模块输入输出



©2025 旋智电子科技（上海）有限公司

行为限定模块管理这些事件同时发生时的响应优先级，并基于 AQCTLA 和 AQCTLB 寄存器的配置在对应事件发生时做出如下操作：

- 输出置高
- 输出置低
- 输出翻转
- 不操作

可以通过 AQCTL.AQCTLALOAD、AQCTL.AQCTLBLOAD 和 AQSFRC.CSFLOAD 寄存器位段来选择影子寄存模式或是直接更新模式。也可以采用如图 8-6 所示 CMPx 寄存器类似的方法，通过 GLDSEL.AQCTLA、GLDSEL.AQCTLB 和 GLDSEL.AQCSFRC 寄存器位段配置全局影子寄存。

表 8-1 到表 8-3 列举了不同计数模式下的行为限定事件优先级。

表 8-1：向上计数时的行为限定事件优先级

优先级	事件
1(最高)	软件强制控制
2	TBCNT=TBPRD (PRD)
3	T0 on up-count (T0U)
4	T1 on up-count (T1U)
5	TBCNT=CMPB on up-count (CBU)
6	TBCNT=CMPA on up-count (CAU)
7 (最低)	TBCNT=0 (ZRO)

表 8-2：向下计数时的行为限定事件优先级

优先级	事件
1(最高)	Software forced event
2	TBCNT=0 (ZRO)
3	T0 on down-count (T0D)
4	T1 on down-count (T1D)
5	TBCNT=CMPB on down-count (CBD)
6	TBCNT=CMPA on down-count (CAD)
7 (最低)	TBCNT=TBPRD (PRD)

表 8-3：上下计数时的行为限定事件优先级

优先级	事件	
	从 0 向上计数到 TBPRD-1 时	从 TBPRD 向下计数到 1 时
1(最高)	软件强制控制	软件强制控制
2	T0 on up-count (T0U)	T0 on down-count (T0D)
3	T1 on up-count (T1U)	T1 on down-count (T1D)
4	TBCNT=CMPB on up-count (CBU)	TBCNT=CMPB on down-count (CBD)
5	TBCNT=CMPA on up-count (CAU)	TBCNT=CMPA on down-count (CAD)
6 (最低)	TBCNT=0 (ZRO)	TBCNT=TBPRD (PRD)

©2025 旋智电子科技（上海）有限公司

图 8-9 和图 8-10 展示了行为限定模块产生的对称和非对称 PWM 波形。

图 8-9：上下计数时的 PWM 双边沿对称波形

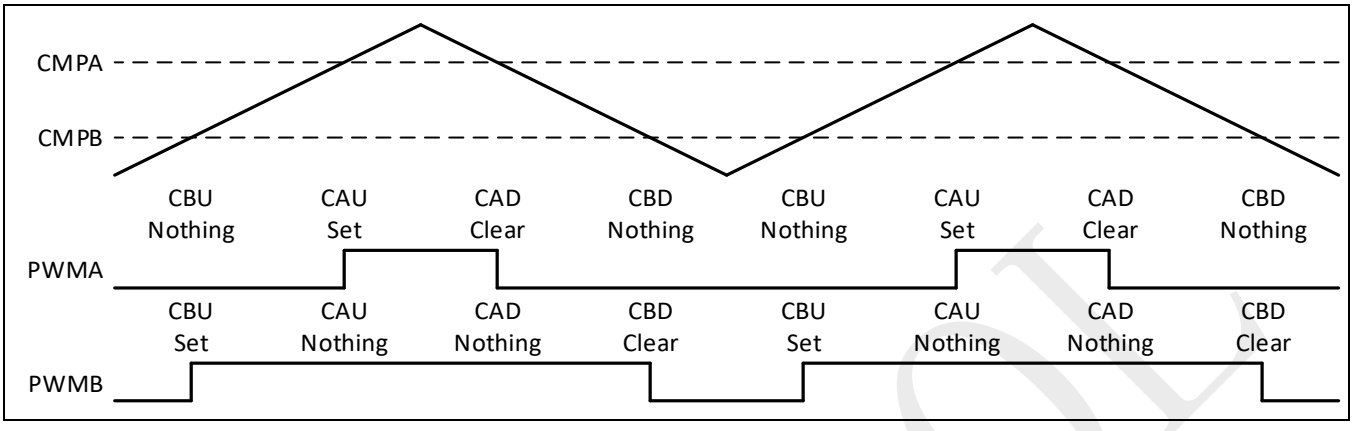
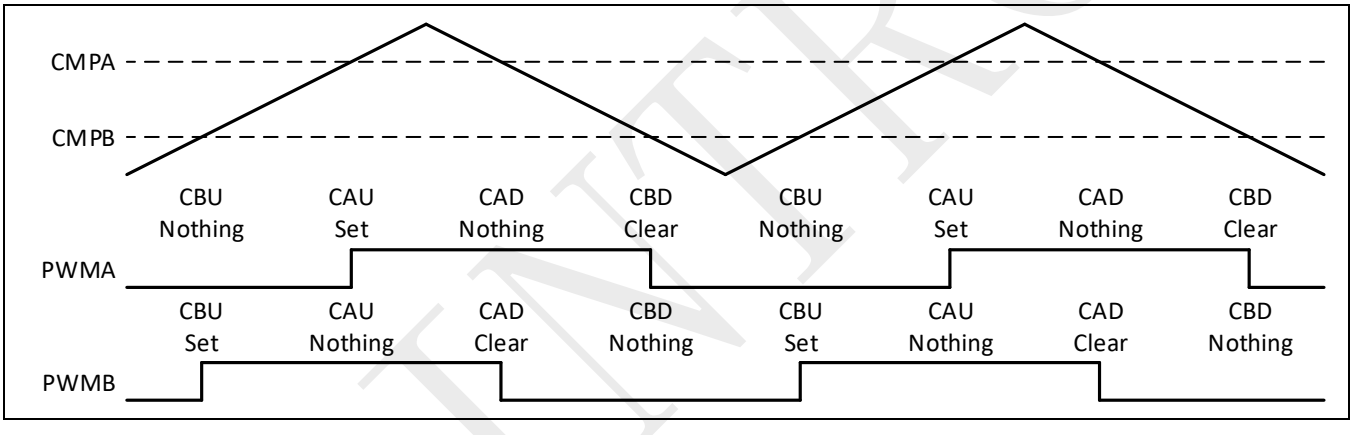


图 8-10：上下计数时的 PWM 双边沿非对称波形



8.5 死区控制（DB）子模块

死区控制子模块将行为限定模块输出的波形做进一步整形，包括对原始波形的上升沿和下降沿做独立的延时以及极性控制。该模块的关键功能包括：

- 基于来自行为限定模块的 PWMA/PWMB 输入产生有相对死区的 PWMA 和 PWMB 信号对；
- 对上升沿加可配置的延迟；
- 对下降沿加可配置的延迟；
- 支持半时钟周期延迟；
- 调整输出信号极性；
- 绕开死区控制信号整形，将行为限定模块输出的波形直接原样输出。

©2025 旋智电子科技（上海）有限公司

图 8-11：死区控制子模块结构

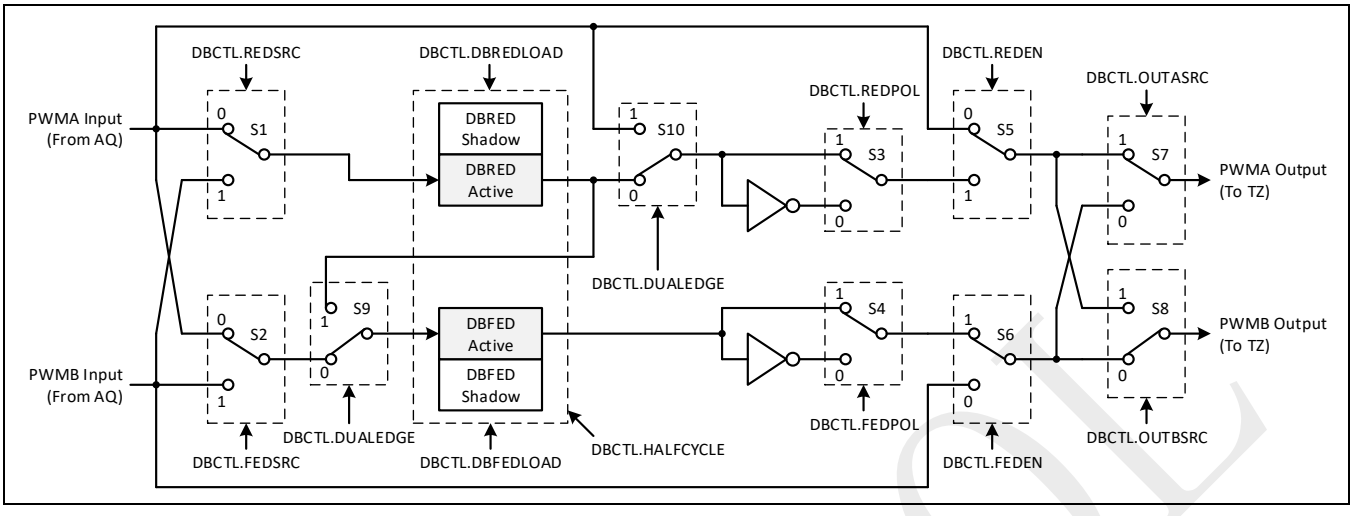


图 8-11 给出了死区控制模块的基本结构。死区控制寄存器（DBCTL）的 REDSRC 和 FEDSRC 位段设定了上升沿延迟通路和下降沿延迟通路的输入信号。随后 S3 到 S10 的各种开关组合可以产生灵活多变的波形。以 DBCTL.REDSRC=0 且 DBCTL.FEDSRC=0 为例，PWMA 被配置为两个延迟通路的输入，典型的死区控制工作模式如表 8-4 所示。

作为一种特殊情形，可以通过设置 DBCTL 寄存器的 DUALEDGE 位段实现双边沿延迟，即对同一路信号完成上升沿和下降沿的同时延迟。这个功能可以便于产生周期和占空比完全相同但存在指定相位差的两路波形。

表 8-4：典型的死区控制工作模式

Mode	Mode 描述	REDPOL S3	FEDPOL S4	REDEN S5	FEDEN S6
1	Both PWMA and PWMB pass through	X	X	0	0
2	Active high complementary	1	0	1	1
3	Active low complementary	0	1	1	1
4	Active high	0	0	1	1
5	Active low	1	1	1	1
6	PWMA output = PWMA input with no delay PWMB output = PWMA input with falling-edge delay	0 or 1	0 or 1	0	1
7	PWMA output = PWMA input with rising-edge delay PWMB output = PWMA input with no delay	0 or 1	0 or 1	1	0

©2025 旋智电子科技（上海）有限公司

图 8-12 示例了死区控制的典型波形。

图 8-12：死区控制典型波形

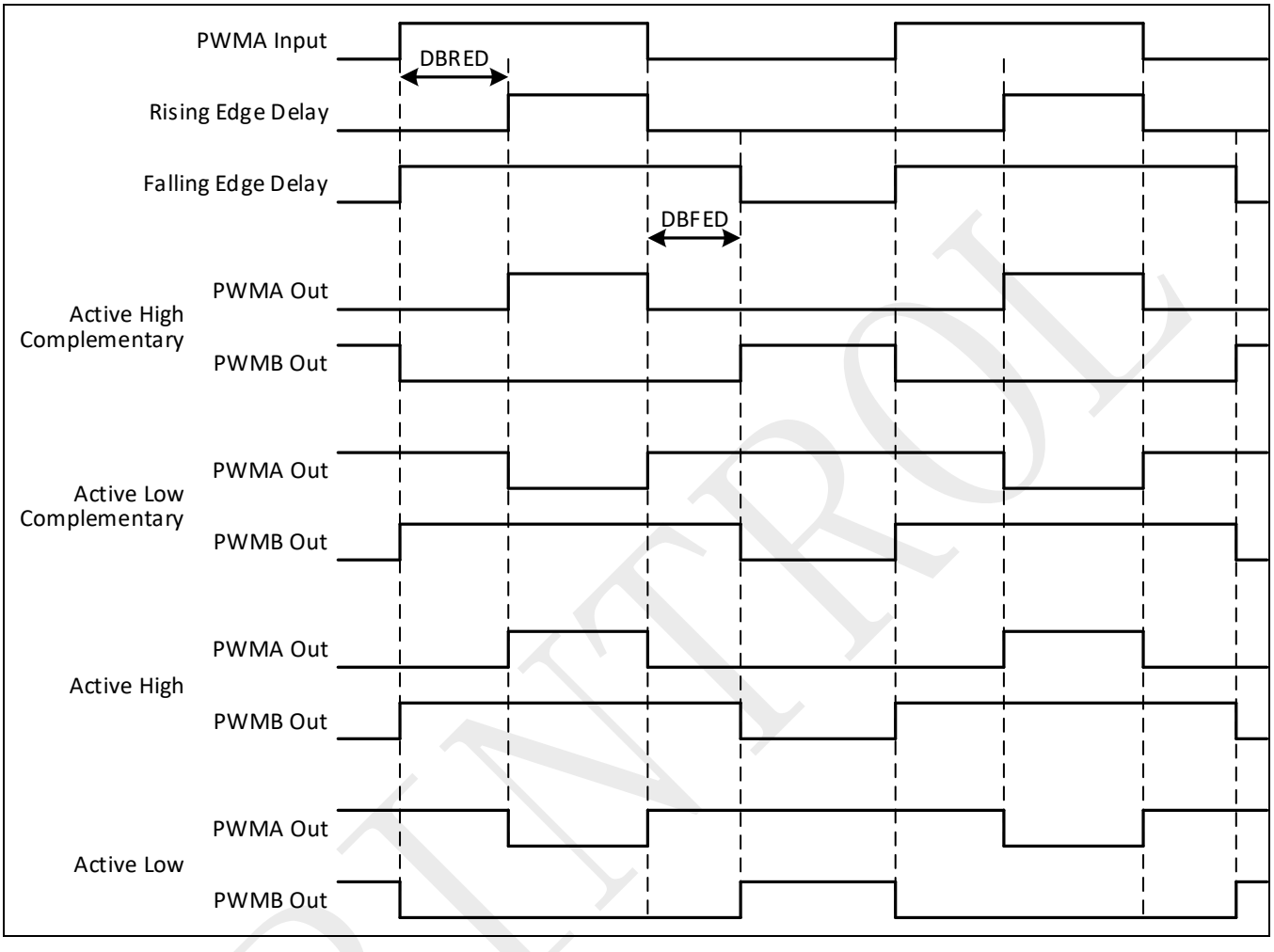
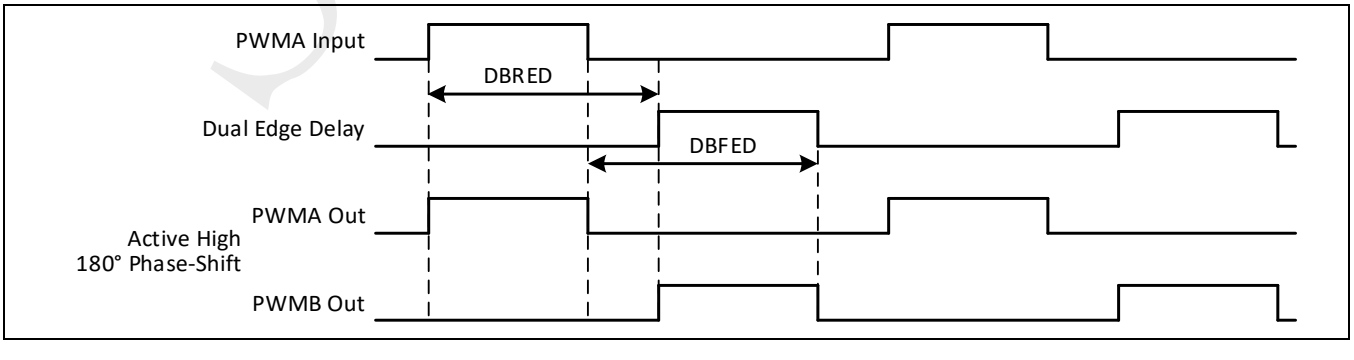


图 8-13 示例了通过使能双边沿延迟并设置 DBRED 和 DBFED 为波形周期的一半来产生 180°相位差的两路波形。

图 8-13：双边沿延迟死区控制波形

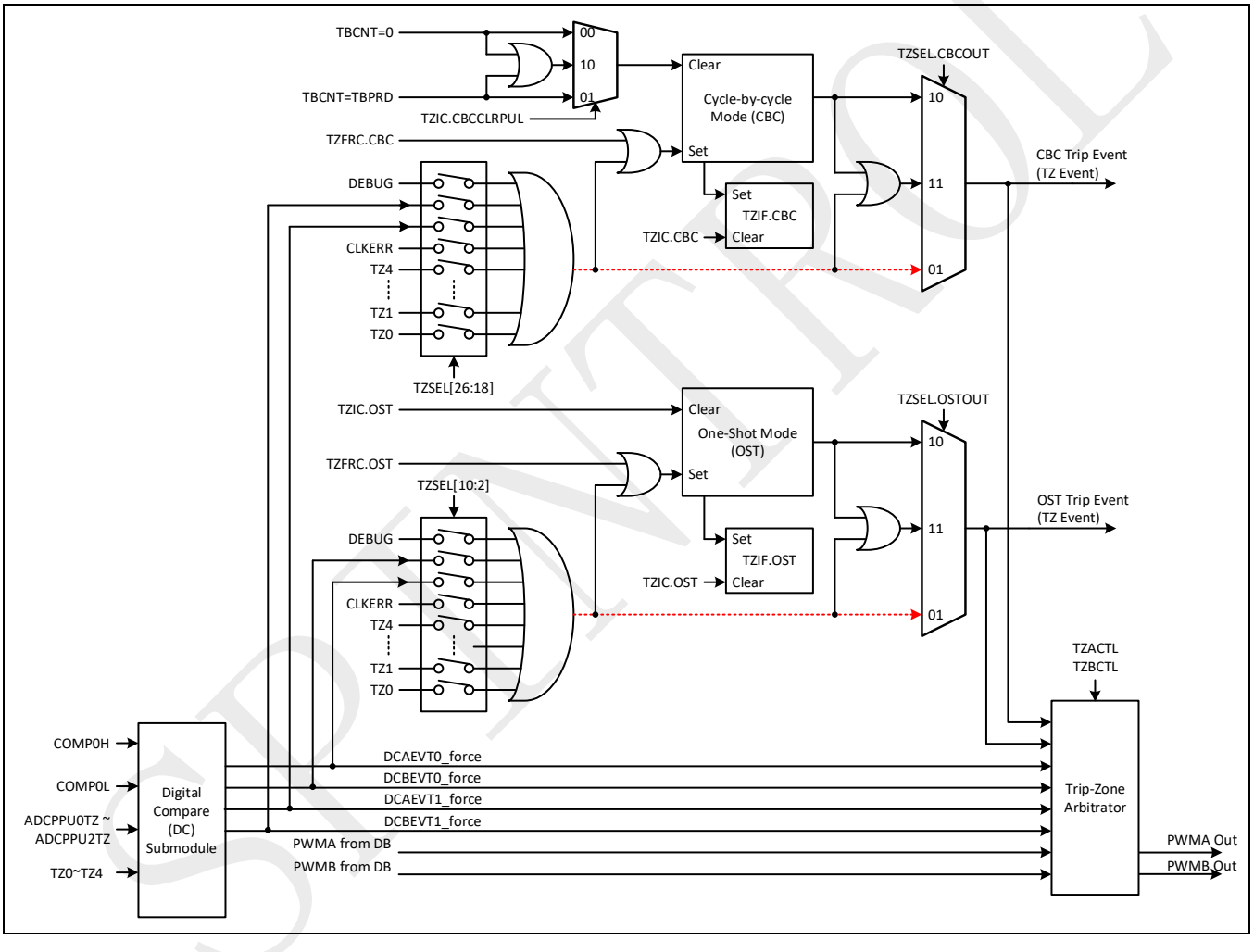


©2025 旋智电子科技（上海）有限公司

8.6 信号封锁（TZ）子模块

信号封锁（trip-zone）子模块处理当外部错误事件或封锁条件发生时的 PWM 输出。如图 8-14 所示，输入事件可以是来自 GPIO 的 TZ 信号、来自 PLL 失锁或者 CLKDET 时钟检测模块的时钟错误信号、CPU 调试信号、以及数字比较（DC）子模块的输出。可以根据 PWMCFG 模块中的 TZ#SRCCTL 寄存器选择 5 个 TZ 信号（TZ0~TZ4）具体对应的 GPIO 以及信号的极性。DCxEVTy 强制信号由下一章节即将讨论的 DC 子模块产生。此外还支持软件触发的封锁，并且可以配置各种封锁事件触发 CPU 中断。

图 8-14：信号封锁子模块结构



封锁事件发生时，PWMA 和 PWMB 输出可以被强制为如下几种情形：

- 置高
- 置低
- 高阻
- 维持原样不变

©2025 旋智电子科技（上海）有限公司

PWMA 输出的事件动作由寄存器 TZACTL 控制，这些事件动作的优先级为 TZU > TZD > DCAEVT0U > DCAEVT0D > DCAEVT1U > DCAEVT1D。这意味着当高优先级事件和低优先级事件同时发生时，输出由高优先级事件的配置决定。例如，当 TZU 和 DCAEVT1U 事件同时发生时，TZU 寄存器字段决定 PWM 输出，而 DCAEVT1U 字段被忽略。

PWM 模块同时支持一次性封锁（OST）和周期性（CBC）封锁。一次性封锁通常用于诸如严重过流或者电路短路等重大异常，周期性封锁常用于限流。

当一次性封锁发生时，PWMA 和 PWMB 的输出会立刻按照 TZACTL 和 TZBCTL 寄存器相应位段的设置改变。同时，TZIF.OST 标志位置 1。如果用户通过封锁中断使能寄存器（TZIE）开启了中断，则同时会产生 PWM_TZINT 中断。TZIF.OST 标志位只能通过往 TZIC.OST 位段写 1 手动清除。

当周期性封锁发生时，PWMA 和 PWMB 的输出会立刻按照 TZACTL 和 TZBCTL 寄存器相应位段的设置改变。同时，TZIF.CBC 标志位置 1。如果用户通过封锁中断使能寄存器（TZIE）开启了中断，则同时会产生 PWM_TZINT 中断。根据 TZIC.CBCCLRPUL 位段的设置，PWM 会在 TBCNT 计数到 0 或者 TBPRD 的时候检查封锁条件是否已不存在，若封锁条件已不存在则自动撤销对输出的强制。但是，TZIF.CBC 标志位依然需要通过往 TZIC.CBC 位段写 1 来手动清除。如果清除 TZIF.CBC 标志位的时候封锁条件依然存在，则该标志位会立即又被置位。

注意： 图 8-14 中虚线所示的异步逻辑通路主要用于观测原始封锁事件信号（相应的 TZSEL.OSTOUT 或 TZSEL.CBCOUT 设定为 1）。实际应用中，必须选中锁存的通路来使能上述一次性和周期性封锁功能，即 TZSEL.OSTOUT 和 TZSEL.CBCOUT 必须设定为 2 或 3。

根据数字比较封锁事件选择寄存器（TZDCSEL）中 DCAH/DCAL 和 DCBH/DCBL 位段的设置，会产生数字比较事件 DCAEVT0/1 和 DCBEVT0/1。当数字比较事件发生时，PWMA 和 PWMB 的输出会立刻按照 TZACTL 和 TZBCTL 寄存器相应位段的设置改变。同时，TZIF.DCAEVT0/1 或 TZIF.DCBEVT0/1 标志位置 1。如果用户通过封锁中断使能寄存器（TZIE）开启了中断，则同时会产生 PWM_TZINT 中断。若数字比较事件消除，则自动撤销对输出的强制。但是，TZIF 寄存器中的相应标志位依然需要通过往 TZIC 相应位段写 1 来手动清除。如果清除标志位时数字比较事件依然存在，则标志位会立即又被置位。

©2025 旋智电子科技（上海）有限公司

图 8-15 示例了信号封锁中断产生逻辑。

图 8-15：信号封锁中断产生

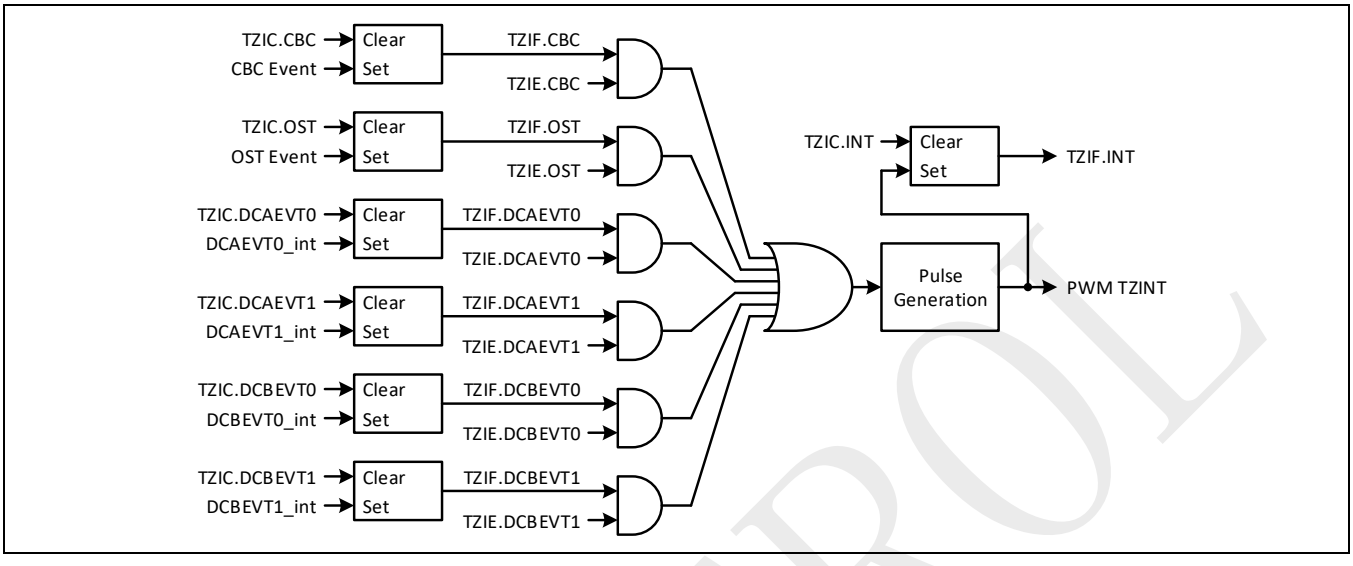


表 8-5 列举了影响 PWM 最终输出波形的各个事件的优先级。

表 8-5：影响 PWM 最终输出波形的事件优先级

优先级	事件
1（最高）	封锁（Trip-zone）事件
2	通过 AQCSFRC 寄存器的软件持续强制
3	死区（Dead-Band）控制逻辑
4	通过 AQSFRC 寄存器的软件单次强制
5（最低）	行为限定（Action-Qualifier）控制逻辑

8.7 数字比较（DC）子模块

如图 8-16 所示，数字比较模块将来自 GPIO 的 TZ0~TZ4 事件、来自比较器（COMP）的 COMP0L 和 COMP0H 事件、来自 ADC 后处理单元的 ADCPPU0TZ~ADCPPU2TZ 事件作为输入，根据用户配置产生原始的 DCAL、DCAH、DCBL、DCBH 信号，再从这四个信号的各种组合中产生原始的 RAWDCAEVT0、RAWDCAEVT1、RAWDCBEVT0、RAWDCBEVT1 事件。从中又可以选择一个事件，在一个时间窗内加以屏蔽，从而实现对事件噪声的过滤和消隐。具体如图 8-17 所示：在 TBCNT=TBPRD 或者 TBCNT=0 时，位移计数器根据 DCFOFFSET 寄存器重置计数值并开始递减计数，待计数到 0 时开启消隐窗口并保持 DCFWINDOW 寄存器所指定的时基周期。在消隐窗口这段时间内的事件会被忽略，而在消隐窗口前后的事件可以正常用于产生 SOC、SYNC 等信号。考虑到实际应用中模块间的串扰，用户可以使能其他 PWM 模块产生的消隐窗口来对本级原始数字比较事件加以屏蔽。DCFCTL 寄存器的 BLANKINV 位段可以实现对消隐窗口的反相。此外如图 8-18 所示，一旦位移计数到 0，即便上一个消隐窗口尚未结束，新的消隐窗口仍将开始。

©2025 旋智电子科技（上海）有限公司

图 8-16：数字比较子模块结构

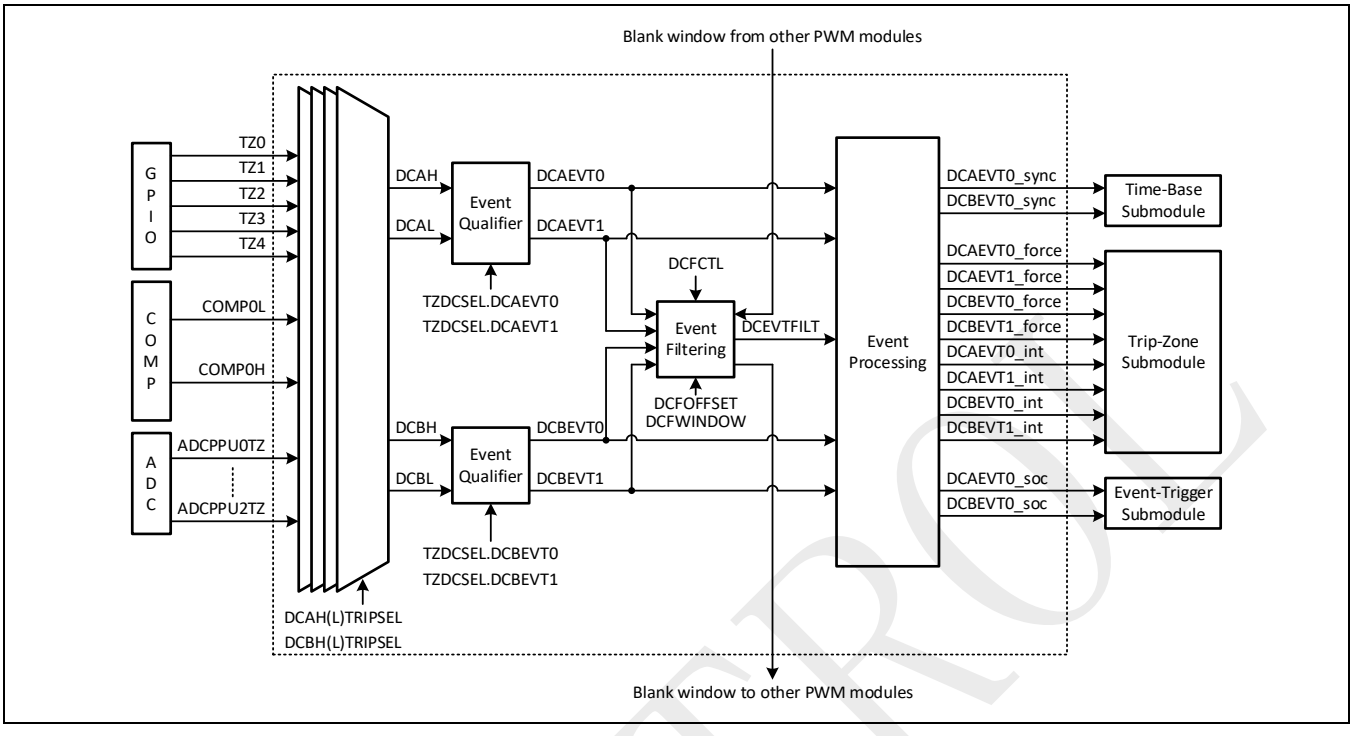
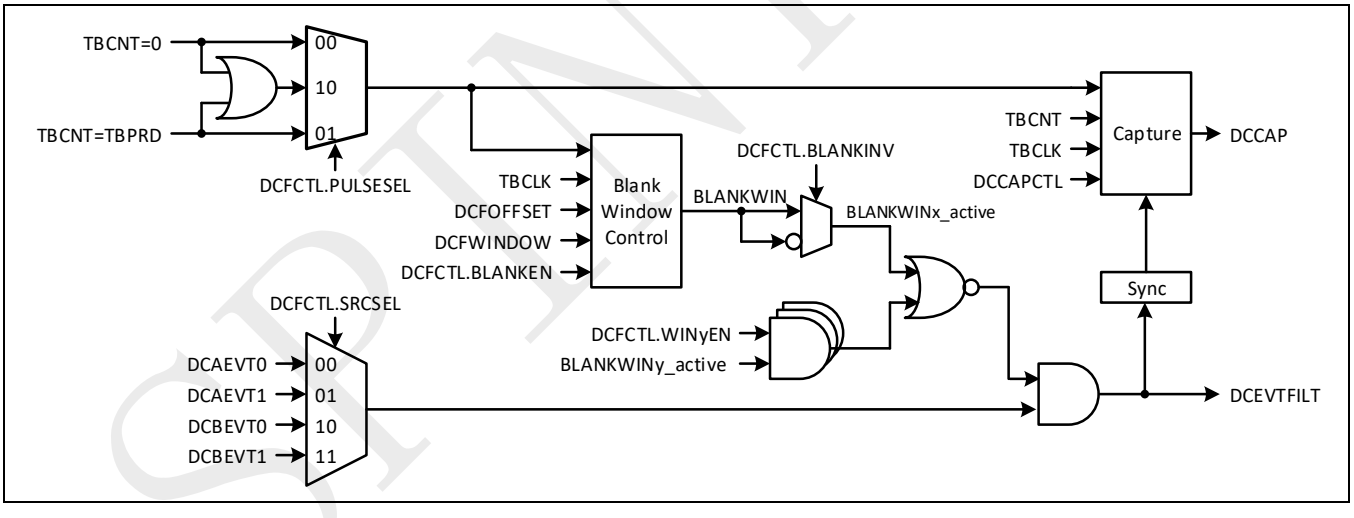
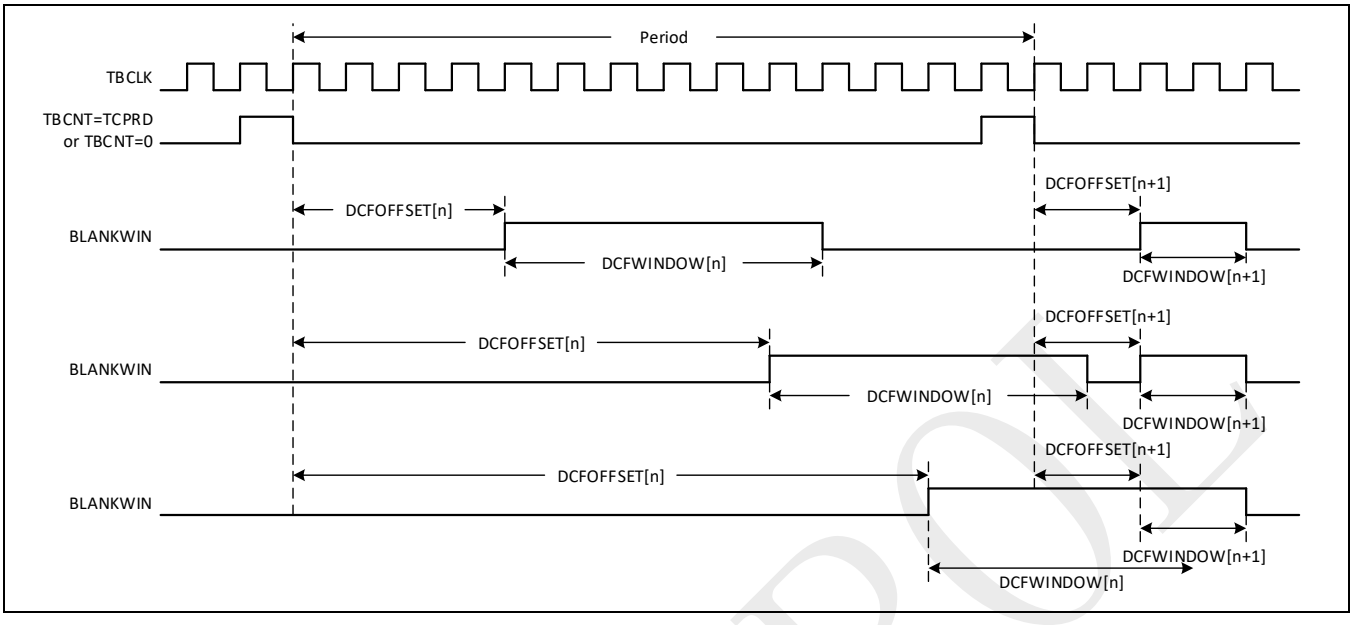


图 8-17：事件过滤逻辑



©2025 旋智电子科技（上海）有限公司

图 8-18：消隐窗口时序图



如图 8-19 所示，原始 DCAEVT0、DCAEVT1、DCBEVT0、DCBEVT1 和经过滤波得到的 DCEVTFILT 通过事件处理模块，并根据 DCACTL 和 DCBCTL 寄存器中的 EVT0SRCSEL 和 EVT1SRCSEL 字段生成最终的 DCAEVT0、DCAEVT1、DCBEVT0 和 DCBEVT1 事件。基于这些事件，可以生成以下进一步的信号：

- 行程区子模块的强制信号

  DCAEVT0_force、DCAEVT1_force、DCBEVT0_force、DCBEVT1_force
- 时基子模块的同步信号

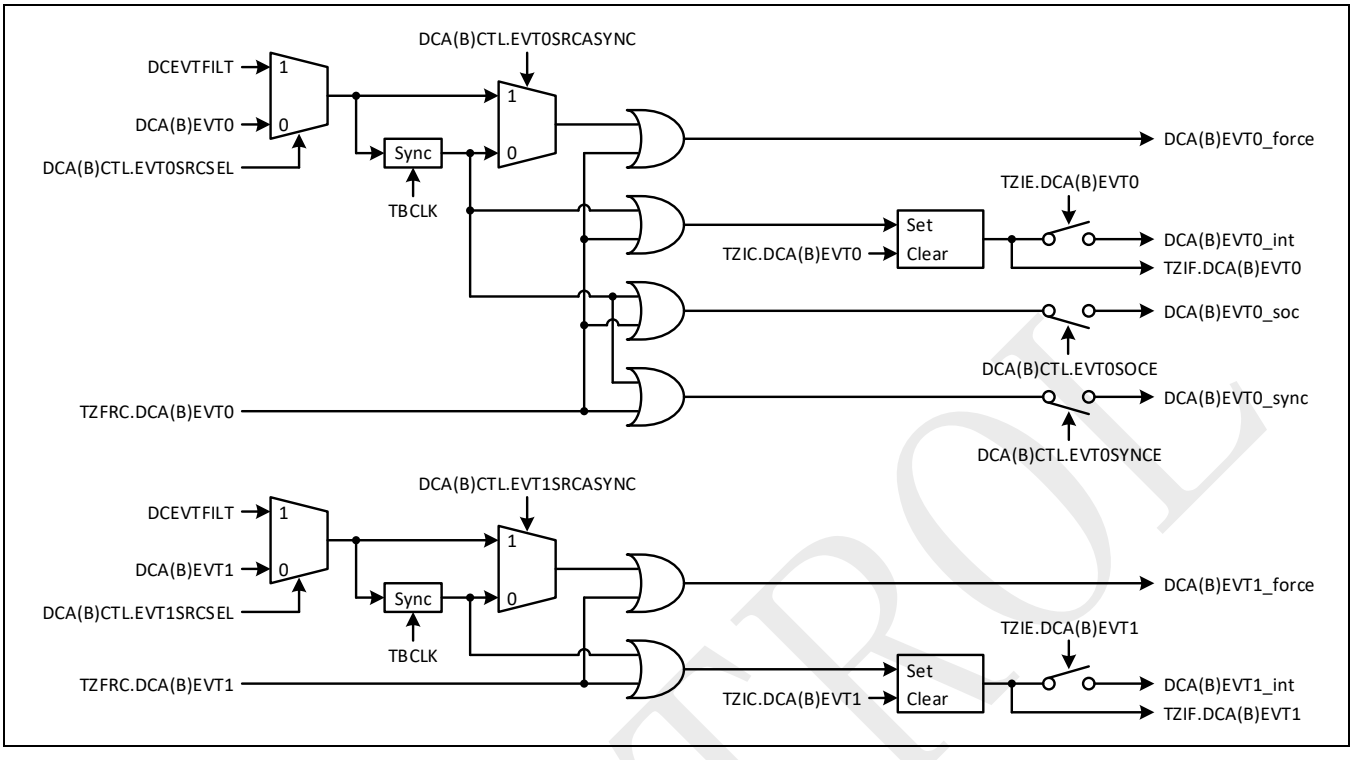
  DCAEVT0_sync、DCBEVT0_sync
- ADC 转换的起始信号

  DCAEVT0_soc、DCBEVT0_soc
- 中断信号

  DCAEVT0_int、DCAEVT1_int、DCBEVT0_int、DCBEVT1_int

SPINTROL

图 8-19：事件处理



下面给出一个用 COMP0 做相电流监测并产生消隐后的数字比较封锁信号来控制 PWMxA 和 PWMxB 输出的例子。

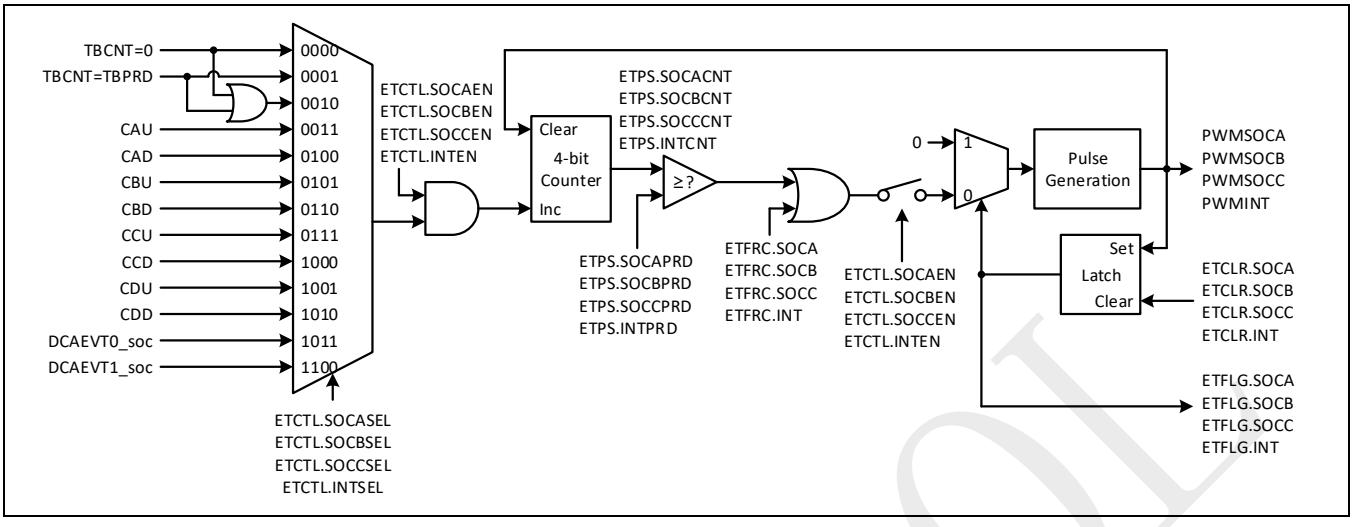
- 通过 DCAHTRIPSEL 寄存器设定 DCAH 事件为 COMP0H、COMP0L、COMP1H、COMP1L、COMP2H 和 COMP2L 信号的逻辑或；
- 设定 TZDCSEL.RAWDCAEVT0=4 来选择 DCAH=1 触发 RAWDCAEVT0 事件；
- 设定 DCFCTL.SRCSEL=0 来选择 RAWDCAEVT0 作为滤波输入；
- 设定 DCACTL.EVT0SRCSEL=1，选择滤波后的 DCEVTFILT 作为最终 DCAEVT0；
- 设定 DCBCTL.EVT0SRCSEL=1，选择滤波后的 DCEVTFILT 作为最终 DCBEVT0；
- 通过 TZACTL 寄存器的 DCAEVT0U 和 DCAEVT0D 位段设定 DCAEVT0 发生时 PWMxA 的输出；
- 通过 TZBCTL 寄存器的 DCBEVT0U 和 DCBEVT0D 位段设定 DCBEVT0 发生时 PWMxB 的输出。

8.8 事件触发（ET）子模块

事件触发子模块管理时基产生子模块、计数比较子模块和数字比较子模块产生的各种事件，并根据配置在对应事件发生时产生送至 CPU 的中断请求脉冲或送至 ADC 的采样起始脉冲。

©2025 旋智电子科技（上海）有限公司

图 8-20：事件触发子模块结构



事件触发子模块的具体结构如图 8-20 所示。为了便于软件调试，除前述硬件信号外，也支持由软件强制触发的 SOCA、SOCB、SOCC 和 INT 脉冲。基于 4 位的计数器，脉冲产生可以有如下场景：

- 从不产生；
- 每次事件到来均产生脉冲；
- 每两次事件到来产生一个脉冲；
- ……
- 每十五次事件到来产生一个脉冲。

8.9 寄存器的全局重置

如图 8-21 所示，可以设置在全局重置脉冲发生时将所有影子寄存器的值更新到对应的寄存器中。通过将 GLDCTL0 寄存器的 GLDEN 位段置 1 可以开启这个功能。GLDSEL 寄存器中的各位段指定了每个带影子寄存功能的寄存器是在全局重置脉冲发生时还是在本地重置脉冲发生时根据影子寄存器更新有效值。

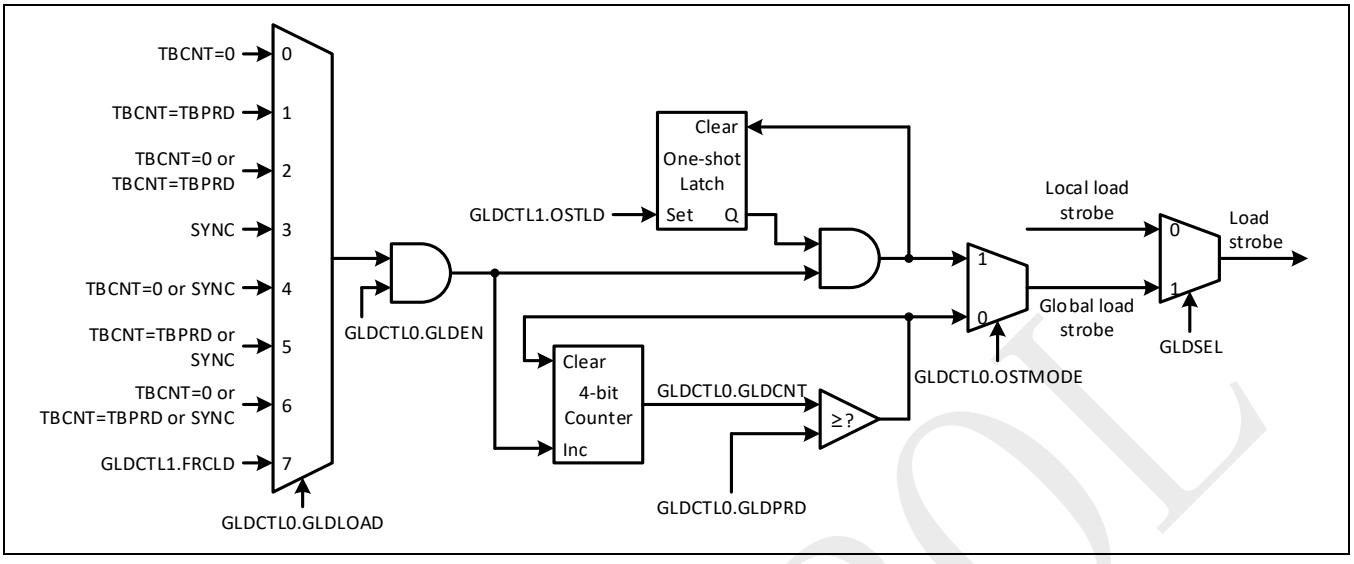
如果 GLDCTL0.OSTMODE=0，则为这种全局重置引入预分频功能，根据 GLDCTL0.GLDPRD 位段的值，每当发生相应数目的重置脉冲后才会真正做一次寄存器有效值的重置。

如果 GLDCTL0.OSTMODE=1，则开启单次重置功能。此时仅有 GLDCTL1.OSTLD 置 1 后的第一个全局重置脉冲会触发寄存器有效值的重置。该脉冲还会将 GLDCTL1.OSTLD 硬件清 0，直到软件重新置 1 以使能下一个脉冲触发寄存器重置。在 GLDCTL1.OSTLD=0 期间，全局重置脉冲将被忽略，不会触发寄存器有效值的重置。

用户还可以通过将 GLDCTL1.FRCLD 位段置 1 来软件触发全局重置。

©2025 旋智电子科技（上海）有限公司

图 8-21：影子寄存器全局读入



8.10 寄存器同时写入

在某些应用中，用户期望能对不同 PWM 模块中相同名字的寄存器同时写入相同的值。PWM 模块中提供了 PWMLINK 寄存器，由此实现对其他 PWM 模块中某个寄存器进行写操作的时候会同时更新本 PWM 模块中对应的寄存器。支持该特性的寄存器包括：TBPRD、CMPA、CMPB、CMPC、CMPD、DBRED、DBFED 和 GLDCTL1。

©2025 旋智电子科技（上海）有限公司

8.11 寄存器

8.11.1 PWM 寄存器列表

表 8-6：PWM 模块基地址

外设模块	基地址
PWM0	0x4000 2000
PWM1	0x4000 2100
PWM2	0x4000 2200
PWM3	0x4000 2300

表 8-7：PWM 寄存器列表

寄存器	偏移地址	描述	复位值
SHADOWSTS	0x00	影子状态寄存器	0x00000000
GLDCTL0*	0x04	全局影子值重置有效值控制寄存器 0	0x00000000
GLDCTL1	0x08	全局影子值重置有效值控制寄存器 1	0x00000000
GLDSEL*	0x0C	全局影子值重置有效值控制选择寄存器	0x00000000
PWMLINK*	0x10	PWM 链接寄存器	0xFFFFFFFF
TBCTL*	0x14	时基控制寄存器	0x00000006
TBPRD	0x18	时基周期寄存器	0x00000000
TBPRDA	0x1C	时基周期有效值寄存器	0x00000000
TBPHS	0x20	时基相位寄存器	0x00000000
TBCNT	0x24	时基计数寄存器	0x00000000
TBFLG	0x28	时基标志寄存器	0x00000000
TBFLGCLR	0x2C	时基标志清除寄存器	0x00000000
CMPCTL*	0x30	计数比较控制寄存器	0x00000000
CMPA	0x34	计数比较阈值 A 寄存器	0x00000000
CMPAA	0x38	计数比较阈值 A 有效值寄存器	0x00000000
CMPB	0x3C	计数比较阈值 B 寄存器	0x00000000
CMPBA	0x40	计数比较阈值 B 有效值寄存器	0x00000000
CMPC	0x44	计数比较阈值 C 寄存器	0x00000000
CMPCA	0x48	计数比较阈值 C 有效值寄存器	0x00000000
CMPD	0x4C	计数比较阈值 D 寄存器	0x00000000
CMPDA	0x50	计数比较阈值 D 有效值寄存器	0x00000000
AQCTL*	0x54	行为限定控制寄存器	0x00000000
AQCTLA	0x58	行为限定 A 路输出控制寄存器	0x00000000
AQCTLAA	0x5C	行为限定 A 路输出控制有效值寄存器	0x00000000
AQCTLB	0x60	行为限定 B 路输出控制寄存器	0x00000000
AQCTLBA	0x64	行为限定 B 路输出控制有效值寄存器	0x00000000
AQSFRC	0x68	行为限定软件强制寄存器	0x00000000
AQCSFRC	0x6C	行为限定软件持续强制寄存器	0x00000000

©2025 旋智电子科技（上海）有限公司

寄存器	偏移地址	描述	复位值
AQCSFRCA	0x70	行为限定软件持续强制有效值寄存器	0x00000000
DBCTL*	0x74	死区产生控制寄存器	0x00000070
DBCTLA	0x78	死区产生控制有效值寄存器	0x00000070
DBRED	0x7C	死区产生上升沿延迟量寄存器	0x00000000
DBREDA	0x80	死区产生上升沿延迟量有效值寄存器	0x00000000
DBFED	0x84	死区产生下降沿延迟量寄存器	0x00000000
DBFEDA	0x88	死区产生下降沿延迟量有效值寄存器	0x00000000
TZSEL*	0x8C	封锁事件选择寄存器	0x00030003
TZFLG	0x90	封锁标志寄存器	0x00000000
TZFLGCLR	0x94	封锁标志清除寄存器	0x00000000
TZDCSEL*	0x98	数字比较封锁事件寄存器	0x00000000
TZACTL	0x9C	封锁 A 路输出控制寄存器	0x00000000
TZBCTL	0xA0	封锁 B 路输出控制寄存器	0x00000000
TZIF	0xA4	封锁中断标志寄存器	0x00000000
TZIC	0xA8	封锁中断清除寄存器	0x00000000
TZIE*	0xAC	封锁中断使能寄存器	0x00000000
TZFRC*	0xB0	封锁软件强制寄存器	0x00000000
DCALTRIPSEL*	0xB4	数字比较 DCAL 事件选择寄存器	0x00000000
DCAHTRIPSEL*	0xB8	数字比较 DCAH 事件选择寄存器	0x00000000
DCBLTRIPSEL*	0xBC	数字比较 DCBL 事件选择寄存器	0x00000000
DCBHTRIPSEL*	0xC0	数字比较 DCBH 事件选择寄存器	0x00000000
DCACTL*	0xC4	数字比较 A 控制寄存器	0x00000000
DCBCTL*	0xC8	数字比较 B 控制寄存器	0x00000000
DCFCTL*	0xCC	数字比较消隐滤波控制寄存器	0x00000010
DCFOFFSET*	0xD0	数字比较消隐窗口偏移量寄存器	0x00000000
DCFOFFSETCNT	0xD4	数字比较消隐窗口偏移计数值寄存器	0x00000000
DCFWINDOW*	0xD8	数字比较消隐窗口大小寄存器	0x00000000
DCFWINDOWCNT	0xDC	数字比较消隐窗口大小计数值寄存器	0x00000000
DCCAPCTL	0xE0	数字比较时基计数捕获控制寄存器	0x00000000
DCCAP	0xE4	数字比较时基计数捕获值寄存器	0x00000000
ETCTL*	0xE8	事件触发控制寄存器	0x1E1E1E1E
ETPS*	0xEC	事件触发预分频寄存器	0x00000000
ETFLG	0xF0	事件触发标志寄存器	0x00000000
ETCLR	0xF4	事件触发清除寄存器	0x00000000
ETFRC*	0xF8	事件触发软件强制寄存器	0x00000000
PWMREGKEY	0xFC	PWM 模块写使能寄存器	0x1ACCE551

注意： 由*标记的寄存器仅当 PWRREGKEY=0x1ACCE551 时才可以改写。

©2025 旋智电子科技（上海）有限公司

8.11.2 PWM 寄存器

表 8-8：影子状态寄存器（SHADOWSTS）位段定义

SHADOWSTS (Shadow Status Register) Offset：0x0 Default：0x00000000 Access：PWMx->SHADOWSTS							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED					DBFED	DBRED	DBCTL
7	6	5	4	3	2	1	0
AQCSFRC	AQCTLB	AQCTLA	CMPD	CMPC	CMPB	CMPA	TBPRD

表 8-9：影子状态寄存器（SHADOWSTS）位段描述

位段	位段名	属性	复位值	描述
31:11	RESERVED_31_11	RO	0x0	保留
10	DBFED	RO	0x0	DBFED 寄存器影子状态 0：DBFEDA=DBFED，即已经用影子值重置有效值 1：DBFED 已经写入新的影子值，但尚未重置到有效值中
9	DBRED	RO	0x0	DBRED 寄存器影子状态 0：DBREDA=DBRED，即已经用影子值重置有效值 1：DBRED 已经写入新的影子值，但尚未重置到有效值中
8	DBCTL	RO	0x0	DBCTL 寄存器影子状态 0：DBCTLA=DBCTL，即已经用影子值重置有效值 1：DBCTL 已经写入新的影子值，但尚未重置到有效值中
7	AQCSFRC	RO	0x0	AQCSFRC 寄存器影子状态 0：AQCSFRCA=AQCSFRC，已用影子值重置有效值 1：AQCSFRC 已经写入新的影子值，但尚未重置到有效值中
6	AQCTLB	RO	0x0	AQCTLB 寄存器影子状态 0：AQCTLBA=AQCTLB，即已用影子值重置有效值

SPINTROL

位段	位段名	属性	复位值	描述
				1：AQCTLB 已经写入新的影子值，但尚未重置到有效值中
5	AQCTLA	RO	0x0	AQCTLA 寄存器影子状态 0：AQCTLAA=AQCTLA，即已用影子值重置有效值 1：AQCTLA 已经写入新的影子值，但尚未重置到有效值中
4	CMPD	RO	0x0	CMPD 寄存器影子状态 0：CMPDA=CMPD，即已经用影子值重置有效值 1：CMPD 已经写入新的影子值，但尚未重置到有效值中
3	CMPC	RO	0x0	CMPC 寄存器影子状态 0：CMPCA=CMPC，即已经用影子值重置有效值 1：CMPC 已经写入新的影子值，但尚未重置到有效值中
2	CMPB	RO	0x0	CMPB 寄存器影子状态 0：CMPBA=CMPB，即已经用影子值重置有效值 1：CMPB 已经写入新的影子值，但尚未重置到有效值中
1	CMPA	RO	0x0	CMPA 寄存器影子状态 0：CMPAA=CMPA，即已经用影子值重置有效值 1：CMPA 已经写入新的影子值，但尚未重置到有效值中
0	TBPRD	RO	0x0	TBPRD 寄存器影子状态 0：TBPRDA=TBPRD，即已经用影子值重置有效值 1：TBPRD 已经写入新的影子值，但尚未重置到有效值中

表 8-10：全局影子值重置有效值控制寄存器 0（GLDCTL0）位段定义

GLDCTL0 (Global Shadow to Active Load Control Register 0) Offset：0x4 Default：0x00000000 Access：PWMx->GLDCTL0							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED			OSTMODE	GLDCNT			

©2025 旋智电子科技（上海）有限公司

7	6	5	4	3	2	1	0
GLDPRD				GLDLOAD			GLDEN

SPINTROL

表 8-11：全局影子值重置有效值控制寄存器 0（GLDCTL0）位段描述

位段	位段名	属性	复位值	描述
31:13	RESERVED_31_13	RO	0x0	保留
12	OSTMODE	RW	0x0	单次全局重置模式 0：周期性持续重置模式，由 GLDCNT 对重置事件持续循环计数，当 GLDCNT=GLDPRD 时重置 1：单次重置模式，仅当 GLDCTL1.OSTLOAD=1 时重置事件有效
11:8	GLDCNT	RO	0x0	全局重置事件计数 记录了重置事件发生的次数
7:4	GLDPRD	RW	0x0	全局重置事件周期 本位段为 0 时将不产生重置脉冲；否则当 GLDCNT=GLDPRD 时产生重置脉冲。 OSTMODE=1 时本位段无效
3:1	GLDLOAD	RW	0x0	全局重置事件选择 0：TBCNT=0 1：TBCNT=TBPRD 2：TBCNT=0 or TBCNT=TBPRD 3：SYNC 事件 4：SYNC 事件或 TBCNT=0 5：SYNC 事件或 TBCNT=TBPRD 6：SYNC 事件或 TBCNT=0 或 TBCNT=TBPRD 7：GLDCTL1.FRCLOAD 位段被软件置 1
0	GLDEN	RW	0x0	全局重置使能 0：各寄存器影子值到有效值的重置独立控制，与 GLDSEL 寄存器设置无关 例：CMPA 的重置由 CMPCTL.CMPALOAD 决定 1：GLDSEL 可以全局控制影子值到有效值的重置

©2025 旋智电子科技（上海）有限公司

表 8-12：全局影子值重置有效值控制寄存器 1（GLDCTL1）位段定义

GLDCTL1 (Global Shadow to Active Load Control Register 1) Offset：0x8 Default：0x00000000 Access：PWMx->GLDCTL1							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						FRCLOAD	OSTLOAD

表 8-13：全局影子值重置有效值控制寄存器 1（GLDCTL1）位段描述

位段	位段名	属性	复位值	描述
31:2	RESERVED_31_2	RO	0x0	保留
1	FRCLOAD	W1S	0x0	触发影子值到有效值的全局重置事件 0：写 0 无效，读总是 0 1：写 1 产生一次全局重置事件，本位段自动清零
0	OSTLOAD	W1S	0x0	启动单次重置等待 0：写 0 无效，读到 0 表明没有等待中的重置 1：写 1 启动单次重置等待，当 GLDCTL0.GLDLOAD 位段指定的事件发生时，重置寄存器有效值，同时本位段自动清零。

表 8-14：全局影子值重置有效值控制选择寄存器（GLDSEL）位段定义

GLDSEL (Global Shadow to Active Load Select Register) Offset：0xC Default：0x00000000 Access：PWMx->GLDSEL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED					DBFED	DBRED	DBCTL
7	6	5	4	3	2	1	0
AQCSFRC	AQCTLB	AQCTLA	CMPD	CMPC	CMPB	CMPA	TBPRD

©2025 旋智电子科技（上海）有限公司

表 8-15：全局影子值重置有效值控制选择寄存器（GLDSEL）位段描述

位段	位段名	属性	复位值	描述
31:11	RESERVED_31_11	RO	0x0	保留
10	DBFED	RW	0x0	DBFED 全局重置选择 0：由 DBCTL.DBFEDLOAD 决定重置时机 1：由 GLDCTL 决定全局重置时机
9	DBRED	RW	0x0	DBRED 全局重置选择 0：由 DBCTL.DBREDLOAD 决定重置时机 1：由 GLDCTL 决定全局重置时机
8	DBCTL	RW	0x0	DBCTL 全局重置选择 0：由 DBCTL.DBCTLLOAD 决定重置时机 1：由 GLDCTL 决定全局重置时机
7	AQCSFRC	RW	0x0	AQCSFRC 全局重置选择 0：由 AQCTL.AQCSFRCLOAD 决定重置时机 1：由 GLDCTL 决定全局重置时机
6	AQCTLB	RW	0x0	AQCTLB 全局重置选择 0：由 AQCTL.AQCTLBLOAD 决定重置时机 1：由 GLDCTL 决定全局重置时机
5	AQCTLA	RW	0x0	AQCTLA 全局重置选择 0：由 AQCTL.AQCTLALOAD 决定重置时机 1：由 GLDCTL 决定全局重置时机
4	CMPD	RW	0x0	CMPD 全局重置选择 0：由 CMPCTL.CMPDLOAD 决定重置时机 1：由 GLDCTL 决定全局重置时机
3	CMPC	RW	0x0	CMPC 全局重置选择 0：由 CMPCTL.CMPCLOAD 决定重置时机 1：由 GLDCTL 决定全局重置时机
2	CMPB	RW	0x0	CMPB 全局重置选择 0：由 CMPCTL.CMPBLOAD 决定重置时机 1：由 GLDCTL 决定全局重置时机
1	CMPA	RW	0x0	CMPA 全局重置选择 0：由 CMPCTL.CMPALOAD 决定重置时机 1：由 GLDCTL 决定全局重置时机

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
0	TBPRD	RW	0x0	TBPRD 全局重置选择 0：由 TBCTL.TBPRDLOAD 决定重置时机 1：由 GLDCTL 决定全局重置时机

表 8-16：PWM 链接控制寄存器（PWMLINK）位段定义

PWMLINK (PWM Link Control Register) Offset：0x10 Default：0xFFFFFFFF Access：PWMx->PWMLINK							
31	30	29	28	27	26	25	24
GLDCTL1				DBFED			
23	22	21	20	19	18	17	16
DBRED				CMPD			
15	14	13	12	11	10	9	8
CMPC				CMPB			
7	6	5	4	3	2	1	0
CMPA				TBPRD			

表 8-17：PWM 链接控制寄存器（PWMLINK）位段描述

位段	位段名	属性	复位值	描述
31:28	GLDCTL1	RW	0xF	往 PWMLINK.GLDCTL1 指定的 PWM 的 GLDCTL1 寄存器写值时同时写本 PWM 的 GLDCTL1 寄存器 0：链接到 PWM0 对应的寄存器 1：链接到 PWM1 对应的寄存器 2：链接到 PWM2 对应的寄存器 3：链接到 PWM3 对应的寄存器 其他：不链接到任何 PWM
27:24	DBFED	RW	0xF	往 PWMLINK.DBFED 指定的 PWM 的 DBFED 寄存器写值时同时写本 PWM 的 DBFED 寄存器 0：链接到 PWM0 对应的寄存器 1：链接到 PWM1 对应的寄存器 2：链接到 PWM2 对应的寄存器 3：链接到 PWM3 对应的寄存器 其他：不链接到任何 PWM
23:20	DBRED	RW	0xF	往 PWMLINK.DBRED 指定的 PWM 的 DBRED 寄存器写值时同时写本 PWM 的 DBRED 寄存器 0：链接到 PWM0 对应的寄存器 1：链接到 PWM1 对应的寄存器 2：链接到 PWM2 对应的寄存器

SPINTROL

位段	位段名	属性	复位值	描述
				3：链接到 PWM3 对应的寄存器 其他：不链接到任何 PWM
19:16	CMPD	RW	0xF	往 PWMLINK.CMPD 指定的 PWM 的 CMPD 寄存器写值时同时写本 PWM 的 CMPD 寄存器 0：链接到 PWM0 对应的寄存器 1：链接到 PWM1 对应的寄存器 2：链接到 PWM2 对应的寄存器 3：链接到 PWM3 对应的寄存器 其他：不链接到任何 PWM
15:12	CMPC	RW	0xF	往 PWMLINK.CMPC 指定的 PWM 的 CMPC 寄存器写值时同时写本 PWM 的 CMPC 寄存器 0：链接到 PWM0 对应的寄存器 1：链接到 PWM1 对应的寄存器 2：链接到 PWM2 对应的寄存器 3：链接到 PWM3 对应的寄存器 其他：不链接到任何 PWM
11:8	CMPB	RW	0xF	往 PWMLINK.CMPB 指定的 PWM 的 CMPB 寄存器写值时同时写本 PWM 的 CMPB 寄存器 0：链接到 PWM0 对应的寄存器 1：链接到 PWM1 对应的寄存器 2：链接到 PWM2 对应的寄存器 3：链接到 PWM3 对应的寄存器 其他：不链接到任何 PWM
7:4	CMPA	RW	0xF	往 PWMLINK.CMPA 指定的 PWM 的 CMPA 寄存器写值时同时写本 PWM 的 CMPA 寄存器 0：链接到 PWM0 对应的寄存器 1：链接到 PWM1 对应的寄存器 2：链接到 PWM2 对应的寄存器 3：链接到 PWM3 对应的寄存器 其他：不链接到任何 PWM
3:0	TBPRD	RW	0xF	往 PWMLINK.TBPRD 指定的 PWM 的 TBPRD 寄存器写值时同时写本 PWM 的 TBPRD 寄存器 0：链接到 PWM0 对应的寄存器 1：链接到 PWM1 对应的寄存器 2：链接到 PWM2 对应的寄存 3：链接到 PWM3 对应的寄存器 其他：不链接到任何 PWM

©2025 旋智电子科技（上海）有限公司

表 8-18：时基控制寄存器（TBCTL）位段定义

TBCTL (Time-Base Control Register) Offset：0x14 Default：0x00000006 Access： PWMx->TBCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED				DBGRUN		TBPDIV	
15	14	13	12	11	10	9	8
TBPDIV	TBDIV			FRCSYNC	SYNCOSEL		
7	6	5	4	3	2	1	0
TBPRDLOCK	TBPRDLOAD		PHSDIR	PHSEN	CNTMODE		CNTRUN

表 8-19：时基控制寄存器（TBCTL）位段描述

位段	位段名	属性	复位值	描述
31:20	RESERVED_31_20	RO	0x0	保留
19:18	DBGRUN	RW	0x0	CPU 停止时 PWM 时基计数器的行为 调试或者 CPU 错误异常都有可能导致 CPU 停止 0：完成下一拍计数后停止 1：完成整个计数周期后停止（向上计数模式为 TBCNT=TBPRD，其余模式计数至 TBCNT=0） 2：保持正常计数
17:15	TBPDIV	RW	0x0	时基计数时钟（TBCLK）相对于 PWM 时钟的二进制预分频，总分频比为 2^{TBPDIV} x (TBDIV + 1) 0：1 分频（复位默认值） 1：2 分频 2：4 分频 3：8 分频 4：16 分频 5：32 分频 6：64 分频 7：128 分频

SPINTROL

位段	位段名	属性	复位值	描述
14:12	TBDIV	RW	0x0	时基计数时钟（TBCLK）预分频后进一步分频比 总的分频比为 2^{TBPDIV} x (TBDIV + 1) 0：1 分频（复位默认值） 1：2 分频 2：3 分频 3：4 分频 4：5 分频 5：6 分频 6：7 分频 7：8 分频
11	FRCSYNC	W1S	0x0	软件强制同步 0：写 0 无效，读总是 0 1：写 1 产生强制同步，本位段自动清零，且不受 PWMREGKEY 保护
10:8	SYNCOSEL	RW	0x0	同步输出选择 0：SYNCI 和软件强制的同步事件 1：TBCNT=0 事件 2：TBCNT=TBPRD 事件 3：TBCNT=CMPA 事件 4：TBCNT=CMPB 事件 5：TBCNT=CMPC 事件 6：TBCNT=CMPD 事件 7：关闭 SYNCO 输出
7	TBPRDLOCK	RW	0x0	TBPRD 有效值锁定 0：TBPRDA 存储的有效值根据 TBCTL.TBPRDLOAD 位段指定的时机重置 1：TBPRDA 存储的有效值维持不变
6:5	TBPRDLOAD	RW	0x0	TBPRD 有效值重置方式 TBCTL.TBPRDLOCK=1 时本位段无效 0：TBCNT=0 时用 TBPRD 重置 TBPRDA 1：SYNC 事件时用 TBPRD 重置 TBPRDA 2：SYNC 事件或 TBCNT=0 时用 TBPRD 重置 TBPRDA 3：写 TBPRD 寄存器时同时更新 TBPRDA
4	PHSDIR	RW	0x0	相位同步后的计数方向 本位段仅当 TBCNT 处于上下计数模式时有效（CNTMODE=2） 0：同步事件发生后向下计数 1：同步事件发生后向上计数

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
3	PHSEN	RW	0x0	相位同步使能 0：不用 TBPHS 寄存器更新 TBCNT 计数值 1：同步事件（SYNCI 输入、软件强制同步或数字比较同步事件）发生时，将 TBCNT 计数值更新为 TBPHS 寄存器的值
2:1	CNTMODE	RW	0x3	计数模式 0：向下计数 1：向上计数 2：上下计数 3：停止计数并维持原计数值（复位默认项）
0	CNTRUN	RW	0x0	计数开关控制 0：计数关闭 1：计数开始

表 8-20：时基周期寄存器（TBPRD）位段定义

TBPRD (Time-Base Period Register) Offset：0x18 Default：0x00000000 Access：PWMx->TBPRD							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 8-21：时基周期寄存器（TBPRD）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x0	时基计数周期

表 8-22：时基周期有效值寄存器（TBPRDA）位段定义

TBPRDA (Time-Base Period Active Register) Offset：0x1C Default：0x00000000 Access：PWMx->TBPRDA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8

©2025 旋智电子科技（上海）有限公司

VAL							
7	6	5	4	3	2	1	0
VAL							

SPINTROL

©2025 旋智电子科技（上海）有限公司

表 8-23：时基周期有效值寄存器（TBPRDA）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RO	0x0	时基计数周期有效值 当 TBCTL.PRDLOAD=3，写 TBPRD 寄存器直接同时改变本寄存器值； 否则，根据 TBCTL.PRDLOAD 指定的方式从 TBPRD 寄存器重置本寄存器值。

表 8-24：时基相位寄存器（TBPHS）位段定义

TBPHS (Time-Base Phase Register)　Offset：0x20　Default：0x00000000
Access：PWMx->TBPHS

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 8-25：时基相位寄存器（TBPHS）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x0	同步事件发生时时基计数器待更新的相位 TBCTL.PHSEN=0：忽略同步事件，TBCNT 计数值维持不变。 TBCTL.PHSEN=1：同步事件发生时更新 TBCNT 为本寄存器值。

SPINTROL

表 8-26：时基计数值寄存器（TBCNT）位段定义

TBCNT (Time-Base Counter Register) Offset：0x24 Default：0x00000000 Access：PWMx->TBCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 8-27：时基计数值寄存器（TBCNT）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x0	时基计数器当前值

表 8-28：时基标志寄存器（TBFLG）位段定义

TBFLG (Time-Base Status Register) Offset：0x28 Default：0x00000000 Access：PWMx->TBFLG							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							CNTDIR
7	6	5	4	3	2	1	0
CNTCMPD	CNTCMPC	CNTCMPB	CNTCMPA	CNTPRD	CNTZRO	CNTMAX	SYNCI

表 8-29：时基标志寄存器（TBFLG）位段描述

位段	位段名	属性	复位值	描述
31:9	RESERVED_31_9	RO	0x0	保留
8	CNTDIR	RO	0x0	时基计数器计数方向 0：时基计数器正递减计数 1：时基计数器正递增计数
7	CNTCMPD	RO	0x0	时基计数至 CMPD 标志 0：TBCNT 从未计数至 CMPD 1：TBCNT 曾计数至 CMPD

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
6	CNTCMPC	RO	0x0	时基计数至 CMPC 标志 0：TBCNT 从未计数至 CMPC 1：TBCNT 曾计数至 CMPC
5	CNTCMPB	RO	0x0	时基计数至 CMPB 标志 0：TBCNT 从未计数至 CMPB 1：TBCNT 曾计数至 CMPB
4	CNTCMPA	RO	0x0	时基计数至 CMPA 标志 0：TBCNT 从未计数至 CMPA 1：TBCNT 曾计数至 CMPA
3	CNTPRD	RO	0x0	时基计数至 TBPRD 标志 0：TBCNT 从未计数至 TBPRD 1：TBCNT 曾计数至 TBPRD
2	CNTZRO	RO	0x0	时基计数至 0 标志 0：TBCNT 从未计数至 0 1：TBCNT 曾计数至 0
1	CNTMAX	RO	0x0	时基计数至 0xFFFF 标志 0：TBCNT 从未计数至 0xFFFF 1：TBCNT 曾计数至 0xFFFF
0	SYNCI	RO	0x0	输入同步事件发生标志 0：输入同步事件从未发生 1：输入同步事件曾经发生

表 8-30：时基标志清除寄存器（TBFLGCLR）位段定义

TBFLGCLR (Time-Base Status Clear Register) Offset：0x2C Default：0x00000000 Access：PWMx->TBFLGCLR							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CNTCMPD	6 CNTCMPC	5 CNTCMPB	4 CNTCMPA	3 CNTPRD	2 CNTZRO	1 CNTMAX	0 SYNCI

SPINTROL

表 8-31：时基标志清除寄存器（TBFLGCLR）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	CNTCMPD	W1C	0x0	时基计数至 CMPD 标志清除 0：写 0 无效，读总是 0 1：写 1 清除对应的标志，本位段自动清零
6	CNTCMPC	W1C	0x0	时基计数至 CMPC 标志清除 0：写 0 无效，读总是 0 1：写 1 清除对应的标志，本位段自动清零
5	CNTCMPB	W1C	0x0	时基计数至 CMPB 标志清除 0：写 0 无效，读总是 0 1：写 1 清除对应的标志，本位段自动清零
4	CNTCMPA	W1C	0x0	时基计数至 CMPA 标志清除 0：写 0 无效，读总是 0 1：写 1 清除对应的标志，本位段自动清零
3	CNTPRD	W1C	0x0	时基计数至 TBPRD 标志清除 0：写 0 无效，读总是 0 1：写 1 清除对应的标志，本位段自动清零
2	CNTZRO	W1C	0x0	时基计数至 0 标志清除 0：写 0 无效，读总是 0 1：写 1 清除对应的标志，本位段自动清零
1	CNTMAX	W1C	0x0	时基计数至 0xFFFF 标志清除 0：写 0 无效，读总是 0 1：写 1 清除对应的标志，本位段自动清零
0	SYNCI	W1C	0x0	输入同步事件发生标志清除 0：写 0 无效，读总是 0 1：写 1 清除对应的标志，本位段自动清零

表 8-32：计数比较控制寄存器（CMPCTL）位段定义

CMPCTL (Counter-Compare Control Register)　Offset：0x30　Default：0x00000000
Access：PWMx->CMPCTL

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CMPDLOCK	14	13	12 CMPDLOAD	11 CMPCLOCK	10	9	8 CMPCLOAD
7 CMPBLOCK	6	5	4 CMPBLOAD	3 CMPALOCK	2	1	0 CMPALOAD

©2025 旋智电子科技（上海）有限公司

表 8-33：计数比较控制寄存器（CMPCTL）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	CMPDLOCK	RW	0x0	CMPD 有效值锁定 0：CMPDA 存储的有效值在 CMPCTL.CMPDLOAD 指定的时机用 CMPD 的值重置 1：CMPDA 存储的有效值维持不变
14:12	CMPDLOAD	RW	0x0	CMPD 有效值重置方式 CMPCTL.CMPDLOCK=1 时本位段无效 0：TBCNT=0 时用 CMPD 重置 CMPDA 1：TBCNT=TBPRD 时用 CMPD 重置 CMPDA 2：TBCNT=0 或 TBCNT=TBPRD 时用 CMPD 重置 3：SYNC 事件时用 CMPD 重置 CMPDA 4：SYNC 事件或 TBCNT=0 时用 CMPD 重置 5：SYNC 事件或 TBCNT=TBPRD 时用 CMPD 重置 6：SYNC 事件、TBCNT=0 或 TBCNT=TBPRD 时用 CMPD 重置 CMPDA 7：写 CMPD 寄存器时同时更新 CMPDA
11	CMPCLOCK	RW	0x0	CMPC 有效值锁定 0：CMPCA 存储的有效值在 CMPCTL.CMPCLOAD 指定的时机用 CMPC 的值重置 1：CMPCA 存储的有效值维持不变
10:8	CMPCLOAD	RW	0x0	CMPC 有效值重置方式 CMPCTL.CMPCLOCK=1 时本位段无效 0：TBCNT=0 时用 CMPC 重置 CMPCA 1：TBCNT=TBPRD 时用 CMPC 重置 CMPCA 2：TBCNT=0 或 TBCNT=TBPRD 时用 CMPC 重置 3：SYNC 事件时用 CMPC 重置 CMPCA 4：SYNC 事件或 TBCNT=0 时用 CMPC 重置 5：SYNC 事件或 TBCNT=TBPRD 时用 CMPC 重置 6：SYNC 事件、TBCNT=0 或 TBCNT=TBPRD 时用 CMPC 重置 CMPCA 7：写 CMPC 寄存器时同时更新 CMPCA
7	CMPBLOCK	RW	0x0	CMPB 有效值锁定 0：CMPBA 存储的有效值在 CMPCTL.CMPBLOAD 指定的时机用 CMPB 的值重置 1：CMPBA 存储的有效值维持不变

SPINTROL

位段	位段名	属性	复位值	描述
6:4	CMPBLOAD	RW	0x0	CMPB 有效值重置方式 CMPCTL.CMPBLOCK=1 时本位段无效 0：TBCNT=0 时用 CMPB 重置 CMPBA 1：TBCNT=TBPRD 时用 CMPB 重置 CMPBA 2：TBCNT=0 或 TBCNT=TBPRD 时用 CMPB 重置 3：SYNC 事件时用 CMPB 重置 CMPBA 4：SYNC 事件或 TBCNT=0 时用 CMPB 重置 5：SYNC 事件或 TBCNT=TBPRD 时用 CMPB 重置 6：SYNC 事件、TBCNT=0 或 TBCNT=TBPRD 时用 CMPB 重置 CMPBA 7：写 CMPB 寄存器时同时更新 CMPBA
3	CMPALOCK	RW	0x0	CMPA 有效值锁定 0：CMPAA 存储的有效值在 CMPCTL.CMPALOAD 指定的时机用 CMPA 的值重置 1：CMPAA 存储的有效值维持不变
2:0	CMPALOAD	RW	0x0	CMPA 有效值重置方式 CMPCTL.CMPALOCK=1 时本位段无效 0：TBCNT=0 时用 CMPA 重置 CMPAA 1：TBCNT=TBPRD 时用 CMPA 重置 CMPAA 2：TBCNT=0 或 TBCNT=TBPRD 时用 CMPA 重置 3：SYNC 事件时用 CMPA 重置 CMPAA 4：SYNC 事件或 TBCNT=0 时用 CMPA 重置 5：SYNC 事件或 TBCNT=TBPRD 时用 CMPA 重置 6：SYNC 事件、TBCNT=0 或 TBCNT=TBPRD 时用 CMPA 重置 CMPAA 7：写 CMPA 寄存器时同时更新 CMPAA

表 8-34：计数比较阈值 A 寄存器（CMPA）位段定义

CMPA (Counter-Compare A Threshold Register)　Offset：0x34　Default：0x00000000
Access：PWMx->CMPA

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

©2025 旋智电子科技（上海）有限公司

SPINTROL

表 8-35：计数比较阈值 A 寄存器（CMPA）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x0	计数比较阈值 A

表 8-36：计数比较阈值 A 有效值寄存器（CMPAA）位段描述

CMPAA (Counter-Compare A Threshold Active Register)　Offset：0x38　Default：0x00000000
Access：PWMx->CMPAA

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 8-37：计数比较阈值 A 有效值寄存器（CMPAA）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RO	0x0	计数比较阈值 A 有效值 TBCNT 等于该值时，产生 TBCNT=CMPA 事件

表 8-38：计数比较阈值 B 寄存器（CMPB）位段定义

CMPB (Counter-Compare B Threshold Register)　Offset：0x3C　Default：0x00000000
Access：PWMx->CMPB

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

©2025 旋智电子科技（上海）有限公司

表 8-39：计数比较阈值 B 寄存器（CMPB）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x0	计数比较阈值 B

表 8-40：计数比较阈值 B 寄存器（CMPBA）位段定义

CMPBA (Counter-Compare B Threshold Active Register)　Offset：0x40　Default：0x00000000
Access：PWMx->CMPBA

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 8-41：计数比较阈值 B 寄存器（CMPBA）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RO	0x0	计数比较阈值 B 有效值 TBCNT 等于该值时，产生 TBCNT=CMPB 事件

表 8-42：计数比较阈值 C 寄存器（CMPC）位段定义

CMPC (Counter-Compare C Threshold Register)　Offset：0x44　Default：0x00000000
Access：PWMx->CMPC

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

©2025 旋智电子科技（上海）有限公司

表 8-43：计数比较阈值 C 寄存器（CMPC）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x0	计数比较阈值 C

表 8-44：计数比较阈值 C 有效值寄存器（CMPCA）位段定义

CMPCA (Counter-Compare C Threshold Active Register)　Offset：0x48　Default：0x00000000
Access：PWMx->CMPCA

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 8-45：计数比较阈值 C 有效值寄存器（CMPCA）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RO	0x0	计数比较阈值 C 有效值 TBCNT 等于该值时，产生 TBCNT=CMPC 事件

表 8-46：计数比较阈值 D 寄存器（CMPD）位段定义

CMPD (Counter-Compare D Threshold Register)　Offset：0x4C　Default：0x00000000
Access：PWMx->CMPD

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 8-47：计数比较阈值 D 寄存器（CMPD）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x0	计数比较阈值 D

©2025 旋智电子科技（上海）有限公司

表 8-48：计数比较阈值 D 有效值寄存器（CMPDA）位段定义

CMPDA (Counter-Compare D Threshold Active Register) Offset：0x50 Default：0x00000000 Access：PWMx->CMPDA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 8-49：计数比较阈值 D 有效值寄存器（CMPDA）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RO	0x0	计数比较阈值 D 有效值 TBCNT 等于该值时，产生 TBCNT=CMPD 事件

表 8-50：行为限定控制寄存器（AQCTL）位段定义

AQCTL (Action-Qualifier Control Register) Offset：0x54 Default：0x00000000 Access：PWMx->AQCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
T1SEL				T0SEL			
7	6	5	4	3	2	1	0
AQCTLBLOCK	AQCTLBLOAD			AQCTLALOCK	AQCTLALOAD		

表 8-51：行为限定控制寄存器（AQCTL）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:12	T1SEL	RW	0x0	T1 事件源选择 0：DCAEVT0 1：DCAEVT1 2：DCBEVT0 3：DCBEVT1 4：TZ0

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				5：TZ1 6：TZ2 7：TZ3 8：TZ4 9：SYNCI 其他：禁用 T1 事件
11:8	T0SEL	RW	0x0	T0 事件源选择 0：DCAEVT0 1：DCAEVT1 2：DCBEVT0 3：DCBEVT1 4：TZ0 5：TZ1 6：TZ2 7：TZ3 8：TZ4 9：SYNCI 其他：禁用 T0 事件
7	AQCTLBLOCK	RW	0x0	AQCTLB 有效值锁定 0：AQCTLBA 维持的有效值由 QCTL.AQCTLBLOAD 指定的时机用 AQCTLB 的值重置 1：AQCTLBA 存储的有效值维持不变
6:4	AQCTLBLOAD	RW	0x0	AQCTLB 有效值重置方式 AQCTL.AQCTLBLOCK=1 本位段无效 0：TBCNT=0 时用 AQCTLB 重置 AQCTLBA 1：TBCNT=TBPRD 时用 AQCTLB 重置 AQCTLBA 2：TBCNT=0 或 TBCNT=TBPRD 时用 AQCTLB 重置 3：SYNC 事件时用 AQCTLB 重置 AQCTLBA 4：SYNC 事件或 BCNT=0 时用 AQCTLB 重置 5：SYNC 事件或 BCNT=TBPRD 时用 AQCTLB 重置 6：SYNC 事件、TBCNT=0 或 TBCNT=TBPRD 时用 AQCTLB 重置 AQCTLBA 7：写 AQCTLB 寄存器时同时更新 AQCTLBA
3	AQCTLALOCK	RW	0x0	AQCTLA 有效值锁定 0：AQCTLAA 维持的有效值由 QCTL.AQCTLALOAD 指定的时机用 AQCTLA 的值重置 1：AQCTLAA 存储的有效值维持不变

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
2:0	AQCTLALOAD	RW	0x0	AQCTLA 有效值重置方式 AQCTL.AQCTLALOCK=1 本位段无效 0：TBCNT=0 时用 AQCTLA 重置 AQCTLAA 1：TBCNT=TBPRD 时用 AQCTLA 重置 AQCTLAA 2：TBCNT=0 或 TBCNT=TBPRD 时用 AQCTLA 重置 3：SYNC 事件时用 AQCTLA 重置 AQCTLAA 4：SYNC 事件或 BCNT=0 时用 AQCTLA 重置 5：SYNC 事件或 BCNT=TBPRD 时用 AQCTLA 重置 6：SYNC 事件、TBCNT=0 或 TBCNT=TBPRD 时用 AQCTLA 重置 AQCTLAA 7：写 AQCTLA 寄存器时同时更新 AQCTLAA

表 8-52：行为限定 A 路输出控制寄存器（AQCTLA）位段定义

AQCTLA (Action-Qualifier Output A Control Register)　Offset：0x58　Default：0x00000000
Access：PWMx->AQCTLA

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED				T1D		T1U	
15	14	13	12	11	10	9	8
T0D		T0U		CBD		CBU	
7	6	5	4	3	2	1	0
CAD		CAU		PRD		ZRO	

表 8-53：行为限定 A 路输出控制寄存器（AQCTLA）位段描述

位段	位段名	属性	复位值	描述
31:20	RESERVED_31_20	RO	0x0	保留
19:18	T1D	RW	0x0	T1 事件发生且 TBCNT 递减计数时的 A 路输出 0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）
17:16	T1U	RW	0x0	T1 事件发生且 TBCNT 递增计数时的 A 路输出 0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）
15:14	T0D	RW	0x0	T0 事件发生且 TBCNT 递减计数时的 A 路输出

SPINTROL

位段	位段名	属性	复位值	描述
				0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）
13:12	T0U	RW	0x0	T0 事件发生且 TBCNT 递增计数时的 A 路输出 0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）
11:10	CBD	RW	0x0	TBCNT=CMPB 事件发生且 TBCNT 递减计数时的 A 路输出 0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）
9:8	CBU	RW	0x0	TBCNT=CMPB 事件发生且 TBCNT 递增计数时的 A 路输出 0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）
7:6	CAD	RW	0x0	TBCNT=CMPA 事件发生且 TBCNT 递减计数时的 A 路输出 0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）
5:4	CAU	RW	0x0	TBCNT=CMPA 事件发生且 TBCNT 递增计数时的 A 路输出 0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）
3:2	PRD	RW	0x0	TBCNT=TBPRD 事件时的 A 路输出 0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
1:0	ZRO	RW	0x0	TBCNT=0 事件时的 A 路输出 0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）

表 8-54：行为限定 A 路输出控制有效值寄存器（AQCTLAA）位段定义

AQCTLAA (Action-Qualifier Output A Control Active Register) Offset：0x5C Default：0x00000000 Access：PWMx->AQCTLAA							
31 30	29 28	27 26	25 24				
RESERVED							
23 22	21 20	19 18	17 16				
RESERVED		T1D	T1U				
15 14	13 12	11 10	9 8				
T0D	T0U	CBD	CBU				
7 6	5 4	3 2	1 0				
CAD	CAU	PRD	ZRO				

表 8-55：行为限定 A 路输出控制有效值寄存器（AQCTLAA）位段描述

位段	位段名	属性	复位值	描述
31:20	RESERVED_31_20	RO	0x0	保留
19:18	T1D	RO	0x0	T1 事件发生且 TBCNT 递减计数时的 A 路输出 0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）
17:16	T1U	RO	0x0	T1 事件发生且 TBCNT 递增计数时的 A 路输出 0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）
15:14	T0D	RO	0x0	T0 事件发生且 TBCNT 递减计数时的 A 路输出 0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）
13:12	T0U	RO	0x0	T0 事件发生且 TBCNT 递增计数时的 A 路输出

SPINTROL

位段	位段名	属性	复位值	描述
				0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）
11:10	CBD	RO	0x0	TBCNT=CMPB 事件发生且 TBCNT 递减计数时的 A 路输出 0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）
9:8	CBU	RO	0x0	TBCNT=CMPB 事件发生且 TBCNT 递增计数时的 A 路输出 0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）
7:6	CAD	RO	0x0	TBCNT=CMPA 事件发生且 TBCNT 递减计数时的 A 路输出 0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）
5:4	CAU	RO	0x0	TBCNT=CMPA 事件发生且 TBCNT 递增计数时的 A 路输出 0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）
3:2	PRD	RO	0x0	TBCNT=TBPRD 事件时的 A 路输出 0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）
1:0	ZRO	RO	0x0	TBCNT=0 事件时的 A 路输出 0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）

©2025 旋智电子科技（上海）有限公司

表 8-56：行为限定 B 路输出控制寄存器（AQCTLB）位段定义

AQCTLB (Action-Qualifier Output B Control Register) Offset：0x60 Default：0x00000000
Access： PWMx->AQCTLB

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED				T1D		T1U	
15	14	13	12	11	10	9	8
T0D		T0U		CBD		CBU	
7	6	5	4	3	2	1	0
CAD		CAU		PRD		ZRO	

表 8-57：行为限定 B 路输出控制寄存器（AQCTLB）位段描述

位段	位段名	属性	复位值	描述
31:20	RESERVED_31_20	RO	0x0	保留
19:18	T1D	RW	0x0	T1 事件发生且 TBCNT 递减计数时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低） 2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）
17:16	T1U	RW	0x0	T1 事件发生且 TBCNT 递增计数时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低） 2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）
15:14	T0D	RW	0x0	T0 事件发生且 TBCNT 递减计数时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低） 2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）
13:12	T0U	RW	0x0	T0 事件发生且 TBCNT 递增计数时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低） 2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）
11:10	CBD	RW	0x0	TBCNT=CMPB 事件发生且 TBCNT 递减计数时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低）

SPINTROL

位段	位段名	属性	复位值	描述
				2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）
9:8	CBU	RW	0x0	TBCNT=CMPB 事件发生且 TBCNT 递增计数时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低） 2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）
7:6	CAD	RW	0x0	TBCNT=CMPA 事件发生且 TBCNT 递减计数时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低） 2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）
5:4	CAU	RW	0x0	TBCNT=CMPA 事件发生且 TBCNT 递增计数时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低） 2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）
3:2	PRD	RW	0x0	TBCNT=TBPRD 事件时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低） 2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）
1:0	ZRO	RW	0x0	TBCNT=0 事件时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低） 2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）

SPINTROL

©2025 旋智电子科技（上海）有限公司

表 8-58：行为限定 B 路输出控制有效值寄存器（AQCTLBA）位段定义

AQCTLBA (Action-Qualifier Output B Control Active Register) Offset：0x64 Default：0x00000000 Access：PWMx->AQCTLBA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED				T1D		T1U	
15	14	13	12	11	10	9	8
T0D		T0U		CBD		CBU	
7	6	5	4	3	2	1	0
CAD		CAU		PRD		ZRO	

表 8-59：行为限定 B 路输出控制有效值寄存器（AQCTLBA）位段描述

位段	位段名	属性	复位值	描述
31:20	RESERVED_31_20	RO	0x0	保留
19:18	T1D	RO	0x0	T1 事件发生且 TBCNT 递减计数时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低） 2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）
17:16	T1U	RO	0x0	T1 事件发生且 TBCNT 递增计数时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低） 2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）
15:14	T0D	RO	0x0	T0 事件发生且 TBCNT 递减计数时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低） 2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）
13:12	T0U	RO	0x0	T0 事件发生且 TBCNT 递增计数时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低） 2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）
11:10	CBD	RO	0x0	TBCNT=CMPB 事件发生且 TBCNT 递减计数时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低）

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）
9:8	CBU	RO	0x0	TBCNT=CMPB 事件发生且 TBCNT 递增计数时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低） 2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）
7:6	CAD	RO	0x0	TBCNT=CMPA 事件发生且 TBCNT 递减计数时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低） 2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）
5:4	CAU	RO	0x0	TBCNT=CMPA 事件发生且 TBCNT 递增计数时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低） 2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）
3:2	PRD	RO	0x0	TBCNT=TBPRD 事件时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低） 2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）
1:0	ZRO	RO	0x0	TBCNT=0 事件时的 B 路输出 0：不变（禁止操作） 1：清零（PWMxB 输出低） 2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）

SPINTROL

©2025 旋智电子科技（上海）有限公司

表 8-60：行为限定软件强制寄存器（AQSFRC）位段定义

AQSFRC (Action-Qualifier Software Force Register) Offset：0x68 Default：0x00000000
Access：PWMx->AQSFRC

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						OTSFRC	
7	6	5	4	3	2	1	0
RESERVED		CSFLOAD		ACTSFB		ACTSFA	

表 8-61：行为限定软件强制寄存器（AQSFRC）位段描述

位段	位段名	属性	复位值	描述
31:10	RESERVED_31_10	RO	0x0	保留
9:8	OTSFRC	W1S	0x0	行为限定单次软件强制，本位段自动清零 0：写 0 无效，读总是 0 1：写 1 产生 A 路单次行为限定软件强制事件 2：写 2 产生 B 路单次行为限定软件强制事件 3：写 3 产生 AB 两路单次行为限定软件强制事件
7:6	RESERVED_7_6	RO	0x0	保留
5:4	CSFLOAD	RW	0x0	AQCSFRC 寄存器有效值重置方式 0：TBCNT=0 时用 AQCSFRC 更新 AQCSFRCA 1：TBCNT=TBPRD 时用 AQCSFRC 更新 AQCSFRCA 2：TBCNT=0 或 TBCNT=TBPRD 时用 AQCSFRC 更新 3：写 AQCSFRC 寄存器时同时更新 AQCSFRCA
3:2	ACTSFB	RW	0x0	B 路行为限定单次软件强制时的输出 0：不变（禁止操作） 1：清零（PWMxB 输出低） 2：置高（PWMxB 输出高） 3：翻转（PWMxB 当前输出翻转）
1:0	ACTSFA	RW	0x0	A 路行为限定单次软件强制时的输出 0：不变（禁止操作） 1：清零（PWMxA 输出低） 2：置高（PWMxA 输出高） 3：翻转（PWMxA 当前输出翻转）

SPINTROL

表 8-62：行为限定软件持续强制寄存器（AQCSFRC）位段定义

AQCSFRC (Action-Qualifier Continuous Software Force Register) Offset：0x6C Default：0x00000000 Access：PWMx->AQCSFRC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						CSFB	
7	6	5	4	3	2	1	0
RESERVED						CSFA	

表 8-63：行为限定软件持续强制寄存器（AQCSFRC）位段描述

位段	位段名	属性	复位值	描述
31:10	RESERVED_31_10	RO	0x0	保留
9:8	CSFB	RW	0x0	B 路软件持续强制 0：不强制 1：强制 B 路输出持续为低 2：强制 B 路输出持续为高 3：不强制
7:2	RESERVED_7_2	RO	0x0	保留
1:0	CSFA	RW	0x0	A 路软件持续强制 0：不强制 1：强制 A 路输出持续为低 2：强制 A 路输出持续为高 3：不强制

表 8-64：行为限定软件持续强制有效值寄存器（AQCSFRCA）位段定义

AQCSFRCA (Action-Qualifier Continuous Software Force Active Register) Offset：0x70 Default：0x00000000 Access：PWMx->AQCSFRCA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						CSFB	
7	6	5	4	3	2	1	0
RESERVED						CSFA	

©2025 旋智电子科技（上海）有限公司

表 8-65：行为限定软件持续强制有效值寄存器（AQCSFRCA）位段描述

位段	位段名	属性	复位值	描述
31:10	RESERVED_31_10	RO	0x0	保留
9:8	CSFB	RO	0x0	B 路软件持续强制 0：不强制 1：强制 B 路输出持续为低 2：强制 B 路输出持续为高 3：不强制
7:2	RESERVED_7_2	RO	0x0	保留
1:0	CSFA	RO	0x0	A 路软件持续强制 0：不强制 1：强制 A 路输出持续为低 2：强制 A 路输出持续为高 3：不强制

表 8-66：死区产生控制寄存器（DBCTL）位段定义

DBCTL (Dead-Band Generator Control Register)　Offset：0x74　Default：0x00000070
Access：PWMx->DBCTL

31	30	29	28	27	26	25	24
RESERVED					DBFEDLOCK	DBFEDLOAD	
23	22	21	20	19	18	17	16
RESERVED	DBREDLOCK	DBREDLOAD		RESERVED	DBCTLLOCK	DBCTLLOAD	
15	14	13	12	11	10	9	8
RESERVED					CTLWE	HALFCYCLE	DUALEDGE
7	6	5	4	3	2	1	0
OUTBSRC	OUTASRC	FEDPOL	REDPOL	FEDSRC	REDSRC	FEDEN	REDEN

表 8-67：死区产生控制寄存器（DBCTL）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26	DBFEDLOCK	RW	0x0	DBFED 有效值锁定 0：DBFEDA 维持的有效值由 DBCTL.DBFEDLOAD 指定的时机用 DBFED 的值重置 1：DBFEDA 存储的有效值维持不变
25:24	DBFEDLOAD	RW	0x0	DBFED 有效值重置方式 DBCTL.DBFEDLOCK=1 时本位段无效 0：TBCNT=0 时用 DBFED 重置 DBFEDA 1：TBCNT=TBPRD 时用 DBFED 重置 DBFEDA 2：TBCNT=0 或 TBCNT=TBPRD 时用 DBFED 重

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				置 3：写 DBFED 寄存器时同时更新 DBFEDA
23	RESERVED_23	RO	0x0	保留
22	DBREDLOCK	RW	0x0	DBRED 有效值锁定 0：DBREDA 存储的有效值由 DBCTL.DBREDLOAD 指定的时机用 DBRED 的值重置 1：DBREDA 存储的有效值维持不变
21:20	DBREDLOAD	RW	0x0	DBRED 有效值重置方式 DBCTL.DBREDLOCK=1 时本位段无效 0：TBCNT=0 时用 DBRED 重置 DBREDA 1：TBCNT=TBPRD 时用 DBRED 重置 DBREDA 2：TBCNT=0 或 TBCNT=TBPRD 时用 DBRED 重置 3：写 DBRED 寄存器时同时更新 DBREDA
19	RESERVED_19	RO	0x0	保留
18	DBCTLLOCK	RW	0x0	DBCTLA 有效值锁定 0：DBCTLA 存储的有效值由 DBCTL.DBCTLLOAD 指定的时机用 DBCTL[9:0]的值重置 1：DBCTLA 存储的有效值维持不变
17:16	DBCTLLOAD	RW	0x0	DBCTLA 有效值重置方式 DBCTL.DBCTLLOCK=1 时本位段无效 0：TBCNT=0 时用 DBCTL[9:0]重置 DBCTLA 1：TBCNT=TBPRD 时用 DBCTL[9:0]重置 DBCTLA 2：TBCNT=0 或 TBCNT=TBPRD 用 DBCTL[9:0]重置 3：写 DBCTL[9:0]时同时更新 DBCTLA
15:11	RESERVED_15_11	RO	0x0	保留
10	CTLWE	W1C	0x0	本寄存器低 10 位写使能 本位段必须与低 10 位同时写 0：不改写低 10 位 1：改写低 10 位
9	HALFCYCLE	RW	0x0	半时钟周期模式使能 0：全时钟周期模式，即死区控制计数时钟为时基计数时钟 TBCLK 1：半时钟周期模式，即死区控制计数时钟为 2 倍时基计数时钟 TBCLK
8	DUALEDGE	RW	0x0	双边沿延迟模式 0：上升沿延迟和下降沿延迟为两个独立通路

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				1：对同一个信号输入先做上升沿延迟，再做下降沿延迟
7	OUTBSRC	RW	0x0	B 路输出源 0：下降沿延迟通路作为 B 路输出 1：上升沿延迟通路作为 B 路输出
6	OUTASRC	RW	0x1	A 路输出源 0：下降沿延迟通路作为 A 路输出 1：上升沿延迟通路作为 A 路输出
5	FEDPOL	RW	0x1	下降沿延迟通路输出极性 0：下降沿延迟通路输出取反变为低电平有效 1：下降沿延迟通路输出保持原始极性
4	REDPOL	RW	0x1	上升沿延迟通路输出极性 0：上升沿延迟通路输出取反变为低电平有效 1：上升沿延迟通路输出保持原始极性
3	FEDSRC	RW	0x0	下降沿延迟通路输入源 0：A 路输入作为下降沿延迟通路源 1：B 路输入作为下降沿延迟通路源
2	REDSRC	RW	0x0	上升沿延迟通路输入源 0：A 路输入作为上升沿延迟通路源 1：B 路输入作为上升沿延迟通路源
1	FEDEN	RW	0x0	下降沿延时通路使能 0：绕开下降沿延时通路，将行为限定子模块输出的信号直接送至下一级 1：启用下降沿延时通路
0	REDEN	RW	0x0	上升沿延时通路使能 0：绕开上升沿延时通路，将行为限定子模块输出的信号直接送至下一级 1：启用上升沿延时通路

表 8-68：死区产生控制有效值寄存器（DBCTLA）位段定义

DBCTLA (Dead-Band Generator Control Active Register)　Offset：0x78　Default：0x00000070
Access：PWMx->DBCTLA

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						HALFCYCLE	DUALEDGE

SPINTROL

7	6	5	4	3	2	1	0
OUTBSRC	OUTASRC	FEDPOL	REDPOL	FEDSRC	REDSRC	FEDEN	REDEN

SPINTROL

©2025 旋智电子科技（上海）有限公司

表 8-69：死区产生控制有效值寄存器（DBCTLA）位段描述

位段	位段名	属性	复位值	描述
31:10	RESERVED_31_10	RO	0x0	保留
9	HALFCYCLE	RO	0x0	半时钟周期模式使能 0：全时钟周期模式，即死区控制计数时钟为时基计数时钟 TBCLK 1：半时钟周期模式，即死区控制计数时钟为 2 倍时基计数时钟 TBCLK
8	DUALEDGE	RO	0x0	双边沿延迟模式 0：上升沿延迟和下降沿延迟为两个独立通路 1：对同一个信号输入先做上升沿延迟，再做下降沿延迟
7	OUTBSRC	RO	0x0	B 路输出源 0：下降沿延迟通路作为 B 路输出 1：上升沿延迟通路作为 B 路输出
6	OUTASRC	RO	0x1	A 路输出源 0：下降沿延迟通路作为 A 路输出 1：上升沿延迟通路作为 A 路输出
5	FEDPOL	RO	0x1	下降沿延迟通路输出极性 0：下降沿延迟通路输出取反变为低电平有效 1：下降沿延迟通路输出保持原始极性
4	REDPOL	RO	0x1	上升沿延迟通路输出极性 0：上升沿延迟通路输出取反变为低电平有效 1：上升沿延迟通路输出保持原始极性
3	FEDSRC	RO	0x0	下降沿延迟通路输入源 0：A 路输入作为下降沿延迟通路源 1：B 路输入作为下降沿延迟通路源
2	REDSRC	RO	0x0	上升沿延迟通路输入源 0：A 路输入作为上升沿延迟通路源 1：B 路输入作为上升沿延迟通路源
1	FEDEN	RO	0x0	下降沿延时通路使能 0：绕开下降沿延时通路，将行为限定子模块输出的信号直接送至下一级 1：启用下降沿延时通路
0	REDEN	RO	0x0	上升沿延时通路使能 0：绕开上升沿延时通路，将行为限定子模块输出的信号直接送至下一级 1：启用上升沿延时通路

©2025 旋智电子科技（上海）有限公司

表 8-70：死区产生上升沿延迟量寄存器（DBRED）位段定义

DBRED (Dead-Band Generator Rising Edge Delay Register) Offset：0x7C Default：0x00000000 Access：PWMx->DBRED							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 8-71：死区产生上升沿延迟量寄存器（DBRED）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x0	上升沿延迟量

表 8-72：死区产生上升沿延迟量有效值寄存器（DBREDA）位段定义

DBREDA (Dead-Band Generator Rising Edge Delay Active Register) Offset：0x80 Default：0x00000000 Access：PWMx->DBREDA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 8-73：死区产生上升沿延迟量有效值寄存器（DBREDA）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RO	0x0	上升沿延迟量有效值

©2025 旋智电子科技（上海）有限公司

表 8-74：死区产生下降沿延迟量寄存器（DBFED）位段定义

DBFED (Dead-Band Generator Falling Edge Delay Register) Offset：0x84 Default：0x00000000 Access：PWMx->DBFED							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 8-75：死区产生下降沿延迟量寄存器（DBFED）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x0	下降沿延迟量

表 8-76：死区产生下降沿延迟量有效值寄存器（DBFEDA）位段定义

DBFEDA (Dead-Band Generator Falling Edge Delay Active Register) Offset：0x88 Default：0x00000000 Access：PWMx->DBFEDA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 8-77：死区产生下降沿延迟量有效值寄存器（DBFEDA）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RO	0x0	下降沿延迟量有效值

SPINTROL

表 8-78：封锁事件选择寄存器（TZSEL）位段定义

TZSEL (Trip-Zone Event Select Register) Offset：0x8C Default：0x00030003
Access： PWMx->TZSEL

31	30	29	28	27	26	25	24
CBCWE	RESERVED				DBGCBC	DCBEVT1	DCAEVT1
23	22	21	20	19	18	17	16
CLKERRCBC	TZ4CBC	TZ3CBC	TZ2CBC	TZ1CBC	TZ0CBC	CBCOUT	
15	14	13	12	11	10	9	8
OSTWE	RESERVED				DBGOST	DCBEVT0	DCAEVT0
7	6	5	4	3	2	1	0
CLKERROST	TZ4OST	TZ3OST	TZ2OST	TZ1OST	TZ0OST	OSTOUT	

表 8-79：封锁事件选择寄存器（TZSEL）位段描述

位段	位段名	属性	复位值	描述
31	CBCWE	W1S	0x0	周期性封锁控制位段写使能 本位段必须与周期性封锁控制位段同时写 0：不改写周期性封锁控制位段 1：改写周期性封锁控制位段，本位段自动清零
30:27	RESERVED_30_27	RO	0x0	保留
26	DBGCBC	RW	0x0	选中（逻辑或）调试或异常等原因引起的 CPU 停止作为周期性封锁事件 0：不选中 1：选中
25	DCBEVT1	RW	0x0	选中（逻辑或）数字比较事件 DCBEVT1 作为周期性封锁事件 0：不选中 1：选中
24	DCAEVT1	RW	0x0	选中（逻辑或）数字比较事件 DCAEVT1 作为周期性封锁事件 0：不选中 1：选中
23	CLKERRCBC	RW	0x0	选中（逻辑或）时钟错误作为周期性封锁事件 0：不选中 1：选中
22	TZ4CBC	RW	0x0	选中（逻辑或）TZ4 作为周期性封锁事件 0：不选中 1：选中

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
21	TZ3CBC	RW	0x0	选中（逻辑或）TZ3 作为周期性封锁事件 0：不选中 1：选中
20	TZ2CBC	RW	0x0	选中（逻辑或）TZ2 作为周期性封锁事件 0：不选中 1：选中
19	TZ1CBC	RW	0x0	选中（逻辑或）TZ1 作为周期性封锁事件 0：不选中 1：选中
18	TZ0CBC	RW	0x0	选中（逻辑或）TZ0 作为周期性封锁事件 0：不选中 1：选中
17:16	CBCOUT	RW	0x3	周期性封锁信号输出选择 0：不输出周期性封锁事件 1：仅选择异步通路，即各选中事件原始逻辑或 2：仅选择锁存通路 3：同时选择异步通路和锁存通路（逻辑或）
15	OSTWE	W1S	0x0	一次性封锁控制位段写使能 本位段必须与一次性封锁控制位段同时写 0：不改写一次性封锁控制位段 1：改写一次性封锁控制位段，本位段自动清零
14:11	RESERVED_14_11	RO	0x0	保留
10	DBGOST	RW	0x0	选中（逻辑或）调试或异常等原因引起的 CPU 停止作为一次性封锁事件 0：不选中 1：选中
9	DCBEVT0	RW	0x0	选中（逻辑或）DCBEVT0 作为一次性封锁事件 0：不选中 1：选中
8	DCAEVT0	RW	0x0	选中（逻辑或）DCAEVT0 作为一次性封锁事件 0：不选中 1：选中
7	CLKERROST	RW	0x0	选中（逻辑或）时钟错误作为一次性封锁事件 0：不选中 1：选中

SPINTROL

位段	位段名	属性	复位值	描述
6	TZ4OST	RW	0x0	选中（逻辑或）TZ4 作为一次性封锁事件 0：不选中 1：选中
5	TZ3OST	RW	0x0	选中（逻辑或）TZ3 作为一次性封锁事件 0：不选中 1：选中
4	TZ2OST	RW	0x0	选中（逻辑或）TZ2 作为一次性封锁事件 0：不选中 1：选中
3	TZ1OST	RW	0x0	选中（逻辑或）TZ1 作为一次性封锁事件 0：不选中 1：选中
2	TZ0OST	RW	0x0	选中（逻辑或）TZ0 作为一次性封锁事件 0：不选中 1：选中
1:0	OSTOUT	RW	0x3	一次性封锁信号输出选择 0：不输出一次性封锁事件 1：仅选择异步通路，即各选中事件原始逻辑或 2：仅选择锁存通路 3：同时选择异步通路和锁存通路（逻辑或）

表 8-80：封锁标志寄存器（TZFLG）位段定义

TZFLG (Trip-Zone Flag Register) Offset：0x90 Default：0x00000000 Access： PWMx->TZFLG							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						DBGCBC	DCBEVT1
15	14	13	12	11	10	9	8
DCAEVT1	CLKERRCBC	TZ4CBC	TZ3CBC	TZ2CBC	TZ1CBC	TZ0CBC	DBGOST
7	6	5	4	3	2	1	0
DCBEVT0	DCAEVT0	CLKERROST	TZ4OST	TZ3OST	TZ2OST	TZ1OST	TZ0OST

表 8-81：封锁标志寄存器（TZFLG）位段描述

位段	位段名	属性	复位值	描述
31:18	RESERVED_31_18	RO	0x0	保留

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
17	DBGCBC	RO	0x0	锁存的 CPU 停止（由调试或异常错误等原因引起）触发的周期性封锁标志 0：该封锁事件未发生 1：该封锁事件发生
16	DCBEVT1	RO	0x0	锁存的数字比较 DCBEVT1 触发的周期封锁标志 0：该封锁事件未发生 1：该封锁事件发生
15	DCAEVT1	RO	0x0	锁存的数字比较 DCAEVT1 触发的周期封锁标志 0：该封锁事件未发生 1：该封锁事件发生
14	CLKERRCBC	RO	0x0	锁存的时钟错误触发的周期封锁标志 0：该封锁事件未发生 1：该封锁事件发生
13	TZ4CBC	RO	0x0	锁存的 TZ4 触发的周期封锁标志 0：该封锁事件未发生 1：该封锁事件发生
12	TZ3CBC	RO	0x0	锁存的 TZ3 触发的周期封锁标志 0：该封锁事件未发生 1：该封锁事件发生
11	TZ2CBC	RO	0x0	锁存的 TZ2 触发的周期封锁标志 0：该封锁事件未发生 1：该封锁事件发生
10	TZ1CBC	RO	0x0	锁存的 TZ1 触发的周期封锁标志 0：该封锁事件未发生 1：该封锁事件发生
9	TZ0CBC	RO	0x0	锁存的 TZ0 触发的周期封锁标志 0：该封锁事件未发生 1：该封锁事件发生
8	DBGOST	RO	0x0	锁存的 CPU 停止（由调试或异常错误等原因引起）触发的一次性封锁标志 0：该封锁事件未发生 1：该封锁事件发生
7	DCBEVT0	RO	0x0	锁存的数字比较 DCBEVT0 触发的一次封锁标志 0：该封锁事件未发生 1：该封锁事件发生

SPINTROL

位段	位段名	属性	复位值	描述
6	DCAEVT0	RO	0x0	锁存的数字比较 DCAEVT0 触发的一次封锁标志 0：该封锁事件未发生 1：该封锁事件发生
5	CLKERROST	RO	0x0	锁存的时钟错误触发的一次封锁标志 0：该封锁事件未发生 1：该封锁事件发生
4	TZ4OST	RO	0x0	锁存的 TZ4 触发的一次封锁标志 0：该封锁事件未发生 1：该封锁事件发生
3	TZ3OST	RO	0x0	锁存的 TZ3 触发的一次封锁标志 0：该封锁事件未发生 1：该封锁事件发生
2	TZ2OST	RO	0x0	锁存的 TZ2 触发的一次封锁标志 0：该封锁事件未发生 1：该封锁事件发生
1	TZ1OST	RO	0x0	锁存的 TZ1 触发的一次封锁标志 0：该封锁事件未发生 1：该封锁事件发生
0	TZ0OST	RO	0x0	锁存的 TZ0 触发的一次封锁标志 0：该封锁事件未发生 1：该封锁事件发生

表 8-82：封锁状态清除寄存器（TZSTSCLR）位段定义

TZFLGCLR (Trip-Zone Flag Clear Register)　Offset：0x94　Default：0x00000000 Access：PWMx->TZFLGCLR							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						DBGCBC	DCBEVT1
15	14	13	12	11	10	9	8
DCAEVT1	CLKERRCBC	TZ4CBC	TZ3CBC	TZ2CBC	TZ1CBC	TZ0CBC	DBGOST
7	6	5	4	3	2	1	0
DCBEVT0	DCAEVT0	CLKERROST	TZ4OST	TZ3OST	TZ2OST	TZ1OST	TZ0OST

表 8-83：封锁状态清除寄存器（TZSTSCLR）位段描述

位段	位段名	属性	复位值	描述
31:18	RESERVED_31_18	RO	0x0	保留

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
17	DBGCBC	W1C	0x0	清除由 CPU 停止（因为调试或错误异常等原因引起）触发的周期封锁标志 0：写 0 无效，读总是 0 1：写 1 清除锁存的标志，本位段自动清零
16	DCBEVT1	W1C	0x0	清除由 DCBEVT1 触发的周期封锁标志 0：写 0 无效，读总是 0 1：写 1 清除锁存的标志，本位段自动清零
15	DCAEVT1	W1C	0x0	清除由 DCAEVT1 触发的周期封锁标志 0：写 0 无效，读总是 0 1：写 1 清除锁存的标志，本位段自动清零
14	CLKERRCBC	W1C	0x0	清除由时钟错误触发的周期封锁标志 0：写 0 无效，读总是 0 1：写 1 清除锁存的标志，本位段自动清零
13	TZ4CBC	W1C	0x0	清除由 TZ4 触发的周期封锁标志 0：写 0 无效，读总是 0 1：写 1 清除锁存的标志，本位段自动清零
12	TZ3CBC	W1C	0x0	清除由 TZ3 触发的周期封锁标志 0：写 0 无效，读总是 0 1：写 1 清除锁存的标志，本位段自动清零
11	TZ2CBC	W1C	0x0	清除由 TZ2 触发的周期封锁标志 0：写 0 无效，读总是 0 1：写 1 清除锁存的标志，本位段自动清零
10	TZ1CBC	W1C	0x0	清除由 TZ1 触发的周期封锁标志 0：写 0 无效，读总是 0 1：写 1 清除锁存的标志，本位段自动清零
9	TZ0CBC	W1C	0x0	清除由 TZ0 触发的周期封锁标志 0：写 0 无效，读总是 0 1：写 1 清除锁存的标志，本位段自动清零
8	DBGOST	W1C	0x0	清除由 CPU 停止（因为调试或错误异常等原因引起）触发的一次封锁标志 0：写 0 无效，读总是 0 1：写 1 清除锁存的标志，本位段自动清零
7	DCBEVT0	W1C	0x0	清除由 DCBEVT0 触发的一次封锁标志 0：写 0 无效，读总是 0 1：写 1 清除锁存的标志，本位段自动清零
6	DCAEVT0	W1C	0x0	清除由 DCAEVT0 触发的一次封锁标志

SPINTROL

位段	位段名	属性	复位值	描述
				0：写 0 无效，读总是 0 1：写 1 清除锁存的标志，本位段自动清零
5	CLKERROST	W1C	0x0	清除由时钟错误触发的一次封锁标志 0：写 0 无效，读总是 0 1：写 1 清除锁存的标志，本位段自动清零
4	TZ4OST	W1C	0x0	清除由 TZ4 触发的一次封锁标志 0：写 0 无效，读总是 0 1：写 1 清除锁存的标志，本位段自动清零
3	TZ3OST	W1C	0x0	清除由 TZ3 触发的一次封锁标志 0：写 0 无效，读总是 0 1：写 1 清除锁存的标志，本位段自动清零
2	TZ2OST	W1C	0x0	清除由 TZ2 触发的一次封锁标志 0：写 0 无效，读总是 0 1：写 1 清除锁存的标志，本位段自动清零
1	TZ1OST	W1C	0x0	清除由 TZ1 触发的一次封锁标志 0：写 0 无效，读总是 0 1：写 1 清除锁存的标志，本位段自动清零
0	TZ0OST	W1C	0x0	清除由 TZ0 触发的一次封锁标志 0：写 0 无效，读总是 0 1：写 1 清除锁存的标志，本位段自动清零

表 8-84：数字比较封锁事件选择寄存器（TZDCSEL）位段定义

TZDCSEL (Trip-Zone Digital Compare Event Select Register) Offset：0x98 Default：0x00000000 Access： PWMx->TZDCSEL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED				RAWDCBEVT1			RAWDCBEVT0
7	6	5	4	3	2	1	0
RAWDCBEVT0		RAWDCAEVT1			RAWDCAEVT0		

表 8-85：数字比较封锁事件选择寄存器（TZDCSEL）位段描述

位段	位段名	属性	复位值	描述
31:12	RESERVED_31_12	RO	0x0	保留
11:9	RAWDCBEVT1	RW	0x0	数字比较封锁原始事件 RAWDCBEVT1 选择

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				0：禁用事件 1：DCBL 为低 2：DCBL 为高 3：DCBH 为低 4：DCBH 为高 5：DCBL 为低且 DCBH 为高 6：DCBL 为高且 DCBH 为低 7：DCBL 为高且 DCBH 为高
8:6	RAWDCBEVT0	RW	0x0	数字比较封锁原始事件 RAWDCBEVT0 选择 0：禁用事件 1：DCBL 为低 2：DCBL 为高 3：DCBH 为低 4：DCBH 为高 5：DCBL 为低且 DCBH 为高 6：DCBL 为高且 DCBH 为低 7：DCBL 为高且 DCBH 为高
5:3	RAWDCAEVT1	RW	0x0	数字比较封锁原始事件 RAWDCAEVT1 选择 0：禁用事件 1：DCAL 为低 2：DCAL 为高 3：DCAH 为低 4：DCAH 为高 5：DCAL 为低且 DCAH 为高 6：DCAL 为高且 DCAH 为低 7：DCAL 为高且 DCAH 为高
2:0	RAWDCAEVT0	RW	0x0	数字比较封锁原始事件 RAWDCAEVT0 选择 0：禁用事件 1：DCAL 为低 2：DCAL 为高 3：DCAH 为低 4：DCAH 为高 5：DCAL 为低且 DCAH 为高 6：DCAL 为高且 DCAH 为低 7：DCAL 为高且 DCAH 为高

SPINTROL

表 8-86：封锁 A 路输出控制寄存器（TZACTL）位段定义

TZACTL (Trip-Zone Output A Control Register) Offset：0x9C Default：0x00000000 Access： PWMx->TZACTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						DCAEVT1D	
15	14	13	12	11	10	9	8
DCAEVT1D	DCAEVT1U			DCAEVT0D			DCAEVT0U
7	6	5	4	3	2	1	0
DCAEVT0U		TZD			TZU		

表 8-87：封锁 A 路输出控制寄存器（TZACTL）位段描述

位段	位段名	属性	复位值	描述
31:18	RESERVED_31_18	RO	0x0	保留
17:15	DCAEVT1D	RW	0x0	DCAEVT1 发生且 TBCNT 递减计数时的封锁输出 0：A 路输出高阻 1：A 路输出置低 2：A 路输出置高 3：A 路输出翻转 7：A 路输出维持原样不变
14:12	DCAEVT1U	RW	0x0	DCAEVT1 发生且 TBCNT 递增计数时的封锁输出 0：A 路输出高阻 1：A 路输出置低 2：A 路输出置高 3：A 路输出翻转 7：A 路输出维持原样不变
11:9	DCAEVT0D	RW	0x0	DCAEVT0 发生且 TBCNT 递减计数时的封锁输出 0：A 路输出高阻 1：A 路输出置低 2：A 路输出置高 3：A 路输出翻转 7：A 路输出维持原样不变
8:6	DCAEVT0U	RW	0x0	DCAEVT0 发生且 TBCNT 递增计数时的封锁输出 0：A 路输出高阻 1：A 路输出置低 2：A 路输出置高

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				3：A 路输出翻转 7：A 路输出维持原样不变
5:3	TZD	RW	0x0	TZ 事件发生且 TBCNT 递减计数时的封锁输出 0：A 路输出高阻 1：A 路输出置低 2：A 路输出置高 3：A 路输出翻转 1xx：A 路输出维持原样不变
2:0	TZU	RW	0x0	TZ 事件发生且 TBCNT 递增计数时的封锁输出 0：A 路输出高阻 1：A 路输出置低 2：A 路输出置高 3：A 路输出翻转 7：A 路输出维持原样不变

表 8-88：封锁 B 路输出控制寄存器（TZBCTL）位段定义

TZBCTL (Trip-Zone Ouput B Control Register)　Offset：0xA0　Default：0x00000000
Access：PWMx->TZBCTL

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						DCBEVT1D	
15	14	13	12	11	10	9	8
DCBEVT1D	DCBEVT1U			DCBEVT0D			DCBEVT0U
7	6	5	4	3	2	1	0
DCBEVT0U		TZD			TZU		

表 8-89：封锁 B 路输出控制寄存器（TZBCTL）位段描述

位段	位段名	属性	复位值	描述
31:18	RESERVED_31_18	RO	0x0	保留
17:15	DCBEVT1D	RW	0x0	DCBEVT1 发生且 TBCNT 递减计数时的封锁输出 0：B 路输出高阻 1：B 路输出置低 2：B 路输出置高 3：B 路输出翻转 7：B 路输出维持原样不变

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
14:12	DCBEVT1U	RW	0x0	DCBEVT1 发生且 TBCNT 递增计数时的封锁输出 0：B 路输出高阻 1：B 路输出置低 2：B 路输出置高 3：B 路输出翻转 7：B 路输出维持原样不变
11:9	DCBEVT0D	RW	0x0	DCBEVT0 发生且 TBCNT 递减计数时的封锁输出 0：B 路输出高阻 1：B 路输出置低 2：B 路输出置高 3：B 路输出翻转 7：B 路输出维持原样不变
8:6	DCBEVT0U	RW	0x0	DCBEVT0 发生且 TBCNT 递增计数时的封锁输出 0：B 路输出高阻 1：B 路输出置低 2：B 路输出置高 3：B 路输出翻转 7：B 路输出维持原样不变
5:3	TZD	RW	0x0	TZ 事件发生且 TBCNT 递减计数时的封锁输出 0：B 路输出高阻 1：B 路输出置低 2：B 路输出置高 3：B 路输出翻转 7：B 路输出维持原样不变
2:0	TZU	RW	0x0	TZ 事件发生且 TBCNT 递增计数时的封锁输出 0：B 路输出高阻 1：B 路输出置低 2：B 路输出置高 3：B 路输出翻转 7：B 路输出维持原样不变

表 8-90：封锁中断标志寄存器（TZIF）位段定义

TZIF (Trip-Zone Interrupt Flag Register) Offset：0xA4 Default：0x00000000 Access： PWMx->TZIF							
31	30	29	28	27	26	25	24

©2025 旋智电子科技（上海）有限公司

RESERVED							
23 RESERVED	22	21 DCBEVT1RAW	20 DCBEVT0RAW	19 DCAEVT1RAW	18 DCAEVT0RAW	17 CBCRAW	16 OSTRAW
15 RESERVED	14	13	12	11	10	9	8
7 RESERVED	6 INT	5 DCBEVT1	4 DCBEVT0	3 DCAEVT1	2 DCAEVT0	1 CBC	0 OST

SPINTROL

表 8-91：封锁中断标志寄存器（TZIF）位段描述

位段	位段名	属性	复位值	描述
31:22	RESERVED_31_22	RO	0x0	保留
21	DCBEVT1RAW	RO	0x0	DCBEVT1 封锁事件标志 0：事件未发生 1：事件已发生
20	DCBEVT0RAW	RO	0x0	DCBEVT0 封锁事件标志 0：事件未发生 1：事件已发生
19	DCAEVT1RAW	RO	0x0	DCAEVT1 封锁事件标志 0：事件未发生 1：事件已发生
18	DCAEVT0RAW	RO	0x0	DCAEVT0 封锁事件标志 0：事件未发生 1：事件已发生
17	CBCRAW	RO	0x0	周期封锁事件标志 0：事件未发生 1：事件已发生
16	OSTRAW	RO	0x0	一次封锁事件标志 0：事件未发生 1：事件已发生
15:7	RESERVED_15_7	RO	0x0	保留
6	INT	RO	0x0	全局封锁中断标志 0：封锁中断未发生 1：封锁中断已发生，并已经通知 CPU 本位段被清除之前不会产生新中断。
5	DCBEVT1	RO	0x0	DCBEVT1 封锁中断标志 0：DCBEVT1 中断未发生 1：DCBEVT1 中断已发生。
4	DCBEVT0	RO	0x0	DCBEVT0 封锁中断标志 0：DCBEVT0 中断未发生 1：DCBEVT0 中断已发生。
3	DCAEVT1	RO	0x0	DCAEVT1 封锁中断标志 0：DCAEVT1 中断未发生 1：DCAEVT1 中断已发生。
2	DCAEVT0	RO	0x0	DCAEVT0 封锁中断标志 0：DCAEVT0 中断未发生

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				1：DCAEVT0 中断已发生。
1	CBC	RO	0x0	周期封锁中断标志 0：周期封锁中断未发生 1：周期封锁中断已发生。
0	OST	RO	0x0	一次封锁事件标志 0：一次封锁中断未发生 1：一次封锁中断已发生。

表 8-92：封锁中断清除寄存器（TZIC）位段定义

TZIC (Trip-Zone Clear Register) Offset：0xA8 Default：0x00000000
Access： PWMx->TZIC

31	30	29	28	27	26	25	24
CBCCLRMODEWE	CBCCLRMODE		RESERVED				
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	INT	DCBEVT1	DCBEVT0	DCAEVT1	DCAEVT0	CBC	OST

表 8-93：封锁中断清除寄存器（TZIC）位段描述

位段	位段名	属性	复位值	描述
31	CBCCLRMODEWE	W1C	0x0	CBCCLRMODE 位段写使能 本位段必须与 CBCCLRMODE 同时写 0：写 0 无效，读总是 0 1：写 1 允许同时写 CBCCLRMODE，本位段自动清零
30:29	CBCCLRMODE	RW	0x0	周期封锁标志自动清除方式 0：TBCNT=0 时自动清除 1：TBCNT=TBPRD 时自动清除 2：TBCNT=0 或 TBCNT=TBPRD 时自动清除 3：不自动清除
28:7	RESERVED_28_7	RO	0x0	保留
6	INT	W1C	0x0	全局封锁中断标志清除 0：写 0 无效，读总是 0 1：写 1 清除中断标志，本位段自动清零。

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
5	DCBEVT1	W1C	0x0	DCBEVT1 事件标志清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
4	DCBEVT0	W1C	0x0	DCBEVT0 事件标志清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
3	DCAEVT1	W1C	0x0	DCAEVT1 事件标志清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
2	DCAEVT0	W1C	0x0	DCAEVT0 事件标志清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
1	CBC	W1C	0x0	周期封锁事件标志手动清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
0	OST	W1C	0x0	一次封锁事件标志清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

表 8-94：封锁中断使能寄存器（TZIE）位段定义

TZIE (Trip-Zone Interrupt Enable Register) Offset：0xAC Default：0x00000000 Access：PWMx->TZIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED		DCBEVT1	DCBEVT0	DCAEVT1	DCAEVT0	CBC	OST

表 8-95：封锁中断使能寄存器（TZIE）位段描述

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
31:6	RESERVED_31_6	RO	0x0	保留
5	DCBEVT1	RW	0x0	DCBEVT1 封锁事件中断使能 0：关闭中断 1：使能中断
4	DCBEVT0	RW	0x0	DCBEVT0 封锁事件中断使能 0：关闭中断 1：使能中断
3	DCAEVT1	RW	0x0	DCAEVT1 封锁事件中断使能 0：关闭中断 1：使能中断
2	DCAEVT0	RW	0x0	DCAEVT0 封锁事件中断使能 0：关闭中断 1：使能中断
1	CBC	RW	0x0	周期封锁事件中断使能 0：关闭中断 1：使能中断
0	OST	RW	0x0	一次封锁事件中断使能 0：关闭中断 1：使能中断

表 8-96：封锁中断强制寄存器（TZFRC）位段定义

TZFRC (Trip-Zone Force Register)　Offset：0xB0　Default：0x00000000
Access：PWMx->TZFRC

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED		DCBEVT1WE	DCBEVT0WE	DCAEVT1WE	DCAEVT0WE	CBCWE	OSTWE
7	6	5	4	3	2	1	0
RESERVED		DCBEVT1	DCBEVT0	DCAEVT1	DCAEVT0	CBC	OST

表 8-97：封锁中断强制寄存器（TZFRC）位段描述

位段	位段名	属性	复位值	描述
31:14	RESERVED_31_14	RO	0x0	保留
13	DCBEVT1WE	W1S	0x0	DCBEVT1 位写使能（必须与 DCBEVT1 同时写）

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
12	DCBEVT0WE	W1S	0x0	DCBEVT0 位写使能（必须与 DCBEVT0 同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
11	DCAEVT1WE	W1S	0x0	DCAEVT1 位写使能（必须与 DCAEVT1 同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
10	DCAEVT0WE	W1S	0x0	DCAEVT0 位写使能（必须与 DCAEVT0 同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
9	CBCWE	W1S	0x0	CBC 位写使能（必须与 CBC 同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
8	OSTWE	W1S	0x0	OST 位写使能（必须与 OST 同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
7:6	RESERVED_7_6	RO	0x0	保留
5	DCBEVT1	W1S	0x0	软件强制 DCBEVT1 封锁事件 0：写 0 无效，读总是 0 1：写 1 产生 DCBEVT1 封锁并置位 TZIF.DCBEVT1 本位段自动清零
4	DCBEVT0	W1S	0x0	软件强制 DCBEVT0 封锁事件 0：写 0 无效，读总是 0 1：写 1 产生 DCBEVT0 封锁并置位 TZIF.DCBEVT0 本位段自动清零
3	DCAEVT1	W1S	0x0	软件强制 DCAEVT1 封锁事件 0：写 0 无效，读总是 0 1：写 1 产生 DCAEVT1 封锁并置位 TZIF.DCAEVT1 本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
2	DCAEVT0	W1S	0x0	软件强制 DCAEVT0 封锁事件 0：写 0 无效，读总是 0 1：写 1 产生 DCAEVT0 封锁并置位 TZIF.DCAEVT0 本位段自动清零
1	CBC	W1S	0x0	软件强制周期封锁事件 0：写 0 无效，读总是 0 1：写 1 产生周期封锁并置位 TZIF.CBC 本位段自动清零
0	OST	W1S	0x0	软件强制一次封锁事件 0：写 0 无效，读总是 0 1：写 1 产生一次封锁并置位 TZIF.OST 本位段自动清零

表 8-98：数字比较 DCAL 事件选择寄存器（DCALTRIPSEL）位段定义

DCALTRIPSEL (Digital Compare AL Trip Select Register)　Offset：0xB4　Default：0x00000000
Access：PWMx->DCALTRIPSEL

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						COMP0H	COMP0L
15	14	13	12	11	10	9	8
RESERVED					ADCPPU2TZ	ADCPPU1TZ	ADCPPU0TZ
7	6	5	4	3	2	1	0
RESERVED	RESERVED	RESERVED	TZ4	TZ3	TZ2	TZ1	TZ0

表 8-99：数字比较 DCAL 事件选择寄存器（DCALTRIPSEL）位段描述

位段	位段名	属性	复位值	描述
31:18	RESERVED_31_18	RO	0x0	保留
17	COMP0H	RW	0x0	选中（逻辑或）COMPH 为 DCAL 事件 0：不选中 1：选中
16	COMP0L	RW	0x0	选中（逻辑或）COMPL 为 DCAL 事件 0：不选中 1：选中
15:11	RESERVED_15_11	RO	0x0	保留
10	ADCPPU2TZ	RW	0x0	选中（逻辑或）ADCPPUTZ2 为 DCAL 事件 0：不选中 1：选中

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
9	ADCPPU1TZ	RW	0x0	选中（逻辑或）ADCPPUTZ1 为 DCAL 事件 0：不选中 1：选中
8	ADCPPU0TZ	RW	0x0	选中（逻辑或）ADCPPUTZ0 为 DCAL 事件 0：不选中 1：选中
7	RESERVED_7	RO	0x0	保留
6	RESERVED_6	RW	0x0	保留
5	RESERVED_5	RW	0x0	保留
4	TZ4	RW	0x0	选中（逻辑或）TZ4 为 DCAL 事件 0：不选中 1：选中
3	TZ3	RW	0x0	选中（逻辑或）TZ3 为 DCAL 事件 0：不选中 1：选中
2	TZ2	RW	0x0	选中（逻辑或）TZ2 为 DCAL 事件 0：不选中 1：选中
1	TZ1	RW	0x0	选中（逻辑或）TZ1 为 DCAL 事件 0：不选中 1：选中
0	TZ0	RW	0x0	选中（逻辑或）TZ0 为 DCAL 事件 0：不选中 1：选中

表 8-100：数字比较 DCAH 事件选择寄存器（DCAHTRIPSEL）位段定义

DCAHTRIPSEL (Digital Compare AH Trip Select Register)　Offset：0xB8　Default：0x00000000
Access：PWMx->DCAHTRIPSEL

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						COMP0H	COMP0L
15	14	13	12	11	10	9	8
RESERVED					ADCPPU2TZ	ADCPPU1TZ	ADCPPU0TZ
7	6	5	4	3	2	1	0
RESERVED	RESERVED	RESERVED	TZ4	TZ3	TZ2	TZ1	TZ0

表 8-101：数字比较 DCAH 事件选择寄存器（DCAHTRIPSEL）位段描述

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
31:18	RESERVED_31_18	RO	0x0	保留
17	COMP0H	RW	0x0	选中（逻辑或）COMPH 为 DCAH 事件 0：不选中 1：选中
16	COMP0L	RW	0x0	选中（逻辑或）COMPL 为 DCAH 事件 0：不选中 1：选中
15:11	RESERVED_15_11	RO	0x0	保留
10	ADCPPU2TZ	RW	0x0	选中（逻辑或）ADCPPUTZ2 为 DCAH 事件 0：不选中 1：选中
9	ADCPPU1TZ	RW	0x0	选中（逻辑或）ADCPPUTZ1 为 DCAH 事件 0：不选中 1：选中
8	ADCPPU0TZ	RW	0x0	选中（逻辑或）ADCPPUTZ0 为 DCAH 事件 0：不选中 1：选中
7	RESERVED_7	RO	0x0	保留
6	RESERVED_6	RW	0x0	保留
5	RESERVED_5	RW	0x0	保留
4	TZ4	RW	0x0	选中（逻辑或）TZ4 为 DCAH 事件 0：不选中 1：选中
3	TZ3	RW	0x0	选中（逻辑或）TZ3 为 DCAH 事件 0：不选中 1：选中
2	TZ2	RW	0x0	选中（逻辑或）TZ2 为 DCAH 事件 0：不选中 1：选中
1	TZ1	RW	0x0	选中（逻辑或）TZ1 为 DCAH 事件 0：不选中 1：选中
0	TZ0	RW	0x0	选中（逻辑或）TZ0 为 DCAH 事件 0：不选中 1：选中

表 8-102：数字比较 DCBL 事件选择寄存器（DCBLTRIPSEL）位段定义

DCBLTRIPSEL (Digital Compare BL Trip Select Register)　Offset：0xBC　Default：0x00000000
Access：PWMx->DCBLTRIPSEL

©2025 旋智电子科技（上海）有限公司

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						COMP0H	COMP0L
15	14	13	12	11	10	9	8
RESERVED					ADCPPU2TZ	ADCPPU1TZ	ADCPPU0TZ
7	6	5	4	3	2	1	0
RESERVED	RESERVED	RESERVED	TZ4	TZ3	TZ2	TZ1	TZ0

表 8-103：数字比较 DCBL 事件选择寄存器（DCBLTRIPSEL）位段描述

位段	位段名	属性	复位值	描述
31:18	RESERVED_31_18	RO	0x0	保留
17	COMP0H	RW	0x0	选中（逻辑或）COMPH 为 DCBL 事件 0：不选中 1：选中
16	COMP0L	RW	0x0	选中（逻辑或）COMPL 为 DCBL 事件 0：不选中 1：选中
15:11	RESERVED_15_11	RO	0x0	保留
10	ADCPPU2TZ	RW	0x0	选中（逻辑或）ADCPPUTZ2 为 DCBL 事件 0：不选中 1：选中
9	ADCPPU1TZ	RW	0x0	选中（逻辑或）ADCPPUTZ1 为 DCBL 事件 0：不选中 1：选中
8	ADCPPU0TZ	RW	0x0	选中（逻辑或）ADCPPUTZ0 为 DCBL 事件 0：不选中 1：选中
7	RESERVED_7	RO	0x0	保留
6	RESERVED_6	RW	0x0	保留
5	RESERVED_5	RW	0x0	保留
4	TZ4	RW	0x0	选中（逻辑或）TZ4 为 DCBL 事件 0：不选中 1：选中
3	TZ3	RW	0x0	选中（逻辑或）TZ3 为 DCBL 事件 0：不选中 1：选中

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
2	TZ2	RW	0x0	选中（逻辑或）TZ2 为 DCBL 事件 0：不选中 1：选中
1	TZ1	RW	0x0	选中（逻辑或）TZ1 为 DCBL 事件 0：不选中 1：选中
0	TZ0	RW	0x0	选中（逻辑或）TZ0 为 DCBL 事件 0：不选中 1：选中

表 8-104：数字比较 DCBH 事件选择寄存器（DCBHTRIPSEL）位段定义

DCBHTRIPSEL (Digital Compare BH Trip Select Register) Offset：0xC0 Default：0x00000000 Access： PWMx->DCBHTRIPSEL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						COMP0H	COMP0L
15	14	13	12	11	10	9	8
RESERVED					ADCPPU2TZ	ADCPPU1TZ	ADCPPU0TZ
7	6	5	4	3	2	1	0
RESERVED	RESERVED	RESERVED	TZ4	TZ3	TZ2	TZ1	TZ0

表 8-105：数字比较 DCBH 事件选择寄存器（DCBHTRIPSEL）位段描述

位段	位段名	属性	复位值	描述
31:18	RESERVED_31_18	RO	0x0	保留
17	COMP0H	RW	0x0	选中（逻辑或）COMPH 为 DCBH 事件 0：不选中 1：选中
16	COMP0L	RW	0x0	选中（逻辑或）COMPL 为 DCBH 事件 0：不选中 1：选中
15:11	RESERVED_15_11	RO	0x0	保留
10	ADCPPU2TZ	RW	0x0	选中（逻辑或）ADCPPUTZ2 为 DCBH 事件 0：不选中 1：选中
9	ADCPPU1TZ	RW	0x0	选中（逻辑或）ADCPPUTZ1 为 DCBH 事件 0：不选中 1：选中

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
8	ADCPPU0TZ	RW	0x0	选中（逻辑或）ADCPPUTZ0 为 DCBH 事件 0：不选中 1：选中
7	RESERVED_7	RO	0x0	保留
6	RESERVED_6	RW	0x0	保留
5	RESERVED_5	RW	0x0	保留
4	TZ4	RW	0x0	选中（逻辑或）TZ4 为 DCBH 事件 0：不选中 1：选中
3	TZ3	RW	0x0	选中（逻辑或）TZ3 为 DCBH 事件 0：不选中 1：选中
2	TZ2	RW	0x0	选中（逻辑或）TZ2 为 DCBH 事件 0：不选中 1：选中
1	TZ1	RW	0x0	选中（逻辑或）TZ1 为 DCBH 事件 0：不选中 1：选中
0	TZ0	RW	0x0	选中（逻辑或）TZ0 为 DCBH 事件 0：不选中 1：选中

表 8-106：数字比较 A 控制寄存器（DCACTL）位段定义

DCACTL (Digital Compare A Control Register)　Offset：0xC4　Default：0x00000000
Access：PWMx->DCACTL

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED		EVT1SRCASYNC	EVT1SRCSEL	EVT0SYNCE	EVT0SOCE	EVT0SRCASYNC	EVT0SRCSEL

表 8-107：数字比较 A 控制寄存器（DCACTL）位段描述

位段	位段名	属性	复位值	描述
31:6	RESERVED_31_6	RO	0x0	保留

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
5	EVT1SRCASYNC	RW	0x0	DCAEVT1 源信号通路选择 0：同步通路 1：异步通路
4	EVT1SRCSEL	RW	0x0	DCAEVT1 最终信号选择 0：选择 RAWDCAEVT1 1：选择 DCEVTFILT
3	EVT0SYNCE	RW	0x0	DCAEVT0 SYNC 信号产生使能 0：不产生 DCAEVT0 SYNC 信号 1：产生 DCAEVT0 SYNC 信号
2	EVT0SOCE	RW	0x0	DCAEVT0 SOC 信号产生使能 0：不产生 DCAEVT0 SOC 信号 1：产生 DCAEVT0 SOC 信号
1	EVT0SRCASYNC	RW	0x0	DCAEVT0 源信号通路选择 0：同步通路 1：异步通路
0	EVT0SRCSEL	RW	0x0	DCAEVT0 最终信号选择 0：选择 RAWDCAEVT0 1：选择 DCEVTFILT

表 8-108：数字比较 B 控制寄存器（DCBCTL）位段定义

DCBCTL (Digital Compare B Control Register)　Offset：0xC8　Default：0x00000000
Access：PWMx->DCBCTL

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED		EVT1SRCASYNC	EVT1SRCSEL	EVT0SYNCE	EVT0SOCE	EVT0SRCASYNC	EVT0SRCSEL

表 8-109：数字比较 B 控制寄存器（DCBCTL）位段描述

位段	位段名	属性	复位值	描述
31:6	RESERVED_31_6	RO	0x0	保留
5	EVT1SRCASYNC	RW	0x0	DCBEVT1 源信号通路选择 0：同步通路 1：异步通路

SPINTROL

位段	位段名	属性	复位值	描述
4	EVT1SRCSEL	RW	0x0	DCBEVT1 最终信号选择 0：选择 RAWDCBEVT1 1：选择 DCEVTFILT
3	EVT0SYNCE	RW	0x0	DCBEVT0 SYNC 信号产生使能 0：不产生 DCBEVT0 SYNC 信号 1：产生 DCBEVT0 SYNC 信号
2	EVT0SOCE	RW	0x0	DCBEVT0 SOC 信号产生使能 0：不产生 DCBEVT0 SOC 信号 1：产生 DCBEVT0 SOC 信号
1	EVT0SRCASYNC	RW	0x0	DCBEVT0 源信号通路选择 0：同步通路 1：异步通路
0	EVT0SRCSEL	RW	0x0	DCBEVT0 最终信号选择 0：选择 RAWDCBEVT0 1：选择 DCEVTFILT

表 8-110：数字比较消隐滤波控制寄存器（DCFCTL）位段定义

DCFCTL (Digital Compare Filter Register)　Offset：0xCC　Default：0x00000010
Access：PWMx->DCFCTL

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						WIN11EN	WIN10EN
15	14	13	12	11	10	9	8
WIN9EN	WIN8EN	WIN7EN	WIN6EN	WIN5EN	WIN4EN	WIN3EN	WIN2EN
7	6	5	4	3	2	1	0
WIN1EN	WIN0EN	PULSESEL		BLANKINV	BLANKEN	SRCSEL	

表 8-111：数字比较消隐滤波控制寄存器（DCFCTL）位段描述

位段	位段名	属性	复位值	描述
31:18	RESERVED_31_18	RO	0x0	保留
17	WIN11EN	RW	0x0	使能 PWM11 产生的窗口对本级 PWM 信号消隐 0：关闭 1：使能（逻辑或）
16	WIN10EN	RW	0x0	使能 PWM10 产生的窗口对本级 PWM 信号消隐

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				0：关闭 1：使能（逻辑或）
15	WIN9EN	RW	0x0	使能 PWM9 产生的窗口对本级 PWM 信号消隐 0：关闭 1：使能（逻辑或）
14	WIN8EN	RW	0x0	使能 PWM8 产生的窗口对本级 PWM 信号消隐 0：关闭 1：使能（逻辑或）
13	WIN7EN	RW	0x0	使能 PWM7 产生的窗口对本级 PWM 信号消隐 0：关闭 1：使能（逻辑或）
12	WIN6EN	RW	0x0	使能 PWM6 产生的窗口对本级 PWM 信号消隐 0：关闭 1：使能（逻辑或）
11	WIN5EN	RW	0x0	使能 PWM5 产生的窗口对本级 PWM 信号消隐 0：关闭 1：使能（逻辑或）
10	WIN4EN	RW	0x0	使能 PWM4 产生的窗口对本级 PWM 信号消隐 0：关闭 1：使能（逻辑或）
9	WIN3EN	RW	0x0	使能 PWM3 产生的窗口对本级 PWM 信号消隐 0：关闭 1：使能（逻辑或）
8	WIN2EN	RW	0x0	使能 PWM2 产生的窗口对本级 PWM 信号消隐 0：关闭 1：使能（逻辑或）
7	WIN1EN	RW	0x0	使能 PWM1 产生的窗口对本级 PWM 信号消隐 0：关闭 1：使能（逻辑或）

SPINTROL

位段	位段名	属性	复位值	描述
6	WIN0EN	RW	0x0	使能 PWM0 产生的窗口对本级 PWM 信号消隐 0：关闭 1：使能（逻辑或）
5:4	PULSESEL	RW	0x1	消隐和捕捉脉冲起始对齐参考点选择 00：对齐到 TBCNT=0 01：对齐到 TBCNT=TBPRD 10：对齐到 TBCNT=0 或 TBCNT=TBPRD 11：无效选项
3	BLANKINV	RW	0x0	消隐窗口反相 0：保持原始效应窗口极性 1：消隐窗口取反
2	BLANKEN	RW	0x0	消隐窗口使能 0：关闭消隐窗口 1：使能消隐窗口
1:0	SRCSEL	RW	0x0	DCEVTFILT 消隐滤波信号源选择 00：对 RAWDCAEVT0 消隐滤波 01：对 RAWDCAEVT1 消隐滤波 10：对 RAWDCBEVT0 消隐滤波 11：对 RAWDCBEVT1 消隐滤波

表 8-112：数字比较消隐窗口偏移量寄存器（DCFOFFSET）位段定义

DCFOFFSET (Digital Compare Filter Offset Register) Offset：0xD0 Default：0x00000000 Access：PWMx->DCFOFFSET							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 8-113：数字比较消隐窗口偏移量寄存器（DCFOFFSET）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
15:0	VAL	RW	0x0	消隐窗口偏移量 定义了从 DCFCTL.PULSESEL 指定的脉冲到消隐窗口开始的偏移量（以时基时钟 TBCLK 的周期数为单位）。

表 8-114：数字比较消隐窗口偏移计数值寄存器（DCFOFFSETCNT）位段定义

DCFOFFSETCNT (Digital Compare Filter Offset Counter Register) Offset：0xD4 Default：0x00000000 Access：PWMx->DCFOFFSETCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

SPINTROL

表 8-115：数字比较消隐窗口偏移计数值寄存器（DCFOFFSETCNT）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RO	0x0	消隐窗口偏移计数器 本位段只读，它记录了当前偏移计数值。由 DCFCTL.PULSESEL 指定的事件发生后，该计数器从 DCFOFFSET 开始递减计数一直到 0，然后维持，直到 DCFCTL.PULSESEL 指定的事件再次发生后，该计数器又从 DCFOFFSET 开始递减计数。

表 8-116：数字比较消隐窗口大小寄存器（DCFWINDOW）位段定义

DCFWINDOW (Digital Compare Filter Window Register)　Offset：0xD8　Default：0x00000000
Access：PWMx->DCFWINDOW

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 8-117：数字比较消隐窗口大小寄存器（DCFWINDOW）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x0	以 TBCLK 周期度量的消隐窗口大小 当消隐窗口偏移计数器计到 0 后，消隐窗口开启，消隐窗口大小计数器开始递减计数，待递减计数到 0 后消隐窗口关闭。若消隐窗口开启时偏移计数器又计数到 0，则立刻重新开启新的消隐窗口。 消隐窗口允许跨 PWM 时基波形周期边界。 当本位段为 0 时，禁用消隐窗口。

©2025 旋智电子科技（上海）有限公司

表 8-118：数字比较消隐窗口大小计数寄存器（DCFWINDOWCNT）位段定义

DCFWINDOWCNT (Digital Compare Filter Window Counter Register) Offset：0xDC Default：0x00000000 Access：PWMx->DCFWINDOWCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 8-119：数字比较消隐窗口大小计数寄存器（DCFWINDOWCNT）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RO	0x0	消隐窗口大小计数器 本位段只读，它记录了当前消隐窗口大小计数值。消隐窗口偏移计数器计数到 0 后，该计数器从 DCFWINDOW 开始递减计数，一直到 0 并停止，直到消隐窗口偏移计数器再次递减计数到 0，该计数器又再次从 DCFWINDOW 开始递减计数。

表 8-120：数字比较时基计数捕获控制寄存器（DCCAPCTL）位段定义

DCCAPCTL (Digital Compare Capture Control Register) Offset：0xE0 Default：0x00000000 Access：PWMx->DCCAPCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						CAPFLG	CAPCLR
7	6	5	4	3	2	1	0
RESERVED	MULTICAPWE	DIRECTWE	CAPENWE	RESERVED	MULTICAP	DIRECT	CAPEN

表 8-121：数字比较时基计数捕获控制寄存器（DCCAPCTL）位段描述

位段	位段名	属性	复位值	描述
31:10	RESERVED_31_10	RO	0x0	保留

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
9	CAPFLG	RO	0x0	锁存的捕获状态 0：数字比较时基计数捕获未发生 1：数字比较时基计数捕获发生
8	CAPCLR	W1C	0x0	清除锁存的捕获状态 0：写 0 无效，读总是 0 1：写 1 清除锁存的状态，该位段自动清零
7	RESERVED_7	RO	0x0	保留
6	MULTICAPWE	W1S	0x0	MULTICAP 写使能（必须与 MULTICAP 同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
5	DIRECTWE	W1S	0x0	DIRECT 位写使能（必须与 DIRECT 位同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
4	CAPENWE	W1S	0x0	CAPEN 位写使能（必须与 CAPEN 位同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
3	RESERVED_3	RO	0x0	保留
2	MULTICAP	RW	0x0	多次捕获模式 0：在每个周期内只捕获一次，下一次捕获要等到 DCFCTL.PULSESEL 定义的脉冲之后 1：每次事件发生时都会捕获，并不断覆盖旧值
1	DIRECT	RW	0x0	DCCAP 捕获值更新方式 0：影子模式 由 DCFCTL.PULSESEL 指定的 TBCNT=TBPRD 或 TBCNT=0 事件发生时，将更新 DCCAP 捕获的最新值更新到其影子寄存器。读取 DCCAP 总是返回影子值。 1：直接模式 读取 DCCAP 寄存器时总是返回最新有效值。
0	CAPEN	RW	0x0	TBCNT 时基计数捕获使能 0：关闭捕获 1：使能捕获

©2025 旋智电子科技（上海）有限公司

表 8-122：数字比较时基计数捕获值寄存器（DCCAP）位段定义

DCCAP (Digital Compare Counter Capture Register) Offset：0xE4 Default：0x00000000 Access：PWMx->DCCAP							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 8-123：数字比较时基计数捕获值寄存器（DCCAP）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RO	0x0	DCEVTFLT 上升沿时捕获的时基计数器的值 在影子模式下，捕获的值在 DCFCTL.PULSESEL 指定的 TBCNT=TBPRD 或 TBCNT=0 事件时更新到影子寄存器。读取 DCCAP 寄存器总是返回影子值。 在直接模式下，读取 DCCAP 寄存器总是返回最新捕获的有效值。

表 8-124：事件触发控制寄存器（ETCTL）位段定义

ETCTL (Event-Trigger Control Register) Offset：0xE8 Default：0x1E1E1E1E Access：PWMx->ETCTL							
31	30	29	28	27	26	25	24
RESERVED			SOCCSEL				SOCCEN
23	22	21	20	19	18	17	16
RESERVED			SOCBSEL				SOCBEN
15	14	13	12	11	10	9	8
RESERVED			SOCASEL				SOCAEN
7	6	5	4	3	2	1	0
RESERVED			INTSEL				INTEN

©2025 旋智电子科技（上海）有限公司

表 8-125：事件触发控制寄存器（ETCTL）位段描述

位段	位段名	属性	复位值	描述
31:29	RESERVED_31_29	RO	0x0	保留
28:25	SOCCSEL	RW	0xF	SOCC 事件选择 0：TBCNT=0 时触发 1：TBCNT=TBPRD 时触发 2：TBCNT=0 或 TBCNT=TBPRD 时触发事件 3：TBCNT=CMPA 且 TBCNT 递增计数时触发 4：TBCNT=CMPA 且 TBCNT 递减计数时触发 5：TBCNT=CMPB 且 TBCNT 递增计数时触发 6：TBCNT=CMPB 且 TBCNT 递减计数时触发 7：TBCNT=CMPC 且 TBCNT 递增计数时触发 8：TBCNT=CMPC 且 TBCNT 递减计数时触发 9：TBCNT=CMPD 且 TBCNT 递增计数时触发 10：TBCNT=CMPD 且 TBCNT 递减计数时触发 11：DCAEVT0.SOC 事件触发 12：DCBEVT0.SOC 事件触发 其他：不产生（禁止）SOCC 事件
24	SOCCEN	RW	0x0	ADC 采样开始脉冲 C（SOCC）产生使能 0：关闭 1：使能
23:21	RESERVED_23_21	RO	0x0	保留
20:17	SOCBSEL	RW	0xF	SOCB 事件选择 0：TBCNT=0 时触发 1：TBCNT=TBPRD 时触发 2：TBCNT=0 或 TBCNT=TBPRD 时触发事件 3：TBCNT=CMPA 且 TBCNT 递增计数时触发 4：TBCNT=CMPA 且 TBCNT 递减计数时触发 5：TBCNT=CMPB 且 TBCNT 递增计数时触发 6：TBCNT=CMPB 且 TBCNT 递减计数时触发 7：TBCNT=CMPC 且 TBCNT 递增计数时触发 8：TBCNT=CMPC 且 TBCNT 递减计数时触发 9：TBCNT=CMPD 且 TBCNT 递增计数时触发 10：TBCNT=CMPD 且 TBCNT 递减计数时触发 11：DCAEVT0.SOC 事件触发 12：DCBEVT0.SOC 事件触发 其他：不产生（禁止）SOCB 事件
16	SOCBEN	RW	0x0	ADC 采样开始脉冲 B（SOCB）产生使能 0：关闭 1：使能
15:13	RESERVED_15_13	RO	0x0	保留

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
12:9	SOCASEL	RW	0xF	SOCA 事件选择 0：TBCNT=0 时触发 1：TBCNT=TBPRD 时触发 2：TBCNT=0 或 TBCNT=TBPRD 时触发事件 3：TBCNT=CMPA 且 TBCNT 递增计数时触发 4：TBCNT=CMPA 且 TBCNT 递减计数时触发 5：TBCNT=CMPB 且 TBCNT 递增计数时触发 6：TBCNT=CMPB 且 TBCNT 递减计数时触发 7：TBCNT=CMPC 且 TBCNT 递增计数时触发 8：TBCNT=CMPC 且 TBCNT 递减计数时触发 9：TBCNT=CMPD 且 TBCNT 递增计数时触发 10：TBCNT=CMPD 且 TBCNT 递减计数时触发 11：DCAEVT0.SOC 事件触发 12：DCBEVT0.SOC 事件触发 其他：不产生（禁止）SOCA 事件
8	SOCAEN	RW	0x0	ADC 采样开始脉冲 A（SOCA）产生使能 0：关闭 1：使能
7:5	RESERVED_7_5	RO	0x0	保留
4:1	INTSEL	RW	0xF	中断事件选择 0：TBCNT=0 时触发 1：TBCNT=TBPRD 时触发 2：TBCNT=0 或 TBCNT=TBPRD 时触发事件 3：TBCNT=CMPA 且 TBCNT 递增计数时触发 4：TBCNT=CMPA 且 TBCNT 递减计数时触发 5：TBCNT=CMPB 且 TBCNT 递增计数时触发 6：TBCNT=CMPB 且 TBCNT 递减计数时触发 7：TBCNT=CMPC 且 TBCNT 递增计数时触发 8：TBCNT=CMPC 且 TBCNT 递减计数时触发 9：TBCNT=CMPD 且 TBCNT 递增计数时触发 10：TBCNT=CMPD 且 TBCNT 递减计数时触发 11：DCAEVT0.SOC 事件触发 12：DCBEVT0.SOC 事件触发 其他：不产生（禁止）中断事件
0	INTEN	RW	0x0	中断脉冲产生使能 0：关闭 1：使能

SPINTROL

表 8-126：事件触发预分频寄存器（ETPS）位段定义

ETPS (Event-Trigger Prescale Register) Offset：0xEC Default：0x00000000 Access： PWMx->ETPS							
31	30	29	28	27	26	25	24
SOCCCNT				SOCCPRD			
23	22	21	20	19	18	17	16
SOCBCNT				SOCBPRD			
15	14	13	12	11	10	9	8
SOCACNT				SOCAPRD			
7	6	5	4	3	2	1	0
INTCNT				INTPRD			

表 8-127：事件触发预分频寄存器（ETPS）位段描述

位段	位段名	属性	复位值	描述
31:28	SOCCCNT	RW	0x0	SOCC 事件计数 该位段记录了 ETCTL.SOCCSEL 指定的 SOCC 事件发生的次数。当 SOCC 脉冲产生时，该位段自动清零。 0：事件未发生 1：事件发生了 1 次 2：事件发生了 2 次 …… 15：事件发生了 15 次
27:24	SOCCPRD	RW	0x0	SOCC 事件周期 当 ETCTL.SOCCEN = 1 且 SOCCCNT = SOCCPRD 时产生 0：禁用 SOCC 计数器且不产生 SOCC 事件 1：当 ETPS.SOCCCNT=1（每次）时产生 SOCC 2：当 ETPS.SOCCCNT=2（每 2 次）时产生 SOCC …… 15：当 ETPS.SOCCCNT=15（每 15 次）时产生 SOCC
23:20	SOCBCNT	RW	0x0	SOCB 事件计数 该位段记录了 ETCTL.SOCBSEL 指定的 SOCB 事件发生的次数。当 SOCB 脉冲产生时，该位段自动清零。 0：事件未发生 1：事件发生了 1 次 2：事件发生了 2 次

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				…… 15：事件发生了 15 次
19:16	SOCBPRD	RW	0x0	SOCB 事件周期 当 ETCTL.SOCBEN = 1 且 SOCBCNT = SOCBPRD 时产生 0：禁用 SOCB 计数器且不产生 SOCB 事件 1：当 ETPS.SOCBCNT=1（每次）时产生 SOCB 2：当 ETPS.SOCBCNT=2（每 2 次）时产生 SOCB …… 15：当 ETPS.SOCBCNT=15（每 15 次）时产生 SOCB
15:12	SOCACNT	RW	0x0	SOCA 事件计数 该位段记录了 ETCTL.SOCASEL 指定的 SOCA 事件发生的次数。当 SOCA 脉冲产生时，该位段自动清零。 0：事件未发生 1：事件发生了 1 次 2：事件发生了 2 次 …… 15：事件发生了 15 次
11:8	SOCAPRD	RW	0x0	SOCA 事件周期 ETCTL.SOCAEN = 1 且 SOCACNT = SOCAPRD 时产生 0：禁用 SOCA 计数器且不产生 SOCA 事件 1：当 ETPS.SOCACNT=1（每次）时产生 SOCA 2：当 ETPS.SOCACNT=2（每 2 次）时产生 SOCA …… 15：当 ETPS.SOCACNT=15（每 15 次）时产生 SOCA
7:4	INTCNT	RW	0x0	中断事件计数 该位段记录了 ETCTL.INTSEL 指定的中断事件发生的次数。当中断脉冲产生时，该位段自动清零。 0：事件未发生 1：事件发生了 1 次 2：事件发生了 2 次

SPINTROL

位段	位段名	属性	复位值	描述
				…… 15：事件发生了 15 次
3:0	INTPRD	RW	0x0	中断事件周期 ETCTL.INTEN = 1 且 INTCNT = INTPRD 时产生中断脉冲 0：禁用中断计数器且不产生中断事件 1：当 ETPS.INTCNT=1（每次）时产生中断脉冲 2：当 ETPS.INTCNT=2（每 2 次）时产生中断脉冲 …… 15：当 ETPS.INTCNT=15（每 15 次）时产生中断脉冲

表 8-128：事件触发标志寄存器（ETFLG）位段定义

ETFLG (Event-Trigger Flag Register) Offset：0xF0 Default：0x00000000 Access：PWMx->ETFLG							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				SOCC	SOCB	SOCA	INT

表 8-129：事件触发标志寄存器（ETFLG）位段描述

位段	位段名	属性	复位值	描述
31:4	RESERVED_31_4	RO	0x0	保留
3	SOCC	RO	0x0	送至 ADC 的采样开始脉冲 SOCC 发生的标志 0：SOCC 脉冲未产生 1：SOCC 脉冲产生 本位段置 1 时不会阻止后续 SOCC 脉冲产生。
2	SOCB	RO	0x0	送至 ADC 的采样开始脉冲 SOCB 发生的标志 0：SOCB 脉冲未产生 1：SOCB 脉冲产生 本位段置 1 时不会阻止后续 SOCB 脉冲产生。

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
1	SOCA	RO	0x0	送至 ADC 的采样开始脉冲 SOCA 发生的标志 0：SOCA 脉冲未产生 1：SOCA 脉冲产生 本位段置 1 时不会阻止后续 SOCA 脉冲产生。
0	INT	RO	0x0	送至 CPU 的中断发生的标志 0：中断脉冲未产生 1：中断脉冲产生 本位段置 1 时将阻止后续中断脉冲发生。

表 8-130：事件触发标志清除寄存器（ETCLR）位段描述

ETCLR (Event-Trigger Clear Register) Offset：0xF4 Default：0x00000000 Access：PWMx->ETCLR							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				SOCC	SOCB	SOCA	INT

表 8-131：事件触发标志清除寄存器（ETCLR）位段描述

位段	位段名	属性	复位值	描述
31:4	RESERVED_31_4	RO	0x0	保留
3	SOCC	W1S	0x0	SOCC 标志清除 0：写 0 无效，读总是 0 1：写 1 清除 ETFLG.SOCC 标志，本位段自动清零
2	SOCB	W1S	0x0	SOCB 标志清除 0：写 0 无效，读总是 0 1：写 1 清除 ETFLG.SOCB 标志，本位段自动清零
1	SOCA	W1S	0x0	SOCA 标志清除 0：写 0 无效，读总是 0 1：写 1 清除 ETFLG.SOCA 标志，本位段自动清零
0	INT	W1S	0x0	中断标志清除

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				0：写 0 无效，读总是 0 1：写 1 清除 ETFLG.INT 标志，本位段自动清零

SPINTROL

©2025 旋智电子科技（上海）有限公司

表 8-132：事件触发软件强制寄存器（ETFRC）位段定义

ETFRC (Event-Trigger Force Register) Offset：0xF8 Default：0x00000000 Access： PWMx->ETFRC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED				SOCCWE	SOCBWE	SOCAWE	INTWE
7	6	5	4	3	2	1	0
RESERVED				SOCC	SOCB	SOCA	INT

表 8-133：事件触发软件强制寄存器（ETFRC）位段描述

位段	位段名	属性	复位值	描述
31:12	RESERVED_31_12	RO	0x0	保留
11	SOCCWE	W1S	0x0	SOCC 位写使能（必须与 SOCC 位同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
10	SOCBWE	W1S	0x0	SOCB 位写使能（必须与 SOCB 位同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
9	SOCAWE	W1S	0x0	SOCA 位写使能（必须与 SOCA 位同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
8	INTWE	W1S	0x0	INT 位写使能（必须与 INT 位同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
7:4	RESERVED_7_4	RO	0x0	保留
3	SOCC	W1S	0x0	软件强制 SOCC 事件 ETCTL.SOCCEN=0 或 ETPS.SOCCPRD=0 时位段无效 0：写 0 无效，读总是 0 1：写 1 强制产生一次 SOCC 事件，位段自动清零
2	SOCB	W1S	0x0	软件强制 SOCB 事件 ETCTL.SOCBEN=0 或 ETPS.SOCBPRD=0 时位段无效 0：写 0 无效，读总是 0 1：写 1 强制产生一次 SOCB 事件，位段自动清零

SPINTROL

位段	位段名	属性	复位值	描述
1	SOCA	W1S	0x0	软件强制 SOCA 事件 ETCTL.SOCAEN=0 或 ETPS.SOCAPRD=0 时位段无效 0：写 0 无效，读总是 0 1：写 1 强制产生一次 SOCA 事件，位段自动清零
0	INT	W1S	0x0	软件强制中断事件 ETCTL.INTEN=0 或 ETPS.INTPRD=0 时本位段无效 0：写 0 无效，读总是 0 1：写 1 强制产生一次中断事件，位段自动清零

表 8-134：PWM 模块写使能寄存器（PWMREGKEY）位段定义

PWMREGKEY (PWM Register Write-Allow Key Register) Offset：0xFC Default：0x1ACCE551 Access： PWMx->PWMREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

表 8-135：PWM 模块写使能寄存器（PWMREGKEY）位段描述

位段	位段名	属性	复位值	描述
31:0	KEY	RW	0x1ACCE551	写入 0x1ACCE551 解锁受保护的 PWM 寄存器

SPINTROL

©2025 旋智电子科技（上海）有限公司

8.11.3 PWMCFG 寄存器列表

表 8-136：PWMCFG 模块基地址

外设模块	基地址
PWMCFG	0x4000 2F00

表 8-137：PWMCFG 寄存器列表

寄存器	偏移地址	描述	复位值
TZ0SRCCTL*	0x00	TZ0 信号源控制寄存器	0x00000001
TZ1SRCCTL*	0x04	TZ1 信号源控制寄存器	0x00000001
TZ2SRCCTL*	0x08	TZ2 信号源控制寄存器	0x00000001
TZ3SRCCTL*	0x0C	TZ3 信号源控制寄存器	0x00000001
TZ4SRCCTL*	0x10	TZ4 信号源控制寄存器	0x00000001
FRCSYNC	0x14	全局 PWM 软件强制同步寄存器	0x00000000
GPIOSYNCICTL*	0x18	GPIO 触发 PWMSYNCI 信号源控制寄存器	0x00000001
GPIOSYNCIEN*	0x1C	GPIO 触发 PWMSYNCI 使能寄存器	0x00000000
TMR0SYNCIEN*	0x20	TIMER0 触发 PWMSYNCI 使能寄存器	0x00000000
TMR1SYNCIEN*	0x24	TIMER1 触发 PWMSYNCI 使能寄存器	0x00000000
TMR2SYNCIEN*	0x28	TIMER2 触发 PWMSYNCI 使能寄存器	0x00000000
SYNCOCTL*	0x2C	PWMSYNCO 输出控制寄存器	0x00000007
SOCAOCTL*	0x30	PWMSOCA 输出控制寄存器	0x00000007
SOCBOCTL*	0x34	PWMSOCB 输出控制寄存器	0x00000007
SOCCOCTL*	0x38	PWMSOCC 输出控制寄存器	0x00000007
PWMCFGREGKEY	0x3C	PWMCFG 模块写使能寄存器	0x1ACCE551

注意： 由*标记的寄存器仅当 PWMCFGREGKEY=0x1ACCE551 时才可以改写。

©2025 旋智电子科技（上海）有限公司

8.11.4 PWMCFG 寄存器

表 8-138：TZ0 信号源控制寄存器（TZ0SRCCTL）位段定义

TZ0SRCCTL (TZ0 Source Control Register) Offset：0x0 Default：0x00000001 Access：PWMCFG->TZ0SRCCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	IOSEL						POL

表 8-139：TZ0 信号源控制寄存器（TZ0SRCCTL）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6:1	IOSEL	RW	0x0	触发 TZ0 的 GPIO 引脚编号
0	POL	RW	0x1	TZ0 极性 0：GPIO 输入低时触发 TZ0 1：GPIO 输入高时触发 TZ0

表 8-140：TZ1 信号源控制寄存器（TZ1SRCCTL）位段定义

TZ1SRCCTL (TZ1 Source Control Register) Offset：0x4 Default：0x00000001 Access：PWMCFG->TZ1SRCCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	IOSEL						POL

表 8-141：TZ1 信号源控制寄存器（TZ1SRCCTL）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6:1	IOSEL	RW	0x0	触发 TZ1 的 GPIO 引脚编号

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
0	POL	RW	0x1	TZ1 极性 0：GPIO 输入低时触发 TZ1 1：GPIO 输入高时触发 TZ1

表 8-142：TZ2 信号源控制寄存器（TZ2SRCCTL）位段定义

TZ2SRCCTL (TZ2 Source Control Register) Offset：0x8 Default：0x00000001 Access：PWMCFG->TZ2SRCCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	IOSEL						POL

表 8-143：TZ2 信号源控制寄存器（TZ2SRCCTL）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6:1	IOSEL	RW	0x0	触发 TZ2 的 GPIO 引脚编号
0	POL	RW	0x1	TZ2 极性 0：GPIO 输入低时触发 TZ2 1：GPIO 输入高时触发 TZ2

表 8-144：TZ3 信号源控制寄存器（TZ3SRCCTL）位段定义

TZ3SRCCTL (TZ3 Source Control Register) Offset：0xC Default：0x00000001 Access：PWMCFG->TZ3SRCCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	IOSEL						POL

SPINTROL

表 8-145：TZ3 信号源控制寄存器（TZ3SRCCTL）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6:1	IOSEL	RW	0x0	触发 TZ3 的 GPIO 引脚编号
0	POL	RW	0x1	TZ3 极性 0：GPIO 输入低时触发 TZ3 1：GPIO 输入高时触发 TZ3

表 8-146：TZ4 信号源控制寄存器（TZ4SRCCTL）位段定义

TZ4SRCCTL (TZ4 Source Control Register)　Offset：0x10　Default：0x00000001
Access：PWMCFG->TZ4SRCCTL

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	IOSEL						POL

表 8-147：TZ4 信号源控制寄存器（TZ4SRCCTL）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6:1	IOSEL	RW	0x0	触发 TZ4 的 GPIO 引脚编号
0	POL	RW	0x1	TZ4 极性 0：GPIO 输入低时触发 TZ4 1：GPIO 输入高时触发 TZ4

表 8-148：全局 PWM 软件强制同步寄存器（FRCSYNC）位段定义

FRCSYNC (Global PWM Force Synchronization Register)　Offset：0x14　Default：0x00000000
Access：PWMCFG->FRCSYNC

31	30	29	28	27	26	25	24
PWMCLK	RESERVED						
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				PWM3SYNC	PWM2SYNC	PWM1SYNC	PWM0SYNC

©2025 旋智电子科技（上海）有限公司

表 8-149：全局 PWM 软件强制同步寄存器（FRCSYNC）位段描述

位段	位段名	属性	复位值	描述
31	PWMCLK	W1S	0x0	软件强制同步所有 PWM 的时钟 0：写 0 无效，读总是 0 1：写 1 同步所有 PWM 的时钟，本位段自动清零
30:4	RESERVED_30_4	RO	0x0	保留
3	PWM3SYNC	W1S	0x0	软件强制 PWM3 同步 0：写 0 无效，读总是 0 1：写 1 产生 PWM3 同步脉冲，本位段自动清零
2	PWM2SYNC	W1S	0x0	软件强制 PWM2 同步 0：写 0 无效，读总是 0 1：写 1 产生 PWM2 同步脉冲，本位段自动清零
1	PWM1SYNC	W1S	0x0	软件强制 PWM1 同步 0：写 0 无效，读总是 0 1：写 1 产生 PWM1 同步脉冲，本位段自动清零
0	PWM0SYNC	W1S	0x0	软件强制 PWM0 同步 0：写 0 无效，读总是 0 1：写 1 产生 PWM0 同步脉冲，本位段自动清零

表 8-150：GPIO 触发 PWMSYNCI 信号源控制寄存器（GPIOSYNCICTL）位段定义

GPIOSYNCICTL (GPIO PWMSYNCI Control Register) Offset：0x18 Default：0x00000001 Access：PWMCFG->GPIOSYNCICTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	IOSEL						POL

表 8-151：GPIO 触发 PWMSYNCI 信号源控制寄存器（GPIOSYNCICTL）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6:1	IOSEL	RW	0x0	触发 PWMSYNCI 的 GPIO 输入引脚编号

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
0	POL	RW	0x1	PWMSYNCI 输入信号极性 0：GPIO 输入低时触发 PWMSYNCI 1：GPIO 输入高时触发 PWMSYNCI

表 8-152：GPIO 触发 PWMSYNCI 使能寄存器（GPIOSYNCIEN）位段定义

GPIOSYNCIEN (GPIO Force PWMSYNCI Enable Register) Offset：0x1C Default：0x00000000 Access：PWMCFGx->GPIOSYNCIEN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				PWM3SYNC	PWM2SYNC	PWM1SYNC	PWM0SYNC

表 8-153：GPIO 触发 PWMSYNCI 使能寄存器（GPIOSYNCIEN）位段描述

位段	位段名	属性	复位值	描述
31:4	RESERVED_31_4	RO	0x0	保留
3	PWM3SYNC	RW	0x0	使能 GPIO 触发 PWM3 同步 0：关闭 1：使能
2	PWM2SYNC	RW	0x0	使能 GPIO 触发 PWM2 同步 0：关闭 1：使能
1	PWM1SYNC	RW	0x0	使能 GPIO 触发 PWM1 同步 0：关闭 1：使能
0	PWM0SYNC	RW	0x0	使能 GPIO 触发 PWM0 同步 0：关闭 1：使能

©2025 旋智电子科技（上海）有限公司

表 8-154：TIMER0 触发 PWMSYNCI 使能寄存器（TMR0SYNCIEN）位段定义

TMR0SYNCIEN (Timer 0 Force PWMSYNCI Enable Register) Offset：0x20 Default：0x00000000 Access：PWMCFG->TMR0SYNCIEN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				PWM3SYNC	PWM2SYNC	PWM1SYNC	PWM0SYNC

表 8-155：TIMER0 触发 PWMSYNCI 使能寄存器（TMR0SYNCIEN）位段描述

位段	位段名	属性	复位值	描述
31:4	RESERVED_31_4	RO	0x0	保留
3	PWM3SYNC	RW	0x0	使能 TIMER0 触发 PWM3 同步 0：关闭 1：使能
2	PWM2SYNC	RW	0x0	使能 TIMER0 触发 PWM2 同步 0：关闭 1：使能
1	PWM1SYNC	RW	0x0	使能 TIMER0 触发 PWM1 同步 0：关闭 1：使能
0	PWM0SYNC	RW	0x0	使能 TIMER0 触发 PWM0 同步 0：关闭 1：使能

表 8-156：TIMER1 触发 PWMSYNCI 使能寄存器（TMR1SYNCIEN）位段定义

TMR1SYNCIEN (Timer 1 Force PWMSYNCI Enable Register) Offset：0x24 Default：0x00000000 Access：PWMCFG->TMR1SYNCIEN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				PWM3SYNC	PWM2SYNC	PWM1SYNC	PWM0SYNC

SPINTROL

表 8-157：TIMER1 触发 PWMSYNCI 使能寄存器（TMR1SYNCIEN）位段描述

位段	位段名	属性	复位值	描述
31:4	RESERVED_31_4	RO	0x0	保留
3	PWM3SYNC	RW	0x0	使能 TIMER1 触发 PWM3 同步 0：关闭 1：使能
2	PWM2SYNC	RW	0x0	使能 TIMER1 触发 PWM2 同步 0：关闭 1：使能
1	PWM1SYNC	RW	0x0	使能 TIMER1 触发 PWM1 同步 0：关闭 1：使能
0	PWM0SYNC	RW	0x0	使能 TIMER1 触发 PWM0 同步 0：关闭 1：使能

表 8-158：TIMER2 触发 PWMSYNCI 使能寄存器（TMR2SYNCIEN）位段定义

TMR2SYNCIEN (Timer 2 Force PWMSYNCI Enable Register)　Offset：0x28　Default：0x00000000
Access：PWMCFG->TMR2SYNCIEN

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				PWM3SYNC	PWM2SYNC	PWM1SYNC	PWM0SYNC

表 8-159：TIMER2 触发 PWMSYNCI 使能寄存器（TMR2SYNCIEN）位段描述

位段	位段名	属性	复位值	描述
31:4	RESERVED_31_4	RO	0x0	保留
3	PWM3SYNC	RW	0x0	使能 TIMER2 触发 PWM3 同步 0：关闭 1：使能
2	PWM2SYNC	RW	0x0	使能 TIMER2 触发 PWM2 同步 0：关闭 1：使能
1	PWM1SYNC	RW	0x0	使能 TIMER2 触发 PWM1 同步 0：关闭 1：使能

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
0	PWM0SYNC	RW	0x0	使能 TIMER2 触发 PWM0 同步 0：关闭 1：使能

表 8-160：PWMSYNCO 输出控制寄存器（SYNCOCTL）位段定义

SYNCOCTL (PWMSYNCO Control Register) Offset：0x2C Default：0x00000007 Access：PWMCFG->SYNCOCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	SYNCO3EN	SYNCO2EN	SYNCO1EN	SYNCO0EN	POL	DURATION	

表 8-161：PWMSYNCO 输出控制寄存器（SYNCOCTL）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6	SYNCO3EN	RW	0x0	使能 PWM3SYNCO 输出到引脚（逻辑或） 0：关闭 1：使能
5	SYNCO2EN	RW	0x0	使能 PWM2SYNCO 输出到引脚（逻辑或） 0：关闭 1：使能
4	SYNCO1EN	RW	0x0	使能 PWM1SYNCO 输出到引脚（逻辑或） 0：关闭 1：使能
3	SYNCO0EN	RW	0x0	使能 PWM0SYNCO 输出到引脚（逻辑或） 0：关闭 1：使能
2	POL	RW	0x1	PWMSYNCO 输出到 IO 管脚的极性 0：PWMSYNCO 有效时将管脚置低 1：PWMSYNCO 有效时将管脚置高
1:0	DURATION	RW	0x3	PWMSYNCO 输出到 IO 管脚的脉冲宽度 0：脉冲宽度为 4 个 PWM 时钟周期 1：脉冲宽度为 8 个 PWM 时钟周期 2：脉冲宽度为 16 个 PWM 时钟周期 3：脉冲宽度为 32 个 PWM 时钟周期

©2025 旋智电子科技（上海）有限公司

表 8-162：PWMSOCA 输出控制寄存器（SOCAOCTL）位段定义

SOCAOCTL (PWMSOCA Output Control Register) Offset：0x30 Default：0x00000007 Access：PWMCFG->SOCAOCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	SOCA3EN	SOCA2EN	SOCA1EN	SOCA0EN	POL	DURATION	

表 8-163：PWMSOCA 输出控制寄存器（SOCAOCTL）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6	SOCA3EN	RW	0x0	使能 PWM3SOCA 输出到引脚（逻辑或） 0：关闭 1：使能
5	SOCA2EN	RW	0x0	使能 PWM2SOCA 输出到引脚（逻辑或） 0：关闭 1：使能
4	SOCA1EN	RW	0x0	使能 PWM1SOCA 输出到引脚（逻辑或） 0：关闭 1：使能
3	SOCA0EN	RW	0x0	使能 PWM0SOCA 输出到引脚（逻辑或） 0：关闭 1：使能
2	POL	RW	0x1	PWMSOCA 输出到 IO 管脚的极性 0：PWMSOCA 有效时将管脚置低 1：PWMSOCA 有效时将管脚置高
1:0	DURATION	RW	0x3	PWMSOCA 输出到 IO 管脚的脉冲宽度 0：脉冲宽度为 4 个 PWM 时钟周期 1：脉冲宽度为 8 个 PWM 时钟周期 2：脉冲宽度为 16 个 PWM 时钟周期 3：脉冲宽度为 32 个 PWM 时钟周期

©2025 旋智电子科技（上海）有限公司

表 8-164：PWMSOCB 输出控制寄存器（SOCBOCTL）位段定义

SOCBOCTL (PWMSOCB Output Control Register) Offset：0x34 Default：0x00000007 Access：PWMCFG->SOCBOCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	SOCB3EN	SOCB2EN	SOCB1EN	SOCB0EN	POL	DURATION	

表 8-165：PWMSOCB 输出控制寄存器（SOCBOCTL）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6	SOCB3EN	RW	0x0	使能 PWM3SOCB 输出到引脚（逻辑或） 0：关闭 1：使能
5	SOCB2EN	RW	0x0	使能 PWM2SOCB 输出到引脚（逻辑或） 0：关闭 1：使能
4	SOCB1EN	RW	0x0	使能 PWM1SOCB 输出到引脚（逻辑或） 0：关闭 1：使能
3	SOCB0EN	RW	0x0	使能 PWM0SOCB 输出到引脚（逻辑或） 0：关闭 1：使能
2	POL	RW	0x1	PWMSOCB 输出到 IO 管脚的极性 0：PWMSOCB 有效时将管脚置低 1：PWMSOCB 有效时将管脚置高
1:0	DURATION	RW	0x3	PWMSOCB 输出到 IO 管脚的脉冲宽度 0：脉冲宽度为 4 个 PWM 时钟周期 1：脉冲宽度为 8 个 PWM 时钟周期 2：脉冲宽度为 16 个 PWM 时钟周期 3：脉冲宽度为 32 个 PWM 时钟周期

SPINTROL

表 8-166：PWMSOCC 输出控制寄存器（SOCCOCTL）位段定义

SOCCOCTL (PWMSOCC Output Control Register) Offset：0x38 Default：0x00000007 Access：PWMCFG->SOCCOCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	SOCC3EN	SOCC2EN	SOCC1EN	SOCC0EN	POL	DURATION	

表 8-167：PWMSOCC 输出控制寄存器（SOCCOCTL）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6	SOCC3EN	RW	0x0	使能 PWM3SOCC 输出到引脚（逻辑或） 0：关闭 1：使能
5	SOCC2EN	RW	0x0	使能 PWM2SOCC 输出到引脚（逻辑或） 0：关闭 1：使能
4	SOCC1EN	RW	0x0	使能 PWM1SOCC 输出到引脚（逻辑或） 0：关闭 1：使能
3	SOCC0EN	RW	0x0	使能 PWM0SOCC 输出到引脚（逻辑或） 0：关闭 1：使能
2	POL	RW	0x1	PWMSOCC 输出到 IO 管脚的极性 0：PWMSOCC 有效时将管脚置低 1：PWMSOCC 有效时将管脚置高
1:0	DURATION	RW	0x3	PWMSOCC 输出到 IO 管脚的脉冲宽度 0：脉冲宽度为 4 个 PWM 时钟周期 1：脉冲宽度为 8 个 PWM 时钟周期 2：脉冲宽度为 16 个 PWM 时钟周期 3：脉冲宽度为 32 个 PWM 时钟周期

©2025 旋智电子科技（上海）有限公司

表 8-168：PWMCFG 模块写使能寄存器（PWMCFG）位段定义

PWMCFGREGKEY (PWMCFG Register Write-Allow Key Register) Offset：0x3C Default：0x1ACCE551 Access：PWMCFG->PWMCFGREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

表 8-169：PWMCFG 模块写使能寄存器（PWMCFG）位段描述

位段	位段名	属性	复位值	描述
31:0	KEY	RW	0x1ACCE551	写入 0x1ACCE551 解锁受保护的 PWMCFG 寄存器

SPINTROL

9 增强型捕获模块（ECAP）

9.1 概述

增强型捕获模块（ECAP）用于对外部输入事件的时序关系进行计时，应用场景包括：

- 机电设备的速度测量（例如使用霍尔传感器的电动车）；
- 测量脉冲信号的周期和占空；
- 对于占空比编码系统的解码；

本指南中描述的 ECAP 模块具有以下特点：

- 32 位时间基准，时间分辨率为 10 纳秒（时钟频率高达 100MHz）；
- 4 个 32 位事件时间戳捕获寄存器；
- 4 个捕获事件均可独立选择边沿极性（上升/下降沿），均可触发中断；
- 与外部事件同步的 4 级序列器；
  - 单轮捕获模式：在依次完成最多 4 次捕获后停止；
  - 持续捕获模式：在最多 4 个捕获事件之间持续轮流执行；
- 支持绝对时间戳捕获和相对时间戳捕获；
- 所有上述资源可用于单个输入引脚或双输入引脚；
- 若不需要捕获功能，可被配置为辅助 PWM（APWM）模式，产生简单的 PWM 波形；

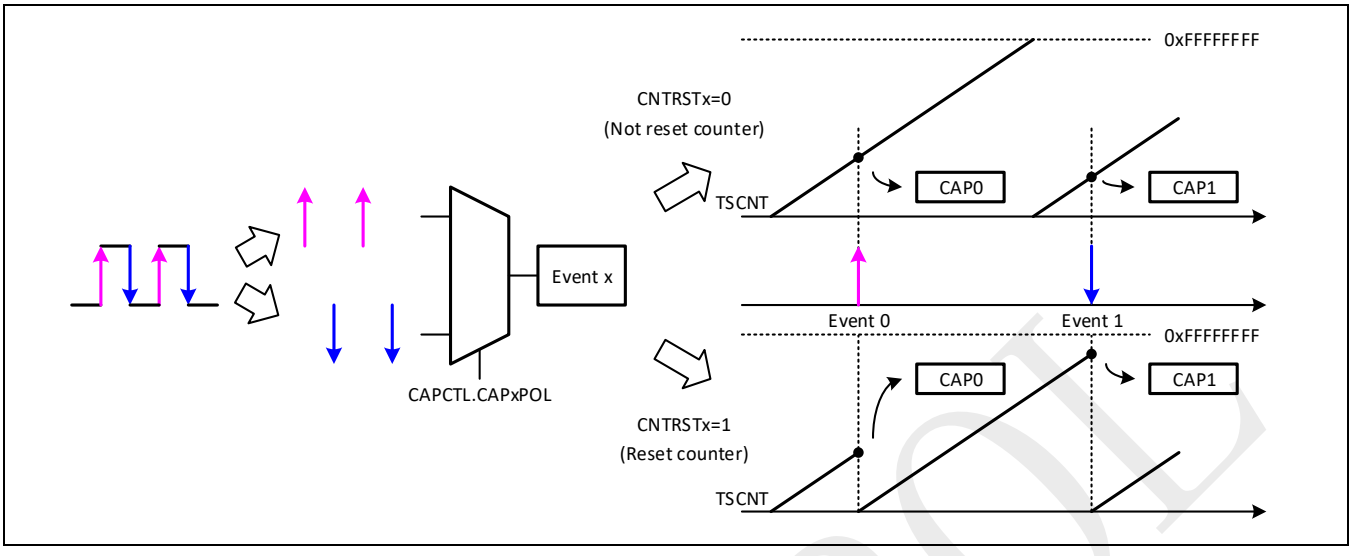
9.2 功能描述

ECAP 的两种基本工作模式如图 9-1 所示。根据 CAPCTL.CAPxPOL 位段的配置，输入信号的上升沿或者下降沿被逐一编号成最多 4 个具有时序先后的捕获事件（CEVT0、CEVT1、CEVT2、CEVT3）。每个捕获事件发生时，时间戳计数器根据 CAPCTL.CNTRSTx 位段的配置决定是否复位。通过两者的配置，用户可以实现诸如信号周期和占空计算等复杂的捕获和测量功能

SPINTROL

©2025 旋智电子科技（上海）有限公司

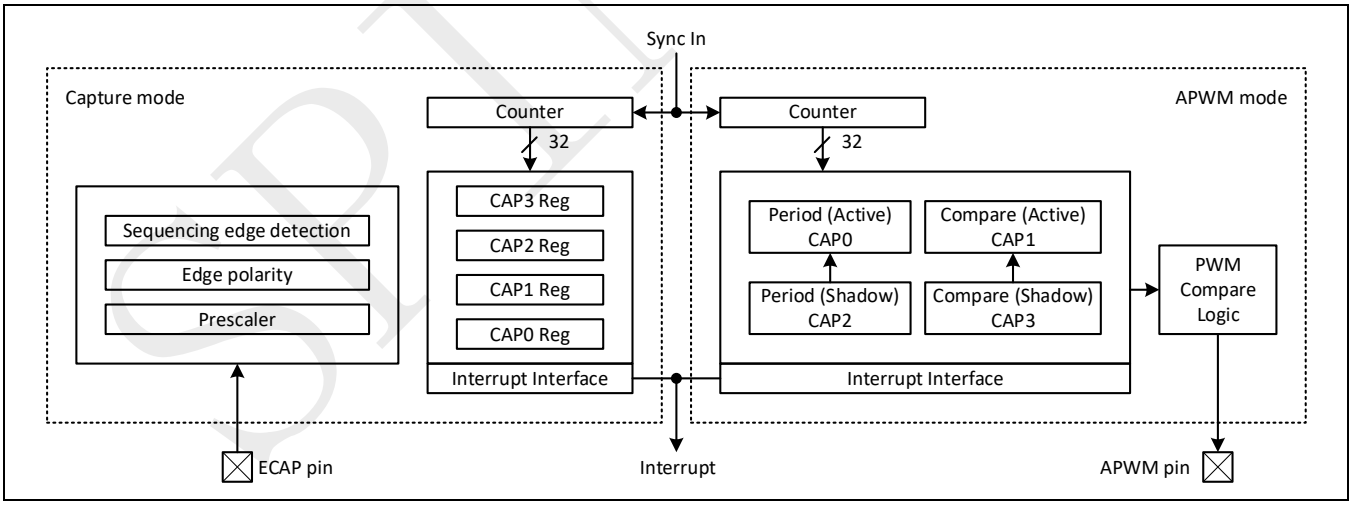
图 9-1：ECAP 基本工作方式



9.3 捕获模式和辅助 PWM 模式

如图 9-2 所示，当 ECAP 模块不用作输入捕获时，用户可以将其用作有 32 位计数能力的单通道 PWM 模块（APWM 模式）。此时计数器处于向上计数模式，并提供非对称 PWM 波形。CAP0 和 CAP1 分别用作周期 PRD 和计数比较值 CMP 的有效寄存器，CAP2 和 CAP3 分别用作周期 PRD 和计数比较值 CMP 的影子寄存器。

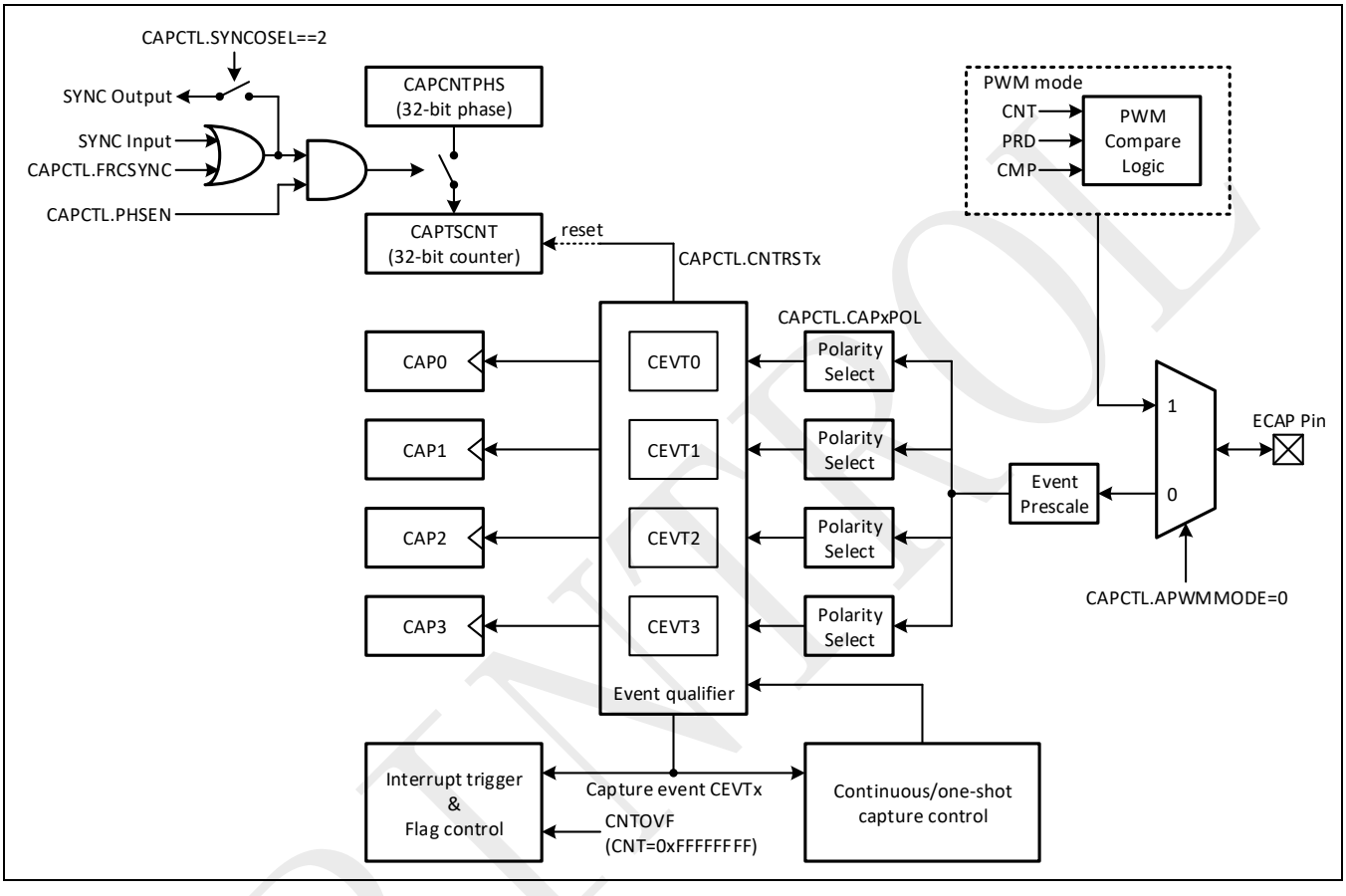
图 9-2：捕获模式和 APWM 模式



9.4 捕获模式详述

捕获模式的功能框图如图 9-3 所示。其中 ECAP 输入来自于由 CAPSRCCTL 寄存器指定的 GPIO 引脚，该引脚的通道复用控制需要使能 GPIO 通道的输入。

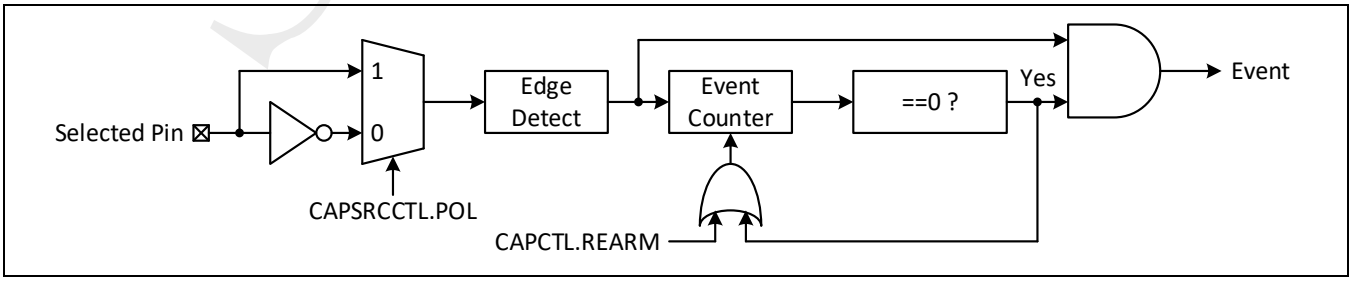
图 9-3：捕获模式功能框图



9.4.1 事件分频器

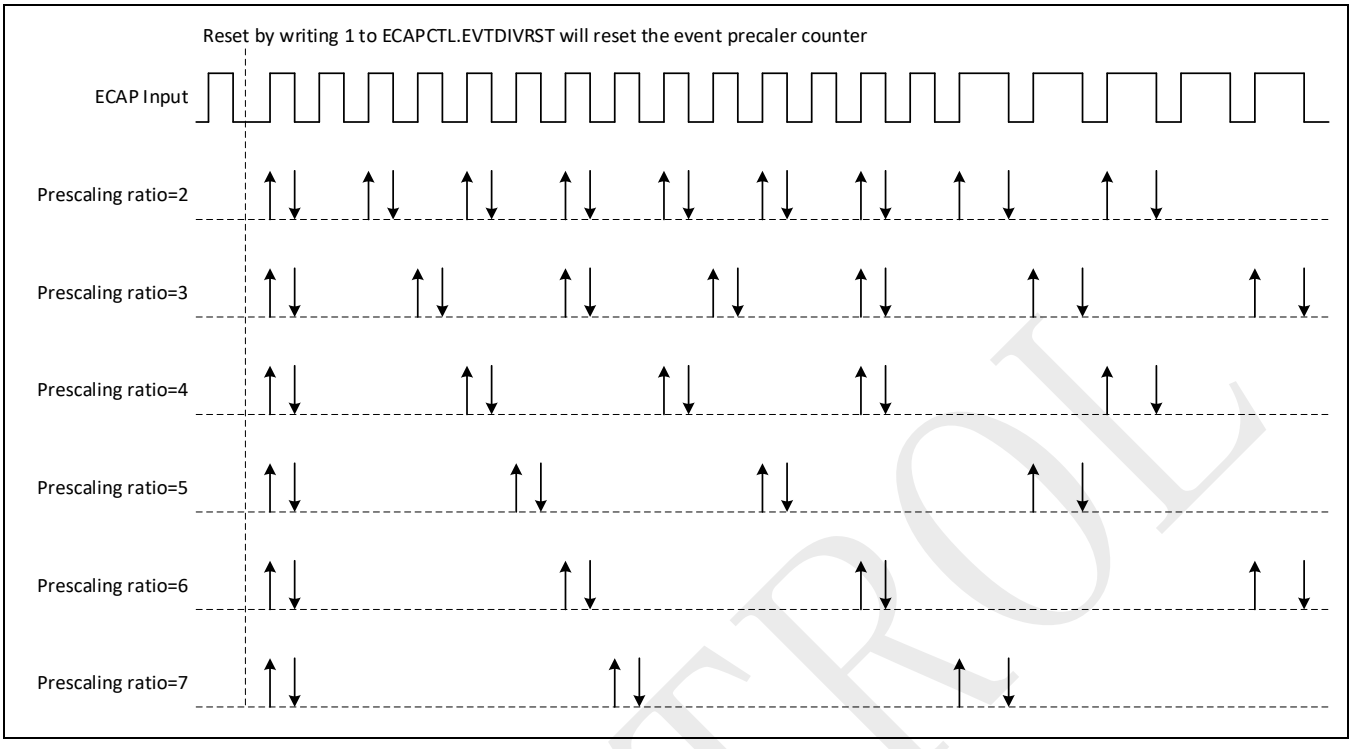
输入的待捕获事件（持续脉冲序列）可以进行预分频，比例为 N = CAPCTL.EVTDIV + 1，这在高频输入作为输入事件时非常有用。图 9-4 和图 9-5 分别给出了预分频的控制逻辑和波形示例。

图 9-4：事件分频控制



©2025 旋智电子科技（上海）有限公司

图 9-5：事件预分频波形



9.4.2 边沿选择和排序编号

每个事件（最多 4 个）对应的输入信号边沿（上升沿或下降沿）都有独立的配置：

- 每个边沿事件都通过模 4 计数器按顺序编号为 CEVT0、CEVT1、CEVT2、CEVT3；
- 每个事件触发的捕获，其结果由模 4 计数器控制存入对应的 CAPx 寄存器。

9.4.3 双引脚捕获

为了便于从两个引脚捕获边沿的时序差异，ECAP 模块支持可选的双引脚模式。在这种情况下，捕获事件 CEVT0 和 CEVT2 来自一个引脚，而捕获事件 CEVT1 和 CEVT3 来自另一个引脚。

9.4.4 单轮捕获和持续捕获控制

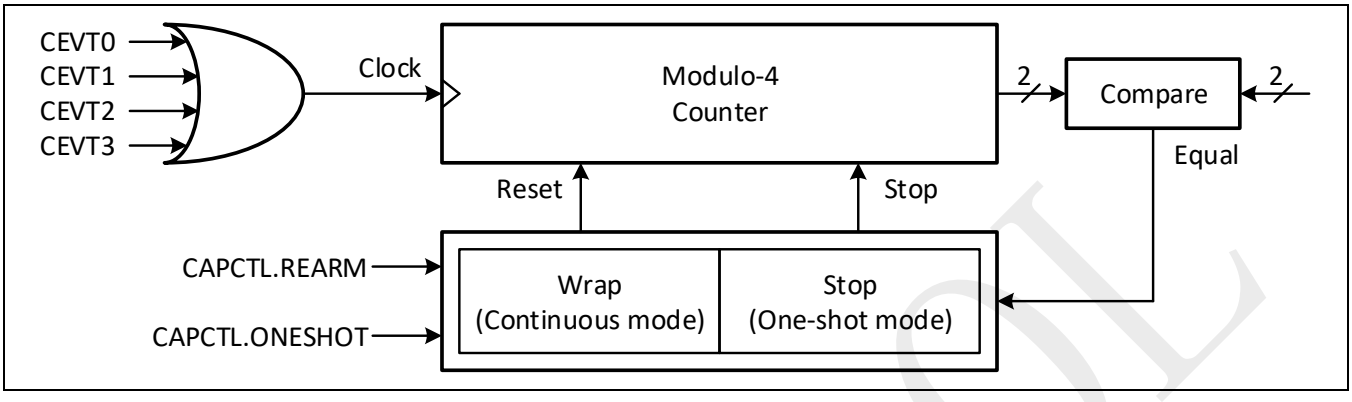
模 4 计数器在 CEVT0 到 CEVT3 事件发生时逐一递增计数，它记录了当前事件的编号，并与 CAPCTL.STOPWARP 位段配合按照如下方式控制捕获的时序：

- 在单轮捕获模式下，模 4 计数器在与阈值匹配时停止计数，并停止更新 CAP0 到 CAP3 寄存器。例如，如果 CAPCTL.STOPWARP 设置为 3，则模 4 计数器运行为 0→1→2→3，然后停止。抓取的值依次写入 CAP0 到 CAP3，以后保持不变。
- 在持续捕获模式下，模 4 计数器保持计数并在与阈值匹配时环回到 0。例如，如果 CAPCTL.STOPWARP 设置为 2，则模 4 计数器按照 0→1→2→0 持续运行，捕获的值不断循环写入 CAP0 到 CAP2。

©2025 旋智电子科技（上海）有限公司

通过往 CAPCTL.REARM 位段写 1 可以重置模 4 计数器为 0，并再次启动单轮模式或连续模式的事件。

图 9-6：单轮捕获和持续捕获逻辑

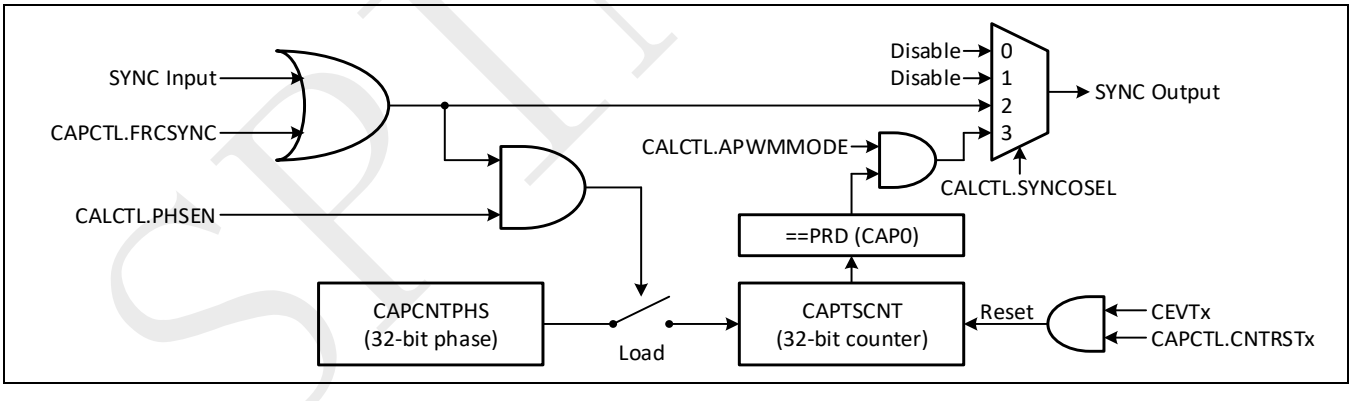


9.4.5 32 位计数器和相位控制

32 位计数器以 ECAP 模块时钟作为计数时钟，提供了事件捕获的时间戳。此外，相位寄存器 CAPCNTPHS 指定了硬件或者软件同步发生时相对于其他计数器的相位偏移，这在需要指定跨模块相位偏移的 PWM 模式下非常有用。

在任意一个捕获事件发生时，均可以配置计数器复位，便于实现对时间差的捕获和计算。例如将上升沿时捕获并复位计数器，在下一个下降沿时再次捕获，则可以得到输入信号的占空。

图 9-7：计数器及同步逻辑



9.4.6 CAP0-CAP3 寄存器

这 4 个 32 位寄存器在对应的捕获事件发生时存入当时的时间戳计数器的计数值。单轮捕获模式下，完成 CAPCTL.STOPWARP 指定次数的捕获后，CAP0~CAP3 自动停止捕获和更新。

在 APWM 模式下，CAP0 和 CAP1 分别用作周期 PRD 和计数比较值 CMP 的有效寄存器，CAP2 和 CAP3 分别用作周期 PRD 和计数比较值 CMP 的影子寄存器。

©2025 旋智电子科技（上海）有限公司

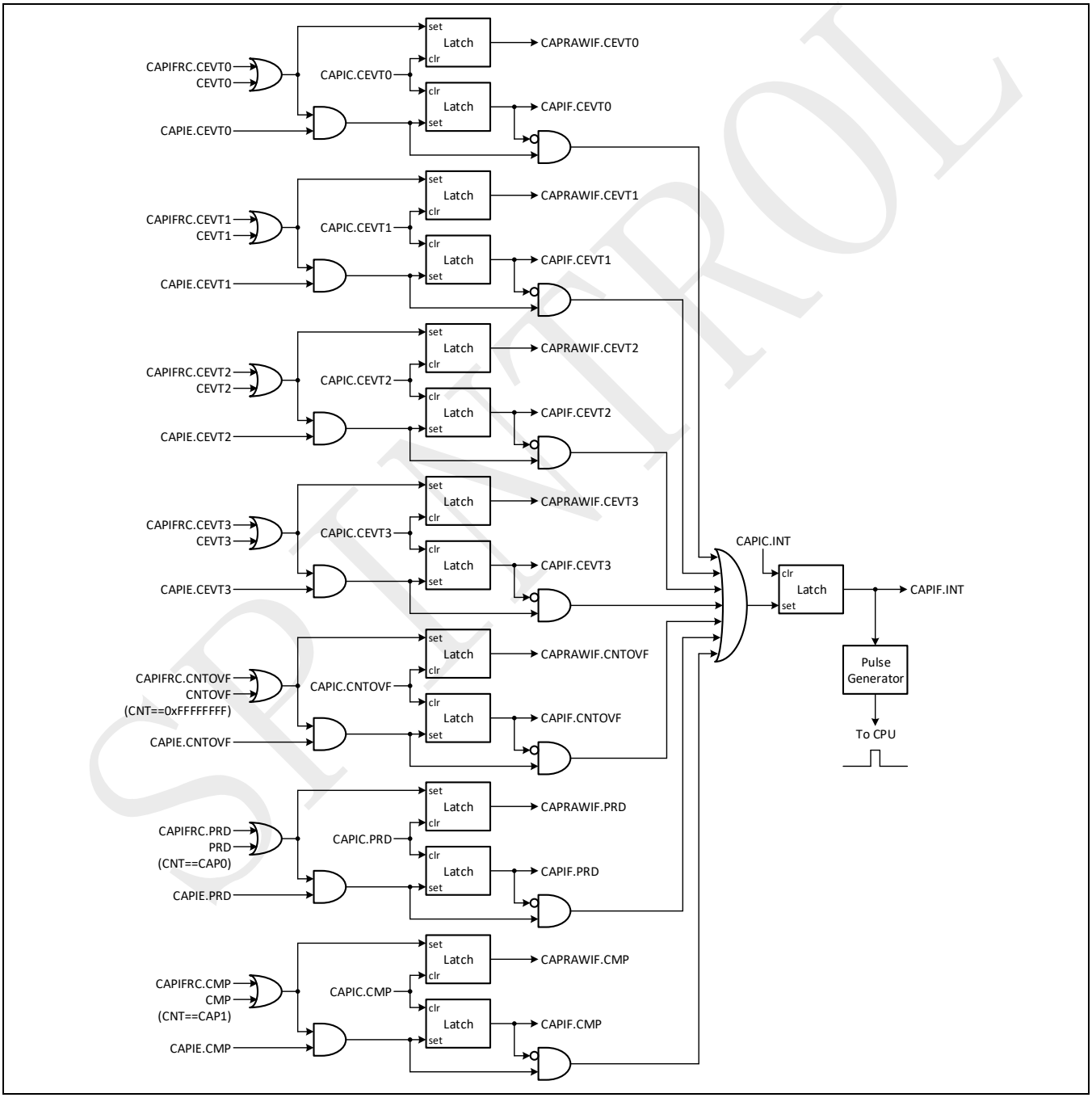
9.4.7 中断控制

捕获事件（CEVT0-CEVT3）、APWM 事件（CNT=PRD 和 CNT=CMP）均可以触发中断。此外，计数溢出（0xFFFFFFFF→0x00000000）也可以作为独立的事件（CNTOVF）用来触发 CPU 中断。

在捕获模式下，有效的中断事件包括 CEVT0、CEVT1、CEVT2、CEVT3 和 CNTOVF。

在 APWM 模式下，有效的中断事件包括 CNT=PRD、CNT=CMP 和 CNTOVF。

图 9-8：ECAP 模块中断



SPINTROL

每当中断事件发生时，会记录到 CAPRAWIF 寄存器对应位。但仅当通过 CAPIE 对应位使能后，该事件才会触发中断，并记录到 CAPIF 寄存器对应位。中断服务代码里必须通过中断清除寄存器 CAPIC 同时清除事件标志和全局中断标志才能保证后续事件能触发新的中断。若保持 CAPIF 中的事件中断标志为 1，则阻止后续同样的事件触发中断，但不影响其他事件触发中断；若保持 CAPIF 中的全局中断标志为 1，则阻止后续所有事件触发中断。往 CAPIC 的位段写 1 也会同时清除 CAPRAWIF 的对应位。

用户可以通过 CAPIFRC 寄存器软件强制一个中断事件，用于代码调试。

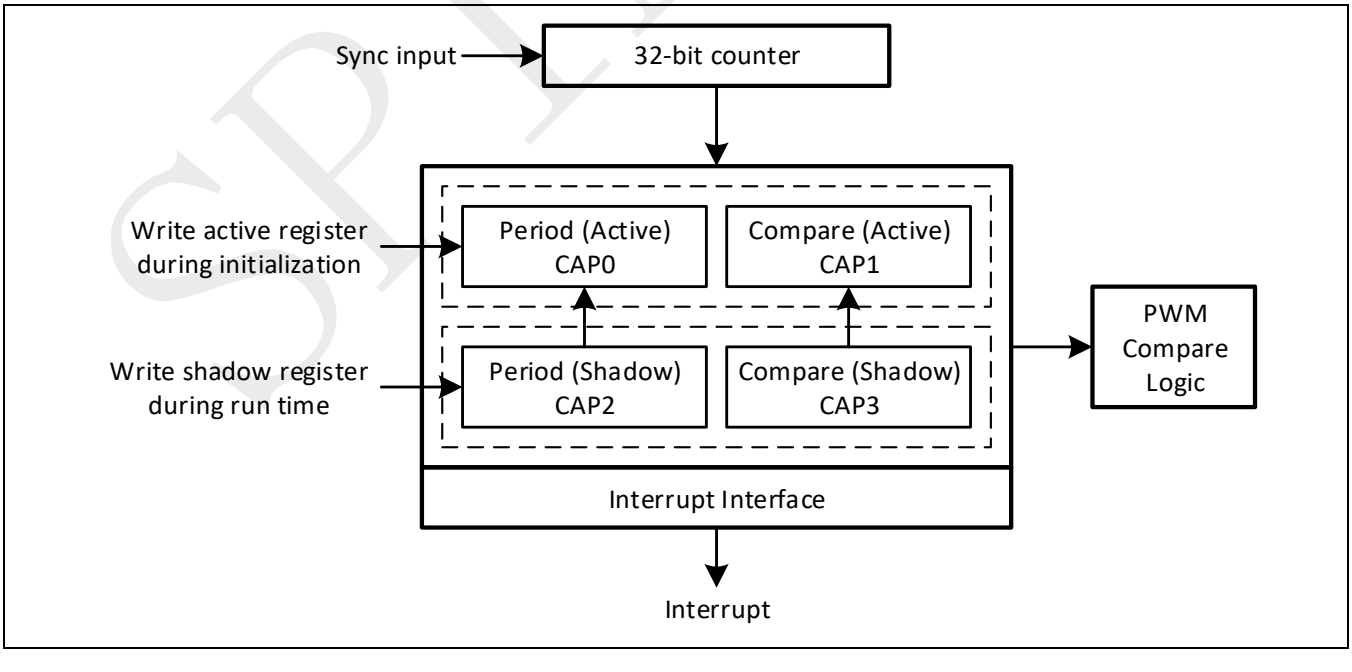
9.5 APWM 模式详述

在 APWM 模式下，32 位计数器工作在向上计数模式，此时捕获寄存器的行为如下：

- CAP0 用作周期有效寄存器（APRD）；
- CAP1 用作比较值有效寄存器（ACMP）；
- CAP2 用作周期影子寄存器（SPRD）；
- CAP3 用作比较值影子寄存器（SCMP）；
- 写 CAP0 和 CAP1 会立即同步更新 CAP2 和 CAP3；
- 写 CAP2 和 CAP3 会启动影子模式
  - ➢ 当 TSCNT = CAP0 时，将 CAP2 加载到 CAP0，以及将 CAP3 加载到 CAP1。

用户可以在初始化时通过 CAP0 和 CAP1 设置有效的周期和比较值，对于后续运行时的更新，用户只需要使用影子寄存器 CAP2 和 CAP3。

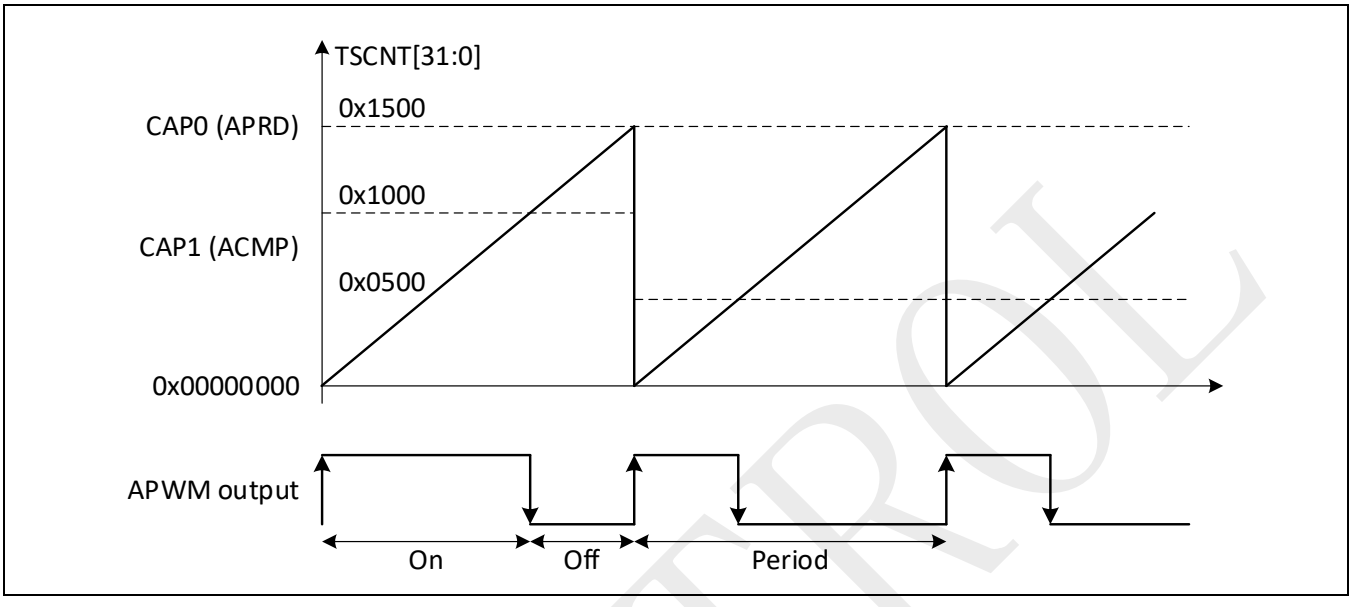
图 9-9：APWM 模式下 PRD 和 CMD 的有效值和影子值



©2025 旋智电子科技（上海）有限公司

如图 9-10 所示，CAPTSCNT 不断地从 0 开始递增计数到 CAP0 又回到 0 继续计数。当计数到 0 时 APWM 输出变高，当计数到 CAP1 时输出变低。

图 9-10：APWM 模式波形（极性为高电平有效）



9.6 ECAP 的应用

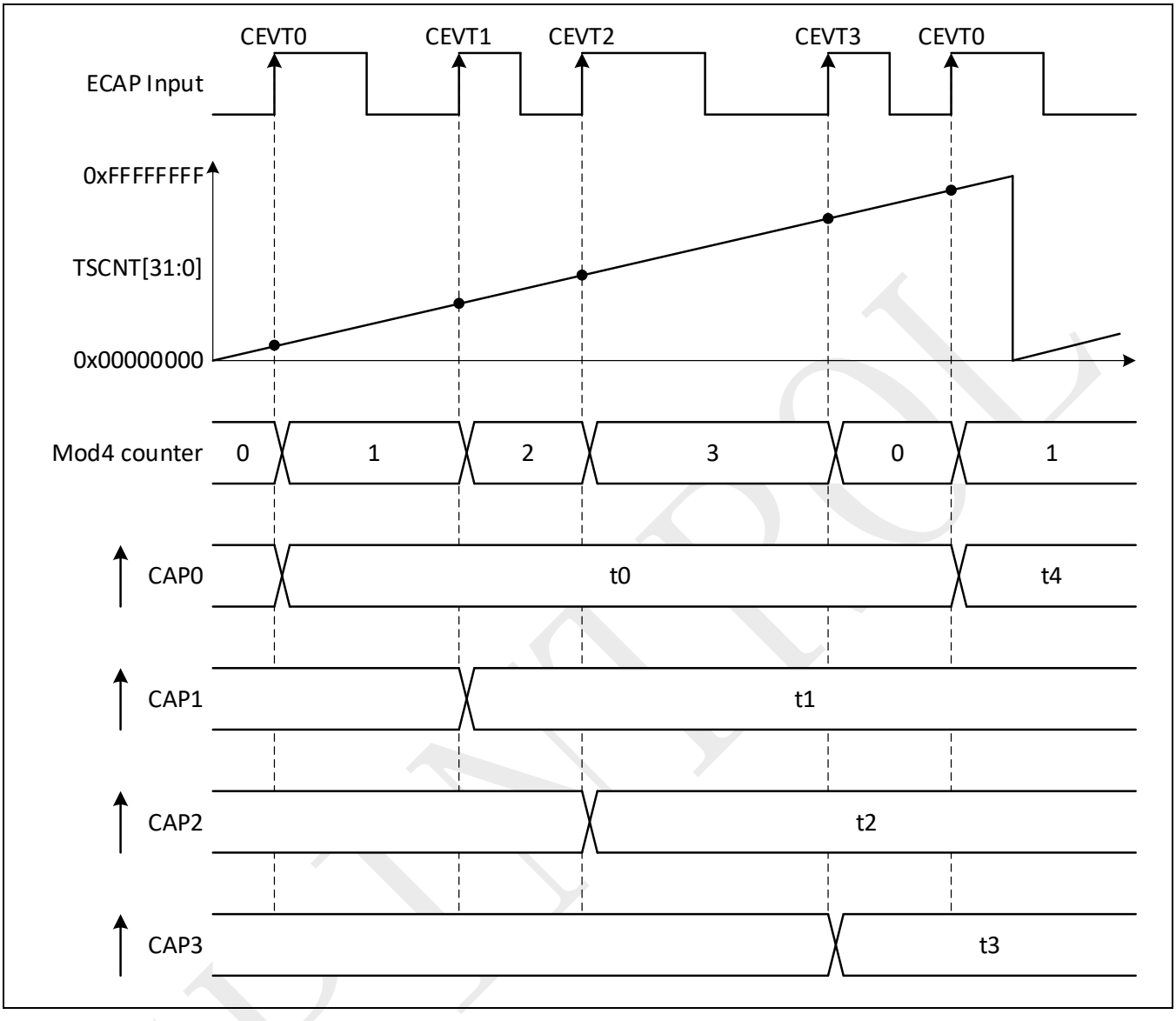
9.6.1 上升沿触发的绝对时间戳捕获

如图 9-11 所示，持续捕获模式（模 4 事件计数自动归零持续计数）下，配置 CAPTSCNT 持续递增计数而不复位，输入上升沿视作有效事件，由此可以获取周期（及频率）信息。

每个有效事件下，CAPTSCNT 当前值被作为时间戳捕获到对应的 CAPx 寄存器，然后模 4 事件计数加 1。当计数到 0xFFFFFFFF 时自动回到 0x00000000 继续计数（未在图中示例给出）。同时，计数溢出标志 CNTOVF 被置位，如若使能，还将触发 CPU 中断。图中标记了各个有效事件的时间点，从中可以看出每到 CEVT3 时 4 个捕获已经依次完成，因此可以选择 CEVT3 作为中断触发源，便于读取所有 CAPx 寄存器的值。输入信号的各个周期可以依次计算为 CAP1 – CAP0、CAP2 – CAP1、CAP3 – CAP2。

©2025 旋智电子科技（上海）有限公司

图 9-11：上升沿检测和绝对时间戳捕获

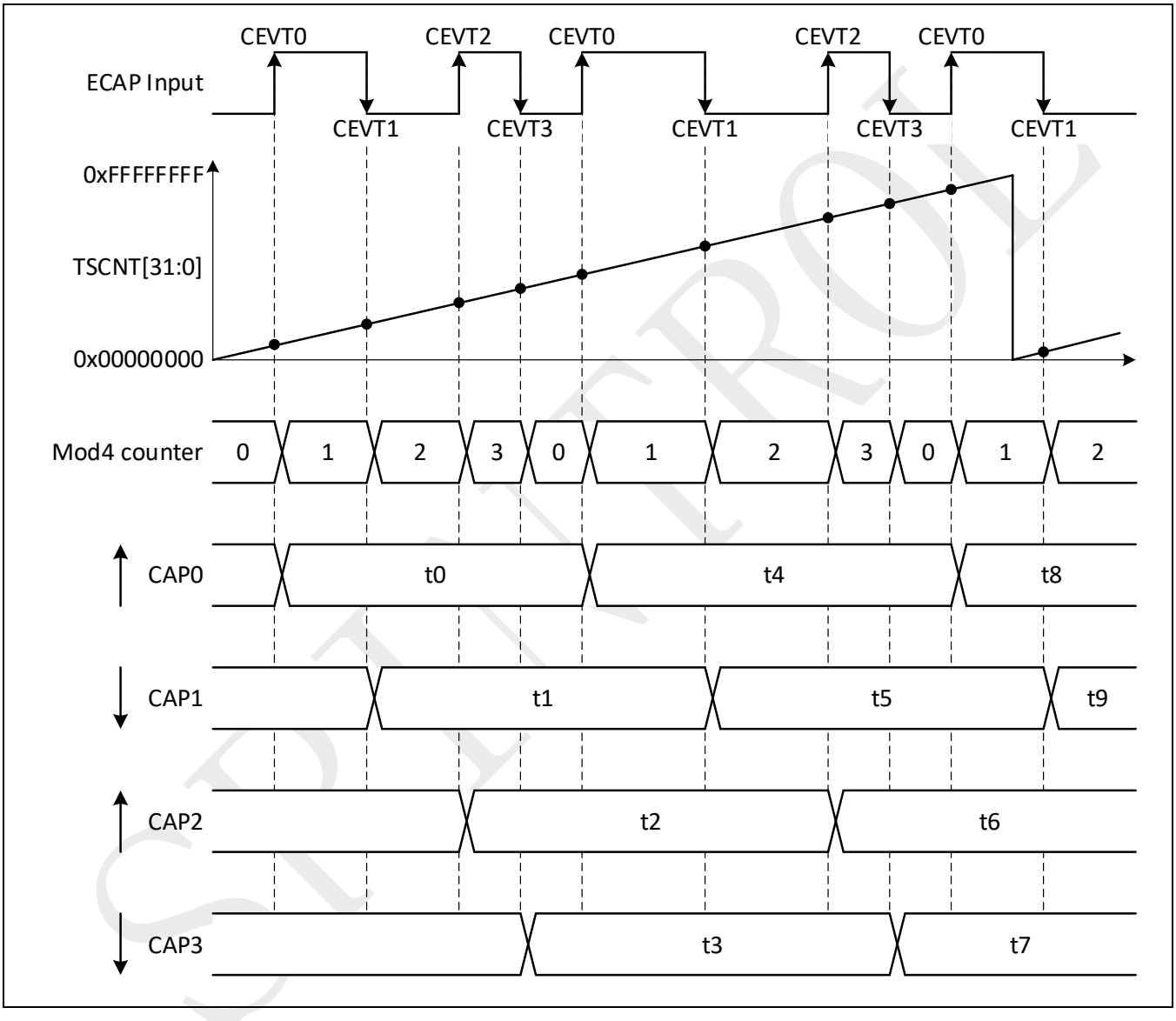


©2025 旋智电子科技（上海）有限公司

9.6.2 由绝对时间戳捕获周期和占空

图 9-12 所示情形与上例几乎相同，只是捕获事件改为上升沿和下降沿交替，由此给出了周期和占空的信息。例如：第一个周期为 T1 = t2 – t0，其中高电平占空为 H1 = t1 – t0，低电平占空为 L1 = t2 – t1。

图 9-12：通过绝对事件戳捕获计算周期和占空

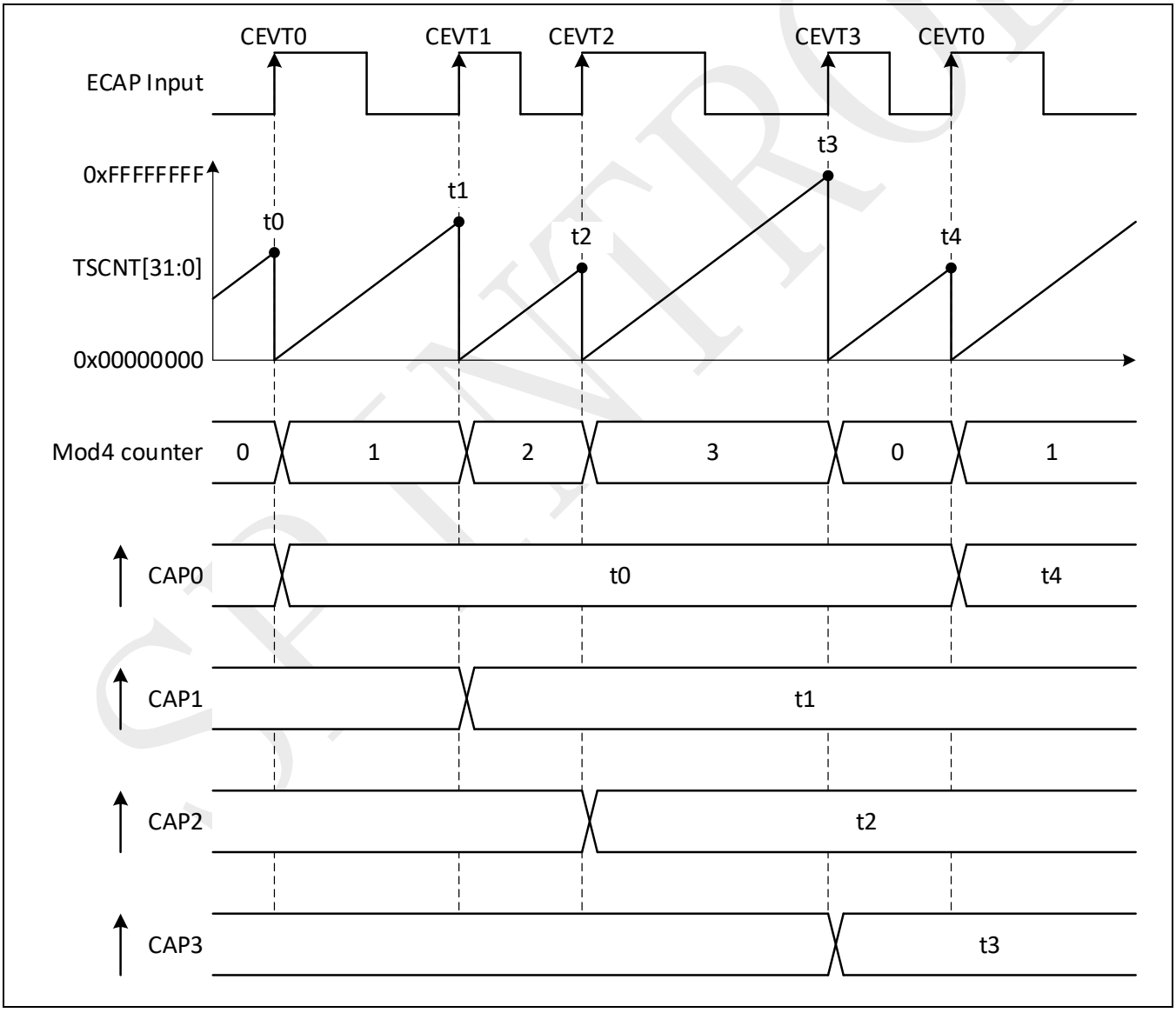


9.6.3 通过计数器复位捕获相对时间差

图 9-13 示例了 ECAP 如何从输入脉冲序列里获取时间差的信息。此处同样采用了持续捕获的模式并设置事件为上升沿有效。所不同的是，每次事件发生时，在捕获计数器当前值后都会复位计数器。

如果 TSCNT 在下一个事件之前达到 0xFFFFFFFF（即最大值），它将循环到 0x00000000 并继续，设置 CNTOVF（计数器溢出）标志，并发生中断（如果启用）。可以看出，这个方式的好处是 CAPx 寄存器的内容已经是所关心的周期信息而不需要 CPU 参与计算，即周期 T1 = t1，T2 = t2 等。从图中还可以看出，CEVT0 适于作为中断触发源，因为此时 T1、T2、T3、T4 的信息都已经成功获得。

图 9-13：通过计数器复位捕获周期

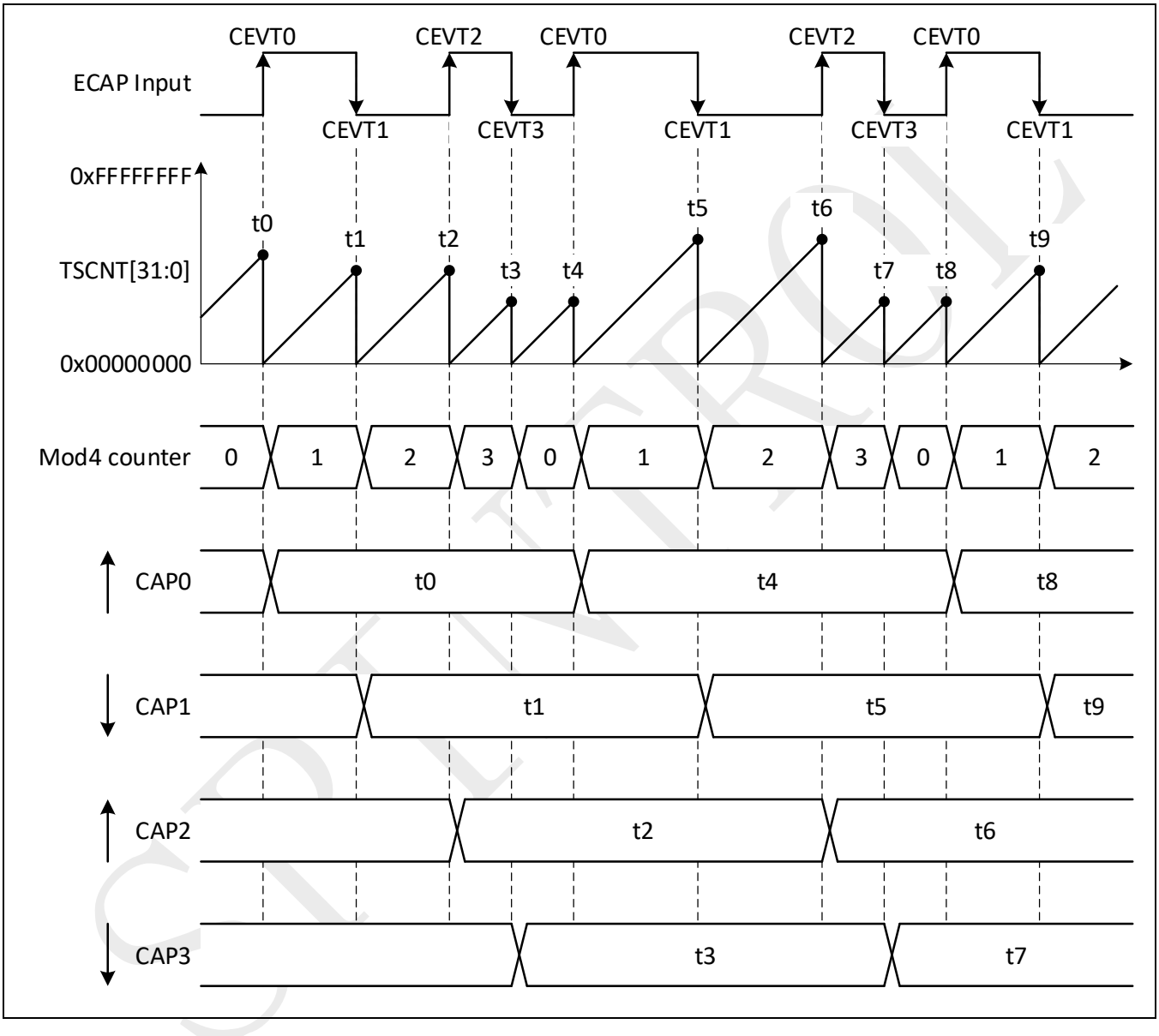


©2025 旋智电子科技（上海）有限公司

9.6.4 相对事件模式捕获周期和占空

图 9-14 所示工作模式与上例几乎相同，只是捕获事件改为上升沿和下降沿交替，由此给出周期和占空信息。例如：第一个周期 T1 = t1 + t2，其中高电平占空 t1，低电平占空 t2。

图 9-14：通过计数器复位捕获周期与占空



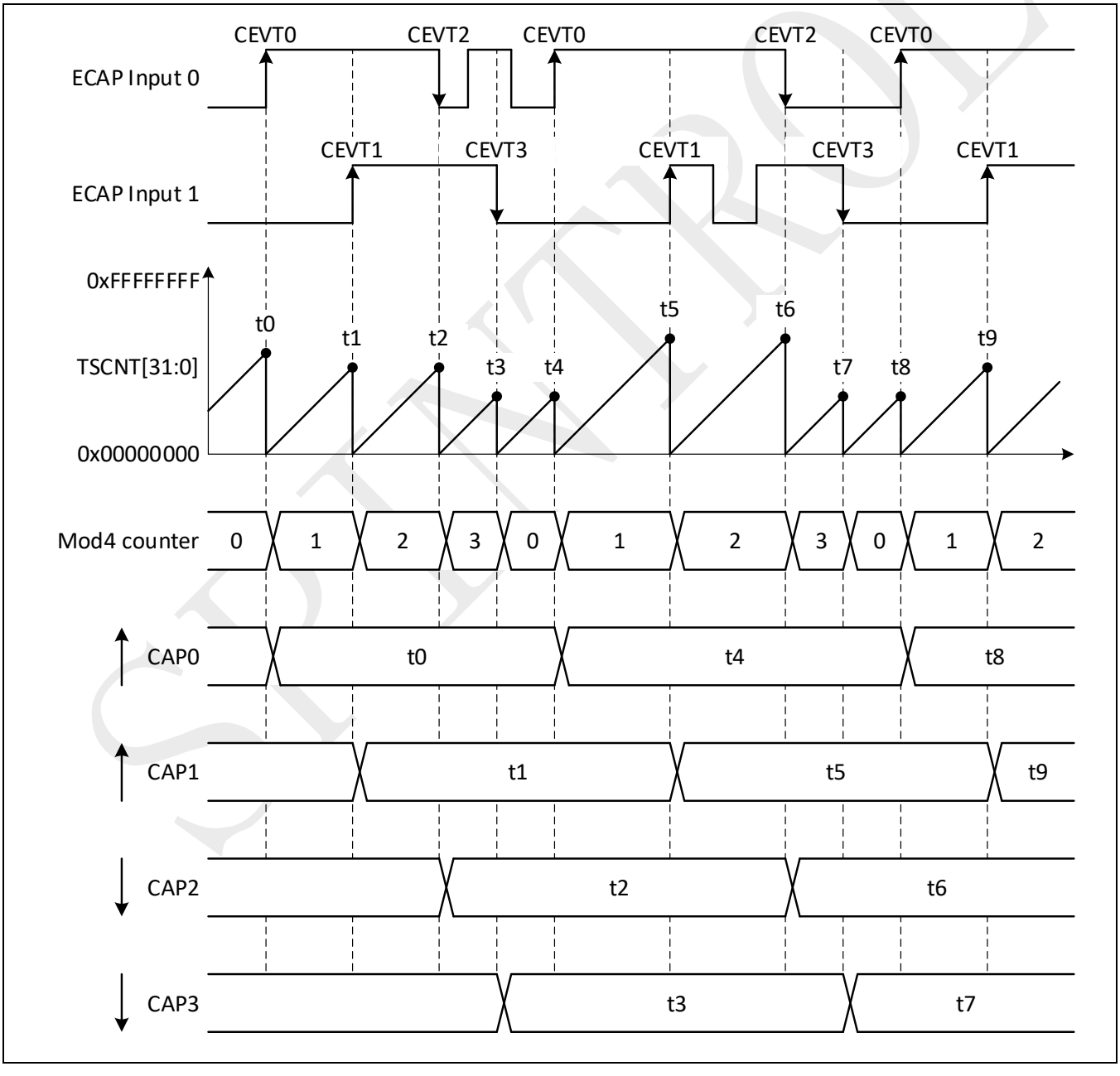
©2025 旋智电子科技（上海）有限公司

9.6.5 基于计数器复位的双引脚捕获

图 9-15 示例了双引脚捕获的波形。其中 CAP0 和 CAP2 分别用于捕获引脚 0 的上升和下降沿，CAP1 和 CAP2 分别用于捕获引脚 1 的上升和下降沿，并通过计数器复位捕获时间差。由此：

- CAP1 捕获了引脚 0 上升沿到**相邻**引脚 1 上升沿的时间差；
- CAP2 捕获了引脚 1 上升沿到**相邻**引脚 0 下降沿的时间差；
- CAP3 捕获了引脚 0 下降沿到**相邻**引脚 1 下降沿的时间差；
- CAP0 捕获了引脚 1 下降沿到**相邻**引脚 0 上升沿的时间差。

图 9-15：基于计数器复位的双引脚捕获



©2025 旋智电子科技（上海）有限公司

9.7 寄存器

9.7.1 ECAP 寄存器列表

表 9-1：ECAP 模块基地址

外设模块	基地址
ECAP	0x4000 3400

表 9-2：ECAP 寄存器列表

寄存器	偏移地址	描述	复位值
CAPSRCCTL*	0x00	捕获信号输入控制寄存器	0x00010001
CAPSYNCICTL*	0x04	同步信号输入控制寄存器	0x00000001
CAPTSCNT*	0x08	时间戳计数寄存器	0x00000000
CAPCNTPHS*	0x0C	时间戳计数相位寄存器	0x00000000
CAP0*	0x10	捕获值 0 寄存器	0x00000000
CAP1*	0x14	捕获值 1 寄存器	0x00000000
CAP2*	0x18	捕获值 2 寄存器	0x00000000
CAP3*	0x1C	捕获值 3 寄存器	0x00000000
CAPCTL*	0x20	捕获控制寄存器	0x000C0000
CAPRAWIF	0x24	捕获中断原始事件标志寄存器	0x00000000
CAPIF	0x28	捕获中断标志寄存器	0x00000000
CAPIE*	0x2C	捕获中断使能寄存器	0x00000000
CAPIC	0x30	捕获中断清除寄存器	0x00000000
CAPIFRC*	0x34	捕获中断强制寄存器	0x00000000
CAPREGKEY	0x38	ECAP 模块写使能寄存器	0x1ACCE551

注意： 由*标记的寄存器仅当 PWRREGKEY=0x1ACCE551 时才可以改写。

SPINTROL

9.7.2 ECAP 寄存器

表 9-3：捕获信号输入控制寄存器（CAPSRCCTL）位段定义

CAPSRCCTL (Capture Source Input Control Register) Access：ECAP->CAPSRCCTL Offset：0x00 Default：0x00010001							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
IOSEL1							POL1
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
IOSEL							POL

表 9-4：捕获信号输入控制寄存器（CAPSRCCTL）位段描述

位段	位段名	属性	复位值	描述
31:24	RESERVED_31_24	RO	0x0	保留
23:17	IOSEL1	RW	0x0	双引脚捕获时第二引脚的 GPIO 编号
16	POL1	RW	0x1	双引脚捕获时第二引脚的极性 0：低电平有效 1：高电平有效
15:8	RESERVED_15_8	RO	0x0	保留
7:1	IOSEL	RW	0x0	输入捕获的 GPIO 引脚编号
0	POL	RW	0x1	输入捕获的极性 0：低电平有效 1：高电平有效

表 9-5：同步信号输入控制寄存器（CAPSYNCICTL）位段定义

CAPSYNCICTL (Capture Synchronization Source Input Control Register) Access：ECAP->CAPSYNCICTL Offset：0x04 Default：0x00000001							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
IOSEL							POL

©2025 旋智电子科技（上海）有限公司

表 9-6：同步信号输入控制寄存器（CAPSYNCICTL）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7:1	IOSEL	RW	0x0	用作同步输入的 GPIO 引脚编号
0	POL	RW	0x1	同步信号的极性 0：低电平有效 1：高电平有效

表 9-7：时间戳计数寄存器（CAPTSCNT）位段定义

CAPTSCNT (Time-Stamp Counter Register) Access：ECAP->CAPTSCNT Offset：0x08 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 9-8：时间戳计数寄存器（CAPTSCNT）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	32 位时间戳计数值

表 9-9：时间戳计数相位寄存器（CAPCNTPHS）位段定义

CAPCNTPHS (Counter Phase Offset Value Register) Offset：0xC Default：0x00000000 Access：ECAP->CAPCNTPHS							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

©2025 旋智电子科技（上海）有限公司

表 9-10：时间戳计数相位寄存器（CAPCNTPHS）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	时间戳计数器在同步时的相位值

表 9-11：捕获值 0 寄存器（CAP0）位段定义

CAP0 (Capture Register 0) Offset：0x10 Default：0x00000000 Access：ECAP->CAP0							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 9-12：捕获值 0 寄存器（CAP0）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	捕获模式下 捕获事件 0 发生时根据 CAPTSCNT 更新本寄存器 APWM 模式下 写本寄存器设置周期有效值，并同时更新 CAP2 为相同值，由此模拟即时更新配置方式； 当 CAPTSCNT = CAP0 时，根据 CAP2 更新本寄存器值，由此模拟影子配置方式。

表 9-13：捕获值 1 寄存器（CAP1）位段定义

CAP1 (Capture Register 1) Offset：0x14 Default：0x00000000 Access：ECAP->CAP1							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

©2025 旋智电子科技（上海）有限公司

表 9-14：捕获值 1 寄存器（CAP1）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	捕获模式下 捕获事件 0 发生时根据 CAPTSCNT 更新本寄存器 APWM 模式下 写本寄存器设置比较值有效值，并同时更新 CAP3 为相同值，由此模拟即时更新配置方式； 当 CAPTSCNT = CAP0 时，根据 CAP3 更新本寄存器值，由此模拟影子配置方式。

表 9-15：捕获值 2 寄存器（CAP2）位段定义

CAP2 (Capture Register 2)　Offset：0x18　Default：0x00000000
Access：ECAP->CAP2

31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 9-16：捕获值 2 寄存器（CAP2）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	捕获模式下 捕获事件 2 发生时根据 CAPTSCNT 更新本寄存器 APWM 模式下 写本寄存器设置周期影子值，当 CAPTSCNT = CAP0 时，根据本寄存器更新 CAP0，由此模拟影子配置方式。

SPINTROL

表 9-17：捕获值 3 寄存器（CAP3）位段定义

CAP3 (Capture Register 3) Offset：0x1C Default：0x00000000 Access：ECAP->CAP3							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 9-18：捕获值 3 寄存器（CAP3）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	捕获模式下 捕获事件 3 发生时根据 CAPTSCNT 更新本寄存器 APWM 模式下 写本寄存器设置比较值影子值，当 CAPTSCNT = CAP0 时，根据本寄存器更新 CAP1，由此模拟影子配置方式。

表 9-19：捕获控制寄存器（CAPCTL）位段定义

CAPCTL (Capture Control Register) Offset：0x20 Default：0x000C0000 Access：ECAP->CAPCTL							
31 DUALPIN	30 EVTDIVRST	29	28 DBGRUN	27 APWMPOL	26 APWMMODE	25 FRCSYNC	24 SYNCOSEL
23 SYNCOSEL	22 PHSEN	21 TSCNTRUN	20 REARM	19	18 STOPWRAP	17 ONESHOT	16 EVTDIV
15	14	13	12 EVTDIV	11	10	9	8 CAPLDEN
7 CNTRST3	6 CAP3POL	5 CNTRST2	4 CAP2POL	3 CNTRST1	2 CAP1POL	1 CNTRST0	0 CAP0POL

表 9-20：捕获控制寄存器（CAPCTL）位段描述

位段	位段名	属性	复位值	描述
31	DUALPIN	RW	0x0	双引脚捕获模式 0：CAP0 ~ CAP3 用于同一引脚捕获 1：CAP0/CAP2 和 CAP1/CAP3 各用于一个引脚

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
30	EVTDIVRST	W1S	0x0	事件分频复位 0：写 0 无效，读总是 0 1：写 1 复位事件分频器，本位段自动清零
29:28	DBGRUN	RW	0x0	CPU halted 和 lockup 时计数器的行为 0：立即停止计数 1：计数到 0 时停止 2：正常计数
27	APWMPOL	RW	0x0	APWM 输出极性 0：低电平有效（CMP 定义低电平时间） 1：高电平有效（CMP 定义高电平时间）
26	APWMMODE	RW	0x0	捕获和 APWM 模式选择 0：捕获模式 - 关闭 CAPTSCNT 计数到 PRD 时的自动复位 - 关闭 CAP2/CAP3 对 CAP0/CAP1 的影子功能 - 允许用户使能 CAP0~CAP3 捕获时间戳计数 - CAPx 输入引脚生效 1：APWM 模式 - CAPTSCNT 计数到 PRD 时复位为 0 继续计数 - 开启 CAP2/CAP3 对 CAP0/CAP1 的影子功能 - 禁止捕获时间戳计数到 CAP0~CAP3 寄存器 - APWMx 输出引脚生效
25	FRCSYNC	W1S	0x0	软件强制时间戳计数同步 0：写 0 无效，读总是 0 1：写 1 将 CAPTSCNT 重置为 CAPCNTPHS 本位段总清零
24:23	SYNCOSEL	RW	0x0	同步输出选择 0：禁止输出 1：禁止输出 2：选择同步输入信号做为输出（直通） 3：选择 CAPTSCNT = PRD 事件做为同步输出
22	PHSEN	RW	0x0	相位同步使能 0：关闭相位同步 1：使能相位同步，在输入同步事件或者软件强制的同步事件发生时，将 CAPTSCNT 重置为 CAPCNTPHS
21	TSCNTRUN	RW	0x0	时间戳计数器运行控制

SPINTROL

位段	位段名	属性	复位值	描述
				0：停止计数 1：保持计数
20	REARM	W1S	0x0	模 4 捕获事件计数归位控制 0：写 0 无效，读总是 0 1：写 1 产生如下效果，本位段自动清零 - 复位模 4 捕获事件计数到 0 - 解锁模 4 计数 - 使能时间戳计数捕获
19:18	STOPWRAP	RW	0x3	单轮捕获模式下的停止值或持续捕获模式下的环回值 0：单轮捕获模式下事件 0 发生后即停止 持续捕获模式下事件 0 发生后重置事件计数 1：单轮捕获模式下事件 1 发生后即停止 持续捕获模式下事件 1 发生后重置事件计数 2：单轮捕获模式下事件 2 发生后即停止 持续捕获模式下事件 2 发生后重置事件计数 3：单轮捕获模式下事件 3 发生后即停止 持续捕获模式下事件 3 发生后重置事件计数
17	ONESHOT	RW	0x0	捕获模式选择 0：持续捕获模式 1：单轮捕获模式
16:9	EVTDIV	RW	0x0	事件分频，每 EVTDIV+1 次事件触发一次捕获
8	CAPLDEN	RW	0x0	使能 CAP0~CAP3 的捕获功能 本位段不影响 CEVT0~CEVT3 事件的产生 0：关闭 1：使能
7	CNTRST3	RW	0x0	捕获事件 3 发生时复位事件戳计数 0：不复位计数（绝对时间模式） 1：复位计数（相对时间模式）
6	CAP3POL	RW	0x0	捕获事件 3 极性选择 0：输入下降沿时触发捕获事件 3 1：输入上升沿时触发捕获事件 3
5	CNTRST2	RW	0x0	捕获事件 2 发生时复位事件戳计数 0：不复位计数（绝对时间模式） 1：复位计数（相对时间模式）

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
4	CAP2POL	RW	0x0	捕获事件 2 极性选择 0：输入下降沿时触发捕获事件 2 1：输入上升沿时触发捕获事件 2
3	CNTRST1	RW	0x0	捕获事件 1 发生时复位事件戳计数 0：不复位计数（绝对时间模式） 1：复位计数（相对时间模式）
2	CAP1POL	RW	0x0	捕获事件 1 极性选择 0：输入下降沿时触发捕获事件 1 1：输入上升沿时触发捕获事件 1
1	CNTRST0	RW	0x0	捕获事件 0 发生时复位事件戳计数 0：不复位计数（绝对时间模式） 1：复位计数（相对时间模式）
0	CAP0POL	RW	0x0	捕获事件 0 极性选择 0：输入下降沿时触发捕获事件 0 1：输入上升沿时触发捕获事件 0

SPINTROL

表 9-21：捕获中断原始事件标志寄存器（CAPRAWIF）位段定义

CAPRAWIF (Capture Interrupt Raw Flag Register) Offset：0x24 Default：0x00000000 Access：ECAP->CAPRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 RESERVED	6 CMP	5 PRD	4 CNTOVF	3 CEVT3	2 CEVT2	1 CEVT1	0 CEVT0

表 9-22：捕获中断原始事件标志寄存器（CAPRAWIF）位段定义

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6	CMP	RO	0x0	计数比较相等事件标志 本位段仅在 APWM 模式下有效 0：事件未发生 1：事件已发生
5	PRD	RO	0x0	计数到周期值事件标志 本位段仅在 APWM 模式下有效 0：事件未发生 1：事件已发生
4	CNTOVF	RO	0x0	计数溢出（计数到 0xFFFFFFFF）事件标志 本位段在捕获和 APWM 模式下均有效 0：事件未发生 1：事件已发生
3	CEVT3	RO	0x0	捕获事件 3 标志 本位段仅在捕获模式下有效 0：事件未发生 1：事件已发生
2	CEVT2	RO	0x0	捕获事件 2 标志 本位段仅在捕获模式下有效 0：事件未发生 1：事件已发生
1	CEVT1	RO	0x0	捕获事件 1 标志 本位段仅在捕获模式下有效

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				0：事件未发生 1：事件已发生
0	CEVT0	RO	0x0	捕获事件 0 标志 本位段仅在捕获模式下有效 0：事件未发生 1：事件已发生

表 9-23：捕获中断标志寄存器（CAPIF）位段定义

CAPIF (Capture Interrupt Flag Register) Offset：0x28 Default：0x00000000 Access：ECAP->CAPIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 INT	6 CMP	5 PRD	4 CNTOVF	3 CEVT3	2 CEVT2	1 CEVT1	0 CEVT0

表 9-24：捕获中断标志寄存器（CAPIF）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	INT	RO	0x0	ECAP 全局中断标志 0：中断未发生 1：中断已产生，并已经通知 CPU 本位段被清除之前不会产生新中断
6	CMP	RO	0x0	计数到比较值中断标志 本位段仅在 APWM 模式下有效 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
5	PRD	RO	0x0	计数到周期值中断标志 本位段仅在 APWM 模式下有效 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
4	CNTOVF	RO	0x0	计数溢出中断标志 本位段在捕获和 APWM 模式下均有效

SPINTROL

位段	位段名	属性	复位值	描述
				0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
3	CEVT3	RO	0x0	捕获事件 3 中断标志 本位段仅在捕获模式下有效 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
2	CEVT2	RO	0x0	捕获事件 2 中断标志 本位段仅在捕获模式下有效 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
1	CEVT1	RO	0x0	捕获事件 1 中断标志 本位段仅在捕获模式下有效 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
0	CEVT0	RO	0x0	捕获事件 0 中断标志 本位段仅在捕获模式下有效 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断

表 9-25：捕获中断使能寄存器（CAPIE）位段定义

CAPIE (Capture Interrupt Enable Register) Offset：0x2C Default：0x00000000 Access：ECAP->CAPIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	CMP	PRD	CNTOVF	CEVT3	CEVT2	CEVT1	CEVT0

©2025 旋智电子科技（上海）有限公司

表 9-26：捕获中断使能寄存器（CAPIE）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6	CMP	RW	0x0	计数到比较值中断使能 本位段仅在 APWM 模式下有效 0：关闭 1：使能
5	PRD	RW	0x0	计数到周期值中断使能 本位段仅在 APWM 模式下有效 0：关闭 1：使能
4	CNTOVF	RW	0x0	计数溢出中断使能 本位段在捕获和 APWM 模式下均有效 0：关闭 1：使能
3	CEVT3	RW	0x0	捕获事件 3 中断使能 本位段仅在捕获模式下有效 0：关闭 1：使能
2	CEVT2	RW	0x0	捕获事件 2 中断使能 本位段仅在捕获模式下有效 0：关闭 1：使能
1	CEVT1	RW	0x0	捕获事件 1 中断使能 本位段仅在捕获模式下有效 0：关闭 1：使能
0	CEVT0	RW	0x0	捕获事件 0 中断使能 本位段仅在捕获模式下有效 0：关闭 1：使能

SPINTROL

表 9-27：捕获中断清除寄存器（CAPIC）位段定义

CAPIC (Capture Interrupt Clear Register) Offset：0x30 Default：0x00000000 Access：ECAP->CAPIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 INT	6 CMP	5 PRD	4 CNTOVF	3 CEVT3	2 CEVT2	1 CEVT1	0 CEVT0

表 9-28：捕获中断清除寄存器（CAPIC）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	INT	W1C	0x0	全局中断清除 0：写 0 无效，读总是 0 1：写 1 清除全局中断标志，本位段自动清零
6	CMP	W1C	0x0	计数到比较值中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
5	PRD	W1C	0x0	计数到周期值中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
4	CNTOVF	W1C	0x0	计数溢出中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
3	CEVT3	W1C	0x0	捕获事件 3 中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
2	CEVT2	W1C	0x0	捕获事件 2 中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
1	CEVT1	W1C	0x0	捕获事件 1 中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
0	CEVT0	W1C	0x0	捕获事件 0 中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

表 9-29：捕获中断强制寄存器（CAPIFRC）位段定义

CAPIFRC (Capture Interrupt Event Force Register) Access：ECAP->CAPIFRC Offset：0x34 Default：0x00000000							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 RESERVED	14 CMPWE	13 PRDWE	12 CNTOVFWE	11 CEVT3WE	10 CEVT2WE	9 CEVT1WE	8 CEVT0WE
7 RESERVED	6 CMP	5 PRD	4 CNTOVF	3 CEVT3	2 CEVT2	1 CEVT1	0 CEVT0

表 9-30：捕获中断强制寄存器（CAPIFRC）位段描述

位段	位段名	属性	复位值	描述
31:15	RESERVED_31_15	RO	0x0	保留
14	CMPWE	W1S	0x0	CMP 位段写使能 本位段必须与 CMP 位段同时写 0：禁止更新 1：写 1 使能改写，本位段自动清零
13	PRDWE	W1S	0x0	PRD 位段写使能 本位段必须与 PRD 位段同时写 0：禁止更新 1：写 1 使能改写，本位段自动清零
12	CNTOVFWE	W1S	0x0	CNTOVFWE 位段写使能 本位段必须与 CNTOVFWE 位段同时写 0：禁止更新 1：写 1 使能改写，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
11	CEVT3WE	W1S	0x0	CEVT3 位段写使能 本位段必须与 CEVT3 位段同时写 0：禁止更新 1：写 1 使能改写，本位段自动清零
10	CEVT2WE	W1S	0x0	CEVT2 位段写使能 本位段必须与 CEVT2 位段同时写 0：禁止更新 1：写 1 使能改写，本位段自动清零
9	CEVT1WE	W1S	0x0	CEVT1 位段写使能 本位段必须与 CEVT1 位段同时写 0：禁止更新 1：写 1 使能改写，本位段自动清零
8	CEVT0WE	W1S	0x0	CEVT0 位段写使能 本位段必须与 CEVT0 位段同时写 0：禁止更新 1：写 1 使能改写，本位段自动清零
7	RESERVED_7	RO	0x0	保留
6	CMP	W1S	0x0	强制计数到比较值事件 0：写 0 无效，读总是 0 1：写 1 强制触发事件，本位段自动清零
5	PRD	W1S	0x0	强制计数到周期值事件 0：写 0 无效，读总是 0 1：写 1 强制触发事件，本位段自动清零
4	CNTOVF	W1S	0x0	强制计数溢出事件 0：写 0 无效，读总是 0 1：写 1 强制触发事件，本位段自动清零
3	CEVT3	W1S	0x0	强制捕获事件 3 0：写 0 无效，读总是 0 1：写 1 强制触发事件，本位段自动清零
2	CEVT2	W1S	0x0	强制捕获事件 2 0：写 0 无效，读总是 0 1：写 1 强制触发事件，本位段自动清零
1	CEVT1	W1S	0x0	强制捕获事件 1 0：写 0 无效，读总是 0 1：写 1 强制触发事件，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
0	CEVT0	W1S	0x0	强制捕获事件 0 0：写 0 无效，读总是 0 1：写 1 强制触发事件，本位段自动清零

SPINTROL

表 9-31：ECAP 模块写使能寄存器（CAPREGKEY）位段定义

CAPREGKEY (Capture Register Write-Allow Key Register) Access：ECAP->CAPREGKEY Offset：0x38 Default：0x1ACCE551							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

表 9-32：ECAP 模块写使能寄存器（CAPREGKEY）位段描述

位段	位段名	属性	复位值	描述
31:0	KEY	RW	0x1ACCE551	写入 0x1ACCE551 解锁受保护的 ECAP 寄存器

SPINTROL

©2025 旋智电子科技（上海）有限公司

10 13 位模数转换器（ADC）

10.1 概述

SPC1169 提供 13 位逐次逼近型模数转换器（SAR-ADC），并有多达 9 个外部模拟输入通道可以送入 ADC 采样。

用户可以很方便地使用 ADC 对一个触发源产生一系列的转换。基本的使用原理是基于多个独立配置的 SOC（Start of conversion）转换请求控制通道。

ADC 模块的主要功能包括：

- 13 位分辨率的 ADC 核心，具有一个采样保持（S/H）电路
  - 支持高达 2.5Msps 采样率；
- 模拟满量程输入为 3.657 V
- 16 个独立通道，可独立配置
  - 触发源；
  - 输入信号选择；
  - 采样转换时间和平均次数；
  - DMA 硬件握手接口（仅支持通道 0～7）；
  - 增益和偏移校准；
  - 记录结果的寄存器；
  - 中断处理；
- 支持如下事件触发 SOC 采样转换请求
  - 软件触发请求；
  - 通用定时器 TIMER0、TIMER1、TIMER2 的触发请求；
  - 外部引脚的触发请求；
  - PWM 触发请求，包括：PWM0SOCA～PWM0SOCC，PWM1SOCA～PWM1SOCC，PWM2SOCA～PWM2SOCC，PWM3SOCA～PWM3SOCB；
  - 通道转换完成 EOC（End of conversion）事件；
- 3 个后处理单元
  - 参考值进行比较，参考值大小和比较的极性可配；
  - 对比较结果进行上下限（阈值可配）卡测和过零检测，相应的事件可配置为触发 CPU 中断或者 PWM 封锁。

SPINTROL

10.2 ADC 架构

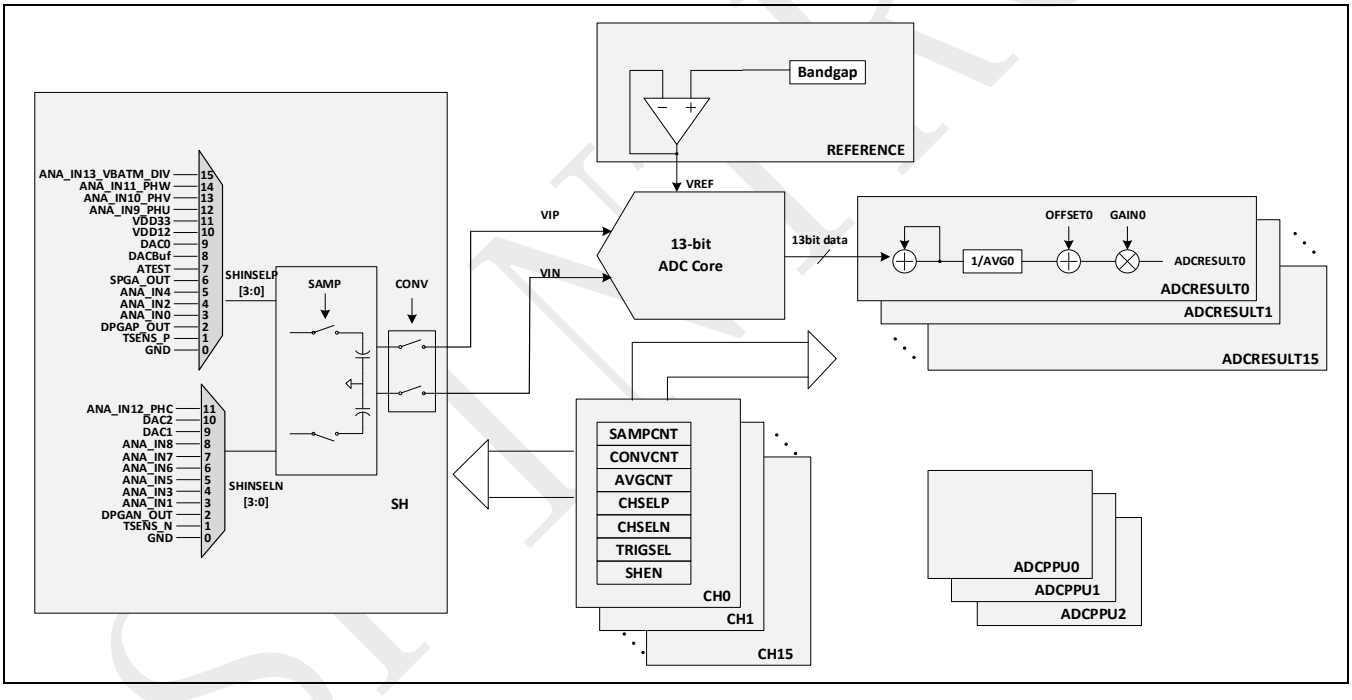
如图 10-1 所示，ADC 的模拟电路部分包括：

- 采样保持电路；
- 核心转换电路；
- 参考信号以及其他配套电路。

数字电路部分包括：

- 16 个控制通道配置；
- 16 个控制通道各自多次采样转换求平均、偏移和增益校准后的最终转换结果；
- 3 个后处理单元（PPU）；
- 与其他模块的接口（图中未画出）。

图 10-1：ADC 框图



10.3 基于控制通道的工作原理

ADC 基于控制通道工作，每个通道可以通过 ADCCHCTL[x]寄存器独立配置，包括：

- 通过 ADCCHCTL[x].TRIGSEL 配置触发 SOC 转换请求的事件；
- 通过 ADCCHCTL[x].SHINSELP 和 ADCCHCTL[x].SHINSELN 配置输入的差分电压信号源；
- 通过 ADCCHCTL[x].AVGCNT 配置对输入多次采样转换并求平均的次数；
- 通过 ADCCHCTL[x].SAMPCNT 配置采样时间；
- 通过 ADCCHCTL[x].CONVCNT 配置转换时间。

©2025 旋智电子科技（上海）有限公司

若有必要，多个控制通道可以有相同的配置，从而可以实现如下场景：

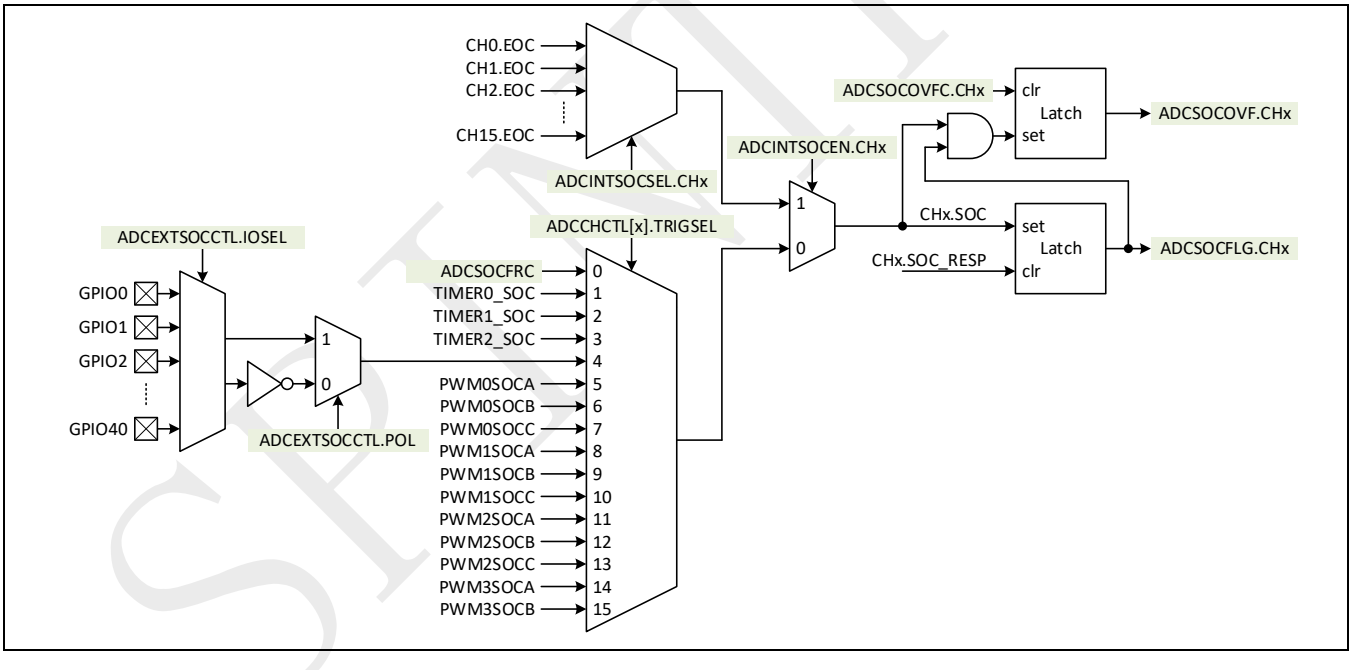
- 在不同的触发事件下对于不同的输入独立采样转换；
- 用单一触发源对同一个输入做过采样；
- 用单一触发源创建一系列不同输入的转换。

10.3.1 SOC 触发源

如图 10-2 所示，当 ADCINTSOCEN.CHx 设为 1 时，通道 x 的 SOC 由 ADCINTSOCSEL.CHx 所指定通道的 EOC 转换完成事件触发；否则，由 ADCCHCTL[x].TRIGSEL 指定的事件触发，包括：

- 软件请求；
- 通用定时器 TIMER0/TIMER1/TIMER2 的 SOC 请求；
- 来自 I/O 引脚的外部请求；
- 来自 PWM 的 SOC 请求；

图 10-2：ADC 通道 SOC 产生



ADCDSOCFLG 寄存器的对应标志位在 SOC 触发时置位，并在 SOC 得到响应（即开始执行）时清零。若通道 x 的 SOC 触发时 ADCSOCFLG.CHx 已经置位，则判定为溢出，并锁存到 ADCSOCOVF.CHx 标志位。溢出标志可以通过往 ADCSOCOVFC.CHx 写 1 清除。

10.3.2 输入选择

通过 ADCCHCTL[x].SHSEL 设置为 1 来启用 ADC 采样器，差分采样器正端和负端如表 10-1 所示。

SPINTROL

表 10-1：ADC 输入选择

选项	ADCCHCTL[x].SHINSELP	ADCCHCTL[x].SHINSELN
15	ANA_IN13_VBATM_DIV[1]	-
14	ANA_IN11_PHW	-
13	ANA_IN10_PHV	-
12	ANA_IN9_PHU	-
11	VDD33	ANA_IN12_PHC
10	VDD12	DAC2
9	DAC0	DAC1
8	DACBUF	ANA_IN8
7	ATEST	ANA_IN7
6	SPGA_OUT	ANA_IN6
5	ANA_IN4	ANA_IN5
4	ANA_IN2	ANA_IN3
3	ANA_IN0	ANA_IN1
2	DPGA_OUTP	DPGA_OUTN
1	TSENS_P	TSENS_N
0	GND	GND

[1] Divided VBATM voltage = VBATM / 13.

10.3.3 采样和转换窗口

采样器电压建立到预期的输入电压所需的时间取决于信号源的驱动力，因此有必要对采样和转换时间窗口大小提供可编程性。这可以通过 ADCCHCTL[x]中的 SAMPCNT 和 CONVCNT 位段实现：

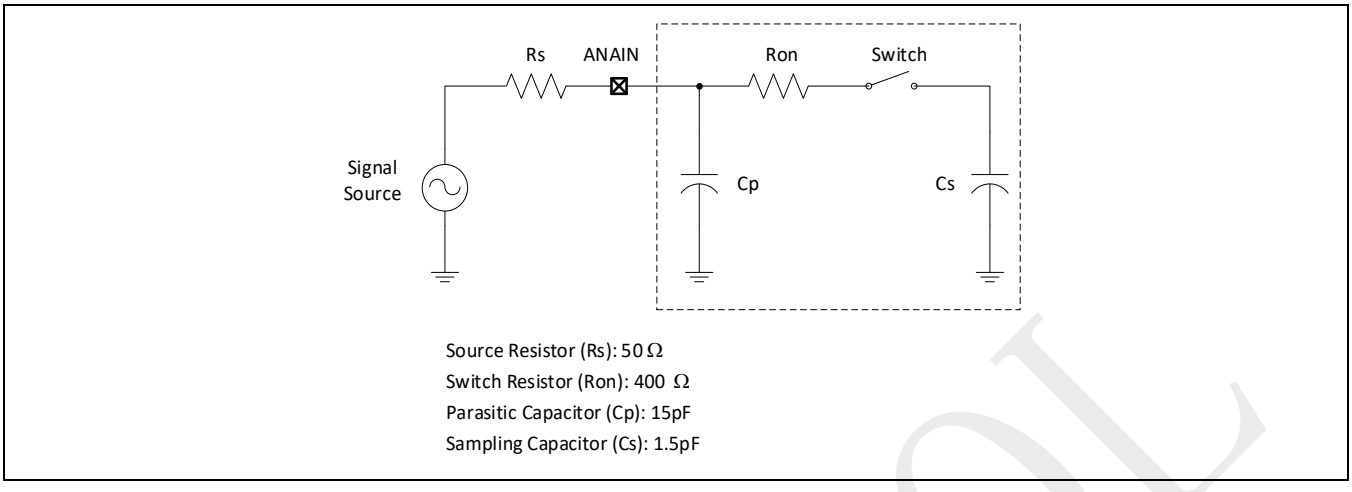
- 通道 x 的采样时间为(ADCCHCTL[x].SAMPCNT + 1)个 ADC 时钟周期；
- 通道 x 的转换时间为(ADCCHCTL[x].CONVCNT + 1)个 ADC 时钟周期；
- 单次处理模拟输入的总时间为采样时间和转换时间的和。

注意：
1. 最小的转换时间为 200ns；
2. 建议采样时间至少为 200ns。

最小采样时间由信号源阻抗、信号路径中的寄生电阻和寄生电容决定。如图 10-3 所示，ANAIN 引脚可以建模为一个 RC 网络，引脚上 0 到 3.3V 的电压摆幅需要约 70ns 的稳定时间才能达到 12 位精度。

©2025 旋智电子科技（上海）有限公司

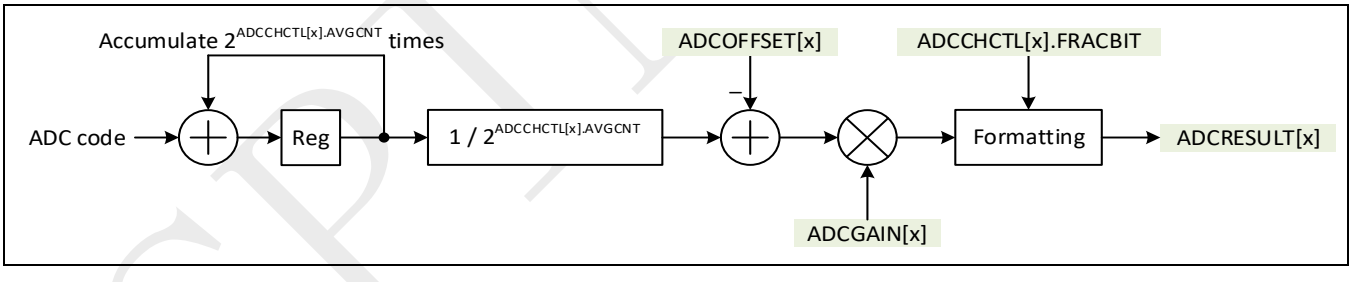
图 10-3：模拟输入引脚模型



10.3.4 结果平均

采样系统通常会受到各种噪声源的影响，例如输入信号噪声、KT/C 噪声、比较器噪声、参考噪声、电源噪声等。为减轻其影响，可以对输入电压连续多次采样转换，累加码值并取平均得到最终结果。 如果用软件来实现，会大大牺牲响应时间和 CPU 资源。因此，如图 10-4 所示，ADC 模块中提供了硬件平均引擎。单个 SOC 请求导致连续 N 次采样和转换，根据 ADCCHCTL[x].AVGCNT 的配置，其中 N 可以是 1、2、4、8、16、32、64 和 128。平均后的最终结果可以保留低 7 位小数部分或截断后只保留整数部分，这由 ADCCHCTL[x].FRACBIT 配置。

图 10-4：ADC 结果平均及偏移和增益校准



10.3.5 偏移和增益校准

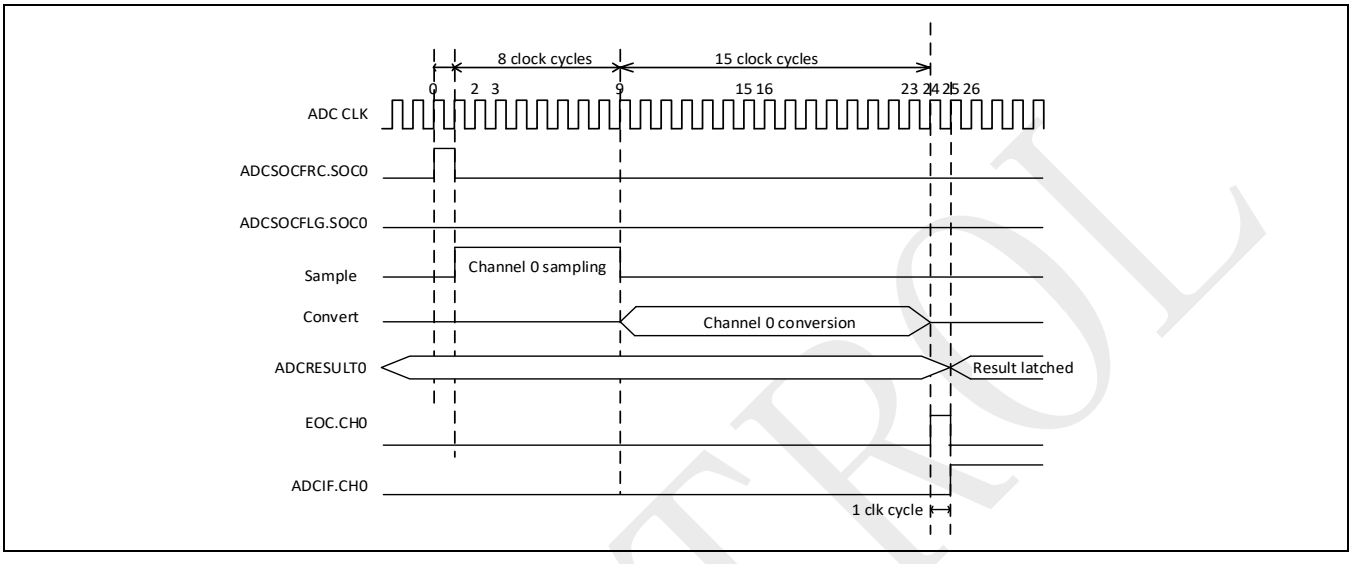
在任何转换器中，都存在偏移误差和满量程增益误差。因此，如图 10-4 所示，模块中引入硬件增益和偏移校正。出厂时，每个 ADC 的偏移误差和增益调整系数都会逐一测试并存储在 Flash 闪存里，并在上电时自动加载到 SHOFFSET 和 SHGAIN 寄存器中。用户可以将它们复制到通道 x 对应的 ADCOFFSET[x]和 ADCGAIN[x]，也可以自行进行实时校准，为每个通道设置独立的 ADCOFFSET[x]和 ADCGAIN[x]。

SPINTROL

10.3.6 时序图

假设 ADCCHCTL[0].SAMPCNT=7 且 ADCCHCTL[0].CONVCNT=14，通道 0 通过软件 SOC 请求触发单次采样和转换的时序如图 10-5 所示。

图 10-5：ADC 时序示例



10.4 ADC 转换优先级

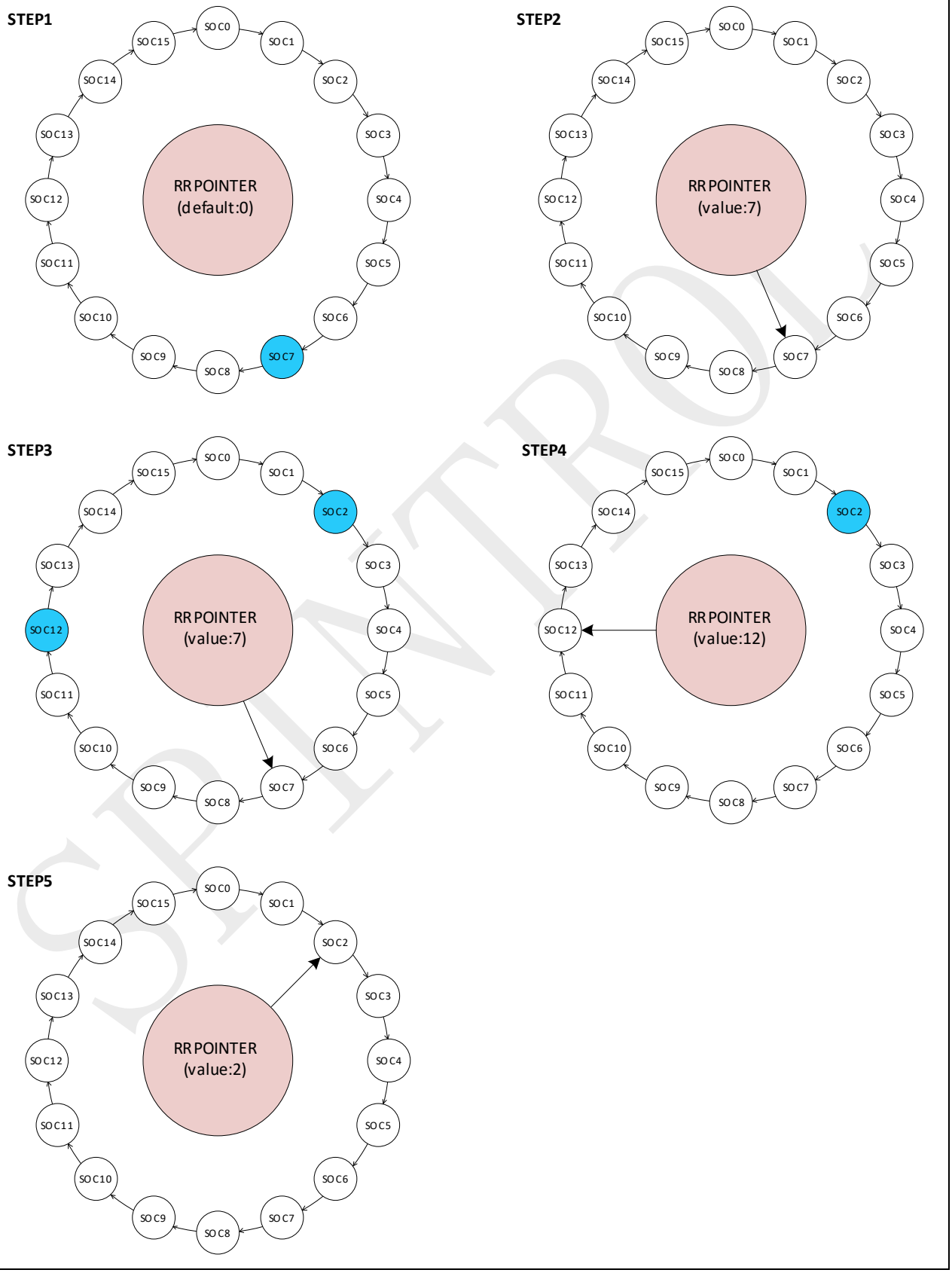
当多个 SOC 请求未决时，采用优先级仲裁来确定它们的执行顺序。
默认的优先级仲裁方案是轮询法（Round robin）。在此方案中，没有任何 SOC 具有比另一个更高的固有优先级。优先级取决于循环指针 RRPOINTER，最高优先级的 SOC 被分配给下一个大于 RRPOINTER 的值，并在 SOC15 之后环回到 SOC0。ADCSOCPRICTL 寄存器的 RRPOINTER 字段指向最后执行的 SOC，芯片复位或向 ADCCTL.RST 写 1 时，RRPOINTER 复位为 0。

图 10-6 给出了循环轮询优先级仲裁的示例。

1. 复位后，SOC0 具有最高优先级；
2. SOC7 被触发；
3. SOC7 立即响应，RRPOINTER 相应地指向 SOC7，现在 SOC8 具有最高优先级。
4. SOC2 和 SOC12 同时触发。
5. 由于 SOC12 在循环轮询中位于 SOC2 之前，先响应 SOC12，而 SOC2 处于挂起状态。RRPOINTER 指向 SOC12。
6. 在 EOC12 后，SOC2 继续执行。RRPOINTER 指向 SOC2，现在 SOC3 具有最高优先级。

©2025 旋智电子科技（上海）有限公司

图 10-6：ADC SOC 轮询仲裁示例



©2025 旋智电子科技（上海）有限公司

ADCSOCPRICTL 寄存器中的 PRIORITY 字段可用于为索引低于 PRIORITY 的一组 SOC 分配更高的优先级。对于组内的 SOC，索引较低的优先级较高。对于组外的 SOC，轮询仲裁仍然适用。例如 PRIORITY=4，则优先级从高到低依次为 SOC0、SOC1、SOC2、SOC3、SOC4~SOC15 组，其中 SOC4~SOC15 组内进行轮询仲裁。

来自优先级较高的组的 SOC 可以在当前转换结束时中断另一组的轮询，并将自己作为下一个要响应的 SOC 插入。当它完成时，如果没有来自具有更高优先级的组的未决 SOC 请求，轮询仲裁将在它被中断的地方继续。

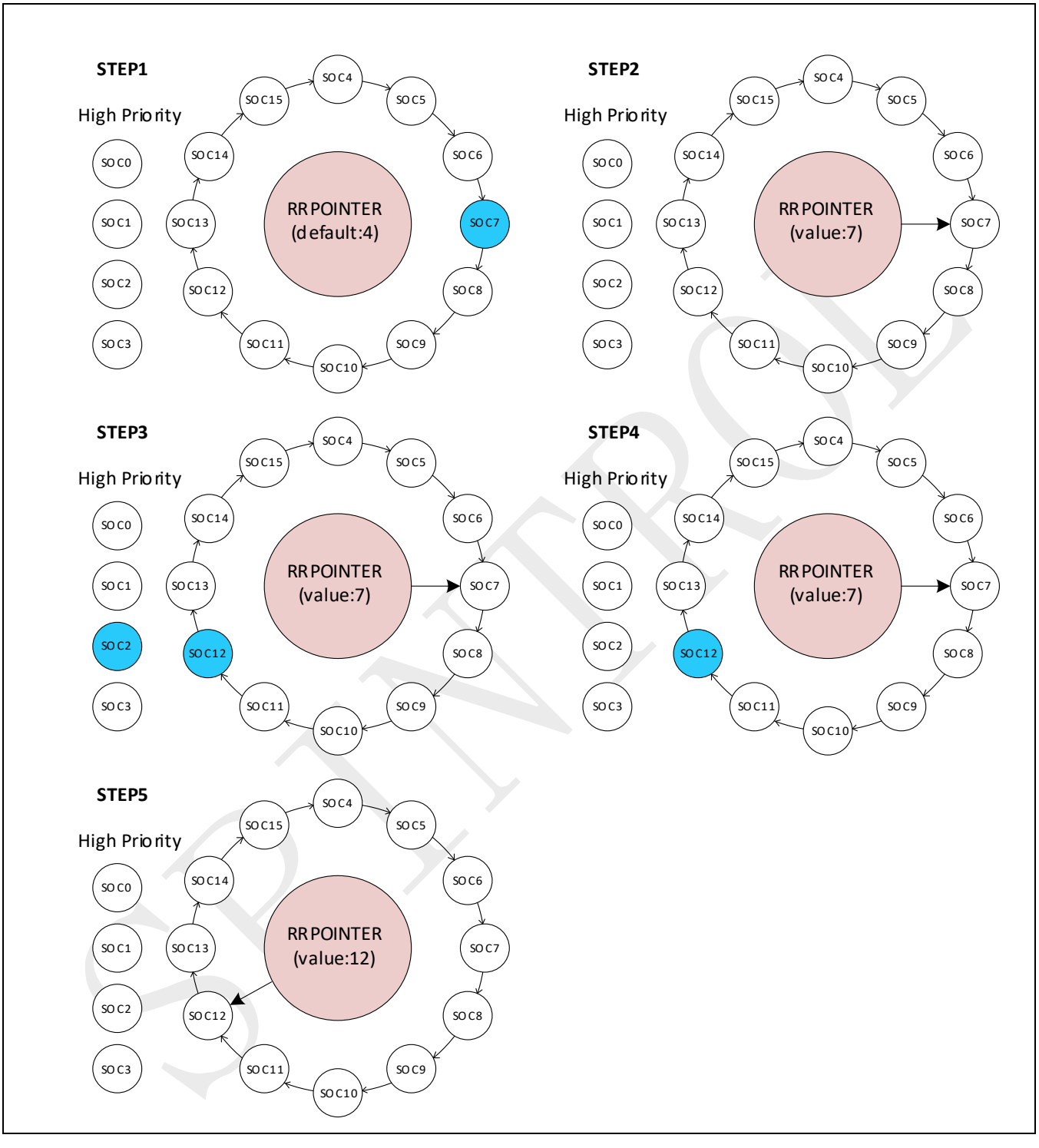
图 10-7 给出了使用高优先级 SOC 的示例，其中 PRIORITY = 4。

1. 由于 PRIORITY 设置为 4，因此 SOC4 在轮询环上排名第一；
2. SOC7 被触发；
3. SOC7 立即得到响应，RRPOINTER 相应地指向 SOC7，SOC8 在轮询环上具有最高优先级；
4. SOC2 和 SOC12 同时触发；
5. SOC2 由于在具有更高优先级的组中所以首先响应，而 SOC12 处于等待状态，RRPOINTER 仍然指向 SOC7；
6. 在 EOC2 时，较高优先级组中没有待处理的 SOC，因此响应挂起的 SOC12。RRPOINTER 指向 SOC12。现在轮到 SOC13 在循环轮询中成为下一个 SOC。

SPINTROL

©2025 旋智电子科技（上海）有限公司

图 10-7：ADC 带优先组的 SOC 仲裁示例

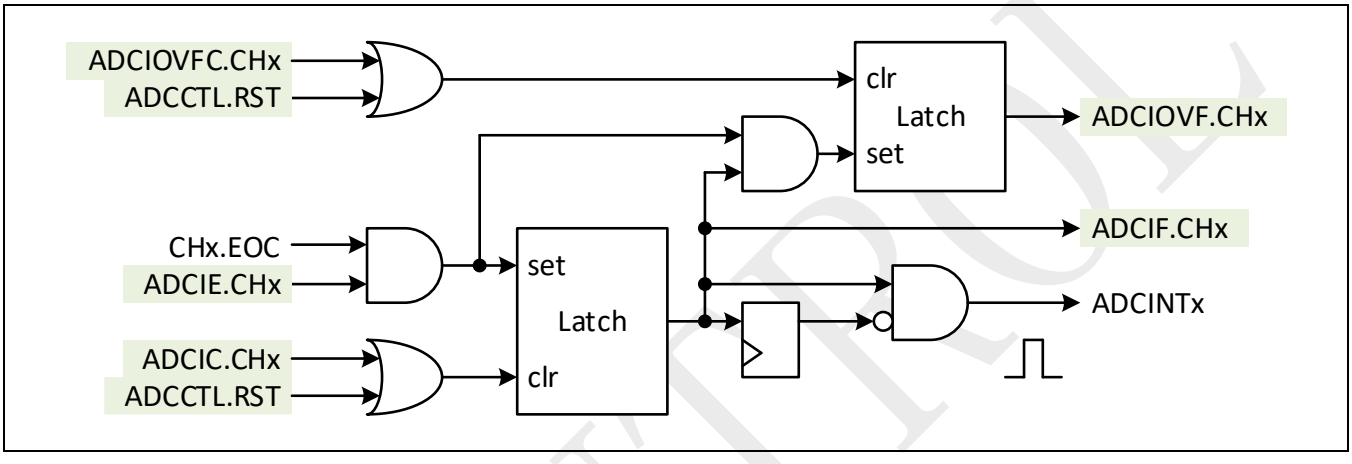


10.5 EOC 及中断

如图 10-8 所示，通道 x 的 EOC（转换结束）事件可以通过将 ADCIE.CHx 设置为 1 来使能，从而产生中断请求 ADCINTx 给 CPU。中断标志锁存在相应的 ADCIF.CHx 中，可以通过向 ADCIC.CHx 或 ADCCTL.RST 写 1 来清除。如果在 ADCIF.CHx 已置位时出现中断请求，则会发生中断溢出事件，该事件将被锁存到 ADCIOVF.CHx。可以通过向 ADCIOVFC.CHx 或 ADCCTL.RST 写 1 来清除。

请注意，如图 10-2 所示，ADCIE.CHx 的配置不影响使用 CHx.EOC 来触发 SOC。

图 10-8：ADC 带优先组的 SOC 仲裁示例



10.6 后处理单元

ADC 数字电路中有 3 个后处理单元（PPU），如图 10-9 所示，它支持以下特性：

- 指定通道的转换结果与可配置的参考进行比较；
- 基于比较结果和可配置的阈值，产生过压、欠压和过零事件；
- 为 PWM 产生波形封锁条件，向 CPU 发出中断请求；
- 测量从指定通道的 SOC 请求被触发到 SOC 请求开始执行的延迟；

10.6.1 参考比较

在许多应用中，需要将 ADC 转换结果与参考进行比较，以获得参考的残余误差或偏移量。PPU 中通过硬件实现该功能，用于减少延迟和软件开销。

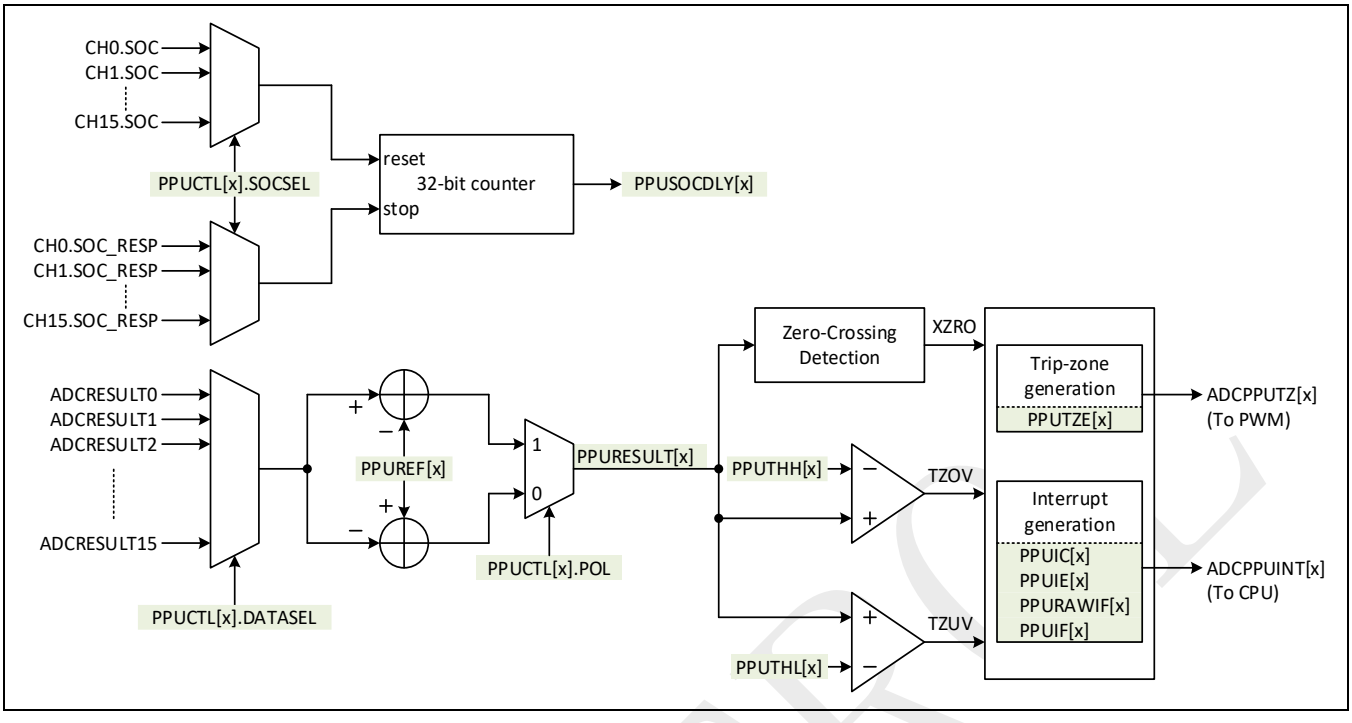
如图 10-9 所示，可以通过将 PPUCTLx.DATASEL 设置为 y 来选择来自通道 y 的 ADCRESULT[y]。它与 PPUREF[x]指定的参考进行比较，极性可通过 PPUCTLx.POL 配置，比较结果如下：

- 当 PPUCTLx.POL = 1 时，PPURESULT[x] = ADCRESULT[y] - PPUREF[x]；
- 当 PPUCTLx.POL = 0 时，PPURESULT[x] = PPUREF[x] - ADCRESULT[y]。

这里的 ADCRESULT[y]、PPUREF[x]和 PPURESULT[x]都是有符号数。此外，无论 ADCCHCTL[y]寄存器的 FRACBIT 位段如何设置，仅使用 ADCRESULT[y]的整数部分参与比较。

图 10-9：ADC 后处理单元框图

©2025 旋智电子科技（上海）有限公司



10.6.2 窗口比较和过零检测

PPURESULT[x]将进一步与预设的窗口进行比较，以检查是否发生过压或欠压事件。窗口的上边界由 PPUTHH[x]决定，下边界由 PPUTHL[x]决定。

此外还会检测 PPURESULT[x]的过零事件，即相应的 ADCRESULT[y]正在越过 PPUREF[x]。

上述3个事件可以通过PPUTZE[x]寄存器位独立启用，合并后以生成PWM的封锁条件。它们也可以通过PPUIE[x]寄存器位独立启用，合并后以生成到CPU的中断。

10.6.3 采样延迟的测量

正如本节中所讨论的，多个 SOC 请求可能会一起出现，但会按优先级依次响应和执行。因此，从 SOC 请求被触发到对应通道的输入电压被真正采样转换的时间可能存在延迟。它在控制系统中引入了误差。测量此延迟将很有帮助，由此软件可以进行一些插值和补偿，以最大程度地减少由其导致的错误。

如图 10-9 所示，该功能在后处理单元中通过 32 位计数器实现。SOC 请求的通道可以通过 PPUCTL[x].SOCSEL 独立选择。每当触发选定的 SOC 时，计数器就会从 0 开始计数。只要 SOC 请求响应并且电压采样启动，计数器停止，值将锁存到 PPUSOCDLY [x]。用户可以将其用于插值或补偿相关的代码里。

10.7 预期码值的计算

ADC 所需的参考电压由带隙基准电路产生。量化的满幅值是 3.657V，从而差分输入范围为-3.657V~3.657V。但各通道实际的输入电压不应该超过电源电压（通常是 3.3V）。

当 ADCCHCTL[x].FRACBIT=0 时，ADCRESULT[x]中的所有小数位将被截断。因此，输出码值范围为-4096 至 4095。对于 AIP 和 AIN 的差分输入，期望的代码范围将为：

$$ADCRESULT[x] = \frac{(AIP - AIN)}{3.657} \times 4096$$

当 ADCCHCTL[x].FRACBIT=1 时，ADCRESULT[x]中的最低 7 位实际上是小数部分。因此，输出码值范围为-524288 至 524287。对于 AIP 和 AIN 的差分输入，期望的代码范围将为：

$$ADCRESULT[x] = \frac{(AIP - AIN)}{3.657} \times 524288$$

SHRAWCODE 寄存器中的码值始终包含低 7 位的小数部分。

10.8 安全特性

10.8.1 输入浮空检测

如果用于模拟输入的引脚未能可靠地连接到外部传感器，则无法正确测量电压，从而在某些应用中导致潜在的严重损坏。ADC 模块中提供了输入浮空检测电路来找出此类错误，从而激活用户自定义的故障安全程序。

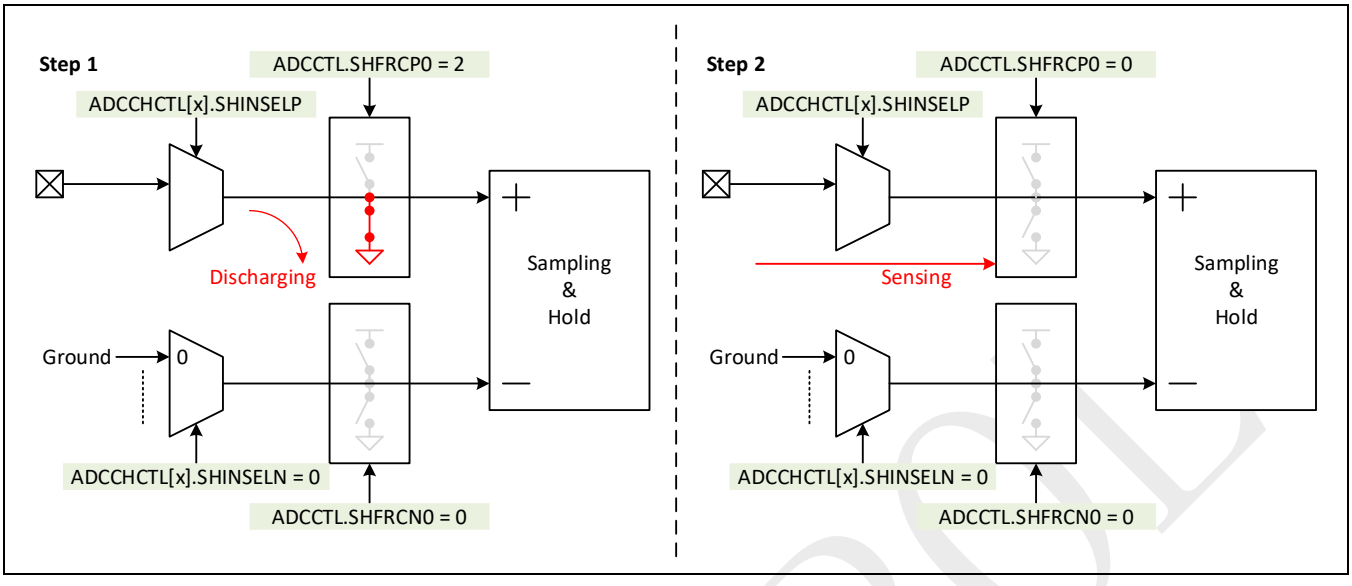
有两种可选方案来检测输入浮空，它们都是基于故障注入，但分别是将节点充电至电源和将节点放电至地。

如图 10-10 所示，通过节点放电的故障注入，可以分两步检测连接到采样器正输入的引脚是否浮空。

- 对输入节点放电
  - 通过配置 ADCCHCTL[x].SHINSELP 将采样器的正输入端连接到外部引脚；
  - 设置 ADCCTL.SHFRCP0 = 2，从而将采样器的正输入端下拉到地；
  - 等待大约 10us 使得输入节点完全放电到电压为 0。

©2025 旋智电子科技（上海）有限公司

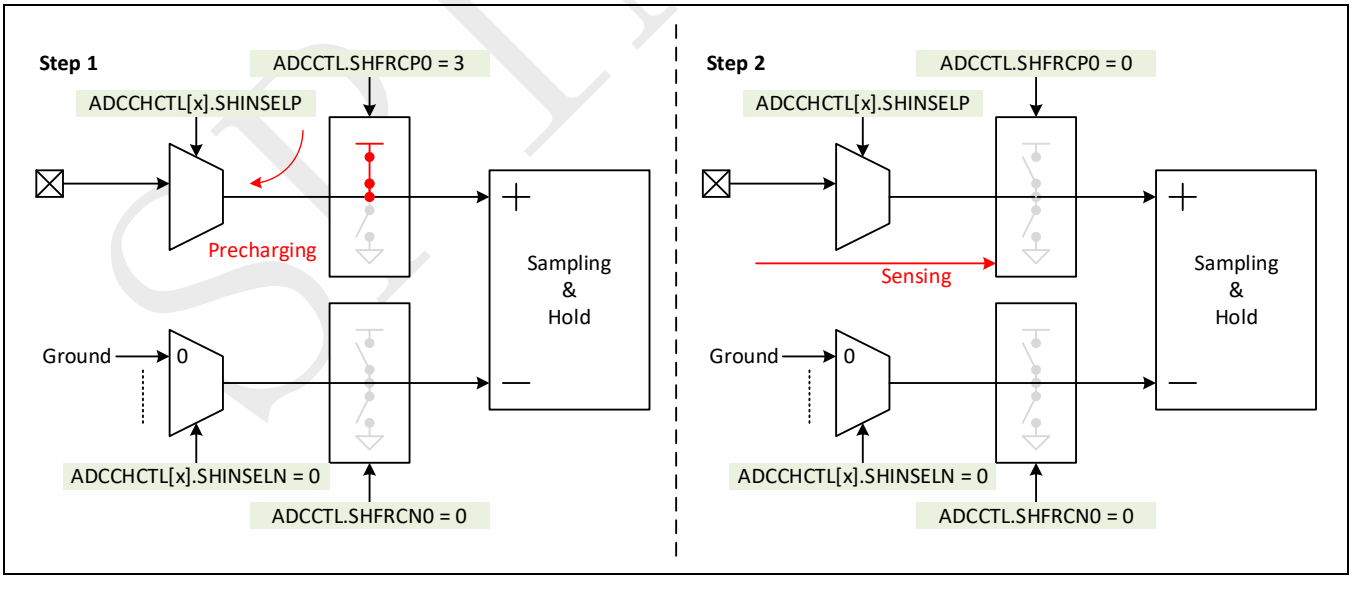
图 10-10：输入引脚悬空的检测（通过节点放电的故障注入）



- 恢复正常连接，检测节点连接
  - 设置 ADCCTL.SHFRCP0 = 0，关闭下拉通路；
  - 设置 ADCCHCTL[x].SHINSELN = 0，将采样器负输入端连到地；
  - 触发通道 x 的 SOC 并得到转换结果 ADCRESULT[x]；
  - 如果回读值总是 0，表明采样器正输入端不能跟踪外部输入的变化，即正输入端浮空。

  用类似的方法可以检测采样器负输入端是否浮空。

图 10-11：输入引脚悬空的检测（通过节点充电的故障注入）



©2025 旋智电子科技（上海）有限公司

如图 10-11 所示，通过节点充电的故障注入，可以分两步检测连接到采样器正输入的引脚是否悬空。

- 对输入节点充电
  - 通过配置 ADCCHCTL[x].SHINSELP 将采样器的正输入端连接到外部引脚；
  - 设置 ADCCTL.SHFRCP0 = 3，从而将采样器的正输入端上拉到电源；
  - 等待大约 10us 使得输入节点完全充电到电源电压。
- 恢复正常连接，检测节点连接
  - 设置 ADCCTL.SHFRCP0 = 0，关闭上拉通路；
  - 设置 ADCCHCTL[x].SHINSELN = 0，将采样器负输入端连到地；
  - 触发通道 x 的 SOC 并得到转换结果 ADCRESULT[x]；
  - 如果回读值总是 3696（3.3 / 3.657 x 4096）附近，表明采样器正输入端不能跟踪外部输入的变化，即正输入端浮空。

用类似的方法可以检测采样器负输入端是否浮空。

10.8.2 输入短路检测

模拟输入也可能出现短路情况。可以按照如下步骤使用充电或放电电路注入故障来检测：

- 获取正常连接时的码值
  - 设置 ADCCTL.SHFRCP0 和 ADCCTL.SHFRCN0 为 0，关闭对采样器输入的上下拉；
  - 通过配置 ADCCHCTL[x].SHINSELP 和 ADCCHCTL[x].SHINSELN 配置采样器的输入引脚；
  - 触发通道 x 的 SOC 并得到第一个 ADCRESULT[x]。
- 获取通过充电注入故障时的码值
  - 根据采样器具体是正端或者负端连接到外部模拟输入引脚，配置 ADCCTL.SHFRCP0 或 ADCCTL.SHFRCN0 为 3，使能充电故障注入；
  - 等待大约 10us 从而输入节点充电完成；
  - 触发通道 x 的 SOC 并得到第二个 ADCRESULT[x]。
- 获取通过放电注入故障时的码值
  - 根据采样器具体是正端或者负端连接到外部模拟输入引脚，配置 ADCCTL.SHFRCP0 或 ADCCTL.SHFRCN0 为 2，使能放电故障注入；
  - 等待大约 10us 从而输入节点放电完成；
  - 触发通道 x 的 SOC 并得到第三个 ADCRESULT[x]。
- 若三个 ADCRESULT[x]非常接近，则判定为输入短路。

注意： 所能检测的短路工况仅限于输入短路到一个低阻节点，其驱动力超过上下拉诊断电路提供的 8uA 的充放电能力。

©2025 旋智电子科技（上海）有限公司

10.9 寄存器

10.9.1 ADC 寄存器列表

表 10-2：ADC 模块基地址

外设模块	基地址
ADC	0x4000 3000

表 10-3：ADC 寄存器列表

寄存器	偏移地址	描述	复位值
ADCIF	0x0	ADC 中断标志寄存器	0x00000000
ADCIC	0x4	ADC 中断清除寄存器	0x00000000
ADCIOVF	0x8	ADC 中断溢出标志寄存器	0x00000000
ADCIOVFC	0xC	ADC 中断溢出清除寄存器	0x00000000
ADCIE*	0x10	ADC 中断使能寄存器	0x00000000
ADCSOCPRICTL*	0x14	ADC SOC 优先级控制寄存器	0x00000000
ADCSOCFLG	0x18	ADC SOC 请求标志寄存器	0x00000000
ADCSOCFRC	0x1C	ADC SOC 请求强制寄存器	0x00000000
ADCSOCOVF	0x20	ADC SOC 请求溢出标志寄存器	0x00000000
ADCSOCOVFC	0x24	ADC SOC 请求溢出清除寄存器	0x00000000
ADCINTSOCEN*	0x28	ADC EOC 中断事件触发 SOC 使能寄存器	0x00000000
ADCINTSOCSEL*	0x2C	ADC EOC 中断事件触发 SOC 选择寄存器	0x00000000
ADCEXTSOCCTL*	0x34	ADC 外部 SOC 输入控制寄存器	0x00000001
ADCCHCTL*	0x38	ADC 通道 n ([0:15])控制寄存器	0x00000000
ADCOFFSET*	0x78	ADC 通道 n ([0:15])偏移校准寄存器	0x00000000
ADCGAIN*	0xB8	ADC 通道 n ([0:15])增益校准寄存器	0x00008000
SHOFFSET*	0xF8	ADC 采样转换核心电路偏移校准寄存器	0x00000000
SHGAIN*	0x104	ADC 采样转换核心电路增益校准寄存器	0x00008000
SHFLG	0x110	ADC 采样转换 EOC 标志寄存器	0x00000000
SHFLGCLR	0x114	ADC 采样保持通道 EOC 标志清除寄存器	0x00000000
ADCCTL*	0x118	ADC 控制寄存器	0x00000000
SHRAWCODE	0x11C	ADC 采样转换原始码值寄存器	0x00000000
ADCRESULT	0x128	ADC 结果寄存器	0x00000000
PPURESULT	0x168	ADC 通道 n ([0:15])结果寄存器	0x00000000
PPUSOCDLY	0x188	后处理单元 n ([0:2])比较结果寄存器	0x00000000
PPURAWIF	0x1A8	后处理单元 n ([0:2]) SOC 响应延时寄存器	0x00000000
PPUIF	0x1C8	后处理单元 n ([0:2])中断原始事件标志寄存器	0x00000000
PPUIC	0x1E8	后处理单元 n ([0:2])中断标志寄存器	0x00000000
PPUIE*	0x208	后处理单元 n ([0:2])中断清除寄存器	0x00000000
PPUTZE*	0x228	后处理单元 n ([0:2])中断使能寄存器	0x00000000
PPUCTL*	0x248	后处理单元 n ([0:2])封锁事件使能寄存器	0x00000400

©2025 旋智电子科技（上海）有限公司

寄存器	偏移地址	描述	复位值
PPUREF*	0x268	后处理单元 n ([0:2])控制寄存器	0x00000000
PPUTHL*	0x288	后处理单元 n ([0:2])比较参考寄存器	0x00000000
PPUTHH*	0x2A8	后处理单元 n ([0:2])下限阈值寄存器	0x00000000
TSENSCTL*	0x2C8	后处理单元 n ([0:2])上限阈值寄存器	0x00000000
ADCREGKEY	0x2CC	温度传感器控制寄存器	0x1ACCE551

注意： 由*标记的寄存器仅当 PWRREGKEY=0x1ACCE551 时才可以改写。

SPINTROL

©2025 旋智电子科技（上海）有限公司

10.9.2 ADC 寄存器

表 10-4：ADC 中断标志寄存器（ADCIF）位段定义

ADCIF (ADC Interrupt Flag Register) Offset：0x0 Default：0x00000000 Access：ADCx->ADCIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH15	14 CH14	13 CH13	12 CH12	11 CH11	10 CH10	9 CH9	8 CH8
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 10-5：ADC 中断标志寄存器（ADCIF）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	CH15	RO	0x0	ADC 通道 15 中断标志 0：中断未发生 1：中断已发生
14	CH14	RO	0x0	ADC 通道 14 中断标志 0：中断未发生 1：中断已发生
13	CH13	RO	0x0	ADC 通道 13 中断标志 0：中断未发生 1：中断已发生
12	CH12	RO	0x0	ADC 通道 12 中断标志 0：中断未发生 1：中断已发生
11	CH11	RO	0x0	ADC 通道 11 中断标志 0：中断未发生 1：中断已发生
10	CH10	RO	0x0	ADC 通道 10 中断标志 0：中断未发生 1：中断已发生
9	CH9	RO	0x0	ADC 通道 9 中断标志 0：中断未发生 1：中断已发生

SPINTROL

位段	位段名	属性	复位值	描述
8	CH8	RO	0x0	ADC 通道 8 中断标志 0：中断未发生 1：中断已发生
7	CH7	RO	0x0	ADC 通道 7 中断标志 0：中断未发生 1：中断已发生
6	CH6	RO	0x0	ADC 通道 6 中断标志 0：中断未发生 1：中断已发生
5	CH5	RO	0x0	ADC 通道 5 中断标志 0：中断未发生 1：中断已发生
4	CH4	RO	0x0	ADC 通道 4 中断标志 0：中断未发生 1：中断已发生
3	CH3	RO	0x0	ADC 通道 3 中断标志 0：中断未发生 1：中断已发生
2	CH2	RO	0x0	ADC 通道 2 中断标志 0：中断未发生 1：中断已发生
1	CH1	RO	0x0	ADC 通道 1 中断标志 0：中断未发生 1：中断已发生
0	CH0	RO	0x0	ADC 通道 0 中断标志 0：中断未发生 1：中断已发生

SPINTROL

©2025 旋智电子科技（上海）有限公司

表 10-6：ADC 中断清除寄存器（ADCIC）位段定义

ADCIC (ADC Interrupt Flag Clear Register) Offset：0x4 Default：0x00000000 Access：ADCx->ADCIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH15	14 CH14	13 CH13	12 CH12	11 CH11	10 CH10	9 CH9	8 CH8
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 10-7：ADC 中断清除寄存器（ADCIC）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	CH15	W1C	0x0	ADC 通道 15 中断清除 0：写 0 无效，读总是 0 1：写 1 清除对应的中断标志，本位段自动清零
14	CH14	W1C	0x0	ADC 通道 14 中断清除 0：写 0 无效，读总是 0 1：写 1 清除对应的中断标志，本位段自动清零
13	CH13	W1C	0x0	ADC 通道 13 中断清除 0：写 0 无效，读总是 0 1：写 1 清除对应的中断标志，本位段自动清零
12	CH12	W1C	0x0	ADC 通道 12 中断清除 0：写 0 无效，读总是 0 1：写 1 清除对应的中断标志，本位段自动清零
11	CH11	W1C	0x0	ADC 通道 11 中断清除 0：写 0 无效，读总是 0 1：写 1 清除对应的中断标志，本位段自动清零
10	CH10	W1C	0x0	ADC 通道 10 中断清除 0：写 0 无效，读总是 0 1：写 1 清除对应的中断标志，本位段自动清零
9	CH9	W1C	0x0	ADC 通道 9 中断清除

SPINTROL

位段	位段名	属性	复位值	描述
				0：写 0 无效，读总是 0 1：写 1 清除对应的中断标志，本位段自动清零
8	CH8	W1C	0x0	ADC 通道 8 中断清除 0：写 0 无效，读总是 0 1：写 1 清除对应的中断标志，本位段自动清零
7	CH7	W1C	0x0	ADC 通道 7 中断清除 0：写 0 无效，读总是 0 1：写 1 清除对应的中断标志，本位段自动清零
6	CH6	W1C	0x0	ADC 通道 6 中断清除 0：写 0 无效，读总是 0 1：写 1 清除对应的中断标志，本位段自动清零
5	CH5	W1C	0x0	ADC 通道 5 中断清除 0：写 0 无效，读总是 0 1：写 1 清除对应的中断标志，本位段自动清零
4	CH4	W1C	0x0	ADC 通道 4 中断清除 0：写 0 无效，读总是 0 1：写 1 清除对应的中断标志，本位段自动清零
3	CH3	W1C	0x0	ADC 通道 3 中断清除 0：写 0 无效，读总是 0 1：写 1 清除对应的中断标志，本位段自动清零
2	CH2	W1C	0x0	ADC 通道 2 中断清除 0：写 0 无效，读总是 0 1：写 1 清除对应的中断标志，本位段自动清零
1	CH1	W1C	0x0	ADC 通道 1 中断清除 0：写 0 无效，读总是 0 1：写 1 清除对应的中断标志，本位段自动清零
0	CH0	W1C	0x0	ADC 通道 0 中断清除 0：写 0 无效，读总是 0 1：写 1 清除对应的中断标志，本位段自动清零

©2025 旋智电子科技（上海）有限公司

表 10-8：ADC 中断溢出标志寄存器（ADCIOVF）位段定义

ADCIOVF (ADC Interrupt Overflow Flag Register) Offset：0x8 Default：0x00000000 Access：ADCx->ADCIOVF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH15	14 CH14	13 CH13	12 CH12	11 CH11	10 CH10	9 CH9	8 CH8
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 10-9：ADC 中断溢出标志寄存器（ADCIOVF）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	CH15	RO	0x0	ADC 通道 15 中断溢出标志 0：中断溢出未发生 1：中断溢出已发生
14	CH14	RO	0x0	ADC 通道 14 中断溢出标志 0：中断溢出未发生 1：中断溢出已发生
13	CH13	RO	0x0	ADC 通道 13 中断溢出标志 0：中断溢出未发生 1：中断溢出已发生
12	CH12	RO	0x0	ADC 通道 12 中断溢出标志 0：中断溢出未发生 1：中断溢出已发生
11	CH11	RO	0x0	ADC 通道 11 中断溢出标志 0：中断溢出未发生 1：中断溢出已发生
10	CH10	RO	0x0	ADC 通道 10 中断溢出标志 0：中断溢出未发生 1：中断溢出已发生
9	CH9	RO	0x0	ADC 通道 9 中断溢出标志 0：中断溢出未发生 1：中断溢出已发生

SPINTROL

位段	位段名	属性	复位值	描述
8	CH8	RO	0x0	ADC 通道 8 中断溢出标志 0：中断溢出未发生 1：中断溢出已发生
7	CH7	RO	0x0	ADC 通道 7 中断溢出标志 0：中断溢出未发生 1：中断溢出已发生
6	CH6	RO	0x0	ADC 通道 6 中断溢出标志 0：中断溢出未发生 1：中断溢出已发生
5	CH5	RO	0x0	ADC 通道 5 中断溢出标志 0：中断溢出未发生 1：中断溢出已发生
4	CH4	RO	0x0	ADC 通道 4 中断溢出标志 0：中断溢出未发生 1：中断溢出已发生
3	CH3	RO	0x0	ADC 通道 3 中断溢出标志 0：中断溢出未发生 1：中断溢出已发生
2	CH2	RO	0x0	ADC 通道 2 中断溢出标志 0：中断溢出未发生 1：中断溢出已发生
1	CH1	RO	0x0	ADC 通道 1 中断溢出标志 0：中断溢出未发生 1：中断溢出已发生
0	CH0	RO	0x0	ADC 通道 0 中断溢出标志 0：中断溢出未发生 1：中断溢出已发生

表 10-10：ADC 中断溢出清除寄存器（ADCIOVFC）位段定义

ADCIOVFC (ADC Interrupt Overflow Flag Clear Register) Offset：0xC Default：0x00000000 Access：ADCx->ADCIOVFC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH15	14 CH14	13 CH13	12 CH12	11 CH11	10 CH10	9 CH9	8 CH8
7	6	5	4	3	2	1	0

©2025 旋智电子科技（上海）有限公司

CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0

表 10-11：ADC 中断溢出清除寄存器（ADCIOVFC）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	CH15	W1C	0x0	ADC 通道 15 中断溢出清除 0：写 0 无效，读总是 0 1：写 1 清除对应的溢出标志，本位段自动清零
14	CH14	W1C	0x0	ADC 通道 14 中断溢出清除 0：写 0 无效，读总是 0 1：写 1 清除对应的溢出标志，本位段自动清零
13	CH13	W1C	0x0	ADC 通道 13 中断溢出清除 0：写 0 无效，读总是 0 1：写 1 清除对应的溢出标志，本位段自动清零
12	CH12	W1C	0x0	ADC 通道 12 中断溢出清除 0：写 0 无效，读总是 0 1：写 1 清除对应的溢出标志，本位段自动清零
11	CH11	W1C	0x0	ADC 通道 11 中断溢出清除 0：写 0 无效，读总是 0 1：写 1 清除对应的溢出标志，本位段自动清零
10	CH10	W1C	0x0	ADC 通道 10 中断溢出清除 0：写 0 无效，读总是 0 1：写 1 清除对应的溢出标志，本位段自动清零
9	CH9	W1C	0x0	ADC 通道 9 中断溢出清除 0：写 0 无效，读总是 0 1：写 1 清除对应的溢出标志，本位段自动清零
8	CH8	W1C	0x0	ADC 通道 8 中断溢出清除 0：写 0 无效，读总是 0 1：写 1 清除对应的溢出标志，本位段自动清零

SPINTROL

位段	位段名	属性	复位值	描述
7	CH7	W1C	0x0	ADC 通道 7 中断溢出清除 0：写 0 无效，读总是 0 1：写 1 清除对应的溢出标志，本位段自动清零
6	CH6	W1C	0x0	ADC 通道 6 中断溢出清除 0：写 0 无效，读总是 0 1：写 1 清除对应的溢出标志，本位段自动清零
5	CH5	W1C	0x0	ADC 通道 5 中断溢出清除 0：写 0 无效，读总是 0 1：写 1 清除对应的溢出标志，本位段自动清零
4	CH4	W1C	0x0	ADC 通道 4 中断溢出清除 0：写 0 无效，读总是 0 1：写 1 清除对应的溢出标志，本位段自动清零
3	CH3	W1C	0x0	ADC 通道 3 中断溢出清除 0：写 0 无效，读总是 0 1：写 1 清除对应的溢出标志，本位段自动清零
2	CH2	W1C	0x0	ADC 通道 2 中断溢出清除 0：写 0 无效，读总是 0 1：写 1 清除对应的溢出标志，本位段自动清零
1	CH1	W1C	0x0	ADC 通道 1 中断溢出清除 0：写 0 无效，读总是 0 1：写 1 清除对应的溢出标志，本位段自动清零
0	CH0	W1C	0x0	ADC 通道 0 中断溢出清除 0：写 0 无效，读总是 0 1：写 1 清除对应的溢出标志，本位段自动清零

表 10-12：ADC 中断使能寄存器（ADCIE）位段定义

ADCIE (ADC Interrupt Enable Register)　Offset：0x10　Default：0x00000000 Access：ADCx->ADCIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16

©2025 旋智电子科技（上海）有限公司

RESERVED							
15 CH15	14 CH14	13 CH13	12 CH12	11 CH11	10 CH10	9 CH9	8 CH8
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 10-13：ADC 中断使能寄存器（ADCIE）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	CH15	RW	0x0	ADC 通道 15 中断使能 0：关闭 1：使能
14	CH14	RW	0x0	ADC 通道 14 中断使能 0：关闭 1：使能
13	CH13	RW	0x0	ADC 通道 13 中断使能 0：关闭 1：使能
12	CH12	RW	0x0	ADC 通道 12 中断使能 0：关闭 1：使能
11	CH11	RW	0x0	ADC 通道 11 中断使能 0：关闭 1：使能
10	CH10	RW	0x0	ADC 通道 10 中断使能 0：关闭 1：使能
9	CH9	RW	0x0	ADC 通道 9 中断使能 0：关闭 1：使能
8	CH8	RW	0x0	ADC 通道 8 中断使能 0：关闭 1：使能
7	CH7	RW	0x0	ADC 通道 7 中断使能 0：关闭 1：使能
6	CH6	RW	0x0	ADC 通道 6 中断使能 0：关闭 1：使能

SPINTROL

位段	位段名	属性	复位值	描述
5	CH5	RW	0x0	ADC 通道 5 中断使能 0：关闭 1：使能
4	CH4	RW	0x0	ADC 通道 4 中断使能 0：关闭 1：使能
3	CH3	RW	0x0	ADC 通道 3 中断使能 0：关闭 1：使能
2	CH2	RW	0x0	ADC 通道 2 中断使能 0：关闭 1：使能
1	CH1	RW	0x0	ADC 通道 1 中断使能 0：关闭 1：使能
0	CH0	RW	0x0	ADC 通道 0 中断使能 0：关闭 1：使能

SPINTROL

©2025 旋智电子科技（上海）有限公司

表 10-14：ADC SOC 优先级控制寄存器（ADCSOCPRICTL）位段定义

ADCSOCPRICTL (ADC Start of Conversion Priority Control Register) Offset：0x14 Default：0x00000000 Access：ADCx->ADCSOCPRICTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RRPOINTER				PRIORITY			

表 10-15：ADC SOC 优先级控制寄存器（ADCSOCPRICTL）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7:4	RRPOINTER	RO	0x0	Round Ring Pointer Holds the latest SOCx in Round Ring, and determine the next order of conversion. 0000：SOC0 是最后执行的，SOC1 优先级最高 0001：SOC1 是最后执行的，SOC2 优先级最高 0010：SOC2 是最后执行的，SOC3 优先级最高 0011：SOC3 是最后执行的，SOC4 优先级最高 0100：SOC4 是最后执行的，SOC5 优先级最高 0101：SOC5 是最后执行的，SOC6 优先级最高 0110：SOC6 是最后执行的，SOC7 优先级最高 0111：SOC7 是最后执行的，SOC8 优先级最高 …… 1111：SOC15 是最后执行的，SOC0 优先级最高
3:0	PRIORITY	RW	0x0	SOC 优先级，决定具有高优先级的 SOC 组 0000：所有 SOC 采用轮询方式仲裁优先级 0001：SOC0 有高优先级，SOC1-SOC7 用轮询仲裁 0010：SOC0-SOC1 有高优先级，SOC2-SOC7 用轮询仲裁 0011：SOC0-SOC2 有高优先级，SOC3-SOC7 用轮询仲裁 0100：SOC0-SOC3 有高优先级，SOC4-SOC7 用轮询仲裁 0101：SOC0-SOC4 有高优先级，SOC5-SOC7 用轮询仲裁 0110：SOC0-SOC5 有高优先级，SOC6-SOC7 用轮询仲裁

SPINTROL

位段	位段名	属性	复位值	描述
				0111：SOC0-SOC6 有高优先级，SOC7-SOC8 用轮询仲裁 …… 1111：SOC0-SOC14 有最高优先级，SOC15 用轮训仲裁

表 10-16：ADC SOC 请求标志寄存器（ADCSOCFLG）位段定义

ADCSOCFLG (ADC SOC Flag Register) Offset：0x18 Default：0x00000000 Access：ADCx->ADCSOCFLG							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 SOC15	14 SOC14	13 SOC13	12 SOC12	11 SOC11	10 SOC10	9 SOC9	8 SOC8
7 SOC7	6 SOC6	5 SOC5	4 SOC4	3 SOC3	2 SOC2	1 SOC1	0 SOC0

表 10-17：ADC SOC 请求标志寄存器（ADCSOCFLG）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	SOC15	RO	0x0	通道 15 的 SOC 请求标志 0：没有事件触发 SOC 请求 1：有事件触发 SOC 请求，采样开始后位段清零
14	SOC14	RO	0x0	通道 14 的 SOC 请求标志 0：没有事件触发 SOC 请求 1：有事件触发 SOC 请求，采样开始后位段清零
13	SOC13	RO	0x0	通道 13 的 SOC 请求标志 0：没有事件触发 SOC 请求 1：有事件触发 SOC 请求，采样开始后位段清零
12	SOC12	RO	0x0	通道 12 的 SOC 请求标志 0：没有事件触发 SOC 请求 1：有事件触发 SOC 请求，采样开始后位段清零
11	SOC11	RO	0x0	通道 11 的 SOC 请求标志 0：没有事件触发 SOC 请求 1：有事件触发 SOC 请求，采样开始后位段清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
10	SOC10	RO	0x0	通道 10 的 SOC 请求标志 0：没有事件触发 SOC 请求 1：有事件触发 SOC 请求，采样开始后位段清零
9	SOC9	RO	0x0	通道 9 的 SOC 请求标志 0：没有事件触发 SOC 请求 1：有事件触发 SOC 请求，采样开始后位段清零
8	SOC8	RO	0x0	通道 8 的 SOC 请求标志 0：没有事件触发 SOC 请求 1：有事件触发 SOC 请求，采样开始后位段清零
7	SOC7	RO	0x0	通道 7 的 SOC 请求标志 0：没有事件触发 SOC 请求 1：有事件触发 SOC 请求，采样开始后位段清零
6	SOC6	RO	0x0	通道 6 的 SOC 请求标志 0：没有事件触发 SOC 请求 1：有事件触发 SOC 请求，采样开始后位段清零
5	SOC5	RO	0x0	通道 5 的 SOC 请求标志 0：没有事件触发 SOC 请求 1：有事件触发 SOC 请求，采样开始后位段清零
4	SOC4	RO	0x0	通道 4 的 SOC 请求标志 0：没有事件触发 SOC 请求 1：有事件触发 SOC 请求，采样开始后位段清零
3	SOC3	RO	0x0	通道 3 的 SOC 请求标志 0：没有事件触发 SOC 请求 1：有事件触发 SOC 请求，采样开始后位段清零
2	SOC2	RO	0x0	通道 2 的 SOC 请求标志 0：没有事件触发 SOC 请求 1：有事件触发 SOC 请求，采样开始后位段清零
1	SOC1	RO	0x0	通道 1 的 SOC 请求标志 0：没有事件触发 SOC 请求 1：有事件触发 SOC 请求，采样开始后位段清零
0	SOC0	RO	0x0	通道 0 的 SOC 请求标志 0：没有事件触发 SOC 请求 1：有事件触发 SOC 请求，采样开始后位段清零

SPINTROL

表 10-18：ADC SOC 请求强制寄存器（ADCSOCFRC）位段定义

ADCSOCFRC (ADC SOC Force Register) Offset：0x1C Default：0x00000000 Access：ADCx->ADCSOCFRC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 SOC15	14 SOC14	13 SOC13	12 SOC12	11 SOC11	10 SOC10	9 SOC9	8 SOC8
7 SOC7	6 SOC6	5 SOC5	4 SOC4	3 SOC3	2 SOC2	1 SOC1	0 SOC0

表 10-19：ADC SOC 请求强制寄存器（ADCSOCFRC）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	SOC15	W1S	0x0	通道 15 软件触发 SOC 请求 0：写 0 无效，读总是 0 1：写 1 触发通道 7 的 SOC，本位段自动清零
14	SOC14	W1S	0x0	通道 14 软件触发 SOC 请求 0：写 0 无效，读总是 0 1：写 1 触发通道 6 的 SOC，本位段自动清零
13	SOC13	W1S	0x0	通道 13 软件触发 SOC 请求 0：写 0 无效，读总是 0 1：写 1 触发通道 5 的 SOC，本位段自动清零
12	SOC12	W1S	0x0	通道 12 软件触发 SOC 请求 0：写 0 无效，读总是 0 1：写 1 触发通道 4 的 SOC，本位段自动清零
11	SOC11	W1S	0x0	通道 11 软件触发 SOC 请求 0：写 0 无效，读总是 0 1：写 1 触发通道 3 的 SOC，本位段自动清零
10	SOC10	W1S	0x0	通道 10 软件触发 SOC 请求 0：写 0 无效，读总是 0 1：写 1 触发通道 2 的 SOC，本位段自动清零
9	SOC9	W1S	0x0	通道 9 软件触发 SOC 请求 0：写 0 无效，读总是 0 1：写 1 触发通道 1 的 SOC，本位段自动清零
8	SOC8	W1S	0x0	通道 8 软件触发 SOC 请求 0：写 0 无效，读总是 0 1：写 1 触发通道 0 的 SOC，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
7	SOC7	W1S	0x0	通道 7 软件触发 SOC 请求 0：写 0 无效，读总是 0 1：写 1 触发通道 7 的 SOC，本位段自动清零
6	SOC6	W1S	0x0	通道 6 软件触发 SOC 请求 0：写 0 无效，读总是 0 1：写 1 触发通道 6 的 SOC，本位段自动清零
5	SOC5	W1S	0x0	通道 5 软件触发 SOC 请求 0：写 0 无效，读总是 0 1：写 1 触发通道 5 的 SOC，本位段自动清零
4	SOC4	W1S	0x0	通道 4 软件触发 SOC 请求 0：写 0 无效，读总是 0 1：写 1 触发通道 4 的 SOC，本位段自动清零
3	SOC3	W1S	0x0	通道 3 软件触发 SOC 请求 0：写 0 无效，读总是 0 1：写 1 触发通道 3 的 SOC，本位段自动清零
2	SOC2	W1S	0x0	通道 2 软件触发 SOC 请求 0：写 0 无效，读总是 0 1：写 1 触发通道 2 的 SOC，本位段自动清零
1	SOC1	W1S	0x0	通道 1 软件触发 SOC 请求 0：写 0 无效，读总是 0 1：写 1 触发通道 1 的 SOC，本位段自动清零
0	SOC0	W1S	0x0	通道 0 软件触发 SOC 请求 0：写 0 无效，读总是 0 1：写 1 触发通道 0 的 SOC，本位段自动清零

表 10-20：ADC SOC 请求溢出标志寄存器（ADCSOCOVF）位段定义

ADCSOCOVF (ADC SOC Overflow Register) Offset：0x20 Default：0x00000000 Access：ADCx->ADCSOCOVF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 SOC15	14 SOC14	13 SOC13	12 SOC12	11 SOC11	10 SOC10	9 SOC9	8 SOC8
7 SOC7	6 SOC6	5 SOC5	4 SOC4	3 SOC3	2 SOC2	1 SOC1	0 SOC0

©2025 旋智电子科技（上海）有限公司

表 10-21：ADC SOC 请求溢出标志寄存器（ADCSOCOVF）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	SOC15	RO	0x0	通道 15 的 SOC 请求溢出标志 0：未发生 SOC 请求溢出 1：ADCSOCFLG 标志为 1 时又有 SOC 事件被触发
14	SOC14	RO	0x0	通道 14 的 SOC 请求溢出标志 0：未发生 SOC 请求溢出 1：ADCSOCFLG 标志为 1 时又有 SOC 事件被触发
13	SOC13	RO	0x0	通道 13 的 SOC 请求溢出标志 0：未发生 SOC 请求溢出 1：ADCSOCFLG 标志为 1 时又有 SOC 事件被触发
12	SOC12	RO	0x0	通道 12 的 SOC 请求溢出标志 0：未发生 SOC 请求溢出 1：ADCSOCFLG 标志为 1 时又有 SOC 事件被触发
11	SOC11	RO	0x0	通道 11 的 SOC 请求溢出标志 0：未发生 SOC 请求溢出 1：ADCSOCFLG 标志为 1 时又有 SOC 事件被触发
10	SOC10	RO	0x0	通道 10 的 SOC 请求溢出标志 0：未发生 SOC 请求溢出 1：ADCSOCFLG 标志为 1 时又有 SOC 事件被触发
9	SOC9	RO	0x0	通道 9 的 SOC 请求溢出标志 0：未发生 SOC 请求溢出 1：ADCSOCFLG 标志为 1 时又有 SOC 事件被触发
8	SOC8	RO	0x0	通道 8 的 SOC 请求溢出标志 0：未发生 SOC 请求溢出 1：ADCSOCFLG 标志为 1 时又有 SOC 事件被触发
7	SOC7	RO	0x0	通道 7 的 SOC 请求溢出标志 0：未发生 SOC 请求溢出 1：ADCSOCFLG 标志为 1 时又有 SOC 事件被触发

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
6	SOC6	RO	0x0	通道 6 的 SOC 请求溢出标志 0：未发生 SOC 请求溢出 1：ADCSOCFLG 标志为 1 时又有 SOC 事件被触发
5	SOC5	RO	0x0	通道 5 的 SOC 请求溢出标志 0：未发生 SOC 请求溢出 1：ADCSOCFLG 标志为 1 时又有 SOC 事件被触发
4	SOC4	RO	0x0	通道 4 的 SOC 请求溢出标志 0：未发生 SOC 请求溢出 1：ADCSOCFLG 标志为 1 时又有 SOC 事件被触发
3	SOC3	RO	0x0	通道 3 的 SOC 请求溢出标志 0：未发生 SOC 请求溢出 1：ADCSOCFLG 标志为 1 时又有 SOC 事件被触发
2	SOC2	RO	0x0	通道 2 的 SOC 请求溢出标志 0：未发生 SOC 请求溢出 1：ADCSOCFLG 标志为 1 时又有 SOC 事件被触发
1	SOC1	RO	0x0	通道 1 的 SOC 请求溢出标志 0：未发生 SOC 请求溢出 1：ADCSOCFLG 标志为 1 时又有 SOC 事件被触发
0	SOC0	RO	0x0	通道 0 的 SOC 请求溢出标志 0：未发生 SOC 请求溢出 1：ADCSOCFLG 标志为 1 时又有 SOC 事件被触发

SPINTROL

表 10-22：ADC SOC 请求溢出标志寄存器（ADCSOCOVFC）位段定义

ADCSOCOVFC (ADC SOC Overflow Clear Register) Offset：0x24 Default：0x00000000 Access：ADCx->ADCSOCOVFC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 SOC15	14 SOC14	13 SOC13	12 SOC12	11 SOC11	10 SOC10	9 SOC9	8 SOC8
7 SOC7	6 SOC6	5 SOC5	4 SOC4	3 SOC3	2 SOC2	1 SOC1	0 SOC0

表 10-23：ADC SOC 请求溢出标志寄存器（ADCSOCOVFC）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	SOC15	W1C	0x0	清除通道 15 的 SOC 请求溢出标志 0：写 0 无效，读总是 0 1：写 1 清除请求溢出标志，本位段自动清零
14	SOC14	W1C	0x0	清除通道 14 的 SOC 请求溢出标志 0：写 0 无效，读总是 0 1：写 1 清除请求溢出标志，本位段自动清零
13	SOC13	W1C	0x0	清除通道 13 的 SOC 请求溢出标志 0：写 0 无效，读总是 0 1：写 1 清除请求溢出标志，本位段自动清零
12	SOC12	W1C	0x0	清除通道 12 的 SOC 请求溢出标志 0：写 0 无效，读总是 0 1：写 1 清除请求溢出标志，本位段自动清零
11	SOC11	W1C	0x0	清除通道 11 的 SOC 请求溢出标志 0：写 0 无效，读总是 0 1：写 1 清除请求溢出标志，本位段自动清零
10	SOC10	W1C	0x0	清除通道 10 的 SOC 请求溢出标志 0：写 0 无效，读总是 0 1：写 1 清除请求溢出标志，本位段自动清零
9	SOC9	W1C	0x0	清除通道 9 的 SOC 请求溢出标志 0：写 0 无效，读总是 0 1：写 1 清除请求溢出标志，本位段自动清零
8	SOC8	W1C	0x0	清除通道 8 的 SOC 请求溢出标志 0：写 0 无效，读总是 0 1：写 1 清除请求溢出标志，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
7	SOC7	W1C	0x0	清除通道 7 的 SOC 请求溢出标志 0：写 0 无效，读总是 0 1：写 1 清除请求溢出标志，本位段自动清零
6	SOC6	W1C	0x0	清除通道 6 的 SOC 请求溢出标志 0：写 0 无效，读总是 0 1：写 1 清除请求溢出标志，本位段自动清零
5	SOC5	W1C	0x0	清除通道 5 的 SOC 请求溢出标志 0：写 0 无效，读总是 0 1：写 1 清除请求溢出标志，本位段自动清零
4	SOC4	W1C	0x0	清除通道 4 的 SOC 请求溢出标志 0：写 0 无效，读总是 0 1：写 1 清除请求溢出标志，本位段自动清零
3	SOC3	W1C	0x0	清除通道 3 的 SOC 请求溢出标志 0：写 0 无效，读总是 0 1：写 1 清除请求溢出标志，本位段自动清零
2	SOC2	W1C	0x0	清除通道 2 的 SOC 请求溢出标志 0：写 0 无效，读总是 0 1：写 1 清除请求溢出标志，本位段自动清零
1	SOC1	W1C	0x0	清除通道 1 的 SOC 请求溢出标志 0：写 0 无效，读总是 0 1：写 1 清除请求溢出标志，本位段自动清零
0	SOC0	W1C	0x0	清除通道 0 的 SOC 请求溢出标志 0：写 0 无效，读总是 0 1：写 1 清除请求溢出标志，本位段自动清零

表 10-24：ADC EOC 中断事件触发 SOC 使能寄存器（ADCINTSOCEN）位段定义

ADCINTSOCEN (ADC Interrupt Trigger SOC Enable Register) Offset：0x28 Default：0x00000000 Access：ADCx->ADCINTSOCEN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH15	14 CH14	13 CH13	12 CH12	11 CH11	10 CH10	9 CH9	8 CH8
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

©2025 旋智电子科技（上海）有限公司

表 10-25：ADC EOC 中断事件触发 SOC 使能寄存器（ADCINTSOCEN）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	CH15	RW	0x0	通道 15 的 EOC 中断事件触发 SOC 使能[1] 0：根据 ADCCHCTL[7].TRIGSEL 选择 SOC 触发源 1：根据 ADCINTSOCSEL[0].CH15 选择的 EOC 触发 SOC
14	CH14	RW	0x0	通道 14 的 EOC 中断事件触发 SOC 使能[1] 0：根据 ADCCHCTL[6].TRIGSEL 选择 SOC 触发源 1：根据 ADCINTSOCSEL[0].CH14 选择的 EOC 触发 SOC
13	CH13	RW	0x0	通道 13 的 EOC 中断事件触发 SOC 使能[1] 0：根据 ADCCHCTL[5].TRIGSEL 选择 SOC 触发源 1：根据 ADCINTSOCSEL[0].CH13 选择的 EOC 触发 SOC
12	CH12	RW	0x0	通道 12 的 EOC 中断事件触发 SOC 使能[1] 0：根据 ADCCHCTL[4].TRIGSEL 选择 SOC 触发源 1：根据 ADCINTSOCSEL[0].CH12 选择的 EOC 触发 SOC
11	CH11	RW	0x0	通道 11 的 EOC 中断事件触发 SOC 使能[1] 0：根据 ADCCHCTL[3].TRIGSEL 选择 SOC 触发源 1：根据 ADCINTSOCSEL[0].CH11 选择的 EOC 触发 SOC
10	CH10	RW	0x0	通道 10 的 EOC 中断事件触发 SOC 使能[1] 0：根据 ADCCHCTL[2].TRIGSEL 选择 SOC 触发源 1：根据 ADCINTSOCSEL[0].CH10 选择的 EOC 触发 SOC
9	CH9	RW	0x0	通道 9 的 EOC 中断事件触发 SOC 使能[1] 0：根据 ADCCHCTL[1].TRIGSEL 选择 SOC 触发源 1：根据 ADCINTSOCSEL[0].CH9 选择的 EOC 触发 SOC
8	CH8	RW	0x0	通道 8 的 EOC 中断事件触发 SOC 使能[1] 0：根据 ADCCHCTL[0].TRIGSEL 选择 SOC 触发源 1：根据 ADCINTSOCSEL[0].CH8 选择的 EOC 触发 SOC
7	CH7	RW	0x0	通道 7 的 EOC 中断事件触发 SOC 使能[1] 0：根据 ADCCHCTL[7].TRIGSEL 选择 SOC 触发源 1：根据 ADCINTSOCSEL[0].CH7 选择的 EOC 触发 SOC

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
6	CH6	RW	0x0	通道 6 的 EOC 中断事件触发 SOC 使能[1] 0：根据 ADCCHCTL[6].TRIGSEL 选择 SOC 触发源 1：根据 ADCINTSOCSEL[0].CH6 选择的 EOC 触发 SOC
5	CH5	RW	0x0	通道 5 的 EOC 中断事件触发 SOC 使能[1] 0：根据 ADCCHCTL[5].TRIGSEL 选择 SOC 触发源 1：根据 ADCINTSOCSEL[0].CH5 选择的 EOC 触发 SOC
4	CH4	RW	0x0	通道 4 的 EOC 中断事件触发 SOC 使能[1] 0：根据 ADCCHCTL[4].TRIGSEL 选择 SOC 触发源 1：根据 ADCINTSOCSEL[0].CH4 选择的 EOC 触发 SOC
3	CH3	RW	0x0	通道 3 的 EOC 中断事件触发 SOC 使能[1] 0：根据 ADCCHCTL[3].TRIGSEL 选择 SOC 触发源 1：根据 ADCINTSOCSEL[0].CH3 选择的 EOC 触发 SOC
2	CH2	RW	0x0	通道 2 的 EOC 中断事件触发 SOC 使能[1] 0：根据 ADCCHCTL[2].TRIGSEL 选择 SOC 触发源 1：根据 ADCINTSOCSEL[0].CH2 选择的 EOC 触发 SOC
1	CH1	RW	0x0	通道 1 的 EOC 中断事件触发 SOC 使能[1] 0：根据 ADCCHCTL[1].TRIGSEL 选择 SOC 触发源 1：根据 ADCINTSOCSEL[0].CH1 选择的 EOC 触发 SOC
0	CH0	RW	0x0	通道 0 的 EOC 中断事件触发 SOC 使能[1] 0：根据 ADCCHCTL[0].TRIGSEL 选择 SOC 触发源 1：根据 ADCINTSOCSEL[0].CH0 选择的 EOC 触发 SOC

表 10-26：ADC EOC 中断事件触发 SOC 选择寄存器（ADCINTSOCSEL）位段定义

ADCINTSOCSEL (ADC Interrupt Trigger SOC Select Register 0)　Offset：0x2C　Default：0x00000000
Access：ADCx->ADCINTSOCSEL

31	30	29	28	27	26	25	24
CH7				CH6			
23	22	21	20	19	18	17	16
CH5				CH4			
15	14	13	12	11	10	9	8
CH3				CH2			
7	6	5	4	3	2	1	0

©2025 旋智电子科技（上海）有限公司

CH1	CH0

表 10-27：ADC EOC 中断事件触发 SOC 选择寄存器（ADCINTSOCSEL）位段描述

位段	位段名	属性	复位值	描述
31:28	CH7	RW	0x0	触发通道 7 SOC 的 EOC 所属通道号
27:24	CH6	RW	0x0	触发通道 6 SOC 的 EOC 所属通道号
23:20	CH5	RW	0x0	触发通道 5 SOC 的 EOC 所属通道号
19:16	CH4	RW	0x0	触发通道 4 SOC 的 EOC 所属通道号
15:12	CH3	RW	0x0	触发通道 3 SOC 的 EOC 所属通道号
11:8	CH2	RW	0x0	触发通道 2 SOC 的 EOC 所属通道号
7:4	CH1	RW	0x0	触发通道 1 SOC 的 EOC 所属通道号
3:0	CH0	RW	0x0	触发通道 0 SOC 的 EOC 所属通道号

表 10-28：ADC 外部 SOC 输入控制寄存器（ADCEXTSOCCTL）位段定义

ADCEXTSOCCTL (ADC External SOC Input Control Register) Offset：0x34 Default：0x00000001 Access：ADCx->ADCEXTSOCCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	IOSEL						POL

表 10-29：ADC 外部 SOC 输入控制寄存器（ADCEXTSOCCTL）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6:1	IOSEL	RW	0x0	用于触发 SOC 的外部引脚号
0	POL	RW	0x1	触发 SOC 的引脚信号极性 0：引脚下降沿触发 1：引脚上升沿触发

©2025 旋智电子科技（上海）有限公司

表 10-30：ADC 通道 n ([0:15]) 控制寄存器（ADCCHCTL）位段定义

ADCCHCTL (ADC Channel n ([0:15]) Control Register) Offset：0x38 Default：0x00000000 Access：ADCx->ADCCHCTL							
31	30	29	28	27	26	25	24
RESERVED		SHSEL	TRIGSEL				SHINSELN
23	22	21	20	19	18	17	16
SHINSELN			SHINSELP				AVGCNT
15	14	13	12	11	10	9	8
AVGCNT		FRACBIT	CONVCNT				
7	6	5	4	3	2	1	0
SAMPCNT							

表 10-31：ADC 通道 n ([0:15]) 控制寄存器（ADCCHCTL）位段描述

位段	位段名	属性	复位值	描述
31:30	RESERVED_31_30	RO	0x0	保留
29	SHSEL	RW	0x0	采样保持器控制 0：关闭 1：使能
28:25	TRIGSEL	RW	0x0	SOC 触发源，可以被 ADCINTSOCSEL 覆盖 0000：ADCTRIG0 - Software only 0001：ADCTRIG1 - Timer 0 0010：ADCTRIG2 - Timer 1 0011：ADCTRIG3 - Timer 2 0100：ADCTRIG4 - EXTSOC 0101：ADCTRIG5 - PWM0SOCA 0110：ADCTRIG6 - PWM0SOCB 0111：ADCTRIG7 - PWM0SOCC 1000：ADCTRIG8 - PWM1SOCA 1001：ADCTRIG9 - PWM1SOCB 1010：ADCTRIG10 - PWM1SOCC 1011：ADCTRIG11 - PWM2SOCA 1100：ADCTRIG12 - PWM2SOCB 1101：ADCTRIG13 - PWM2SOCC 1110：ADCTRIG14 - PWM3SOCA 1111：ADCTRIG15 - PWM3SOCB
24:21	SHINSELN	RW	0x0	采样保持的负端输入源选择 0000：Analog ground 0001：T-sensor negative output 0010：Negative output of DPGA 0011：ANA_IN1 pin input 0100：ANA_IN3 pin input 0101：ANA_IN5 pin input

SPINTROL

位段	位段名	属性	复位值	描述
				0110：ANA_IN6 pin input 0111：ANA_IN7 pin input 1000：ANA_IN8 pin input 1001：DAC1 output 1010：DAC2 output 1011：ANA_IN12 pin input / 3-phase common-node sensed signal
20:17	SHINSELP	RW	0x0	采样保持的正端输入源选择 0000：Analog ground 0001：T-sensor positive output 0010：Positive output of DPGA 0011：ANA_IN0 pin input 0100：ANA_IN2 pin input 0101：ANA_IN4 pin input 0110：SPGA output 0111：ATEST output 1000：DAC buffer output 1001：DAC0 output 1010：Digital 1.2V power 1011：Analog 3.3V power 1100：ANA_IN9 pin input / U-Phase sensed signal 1101：ANA_IN10 pin input / V-Phase sensed signal 1110：ANA_IN11 pin input / W-Phase sensed signal 1111：ANA_IN13 pin input / VBATMDIV sensed signal
16:14	AVGCNT	RW	0x0	多次采样转换求平均的次数 000：单次采样转换 001：每个 SOC 触发 2 次连续的采样转换并平均 010：每个 SOC 触发 4 次连续的采样转换并平均 011：每个 SOC 触发 8 次连续的采样转换并平均 100：每个 SOC 触发 16 次连续的采样转换并平均 101：每个 SOC 触发 32 次连续的采样转换并平均 110：每个 SOC 触发 64 次连续的采样转换并平均 111：每个 SOC 触发 128 次连续的采样转换并平均
13	FRACBIT	RW	0x0	保持低 7 位为多次采样转换求平均的小数部分 0：ADC 结果是一个 13 位的整数 1：ADC 结果是一个 20 位的数，低 7 位为小数
12:8	CONVCNT	RW	0x0	设置转换时间是(CONVCNT + 1)个 ADC 模块时钟
7:0	SAMPCNT	RW	0x0	设置采样时间是(SAMPCNT + 1)个 ADC 模块时钟

©2025 旋智电子科技（上海）有限公司

表 10-32：ADC 通道 n ([0:7]) 偏移校准寄存器（ADCOFFSET）位段定义

ADCOFFSET (ADC Offset Trim Register n ([0:15])) Offset：0x78 Default：0x00000000 Access：ADCx->ADCOFFSET							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 10-33：ADC 通道 n ([0:7]) 偏移校准寄存器（ADCOFFSET）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	通道偏移校准值， 为一个有符号数，低 7 位为小数部分

表 10-34：ADC 通道 n ([0:7]) 增益校准寄存器（ADCGAIN）位段定义

ADCGAIN (ADC Gain Trim Register n ([0:15])) Offset：0xB8 Default：0x00008000 Access：ADCx->ADCGAIN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 10-35：ADC 通道 n ([0:7]) 增益校准寄存器（ADCGAIN）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x8000	设置通道增益校准系数为 VAL/32768 （即低 15 位为小数部分）

©2025 旋智电子科技（上海）有限公司

表 10-36：ADC 采样转换核心电路偏移校准寄存器（SHOFFSET）位段定义

SHOFFSET (ADC SH0 Offset Trim Register) Offset：0xF8 Default：0x00000000 Access：ADCx->SHOFFSET							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 10-37：ADC 采样转换核心电路偏移校准寄存器（SHOFFSET）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	采样转换核心电路偏移校准值， 为一个有符号数，低 7 位为小数部分

表 10-38：ADC 采样转换核心电路增益校准寄存器（SHGAIN）位段定义

SHGAIN (ADC SH0 Gain Trim Register) Offset：0x104 Default：0x00008000 Access：ADCx->SHGAIN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 10-39：ADC 采样转换核心电路增益校准寄存器（SHGAIN）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x8000	设置采样保持核心电路增益校准系数为 VAL/32768（即低 15 位为小数部分）

©2025 旋智电子科技（上海）有限公司

表 10-40：ADC 采样转换 EOC 标志寄存器（SHFLG）位段定义

SHFLG (ADC Sample Hold Channel EOC Flag Register) Offset：0x110 Default：0x00000000 Access：ADCx->SHFLG							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							SHRDY0

表 10-41：ADC 采样转换 EOC 标志寄存器（SHFLG）位段描述

位段	位段名	属性	复位值	描述
31:1	RESERVED_31_1	RO	0x0	保留
0	SHRDY0	RO	0x0	SHRAWCODE 数据就绪标志 0：读 0 表示 SHRAWCODE 数据无效 写 0 无效 1：读 1 表示 SHRAWCODE 数据有效 写 1 清除本标志

表 10-42：ADC 采样转换 EOC 标志寄存器（SHFLGCLR）位段定义

SHFLGCLR (ADC Sample Hold Channel EOC Flag Clear Register) Offset：0x114 Default：0x00000000 Access：ADCx->SHFLGCLR							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							SHRDY0

表 10-43：ADC 采样转换 EOC 标志寄存器（SHFLGCLR）位段描述

位段	位段名	属性	复位值	描述
31:1	RESERVED_31_1	RO	0x0	保留
0	SHRDY0	W1C	0x0	清除 SHRAWCODE0 数据就绪标志 0：写 0 无效，读永远为 0 1：写 1 清除 SHFLG[SHRDY0]标志，本位段自动清零

©2025 旋智电子科技（上海）有限公司

表 10-44：ADC 控制寄存器（ADCCTL）位段定义

ADCCTL (ADC Control Register) Offset：0x118 Default：0x00000000 Access：ADCx->ADCCTL							
31	30	29	28	27	26	25	24
RESERVED							
23 DMAEN7	22 DMAEN6	21 DMAEN5	20 DMAEN4	19 DMAEN3	18 DMAEN2	17 DMAEN1	16 DMAEN0
15	14	13	12	11	10	9	8
RESERVED							
7 SHFRCN0	6	5 SHFRCP0	4	3 RST	2 EN	1 REFEN	0 RESERVED

表 10-45：ADC 控制寄存器（ADCCTL）位段描述

位段	位段名	属性	复位值	描述
31:24	RESERVED_31_24	RO	0x0	保留
23	DMAEN7	RW	0x0	通道 7 结果 DMA 使能 0：关闭 1：使能
22	DMAEN6	RW	0x0	通道 6 结果 DMA 使能 0：关闭 1：使能
21	DMAEN5	RW	0x0	通道 5 结果 DMA 使能 0：关闭 1：使能
20	DMAEN4	RW	0x0	通道 4 结果 DMA 使能 0：关闭 1：使能
19	DMAEN3	RW	0x0	通道 3 结果 DMA 使能 0：关闭 1：使能
18	DMAEN2	RW	0x0	通道 2 结果 DMA 使能 0：关闭 1：使能
17	DMAEN1	RW	0x0	通道 1 结果 DMA 使能 0：关闭 1：使能
16	DMAEN0	RW	0x0	通道 0 结果 DMA 使能 0：关闭 1：使能

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
15:8	RESERVED_15_8	RO	0x0	保留
7:6	SHFRCN0	RW	0x0	采样保持器负输入端强制上下拉 0：正常连接（无上下拉） 2：强制下拉到地 3：强制上拉到电源
5:4	SHFRCP0	RW	0x0	采样保持器正输入端强制上下拉 0：正常连接（无上下拉） 2：强制下拉到地 3：强制上拉到电源
3	RST	W1C	0x0	ADC 数字控制复位 0：写 0 无效，读总是 0 1：写 1 复位数字控制，本位段自动清零
2	EN	RW	0x0	ADC 使能 0：关闭 1：使能
1	REFEN	RW	0x0	参考源使能 0：关闭 1：使能
0	RESERVED_0	RO	0x0	保留

表 10-46：ADC 采样转换原始码值（SHRAWCODE）位段定义

SHRAWCODE (ADC SH0 Raw Code Register) Offset：0x11C Default：0x00000000 Access：ADCx->SHRAWCODE							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 10-47：ADC 采样转换原始码值（SHRAWCODE）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	原始码值 为一个有符号数，低 7 位是小数部分

SPINTROL

表 10-48：ADC 通道 n ([0:15])结果寄存器（ADCRESULT）位段定义

ADCRESULT (ADC Result Register n ([0:15])) Offset：0x128 Default：0x00000000 Access：ADCx->ADCRESULT							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 10-49：ADC 通道 n ([0:15])结果寄存器（ADCRESULT）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	ADC 结果(是-4096 ~ 4095 的有符号数)

表 10-50：后处理单元 n ([0:2])比较结果寄存器（PPURESULT）位段定义

PPURESULT (PPU Comparison Result Register n ([0:2])) Offset：0x168 Default：0x00000000 Access：ADCx->PPURESULT							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 10-51：后处理单元 n ([0:2])比较结果寄存器（PPURESULT）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	符号位扩展后的 32 位有符号数

©2025 旋智电子科技（上海）有限公司

表 10-52：后处理单元 n ([0:2]) SOC 响应延时寄存器（PPUSOCDLY）位段定义

PPUSOCDLY (PPU SOC Delay Register n ([0:2])) Offset：0x188 Default：0x00000000 Access：ADCx->PPUSOCDLY							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 10-53：后处理单元 n ([0:2]) SOC 响应延时寄存器（PPUSOCDLY）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	PPUCTL[n].SOCSEL 所指定的 SOC 从触发到实际响应的延时（所对应的 ADC 模块时钟周期数）

表 10-54：后处理单元 n ([0:2]) 中断原始事件标志寄存器（PPURAWIF）位段定义

PPURAWIF (PPU Interrupt Raw Flag Register n ([0:2])) Offset：0x1A8 Default：0x00000000 Access：ADCx->PPURAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED					XZRO	TZOV	TZUV

表 10-55：后处理单元 n ([0:2]) 中断原始事件标志寄存器（PPURAWIF）位段描述

位段	位段名	属性	复位值	描述
31:3	RESERVED_31_3	RO	0x0	保留
2	XZRO	RO	0x0	后处理单元比较结果过零事件标志 0：事件未发生 1：事件已发生
1	TZOV	RO	0x0	后处理单元比较结果过压事件标志 0：事件未发生 1：事件已发生

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
0	TZUV	RO	0x0	后处理单元比较结果欠压事件标志 0：事件未发生 1：事件已发生

表 10-56：后处理单元 n ([0:2]) 中断标志寄存器（PPUIF）位段定义

PPUIF (PPU Interrupt Flag Register n ([0:2])) Offset：0x1C8 Default：0x00000000 Access：ADCx->PPUIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				INT	XZRO	TZOV	TZUV

表 10-57：后处理单元 n ([0:2]) 中断标志寄存器（PPUIF）位段描述

位段	位段名	属性	复位值	描述
31:4	RESERVED_31_4	RO	0x0	保留
3	INT	RO	0x0	后处理单元全局中断标志 0：中断未发生 1：中断已产生，并已经通知 CPU 本位段被清除之前不会产生新中断
2	XZRO	RO	0x0	后处理单元比较结果过零中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
1	TZOV	RO	0x0	后处理单元比较结果过压中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
0	TZUV	RO	0x0	后处理单元比较结果欠压中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断

©2025 旋智电子科技（上海）有限公司

表 10-58：后处理单元 n ([0:2]) 中断清除寄存器（PPUIC）位段定义

PPUIC (PPU Interrupt Clear Register n ([0:2])) Offset：0x1E8 Default：0x00000000 Access：ADCx->PPUIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				INT	XZRO	TZOV	TZUV

表 10-59：后处理单元 n ([0:2]) 中断清除寄存器（PPUIC）位段描述

位段	位段名	属性	复位值	描述
31:4	RESERVED_31_4	RO	0x0	保留
3	INT	W1C	0x0	后处理单元全局中断清除 0：写 0 无效，读总是 0 1：写 1 清除中断标志，本位段自动清零
2	XZRO	W1C	0x0	后处理单元比较结果过零中断清除 0：写 0 无效，读总是 0 1：写 1 清除中断标志，本位段自动清零
1	TZOV	W1C	0x0	后处理单元比较结果过压中断清除 0：写 0 无效，读总是 0 1：写 1 清除中断标志，本位段自动清零
0	TZUV	W1C	0x0	后处理单元比较结果欠压中断清除 0：写 0 无效，读总是 0 1：写 1 清除中断标志，本位段自动清零

表 10-60：后处理单元 n ([0:2]) 中断使能寄存器（PPUIE）位段定义

PPUIE (PPU Interrupt Enable Register n ([0:2])) Offset：0x208 Default：0x00000000 Access：ADCx->PPUIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED					XZRO	TZOV	TZUV

SPINTROL

表 10-61：后处理单元 n ([0:2]) 中断使能寄存器（PPUIE）位段描述

位段	位段名	属性	复位值	描述
31:3	RESERVED_31_3	RO	0x0	保留
2	XZRO	RW	0x0	后处理单元比较结果过零中断使能 0：关闭 1：使能
1	TZOV	RW	0x0	后处理单元比较结果过压中断使能 0：关闭 1：使能
0	TZUV	RW	0x0	后处理单元比较结果欠压中断使能 0：关闭 1：使能

表 10-62：后处理单元 n ([0:2]) 封锁事件使能寄存器（PPUTZE）位段定义

PPUTZE (PPU Trip-Zone Event Enable Register n ([0:2]))　Offset：0x228　Default：0x00000000
Access：ADCx->PPUTZE

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED					XZRO	TZOV	TZUV

表 10-63：后处理单元 n ([0:2]) 封锁事件使能寄存器（PPUTZE）位段描述

位段	位段名	属性	复位值	描述
31:3	RESERVED_31_3	RO	0x0	保留
2	XZRO	RW	0x0	使能后处理单元比较结果过零触发 PWM 封锁 0：关闭 1：使能
1	TZOV	RW	0x0	使能后处理单元比较结果过压触发 PWM 封锁 0：关闭 1：使能
0	TZUV	RW	0x0	使能后处理单元比较结果欠压触发 PWM 封锁 0：关闭 1：使能

©2025 旋智电子科技（上海）有限公司

表 10-64：后处理单元 n ([0:2]) 控制寄存器（PPUCTL）位段定义

PPUCTL (PPU Control Register n ([0:2])) Offset：0x248 Default：0x00000400 Access：ADCx->PPUCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED					POL	DATASEL	
7	6	5	4	3	2	1	0
DATASEL		SOCSEL				CBCEN	EN

表 10-65：后处理单元 n ([0:2]) 控制寄存器（PPUCTL）位段描述

位段	位段名	属性	复位值	描述
31:11	RESERVED_31_11	RO	0x0	保留
10	POL	RW	0x1	比较的极性 0：参考值减去所选通道的转换结果 1：所选通道的转换结果减去参考值
9:6	DATASEL	RW	0x0	选择参与比较的转换结果所属的通道 取值范围应当在[0, 7]内
5:2	SOCSEL	RW	0x0	选择待测响应延时的 SOC 所属的通道 取值范围应当在[0, 7]内
1	CBCEN	RW	0x0	后处理单元中断事件标志位自清除使能 0：关闭（只能通过 PPUIC[2:0]位段写 1 清除） 1：使能（根据新的比较结果更新 PPURAWIF[2:0]和 PPUIF[2:0]）
0	EN	RW	0x0	后处理单元使能 0：关闭 1：使能

表 10-66：后处理单元 n ([0:2]) 比较参考寄存器（PPUREF）位段定义

PPUREF (PPU Reference Register n ([0:2])) Offset：0x268 Default：0x00000000 Access：ADCx->PPUREF							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0

©2025 旋智电子科技（上海）有限公司

VAL

表 10-67：后处理单元 n ([0:2]) 比较参考寄存器（PPUREF）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	比较的参考值，为[-4096, 4095]范围内的有符号数

表 10-68：后处理单元 n ([0:2]) 下限阈值寄存器（PPUTHL）位段定义

PPUTHL (PPU Trip-Zone Low-Side Threshold Register n ([0:2])) Offset：0x288 Default：0x00000000 Access：ADCx->PPUTHL							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 10-69：后处理单元 n ([0:2]) 下限阈值寄存器（PPUTHL）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	判定欠压的下限阈值，为[-8192, 8191]范围内的有符号数

表 10-70：后处理单元 n ([0:2]) 上限阈值寄存器（PPUTHH）位段定义

PPUTHH (PPU Trip-Zone High-Side Threshold Register n ([0:2])) Offset：0x2A8 Default：0x00000000 Access：ADCx->PPUTHH							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 10-71：后处理单元 n ([0:2]) 上限阈值寄存器（PPUTHH）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	判定过压的上限阈值，为[-8192, 8191]范围内的有符号数

©2025 旋智电子科技（上海）有限公司

SPINTROL

表 10-72：温度传感器控制寄存器（TSENSCTL）位段定义

TSENSCTL (Temperature Sensor Control Register) Offset：0x2C8 Default：0x00000000 Access：ADCx->TSENSCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	DEMSEL				SWAPBJT	OUTINV	EN

表 10-73：温度传感器控制寄存器（TSENSCTL）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6:3	DEMSEL	RW	0x0	偏置电流分支选择，以实现动态元素匹配
2	SWAPBJT	RW	0x0	交换温度传感器三极管 0：不交换三极管 1：交换三极管
1	OUTINV	RW	0x0	交换温度传感器输出正负端 0：正负端原样输出 1：正负端交换输出
0	EN	RW	0x0	温度传感器使能 0：关闭 1：使能

表 10-74：ADC 模块写使能寄存器（ADCREGKEY）位段定义

ADCREGKEY (ADC Register Write-Allow Key Register) Offset：0x2CC Default：0x1ACCE551 Access：ADCx->ADCREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

表 10-75：ADC 模块写使能寄存器（ADCREGKEY）位段描述

位段	位段名	属性	复位值	描述
31:0	KEY	RW	0x1ACCE551	写入 0x1ACCE551 解锁受保护的 ADC 寄存器

©2025 旋智电子科技（上海）有限公司

11 13 位检测模数转换器（ADC）

11.1 概述

SPC1169 提供一个 13 位的监控 ADC 模块，用于监测 TMCU/VDD33/VDD12/VAREF13，以确保安全性。

13 位检测 ADC 模块的功能包括：

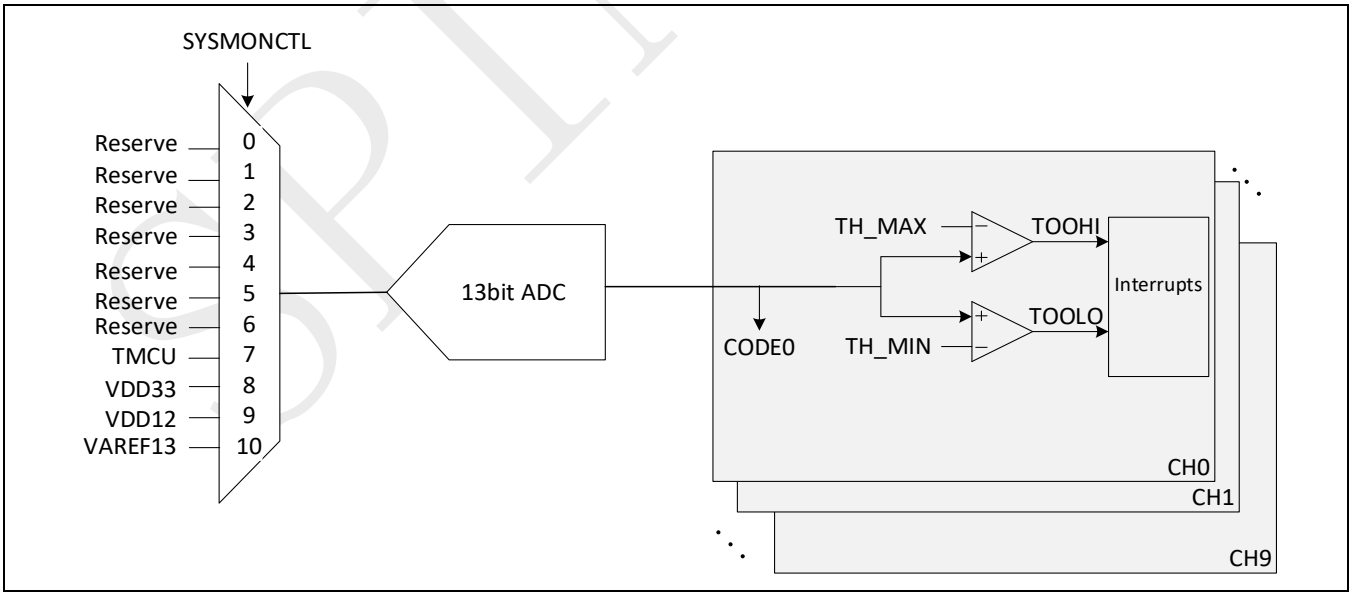
- 带有内置采样保持（S/H）的 13 位 ADC 核心
  - 1 微秒的转换时间
  - 最多支持 11 个通道的多路复用输入
- 单端输入，输入范围为 0V 至 3.657V
- 每个通道具有独立的使能和阈值控制
- 如果任何通道超出阈值窗口，则触发非屏蔽中断（NMI）

11.2 架构及操作

如图 11-1 所示，13 位检测 ADC 电路包括：

- 一个采样器和一个核心
- 13 位检测 ADC 的参考电压
- 超出阈值窗口的检测电路

图 11-1：检测 ADC 框图



13 位检测 ADC 用于检测 TMCU，VDD33 等，如表 11-1 所示。检测结果将保存在相应的寄存器中（例如 VDD33CODE…）。

©2025 旋智电子科技（上海）有限公司

可以单独使能检测项，例如，如果 SYSMONCTL. VDD33EN=1，则将监测 VDD33 电压，并更新 VDD33CODE 寄存器。

如果结果代码超出阈值范围，则相应的错误事件标志将被锁存到 NMRAWIF 寄存器中。若 NMIE 寄存器位使能，将触发非屏蔽中断，NMIF 的相应位将被置为。例如，如果 VDD33CODE 超过 VDD33TH.MAX，NMRAWIF.VDD33ERR 将被置位。如果 NMIE. VDD33ERR=1，则同时 NMIF. VDD33ERR 被置位。

注意： 若使用 13 位检测模数转换器检测 MCU 的温度，其步骤为：

1. 使能 13 位模数转换器的 Bandgap，SDK 驱动库中已提供对应函数接口；
2. 使能 MCU 端的 TSensor；

用户可以通过更改 SYSMONCTL.MONPRD 来设置系统监视间隔，其周期为（MONPRD*8+1）个系统时钟周期，建议将间隔设置为大于 10 微秒以便电压稳定。

详细的监测项目列在表 11-1 中。

表 11-1：检测 ADC 检测的对象

Step	Item	Physical Item	Note
0	Reserve	Reserve	Reserve
1	Reserve	Reserve	Reserve
2	Reserve	Reserve	Reserve
3	Reserve	Reserve	Reserve
4	Reserve	Reserve	Reserve
5	Reserve	Reserve	Reserve
6	Reserve	Reserve	Reserve
7	TMCU	TSEN1&TSEN0	MCU temperature
8	VDD33	VDD33	VDD33 voltage
9	VDD12	VDD12	VDD12 voltage
10	ADCREF13	ADCREF	13-bit ADC reference

测量得到的电压可以通过以下公式计算得出：

$$Vmeas = CODE * 3.657 / 4096 \approx CODE / 1120 \ (V)$$

举例来说，如果 VDD33CODE 为 3696，那么对应的测量电压可以计算为 VDD33 = 3696/1120 = 3.3V。因此，VDD33 为 3.3V。

温度的测量值可以通过以下公式计算：

$$TMCU = (TMCUOFFSET + TMCUGAIN \times TMCUCODE) / 65536 \ (°C)$$

其中这些参数存储在 Flash 中，如表 11-2 所示。

©2025 旋智电子科技（上海）有限公司

表 11-2：计算温度所需的参数

参数	Address	默认值(有符号整数)
TMCUOFFSET	0x1100 1178	0xFEE53EFE (-18530562)
TMCUGAIN	0x1100 117C	0x0003CA01 (248321)

11.3 寄存器

请参考章节 25.10 的寄存器列表及其定义。

SPINTROL

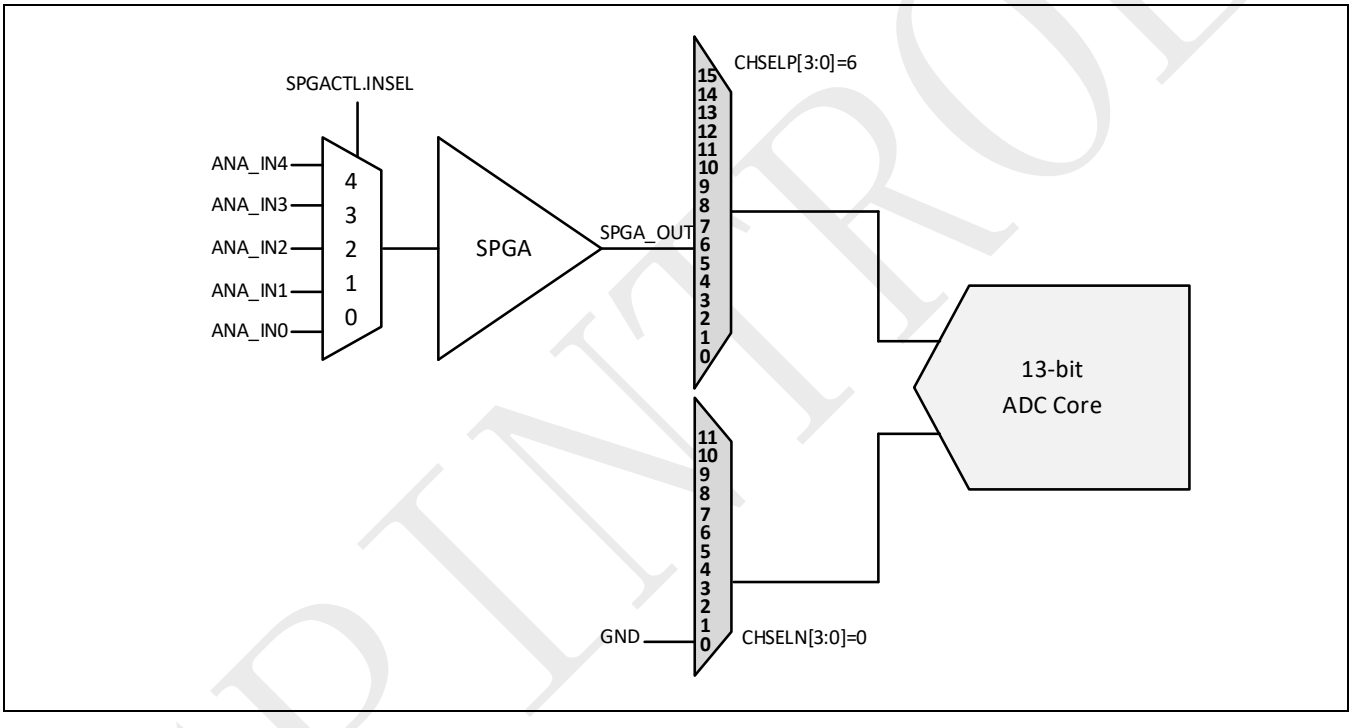
12 单端可编程增益放大器（SPGA）

12.1 概述

SPC1169 实现了一个单端可编程增益放大器（PGA），可以选择 ANA_IN0 到 ANA_IN4 作为输入，并支持最高达 64 倍增益。

12.2 架构

图 12-1：SPGA 框图



12.3 输入选择

SPGA 的输入选择如表 12-1 所示。

表 12-1：SPGA 输入选择

SPGACTL.INSEL	Input
0	ANA_IN0
1	ANA_IN1
2	ANA_IN2
3	ANA_IN3
4	ANA_IN4

©2025 旋智电子科技（上海）有限公司

12.4 增益选择

SPGA 的增益选择如所表 12-2 示：

表 12-2：SPGA 增益选择

SPGACTL.GAIN	Gain
0	1
1	2
2	4
3	8
4	16
5	32
6	48
7	64

SPGA 的输出可以通过以下公式计算：

$$OUT = GAIN \times IN$$

为了避免 SPGA 处于饱和区域，SPGA 的输出范围需要在 0.3V 到 VDD3-0.3V 之间，并且输入范围在 0V 到 VDD3-1.3V 之间。下表显示了不同增益设置下的输入和输出范围（其中 VDD3 通常为 3.3V）。

表 12-3：PGA 单端模式不同增益的输入/输出电压范围

SPGACTL.GAIN	GAIN	Input range (V)			Output range (V)
0	1	0.3	~	2	0.3~2
1	2	0.15	~	1.5	0.3~3
2	4	0.075	~	0.75	0.3~3
3	8	0.0375	~	0.375	0.3~3
4	16	0.01875	~	0.1875	0.3~3
5	32	0.009375	~	0.09375	0.3~3
6	48	0.00625	~	0.0625	0.3~3
7	64	0.0046875	~	0.046875	0.3~3

12.5 寄存器

请参考章节 13.5 的寄存器列表及其定义。

©2025 旋智电子科技（上海）有限公司

13 差分可编程增益放大器（DPGA）

13.1 概述

SPC1169 提供一种带有两个专用引脚的差分可编程增益放大器（DPGA）。DPGA 是一种电流传感放大器，用于通过 13 位 ADC 测量接近地的差分电压，它的增益通过内部控制寄存器进行数字编程，其特点如下：

- 可编程增益：2、4、8、16、24、32、48 和 64
- 差分输入电压：$\pm 2.7/Gain_{DPGA}$
- 宽广的共模输入范围：-1.5V~2V（增益为 2 倍）
- 建立时间：250ns 至 1.4us
- 增益误差在+/-1%之内

13.2 输入及输入电压范围

DPGA 的输出范围为 0.3V 至 3V，因此 DPGA 的差分输入电压为$\pm 2.7/Gain_{DPGA}$。DPGA 的增益设置和差分输入范围如下表 13-1 所示。

表 13-1：增益及差分输入/输出范围

DPGACTL.GAIN	$Gain_{DPGA}$	Input range (V)	Output range (V)
0	2	-1.35~1.35	0.3~3
1	4	-0.675~+0.675	0.3~3
2	8	-0.337~+0.337	0.3~3
3	16	-0.168~+0.168	0.3~3
4	24	-0.112~+0.112	0.3~3
5	32	-0.084~+0.084	0.3~3
6	48	-0.056~+0.056	0.3~3
7	64	-0.042~+0.042	0.3~3

13.3 应用

图 13-1 显示了如何将 DPGA 用作低侧电流感测放大器。电机电流通过感测电阻 R_{sh} 转换为电压，LP 滤波器用于抑制电磁干扰，建议使用的 R_{filt} 值为 12 欧姆，C_{filt} 为 1nF。DPGA 能够通过电阻 R_{sh} 测量正向和负向电流，输入信号经过 DPGA 放大，并送到 13 位 ADC。R_{stray} 是从感测电阻到地的杂散电阻。

对于 DPGAIP 引脚和 DPGAIN 引脚的输入，输出如下：

$$V_{DPGA_OUTP} = V_{CMOUT} + (V_{DPGAIP} - V_{DPGAIN}) \times Gain_{DPGA} / 2$$

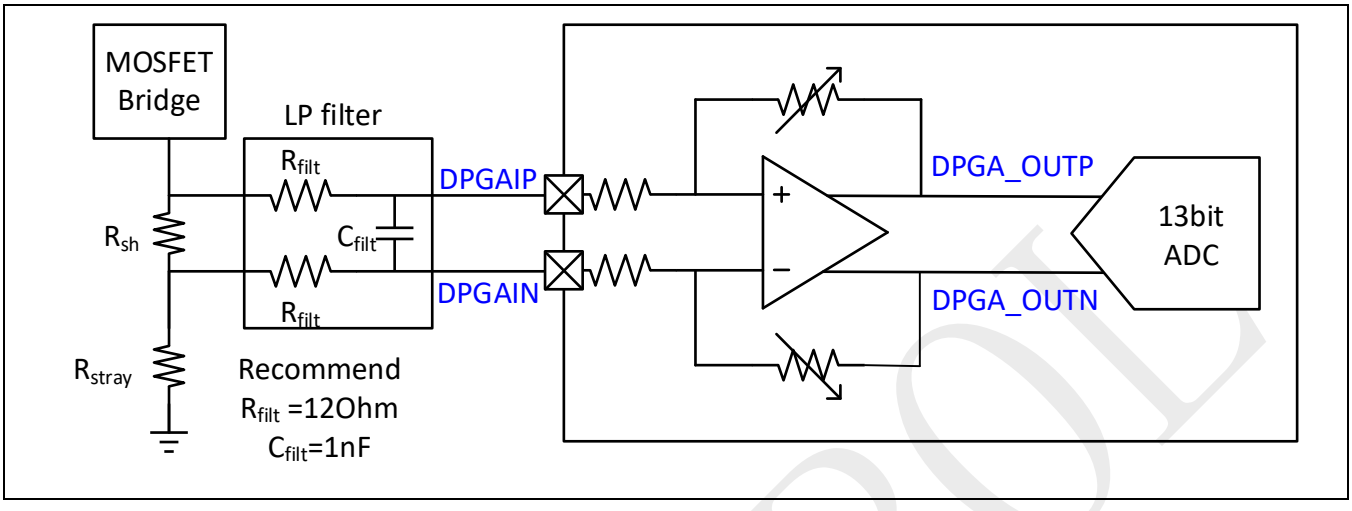
$$V_{DPGA_OUTN} = V_{CMOUT} - (V_{DPGAIP} - V_{DPGAIN}) \times Gain_{DPGA} / 2$$

其中 VCMOUT 是输出的共模电压，计算公式为：

©2025 旋智电子科技（上海）有限公司

$V_{CMOUT} = V_{DVDD33} / 2$

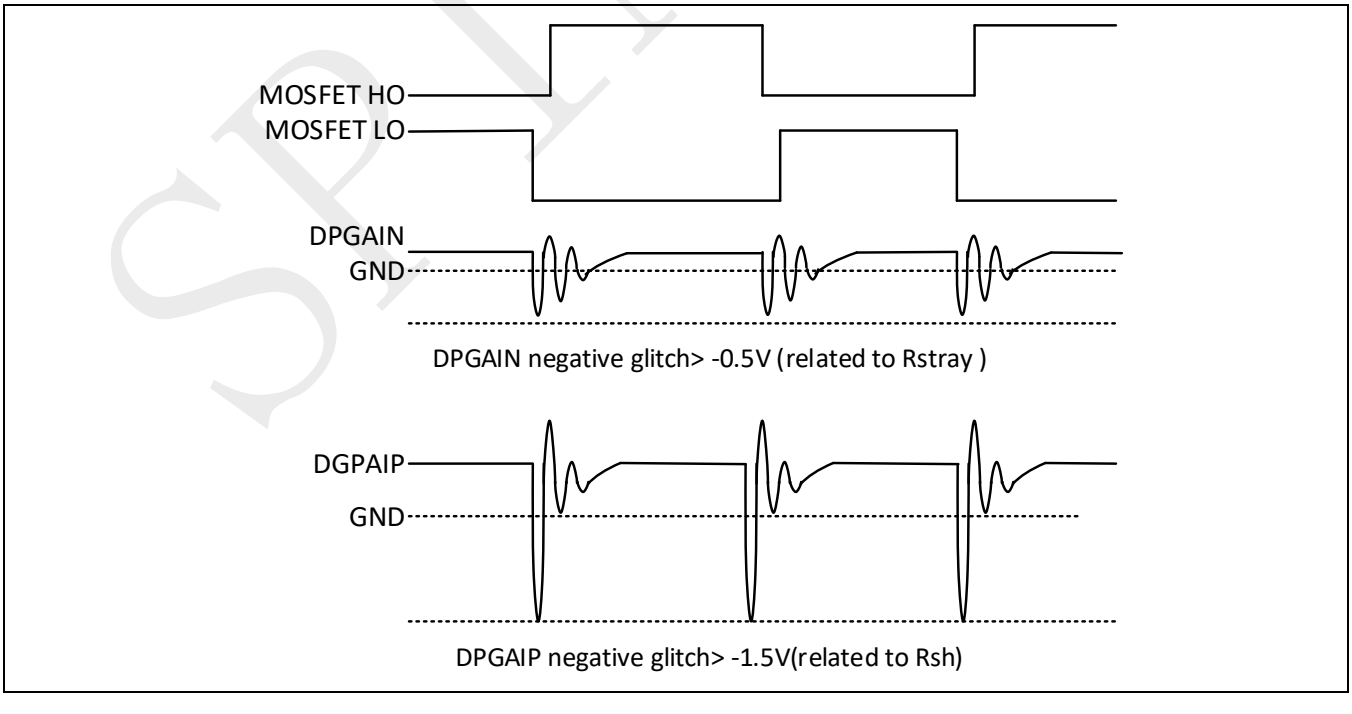
图 13-1：DPGA 应用框图



当电机桥驱动器的高侧 MOSFET 关闭并且低侧 MOSFET 打开时，由于寄生电感，DPGAIP 和 DPGAIN 处会产生电压尖峰，如图 13-2 所示。

- DPGAIN 负向尖峰应在-0.5V 以上，因为 DPGA 的输入共模范围为-0.5V 至 2V。
  - PCB 上的 R_{stray} 应足够小；
- DPGAIP 负向尖峰应在-1.5V 以上，因为 DPGA 输入引脚可以承受-1.5V。
  - PCB 上的 R_{sh} 应选择合理的值；

图 13-2：FPGA 输入范围的要求



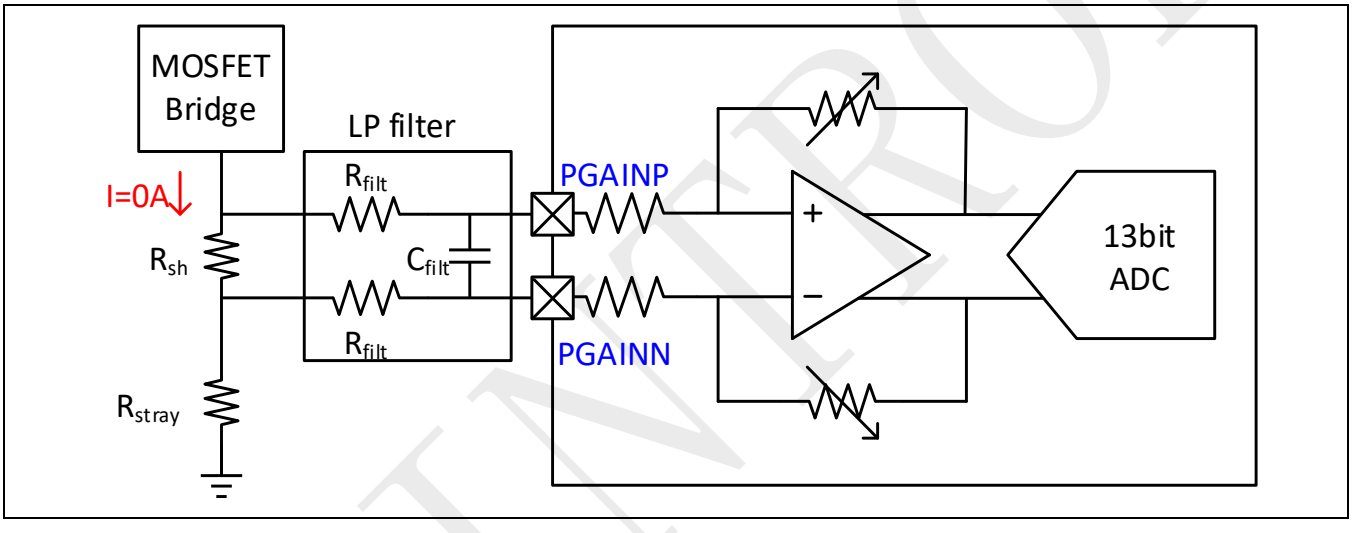
SPINTROL

13.4 偏置校准

为了进行正确的偏置校准，应该测量从感应电阻到 DPGA 再到 13 位 ADC 的完整路径。为此，应确保感应电阻中没有电流（0A），如图 13-3 所示。以下是推荐的偏移校准步骤：

- 将 DPGA 输出连接到 ADC 输入
- 设置所需的 DPGA 增益
- 确保感应电阻中没有电流流过（0A）
- 进行 ADC 测量
- ADC 的数字输出即为完整路径的偏移值
- 如有需要，可以针对其他 DPGA 增益重复测量

图 13-3：DPGA 偏置校准的路径



SPINTROL

©2025 旋智电子科技（上海）有限公司

13.5 寄存器

13.5.1 PGA 寄存器列表

表 13-2：PGA Module 基地址

外设模块	基地址
PGA	0x4000 3000

表 13-3：PGA Register Map

寄存器	偏移地址	描述	复位值
DPGACTL*	0x2D0	DPGA 控制寄存器	0x00000000
SPGACTL*	0x2D4	SPGA 控制寄存器	0x00000000
PGAREGKEY	0x2F0	PGA 模块写使能寄存器	0x1ACCE551

注意： 由*标记的寄存器仅当 PWRREGKEY=0x1ACCE551 时才可以改写。

SPINTROL

13.5.2 PGA 寄存器

表 13-4：DPGA 控制寄存器（DPGACTL）位段定义

DPGACTL (Differential PGA Control Register) Offset：0x2D0 Default：0x00000000 Access：PGAx->DPGACTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							D2SBUFEN
7 PUSTRENGTH	6 PUIN	5 PUIP	4	3	2	1 BYP	0 EN
			GAIN				

表 13-5：DPGA 控制寄存器（DPGACTL）位段描述

位段	位段名	属性	复位值	描述
31:9	RESERVED_31_9	RO	0x0	保留
8	D2SBUFEN	RW	0x0	PGA 双端到单端的 buffer 使能 0：关闭 1：使能
7	PUSTRENGTH	RW	0x0	PGA 上拉强度 0：2 mA 1：10 mA
6	PUIN	RW	0x0	PGA 负端输入上拉使能 0：关闭 1：使能
5	PUIP	RW	0x0	PGA 正端输入上拉使能 0：关闭 1：使能
4:2	GAIN	RW	0x0	PGA 差分增益 000：2x 001：4x 010：8x 011：16x 100：24x 101：32x 110：48x 111：64x
1	BYP	RW	0x0	PGA 旁路直通 0：输入经过放大到输出 1：输入旁路直通到输出

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
0	EN	RW	0x0	PGA 使能 0:关闭 1：使能

表 13-6：SPGA 控制寄存器（SPGACTL）位段定义

SPGACTL (Single-ended PGA Control Register) Offset：0x2D4 Default：0x00000000 Access：PGAx->SPGACTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	GAIN			INSEL			EN

表 13-7：SPGA 控制寄存器（SPGACTL）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6:4	GAIN	RW	0x0	PGA 增益 000：1x 001：2x 010：4x 011：8x 100：16x 101：32x 110：48x 111：64x
3:1	INSEL	RW	0x0	PGA 输入选择 000：ANA_IN0 001：ANA_IN1 010：ANA_IN2 011：ANA_IN3 100：ANA_IN4 101：T-sensor 负端输出 110：T-sensor 正端输出 111：ATEST 输出
0	EN	RW	0x0	PGA 使能 0：关闭 1：使能

©2025 旋智电子科技（上海）有限公司

表 13-8：PGA 模块写使能寄存器（PGAREGKEY）位段定义

PGAREGKEY (PGA Register Write-Allow Key Register) Offset：0x2F0 Default：0x1ACCE551 Access：PGAx->PGAREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

表 13-9：PGA 模块写使能寄存器（PGAREGKEY）位段描述

位段	位段名	属性	复位值	描述
31:0	KEY	RW	0x1ACCE551	写入 0x1ACCE551 解锁受保护的 PGA 寄存器

SPINTROL

©2025 旋智电子科技（上海）有限公司

14 模拟比较器（COMP）

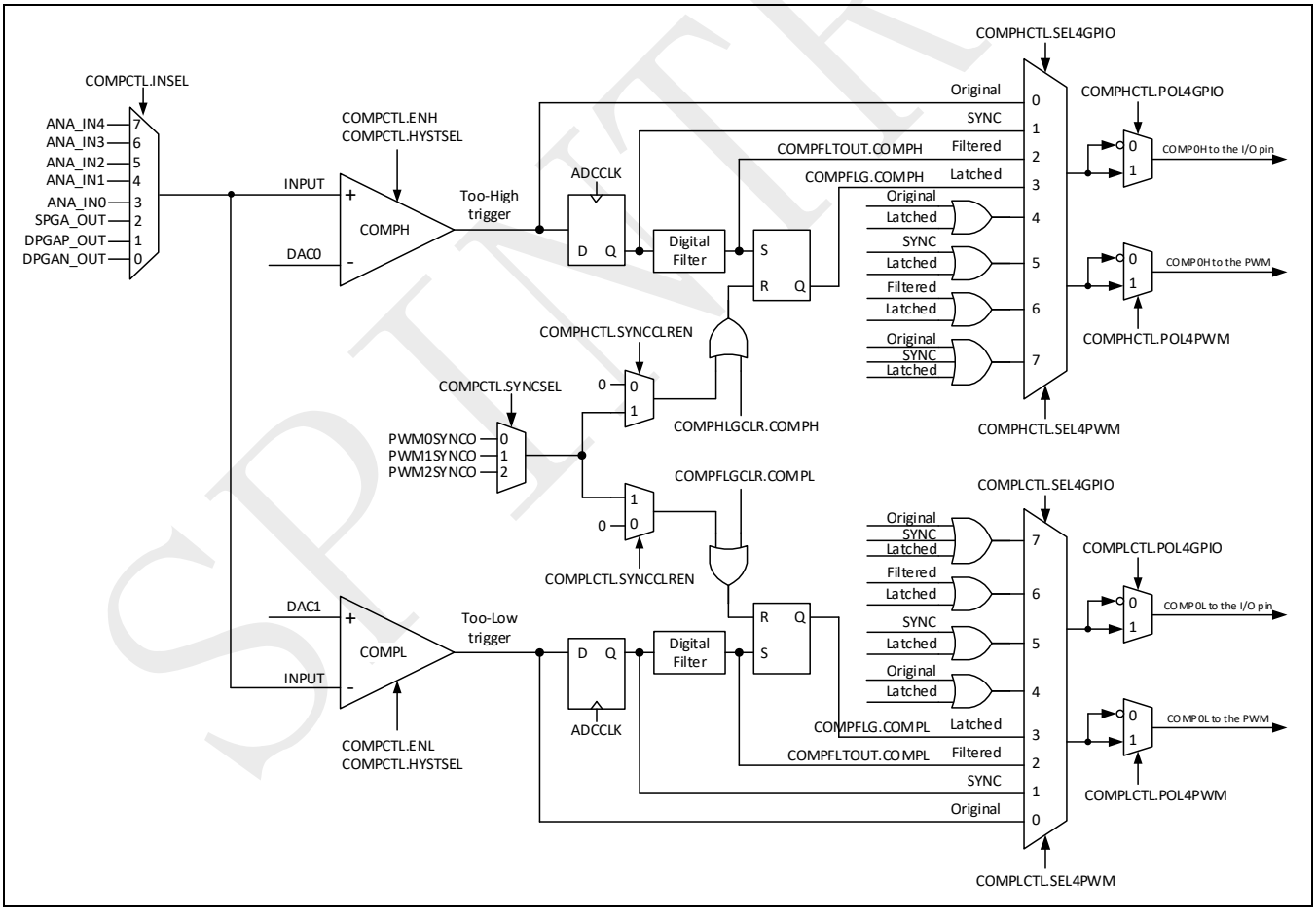
14.1 概述

在 SPC1169 中，有 2 个高速轨对轨输入比较器，用于检测电压是否过高或过低以进行过流检测。

14.2 比较器架构

比较器按照成对设计和使用。图 14-1 以 COMPH 和 COMPL 为例子给出了框图。其中，包括两个比较器，一个用于监测电压过高，另一个用于检查电压过低，还包括比较器的输入 MUX。每个比较器可启用滞后，典型值为 30mV。每个比较器的输出可用作 PWM 错误处理事件，典型延迟为 50ns。

图 14-1：比较器框图



14.3 比较器输入选择

比较器的输入选择如表 14-1 所示。

表 14-1：比较器输入选择

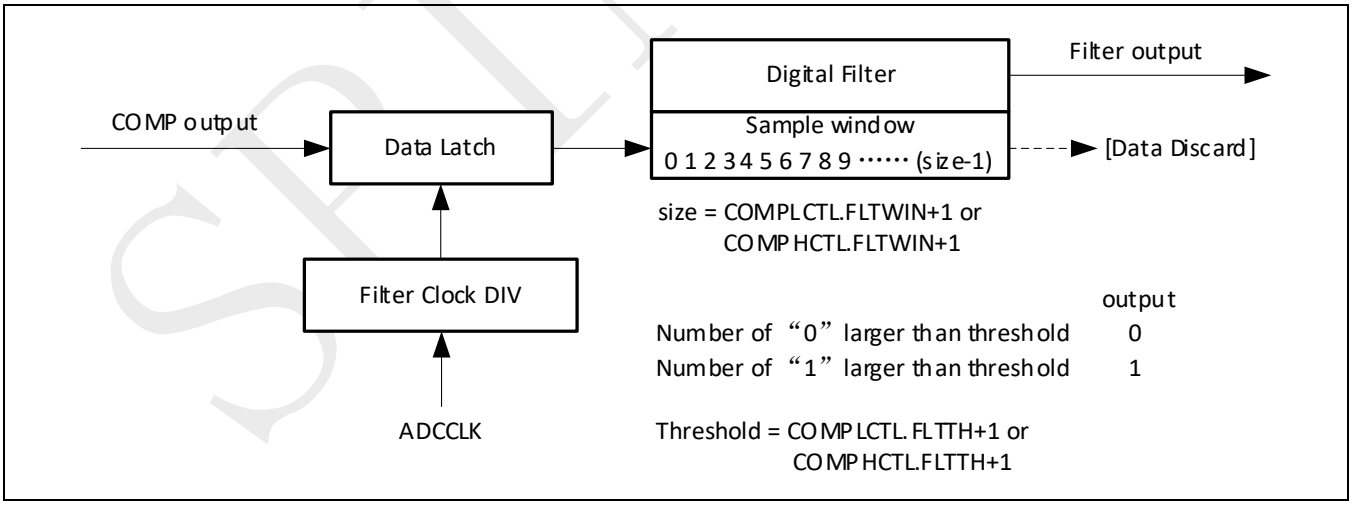
COMPCTL.INSEL	Input selection
7	ANA_IN4
6	ANA_IN3
5	ANA_IN2
4	ANA_IN1
3	ANA_IN0
2	SPGA_OUT
1	DPGA_OUTP
0	DPGA_OUTN

14.4 数字滤波器

数字滤波器根据 COMPL(H)CTL.FLTDIV 的配置，以 PWM 模块时钟分频之后的速率对比较器的输出进行采样。根据 COMPL(H)CTL.FLTWIN 的配置，移位寄存器的前若干个数据参与滤波运算。当这些数里 1 或 0 的个数大于 COMPL(H)CTL.FLTTH 所指定的阈值时，输出对应判定为 1 或 0，否则输出保持不变。为了正确操作，阈值应大于 COMPLCTL.FLTWIN/2 和 COMPHCTL.FLTWIN/2。

数字滤波器的概念模型如图 14-2 所示。

图 14-2：数字滤波器



注意： 考虑到噪声影响，建议采取以下措施：

1. 在 PCB 上为发送给比较器的输入信号添加适当的电容。
2. 使能延滞功能以避免由于噪声而导致的误触发。
3. 使能数字滤波器。

©2025 旋智电子科技（上海）有限公司

14.5 寄存器

14.5.1 COMP 寄存器列表

表 14-2：COMP 模块基地址

外设模块	基地址
COMP	0x4000 3000

表 14-3：COMP 寄存器列表

寄存器	偏移地址	描述	复位值
COMPFLTOUT	0x2F4	比较器滤波输出寄存器	0x00000000
COMPFLG	0x2F8	比较器标志寄存器	0x00000000
COMPFLGCLR	0x2FC	比较器标志清除寄存器	0x00000000
COMPCTL*	0x300	比较器控制寄存器	0x00000000
COMPLCTL*	0x304	COMPL 控制寄存器	0x00000088
COMPHCTL*	0x308	COMPH 控制寄存器	0x00000088
DAC0CTL*	0x360	DAC0 控制寄存器	0x00000002
DAC0CODE	0x364	DAC0 码值寄存器	0x00000000
DAC0CODEA	0x368	DAC0 有效码值寄存器	0x00000000
RAMP0DLY*	0x36C	RAMP0 延时影子寄存器	0x00000000
RAMP0DLYA	0x370	RAMP0 有效延时寄存器	0x00000000
RAMP0DEC*	0x374	RAMP0 递减步长影子寄存器	0x00000000
RAMP0DECA	0x378	RAMP0 有效递减步长寄存器	0x00000000
RAMP0MAX*	0x37C	RAMP0 计数数值重载影子寄存器	0x00000000
RAMP0MAXA	0x380	RAMP0 有效计数数值重载寄存器	0x00000000
RAMP0CNT	0x384	RAMP0 计数寄存器	0x00000000
DAC1CTL*	0x388	DAC1 控制寄存器	0x00000002
DAC1CODE	0x38C	DAC1 码值寄存器	0x00000000
DAC1CODEA	0x390	DAC1 有效码值寄存器	0x00000000
DAC2CTL*	0x3B0	DAC2 控制寄存器	0x00000002
DAC2CODE	0x3B4	DAC2 码值寄存器	0x00000000
DAC2CODEA	0x3B8	DAC2 有效码值寄存器	0x00000000
DACBUFCTL*	0x3F8	DAC 缓冲器控制寄存器	0x00000000
COMPREGKEY	0x3FC	COMP 模块写使能寄存器	0x1ACCE551

注意： 由*标记的寄存器仅当 PWRREGKEY=0x1ACCE551 时才可以改写。

©2025 旋智电子科技（上海）有限公司

14.5.2 COMP 寄存器

表 14-4：比较器滤波输出寄存器（COMPFLTOUT）位段定义

COMPFLTOUT (Comparator Filter Output Register) Offset：0x2F4 Default：0x00000000 Access：COMPx->COMPFLTOUT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						COMPH	COMPL

表 14-5：比较器滤波输出寄存器（COMPFLTOUT）位段描述

位段	位段名	属性	复位值	描述
31:2	RESERVED_31_2	RO	0x0	保留
1	COMPH	RO	0x0	COMPH 滤波输出状态 0：输出为低，表明输入低于参考值 1：输出为高，表明输入高于参考值
0	COMPL	RO	0x0	COMPL 滤波输出状态 0：输出为低，表明输入高于参考值 1：输出为高，表明输入低于参考值

表 14-6：比较器标志寄存器（COMPFLG）位段定义

COMPFLG (Comparator Flag Register) Offset：0x2F8 Default：0x00000000 Access：COMPx->COMPFLG							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						COMPH	COMPL

©2025 旋智电子科技（上海）有限公司

表 14-7：比较器标志寄存器（COMPFLG）位段描述

位段	位段名	属性	复位值	描述
31:2	RESERVED_31_2	RO	0x0	保留
1	COMPH	RO	0x0	锁存的 COMPH 滤波结果标志 0：COMPH 滤波结果一直为低 1：COMPH 滤波结果曾经为高
0	COMPL	RO	0x0	锁存的 COMPL 滤波结果标志 0：COMPL 滤波结果一直为低 1：COMPL 滤波结果曾经为高

表 14-8：比较器标志清除寄存器（COMPFLGCLR）位段定义

COMPFLGCLR (Comparator Flag Clear Register)　Offset：0x2FC　Default：0x00000000
Access：COMPx->COMPFLGCLR

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						COMPH	COMPL

表 14-9：比较器标志清除寄存器（COMPFLGCLR）位段描述

位段	位段名	属性	复位值	描述
31:2	RESERVED_31_2	RO	0x0	保留
1	COMPH	W1C	0x0	COMPH 标志清除 0：写 0 无效，读总是 1 1：写 1 清除标志，本位段自动清零
0	COMPL	W1C	0x0	COMPL 标志清除 0：写 0 无效，读总是 1 1：写 1 清除标志，本位段自动清零

SPINTROL

表 14-10：比较器控制寄存器（COMPCTL）位段定义

COMPCTL (Comparator 0 Control Register) Offset：0x300 Default：0x00000000 Access：COMPx->COMPCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						SYNCSEL	
7	6	5	4	3	2	1	0
RESERVED	INSEL			HYSTSEL		ENH	ENL

表 14-11：比较器控制寄存器（COMPCTL）位段描述

位段	位段名	属性	复位值	描述
31:10	RESERVED_31_10	RO	0x0	保留
9:8	SYNCSEL	RW	0x0	PWM 同步输出信号选择 00：PWM0 同步输出 01：PWM1 同步输出 10：PWM2 同步输出 11：PWM3 同步输出
7	RESERVED_7	RO	0x0	保留
6:4	INSEL	RW	0x0	COMP 输入选择 000：DPGA 负端输出 001：DPGA 正端输出 010：SPGA 输出 011：ANA_IN0 100：ANA_IN1 101：ANA_IN2 110：ANA_IN3 111：ANA_IN4
3:2	HYSTSEL	RW	0x0	COMP 延滞选择 00：0 mV 01：12 mV 10：24 mV 11：36 mV
1	ENH	RW	0x0	COMPH 使能 0：关闭 1：使能
0	ENL	RW	0x0	COMPL 使能 0：关闭 1：使能

©2025 旋智电子科技（上海）有限公司

表 14-12：COMPL 控制寄存器（COMPLCTL）位段定义

COMPLCTL (COMP0L Control Register) Offset：0x304 Default：0x00000088 Access：COMPx->COMPLCTL							
31 RESERVED	30	29 FLTRST	28	27	26 FLTTH	25	24
23	22	21 FLTWIN	20	19	18 FLTDIV	17	16
15	14	13	12 FLTDIV	11	10	9	8 SYNCCLREN
7 POL4GPIO	6	5 SEL4GPIO	4	3 POL4PWM	2	1 SEL4PWM	0

表 14-13：COMPL 控制寄存器（COMPLCTL）位段描述

位段	位段名	属性	复位值	描述
31:30	RESERVED_31_30	RO	0x0	保留
29	FLTRST	W1S	0x0	复位并初始化滤波器 0：写 0 无效，读总是 0 1：写 1 用当前输入值填充滤波器的寄存器链 本位段自动清零
28:24	FLTTH	RW	0x0	当样本里有(FLTTH+1)个 0 或 1 时输出对应翻转
23:19	FLTWIN	RW	0x0	设置滤波样本数目为(FLTWIN+1)
18:9	FLTDIV	RW	0x0	滤波采样时钟为 ADC 时钟经过(FLTDIV+1)分频
8	SYNCCLREN	RW	0x0	使能 PWMSYNCO 清除 COMPFLG.COMPL 标志 0：关闭 1：使能
7	POL4GPIO	RW	0x1	COMPL 输出到引脚的监测信号的极性 0：COMPL 事件时输出低（对 COMPL 取反） 1：COMPL 事件时输出高（与 COMPL 相同）
6:4	SEL4GPIO	RW	0x0	COMPL 输出到引脚的监测信号的选择 0：原始信号 1：经过时钟同步后的信号 2：滤波输出的实时结果 3：滤波输出锁存的标志 4：原始输出或锁存的标志 5：同步后的信号或锁存的标志 6：滤波输出的结果或锁存的标志 7：原始输出、同步后的信号或锁存的标志

SPINTROL

位段	位段名	属性	复位值	描述
3	POL4PWM	RW	0x1	COMPL 输出到 PWM 的封锁信号的极性 0：COMPL 事件时输出低（对 COMPL 取反） 1：COMPL 事件时输出高（与 COMPL 相同）
2:0	SEL4PWM	RW	0x0	COMPL 输出到 PWM 的封锁信号的选择 0：原始信号 1：经过时钟同步后的信号 2：滤波输出的实时结果 3：滤波输出锁存的标志 4：原始输出或锁存的标志 5：同步后的信号或锁存的标志 6：滤波输出的结果或锁存的标志 7：原始输出、同步后的信号或锁存的标志

表 14-14：COMPH 控制寄存器（COMPHCTL）位段定义

COMPHCTL (COMP0H Control Register)　Offset：0x308　Default：0x00000088
Access：COMPx->COMPHCTL

31	30	29	28	27	26	25	24
RESERVED		FLTRST	FLTTH				
23	22	21	20	19	18	17	16
FLTWIN					FLTDIV		
15	14	13	12	11	10	9	8
FLTDIV							SYNCCLREN
7	6	5	4	3	2	1	0
POL4GPIO	SEL4GPIO			POL4PWM	SEL4PWM		

表 14-15：COMPH 控制寄存器（COMPHCTL）位段描述

位段	位段名	属性	复位值	描述
31:30	RESERVED_31_30	RO	0x0	保留
29	FLTRST	W1S	0x0	复位并初始化滤波器 0：写 0 无效，读总是 0 1：写 1 用当前输入值填充滤波器的寄存器链 本位段自动清零
28:24	FLTTH	RW	0x0	当样本里有(FLTTH+1)个 0 或 1 时输出对应翻转
23:19	FLTWIN	RW	0x0	设置滤波样本数目为(FLTWIN+1)
18:9	FLTDIV	RW	0x0	滤波采样时钟为 ADC 时钟经过(FLTDIV+1)分频
8	SYNCCLREN	RW	0x0	使能 PWMSYNCO 清除 COMPFLG.COMPH 标志 0：关闭 1：使能

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
7	POL4GPIO	RW	0x1	COMPH 输出到引脚的监测信号的极性 0：COMPH 事件时输出低（对 COMPH 取反） 1：COMPH 事件时输出高（与 COMPH 相同）
6:4	SEL4GPIO	RW	0x0	COMPH 输出到引脚的监测信号的选择 0：原始信号 1：经过时钟同步后的信号 2：滤波输出的实时结果 3：滤波输出锁存的标志 4：原始输出或锁存的标志 5：同步后的信号或锁存的标志 6：滤波输出的结果或锁存的标志 7：原始输出、同步后的信号或锁存的标志
3	POL4PWM	RW	0x1	COMPH 输出到 PWM 的封锁信号的极性 0：COMPH 事件时输出低（对 COMPH 取反） 1：COMPH 事件时输出高（与 COMPH 相同）
2:0	SEL4PWM	RW	0x0	COMPH 输出到 PWM 的封锁信号的选择 0：原始信号 1：经过时钟同步后的信号 2：滤波输出的实时结果 3：滤波输出锁存的标志 4：原始输出或锁存的标志 5：同步后的信号或锁存的标志 6：滤波输出的结果或锁存的标志 7：原始输出、同步后的信号或锁存的标志

表 14-16：DAC0 控制寄存器（DAC0CTL）位段定义

DAC0CTL (DAC0 Control Register)　Offset：0x360　Default：0x00000002
Access：COMPx->DAC0CTL

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
DBGRUN		RESERVED					
7	6	5	4	3	2	1	0
RAMPLOAD	RAMPEN	RESERVED		SYNCSEL		CODELOAD	EN

SPINTROL

表 14-17：DAC0 控制寄存器（DAC0CTL）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:14	DBGRUN	RW	0x0	斜坡发生器在 CPU halted 或 lockup 时的行为 0：立刻停止 1：在下一个 PWMSYNCO 事件时停止 2：继续运行
13:8	RESERVED_13_8	RO	0x0	保留
7	RAMPLOAD	RW	0x0	斜坡发生器加载初值的配置模式 0：影子模式 RAMP 计数器从 RAMPMAXA 寄存器加载数值 1：即刻模式 RAMP 计数器直接从 RAMPMAX 加载数值
6	RAMPEN	RW	0x0	DAC 斜坡发生器使能 0：关闭 1：使能 此时 DAC0CODEA 为 RAMP0CNT 的高 10 位
5:4	RESERVED_5_4	RO	0x0	保留
3:2	SYNCSEL	RW	0x0	PWM 同步输出信号选择 00：PWM0 同步输出 01：PWM1 同步输出 10：PWM2 同步输出 11：PWM3 同步输出
1	CODELOAD	RW	0x1	DAC 码值更新方式 0：写入 DAC0CODE 的值在 PWMSYNCO 事件时才会更新到 DAC0CODEA 并生效 1：写 DAC0CODE 时即刻更新 DAC0CODEA 并生效
0	EN	RW	0x0	DAC 使能 0：关闭 1：使能

©2025 旋智电子科技（上海）有限公司

表 14-18：DAC0 码值寄存器（DAC0CODE）位段定义

DAC0CODE (DAC0 Code Register) Offset：0x364 Default：0x00000000 Access：COMPx->DAC0CODE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						VAL	
7	6	5	4	3	2	1	0
VAL							

表 14-19：DAC0 码值寄存器（DAC0CODE）位段描述

位段	位段名	属性	复位值	描述
31:10	RESERVED_31_10	RO	0x0	保留
9:0	VAL	RW	0x0	DAC0 码值 DAC0CTL.CODELOAD = 1 时，写本寄存器时会同时更新 DAC0CODEA

表 14-20：DAC0 有效码值寄存器（DAC0CODEA）位段定义

DAC0CODEA (DAC0 Active Code Register) Offset：0x368 Default：0x00000000 Access：COMPx->DAC0CODEA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						VAL	
7	6	5	4	3	2	1	0
VAL							

表 14-21：DAC0 有效码值寄存器（DAC0CODEA）位段描述

位段	位段名	属性	复位值	描述
31:10	RESERVED_31_10	RO	0x0	保留
9:0	VAL	RO	0x0	DAC0 的有效码值 DAC0CTL.CODELOAD = 1：写 DAC0CODE 时会同时更新本寄存器； DAC0CTL.CODELOAD = 0：当指定的 PWMSYNCO 发生时会从 DAC0CODE 更新本寄存器。

©2025 旋智电子科技（上海）有限公司

表 14-22：RAMP0 延时影子寄存器（RAMP0DLY）位段定义

RAMP0DLY (RAMP0 Delay Shadow Register) Offset：0x36C Default：0x00000000 Access：COMPx->RAMP0DLY							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 14-23：RAMP0 延时影子寄存器（RAMP0DLY）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x0	从指定的 PWMSYNCO 发生到斜坡产生的延时 本寄存器不直接控制延时大小，写入的值需要等到指定的 PWMSYNCO 事件发生才能更新到 RAMP0DLYA 寄存器并生效

表 14-24：RAMP0 有效延时寄存器（RAMP0DLYA）位段定义

RAMP0DLYA (RAMP0 Delay Active Register) Offset：0x370 Default：0x00000000 Access：COMPx->RAMP0DLYA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 14-25：RAMP0 有效延时寄存器（RAMP0DLYA）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RO	0x0	从指定的 PWMSYNCO 发生到斜坡产生的延时 本寄存器控制当前延时大小，当 PWMSYNCO 事件发生时根据 RAMP0DLY 更新本寄存器并生效

©2025 旋智电子科技（上海）有限公司

表 14-26：RAMP0 递减步长影子寄存器（RAMP0DEC）位段定义

RAMP0DEC (RAMP0 Decrement Shadow Register) Offset：0x374 Default：0x00000000 Access：COMPx->RAMP0DEC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 14-27：RAMP0 递减步长影子寄存器（RAMP0DEC）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x0	斜坡产生的递减步长 本寄存器不直接控制递减步长，写入的值需要等到指定的 PWMSYNCO 事件发生才能更新到 RAMP0DECA 寄存器并生效

表 14-28：RAMP0 有效递减步长寄存器（RAMP0DECA）位段定义

RAMP0DECA (RAMP0 Decrement Active Register) Offset：0x378 Default：0x00000000 Access：COMPx->RAMP0DECA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 14-29：RAMP0 有效递减步长寄存器（RAMP0DECA）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RO	0x0	斜坡产生的递减步长 本寄存器控制当前递减步长，当 PWMSYNCO 事件发生时根据 RAMP0DEC 更新本寄存器并生效

©2025 旋智电子科技（上海）有限公司

表 14-30：RAMP0 计数数值重载影子寄存器（RAMP0MAX）位段定义

RAMP0MAX (RAMP0 Reload Value Shadow Register) Offset：0x37C Default：0x00000000 Access：COMPx->RAMP0MAX							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 14-31：RAMP0 计数数值重载影子寄存器（RAMP0MAX）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x0	当 DAC0CTL.RAMPLOAD=1 时，斜波发生器复位时从这个寄存器重新加载的计数数值

表 14-32：RAMP0 有效计数数值重载寄存器（RAMP0MAXA）位段定义

RAMP0MAXA (RAMP0 Reload Value Active Register) Offset：0x380 Default：0x00000000 Access：COMPx->RAMP0MAXA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 14-33：RAMP0 有效计数数值重载寄存器（RAMP0MAXA）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RO	0x0	当 DAC0CTL.RAMPLOAD=0 时，斜波发生器复位时从这个寄存器重新加载的计数数值

©2025 旋智电子科技（上海）有限公司

表 14-34：RAMP0 计数寄存器（RAMP0CNT）位段定义

RAMP0CNT (RAMP0 Count Register) Offset：0x384 Default：0x00000000 Access：COMPx->RAMP0CNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 14-35：RAMP0 计数寄存器（RAMP0CNT）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RO	0x0	斜坡产生的计数值 当 DAC0CTL.RAMPEN = 1 时，DAC0 的码值为本寄存器的高 10 位

表 14-36：DAC1 控制寄存器（DAC1CTL）位段定义

DAC1CTL (DAC1 Control Register) Offset：0x388 Default：0x00000002 Access：COMPx->DAC1CTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
DBGRUN		RESERVED					
7	6	5	4	3	2	1	0
RAMPLOAD	RAMPEN	RESERVED		SYNCSEL		CODELOAD	EN

表 14-37：DAC1 控制寄存器（DAC1CTL）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:14	DBGRUN	RW	0x0	斜坡发生器在 CPU halted 或 lockup 时的行为 0：立刻停止 1：在下一个 PWMSYNCO 事件时停止 2：继续运行
13:8	RESERVED_13_8	RO	0x0	保留

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
7	RAMPLOAD	RW	0x0	斜坡发生器加载初值的配置模式 0：影子模式 RAMP 计数器从 RAMPMAXA 寄存器加载数值 1：即刻模式 RAMP 计数器直接从 RAMPMAX 加载数值
6	RAMPEN	RW	0x0	DAC 斜坡发生器使能 0：关闭 1：使能 此时 DAC0CODEA 为 RAMP0CNT 的高 10 位
5:4	RESERVED_5_4	RO	0x0	保留
3:2	SYNCSEL	RW	0x0	PWM 同步输出信号选择 00：PWM0 同步输出 01：PWM1 同步输出 10：PWM2 同步输出 11：PWM3 同步输出
1	CODELOAD	RW	0x1	DAC 码值更新方式 0：写入 DAC0CODE 的值在 PWMSYNCO 事件时才会更新到 DAC0CODEA 并生效 1：写 DAC0CODE 时即刻更新 DAC0CODEA 并生效
0	EN	RW	0x0	DAC 使能 0：关闭 1：使能

表 14-38：DAC1 码值寄存器（DAC1CODE）位段定义

DAC1CODE (DAC1 Code Register)　Offset：0x38C　Default：0x00000000
Access：COMPx->DAC1CODE

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						VAL	
7	6	5	4	3	2	1	0
VAL							

©2025 旋智电子科技（上海）有限公司

表 14-39：DAC1 码值寄存器（DAC1CODE）位段描述

位段	位段名	属性	复位值	描述
31:10	RESERVED_31_10	RO	0x0	保留
9:0	VAL	RW	0x0	DAC1 码值 DAC1CTL.CODELOAD = 1 时，写本寄存器时会同时更新 DAC1CODEA

表 14-40：DAC1 有效码值寄存器（DAC1CODEA）位段定义

DAC1CODEA (DAC1 Active Code Register)　Offset：0x390　Default：0x00000000
Access：COMPx->DAC1CODEA

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						VAL	
7	6	5	4	3	2	1	0
VAL							

表 14-41：DAC1 有效码值寄存器（DAC1CODEA）位段描述

位段	位段名	属性	复位值	描述
31:10	RESERVED_31_10	RO	0x0	保留
9:0	VAL	RO	0x0	DAC1 的有效码值 DAC1CTL.CODELOAD = 1：写 DAC1CODE 时会同时更新本寄存器； DAC1CTL.CODELOAD = 0：当指定的 PWMSYNCO 发生时会从 DAC1CODE 更新本寄存器。

表 14-42:：DAC2 控制寄存器（DAC2CTL）位段定义

DAC2CTL (DAC2 Control Register)　Offset：0x3B0　Default：0x00000002
Access：COMPx->DAC2CTL

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				SYNCSEL		CODELOAD	EN

SPINTROL

表 14-43：DAC2 控制寄存器（DAC2CTL）位段描述

位段	位段名	属性	复位值	描述
31:4	RESERVED_31_4	RO	0x0	保留
3:2	SYNCSEL	RW	0x0	PWM 同步输出信号选择 00：PWM0 同步输出 01：PWM1 同步输出 10：PWM2 同步输出 11：PWM3 同步输出
1	CODELOAD	RW	0x1	DAC 码值更新方式 0：写入 DAC2CODE 的值在 PWMSYNCO 事件时才会更新到 DAC2CODEA 并生效 1：写 DAC2CODE 时即刻更新 DAC2CODEA 并生效
0	EN	RW	0x0	DAC 使能 0：关闭 1：使能

表 14-44：DAC2 码值寄存器（DAC2CODE）位段定义

DAC2CODE (DAC2 Code Register)　Offset：0x3B4　Default：0x00000000
Access：COMPx->DAC2CODE

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						VAL	
7	6	5	4	3	2	1	0
VAL							

表 14-45：DAC2 码值寄存器（DAC2CODE）位段描述

位段	位段名	属性	复位值	描述
31:10	RESERVED_31_10	RO	0x0	保留
9:0	VAL	RW	0x0	DAC2 码值 DAC2CTL.CODELOAD = 1 时，写本寄存器时会同时更新 DAC2CODEA

©2025 旋智电子科技（上海）有限公司

表 14-46：DAC2 有效码值寄存器（DAC2CODEA）位段定义

DAC2CODEA (DAC2 Active Code Register) Offset：0x3B8 Default：0x00000000 Access：COMPx->DAC2CODEA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						VAL	
7	6	5	4	3	2	1	0
VAL							

表 14-47：DAC2 有效码值寄存器（DAC2CODEA）位段描述

位段	位段名	属性	复位值	描述
31:10	RESERVED_31_10	RO	0x0	保留
9:0	VAL	RO	0x0	DAC2 的有效码值 DAC2CTL.CODELOAD = 1：写 DAC2CODE 时会同时更新本寄存器； DAC2CTL.CODELOAD = 0：当指定的 PWMSYNCO 发生时会从 DAC2CODE 更新本寄存器。

表 14-48：DAC 缓冲器控制寄存器（DACBUFCTL）位段定义

DACBUFCTL (DAC Buffer Control Register) Offset：0x3F8 Default：0x00000000 Access：COMPx->DACBUFCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED					INSEL	OE	EN

©2025 旋智电子科技（上海）有限公司

表 14-49：DAC 缓冲器控制寄存器（DACBUFCTL）位段描述

位段	位段名	属性	复位值	描述
31:3	RESERVED_31_3	RO	0x0	保留
2	INSEL	RW	0x0	DAC 缓冲器输入选择 0：DAC0 1：DAC1 2：DAC2 3：DAC3
1	OE	RW	0x0	GPIO 引脚输出使能 0：关闭 1：使能
0	EN	RW	0x0	DAC 缓冲器使能 0：关闭 1：使能

表 14-50：COMP 模块写使能寄存器（COMPREGKEY）位段定义

COMPREGKEY (COMP Register Write-Allow Key Register) Offset：0x3FC Default：0x1ACCE551 Access：COMPx->COMPREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

表 14-51：COMP 模块写使能寄存器（COMPREGKEY）位段描述

位段	位段名	属性	复位值	描述
31:0	KEY	RW	0x1ACCE551	写入 0x1ACCE551 解锁受保护的 COMP 寄存器

©2025 旋智电子科技（上海）有限公司

15 相位比较器（PHCOMP）

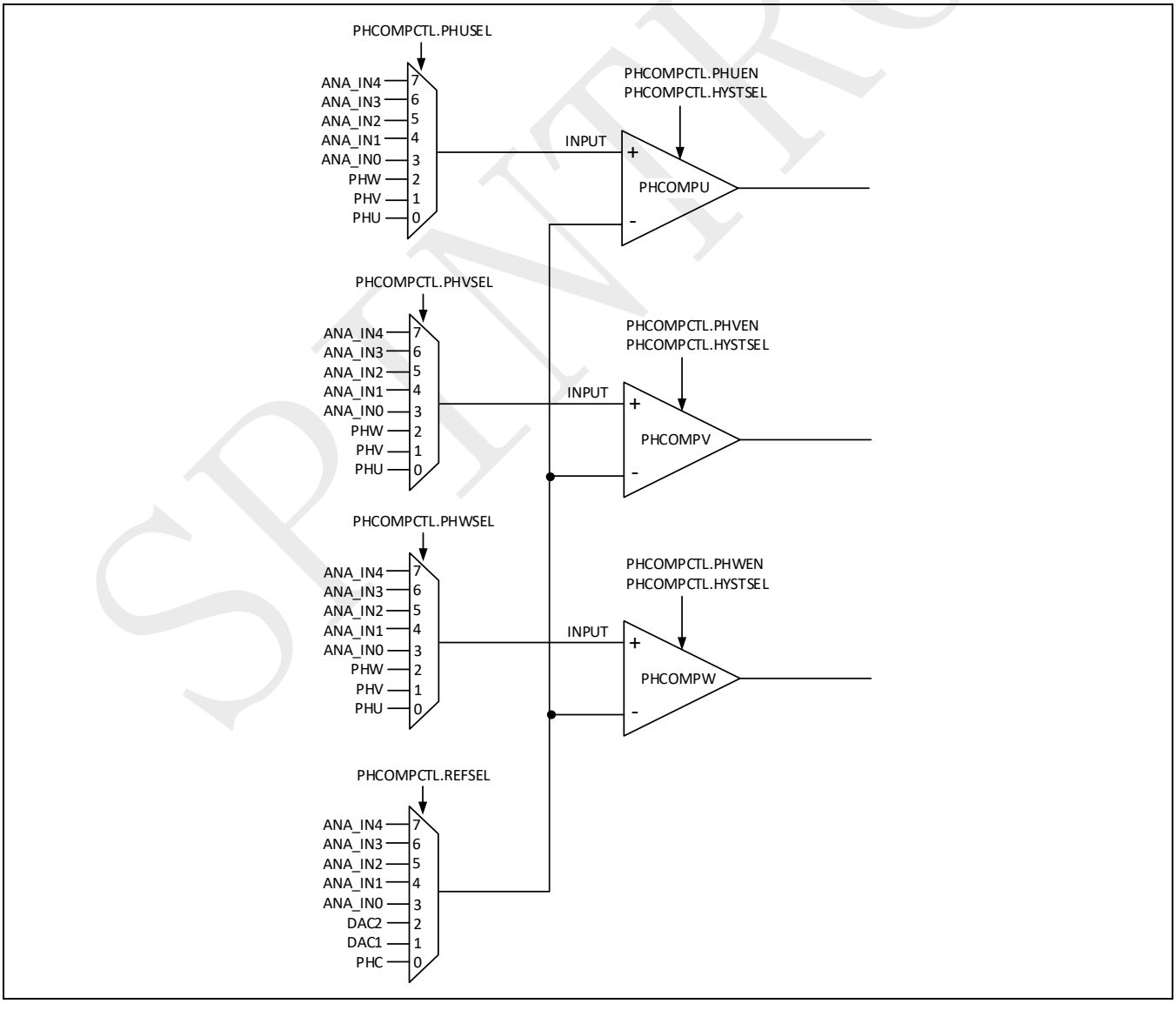
15.1 概述

SPC1169 包含 3 个高速全差动相位比较器，可以将三相电压（PHU/PHV/PHW）与虚拟中性点电压（PHC）进行比较。相位比较器具有输入多路选择器，也可以用作普通比较器，用于将 ADC 输入与参考电压（ADC 输入或 DAC 输出）进行比较。

15.2 相位比较器架构

图 15-1 显示了相位比较器的框图，其中包括三个相位比较器，还包括输入多路选择器用于选择比较器的输入信号。每个相位比较器都具有典型值为 30mV 的延滞特性，每个相位比较器的输出典型延迟时间为 200ns。

图 15-1：相位比较器框图



©2025 旋智电子科技（上海）有限公司

15.3 后处理过程

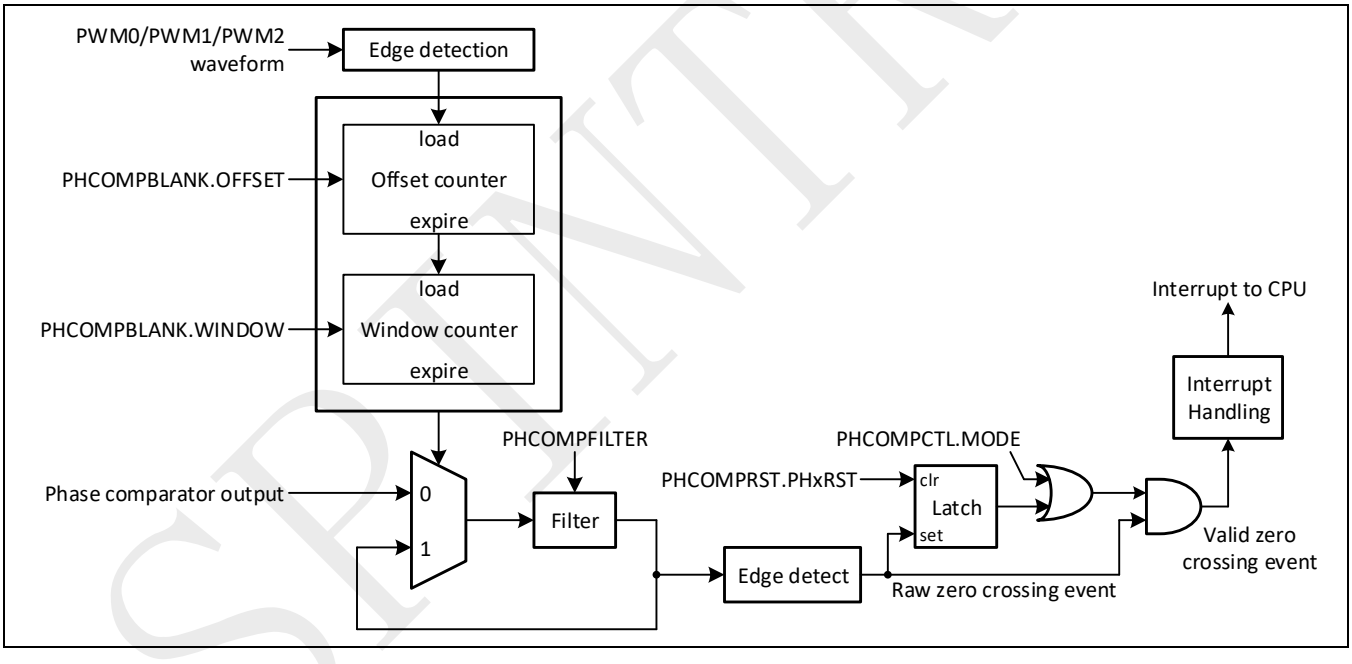
如图 15-2 所示，相位比较器的原始输出将经过一系列处于 HCLK 时钟域的后处理操作。

- 带有可编程偏移和窗口的消隐功能（目的是为了消除由于 PWM 切换而导致的干扰）
- 带有可编程时钟分频的数字滤波功能
- 有限状态机用于消除由于 PWM 切换而导致的首次过零点事件
  - 相位比较器模式（PHCOMPCTL.MODE=0）

    在 PHCOMPRST 寄存器中将相应的 PHxRST 位写为 1 以重置有限状态机后，将忽略滤波器输出的第一个检测到的过零点事件，直到再次重置状态机之前，所有后续的过零点事件都将发送到中断处理模块。
  - 正常比较器模式（PHCOMPCTL.MODE=1）

    有限状态机被绕过，所有检测到的过零点事件都被视为有效。
- 中断处理

图 15-2：相位比较器后处理过程

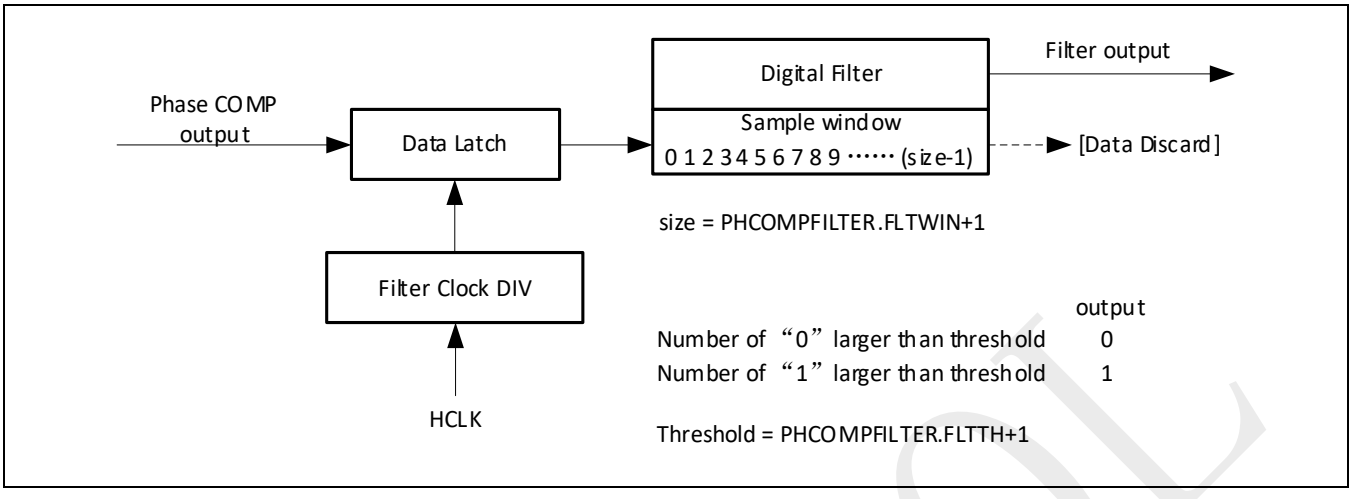


数字滤波器模块的详细信息如图 15-3 所示。

相位比较器的输出被发送到数字滤波器的采样窗口中。数字滤波器的采样时钟来自 HCLK，并且可以通过 PHCOMPFILTER.FLTDIV 进行分频，窗口大小由 PHCOMPFILTER.FLTWIN 设定。数字滤波器计算采样窗口中的大多数数值的数量，如果该数量大于预定义的阈值 PHCOMPFILTER.FLTTH，则滤波器输出将改变为大多数值；否则，滤波器输出将不会改变。为了确保数字滤波器正常运行，阈值应大于 PHCOMPFILTER.FLTWIN/2。

©2025 旋智电子科技（上海）有限公司

图 15-3：数字滤波器的行为



注意： 考虑到噪声影响，建议采取以下措施：

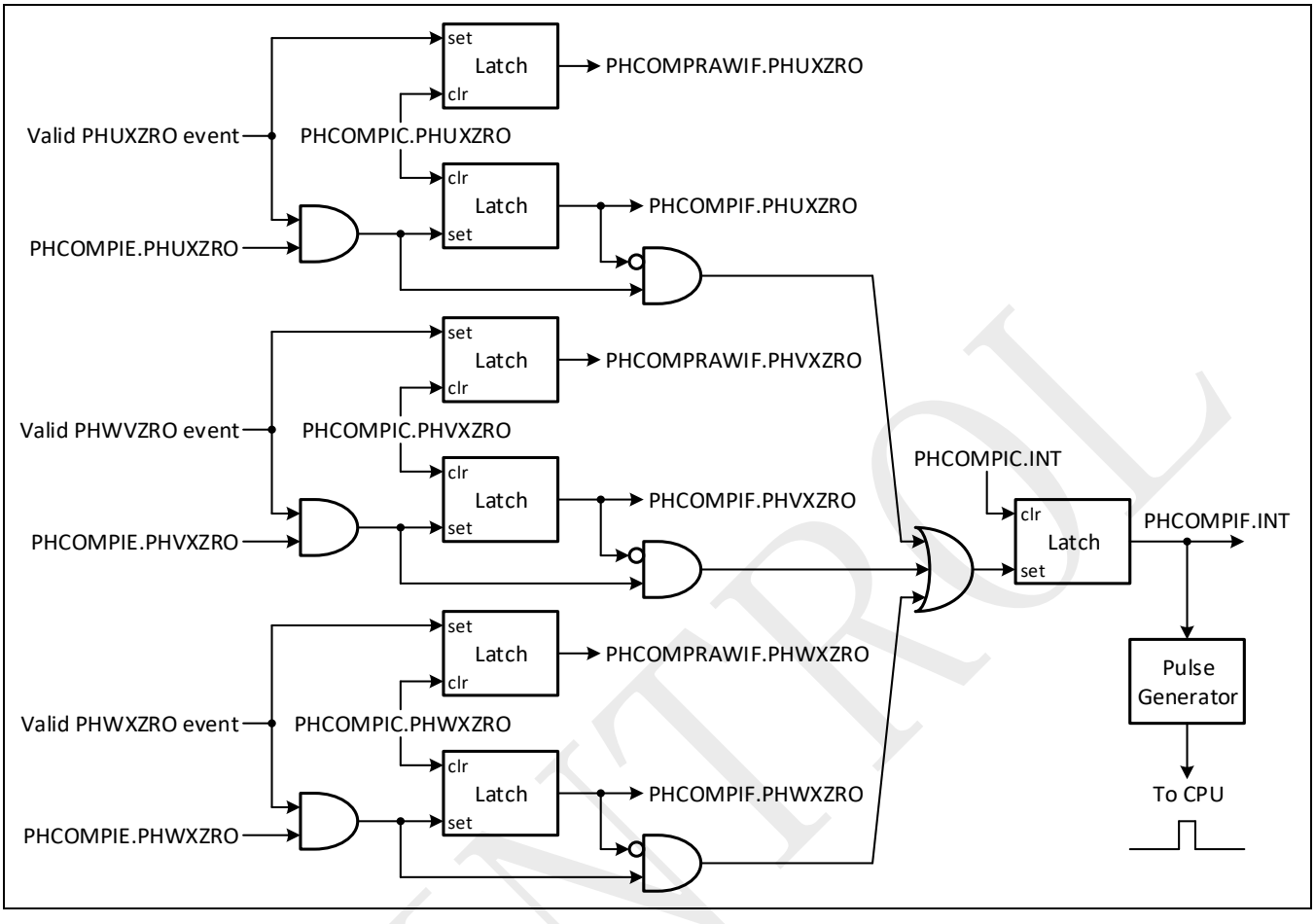
1. 在 PCB 上为发送给比较器的输入信号添加适当的电容。
2. 使能延滞功能以避免由于噪声而导致的误触发。
3. 使能数字滤波器。

相位比较器中断处理模块的详细信息如图 15-4 所示。

- 三相过零点事件触发相应的事件标志位
- 当使能时，过零点事件触发相应的中断标志位，如果使能该中断，将触发中断；在该事件标志被被置位期间，如果继续发生该中断，则该次中断被忽略，直到标志位被清除。
- 一旦向 CPU 发出中断请求，PHCOMPIF.INT 中断标志位也被设置，直到这个标志位被清除之前，将不会产生中断脉冲。

SPINTROL

图 15-4：相位比较器中断



SPINTROL

©2025 旋智电子科技（上海）有限公司

15.4 寄存器

15.4.1 PHCOMP 寄存器列表

表 15-1：PHCOMP 模块基地址

外设模块	基地址
PHCOMP	0x4000 0500

表 15-2：PHCOMP 寄存器列表

寄存器	偏移地址	描述	复位值
PHCOMPSTS	0x0	PHCOMP 状态寄存器	0x00000000
PHCOMPRAWIF	0x4	PHCOMP 中断原始标志寄存器	0x00000000
PHCOMPIF	0x8	PHCOMP 中断标志寄存器	0x00000000
PHCOMPIC	0xC	PHCOMP 中断清除寄存器	0x00000000
PHCOMPIE*	0x10	PHCOMP 中断使能寄存器	0x00000000
PHCOMPCTL*	0x14	PHCOMP 控制寄存器	0x00000420
PHCOMPBLANK*	0x18	PHCOMP 消影寄存器	0x00000000
PHCOMPFILTER*	0x1C	PHCOMP 滤波寄存器	0x00000000
PHCOMPRST	0x20	PHCOMP 有限状态机复位寄存器	0x00000000
PHCOMPREGKEY	0x24	PHCOMP 模块写使能寄存器	0x1ACCE551

注意： 由*标记的寄存器仅当 PWRREGKEY=0x1ACCE551 时才可以改写。

SPINTROL

15.4.2 PHCOMP 寄存器

表 15-3：PHCOMP 状态寄存器（PHCOMPSTS）位段定义

PHCOMPSTS (Phase Comparison Status Register) Offset：0x0 Default：0x00000000 Access：PHCOMPx->PHCOMPSTS							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED		PHWRAW	PHVRAW	PHURAW	PHWFLT	PHVFLT	PHUFLT

表 15-4：PHCOMP 状态寄存器（PHCOMPSTS）位段描述

位段	位段名	属性	复位值	描述
31:6	RESERVED_31_6	RO	0x0	保留
5	PHWRAW	RO	0x0	W 相实时原始输出状态 0：低 1：高
4	PHVRAW	RO	0x0	V 相实时原始输出状态 0：低 1：高
3	PHURAW	RO	0x0	U 相实时原始输出状态 0：低 1：高
2	PHWFLT	RO	0x0	W 相实时滤波输出状态 0：低 1：高
1	PHVFLT	RO	0x0	V 相实时滤波输出状态 0：低 1：高
0	PHUFLT	RO	0x0	U 相实时滤波输出状态 0：低 1：高

©2025 旋智电子科技（上海）有限公司

表 15-5：PHCOMP 中断原始标志寄存器（PHCOMPRAWIF）位段定义

PHCOMPRAWIF (Phase Comparison Interrupt Raw Flag Register) Offset：0x4 Default：0x00000000 Access：PHCOMPx->PHCOMPRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED		PHWZCH	PHWZCL	PHVZCH	PHVZCL	PHUZCH	PHUZCL

表 15-6：PHCOMP 中断原始标志寄存器（PHCOMPRAWIF）位段描述

位段	位段名	属性	复位值	描述
31:6	RESERVED_31_6	RO	0x0	保留
5	PHWZCH	RO	0x0	W 相过零点到高电平事件标志 0：事件没有发生 1：事件已经发生
4	PHWZCL	RO	0x0	W 相过零点到低电平事件标志 0：事件没有发生 1：事件已经发生
3	PHVZCH	RO	0x0	V 相过零点到高电平事件标志 0：事件没有发生 1：事件已经发生
2	PHVZCL	RO	0x0	V 相过零点到低电平事件标志 0：事件没有发生 1：事件已经发生
1	PHUZCH	RO	0x0	U 相过零点到高电平事件标志 0：事件没有发生 1：事件已经发生
0	PHUZCL	RO	0x0	U 相过零点到低电平事件标志 0：事件没有发生 1：事件已经发生

SPINTROL

表 15-7：PHCOMP 中断标志寄存器（PHCOMPIF）位段定义

PHCOMPIF (Phase Comparison Interrupt Flag Register) Offset：0x8 Default：0x00000000 Access： PHCOMPx->PHCOMPIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 RESERVED	6 INT	5 PHWZCH	4 PHWZCL	3 PHVZCH	2 PHVZCL	1 PHUZCH	0 PHUZCL

表 15-8：PHCOMP 中断标志寄存器（PHCOMPIF）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6	INT	RO	0x0	全局 PHCOMP 中断标志 0：没有中断发生 1：中断已经发生 标志位置位期间如果还有中断发生，将被忽略，直至标志位清零
5	PHWZCH	RO	0x0	W 相过零点到高电平中断标志 0：没有中断发生 1：中断已经发生 标志位置位期间如果还有中断发生，将被忽略，直至标志位清零
4	PHWZCL	RO	0x0	W 相过零点到高低平中断标志 0：没有中断发生 1：中断已经发生 标志位置位期间如果还有中断发生，将被忽略，直至标志位清零
3	PHVZCH	RO	0x0	V 相过零点到高电平中断标志 0：没有中断发生 1：中断已经发生 标志位置位期间如果还有中断发生，将被忽略，直至标志位清零
2	PHVZCL	RO	0x0	V 相过零点到高低平中断标志 0：没有中断发生 1：中断已经发生 标志位置位期间如果还有中断发生，将被忽略，直至标志位清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
1	PHUZCH	RO	0x0	V 相过零点到高电平中断标志 0：没有中断发生 1：中断已经发生 标志位置位期间如果还有中断发生，将被忽略，直至标志位清零
0	PHUZCL	RO	0x0	V 相过零点到高低平中断标志 0：没有中断发生 1：中断已经发生 标志位置位期间如果还有中断发生，将被忽略，直至标志位清零

表 15-9：PHCOMP 中断清除寄存器（PHCOMPIC）位段定义

PHCOMPIC (Phase Comparison Interrupt Clear Register) Offset：0xC Default：0x00000000 Access：PHCOMPx->PHCOMPIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 RESERVED	6 INT	5 PHWZCH	4 PHWZCL	3 PHVZCH	2 PHVZCL	1 PHUZCH	0 PHUZCL

表 15-10：PHCOMP 中断清除寄存器（PHCOMPIC）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6	INT	W1C	0x0	全局中断清除 0：写 0 无效，读总为 0 1：写 1 清除中断标志，本位段自动清零
5	PHWZCH	W1C	0x0	W 相过零点到高电平中断清除 0：写 0 无效，读总为 0 1：写 1 清除中断标志，本位段自动清零
4	PHWZCL	W1C	0x0	W 相过零点到低电平中断清除 0：写 0 无效，读总为 0 1：写 1 清除中断标志，本位段自动清零
3	PHVZCH	W1C	0x0	V 相过零点到高电平中断清除 0：写 0 无效，读总为 0 1：写 1 清除中断标志，本位段自动清零
2	PHVZCL	W1C	0x0	V 相过零点到低电平中断清除 0：写 0 无效，读总为 0 1：写 1 清除中断标志，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
1	PHUZCH	W1C	0x0	U 相过零点到高电平中断清除 0：写 0 无效，读总为 0 1：写 1 清除中断标志，本位段自动清零
0	PHUZCL	W1C	0x0	U 相过零点到低电平中断清除 0：写 0 无效，读总为 0 1：写 1 清除中断标志，本位段自动清零

表 15-11：PHCOMP 中断使能寄存器（PHCOMPIE）位段定义

PHCOMPIE (Phase Comparison Interrupt Enable Register) Offset：0x10 Default：0x00000000 Access：PHCOMPx->PHCOMPIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED		PHWZCH	PHWZCL	PHVZCH	PHVZCL	PHUZCH	PHUZCL

表 15-12：PHCOMP 中断使能寄存器（PHCOMPIE）位段描述

位段	位段名	属性	复位值	描述
31:6	RESERVED_31_6	RO	0x0	保留
5	PHWZCH	RW	0x0	W 相过零点到高电平中断使能 0：关闭 1：使能
4	PHWZCL	RW	0x0	W 相过零点到低电平中断使能 0：关闭 1：使能
3	PHVZCH	RW	0x0	V 相过零点到高电平中断使能 0：关闭 1：使能
2	PHVZCL	RW	0x0	V 相过零点到低电平中断使能 0：关闭 1：使能
1	PHUZCH	RW	0x0	U 相过零点到高电平中断使能 0：关闭 1：使能
0	PHUZCL	RW	0x0	U 相过零点到低电平中断使能 0：关闭 1：使能

©2025 旋智电子科技（上海）有限公司

表 15-13：PHCOMP 控制寄存器（PHCOMPCTL）位段定义

PHCOMPCTL (Phaes Comparison Control Register) Offset：0x14 Default：0x00000420
Access：PHCOMPx->PHCOMPCTL

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED		PWM3BLANK	PWM2BLANK	PWM1BLANK	PWM0BLANK	MODE	HYSTSEL
15	14	13	12	11	10	9	8
HYSTSEL	PHREFSEL			PHWSEL			PHWEN
7	6	5	4	3	2	1	0
PHVSEL			PHVEN	PHUSEL			PHUEN

表 15-14：PHCOMP 控制寄存器（PHCOMPCTL）位段描述

位段	位段名	属性	复位值	描述
31:22	RESERVED_31_22	RO	0x0	保留
21	PWM3BLANK	RW	0x0	PWM3 翻转触发消影 0：Disable 1：Enable
20	PWM2BLANK	RW	0x0	PWM2 翻转触发消影 0：Disable 1：Enable
19	PWM1BLANK	RW	0x0	PWM1 翻转触发消影 0：Disable 1：Enable
18	PWM0BLANK	RW	0x0	PWM0 翻转触发消影 0：关闭 1：使能
17	MODE	RW	0x0	工作模式 0：EMF 感应模式 1：普通电压比较器模式
16:15	HYSTSEL	RW	0x0	延滞选择 00：0 mV 01：12 mV 10：24 mV 11：36 mV
14:12	PHREFSEL	RW	0x0	参考源选择 000：中性点感应信号 001：DAC1 输出 010：DAC2 输出 011：ANA_IN0 引脚输入 100：ANA_IN1 引脚输入 101：ANA_IN2 引脚输入

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				110：ANA_IN3 引脚输入 111：ANA_IN4 引脚输入
11:9	PHWSEL	RW	0x2	PHCOMP 的 W 相输入选择 000：U 相感应信号 001：V 相感应信号 010：W 相感应信号 011：ANA_IN0 引脚输入 100：ANA_IN1 引脚输入 101：ANA_IN2 引脚输入 110：ANA_IN3 引脚输入 111：ANA_IN4 引脚输入
8	PHWEN	RW	0x0	PHCOMP 的 W 相比较使能 0：关闭 1：使能
7:5	PHVSEL	RW	0x1	PHCOMP 的 V 相输入选择 000：U 相感应信号 001：V 相感应信号 010：W 相感应信号 011：ANA_IN0 引脚输入 100：ANA_IN1 引脚输入 101：ANA_IN2 引脚输入 110：ANA_IN3 引脚输入 111：ANA_IN4 引脚输入
4	PHVEN	RW	0x0	PHCOMP 的 V 相比较使能 0：关闭 1：使能
3:1	PHUSEL	RW	0x0	PHCOMP 的 U 相输入选择 000：U 相感应信号 001：V 相感应信号 010：W 相感应信号 011：ANA_IN0 引脚输入 100：ANA_IN1 引脚输入 101：ANA_IN2 引脚输入 110：ANA_IN3 引脚输入 111：ANA_IN4 引脚输入
0	PHUEN	RW	0x0	PHCOMP 的 U 相比较使能 0：关闭 1：使能

SPINTROL

©2025 旋智电子科技（上海）有限公司

表 15-15：PHCOMP 消影寄存器（PHCOMPBLANK）位段定义

PHCOMPBLANK (Phase Comparison Blanking Control) Offset：0x18 Default：0x00000000 Access：PHCOMPx->PHCOMPBLANK							
31	30	29	28	27	26	25	24
WINSIZE							
23	22	21	20	19	18	17	16
WINSIZE							
15	14	13	12	11	10	9	8
OFFSET							
7	6	5	4	3	2	1	0
OFFSET							

表 15-16：PHCOMP 消影寄存器（PHCOMPBLANK）位段

位段	位段名	属性	复位值	描述
31:16	WINSIZE	RW	0x0	消影窗口大小
15:0	OFFSET	RW	0x0	消影窗口与 PWM 翻转点的距离

表 15-17：PHCOMP 滤波寄存器（PHCOMPFILTER）位段定义

PHCOMPFILTER (Phase Comparison Filter Control) Offset：0x1C Default：0x00000000 Access：PHCOMPx->PHCOMPFILTER							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED				FLTTH			
15	14	13	12	11	10	9	8
FLTTH	FLTWIN					FLTDIV	
7	6	5	4	3	2	1	0
FLTDIV							

©2025 旋智电子科技（上海）有限公司

表 15-18：PHCOMP 滤波寄存器（PHCOMPFILTER）位段描述

位段	位段名	属性	复位值	描述
31:20	RESERVED_31_20	RO	0x0	保留
19:15	FLTTH	RW	0x0	设置滤波器输出翻转的阈值为（FLTTH+1）
14:10	FLTWIN	RW	0x0	设置滤波器的滤波窗口为（FLTWIN+1）
9:0	FLTDIV	RW	0x0	设置滤波器时钟分频为（FLTDIV+1）

表 15-19：PHCOMP 有限状态机复位寄存器（PHCOMPRST）位段定义

PHCOMPRST (Phase Comparison FSM Reset Register)　Offset：0x20　Default：0x00000000
Access：PHCOMPx->PHCOMPRST

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED					PHWRST	PHVRST	PHURST

表 15-20：PHCOMP 有限状态机复位寄存器（PHCOMPRST）位段描述

位段	位段名	属性	复位值	描述
31:3	RESERVED_31_3	RO	0x0	保留
2	PHWRST	W1C	0x0	复位 PHCOMP 的 W 相状态机 在 BEMF 模式下，第一个过零点事件在复位之后将被忽略。 0：写 0 无效，读总为 0 1：写 1 清除中断标志，本位段自动清零
1	PHVRST	W1C	0x0	复位 PHCOMP 的 V 相状态机 在 BEMF 模式下，第一个过零点事件在复位之后将被忽略。 0：写 0 无效，读总为 0 1：写 1 清除中断标志，本位段自动清零
0	PHURST	W1C	0x0	复位 PHCOMP 的 U 相状态机 在 BEMF 模式下，第一个过零点事件在复位之后将被忽略。 0：写 0 无效，读总为 0 1：写 1 清除中断标志，本位段自动清零

©2025 旋智电子科技（上海）有限公司

表 15-21：PHCOMP 模块写使能寄存器（PHCOMPREGKEY）位段定义

PHCOMPREGKEY (Phase Comparison Register Write-Allow Key Register) Offset：0x24 Default：0x1ACCE551 Access：PHCOMPx->PHCOMPREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

表 15-22：PHCOMP 模块写使能寄存器（PHCOMPREGKEY）位段描述

位段	位段名	属性	复位值	描述
31:0	KEY	RW	0x1ACCE551	写入 0x1ACCE551 解锁受保护的 COMP 寄存器

SPINTROL

16 数模转换器（DAC）

16.1 概述

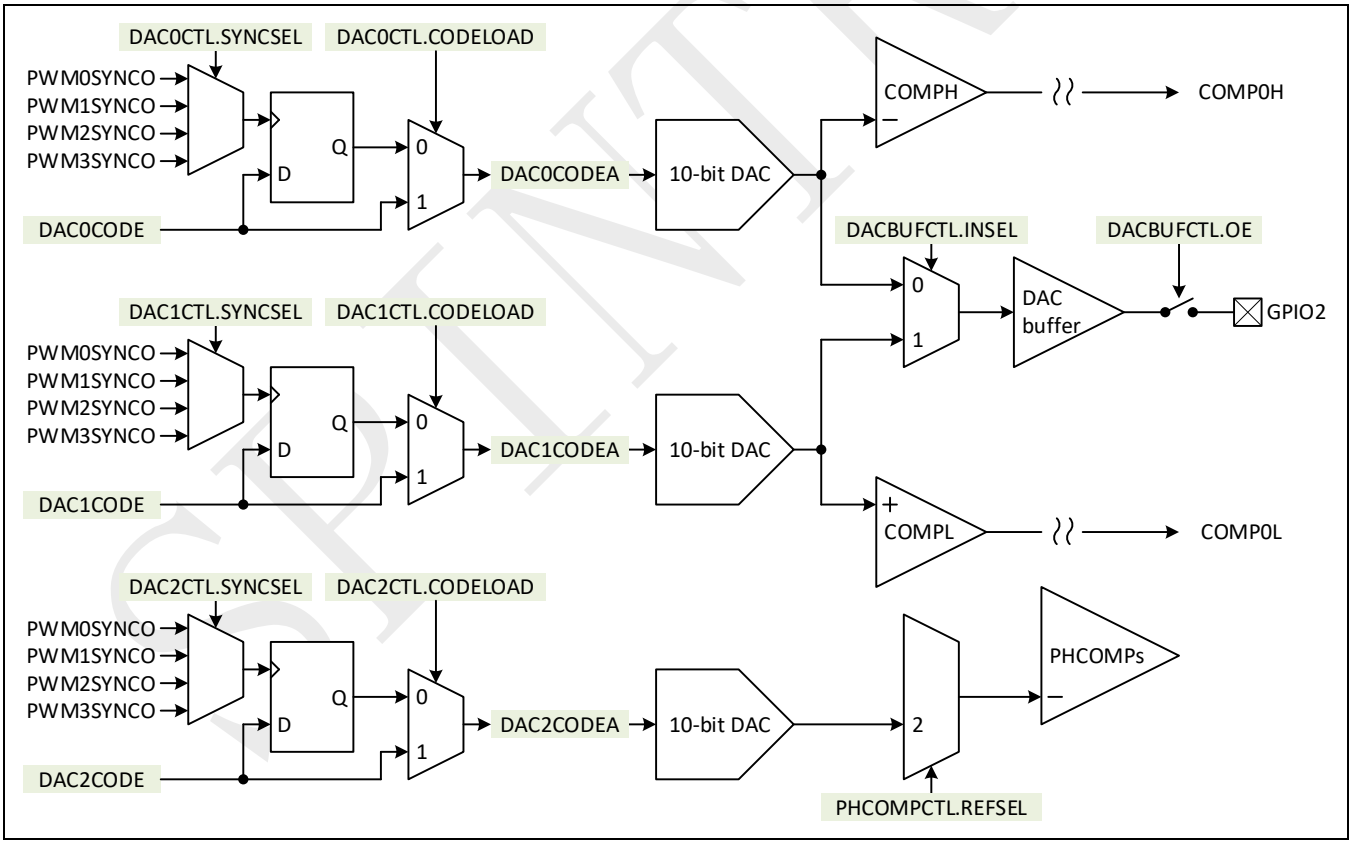
SPC1169 缓冲型 DAC 模块由三个内部的 10 位 DAC 和一个能够驱动外部负载的模拟输出缓冲器组成。如图 16-1 所示，DAC0 和 DAC1 可以被送入 DAC 缓冲器，而 DAC2 则用作相位比较器的内部参考电压。缓冲型 DAC 是一个通用的 DAC，可以配置生成直流电压和交流波形，如三角波、方波等等。每个 DAC 输出根据 DVDD33 以 10 位分辨率进行分割，对于给定 DACx 的有效码值为 DACxCODEA 时，其输出为：

$$V_{DACx} = V_{DVDD3} / 1024 * DACxCODEA$$

16.2 DAC 架构

DAC 框图如图 16-1 所示。

图 16-1：DAC 框图



©2025 旋智电子科技（上海）有限公司

缓冲 DAC 模块有两组 DAC 码值寄存器：DACxCODE 和 DACxCODEA。DACxCODEA 是一个只读的活跃码值寄存器，直接控制着缓冲 DAC 的输出数值；而 DACxCODE 是一个可写的影子寄存器，具有以下两种操作模式：

- 立即加载（DACxCTL.CODELOAD = 1）
  - ➢ 将数值写入 DACxCODE 会同时更新 DACxCODEA 的值；
- 影子加载（DACxCTL.CODELOAD = 0）
  - ➢ DACxCODEA 的值会在 DACxCTL.SYNCSEL 定义的 PWMySYNCO 事件发生时更新为 DACxCODE 的值；

比较器的高阈值和低阈值由两个 DAC 生成，详细如表 14-1 所示。

SPC1169 提供一个内部 DAC 缓冲器，每个 DAC 的输出可以根据 DAC 缓冲器的 MUX 设置（DACBUFCTL.INSEL）发送到 DAC 缓冲器中，具体设置参见表 16-1。

表 16-1：DAC 缓冲器输入选择

DACBUFCTL.INSEL	DAC buffer 输出
0	DAC0
1	DAC1

注意：DAC 缓冲器输出可以通过 ADC 输入 MUX 直接发送到 ADC（请参阅 ADC 输入选择中的：SHINSELP=8）。DAC 缓冲输出也可以发送到 ANA_IN2 引脚，这取决于设置 DACBFCTL.OE。

16.3 斜坡发生器

如图 16-2 所示，斜坡发生器可以为 COMPyH 比较器产生一个下降的斜坡参考电压。只有 DAC0 和 DAC1 支持这个功能。

图 16-2 显示了斜坡发生器图。斜坡有两组寄存器，一组是只读有源寄存器，名为 RAMPxDLYA、RAMPxDECA 和 RAMPxMAXA，可以直接控制斜坡功能。另一个是一组可写影子寄存器，称为 RAMPxDLY、RAMPxDEC 和 RAMPxMAX。只有在特定条件下，这些影子寄存器才会加载到只读寄存器，例如 RAMPEN 或 PWMSYNC 的上升沿，如图 16-2 所示。基本工作过程可描述为：

- RAMPxCNT 的最高 10 位将送到 CODEA 寄存器，用于控制 10 位 DAC 输出；RAMPxMAXA 是初始 DAC 值，RAMPxDLYA 是延迟计数器。对于每个 PWMCLK 周期，RAMPxDLYA 将减少一，直到达到零。每当 RAMPxDLYA 减少到零，RAMPxCNT 计数器以 RAMPxDECA 为步长递减计数；

详细其工作原理如下：

- 通过 DACxCTL.RAMPEN 使能斜坡发生器；
- 触发初始化加载影子寄存器 RAMPxDLY、RAMPxDEC、RAMPxLOAD 到对应的有效值寄存器 RAMPxDLYA、RAMPxDECA、RAMPxLOADA；
- RAMPxCNT 寄存器将首先从 RAMPxMAX 加载，并且该寄存器保持稳定；

©2025 旋智电子科技（上海）有限公司

- 若触发 PWMSYNC，然后，对于每个 PWMCLK 周期，RAMPxDLYA 将减少一；当 RAMPxDLYA 达到 0 时，RAMPxCNT 的值将减去 RAMPxDECA；再然后，RAMPxDLYA 将在下一个 PWMCLK 周期中加载初始值并重复。也即，DAC 输出将在每个（RAMPxDLYA+1）个 PWMCLK 周期中减少 RAMPxDECA 步。
- 当 RAMPLOAD=0 时，斜坡发生器块处于影子模式。这意味着在 COMPH 被触发时，RAMPxMAX 值不会立即更新到 RAMPxMAXA 中。RAMPxCNT 将受到之前的 RAMPxMAXA 值的控制，并且最新的 RAMPxMAX 值将在下一个 PWMSYNC 上升沿时更新到 RAMPxMAXA 中；
- 如果 RAMPLOAD=1，则斜坡发生器块处于直接模式，这意味着当 COMPH 被触发时，RAMPxMAX 值将直接更新到 RAMPxMAXA 并控制 RAMPxCNT；

图 16-2：DAC 斜坡发生器框图

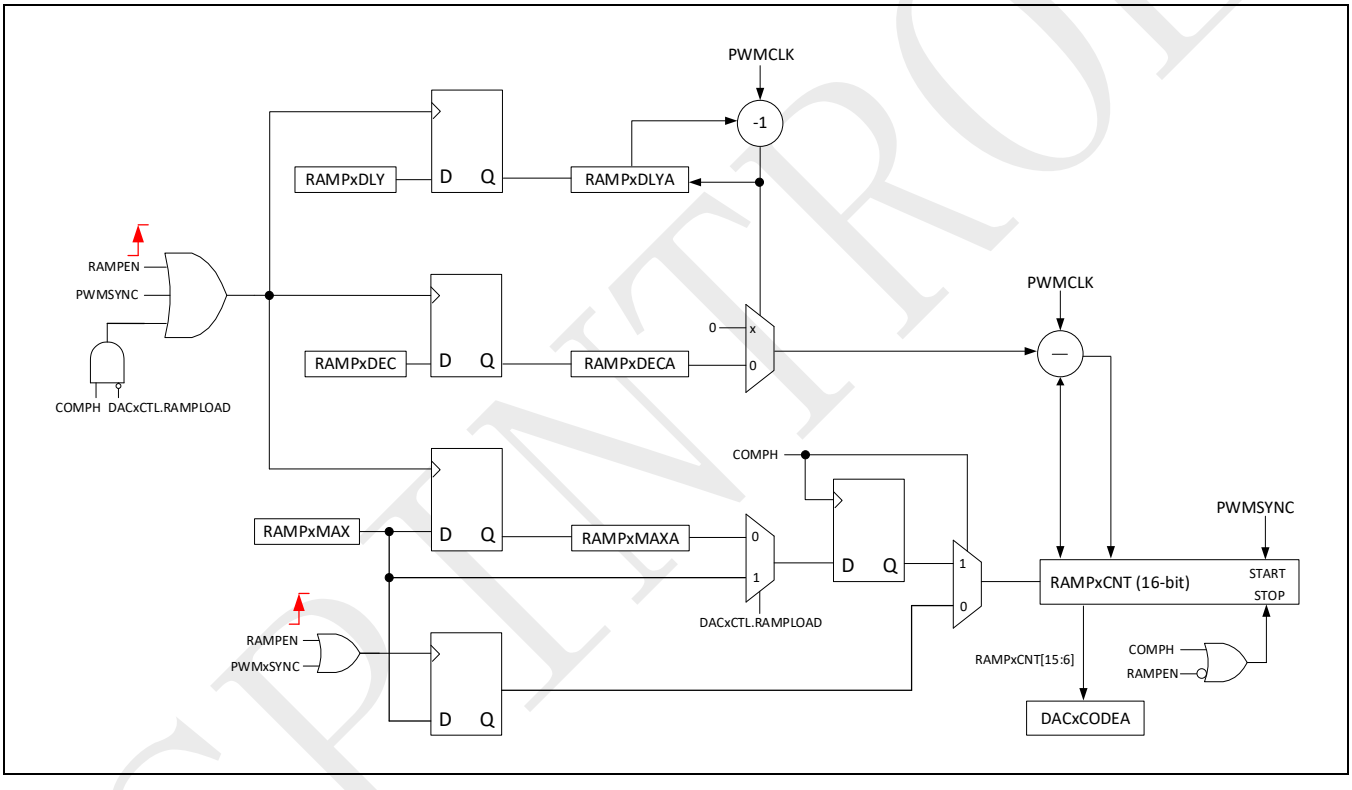
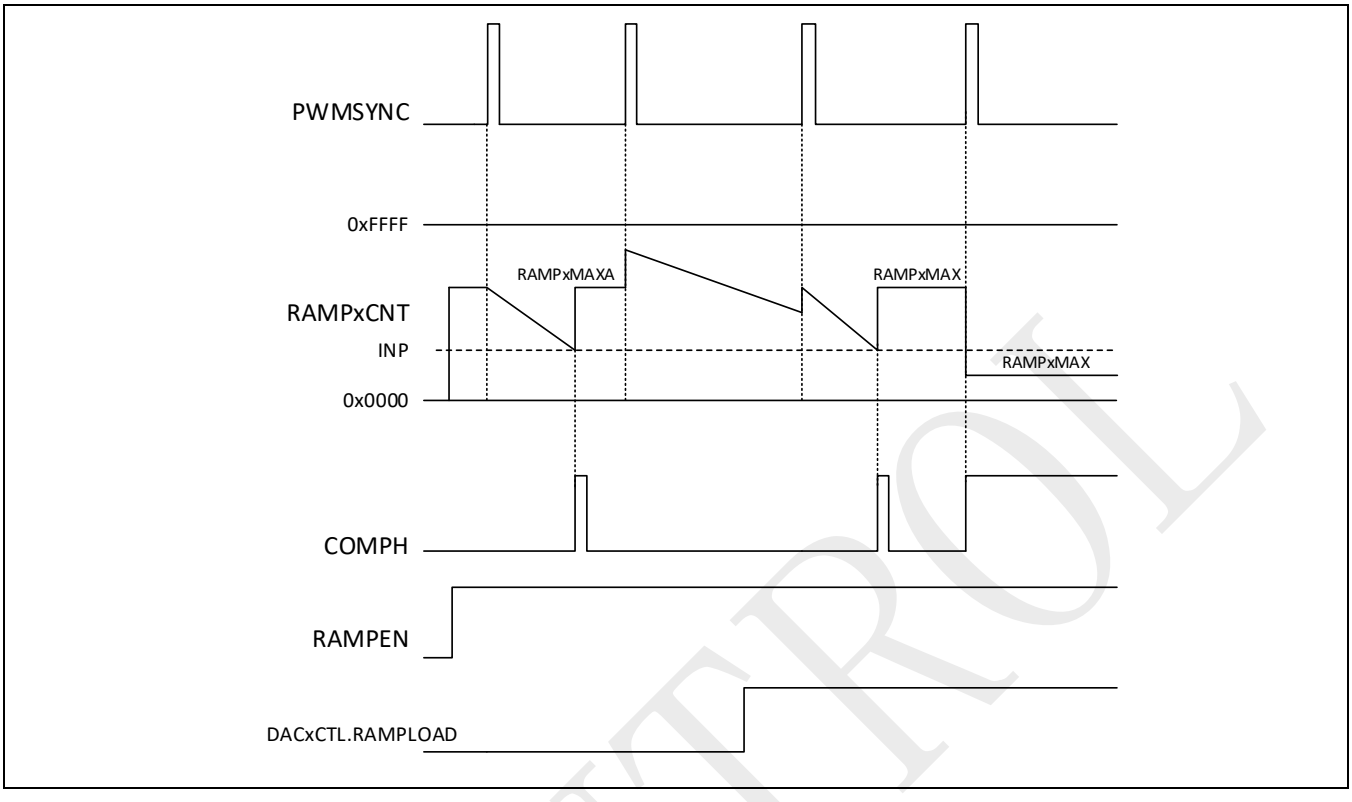


图 16-3 显示了在 PWMSYNC 触发和 COMPH 触发时的 DAC 输出（与 RAMPCNT 相同），假设过高 COMP 的正端输入是固定的（在图 16-3 中表示为 INP）。

©2025 旋智电子科技（上海）有限公司

图 16-3：斜坡发生器波形



16.4 寄存器

请参考章节 14.5 的寄存器列表及其定义。

SPINTROL

17 差分转单端缓冲器（D2S）

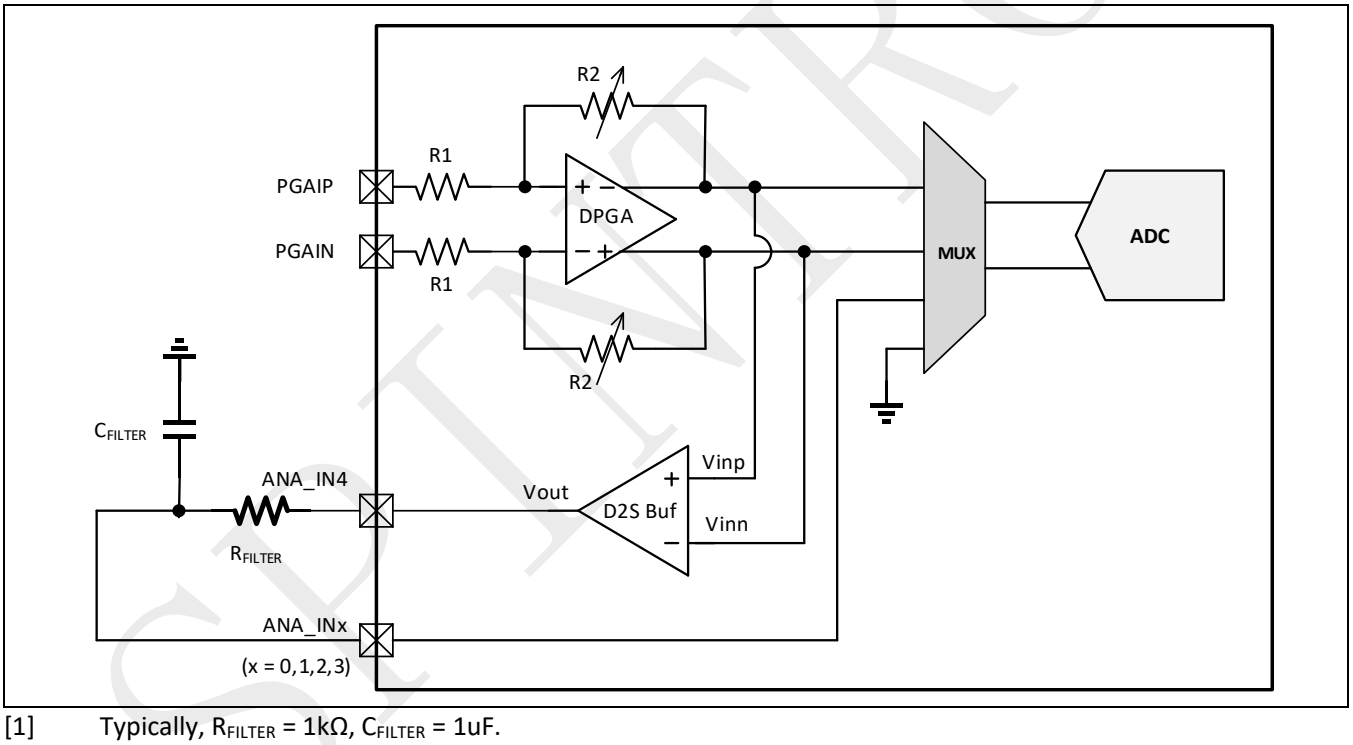
17.1 概述

SPC1169 提供一个 D2S（差分到单端）缓冲器，它可以将 DPGA 的差分输出转换为单端输出，并将其送到 ANA_IN4 引脚。通过外部的 RC 滤波器（通常，R=1kΩ，C=1uF），可以将单端输出送回到 ANA_INx（x=0~3）引脚，并通过 13 位 ADC 进行测量。此时，13 位 ADC 可以获得感应电阻的直流平均电压。

17.2 功能描述

D2S 实际应用方式如图 17-1 所示。

图 17-1：差分转单端缓冲器应用框图



[1] Typically, R_{FILTER} = 1kΩ, C_{FILTER} = 1uF.

D2S 缓冲器的增益固定为 0.5，D2S 缓冲器的输入和输出之间的方程式为：

$$Vout = (Vinp – Vinn) \times 0.5 + VDD3/2$$

通常情况下，VDD3=3.3V，因此输出电压为(Vinp-Vinn)x0.5+1.65。DS2 缓冲器的输入范围为 0.3V 至 VDD3 - 0.3V。

注意： 用户可以在 SDK 中调用 D2S_Enable()/D2S_Disable()函数来使能/禁用 D2S 缓冲器。

©2025 旋智电子科技（上海）有限公司

18 温度传感器

18.1 温度传感器概述

SPC1169 实现了一个内部温度传感器来测量设备的结温度。传感器输出可以使用 13 位 ADC 进行采样，如图 10-1 所示。

18.2 传递函数

温度传感器输出和由此产生的 ADC 值随着结温度的增加而增加。

以下等式显示了温度和 ADC 代码之间的关系：

$$\text{Temperature} = (ADC_CODE - T_{OFFSET}) * T_{SLOPE} + 25$$

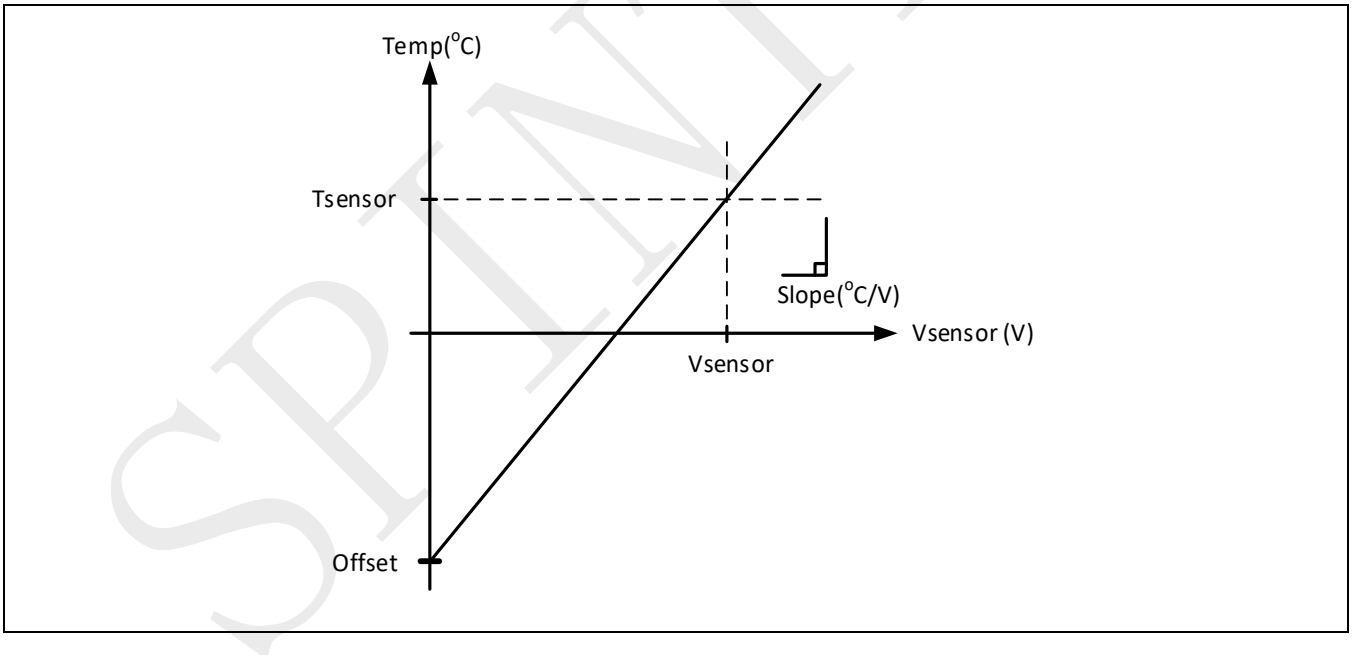
其中T_{SLOPE} = 3.808 °C，T_{OFFSET} = 81.

温度和传感器输出电压（Vsensor）之间的关系如图 18-1 所示，传感器的传递函数定义为：

$$\text{Temperature} = \text{Slope} \times \text{Vsensor} + \text{Offset}$$

其中 Slope = 4238.97 °C/V，Offset = -283.36 °C.

图 18-1：温度传感器传递函数



因此，测量温度的步骤如下所示：

1. 在目标温度 T 下，将传感器输出测量为 Vsensor。应使用 ADC 增益和偏移微调值校准 Vsensor。
2. 使用上述公式计算目标温度。

SPINTROL

18.3 寄存器

请参阅

SPINTROL

©2025 旋智电子科技（上海）有限公司

表 10-72 和表 10-73 了解 T 传感器相关寄存器的定义。

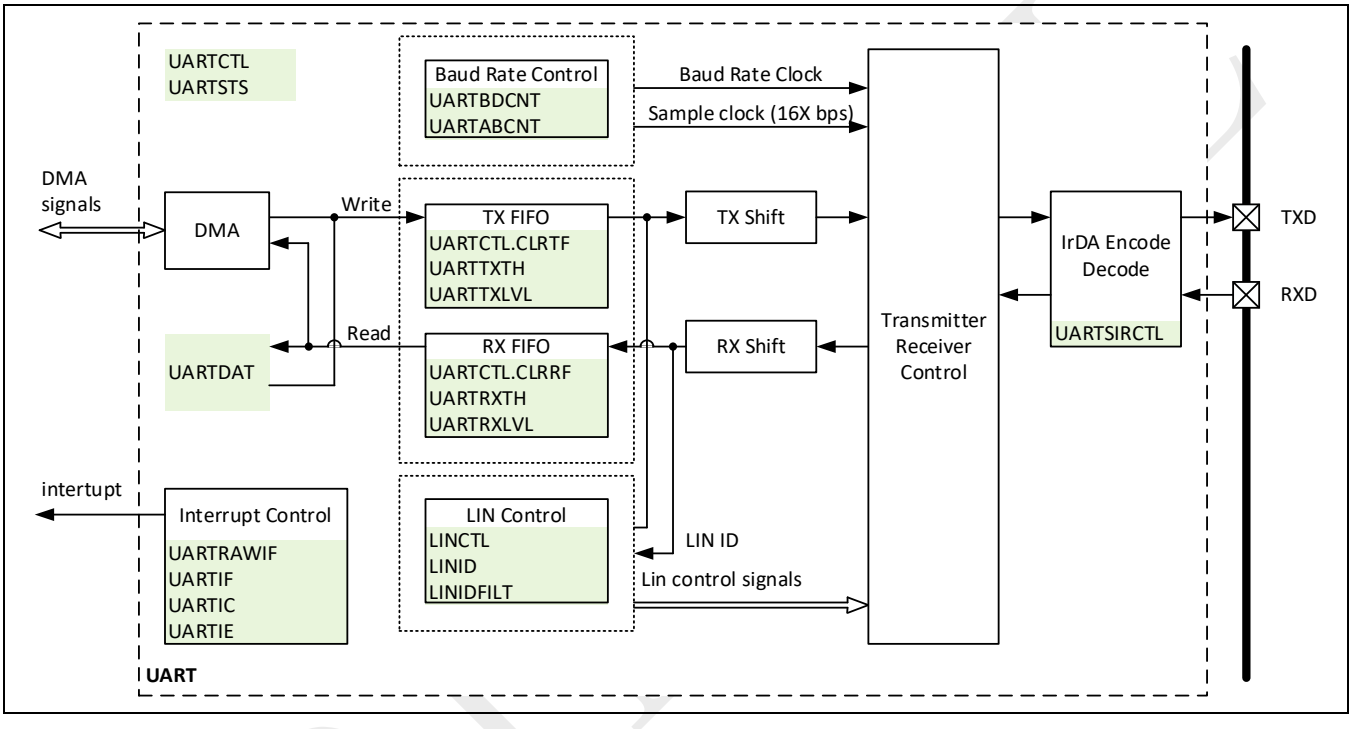
SPINTROL

19 通用异步收发器（UART）

19.1 概述

SPC1169 提供 2 个 UART 模块，该端口包含一个 UART、一个符合 IrDA 串行红外规范的慢速串行红外传输编码器和接收解码器、LIN 控制模块。方框图如图 19-1 所示。

图 19-1：UART 框图



UART 端口对从外围设备接收到的数据字节执行串行到并行转换，并对从 CPU 接收的数据字节进行并行到串行转换。

CPU 可以从 UARTSTS 读取完整的 UART 状态。状态信息包括传输操作的类型和条件以及与 UART 相关的错误条件（奇偶校验、溢出、成帧或中断）。

UART 具有一个发送 FIFO（64 字节深度）和一个接收 FIFO（64 个字节深度）。传输 FIFO 保存来自 CPU 的数据，直到数据在串行链路上传输；接收 FIFO 缓冲来自串行链路的数据，直到 CPU 读取为止。Wrtie 寄存器 UARTDAT 将向传输 FIFO 写入新数据；读取寄存器 UARTDAT 将从接收 FIFO 获得接收数据。

UART 包括一个可编程波特率生成器，该生成器可以将输入模块时钟除以 16 到（216-1），从而产生一个波特率时钟和一个采样时钟（16X bps），该采样时钟可用于驱动内部发送和接收逻辑。该模块还支持自动波特率检测。UART 端口支持 9600、19.2K、38.4K、57.6K、115.2K、230K、460K、921K、1.8M、3.6M 等波特率。

软件可以对中断进行编程以满足其要求，从而最大限度地减少处理通信链路所需的计算数量。UART 工作在由软件（轮询或中断）或 DMA 驱动的环境中。

UART 还支持 LIN 控制功能。

©2025 旋智电子科技（上海）有限公司

19.2 UART 特征

UART 特性如下：

- 独立控制的发送、接收和中断
- 可编程串行接口
  - 5 - 8 个数据位
  - 偶、奇或无奇偶检测
  - 支持 1 个、1.5 个以及 2 个停止位
  - 波特率生成最高可达 12.5 Mbps
  - 错误起始位检测
  - 中断信号发送和检测
- 自动波特率检测
- 接收数据 16X 过采样
- 不归零（NRZ）编码/解码功能
- 64 字节传输 FIFO
- 64 字节接收 FIFO
- 可编程 FIFO 触发阈值
- 完整的状态报告功能
- 生成和检测换行符的能力
- 内部诊断功能，包括：
  - 环回测试以实现通讯链路故障隔离
  - 中断、奇偶校验和成帧误差模拟
- 符合红外数据协会（IrDA）规范的串行红外异步接口
- LIN 功能支持
- DMA 支持

SPINTROL

19.3 UART 端口信号

表 19-1 罗列并描述了 UART 端口信号。它们都是数字 CMOS 电平信号，采用通道复用最终连到 GPIO 引脚。有关详细信息，请参阅 GPIO 引脚配置寄存器。

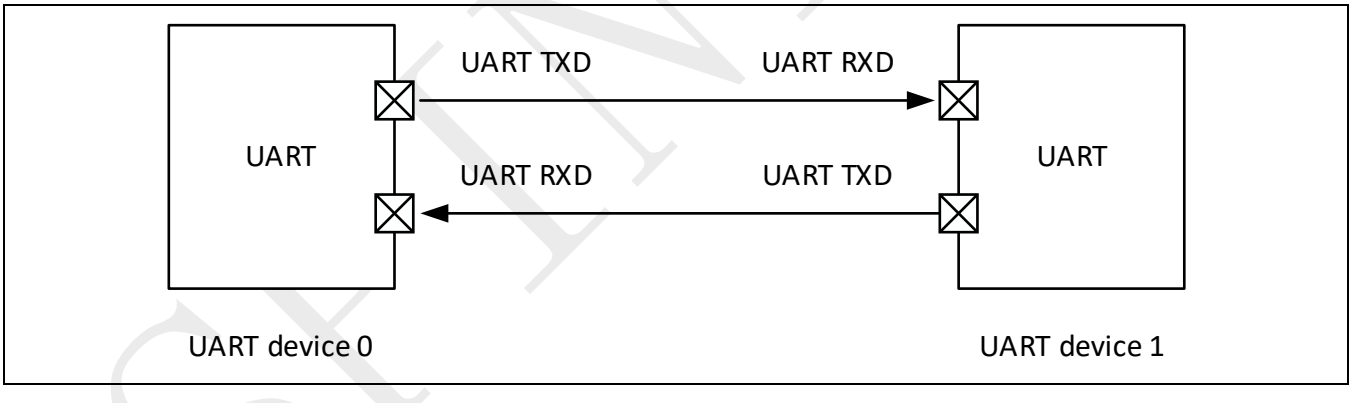
表 19-1：UART 端口信号

信号名	类型	描述
RXD	输入	**串行输入** 送入接收移位寄存器的串行数据。 – 在红外模式下，该信号连接到红外接收器解码器的输入； 在 LIN 模式下，该信号连接到 LIN 收发器的 RXD 引脚。
TXD	输出	**串行输出** 送到通信链路的串行数据，复位时该信号为 1。 – 在红外模式下，该信号连接到红外发送器编码器的输出； 在 LIN 模式下，该信号连接到 LIN 收发器的 TXD 引脚。

19.3.1 UART 设备连接

当 UART 与其他模块连接时，连接图如图 19-2 所示。

图 19-2：UART 设备连接图

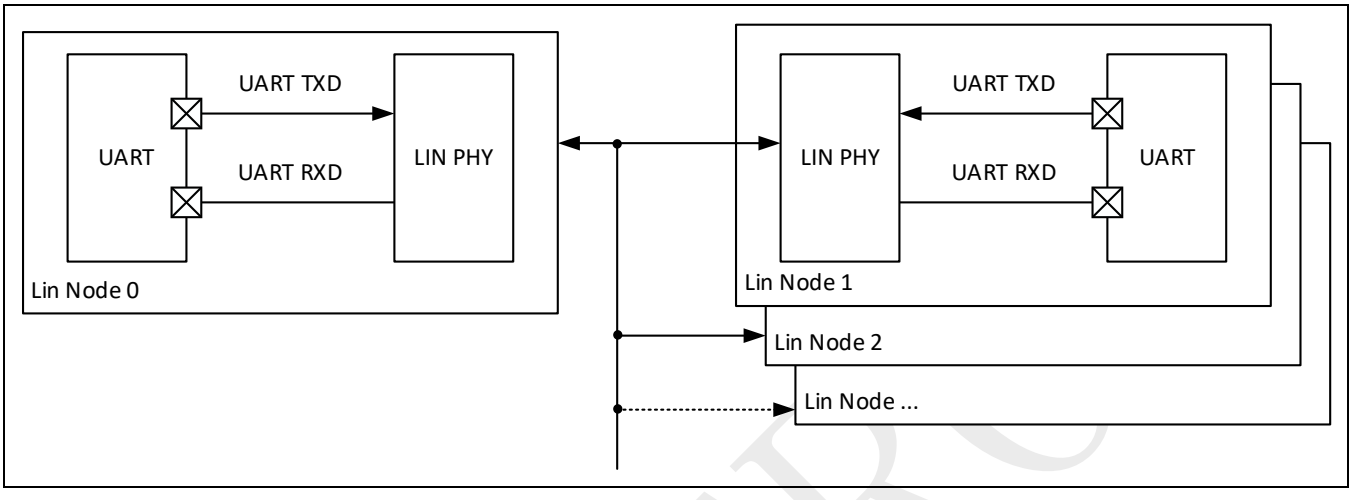


©2025 旋智电子科技（上海）有限公司

19.3.2 LIN 设备连接

当用在 LIN 模式下时，需要按照图 19-3 所示方式通过外部 LIN 收发器连接到其他 LIN 设备。

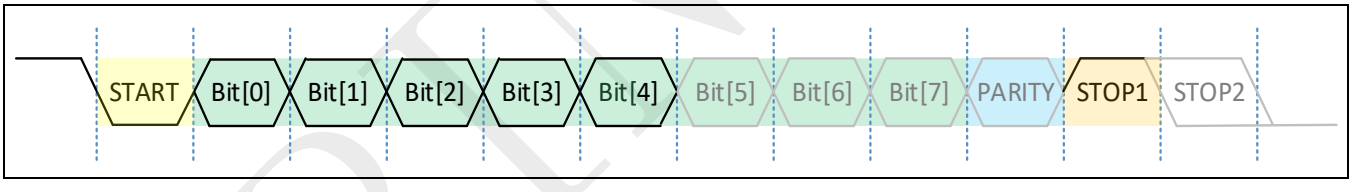
图 19-3：LIN 设备连接图



19.4 UART 操作

UART 数据帧格式如图 19-4 所示。

图 19-4：UART 帧格式



接收数据采样计数器频率是比特频率值的 16 倍（采样时钟是 16X bps）。16X 采样时钟由波特率生成器创建。每个位在在中间采样三次。图 19-4 中的阴影位可以通过软件进行编程。

每个帧的长度在 7 到 12 位之间，这取决于编程的数据位的大小、奇偶校验是否被启用以及停止位是否被编程。帧通过发送由高到低转换表示的起始比特开始。起始位之后是从最低有效位（LSB）开始的 5~8 位数据。数据位后面跟着一个可选的奇偶校验位。数据帧以 1/1.5/2 停止位结束。

如果启用了帧奇偶校验，则在以下情况下设置帧奇偶校验位：

- 偶数奇偶校验被启用并且数据字节具有奇数个 1；
- 或者奇数奇偶校验被启用并且数据字节具有偶数个 1；
- 或者将奇偶校验位设置为粘性高。

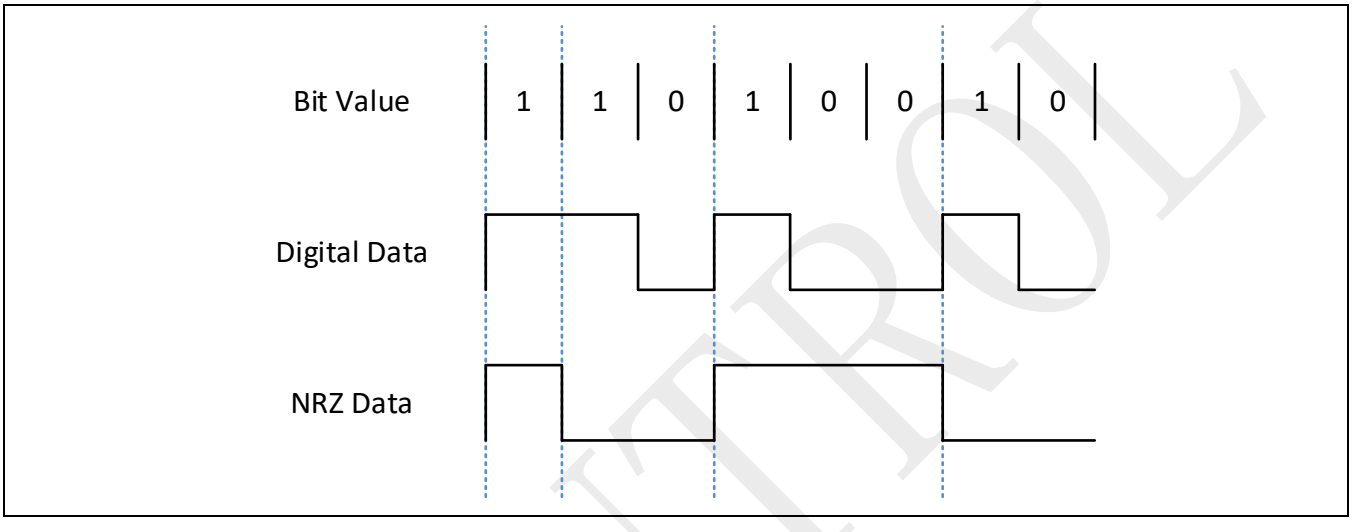
当 UARTCTL.STOPBIT 被设置为 2 位停止，UARTCTL.DATABIT 被设定为 5 位数据时，实际停止位宽为 1.5 位宽。

©2025 旋智电子科技（上海）有限公司

UART 模块有两个 FIFO：一个发送，一个接收。传输 FIFO 深度为 64 字节，宽度为 8 位。接收 FIFO 深度为 64 字节，宽度为 11 位。三位用于跟踪错误，在读取接收 FIFO 时将反映在 UARTSTS 中。读取 UARTDAT 将只获得另外 8 位数据。

UART 可以使用 NRZ 编码来表示各个比特值。要启用 NRZ 编码，请设置 UARCTL.NRZMODE 位。位值 1 由行转换表示，0 由无行转换表示。图 19-5 显示了 NRZ 编码中的数据字节 0b0100_1011。首先传输字节中的 LSB。

图 19-5：UART 数据 NRZ 编码



19.4.1 复位

UART 模块复位后处于关闭状态。若要使能 UART，用户需要先通过 PINMUX 相关寄存器配置 I/O 的功能，然后再对 UARTCTL.TXEN 和 UARTCTL.RXEN 写 1。UART 使能后，接收器等待帧的起始位，发送器则在发送 FIFO 非空时开始传输。

当清除 UARTCTL.TXEN 和 UARTCTL.RXEN 以关闭 UART 时，接收器和发送器会完成当前帧的传输后才停止收发新的数据。UART 关闭期间，FIFO 中的数据会保留，且发送 FIFO 中的遗留数据或新写入的数据会使得 UART 再次使能后立刻开始发送。

19.4.2 FIFO 操作

UART 有一个发送 FIFO 和一个接收 FIFO，每个 FIFO 保存 64 字节的数据。有两种将数据移入和移出 FIFO 的方法：直接访问 UARTDAT 和通过 DMA 访问 UARTDAT。

19.4.2.1 FIFO polling mode operation

CPU 直接访问 UARTDAT 被称为 FIFO 操作的轮询模式。在轮询模式下，软件通过 UARTSTS、UARTRAWIF 检查接收机和发射机状态。CPU 轮询接收和发送数据服务的以下位：

- 接收数据服务-CPU 检查 UARTRAWIF.RXREQ，当一个或多个字节保留在接收 FIFO 中时设置该值。
- 传输数据服务-CPU 检查 UARTRAWIF.TXREQ，该值在变送器需要数据时设置。

©2025 旋智电子科技（上海）有限公司

- CPU 还可以检查 UARTS.TXEMPTY，这是在传输 FIFO 为空时设置的。

有关更多状态或原始标志位函数，请参阅寄存器。

19.4.2.2 FIFO DMA mode operation

在这种模式下，FIFO 由 DMA 控制。

当传输 FIFO 数据请求事件发生时，DMA 需要将数据移动到传输 FIFO；当接收 FIFO 数据请求事件发生时，DMA 需要读取数据并移动到 usr 指定的内存空间。

有关 DMA 的更多知识，请参阅 DMA 控制章节。

当超时发生时，usr 需要清除尾随字节，并需要根据传输链的 DMA 末端（UARTRAWIF.DMAEOC）标志执行这些操作。

19.4.3 接收超时和删除尾字节

在接收到一个数据之后规定的时间窗口内如果没有新的数据接收，则产生超时事件。时间窗口大小为 UARTTOTH 个波特周期。这通常代表传输异常，此时接收 FIFO 内的数据被称作尾字节，存在数据完整性的问题。在接收超时中断服务程序中，用户需要参考如下步骤处理超时和尾字节：

- 往 UARTIE.RXTIMEOUT 位段写 0 以暂时关闭接收超时中断；
- 查询 UARTSTS.RXNE，若为 1 则通过 UARTDAT 读取接收 FIFO 内的尾字节，不断重复本步骤直到 UARTSTS.RXNE 为 0；
- 往 UARTIE.RXTIMEOUT 位段写 1 以重新使能接收超时中断。

19.4.4 波特率配置

UART 包含一个可编程波特率生成器，该生成器可以采用固定的输入时钟并将其分频以生成首选波特率。波特率是通过采用 UART 外围时钟（CLK_UART）来计算的。

波特率生成器将生成波特率时钟和用于采样 RXD 数据的采样时钟。采样时钟为 16X bps 时钟。UARTBDCNT 以 16 位二进制格式存储除数。在初始化期间加载这些除数锁存器，以确保波特率生成器正常工作。UARTBDCNT 值不能设置为低于 16，如果 CPU 写入值低于 16，硬件将强制设置为 16。移入或移出 UART 的数据的波特率由以下公式给出：

$$\text{Baud rate} = f_{CLK_UART} / \text{UARTBDCNT}$$

例如：如果 Divisor 为 1302，UART 时钟为 50 MHz，则波特率为 38402 bps 例如：如果 Divisor 为 1301，UART 频率为 50 MHz 时，则波特率是 38402 bps

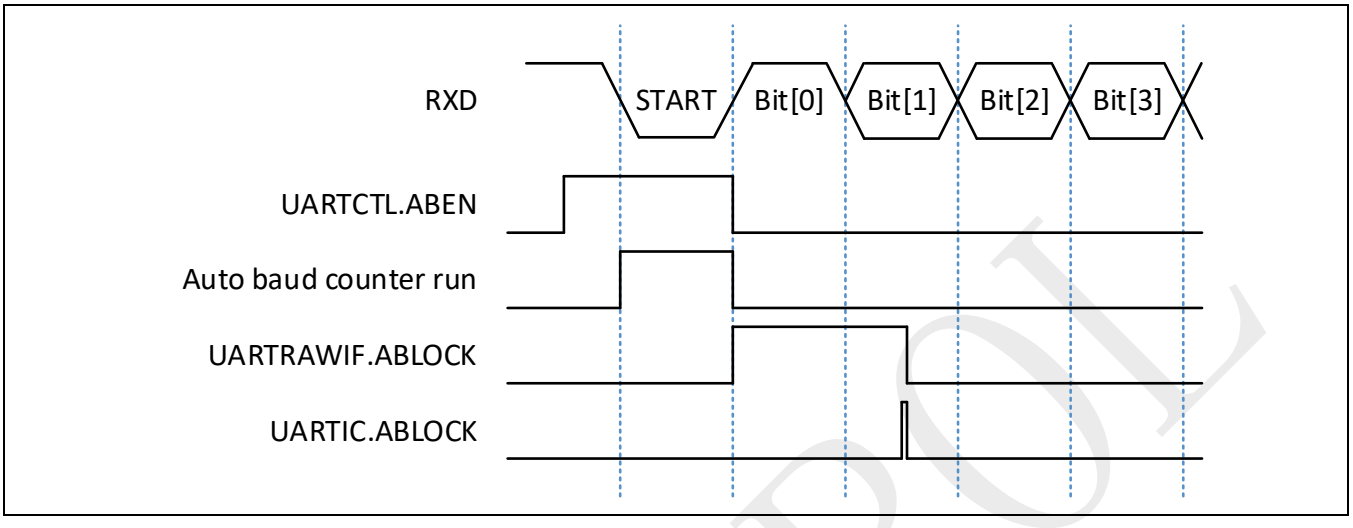
---

注意：
1. 在传输过程中，用户不应当改写 UARTBDCNT 寄存器；
2. 若每个帧长为 n 位，则用户需要保证计算得到的波特率和期望的波特率之间的相对误差在±1/(2n-1)范围内。

©2025 旋智电子科技（上海）有限公司

19.4.5 自动波特率检测

图 19-6：自动波特率侦测



如图 19-6 所示，当启用（UARTCTL.ABEN）时，UART 计数起始位脉冲内的时钟周期数（CLK_UART）。然后将该数字写入 UARTABCNT 和 UARTBDCNT（UARTCTL.ABUP 应为 0）。接收到的字节的第一位应该是 1。

当 UARTABCNT 由硬件写入时，会生成自动波特锁定中断（如果启用），并设置 UARTRAWIF.ABLOCK。如果首选，CPU 可以读取 UARTABCNT，并使用此信息以 CPU 计算的波特率对 UARTBDCNT 进行编程（UARTCTL.ABUP 为 1）。波特率编程后，CPU 验证预定字节（通常为 AT 或 AT）是否被正确接收。

当检测到波特率时，自动波特率电路通过清除 UARTCTL.ABEN 来禁用自身。要重新启用自动波特率检测，请再次设置 UARTCTL_ABEN 字段。

注意：
1. 在正在传输或接收数据时，不允许更改波特率。
2. 自动波特率检测不支持 IrDA（红外串口）模式或 LIN 模式。

©2025 旋智电子科技（上海）有限公司

19.4.6 串行红外异步接口

慢速串行红外（SIR）接口与 UART 一起使用，以支持使用红外传输的双向无线通信。SIR 接口提供发送编码器和接收解码器，以支持符合 IrDA 串行红外规范的物理链路。

SIR 接口不包含实际的 IR LED 驱动器或接收放大器。连接到 SIR 接口的 I/O 引脚只有数字信号。SIR 支持双向通信，不使用全双工通信，因为发射 LED 的反射将进入接收器。

设置 UARTSIRCTL.TXSIREN，UARTSIRCTL.RXSIREN 将分别启用 TXD、RXD 红外编码或解码功能。

注意： 只有当 UART 模块完全空闲和禁用时，才能打开和关闭安全气囊系统接口。动态更改可能会产生问题，因此不建议这样做。
UART 默认设置激活电平为帧字节时间的 3/16 位，如果默认激活驱动时间不合适，则设置 UARTSIRCTL.XMODE、UARTSIRCTL TXPW。

该编码方案在每个零值位的中间发送一个 3/16 比特宽的脉冲，对于值为 1 的位不发送脉冲。每个零值位的脉冲必须出现，即使对于它们之间没有边缘的连续位也是如此。图 19-7 显示了发送和接收操作的示例。

图 19-7：红外发送和接收

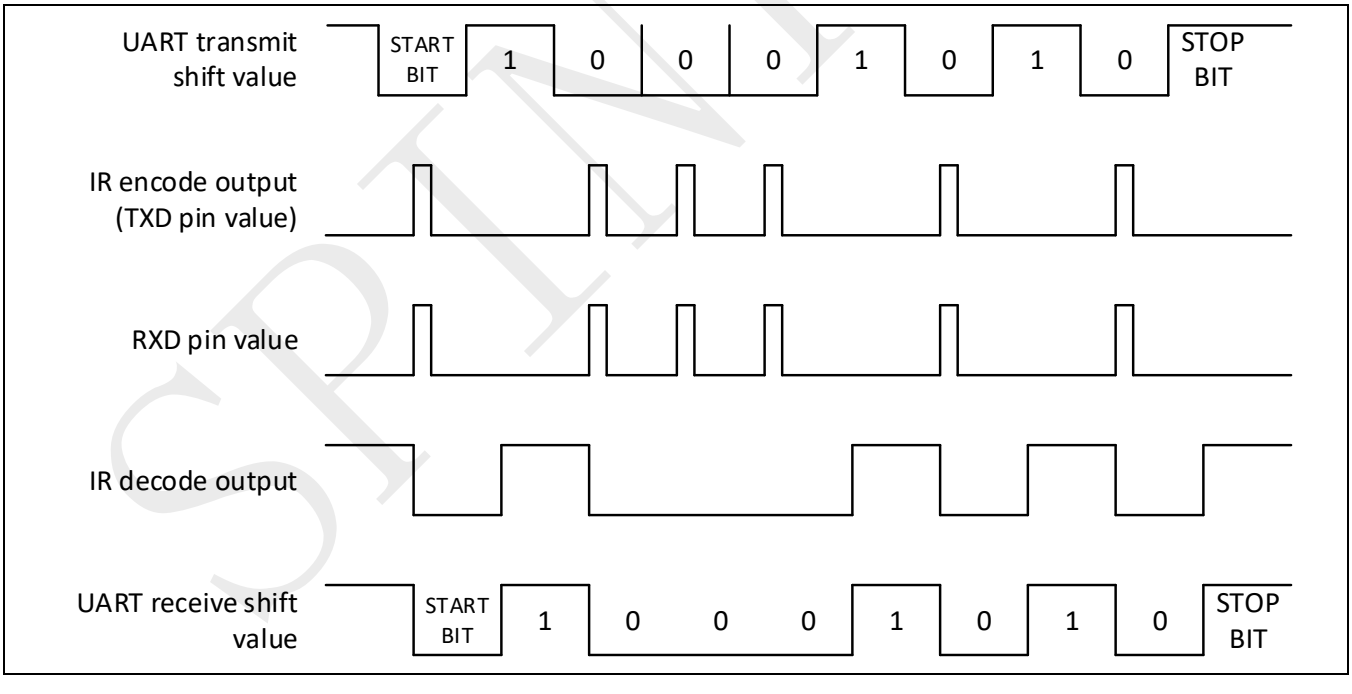


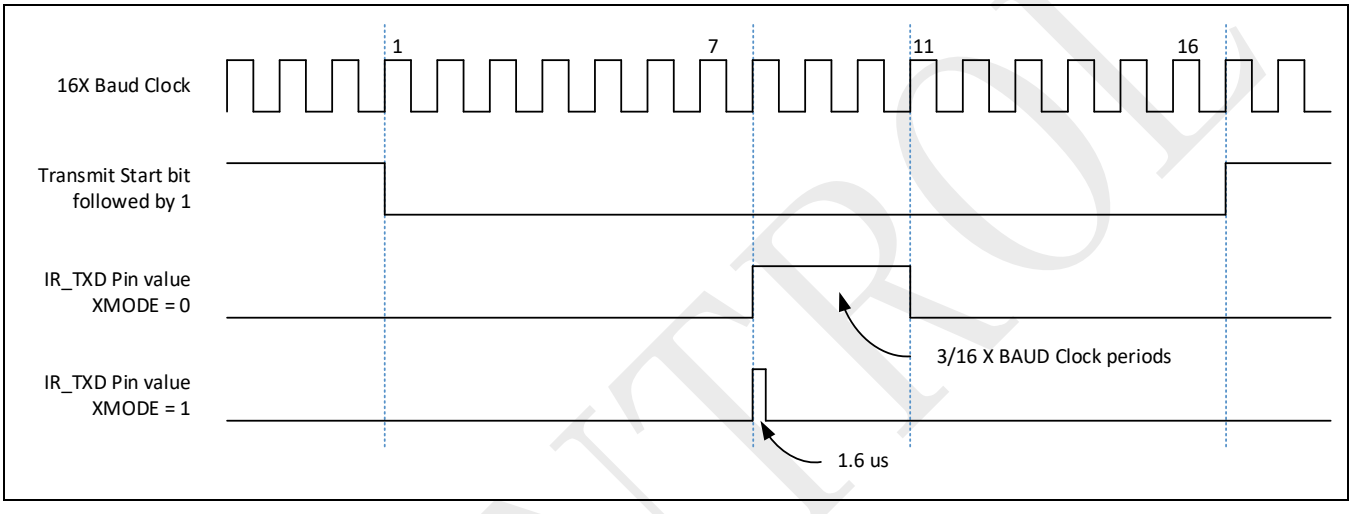
图 19-7 中的顶行显示了从 UART 发送的异步传输。第二行显示了 IR 编码器在 TXD 引脚处产生的脉冲。在 START 位和任何为零的数据位的中间生成 3/16 位脉冲。第三行显示在 RXD 输入引脚处接收到的值。第四行显示接收解码器的输出。接收解码器在检测到脉冲时将接收数据线驱动为低电平。底线显示 UART 接收器如何中断解码器操作。最后一行与第一行相同，但它被移动了半个位周期。

©2025 旋智电子科技（上海）有限公司

建议为传输脉冲宽度设置 UARTISR.XMODE。当 UARTISR.XMODE 清除时，每个零位的脉冲宽度为位时间的 3/16。UARTISR.XMODE 字段必须设置为 0x1，以便在每个零位的中间生成指定时间的脉冲（UARTSIRCTL.TXPW）。UARTSIRCTL.TXPW 的最小值为 1，低于 1 的写入值将被硬件重置为 1。设置 UARTISR.XMODE 字段时产生的较短红外脉冲可降低 LED 功耗。在 2400bps 时，对于传输的每个零位，LED 正常亮起 78us。当设置 UARTISR.XMODE 字段时，UARTSIRCTL.TXPW 为 80，CLK_UART 为 50MHZ，LED 仅亮 1.6us。

图 19-8 显示了 XMODE 操作的一个示例。

图 19-8：XMODE 示例



为了防止发射器 LED 反射反馈到接收器，当 IR 发射编码器发射数据时禁用 IR 接收解码器，当 IR 接收解码器接收数据时禁用红外发射编码器接收解码器。

注意： 这一限制具体是为了确保没有反馈；不执行过滤以去除该反馈数据。

19.5 LIN 操作

局域互联网络（LIN）是针对低成本汽车电子网络提出的概念，是对现有汽车多路复用网络组合的补充。UART 模块在 UART 功能基础上增加了一些额外的硬件以支持 LIN 协议，包括：

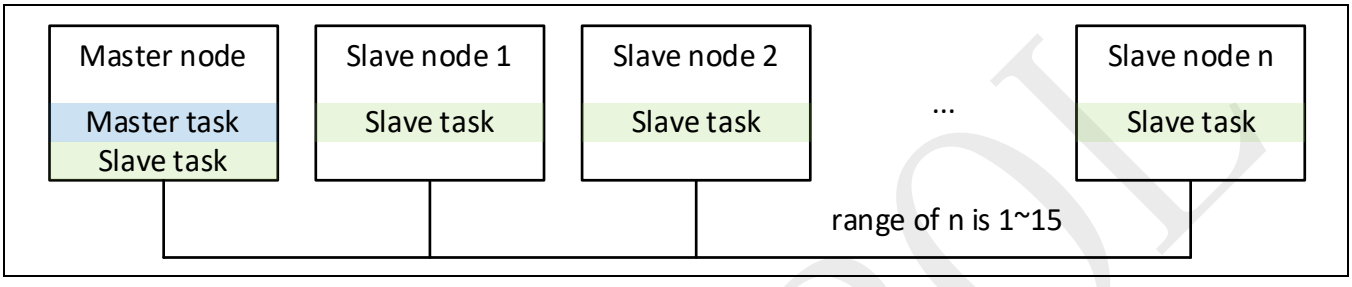
- 支持主机或从机模式；
- 硬件发送帧头；
- 同步间隔段长度可配；
- 同步间隔段间隔符长度可配；
- 受保护 ID（PID）校验位自动运算；
- 支持 LIN PID 接收匹配筛选；
- 支持 LIN 位错误、校验和错误、ID 匹配等中断；

©2025 旋智电子科技（上海）有限公司

19.5.1 LIN 协议

如图 19-9 所示，LIN 的拓扑结构为单线总线，采用了单一主机多从机的概念。总线电平为 12V，传输位速率最高为 20Kbps。由于物理层限制，一个 LIN 网络最多可以连接 16 个节点，典型应用一般都在 12 个节点以下，主机节点有且只有一个，从机节点有 1 到 15 个。主机节点包含主机任务和从机任务，从机节点只包含从机任务。

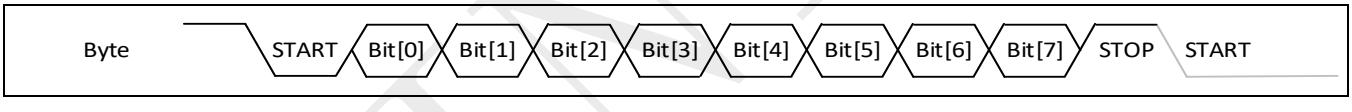
图 19-9：LIN 总线拓扑图



19.5.2 帧的结构

LIN 字节字段定义为 1 个起始位、8 个数据位和 1 个停止位。并且没有奇偶校验位。如图 19-10 所示。

图 19-10：LIN 字节字段

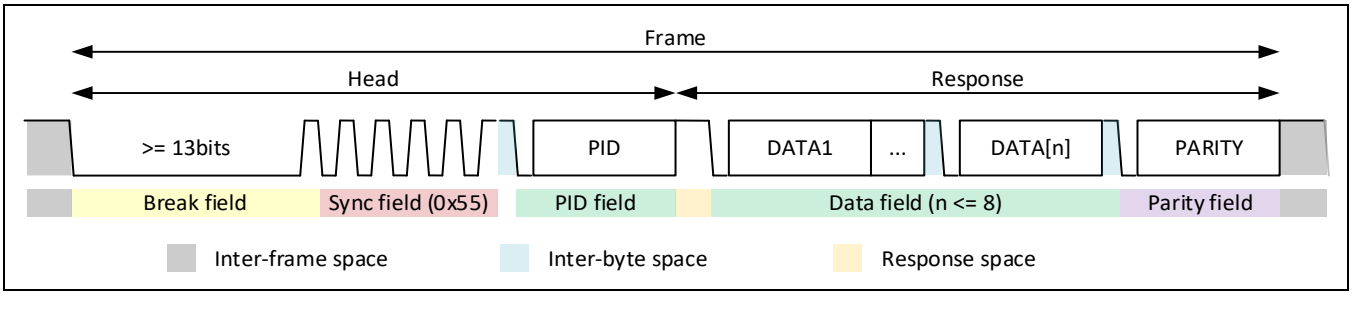


UART LIN 可以配置为主节点或从节点（LINCTL.master），主节点可以执行从任务。

LIN 框架和传输的定义如图 19-11 和图 19-12 所示。主任务发送帧头，从任务发送响应，LIN 框架由框架头和响应组成。除了 break 字段外，所有其他字段都是如图 19-10 所示定义的字节字段。

主任务首先发送长度大于 13 位的中断字段（LINCTL.SBRKLEN），并以 1 个中断分隔符（长度大于 1 位（LINCTL.SBRKDEL））结束。从任务只需要检测 11 个连续的主位，并将其视为检测到的中断字段，然后等待中断分隔符。

图 19-11：LIN 帧的结构

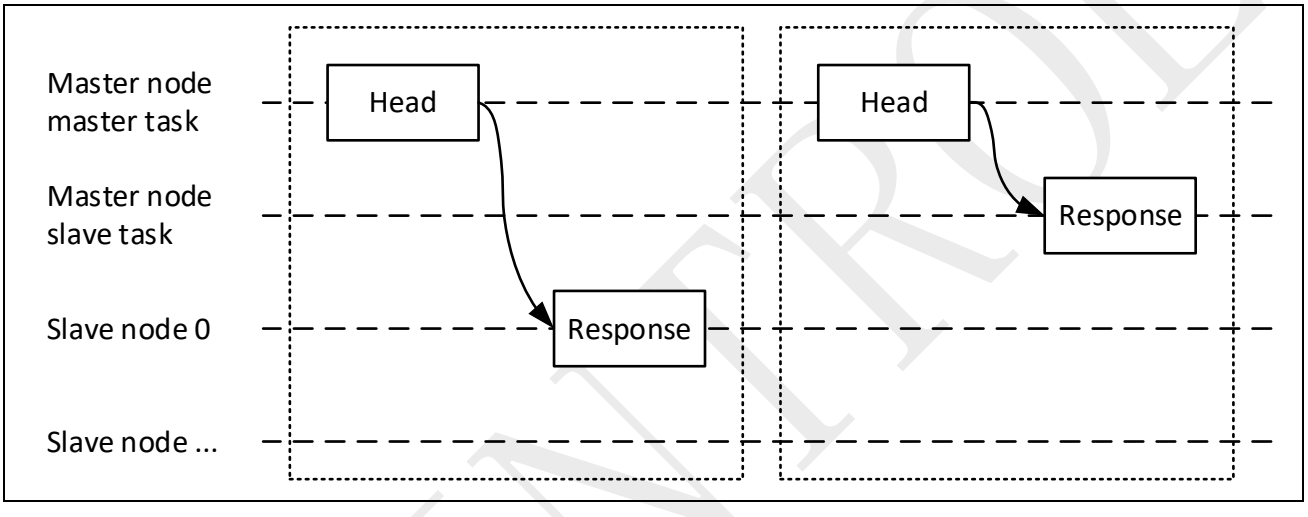


©2025 旋智电子科技（上海）有限公司

然后主任务发送同步字段（0x55）和保护 ID（PID），完成帧头发送。从机接收同步字段数据并调整其波特率（从以上描述中可知，只有正确识别出“Break filed”之后，才能进入波特率调整阶段，在此阶段中计算波特率大小的公式为：（第 7 位的下降沿时刻 - 起始位的下降沿时刻 / 8），若启用自动波特率功能，将在侦测出“Sync file”第 7 位的下降沿后生成自动波特率锁定中断，并设置 UARTRAWIF.ABLOCK），并接收 PID。然后，每个从任务基于 PID 处理帧响应，该函数由 LIN 系统设计者预定义。只有一个从任务可以发送帧响应，其他从任务可以接收该响应或什么都不做。帧响应数据长度由 LINCTL.RESPLEN 控制。

如果从任务处理响应，则需要将 LINCTL.RESP 值设置为 0x2；如果从任务只接收到其他响应，则需要将 LINCTL.RESP 值设置为 0x1；默认值为无所事事，LINCTL.RESP 为 0；

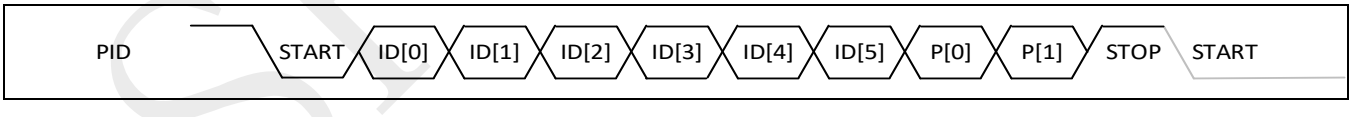
图 19-12：LIIN 传输



19.5.3 受保护 ID

LIN 保护的 ID 字节字段如图 19-13 所示。

图 19-13：LIN 保护的 ID 字节字段



为帧标识符（LINID.ID）保留六个比特，可以使用 0 到 63 范围内的值。帧标识符分为三类：

- 0 到 59（0x3B）用于信号承载帧；
- 60（0x3C）和 61（0x3D）用于携带诊断和配置数据；
- 62（0x3E）和 63（0x3F）保留用于未来的协议增强

用户可以为 ID 接收过滤器设置 LINIDFILT 寄存器。

奇偶校验位 P[1:0]由硬件在帧标识符位上计算如下公式：

$$P[0] = ID[0] \wedge ID[1] \wedge ID[2] \wedge ID[4]$$

$$P[1] = \sim(ID[1] \wedge ID[3] \wedge ID[4] \wedge ID[5])$$

©2025 旋智电子科技（上海）有限公司

19.5.4 校验和段

帧响应发送 1~8 字节的数据字段，然后发送奇偶校验字段。奇偶校验或调用的校验和可以通过两种方法（LINCTL.CHKSUM）计算：

- 标准型校验和
  - 校验对象为数据段各字节
  - 适用于诊断帧以及与 LIN1.x 从机节点通信
- 增强型校验和
  - 校验对象为数据段各字节以及受保护 ID
  - 与 LIN2.x 从机节点通信（诊断帧除外）

校验方法为将校验对象的各字节做带进位的二进制加法，每当结果大于 256 时就减去 255，并将所得最终的和逐位取反，以该结果作为校验和。

注意：	没有根据帧 ID 自动切换校验和类型的硬件支持（例如：不支持针对帧 ID 为 0x3C 或 0x3D 的帧自动切换为经典型校验和），用户需要依据主机节点发布的规则，根据帧 ID 自行通过配置 LINCTL.CHKSUM 来选择校验和的类型。

19.6 中断

UARTRAWIF 将显示中断事件原始标志，即使中断未启用。写入 UARTIC 位 1 将清除原始标志寄存器中的相关位，写入 0 无效。

UARTRAWIF.RXREQ 和 UARTRAWIF.TXREQ 在 UARTIC 中没有清除位，因为它们是由 FIFO 硬件逻辑自动清除的。

如果启用了 UARTIE 的一位，则在设置 UARTRAWIF 中的关联位时，将设置 UARTIF 的关联位。向 UARTIC 的每个位写入 1 也将清除 UARTIF 中的相关位。

19.6.1 接受中断

当接收字节期间接收帧错误或检测中断信号，并且 UARTIE 中启用了相关中断时，UART 将触发 CPU 的帧错误（UARTIF.RXFRMER）、奇偶校验错误（UARTIF.RXPRTYERR）或接收中断（UARTIF.RXBRK）中断。

然而，UARTSTS 中这些事件的状态位将始终根据读取 UARTDAT 从接收 FIFO 中新弹出的数据显示。这些状态位在 FIFO 中选通，并在读取 FIFO 时反映出来。

19.6.2 接收超时中断

当启用接收超时中断（UARTIE.RXTIMEOUT）并发生超时事件时，会发生接收超时中断。

©2025 旋智电子科技（上海）有限公司

19.6.3 波特率自动计数器锁定接收中断

当自动波特率功能被启用时，其中断被启用并检测第一个字节起始位完成，UART 将触发中断并设置 UARTIF.ABLOCK 位。

19.6.4 接受 FIFO 中断

若要发生接收 FIFO 中断，必须启用接收 FIFO 中断。

UARTIE.RXREQ 表示当接收 FIFO（UARTRXLVL）的电平等于或高于其阈值（UARTRXTH）时的接收 FIF UARTIE O 请求，UART 将向 CPU 发送中断。

UARTIE.RXOVF，UARTIE.RXUDF 表示发生接收 FIFO 访问溢出或下溢。

19.6.5 发送 FIFO 中断

当 CPU 写入 UARTDA 且传输 FIFO 已满时，传输 FIFO 具有写入传输 FIFO 溢出中断。

当传输 FIFO（UARTTXLVL）的电平低于其阈值（UARTTXTH）时，也有一个传输 FIFO 请求中断。

19.6.6 LIN 错误中断

当 UART 作为 LIN 模式工作时，将启用 LIN 位错误检查、校验和检查和 LIN ID 检查（UARTIE.LINBITERR、UARTIE.LINCSERR、UARTIE_LINIDMATCH）。

19.6.7 DMA 中断

当 DMA 完成其传输时，如果相关中断被启用，将生成链结束中断（UARTIF.DMAEOC）。

SPINTROL

©2025 旋智电子科技（上海）有限公司

19.7 寄存器

19.7.1 UART 寄存器列表

表 19-2：UART 模块基地址

外设模块	基地址
UART0	0x4000 1400
UART1	0x4000 1600

表 19-3：UART 寄存器列表

寄存器	偏移地址	描述	复位值
UARTDAT	0x0	UART 数据寄存器	0x00000000
UARTCTL*	0x4	UART 控制寄存器	0x00000000
LINCTL*	0x8	LIN 控制寄存器	0x00E00000
LINID	0xC	LIN ID 寄存器	0x00000000
LINIDFILT*	0x10	LIN ID 筛选标准寄存器	0x00000000
BUSIDLETH*	0x14	UART/LIN 线路空闲阈值寄存器	0xFFFFFFFF
UARTBDCNT*	0x18	UART 波特率计数寄存器	0x00000010
UARTABCNT	0x1C	UART 自动波特率计数寄存器	0x00000000
UARTTOTH*	0x20	UART 接收超时阈值寄存器	0x00000030
UARTRXLVL	0x24	UART 接收 FIFO 当前长度寄存器	0x00000000
UARTTXLVL	0x28	UART 发送 FIFO 当前长度寄存器	0x00000000
UARTRXTH*	0x2C	UART 接收 FIFO 长度阈值寄存器	0x00000000
UARTTXTH*	0x30	UART 发送 FIFO 长度阈值寄存器	0x00000000
UARTSIRCTL*	0x34	UART 红外控制寄存器	0x0000010C
UARTSTS	0x40	UART 状态寄存器	0x0000000C
UARTRAWIF	0x44	UART 中断原始事件标志寄存器	0x00000000
UARTIF	0x48	UART 中断标志寄存器	0x00000000
UARTIC	0x4C	UART 中断清除寄存器	0x00000000
UARTIE*	0x50	UART 中断使能寄存器	0x00000000
UARTDGCTL*	0x54	UART 接收信号消抖控制寄存器	0x00000000
UARTREGKEY	0x58	UART 模块写使能寄存器	0x1ACCE551

注意： 由*标记的寄存器仅当 UARTREGKEY=0x1ACCE551 时才可以改写。

©2025 旋智电子科技（上海）有限公司

19.7.2 UART 寄存器

表 19-4：UART 数据寄存器（UARTDAT）位段定义

UARTDAT (UART Data Register) Offset：0x0 Default：0x00000000 Access：UARTx->UARTDAT							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 19-5：UART 数据寄存器（UARTDAT）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	UART 总线上传输的数据 写本寄存器往发送 FIFO 放入数据； 读本寄存器从接收 FIFO 取出数据。

表 19-6：UART 控制寄存器（UARTCTL）位段定义

UARTCTL (UART Control Register) Offset：0x4 Default：0x00000000 Access：UARTx->UARTCTL							
31 SWAPPIN	30 RFEMODE	29 IDLECNTEN	28	27	26 RESERVED	25	24
23	22	21 RESERVED	20	19	18 SETBRK	17 CLRTF	16 CLRRF
15 STOPBIT	14 PRTYEN	13	12 PRTYBIT	11	10 DATABIT	9	8 NRZMODE
7 ABUP	6 ABEN	5 BUS32	4 LPBK	3 TXDMAEN	2 RXDMAEN	1 TXEN	0 RXEN

©2025 旋智电子科技（上海）有限公司

表 19-7：UART 控制寄存器（UARTCTL）位段描述

位段	位段名	属性	复位值	描述
31	SWAPPIN	RW	0x0	交换 TXD 和 RXD 引脚功能 0：TXD 为发送数据，RXD 为接收数据 1：RXD 为发送数据，TXD 为接收数据
30	RFEMODE	RW	0x0	接收 FIFO 对于停止位错误、校验错误、线路中断条件的报错方式 0：仅当接收 FIFO 非空且存入有错数据时报错 1：不管接收 FIFO 状态，收到错误数据就报错
29	IDLECNTEN	RW	0x0	总线空闲计数使能 0：关闭 1：使能
28:19	RESERVED_28_19	RO	0x0	保留
18	SETBRK	W1S	0x0	线路中断条件发送请求 本位段仅作用于 TXD 引脚，不影响发送 FIFO 0：写 0 无效 1：写 1 触发一次线路中断条件发送请求。 本位段在 UART 开始发送线路中断条件时自动清零。此后若在发送过程中本位段被再次写 1，则 TXD 引脚在下个字节帧继续维持低，即接收器将会看到线路上有多个字节帧时长低电平的单次线路中断条件。
17	CLRTF	W1C	0x0	清除发送 FIFO，本位段不影响发送移位寄存器 0：写 0 无效，读总是 0 1：写 1 清除发送 FIFO 并设置 TXREQ 标志位； 本位段自动清零
16	CLRRF	W1C	0x0	清除接收 FIFO，本位段不影响接收移位寄存器 0：写 0 无效，读总是 0 1：写 1 清除接收 FIFO，本位段自动清零
15	STOPBIT	RW	0x0	停止位宽度 0：字节帧有 1 个停止位 1：字节帧是 5 个数据位时有 1.5 个停止位 字节帧是 6/7/8 个数据位时有 2 个停止位
14	PRTYEN	RW	0x0	UART 模式下字节帧校验位使能 LIN 模式下接收到受保护 ID 段时校验使能 0：关闭 1：使能

SPINTROL

位段	位段名	属性	复位值	描述
13:12	PRTYBIT	RW	0x0	UART 校验位 0：偶校验 1：奇校验 2：固定 1 校验（Mark） 3：固定 0 校验（Space）
11:10	DATABIT	RW	0x0	每个字节帧的数据位宽 0：5 位数据 1：6 位数据 2：7 位数据 3：8 位数据
9:8	NRZMODE	RW	0x0	非归零码（NRZ）模式，红外模式下本位段无效 0：NRZ level，高电平为 1，低电平为 0 2：NRZ space，电平与前位相同为 1，不同为 0 3：NRZ mark，电平与前位不同为 1，相同为 0
7	ABUP	RW	0x0	自动波特率计数更新模式 0：波特计数器在自动波特率锁定时由硬件更新 1：波特计数器需要软件写 UARTBDCNT 更新
6	ABEN	W1S	0x0	自动波特率检测使能 0：关闭 1：使能
5	BUS32	RW	0x0	通过 UARTDAT 寄存器访问 FIFO 时的数据位宽 0：读写 FIFO 时仅 UARTDAT 的最低字节有效 1：读写 FIFO 时 UARTDAT 的 四个字节均有效
4	LPBK	RW	0x0	环回模式，提供如下诊断能力 (1) 发送引脚始终输出高 (2) 接收器输入与接收引脚断开 (3) 发送器的输出数据直接送到接收器的输入 (4) 支持发送器发送线路中断条件到接收器 0：正常工作模式 1：环回工作模式
3	TXDMAEN	RW	0x0	发送器 DMA 使能 0：关闭 1：使能

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
2	RXDMAEN	RW	0x0	接收器 DMA 使能 0：关闭 1：使能
1	TXEN	RW	0x0	发送器使能 0：关闭 1：使能
0	RXEN	RW	0x0	接收器使能 0：关闭 1：使能

表 19-8：LIN 控制寄存器（LINCTL）位段定义

LINCTL (LIN Control Register)　Offset：0x8　Default：0x00E00000
Access：UARTx->LINCTL

31	30	29	28	27	26	25	24
RESERVED							TXCHKSUM
23	22	21	20	19	18	17	16
CHKSUM	ATXCHKSUM	ATOTH	RFMODE	RESERVED	RESPSPACE		
15	14	13	12	11	10	9	8
RESERVED	BYTESPACE		RESPLEN				RESP
7	6	5	4	3	2	1	0
RESP	SBRKDEL		SBRKLEN			MASTER	LINEN

表 19-9：LIN 控制寄存器（LINCTL）位段描述

位段	位段名	属性	复位值	描述
31:25	RESERVED_31_25	RO	0x0	保留
24	TXCHKSUM	W1S	0x0	单次校验和段发送请求 0：写 0 无效，校验和段发送后回读是 0 1：写 1 请求发送校验和段，此时， 若数据段已经发送结束，校验和段会立即开始发送； 若数据段尚未发送结束，请求进入等待，直到数据段发送结束才开始发送校验和。 校验和段开始发送后，此位段自动清零
23	CHKSUM	RW	0x1	校验和模式 注意：不支持对任何特殊帧（例如 ID 为 0x3C 或 0x3D）的校验和模式自动切换 0：经典型校验和，对数据段各字节做带进位的模 256 加法 1：增强型校验和，对数据段各字节以及受保护 ID 做带进位的模 256 加法

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
22	ATXCHKSUM	RW	0x1	校验和段发送方式 0：通过对 TXCHKSUM 写 1 手动请求发送 1：在数据段发送结束后自动发送校验和段
21	ATOTH	RW	0x1	接收超时阈值方式 0：手动设置为 UARTTOTH 个波特 1：自动按照 LIN 标准设置，即额定帧传输时间再加 40%余量
20	RFMODE	RW	0x0	接收 FIFO 工作模式 0：数据段和校验和段都会存入 FIFO，从而有助于分析校验和段错误 1：仅当校验和段无误时才把数据段存入 FIFO
19	RESERVED_19	RO	0x0	保留
18:16	RESPSPACE	RW	0x0	设定应答间隔为 RESPSPACE 个波特
15	RESERVED_15	RO	0x0	保留
14:13	BYTESPACE	RW	0x0	设置字节间间隔为 BYTESPACE 个波特
12:9	RESPLEN	RW	0x0	应答数据字节数 0：应答字节数由 LINID[5:4]决定 3：8 字节；2：4 字节；其余：2 字节 1：1 字节 2：2 字节 3：3 字节 4：4 字节 5：5 字节 6：6 字节 7：7 字节 8：8 字节
8:7	RESP	RW	0x0	采取的应答响应 0：忽略应答部分 1：接收应答数据 2：发送应答数据 3：收发应答数据（用于调试）
6:5	SBRKDEL	RW	0x0	同步间隔段间隔符 0：1 个隐性位 1：2 个隐性位 2：3 个隐性位 3：4 个隐性位
4:2	SBRKLEN	RW	0x0	同步间隔 0：13 个显性位 1：14 个显性位 2：15 个显性位

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				3：16 个显性位 4：17 个显性位 5：18 个显性位 6：19 个显性位 7：20 个显性位
1	MASTER	RW	0x0	LIN 节点类型 0：从机节点 1：主机节点
0	LINEN	RW	0x0	LIN 模式使能 0：关闭 1：使能

表 19-10：LIN ID 寄存器（LINID）位段定义

LINID (LIN ID Register) Offset：0xC Default：0x00000000 Access：UARTx->LINID							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
ID							

表 19-11：LIN ID 寄存器（LINID）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7:0	ID	RW	0x0	LIN ID 写本位段设置要发送的受保护 ID 读本位段获取接收到的受保护 ID

表 19-12：LIN ID 筛选标准寄存器（LINIDFILT）位段定义

LINIDFILT (LIN ID Filter Register) Offset：0x10 Default：0x00000000 Access：UARTx->LINIDFILT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
MASK							

©2025 旋智电子科技（上海）有限公司

7	6	5	4	3	2	1	0
REFID							

表 19-7：LIN ID 筛选标准寄存器（LINIDFILT）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:8	MASK	RW	0x0	LIN ID 掩码 0：接收到的 ID 的对应位可以是任意值 1：要求接收到的 ID 的对应位与 REFID 一致
7:0	REFID	RW	0x0	LIN ID 比较参考值

表 19-8：UART/LIN 线路空闲阈值寄存器（BUSIDLETH）位段定义

BUSIDLETH (Bus Idle Threshold Register) Offset：0x14 Default：0xFFFFFFFF Access：UARTx->BUSIDLETH							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 19-9：UART/LIN 线路空闲阈值寄存器（BUSIDLETH）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0xFFFFFFFF	判定 UART/LIN 线路空闲时 RXD 引脚持续保持不变的 AHB 时钟周期数

表 19-10：UART 波特计数寄存器（UARTBDCNT）位段定义

UARTBDCNT (UART Baud Count Register) Offset：0x18 Default：0x00000010 Access：UARTx->UARTBDCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

©2025 旋智电子科技（上海）有限公司

表 19-11：UART 波特计数寄存器（UARTBDCNT）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x10	设定 UART 波特率为 f_{CLK_UART} / UARTBDCNT. 最小值为 16，写成一个小于 16 的值会被复位成 16

表 19-12：UART 自动波特率计数寄存器（UARTABCNT）位段定义

UARTABCNT (UART Auto-Baud Count Register) Offset：0x1C Default：0x00000000 Access：UARTx->UARTABCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 19-13：UART 自动波特率计数寄存器（UARTABCNT）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RO	0x0	检测到的一个起始位对应的 UART 时钟周期数

表 19-14：UART 接收超时阈值寄存器（UARTTOTH）位段定义

UARTTOTH (UART Timeout Threshold Register) Offset：0x20 Default：0x00000030 Access：UARTx->UARTTOTH							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
VAL							

©2025 旋智电子科技（上海）有限公司

表 19-15：UART 接收超时阈值寄存器（UARTTOTH）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7:0	VAL	RW	0x30	超时对应的波特数 最小值为 1，写 0 会被复位成 1 对于 UART 模式，在指定时间内未收到新的 数据判定为超时 对于 LIN 模式，在规定时间内未收到预期数目的应答字节和校验和则判定为超时

表 19-16：UART 接收 FIFO 当前长度寄存器（UARTRXLVL）位段定义

UARTRXLVL (UART Receive FIFO Level Register) Offset：0x24 Default：0x00000000 Access：UARTx->UARTRXLVL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	VAL						

表 19-17：UART 接收 FIFO 当前长度寄存器（UARTRXLVL）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6:0	VAL	RO	0x0	接收 FIFO 内保存的字节数（0~64）

表 19-18：UART 发送 FIFO 当前长度寄存器（UARTTXLVL）位段定义

UARTTXLVL (UART Transmit FIFO Level Register) Offset：0x28 Default：0x00000000 Access：UARTx->UARTTXLVL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	VAL						

©2025 旋智电子科技（上海）有限公司

表 19-19：UART 发送 FIFO 当前长度寄存器（UARTTXLVL）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6:0	VAL	RO	0x0	发送 FIFO 内保存的字节数（0~64）

表 19-20：UART 接收 FIFO 长度阈值寄存器（UARTRXTH）位段定义

UARTRXTH (UART Receive FIFO Threshold Register) Offset：0x2C Default：0x00000000
Access：UARTx->UARTRXTH

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED		VAL					

表 19-21：UART 接收 FIFO 长度阈值寄存器（UARTRXTH）位段描述

位段	位段名	属性	复位值	描述
31:6	RESERVED_31_6	RO	0x0	保留
5:0	VAL	RW	0x0	设置阈值使得 UARTRXLVL > UARTRXTH 时触发接收 FIFO 数据读取请求

表 19-22：UART 发送 FIFO 长度阈值寄存器（UARTTXTH）位段定义

UARTTXTH (UART Transmit FIFO Threshold Register) Offset：0x30 Default：0x00000000
Access：UARTx->UARTTXTH

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED		VAL					

表 19-29：UART 发送 FIFO 长度阈值寄存器（UARTTXTH）位段描述

位段	位段名	属性	复位值	描述
31:6	RESERVED_31_6	RO	0x0	保留
5:0	VAL	RW	0x0	设置阈值使得 UARTTXLVL ≤ UARTTXTH 时触发发送 FIFO 数据写入请求

©2025 旋智电子科技（上海）有限公司

表 19-23：UART 红外控制寄存器（UARTSIRCTL）位段定义

UARTSIRCTL (UART Infrared Control Register) Offset：0x34 Default：0x0000010C Access：UARTx->UARTSIRCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED		TXPW					
7	6	5	4	3	2	1	0
TXPW			XMODE	TXPOL	RXPOL	TXSIREN	RXSIREN

表 19-24：UART 红外控制寄存器（UARTSIRCTL）位段描述

位段	位段名	属性	复位值	描述
31:14	RESERVED_31_14	RO	0x0	保留
13:5	TXPW	RW	0x8	发送数据 0 时的脉冲宽度 最小值为 1，写 0 会自动复位成 1
4	XMODE	RW	0x0	发送数据 0 时的脉冲宽度控制模式 0：脉冲宽度固定为 3/16 个波特 1：脉冲宽度固定为 TXPW 个 UART 模块时钟
3	TXPOL	RW	0x1	发送数据的极性 0：发送负脉冲表示 0，发送高电平表示 1 1：发送正脉冲表示 0，发送低电平表示 1
2	RXPOL	RW	0x1	接收数据的极性 0：接收负脉冲表示 0，接收高电平表示 1 1：接收正脉冲表示 0，接收低电平表示 1
1	TXSIREN	RW	0x0	发送器红外模式使能 0：发送器工作于普通模式 IrDA 编码器关闭，发送器输出直接送到 TXD 引脚 1：发送器工作于红外模式 发送器输出经过 IrDA 编码后再送到 TXD 引脚
0	RXSIREN	RW	0x0	接收器红外模式使能 0：接收器工作于普通模式 IrDA 解码器关闭，RXD 引脚直接输入到接收器 1：接收器工作于红外模式 RXD 引脚信号进过 IrDA 解码后再输入到接收器

©2025 旋智电子科技（上海）有限公司

表 19-25：UART 状态寄存器（UARTSTS）位段定义

UARTSTS (UART Status Register) Offset：0x40 Default：0x0000000C Access：UARTx->UARTSTS							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						TXBUSY	RXBUSY
7	6	5	4	3	2	1	0
RXERR	RXBRK	RXFRMERR	RXPRTYERR	TXNF	TXEMPTY	RXFULL	RXNE

表 19-26：UART 状态寄存器（UARTSTS）位段描述

位段	位段名	属性	复位值	描述
31:10	RESERVED_31_10	RO	0x0	保留
9	TXBUSY	RO	0x0	发送器忙状态 0：发送器空闲 1：发送器在忙（发送数据）
8	RXBUSY	RO	0x0	接收器忙状态 0：接收器空闲 1：接收器在忙（接收数据）
7	RXERR	RO	0x0	接收 FIFO 中的未读数据含有线路中断条件字节、校验错误或者停止位错误 0：接收 FIFO 中都是正常数据 1：接收 FIFO 中包含未读的异常数据
6	RXBRK	RO	0x0	从接收 FIFO 刚读出的数据是线路中断条件 对于 32 位访问模式，表明读出的字节中至少有一个线路中断条件 0：由 UARTDAT 读出的全零字节是正常数据 1：由 UARTDAT 读出的全零字节是线路中断条件
5	RXFRMERR	RO	0x0	从接收 FIFO 刚读出的数据含停止位错 对于 32 位访问模式，表明读出的字节中至少有一个含停止位错 0：通过 UARTDAT 读出的数据不含停止位错 1：通过 UARTDAT 读出的数据含停止位错
4	RXPRTYERR	RO	0x0	从接收 FIFO 刚读出的数据含校验错 对于 32 位访问模式，表明读出的字节中至少有一个含校验错

SPINTROL

位段	位段名	属性	复位值	描述
				0：通过 UARTDAT 读出的数据不含校验错 1：通过 UARTDAT 读出的数据含校验错
3	TXNF	RO	0x1	发送 FIFO 非满 0：发送 FIFO 满，无法存放新数据 1：发送 FIFO 含至少一个空位可以存放数据
2	TXEMPTY	RO	0x1	发送 FIFO 空 0：发送 FIFO 里有至少一个数据 1：发送 FIFO 为空（没有数据）
1	RXFULL	RO	0x0	接收 FIFO 满 0：接收 FIFO 含至少一个空位可以存放数据 1：接收 FIFO 满，无法存放新数据
0	RXNE	RO	0x0	接收 FIFO 非空 0：接收 FIFO 为空（没有数据） 1：接收 FIFO 里有至少一个数据

表 19-27：UART 中断原始事件标志寄存器（UARTRAWIF）位段定义

UARTRAWIF (UART Interrupt Raw Flag Register) Offset：0x44 Default：0x00000000 Access：UARTx->UARTRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							RESERVED
15 DMAEOC	14 BUSIDLE	13 LINSBERR	12 LINCSERR	11 LINBITERR	10 LINIDMATCH	9 ABLOCK	8 TXREQ
7 TXOVF	6 RXTIMEOUT	5 RXBRK	4 RXFRMERR	3 RXPRTYERR	2 RXREQ	1 RXOVF	0 RXUDF

表 19-28：UART 中断原始事件标志寄存器（UARTRAWIF）位段描述

位段	位段名	属性	复位值	描述
31:17	RESERVED_31_17	RO	0x0	保留
16	RESERVED_16	RO	0x0	保留
15	DMAEOC	RO	0x0	DMA 搬运完成事件标志 0：事件未发生 1：事件已发生
14	BUSIDLE	RO	0x0	RXD 线路空闲事件标志 0：事件未发生 1：事件已发生
13	LINSBERR	RO	0x0	LIN 同步段错误事件标志

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				0：事件未发生 1：事件已发生
12	LINCSERR	RO	0x0	LIN 校验和段错误事件标志 0：事件未发生 1：事件已发生
11	LINBITERR	RO	0x0	LIN 位电平错误事件标志 0：事件未发生 1：事件已发生
10	LINIDMATCH	RO	0x0	LIN 接收到符合筛选标准的 PID 事件标志 0：事件未发生 1：事件已发生
9	ABLOCK	RO	0x0	自动波特率检测完成事件标志 0：事件未发生 1：事件已发生
8	TXREQ	RO	0x0	发送 FIFO 数据写请求事件标志 0：值为 0 表示发送 FIFO 中含足够的数据； 当 UARTTXLVL ≤ UARTTXTH 时位段变 1 1：值为 1 表示发送 FIFO 中需写入更多数据； 当 UARTTXLVL > UARTTXTH 时位段变 0
7	TXOVF	RO	0x0	发送 FIFO 上溢事件标志 表示发送 FIFO 满时发生了 UARTDAT 写操作 0：事件未发生 1：事件已发生
6	RXTIMEOUT	RO	0x0	接收超时事件标志 UART 模式下表示在指定时间内未收到新数据 LIN 模式下表示指定时间内未收到完整的应答 0：事件未发生 1：事件已发生
5	RXBRK	RO	0x0	接收到线路中断条件事件标志 0：事件未发生 1：事件已发生
4	RXFRMERR	RO	0x0	接收到字节帧停止位错误事件标志 0：事件未发生 1：事件已发生
3	RXPRTYERR	RO	0x0	接收到校验错误事件标志 UART 模式下表示字节的校验位错误 LIN 模式下表示受保护 ID 段的校验错误

SPINTROL

位段	位段名	属性	复位值	描述
				0：事件未发生 1：事件已发生
2	RXREQ	RO	0x0	接收 FIFO 数据读请求事件标志 0：值为 0 表示接收 FIFO 中未存入足够的数据；当 UARTRXLVL > UARTRXTH 时位段变 1 1：值为 1 表示接收 FIFO 中存入了足够的数据；当 UARTRXLVL ≤ UARTRXTH 位段变 0
1	RXOVF	RO	0x0	接收 FIFO 上溢事件标志 表示接收 FIFO 为满时收到新数据（会被丢弃） 0：事件未发生 1：事件已发生
0	RXUDF	RO	0x0	接收 FIFO 下溢事件标志 表示接收 FIFO 为空时发生了 UARTDAT 读操作 0：事件未发生 1：事件已发生

表 19-29：UART 中断标志寄存器（UARTIF）位段定义

UARTIF (UART Interrupt Flag Register) Offset：0x48 Default：0x00000000 Access：UARTx->UARTIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						INT	RESERVED
15	14	13	12	11	10	9	8
RESERVED	BUSIDLE	LINSBERR	LINCSERR	LINBITERR	LINIDMATCH	ABLOCK	TXREQ
7	6	5	4	3	2	1	0
TXOVF	RXTIMEOUT	RXBRK	RXFRMERR	RXPRTYERR	RXREQ	RXOVF	RXUDF

表 19-30：UART 中断标志寄存器（UARTIF）位段描述

位段	位段名	属性	复位值	描述
31:18	RESERVED_31_18	RO	0x0	保留
17	INT	RO	0x0	UART 全局中断标志 0：中断未发生 1：中断已产生，并已经通知 CPU 本位段被清除之前不会产生新中断
16	RESERVED_16	RO	0x0	保留
15	RESERVED_15	RO	0x0	保留

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
14	BUSIDLE	RO	0x0	总线空闲中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
13	LINSBERR	RO	0x0	LIN 同步段错误中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
12	LINCSERR	RO	0x0	LIN 校验和段错误中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
11	LINBITERR	RO	0x0	LIN 位电平错误中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
10	LINIDMATCH	RO	0x0	LIN 接收到符合筛选标准的 PID 中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
9	ABLOCK	RO	0x0	自动波特率检测完成中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
8	TXREQ	RO	0x0	发送 FIFO 数据写请求中断标志 0：中断未发生 1：中断已发生
7	TXOVF	RO	0x0	发送 FIFO 上溢中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
6	RXTIMEOUT	RO	0x0	接收超时中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
5	RXBRK	RO	0x0	接收到线路中断条件中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
4	RXFRMERR	RO	0x0	接收到字节帧停止位错误中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
3	RXPRTYERR	RO	0x0	接收到校验错误中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
2	RXREQ	RO	0x0	接收 FIFO 数据读请求中断标志 0：中断未发生 1：中断已发生
1	RXOVF	RO	0x0	接收 FIFO 上溢中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
0	RXUDF	RO	0x0	接收 FIFO 下溢中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断

表 19-38：UART 中断清除寄存器（UARTIC）位段定义

UARTIC (UART Interrupt Clear Register) Offset：0x4C Default：0x00000000 Access：UARTx->UARTIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						INT	RESERVED
15	14	13	12	11	10	9	8
RESERVED	BUSIDLE	LINSBERR	LINCSERR	LINBITERR	LINIDMATCH	ABLOCK	RESERVED
7	6	5	4	3	2	1	0
TXOVF	RXTIMEOUT	RXBRK	RXFRMERR	RXPRTYERR	RESERVED	RXOVF	RXUDF

表 19-39：UART 中断清除寄存器（UARTIC）位段描述

位段	位段名	属性	复位值	描述
31:18	RESERVED_31_18	RO	0x0	保留
17	INT	W1C	0x0	全局中断清除 0：写 0 无效，读总是 0 1：写 1 清除中断标志，本位段自动清零
16	RESERVED_16	W1C	0x0	保留

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
15	RESERVED_15	W1C	0x0	保留
14	BUSIDLE	W1C	0x0	总线空闲中断清除 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
13	LINSBERR	W1C	0x0	LIN 同步段错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
12	LINCSERR	W1C	0x0	LIN 校验和段错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
11	LINBITERR	W1C	0x0	LIN 位电平错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
10	LINIDMATCH	W1C	0x0	LIN 接收到符合筛选标准的 PID 中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
9	ABLOCK	W1C	0x0	自动波特率检测完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
8	RESERVED_8	RO	0x0	保留
7	TXOVF	W1C	0x0	发送 FIFO 上溢中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
6	RXTIMEOUT	W1C	0x0	接收超时中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
5	RXBRK	W1C	0x0	接收到线路中断条件中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
4	RXFRMERR	W1C	0x0	接收到字节帧停止位错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
3	RXPRTYERR	W1C	0x0	接收到校验错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
2	RESERVED_2	RO	0x0	保留
1	RXOVF	W1C	0x0	接收 FIFO 上溢中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
0	RXUDF	W1C	0x0	接收 FIFO 下溢中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

表 19-40：UART 中断使能寄存器（UARTIE）位段定义

UARTIE (UART Interrupt Enable Register)　Offset：0x50　Default：0x00000000 Access：UARTx->UARTIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							RESERVED
15	14	13	12	11	10	9	8
RESERVED	BUSIDLE	LINSBERR	LINCSERR	LINBITERR	LINIDMATCH	ABLOCK	TXREQ
7	6	5	4	3	2	1	0
TXOVF	RXTIMEOUT	RXBRK	RXFRMERR	RXPRTYERR	RXREQ	RXOVF	RXUDF

表 19-31：UART 中断使能寄存器（UARTIE）位段描述

位段	位段名	属性	复位值	描述
31:17	RESERVED_31_17	RO	0x0	保留
16	RESERVED_16	RW	0x0	保留
15	RESERVED_15	RW	0x0	保留
14	BUSIDLE	RW	0x0	总线空闲中断使能 0：关闭 1：使能

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
13	LINSBERR	RW	0x0	LIN 同步段错误中断使能 当 UARTCTL.ABEN 被置 1 时自动波特率检测使能，需禁用此位段以免误报。 0：关闭 1：使能
12	LINCSERR	RW	0x0	LIN 校验和段错误中断使能 0：关闭 1：使能
11	LINBITERR	RW	0x0	LIN 位电平错误中断使能 0：关闭 1：使能
10	LINIDMATCH	RW	0x0	LIN 接收到符合筛选标准的 PID 中断使能 0：关闭 1：使能
9	ABLOCK	RW	0x0	自动波特率检测完成中断使能 0：关闭 1：使能
8	TXREQ	RW	0x0	发送 FIFO 数据写请求中断使能 0：关闭 1：使能
7	TXOVF	RW	0x0	发送 FIFO 上溢中断使能 0：关闭 1：使能
6	RXTIMEOUT	RW	0x0	接收超时中断使能 0：关闭 1：使能
5	RXBRK	RW	0x0	接收到线路中断条件中断使能 0：关闭 1：使能
4	RXFRMERR	RW	0x0	接收到字节帧停止位错误中断使能 0：关闭 1：使能
3	RXPRTYERR	RW	0x0	接收到校验错误中断使能 0：关闭 1：使能
2	RXREQ	RW	0x0	接收数据读请求中断使能 0：关闭 1：使能

SPINTROL

位段	位段名	属性	复位值	描述
1	RXOVF	RW	0x0	接收 FIFO 上溢中断使能 0：关闭 1：使能
0	RXUDF	RW	0x0	接收 FIFO 下溢中断使能 0：关闭 1：使能

SPINTROL

©2025 旋智电子科技（上海）有限公司

表 19-32：UART 接收信号消抖控制寄存器（UARTDGCTL）位段定义

UARTDGCTL (UART RXD Deglitch Control) Offset：0x54 Default：0x00000000 Access：UARTx->UARTDGCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED				FLTDIV			
7	6	5	4	3	2	1	0
FLTTH				FLTWIN			

表 19-33：UART 接收信号消抖控制寄存器（UARTDGCTL）位段描述

位段	位段名	属性	复位值	描述
31:12	RESERVED_31_12	RO	0x0	保留
11:8	FLTDIV	RW	0x0	设置 UART 模块时钟经过(FLTDIV+1)分频后用作输入信号消抖时钟
7:4	FLTTH	RW	0x0	设置消抖滤波阈值为采样窗口内含至少(FLTTH+1)个 1 或 0，才判定为输入 1 或 0，否则保持原来的值不变
3:0	FLTWIN	RW	0x0	设置消抖滤波窗口为(FLTWIN+1)个采样点

表 19-34：UART 模块写使能寄存器（UARTREGKEY）位段定义

UARTREGKEY (Uart Register Write-Allow Key Register) Offset：0x58 Default：0x1ACCE551 Access：UARTx->UARTREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

表 19-35：UART 模块写使能寄存器（UARTREGKEY）位段描述

位段	位段名	属性	复位值	描述
31:0	KEY	RW	0x1ACCE551	写入 0x1ACCE551 解锁受保护的 UART 寄存器

SPINTROL

20 串行外设接口（SPI）

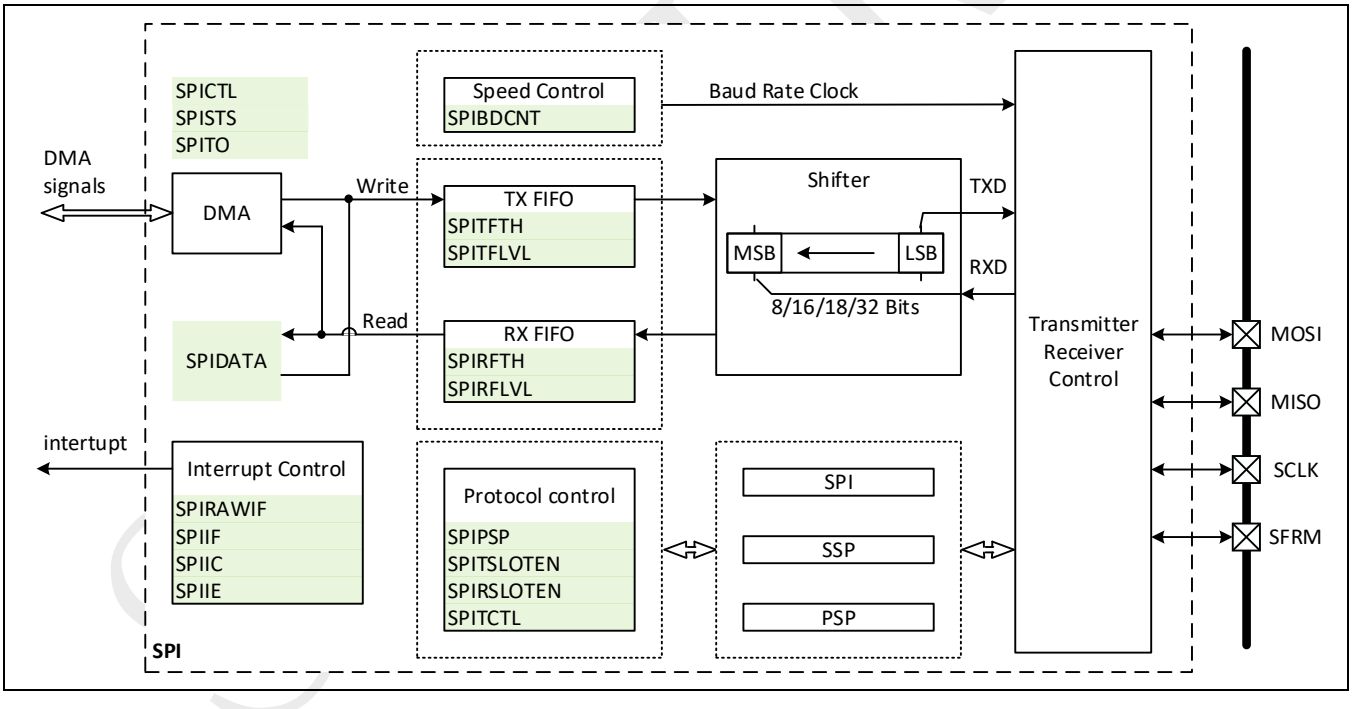
20.1 概述

SPI 接口是一个同步全双工串行数据传输接口，可以用来连接闪存、ADC 等多种外部设备。SPI 接口可以编程为在主模式（连接的外围设备作为从设备）或从模式（连接外围设备作为主设备）下操作。并支持 SSP（TI 同步串行协议）、SPI（Motorola 串行外设接口）、PSP（可编程串行协议）。

SPI 端口通过可配置的接口时钟支持不同的串行比特率。串行数据样本大小可以配置为 8、16、18、32 位的长度。为发送数据提供 FIFO，为接收数据提供第二独立 FIFO。两个 FIFO 的深度均为 16 个样本，宽度均为 32 位。FIFO 可以由 CPU 加载或清空。SPI 可以通过轮询、中断或 DMA 来驱动。

SPI 模块主要由控制、状态和数据寄存器、移位器逻辑、FIFO 控制逻辑、发送器和接收器控制、中断控制组成，如图 20-1 所示。

图 20-1：SPI 框图



20.2 特性

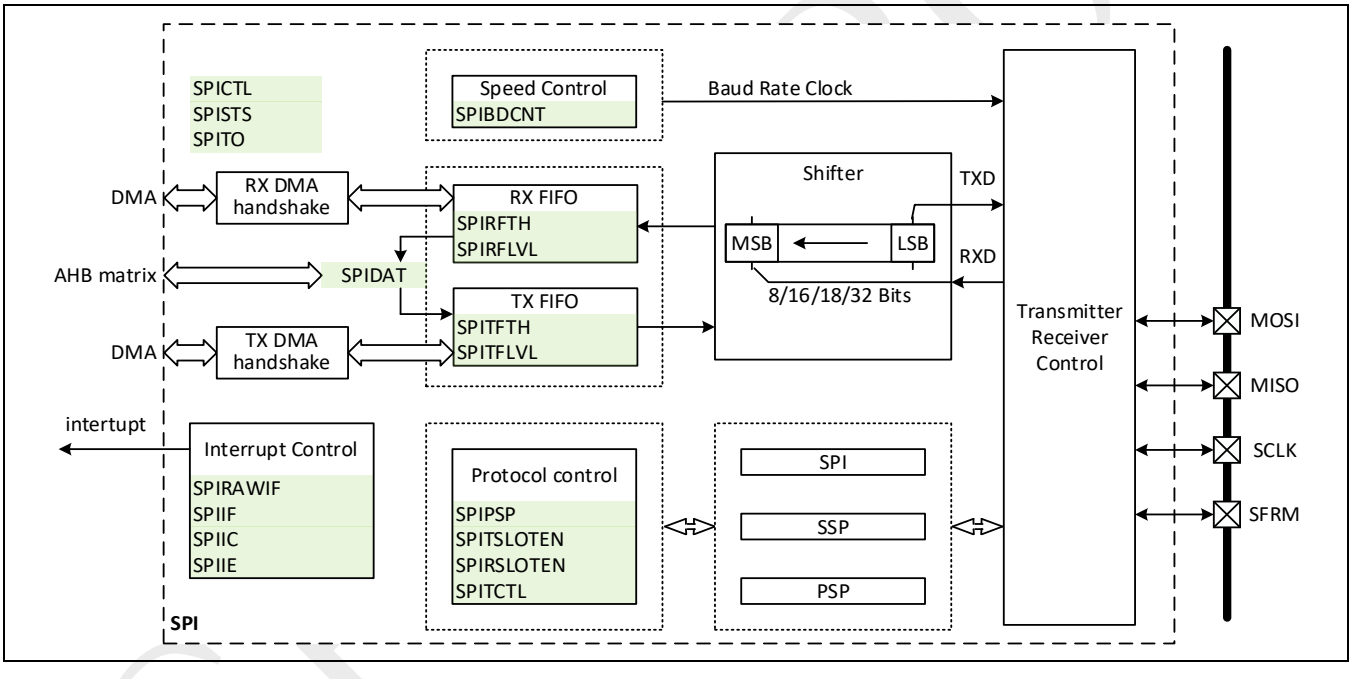
SPC1169 的 SPI 有如下特性：

- 全双工串行同步传输；
- 最高 50 Mbps 通信速率；
- 支持 Motorola SPI 格式和 TI SSP 格式；
- 支持 PSP（可编程串行协议）

©2025 旋智电子科技（上海）有限公司

- 主机或从机操作；
- 可变长度传输帧格式选择；
- 仅接收操作；
- 可编程数据帧大小：8/16/18/32 位；
- SFRM 的可编程极性；
- 可编程三态输出；
- MSB 第一数据顺序；
- 两个独立的发送和接收 FIFO，深度为 16 个样本，宽度为 32 位，均支持打包模式；
- 支持 DMA。

图 20-2：SPI 框图



20.3 SPI 端口信号

表 20-1 罗列并描述了 SPI 端口信号。

表 20-1：SPI 端口信号

信号名	类型	描述
MOSI	输入/输出	**数据传输信号** 主机模式：发送数据 从机模式：接收数据
MISO	输入/输出	**数据传输信号** 主机模式：接收数据 从机模式：发送数据
SCLK	输入/输出	**串行时钟信号** 主机模式：输出时钟信号（数据传输时翻转） 从机模式：接收时钟信号
SFRM	输入/输出	**片选信号** 主机模式：输出片选信号（数据传输时为有效电平） 从机模式：接收片选信号

20.4 SPI 功能描述

本节描述了 SPI 的各种操作模式，包括波特率生成、FIFO 操作和中断。

20.4.1 波特率产生

SPI 包含一个可编程的波特率生成器，可以将 SPI 时钟（CLK_SPI）进行分频，生成所需的波特率。通过以下公式计算 SPI 传入或传出的数据的波特率：

$$\text{BaudRate} = \frac{CLK_SPI}{\text{SPIBDCNT} + 1}$$

其中 SPIBDCNT 以 16 位二进制格式存储除数，在初始化期间加载这些除数寄存器，以确保波特率生成器正常运行。

20.4.2 FIFO 操作

SPC1169 拥有两个独立的 FIFO，分别用于发送和接收。CPU 可以装载和清空这些 FIFO。CPU 每次访问可以传输一个 FIFO 条目（FIFO Entry），而且这些访问必须是 32 位宽度的。CPU 写数据到 TXFIFO 是 32 位宽度的，但是串行化逻辑会忽略超过 FIFO 数据大小（SPICTL0.SIZE）的高位数据。CPU 从 RXFIFO 读数据也是 32 位宽度的，FIFO 数据大小不够 32 位的，会在高位补 0。

从 CPU 的角度看来，TXFIFO 和 RXFIFO 共用同一个 32 位宽的位置。对于数据发送，SSP 接口从 TXFIFO 中获取数据，将数据串行化并通过 TXD 信号接口将数据发送到外部设备；对于数据接收，通过 RXD 信号接口将串行数据转换为并行数据并写入到 RXFIFO 中。取决于来自 CPU 的访问

©2025 旋智电子科技（上海）有限公司

是读还是写操作，FIFO 数据寄存器会自动定位到 RXFIFO 或者 TXFIFO。从存储器映射的角度来看，TXFIFO 和 RXFIFO 共享相同的地址。

假若可编程的 FIFO 触发阈值被超过了，会产生一个中断服务请求，用来通知 CPU 清空 RXFIFO 或重新装满 TXFIFO。

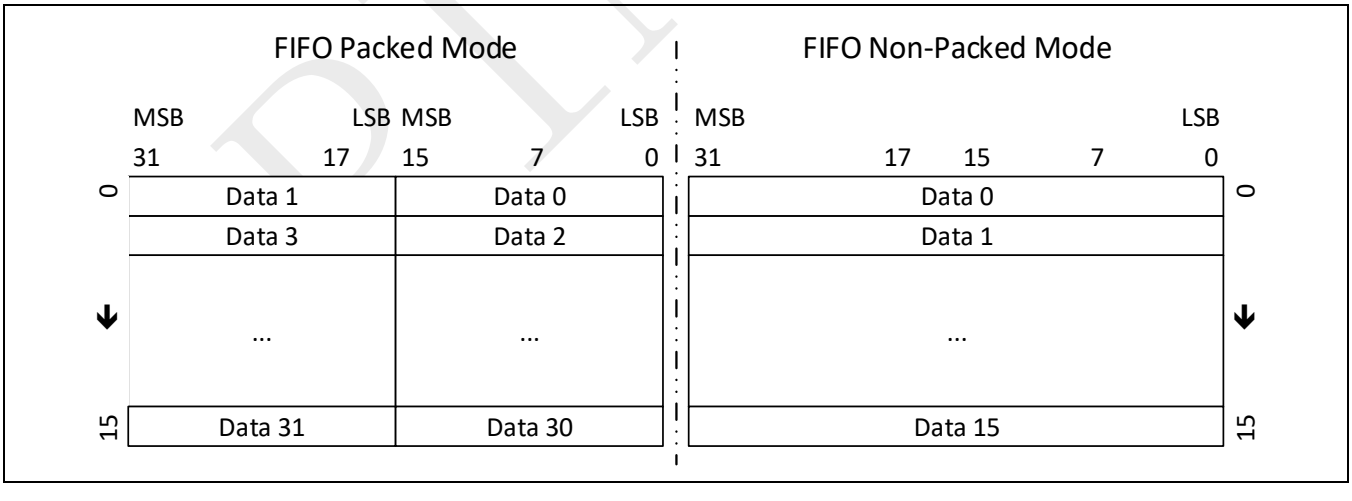
注意： 如果 SPI 被配置为从机且工作在 Receive Only 模式（SPICTL0.RXONLY=1），那么在接收数据的时候没有必要去写 TXFIFO（半双工）。但是，写 TXFIFO 仍然会发送数据。

20.4.2.1 FIFO 存储的并行数据格式

样本数据大小可以是 8、16、18 或 32 位。FIFO 中的数据存储为每个数据样本 32 位宽或每个样本 16 位宽（压缩模式）。在 32 位或 16 位采样模式内，存储的数据采样是右对齐的，字的 LSB 位于位 0。对于读取模式，未使用的位被打包为 MSB 之上的零，而对于写入模式，则被打包为“不在乎”位。SPI 中的逻辑将自动格式化传输 FIFO 中的数据，以便以所选帧格式在 TXD 信号上正确传输采样数据。

当 FIFO 在压缩模式下操作时，如图 20 2 所示，每个 FIFO 的深度为 32 行，宽度为 16 位，共有 32 个数据样本。每个样本的长度可以是 8 或 16 位。当数据被串行化和传输时，首先传输位 15-0，然后是位 31-16。打包模式的最佳实践是将 FIFO 视为两个样本的单个条目。因此，CPU 应该一次写入和读取 32 位数据，其中每个写入或读取传输两个样本。必须在此模式下读取或写入整个 FIFO 宽度（32 位）。此外，FIFO 触发阈值需要以 32 位读取和写入来计算，而不是基于 16 位。

图 20-3：FIFO 压缩和非压缩模式



读取 SPISTS.RFEVEN and SPISTS.TFEVEN 可以判断 FIFO 中的样本数据是偶数个还是奇数个。这些数据位有助于在 FIFO 压缩模式下正确地读取数据。

©2025 旋智电子科技（上海）有限公司

20.4.2.2 RXFIFO 中尾部字节（trailing bytes）

当接收 FIFO 中的样本数小于其触发阈值水平且未接收到额外数据时，剩余字节称为接收 FIFO 尾随字节。接收 FIFO 尾部字节可以由 CPU 处理。通过超时机制和超时后接收 FIFO 内数据的存在来识别接收 FIFO 的尾随字节。

尾部字节由实现中的超时逻辑触发。当接收 FIFO 空闲了 SPITO 定义的一段时间时，就会发生超时。当发生超时时，接收器超时中断位 SPIIF.TO 被设置为 1，并且如果接收器超时中断使能位 SPIIE.TO 被设定，则超时中断向 CPU 发出超时条件已经发生的信号。在接收 FIFO 中接收到新的数据样本后，超时计时器被重置。一旦设置了 SPIIF.TO 位，就必须通过向 SPIIC.TO 位写入 0x1 来清除它。

默认情况下，接收 FIFO 中剩余的尾随字节由 CPU 轮询方法处理。如果发生超时，CPU 只有在通过设置 Receiver Time out interrupt Enable（SPIIE.TO）字段启用超时中断的情况下才会被超时中断中断。要从接收 FIFO 中读取尾随字节，软件应等待超时中断，然后读取由接收 FIFO 偶样状态（SPISTS.RFEVEN）、接收 FIFO 级别（SPIRFLVL）指示的所有尾随字节，以及 SPI 状态寄存器中的接收 FIFO 非空（SPISTS.RXNE）字段。

注意：	如果 FIFO 压缩模式被使能（SPICTL0.FPCKEN=1），尾部字节必须通过查询方式来清除。如果 SPISTS.RFODDSTS 位段的值为“1”，那么最后一个 FIFO entry 只包含一个数据样本。

20.4.2.3 FIFO 阈值

每个 FIFO 都有一个可配置的触发阈值，用来触发中断。当 RXFIFO 中的条目个数超过了 RXFIFO 的触发阈值（SPIRFTH），会产生一个中断（如果该中断被使能）去通知 CPU 清空 RXFIFO。当 TXFIFO 中的条目个数小于或者等于 TXFIFO 的触发阈值（SPITFTH）加 1，会产生一个中断（如果该中断被使能）去通知 CPU 将 TXFIFO 填满。

CPU 可以查询 SPI 状态寄存器来判断 FIFO 中有多少数据或者 FIFO 是满还是空。软件负责选择合适的 RXFIFO/TXFIFO 触发阈值，以防止 RXFIFO 上溢或者 TXFIFO 下溢。

20.4.3 SPI 中断

中断标志寄存器（SPIRAWIF、SPIIF）包含用于指示溢出错误以及发送 FIFO 和接收 FIFO 服务请求的位，这些由硬件检测到的事件中的每一个都会向中断控制器发送中断请求。SPISTS 寄存器也包含一些标志位用来指示 SPI 接口是否忙于发送数据，TXFIFO 是否非满以及 RXFIFO 是否非空。会有相应的中断信号被送到中断控制器。这些事件可以产生下面的中断：接收器超时、RXFIFO 上溢、RXFIFO 服务请求以及 TXFIFO 服务请求。其中接收 FIFO 服务请求和发送 FIFO 服务请求还会引发 DMA 请求。

引发中断请求的位会保持设置状态，直到通过向 SPIIC 中的相应位写入 1 清除（由 FIFO 硬件逻辑控制的 TXREQ 和 RXREQ 位除外），一旦标志位被清除，中断就会被清除。

©2025 旋智电子科技（上海）有限公司

20.5 Motorola SPI 操作

20.5.1 设备连接

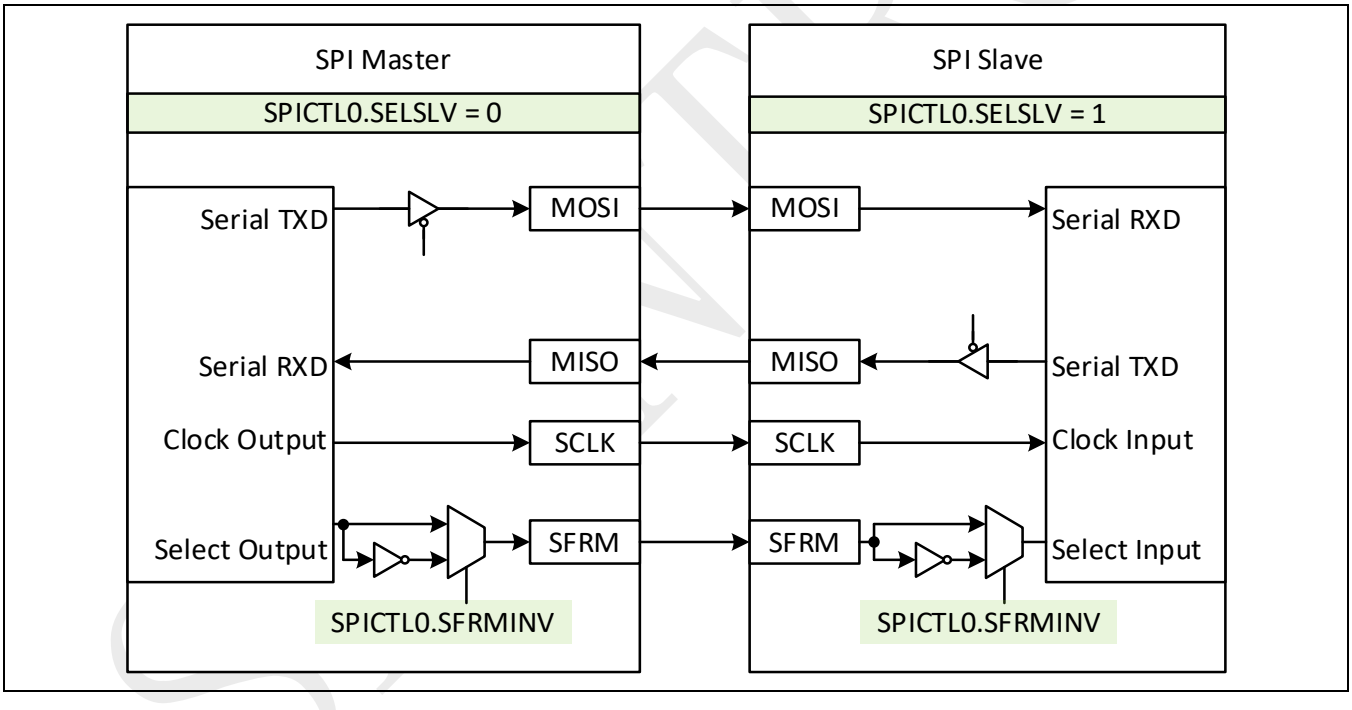
SFRM 在数据传输期间充当片选信号，用于启用外部设备（目标传输），并在数据传输期间保持低电平，并且软件可以更改其电平。

SCLK 只在数据传输期间存在（不连续运行），软件设置可以选择在 SCLK 的哪个边沿上发送数据以及采样接收数据。

SPI 可以配置为主模式或从模式，支持全双工格式。主模式发送时钟 SCLK 和信号 SFRM 给从模式，从模式根据 SCLK 和 SFRM 开始传输数据。主模式的数据端口（MOSI/MISO）与同名的从模式数据端口连接。主模式将数据发送到 MOSI，从 MISO 接收数据；从模式从 MOSI 接收数据，并将数据发送到 MISO。

如图 20-4 所示，SPI 的典型连接用于两个控制器之间的通信：一个主模式和一个从模式。

图 20-4：SPI 主从链接



有三种可能的数据传输方法：

- 主模式发送数据；从模式发送虚拟数据。
- 主模式发送数据；从模式发送数据。
- 主模式发送虚拟数据；从模式发送数据。

©2025 旋智电子科技（上海）有限公司

20.5.2 协议描述

SPI 格式有四个子模式，取决于 SCLK 的相位模式（用于传输和接收数据的边沿）和 SCLK 的极性模式（SCLK 的空闲状态），由寄存器 SPICTL0.SPIMODE 控制：

- PHS 为 0，POL 为 0：SCLK 在空闲时为低电平，第一个边沿用于采样数据；
- PHS 为 0，POL 为 1：SCLK 在空闲时为高电平，第一个边沿用于采样数据；
- PHS 为 1，POL 为 0：SCLK 在空闲时为低电平，第二个边沿用于采样数据；
- PHS 为 1，POL 为 1：SCLK 在空闲时为高电平，第二个边沿用于采样数据；

在 SPI 格式下，SPI 端口可以是主模式或从模式，并且时钟（SCLK）和帧（SFRM）方向都由主设备控制（SPICTL0.SELSLV 为 0）。

在 SPI 格式下，如果 SPI 端口为主模式且 SPIPSP.ETDS 被清除，TXD 的传输结束数据状态为低电平。如果 SPI 端口为主模式且 SPIPSP.ETDS 被设置，SPIx_TXD 的传输结束数据状态保持为最后一个传输的位（LSB）。如果 SPI 端口为从模式，则 SPIPSP.ETDS 为未定义。

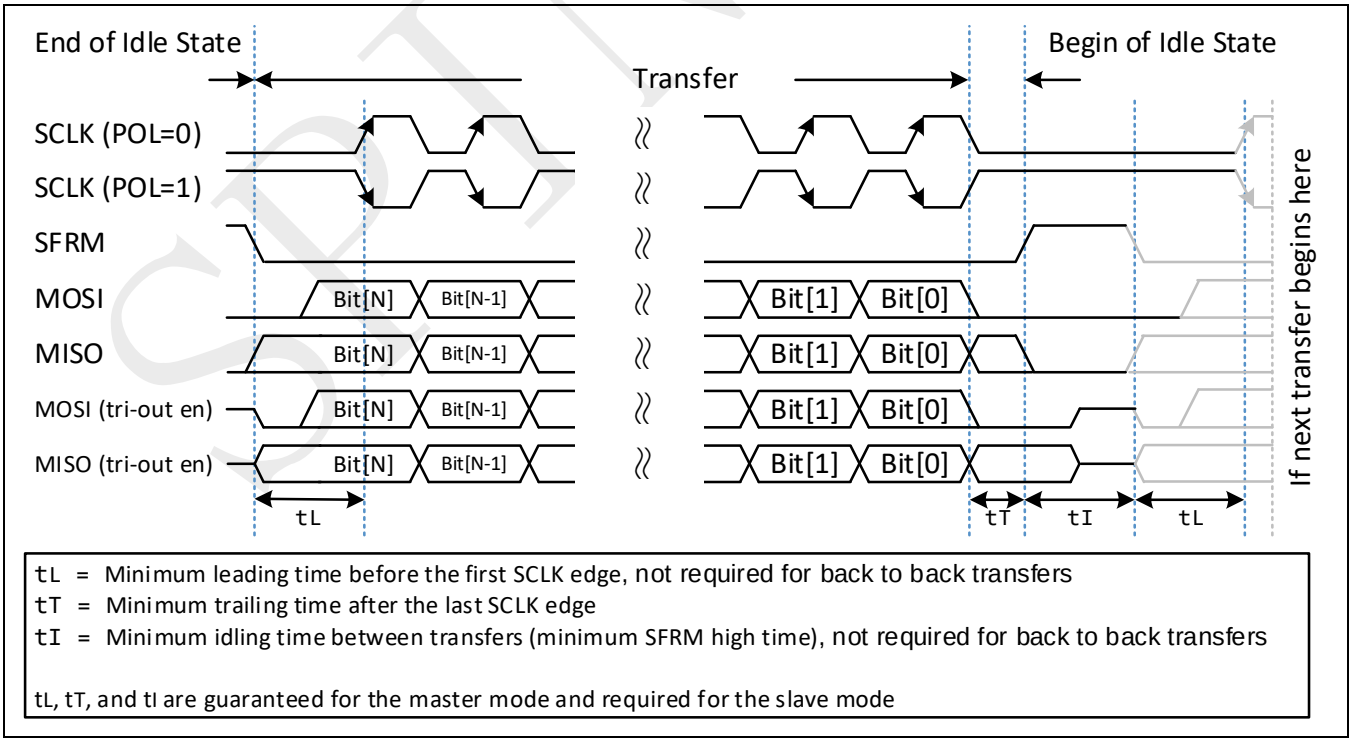
在帧激活之前和 LSB 接收之后，RXD 为未定义状态

当 SPI 端口配置为主模式且 SPICTL0.TRIEN 被设置时，忽略 SPIPSP.ETDS，并且 TXD 在数据传输之间变为高阻态。

如果 SPI 端口被禁用或复位，TXD 被强制为零（若启用了 TXD 三态输出，则为高阻态）。

图 20-5 和图 20-6 展示了单个传输帧的 SPI 帧协议的四种可能配置。

图 20-5：SPI 协议（PHS=0）



©2025 旋智电子科技（上海）有限公司

图 20-5 显示了 PHS 为 0 时的 SPI 协议。当 SCLK POL 为 0 时，解释此图。当 SPI 端口被禁用或处于空闲模式时，SCLK 和 TXD 为低，SFRM 为高。当发送数据准备发送时，SFRM 变低（SCLK 的第一个上升沿之前的一个时钟周期），并在帧的剩余部分保持低。在 SCLK 半周期之后，串行数据的 MSB（最高有效位）被驱动到 TXD 上。在第一位周期进行到一半时，SCLK 断言为高，并继续对剩余的数据位进行切换。数据在 SCLK 的下降沿上转换，并在 SCLK 上升沿上采样。每帧可以传输 8、16、18 或 32 个比特（实际上，每帧可以传送 3-32 个比特）。

随着 SFRM 的断言，接收数据从 RXD 上的外围设备同时驱动，首先是 MSB。数据在 SCLK 下降沿上转换，并由控制器在 SCLK 上升沿上采样。在帧结束时，在最后一位被锁存在其目的地并且完成的输入字被移位到接收 FIFO 中之后，SFRM 在一个时钟周期（SCLK 的最后一个下降沿之后的半个 SCLK 周期）被去断言为高。

在发送帧的最后一位之后，外围设备可以将 RXD 驱动到高阻抗状态。当控制器进入空闲模式时，TXD 可以是低电平或保留最后一个传输值。图 21-6 显示了 PHS 为 1 时的 SPI 协议。该过程与图 20-5 类似。

图 20-6：SPI 协议（PHS=1）

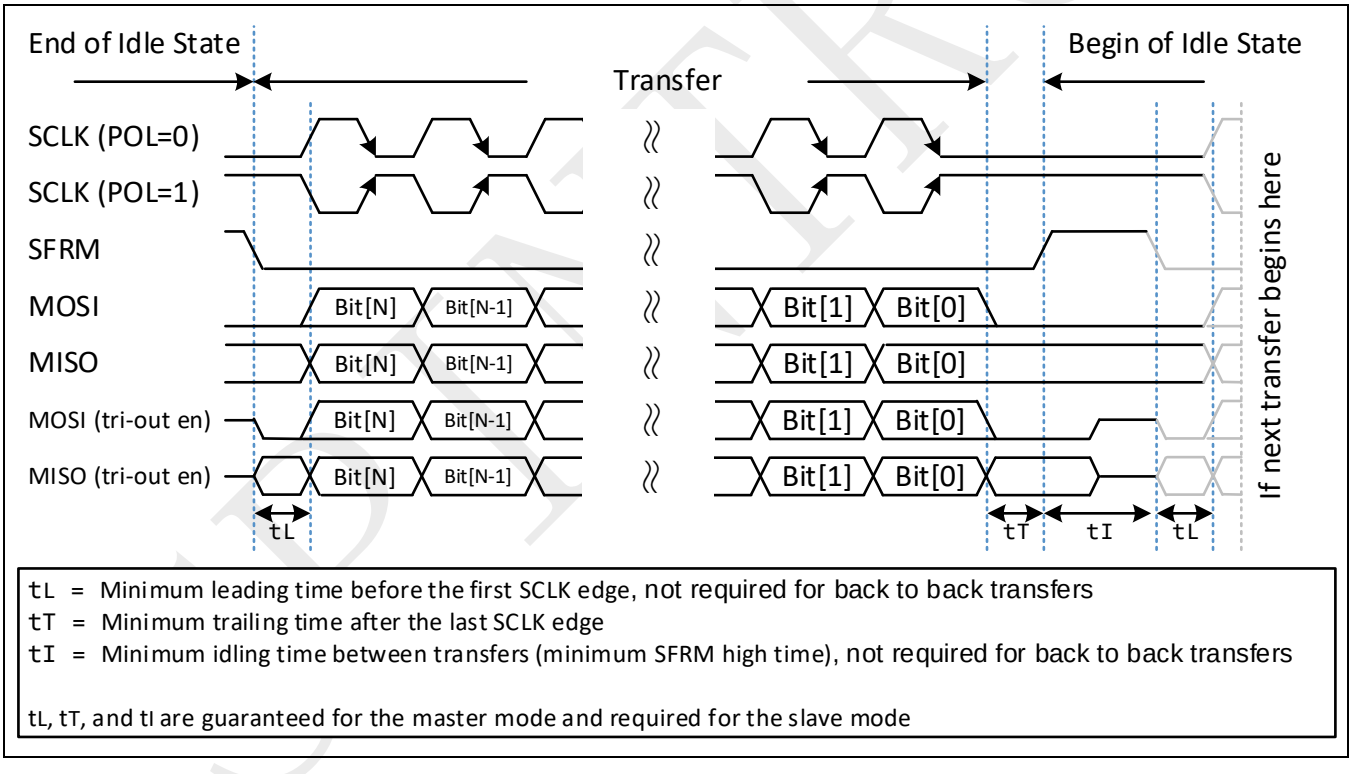
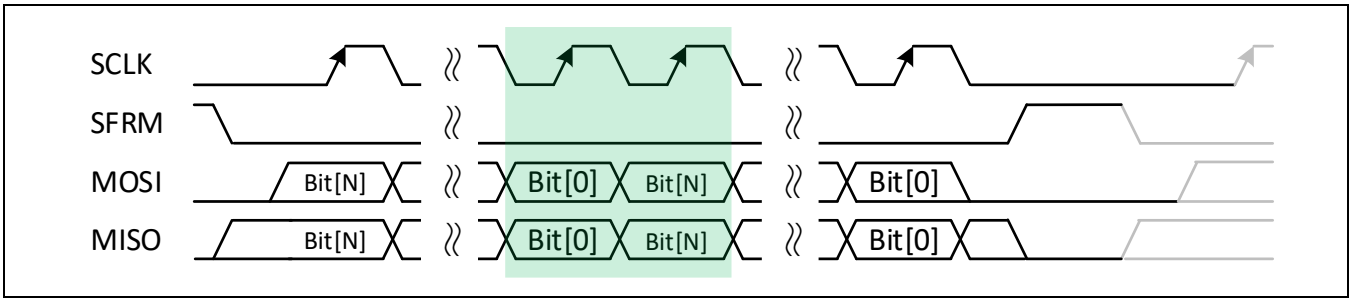


图 20-7：SPI 帧背靠背传输



每当发送 FIFO 非空时，主机会将 SFRM 置为有效，产生 SCLK 时钟，将数据从发送 FIFO 取出放入发送移位寄存器，开始数据位的传输。在完成当前数据最后一位传输后，若发送 FIFO 为空，主机会将 SFRM 置为无效，从而结束一帧数据。如图 20-7 所示，若完成最后一位传输后发送 FIFO 非空，则主机会保持 SFRM 有效的状态，并立即从发送 FIFO 取出数据开始新的传输。直到发送完最后一位后检测到发送 FIFO 为空，主机才会将 SFRM 置为无效，从而结束一帧数据。这称为背靠背传输，可以将多个数据的传输合并在同一帧里完成。

对于从机而言，在背靠背传输模式下，用户需要及时往发送 FIFO 写入数据，以免出现发送 FIFO 下溢。

注意：
1. 在 SFRM 恢复高电平时，SPI 端口的输入时钟不能处于活动状态。
2. 当 SPI 端口作为从设备与时钟同步时，必须设置 SPICTL0.SLVCKRUN。
3. 当 SPI 三态输出使能时，三态输出相位（SPICTL0.TRIPHS）必须为 0。

20.6 TI SSP 操作

20.6.1 设备连接

在 SSP 帧协议中，SFRM 信号在每个帧的开始处脉冲为高电平，持续一个（串行）数据周期。软件可以设置其活动时的电平。

SCLK 仅在数据传输期间切换（不连续运行），数据在 SCLK 上升沿传输，接收数据在下降沿采样。当 SCLK 由另一个设备驱动时，它可以连续运行或仅在数据传输期间驱动。

SSP 可以配置为主设备或从设备，支持全双工格式。

在该模式下，设备的连接方式与图 20-4 相同。

20.6.2 协议描述

当 SSP 控制器中的输出数据准备发送时，SFRM 在一个时钟周期内置为高电平。在接下来的时钟周期中，要发送的数据逐位驱动到 TXD 上，最高有效位（MSB）首先发送。对于接收数据，外设类似地将数据驱动到 RXD 引脚上。字长可以是 8、16、18 或 32 位。所有输出转换发生在 SCLK 上升沿，而数据采样发生在下降沿。

©2025 旋智电子科技（上海）有限公司

在传输结束时，TXD 信号要么保持上一位发送的值（位 0），要么进入高阻抗（SPICTL0.TRIEN，SPICTL0.TRIPHS）状态，如图 20-8 和图 20-9 所示。

如果 SSP 端口被禁用或复位，TXD 将被强制为零（除非启用 TXD 三态输出，否则将进入高阻抗状态）。

图 20-8：SSP 协议（单次传输）

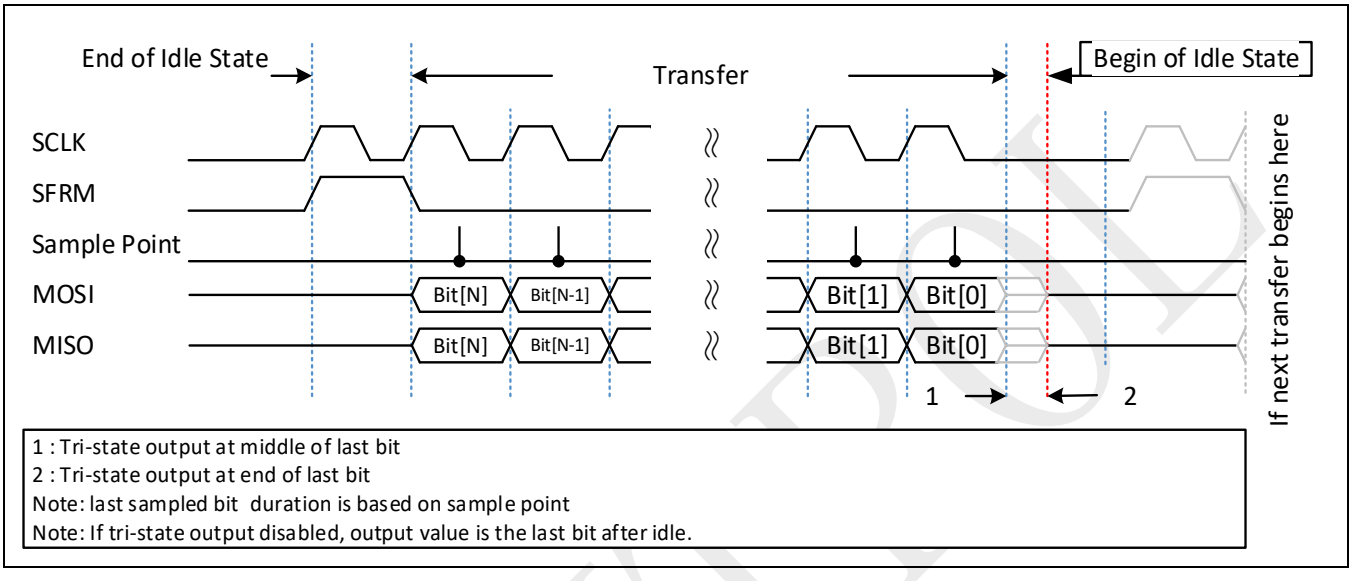


图 20-8 展示了单个传输帧的 SSP 工作原理。一旦传输 FIFO 中包含数据，SFRM 会在一个 SCLK 周期内脉冲为高电平。待发送的值从传输 FIFO 传输到串行移位寄存器，在下一个 SCLK 上升沿时，数据的最高有效位（MSB）被移位到 TXD 引脚上。同样，外部串行从设备通过下级电路将接收到的数据的最高有效位（MSB）移位到 RXD 引脚上。然后，SSP 端口和外部串行从设备会在每个 SCLK 的下降沿将每个数据位锁存到串行移位寄存器中。

图 20-9：SSP 协议（背靠背传输）

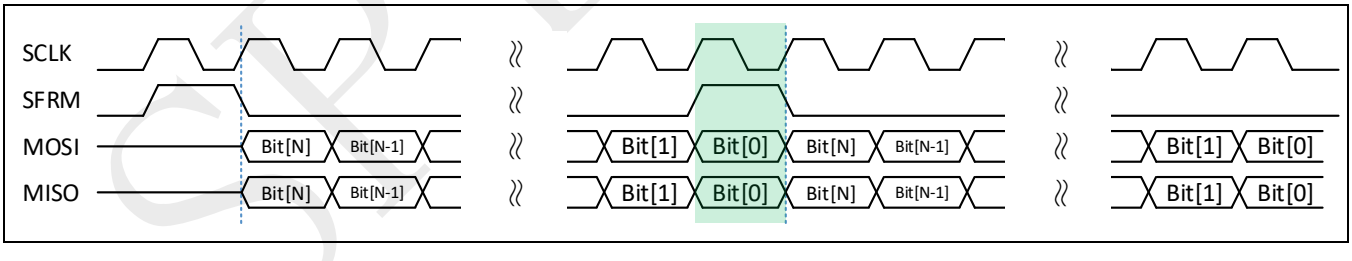


图 20-10 展示了在连续传输时的 SSP 工作原理。对于连续传输，一个传输帧的开始紧接着上一个传输帧的完成。一个传输帧的最高有效位（MSB）紧接着上一个传输帧的最低有效位（LSB）而没有任何“死区”时间。

注意：
1. 如果 SPICTL0.EN 被清零，TXD 线将变为低电平。
2. 在 SSP 模式下，SPIPSP.ETDS 没有作用。
3. 在发送最高有效位 MSB 前和发送最低有效位 LSB 后，RXD 处于未定义状态。

要启用连续传输模式，请确保在数据传输过程中传输 FIFO 中有超过一个数据。如果不想使用连续传输模式，应禁用 FIFO packed 模式，并确保在当前数据传输完成之前传输 FIFO 为空。

注意：当 SSP 为从设备且 SCLK 不连续运行时，必须设置 SPICTL0.SLVCKRUN 以确保从 CLK_AHB 域中的同步控制信号传输到 CLK_SPI 域。否则，在启用 SPI 模块并开始传输后，会丢失第一个数据。

20.7 可编程串行协议 PSP

20.7.1 设备连接

SFRM 方向、延迟、极性和宽度可编程。

SCLK 只在数据传输期间存在，软件可以选择 SCLK 的下降沿或上升沿来传输数据和采样接收数据，此外，软件还可以控制 SCLK 的空闲状态以及数据传输前后的活动时钟数。当 SCLK 由另一个设备驱动时，可以连续运行或仅在数据传输期间活动，但对于 PSP 参数存在一定的限制。

SPI 可以配置为主模式或从模式，并可以编程为全双工或半双工格式。

在此模式下，设备连接方式与图 20-4 相同。

20.7.2 协议描述

PSP 格式定义了可编程参数，用于确定数据采样之间的传输时序。SPIPSP.CLKMODE 定义了四种串行时钟模式，这些模式选择了驱动数据和采样接收数据时所使用的 SCLK 降沿情况。

如

©2025 旋智电子科技（上海）有限公司

图 20-10 和

SPINTROL

图 20-11 所示，使用以下字段对 TXD、SCLK 和 SFRM 信号的模式进行编程：SPIPSP.ETDS、SPIPSP.SFRMPOL 和 SPIPSP.CLKMODE。当准备好传输数据时，SCLK 将保持在其空闲状态，持续时间为 SPIPSP.STARTDLY 个串行时钟（SCLK）周期数。

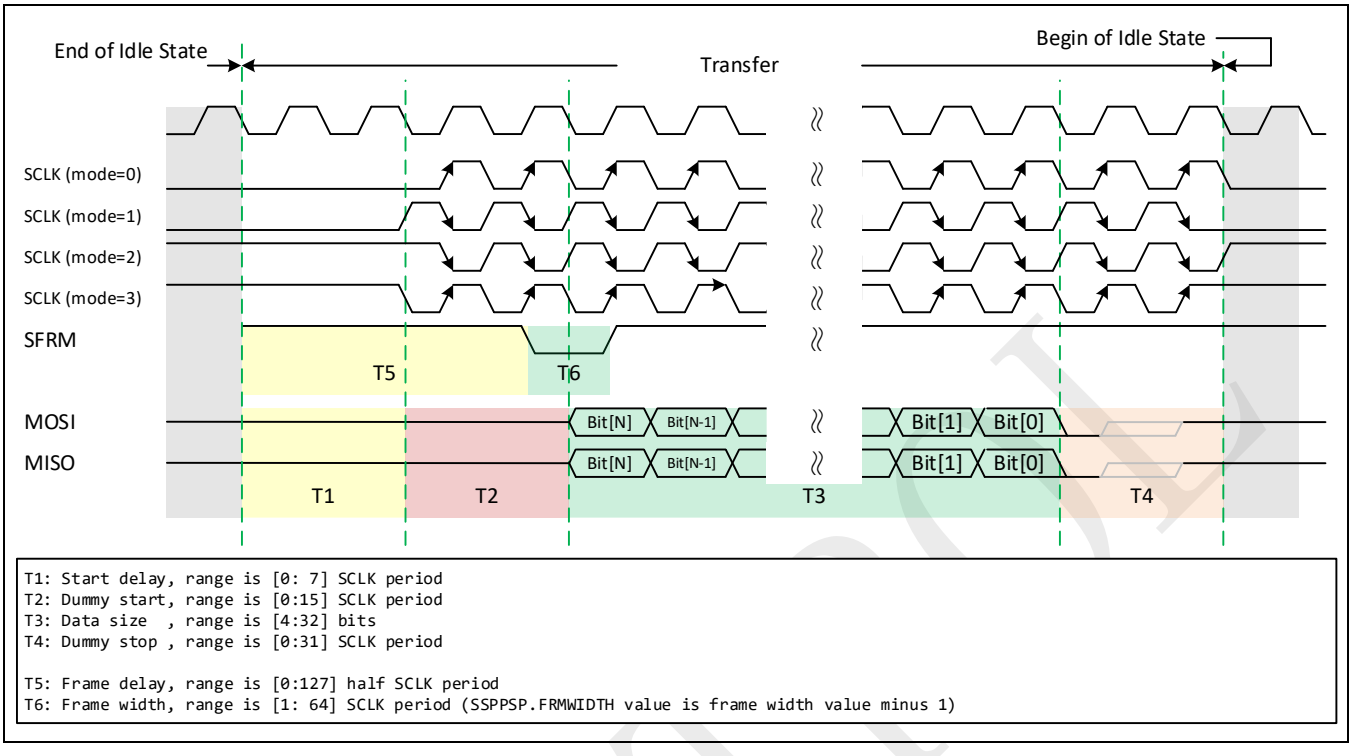
然后，SCLK 开始切换。TXD 将保持在其空闲状态，持续时间为 SPIPSP.DMYSTART 个串行时钟周期数。在 SPIPSP.FRMDLY 个半串行时钟周期数后，SFRM 将有效。SFRM 保持有效状态，持续时间为 SPISP.FRMWIDTH 个串行时钟周期数，然后变为无效。

可以通过将 SPICTL0.SIZE 设置为所选数据大小可设置每帧传输串行数据位宽为 8、16、18 或 32。一旦传输完成最后一位（LSB），SCLK 继续切换，持续时间为 SPIPSP.DMYSTOP 个串行时钟周期数。根据 SPIPSP.EDTS 的值，当 PSP 端口被禁用或复位时，TXD 要么保持最后一位传输的值，要么被强制为 0。

SPINTROL

©2025 旋智电子科技（上海）有限公司

图 20-10：PSP 协议（单次传输）



当 SFRM 有效时，接收数据会同时从外设驱动到 RXD 引脚，最高有效位（MSb）先传输。数据在基于所选的串行时钟模式（SPIPSP.CLKMODE）的 SCLK 边沿进行转换，并在相反的时钟边沿由 PSP 端口进行采样。

注意： PSP 模式不应配置如下：

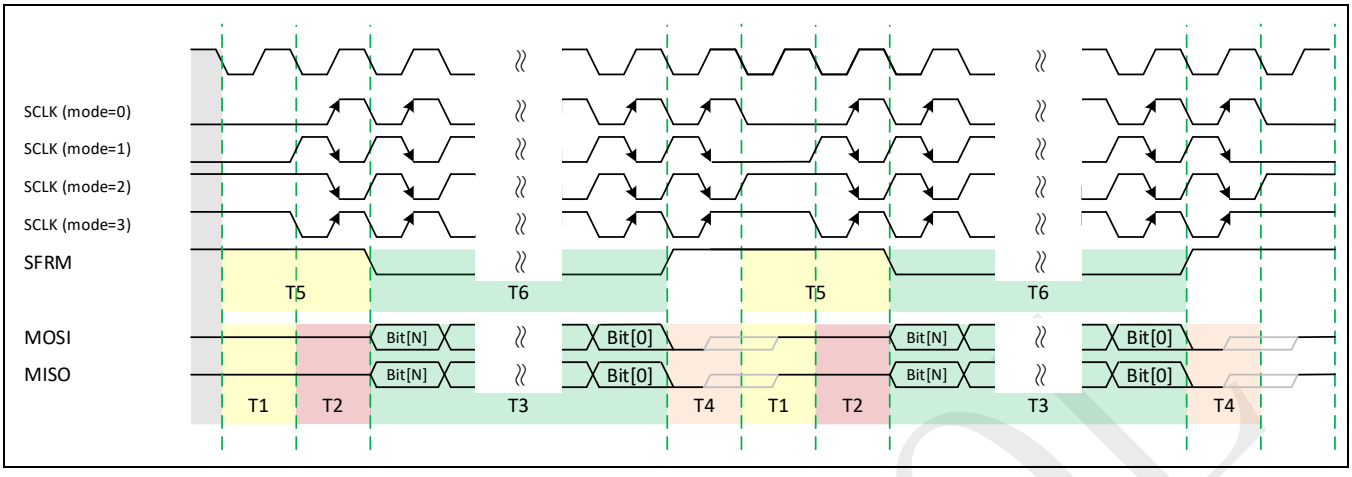
(T5 / 2 > (T1 + T2))，或者((T5 / 2 + T6) > (T1 + T2 + T3 + T4))，或者(T5 / 2 < T1)，或者((T2 == 0) && (T4 == 0) && (T6 >= SIZE))

如果配置 PSP 模式为以上参数，将通信将失败。

SFRM 延迟（T5）不能延伸到 T4 的结束之后。SFRM 宽度（T6）必须至少在一个 SCLK 周期内处于有效状态，并且应该在 T4 结束之前解除有效状态。

©2025 旋智电子科技（上海）有限公司

图 20-11：PSP 协议（连续传输）



SPINTROL

©2025 旋智电子科技（上海）有限公司

20.8 寄存器

20.8.1 SPI 寄存器列表

表 20-2：SPI 模块基地址

外设模块	基地址
SPI0	0x4000 1500
SPI1	0x4000 1700

表 20-3：SPI 寄存器列表

寄存器	偏移地址	描述	复位值
SPIDATA	0x0	SPI 数据寄存器	0x00000000
SPICTL0*	0x4	SPI 控制寄存器 0	0x00000000
SPICTL1*	0x8	SPI 控制寄存器 1	0x00000000
SPISTS	0xC	SPI 状态寄存器	0x00000003
SPIRFLVL	0x10	SPI 接收 FIFO 当前长度寄存器	0x00000000
SPITFLVL	0x14	SPI 发送 FIFO 当前长度寄存器	0x00000000
SPIRFTH*	0x18	SPI 接收 FIFO 长度阈值寄存器	0x00000000
SPITFTH*	0x1C	SPI 发送 FIFO 长度阈值寄存器	0x00000000
SPIRAWIF	0x20	SPI 中断原始事件寄存器	0x00000000
SPIIF	0x24	SPI 中断标志寄存器	0x00000000
SPIIC	0x28	SPI 中断清除寄存器	0x00000000
SPIIE*	0x2C	SPI 中断使能寄存器	0x00000000
SPITO*	0x30	SPI 超时寄存器	0x00000000
SPIPSP*	0x34	SPI 可编程串行协议寄存器	0x00000000
SPITCTL*	0x40	SPI 测试寄存器	0x00000000
SPIBDCNT*	0x44	SPI 波特率计数寄存器	0x00000000
SPIREGKEY	0x48	SPI 模块写使能寄存器	0x1ACCE551

注意： 由*标记的寄存器仅当 SPIREGKEY=0x1ACCE551 时才可以改写。

©2025 旋智电子科技（上海）有限公司

20.8.2 SPI 寄存器

表 20-4：SPI 数据寄存器（SPIDATA）位段定义

SPIDATA (SPI Data Register) Offset：0x0 Default：0x00000000 Access：SPIx->SPIDATA							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 20-5：SPI 数据寄存器（SPIDATA）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	发送或接收的数据 写本寄存器往发送 FIFO 放入数据； 读本寄存器从接收 FIFO 取出数据。

表 20-6：SPI 控制寄存器 0（SPICTL0）位段定义

SPICTL0 (SPI Control Register 0) Offset：0x4 Default：0x00000000 Access：SPIx->SPICTL0							
31	30	29	28	27	26	25	24
RESERVED						RESERVED	SWAPPIN
23	22	21	20	19	18	17	16
DMATXREQ	DMARXREQ	MWIREDS	FRMSLOT			MODE	TRIPHS
15	14	13	12	11	10	9	8
TRIEN	FPACK	FRMFMT		SIZE			
7	6	5	4	3	2	1	0
SIZE	RESERVED	SFRMINV	SPIMODE		SELSLV	SLVCKRUN	EN

表 20-7：SPI 控制寄存器 0（SPICTL0）位段描述

位段	位段名	属性	复位值	描述
31:26	RESERVED_31_26	RO	0x0	保留
25	RESERVED_25	RW	0x0	保留
24	SWAPPIN	RW	0x0	MOSI 和 MISO 互换 0: MOSI 用于主设备数据输出和从设备数据输入

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				MISO 用于主设备数据输入和从设备数据输出 1： MOSI 用于主设备数据输入和从设备数据输出 MISO 用于主设备数据输出和从设备数据输入
23	DMATXREQ	RW	0x0	DMA 发送请求使能 0：关闭 1：使能
22	DMARXREQ	RW	0x0	DMA 接收请求使能 0：关闭 1：使能
21	MWIREDS	RW	0x0	Microwire 传输命令数据大小 0：传输 8 位命令 1：传输 16 位命令
20:18	FRMSLOT	RW	0x0	帧时隙 仅在网络模式下使用 指定每帧的时隙数（实际的时隙数为该字段值加 1，因此可以指定 1 至 8 个时隙）
17	MODE	RW	0x0	模式选择 0：SPI 模式 1：PSP 格式的网络模式
16	TRIPHS	RW	0x0	在最后阶段使 STXD 处于三态使能 0：在最低有效位（LSB）开始后的 1/2 个时钟周期后，STXD 处于三态状态。 注意：SPI/MWIRE 帧格式必须为 0b0 1：在结束 LSB 的时钟边沿上，STXD 处于三态状态
15	TRIEN	RW	0x0	STXD 三态使能 0：STXD 非三态 1：当不发送数据时 STXD 为三态
14	FPACK	RW	0x0	FIFO Packing 模式使能 0：FIFO packing 模式关闭 1：FIFO packing 模式使能
13:12	FRMFMT	RW	0x0	帧格式 00：Motorola 串行外围接口（SPI） 01：德州仪器同步串行协议（SSP） 10：National Semiconductor Microwire 11：可编程串行协议（PSP）
11:7	SIZE	RW	0x0	数据大小选择（最小值为 3）

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
6	RESERVED_6	RW	0x0	保留
5	SFRMINV	RW	0x0	反转帧信号 0：SFRM 极性由 PSP 极性位确定 1：SFRM 与正常 SFRM 相反（根据 PSP 极性位定义）
4:3	SPIMODE	RW	0x0	SPI 时钟模式选择（仅用于 SPI 帧格式） 00：PHS = 0, POL = 0 01：PHS = 0, POL = 1 10：PHS = 1, POL = 0 11：PHS = 1, POL = 1
2	SELSLV	RW	0x0	选择从机模式 0：主机模式，SPI 口负责驱动 SFRM 以及 SCLK 1：从机模式，SPI 口接收 SFRM 以及 SCLK 信号
1	SLVCKRUN	RW	0x0	从模式下的时钟自由运行 只用于从模式。 0：仅在数据传输期间将时钟输入到 SCLK 1：时钟输入到 SCLK 持续运行。在 SPI 格式中必须启用
0	EN	RW	0x0	同步串口使能 0：关闭 SPI 1：使能 SPI

表 20-8：SPI 控制寄存器 1（SPICTL1）位段定义

SPICTL1 (SPI Control Register 1)　Offset：0x8　Default：0x00000000
Access：SPIx->SPICTL1

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED			RESERVED	RESERVED	RESERVED	RESERVED	RESERVED
15	14	13	12	11	10	9	8
RESERVED	RESERVED	RESERVED	RESERVED				
7	6	5	4	3	2	1	0
RESERVED							SFRMLCK

©2025 旋智电子科技（上海）有限公司

表 20-9：SPI 控制寄存器 1（SPICTL1）位段描述

位段	位段名	属性	复位值	描述
31:21	RESERVED_31_21	RO	0x0	保留
20	RESERVED_20	RW	0x0	保留
19	RESERVED_19	RW	0x0	保留
18	RESERVED_18	RW	0x0	保留
17	RESERVED_17	RW	0x0	保留
16	RESERVED_16	RW	0x0	保留
15	RESERVED_15	RW	0x0	保留
14	RESERVED_14	RW	0x0	保留
13	RESERVED_13	RW	0x0	保留
12:1	RESERVED_12_1	RW	0x0	保留
0	SFRMLCK	RW	0x0	SFRM 信号锁定使能 仅适用于 SPI 格式，主机模式。 0：禁用 SFRM 锁定状态 1：启用 SFRM 锁定功能。在 SPI 开始发送数据后，SFRM 将被锁定在活动电平上，然后需要手动清除该位以释放被锁定的 SFRM 到正常状态。

表 20-10：SPI 状态寄存器（SPISTS）位段定义

SPISTS (SPI Status Register)　Offset：0xC　Default：0x00000003
Access：SPIx->SPISTS

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED			SFRMLCK	NETBUSY	ACTSLOT		
7	6	5	4	3	2	1	0
RFEVEN	TFEVEN	SLVCLK	BUSY	RXNE	RXFULL	TXEMPTY	TXNF

表 20-11：SPI 状态寄存器（SPISTS）位段描述

位段	位段名	属性	复位值	描述
31:13	RESERVED_31_13	RO	0x0	保留
12	SFRMLCK	RO	0x0	SFRM 锁定 仅 SPI 格式有效 0：SFRM 没有锁定 1：SFRM 已锁定

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
11	NETBUSY	RO	0x0	网络模式忙碌 0：SPI 端口处于网络模式，并且当前没有活动的帧 1：SPI 端口处于网络模式，并且当前有活动的帧
10:8	ACTSLOT	RO	0x0	当前活动的时间时隙 由于 SPI 端口的 SCLK 域和内部总线时钟域之间的同步，该字段的值在当前活动时隙的开始和结束之间变为 s 表。 000：时隙 0 处于活动状态 001：时隙 1 处于活动状态 010：时隙 2 处于活动状态 011：时隙 3 处于活动状态 100：时隙 4 处于活动状态 101：时隙 5 处于活动状态 110：时隙 6 处于活动状态 111：时隙 7 处于活动状态
7	RFEVEN	RO	0x0	RXFIFO 偶数样本状态 只有在启用 FIFO Packing 时，才需要查看此位。否则，此位为 0。 0：RXFIFO 条目包含奇数样本 1：RXFIFO 条目包含偶数样本
6	TFEVEN	RO	0x0	TXFIFO 偶数样本状态 只有在启用 FIFO Packing 时才需要查看此位。否则，此位为 0。 0：TXFIFO 条目包含奇数个样本 1：TXFIFO 条目包含偶数个样本
5	SLVCLK	RO	0x0	时钟同步状态 只有当 SPIx 端口作为从设备或者处于特殊 FIFO 测试模式时，软件才需要读取这个位。在允许外部设备触发外部 SFRM 或 SCLK 信号之前，软件应等待该位清零。 0：SPIx 端口准备好进行从设备时钟操作。 1：SPIx 端口当前正忙于同步从设备模式信号。

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
4	BUSY	RO	0x0	SPI 忙碌 0：SPIx 端口处于空闲或禁用状态 1：SPIx 端口当前正在传输或接收帧数据
3	RXNE	RO	0x0	接收 FIFO 非空 0：RXFIFO 为空 1：RXFIFO 不为空
2	RXFULL	RO	0x0	接收 FIFO 满 0：接收 FIFO 中包含空位置 1：接收 FIFO 已完全填满
1	TXEMPTY	RO	0x1	发送 FIFO 非满 0：Transmit FIFO 中包含一个或多个数据条目 1：Transmit FIFO 完全为空。 但是可能还有数据在传输移位寄存器中，并且传输仍在进行中。
0	TXNF	RO	0x1	发送 FIFO 非满 0：TXFIFO 已满 1：TXFIFO 未满

表 20-12：SPI 接收 FIFO 当前长度寄存器（SPIRFLVL）位段定义

SPIRFLVL (SPI Receive FIFO Level Register)　Offset：0x10　Default：0x00000000
Access：SPIx->SPIRFLVL

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED			VAL				

表 20-13：SPI 接收 FIFO 当前长度寄存器（SPIRFLVL）位段描述

位段	位段名	属性	复位值	描述
31:5	RESERVED_31_5	RO	0x0	保留
4:0	VAL	RO	0x0	接收 FIFO 中有效数据个数

©2025 旋智电子科技（上海）有限公司

表 20-14：SPI 发送 FIFO 当前长度寄存器（SPITFLVL）位段定义

SPITFLVL (SPI Transmit FIFO Level Register) Offset：0x14 Default：0x00000000 Access：SPIx->SPITFLVL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED			VAL				

表 20-15：SPI 发送 FIFO 当前长度寄存器（SPITFLVL）位段描述

位段	位段名	属性	复位值	描述
31:5	RESERVED_31_5	RO	0x0	保留
4:0	VAL	RO	0x0	发送 FIFO 中有效数据个数

表 20-16：SPI 接收 FIFO 阈值寄存器（SPIRFTH）位段定义

SPIRFTH (SPI Receive FIFO Threshold Register) Offset：0x18 Default：0x00000000 Access：SPIx->SPIRFTH							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				VAL			

表 20-17：SPI 接收 FIFO 阈值寄存器（SPIRFTH）位段描述

位段	位段名	属性	复位值	描述
31:4	RESERVED_31_4	RO	0x0	保留
3:0	VAL	RW	0x0	当 SPIRFLVL 大于 SPITRFTH 时触发接收数据可用中断

©2025 旋智电子科技（上海）有限公司

表 20-18：SPI 发送 FIFO 阈值寄存器（SPITFTH）位段定义

SPITFTH (SPI Transmit FIFO Threshold Register) Offset：0x1C Default：0x00000000 Access：SPIx->SPITFTH							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				VAL			

表 20-19：SPI 发送 FIFO 阈值寄存器（SPITFTH）位段描述

位段	位段名	属性	复位值	描述
31:4	RESERVED_31_4	RO	0x0	保留
3:0	VAL	RW	0x0	当 SPITFLVL 大于 SPITFTH 时触发发送数据中断

表 20-20：SPI 中断原始标志寄存器（SPIRAWIF）位段定义

SPIRAWIF (SPI Interrupt Raw Flag Register) Offset：0x20 Default：0x00000000 Access：SPIx->SPIRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						RESERVED	RESERVED
7	6	5	4	3	2	1	0
BITCNTERR	TXREQ	TXOV	TXUD	TO	RXREQ	RXOV	RXUD

表 20-21：SPI 中断原始标志寄存器（SPIRAWIF）位段描述

位段	位段名	属性	复位值	描述
31:10	RESERVED_31_10	RO	0x0	保留
9	RESERVED_9	RO	0x0	保留
8	RESERVED_8	RO	0x0	保留
7	BITCNTERR	RO	0x0	比特计数错误中断标志 0：未发生比特计数错误 1：发生了比特计数错误
6	TXREQ	RO	0x0	发送 FIFO 服务请求中断标志 0：TXFIFO 未达到触发阈值 1：TXFIFO 达到触发阈值

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
5	TXOV	RO	0x0	发送 FIFO 溢出中断标志 0：TXFIFO 未发生溢出 1：TXFIFO 发生溢出
4	TXUD	RO	0x0	发送 FIFO 欠流中断标志 0：TXFIFO 未发生欠流 1：TXFIFO 发生欠流
3	TO	RO	0x0	接收超时中断标志 0：未发生超时 1：发生了超时
2	RXREQ	RO	0x0	接收 FIFO 服务请求中断标志 0：RXFIFO 未达到触发阈值 1：RXFIFO 达到触发阈值
1	RXOV	RO	0x0	接收 FIFO 溢出中断标志 0：RXFIFO 未发生溢出 1：RXFIFO 发生溢出
0	RXUD	RO	0x0	接收 FIFO 欠流中断标志 0：RXFIFO 未发生欠流 1：RXFIFO 发生欠流

表 20-22：SPI 中断标志寄存器（SPIIF）位段定义

SPIIF (SPI Interrupt Flag Register)　Offset：0x24　Default：0x00000000
Access：SPIx->SPIIF

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED					INT	RESERVED	RESERVED
7	6	5	4	3	2	1	0
BITCNTERR	TXREQ	TXOV	TXUD	TO	RXREQ	RXOV	RXUD

表 20-23：SPI 中断标志寄存器（SPIIF）位段描述

位段	位段名	属性	复位值	描述
31:11	RESERVED_31_11	RO	0x0	保留
10	INT	RO	0x0	全局中断标志 0：未发生中断 1：发生了某些中断
9	RESERVED_9	RO	0x0	保留
8	RESERVED_8	RO	0x0	保留

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
7	BITCNTERR	RO	0x0	比特计数错误中断标志 0：未发生比特计数错误 1：发生了比特计数错误
6	TXREQ	RO	0x0	发送 FIFO 服务请求中断标志 0：TXFIFO 未达到触发阈值 1：TXFIFO 达到触发阈值
5	TXOV	RO	0x0	发送 FIFO 溢出中断标志 0：TXFIFO 未发生溢出 1：TXFIFO 发生溢出
4	TXUD	RO	0x0	发送 FIFO 欠流中断标志 0：TXFIFO 未发生欠流 1：TXFIFO 发生欠流
3	TO	RO	0x0	接收超时中断标志 0：未发生超时 1：发生了超时
2	RXREQ	RO	0x0	接收 FIFO 服务请求中断标志 0：RXFIFO 未达到触发阈值 1：RXFIFO 达到触发阈值
1	RXOV	RO	0x0	接收 FIFO 溢出中断标志 0：RXFIFO 未发生溢出 1：RXFIFO 发生溢出
0	RXUD	RO	0x0	接收 FIFO 欠流中断标志 0：RXFIFO 未发生欠流 1：RXFIFO 发生欠流

表 20-24：SPI 中断清除寄存器（SPIIC）位段定义

SPIIC (SPI Interrupt Clear Register)　Offset：0x28　Default：0x00000000
Access：SPIx->SPIIC

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED					INT	RESERVED	RESERVED
7	6	5	4	3	2	1	0
BITCNTERR	RESERVED	TXOV	TXUD	TO	RESERVED	RXOV	RXUD

©2025 旋智电子科技（上海）有限公司

表 20-25：SPI 中断清除寄存器（SPIIC）位段描述

位段	位段名	属性	复位值	描述
31:11	RESERVED_31_11	RO	0x0	保留
10	INT	W1C	0x0	全局中断清除 0：写入 0 无效，读取始终为 0 1：写入 1 清除该标志。该位会自动清除。
9	RESERVED_9	W1C	0x0	保留
8	RESERVED_8	W1C	0x0	保留
7	BITCNTERR	W1C	0x0	比特计数错误中断清除 0：写入 0 无效，读取始终为 0 1：写入 1 清除该标志。该位会自动清除。
6	RESERVED_6	RO	0x0	保留
5	TXOV	W1C	0x0	发送 FIFO 溢出中断清除 0：写入 0 无效，读取始终为 0 1：写入 1 清除该标志。该位会自动清除。
4	TXUD	W1C	0x0	发送 FIFO 欠流中断清除 0：写入 0 无效，读取始终为 0 1：写入 1 清除该标志。该位会自动清除。
3	TO	W1C	0x0	接收超时中断清除 0：写入 0 无效，读取始终为 0 1：写入 1 清除该标志。该位会自动清除。
2	RESERVED_2	RO	0x0	保留
1	RXOV	W1C	0x0	接收 FIFO 溢出中断清除 0：写入 0 无效，读取始终为 0 1：写入 1 清除该标志。该位会自动清除。
0	RXUD	W1C	0x0	接收 FIFO 欠流中断清除 0：写入 0 无效，读取始终为 0 1：写入 1 清除该标志。该位会自动清除。

表 20-26：SPI 中断使能寄存器（SPIIE）位段定义

SPIIE (SPI Interrupt Enable Register) Offset：0x2C Default：0x00000000 Access：SPIx->SPIIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						RESERVED	RESERVED
7	6	5	4	3	2	1	0
BITCNTERR	TXREQ	TXOV	TXUD	TO	RXREQ	RXOV	RXUD

©2025 旋智电子科技（上海）有限公司

表 20-27：SPI 中断使能寄存器（SPIIE）位段描述

位段	位段名	属性	复位值	描述
31:10	RESERVED_31_10	RO	0x0	保留
9	RESERVED_9	RW	0x0	保留
8	RESERVED_8	RW	0x0	保留
7	BITCNTERR	RW	0x0	比特计数错误中断使能 0：禁用 1：启用
6	TXREQ	RW	0x0	发送 FIFO 服务请求中断使能 0：禁用 1：启用
5	TXOV	RW	0x0	发送 FIFO 溢出中断使能 0：禁用 1：启用
4	TXUD	RW	0x0	发送 FIFO 欠流中断使能 0：禁用 1：启用
3	TO	RW	0x0	接收超时中断使能 0：禁用 1：启用
2	RXREQ	RW	0x0	接收 FIFO 服务请求中断使能 0：禁用 1：启用
1	RXOV	RW	0x0	接收 FIFO 溢出中断使能 0：禁用 1：启用
0	RXUD	RW	0x0	接收 FIFO 欠流中断使能 0：禁用 1：启用

表 20-28：SPI 超时寄存器（SPITO）位段定义

SPITO (SPI Time Out Register)　Offset：0x30　Default：0x00000000 Access：SPIx->SPITO							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

©2025 旋智电子科技（上海）有限公司

表 20-29：SPI 超时寄存器（SPITO）位段描述

位段	位段名	属性	复位值	描述
31:24	RESERVED_31_24	RO	0x0	保留
23:0	VAL	RW	0x0	超时间隔定义为 t_CLK_SPI * (VAL+1)

表 20-30：SPI 可编程串行协议寄存器（SPIPSP）位段定义

SPIPSP (SPI Programmable Serial Protocol Register)　Offset：0x34　Default：0x00000000
Access：SPIx->SPIPSP

31	30	29	28	27	26	25	24
RESERVED	DMYSTOP					SYNC	RESERVED
23	22	21	20	19	18	17	16
FRMWIDTH						FRMDLY	
15	14	13	12	11	10	9	8
FRMDLY					DMYSTART		
7	6	5	4	3	2	1	0
DMYSTART	STARTDLY			ETDS	SFRMPOL	CLKMODE	

表 20-31：SPI 可编程串行协议寄存器（SPIPSP）位段描述

位段	位段名	属性	复位值	描述
31	RESERVED_31	RO	0x0	保留
30:26	DMYSTOP	RW	0x0	Dummy stop（虚拟停止） 仅在 PSP 帧格式中使用。 在 PSP 网络模式下不要使用，且在网络模式下必须为 0。
25	SYNC	RW	0x0	帧同步相对时序位 仅在 PSP 帧格式中使用。 0：在 DMYSTOP 时序结束后，下一帧被断言。 1：与前一帧的 LSB 同时被断言。
24	RESERVED_24	RO	0x0	保留
23:18	FRMWIDTH	RW	0x0	串行帧宽度 仅在 PSP 帧格式中使用。 仅在 SPI 主模式下有效。
17:11	FRMDLY	RW	0x0	串行帧延迟 仅在 PSP 帧格式中使用。 在 PSP 网络模式下不要使用，且在网络模式下必须为 0。
10:7	DMYSTART	RW	0x0	Dummy Start（虚拟启动） 仅在 PSP 帧格式中使用。

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				在 PSP 网络模式下不要使用，且在网络模式下必须为 0。
6:4	STARTDLY	RW	0x0	启动延迟 仅在 PSP 帧格式中使用。 在 PSP 网络模式下不要使用，且在网络模式下必须为 0。
3	ETDS	RW	0x0	传输结束数据状态 仅在 PSP 帧格式中使用。 0：低电平 1：LSB 值的电平
2	SFRMPOL	RW	0x0	串行帧极性 0：SFRM 为低电平活动（0b0） 1：SFRM 为高电平活动（0b1）
1:0	CLKMODE	RW	0x0	串行比特率时钟模式 仅在 PSP 帧格式中使用。 00：数据驱动（下降沿），数据采样（上升沿），空闲状态（低电平） 01：数据驱动（上升沿），数据采样（下降沿），空闲状态（低电平） 10：数据驱动（上升沿），数据采样（下降沿），空闲状态（高电平） 11：数据驱动（下降沿），数据采样（上升沿），空闲状态（高电平）

表 20-32：SPI 测试控制寄存器（SPITCTL）位段定义

SPITCTL (SPI Test Control Register) Offset：0x40 Default：0x00000000 Access：SPIx->SPITCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 RESERVED	6 RESERVED	5 RESERVED	4 RESERVED	3 RESERVED	2 RESERVED	1 RESERVED	0 LOOP

表 20-33：SPI 测试控制寄存器（SPITCTL）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6	RESERVED_6	RW	0x0	保留
5	RESERVED_5	RW	0x0	保留

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
4	RESERVED_4	RW	0x0	保留
3	RESERVED_3	RW	0x0	保留
2	RESERVED_2	RW	0x0	保留
1	RESERVED_1	RW	0x0	保留
0	LOOP	RW	0x0	环回模式 0：启用正常的串行端口操作。 1：TX 串行移位器的输出与 RX 串行移位器的输入内部连接。

表 20-34：SPI 波特率计数寄存器（SPIBDCNT）位段定义

SPIBDCNT (SPI Baud Count Register) Offset：0x44 Default：0x00000000 Access：SPIx->SPIBDCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 20-35：SPI 波特率计数寄存器（SPIBDCNT）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x0	SPI 波特率将为 SPI 时钟 / (SPIBDCNT + 1)。

表 20-36：SPI 模块写使能寄存器（SPIREGKEY）位段定义

SPIREGKEY (SPI Register Write-Allow Key Register) Offset：0x48 Default：0x1ACCE551 Access：SPIx->SPIREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

©2025 旋智电子科技（上海）有限公司

表 20-37：SPI 模块写使能寄存器（SPIREGKEY）位段描述

位段	位段名	属性	复位值	描述
31:0	KEY	RW	0x1ACCE551	写入 0x1ACCE551 解锁受保护的 SPI 寄存器

SPINTROL

21 内部集成电路总线（I2C）

21.1 概述

I2C 模块为 MCU 和任何通过 I2C 串行总线相连的 I2C 总线兼容设备之间提供了一个接口。连接在 I2C 总线上的外部部件，通过两线 I2C 接口能够串行地向 MCU 发送数据或者从 MCU 接收数据。

I2C 总线是多主机总线，因而，I2C 模块支持多主机模式，允许多个能够控制总线的设备与其连接。每个 I2C 设备可以通过其唯一的地址来识别，而且根据其功能，I2C 设备既可以作为发送器，也可以作为接收器。除了作为发送器或接收器之外，连接在 I2C 总线上的设备在执行数据传输时也可以被视为主机或者从机。需要注意的是，主机是在总线上发起数据传输并生成时钟信号以允许该数据传输的设备。在数据传输过程中，任何被这个主机寻址的设备都被视为从机。

21.2 特性

SPC1169 实现了一个硬件 I2C 模块，其特性如下：

- 兼容 I2C 标准 2.1
- 支持四种速度模式：标准模式（100kb/s），快速模式（400kb/s），快速模式 plus（Fast mode plus）（1Mb /s）和高速模式（3.4Mb/s）
- 时钟同步
- 支持 7 位或者 10 位地址寻址
- 支持 7 位或者 10 位地址模式下的混合格式传输
- 两个独立的发送和接收 FIFO，每个 FIFO 为 16 x 32 位

21.3 I2C 端口信号描述

表 21-1：I2C 端口信号描述

信号名称	类型	描述
I2C_SCL	输入/输出	I2C serial clock line（漏极开路）
I2C_SDA	输入/输出	I2C serial data line（漏极开路）

21.4 I2C 功能描述

I2C 模块通过软件可以被配置为：

- 仅 I2C 主机模式，与其他 I2C 主机通信
- 仅 I2C 从机模式，与一个以上 I2C 主机通信

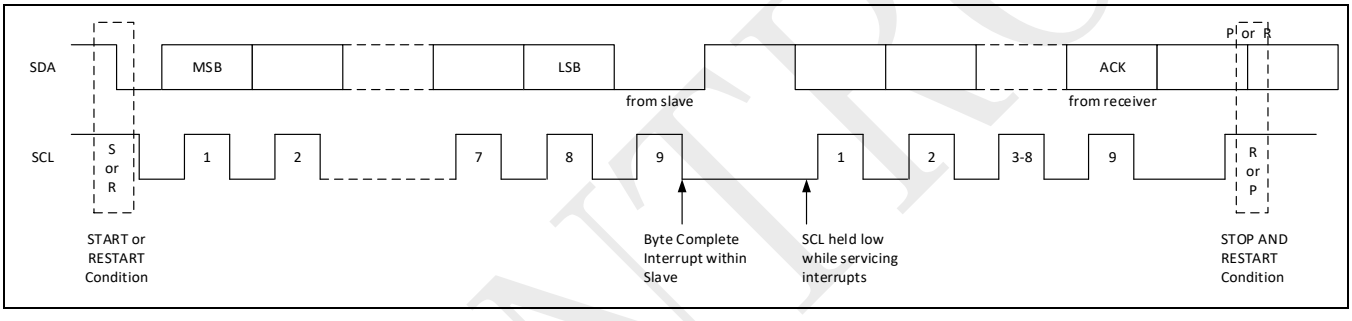
©2025 旋智电子科技（上海）有限公司

主机负责生成时钟并控制数据传输。从机负责向主机发送数据或从主机接收数据。数据的应答由接收数据的设备发送，该设备可以是主机，也可以是从机。如前所述，I2C 协议也允许多个主机存在于 I2C 总线上，并采用仲裁程序来确定总线所有权。

每个从机都有一个由系统设计者确定的唯一的地址。当一个主机想要和一个从机通信时，主机会发送一个开始/重启（START/RESTART）信号，紧接着是从机的地址和一个控制位（读/写），该控制位用以确定主机是向从机传输数据还是从从机接收数据。然后，从机在地址之后发送一个应答（ACK）脉冲。

假如主机（发送器）向从机（接收器）写，接收器接收一个字节的数据；该传输持续到主机通过停止（STOP）信号来终止传输。假如主机（接收器）从从机读，从机（发送器）向主机发送一个字节的数据；随后主机使用 ACK 信号来应答，此次传输将继续进行，直到主机在接收到最后一个字节后使用 NACK 信号来终止传输。最后主机发出停止（STOP）信号或者在发出重启（RESTART）信号后寻址另外一个从机设备。图 21-1 描述了上述行为。

图 21-1：I2C 总线数据传输



I2C 是一个同步串行接口。SDA 线是一个双向信号线，只有当 SCL 线为低电平时才会改变（产生 STOP、START 和 RESTART 信号除外）。输出驱动器是漏极开路或者集电极开路，从而可以在总线上实现“线与”功能。总线可以承载的设备最大数量仅受 400pF 的最大电容规格限制。数据以字节包的形式传输。

注意： 向 FIFO 中放入数据会产生一个 START 信号，清空 FIFO 会产生一个 STOP 信号。更多信息，请参考章节 21.4.1。

21.4.1 START 和 STOP 信号的产生

当 I2C 模块作为主机运行时，将数据写入发送 FIFO 会导致在 I2C 总线上产生一个 START 信号，而发送 FIFO 为空时则会导致在 I2C 总线上产生一个 STOP 信号。

当作为从机运行时，根据 I2C 协议，I2C 模块不会产生 START 和 STOP 信号。然而，如果接收到一个读请求，该 I2C 模块会将 SCL 线拉低直道要读取的数据已经提供给它。此操作会拖住 I2C 总线，直到从机开始提供要读取的数据或者通过将 I2CENABLE.EN 寄存器位清零来关闭 I2C 从机。

©2025 旋智电子科技（上海）有限公司

21.4.2 混合格式传输

SPC1169 I2C 模块支持在 7 位和 10 位寻址模式下的读/写交替传输。但是，不支持混合寻址模式下的传输，即 7 位寻址模式下传输数据后，紧接着进行 10 位寻址模式的数据传输，反之亦然。

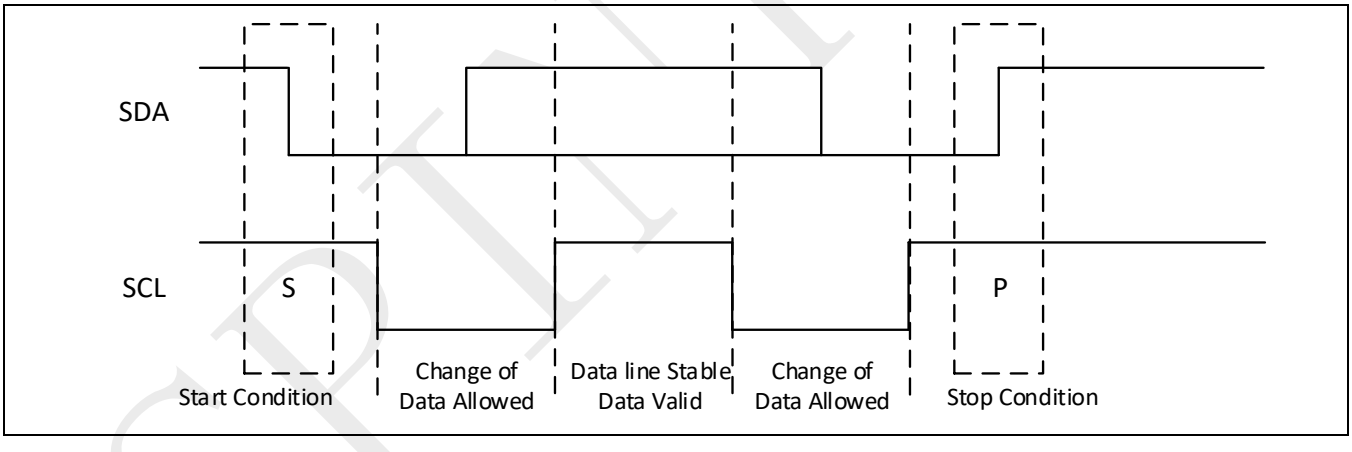
在主机模式下，通过置“1”I2CCTL.RESTARTEN 寄存器位，来初始化混合格式的传输。当完成一次传输时，I2C 模块会检查发送 FIFO 并执行下一次传输。若当前传输方向与上一次传输不同不同，则混合格式传输会被用来启动本次传输。如果当前 I2C 传输完成时，发送 FIFO 为空，则会发出一个 STOP 信号；下一次传输则会在一个 START 信号后面启动。

21.5 I2C 协议

21.5.1 START 和 STOP 信号

当总线空闲时，SCL 和 SDA 信号线都通过总线上的外部上拉电阻拉高。当主机想要在总线上开始一次传输时，主机会发出 START 信号。START 信号被定义为当 SCL 为高电平“1”时，SDA 信号由高电平变到低电平；当主机必须终止传输时，它会发出一个 STOP 信号。STOP 信号被定义为当 SCL 为高电平“1”时，SDA 信号由低电平变到高电平。图 21-2 展示了 START 和 STOP 信号的时序。当数据在总线上被被传输时，在 SCL 为“1”时，SDA 线必须保持稳定。

图 21-2：START 和 STOP 信号



21.5.2 地址格式

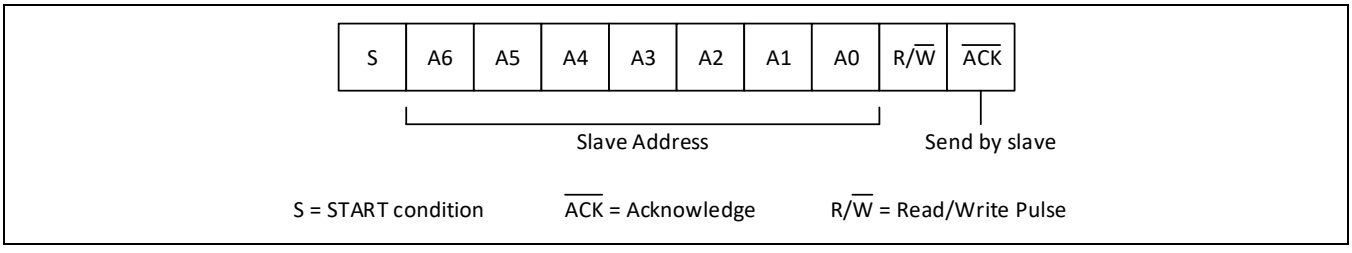
2C 模块支持两种寻址格式：7 位地址和 10 位地址格式。

7 位地址格式

在 7 位寻址格式中，第一个字节的前 7 位（位[7:1]）设置寻址地址，最后一位（位 0）是读写控制位，如图 21-3 所示。当位 0（R/W）被设置为“0”时，意味着主机向从机写数据；当位 0（R/W）被设置为“1”时，意味着主机从从机读数据。

©2025 旋智电子科技（上海）有限公司

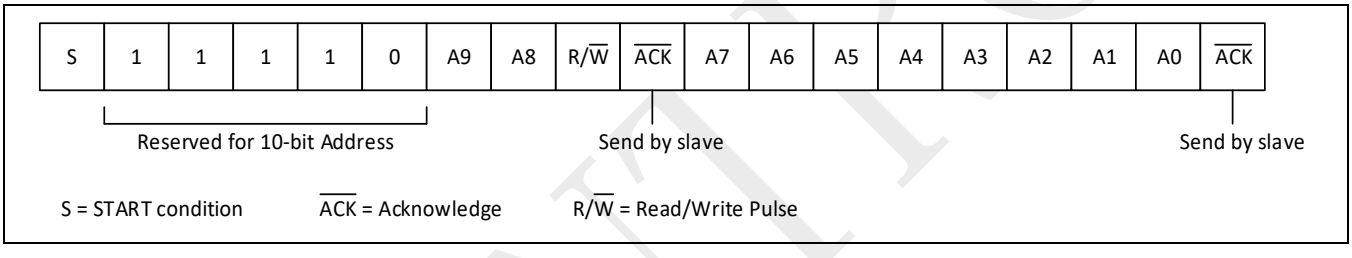
图 21-3：7 地址格式格式



10 位地址格式

在 10 位寻址格式中，10 位地址被分成两次传输。第一个被传输的字节的位定义为：前五位（位[7:3]）用于通知从机这是一个 10 位地址传输，接下来的俩位（位[2:1]）用于设置从机地址的位[9:8]，最后一位（位 0）是读/写控制位。第二个被传输的字节用来设置从机地址的位[7:0]。10 位地址格式如图 21-4 所示。

图 21-4：10 地址格式



21.5.3 收发协议

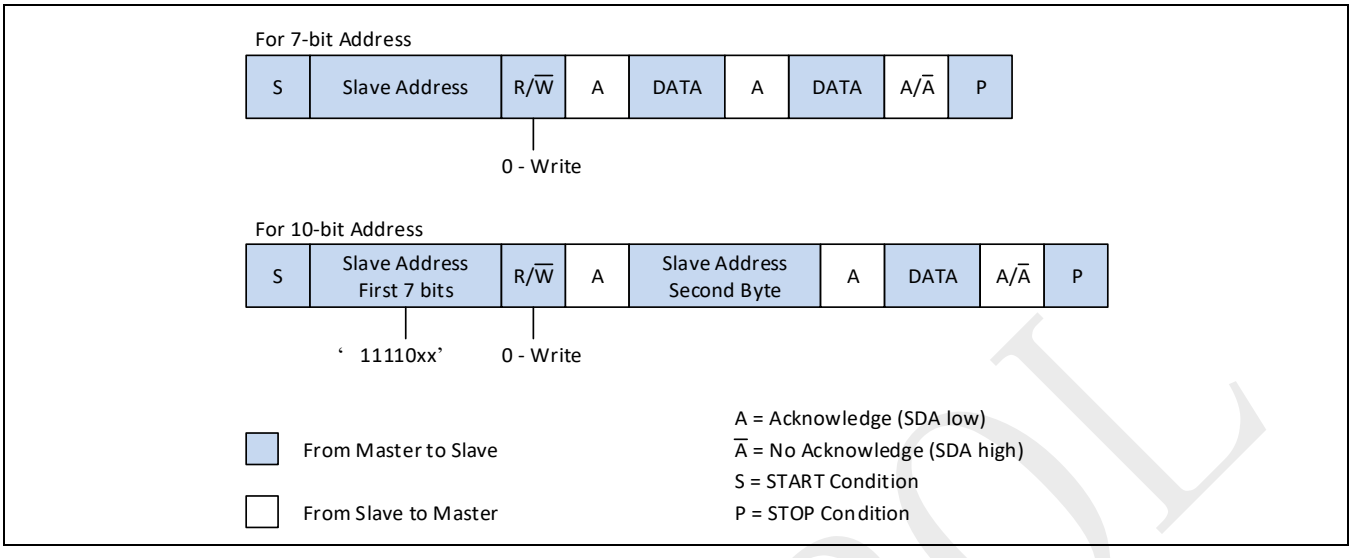
主机可以发起向总线传输数据（主机作为发送器）或者从总线接收数据（主机作为接收器）。从机负责响应来至主机的请求，向总线发送数据（从机作为发送器）或者从总线接收数据（从机作为接收器）。

主机-发送器和从机-接收器

所有的数据以字节为单位进行传输，每次数据传输的字节数没有限制。在主机发送地址和读写控制位或者主机向从从机发送一个字节数据后，从机-接收器必须用 ACK 信号响应。当从机-接收器用 ACK 脉冲信号响应时，主机会发出一个 STOP 信号中止传输；于此同时，从机必须将 SDA 线保持在高电平，以便主机可以中止传输。

©2025 旋智电子科技（上海）有限公司

图 21-5：主机-发送协议

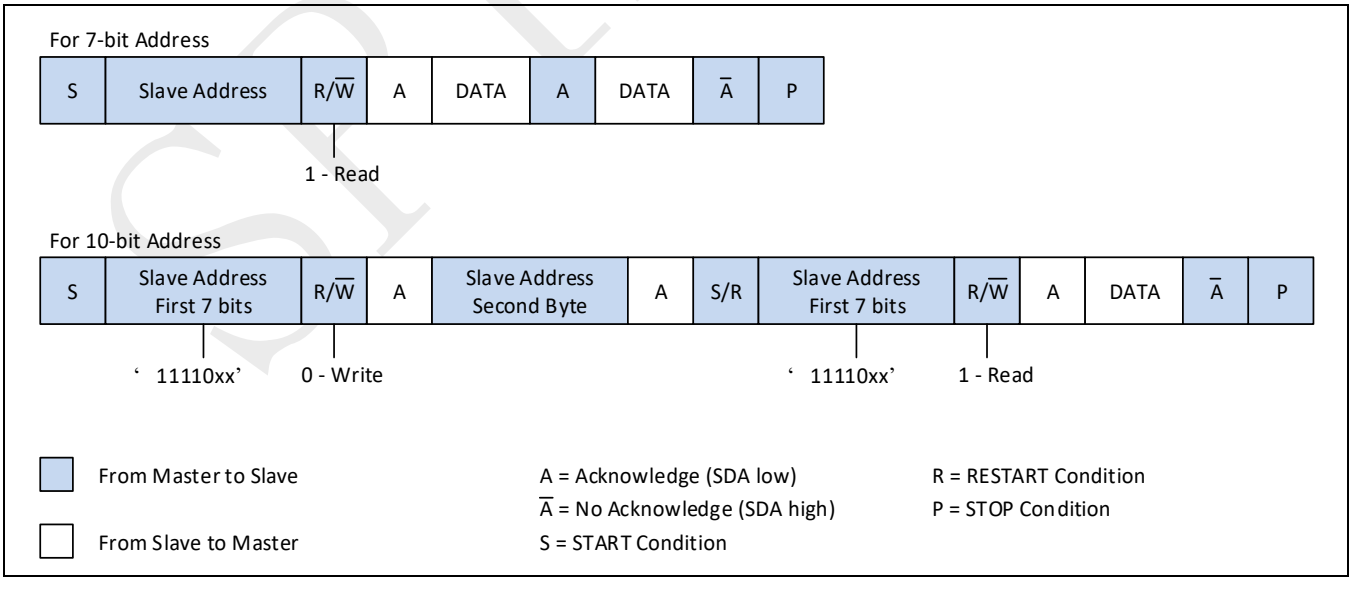


主机-接收器和从机-发送器

假如主机接收数据如图 21-6 所示的那样，那么主机在接收到一个字节的数据（最后一个字节除外）后，会向从机-发送器发送一个 ACK 脉冲信号作为响应。主机-接收器利用这种方式通知从机-发送器这是最后一个字节数据。在检测到 NACK 信号后，从机-发送器会放弃 SDA 线，以便主机可以发出一个 STOP 信号。

当主机拒绝通过发送一个 STOP 信号来放弃总线时，主机可以发出一个 RESTART 信号。这个 RESTART 信号和 START 信号是等同的，唯一的区别是 RESTART 信号是在 ACK 脉冲信号后发生的。此时，如果工作在主机模式，那么 I2C 模块可以继续和同一个从机进行通信而且可以改变传输方向。

图 21-6：主机-接收协议



©2025 旋智电子科技（上海）有限公司

21.6 寄存器

21.6.1 I2C 寄存器列表

表 21-2：I2C 模块基地址

外设模块	基地址
I2C0	0x4000 3500

表 21-3：I2C 寄存器列表

寄存器	偏移地址	描述	复位值
I2CCTL*	0x0	I2C 控制寄存器	0x00000062
I2CMADDR	0x4	I2C 主机寻址寄存器	0x00090055
I2CSADDR*	0x8	I2C 从机地址寄存器	0x00000055
I2CDATACMD	0xC	I2C 数据和指令寄存器	0x00000000
I2CSSLCNT*	0x10	I2C 标准速度模式时钟低电平计数寄存器	0x0000024C
I2CSSHCNT*	0x14	I2C 标准速度模式时钟高电平计数寄存器	0x000001F4
I2CFSLCNT*	0x18	I2C 快速速度模式时钟低电平计数寄存器	0x000000A3
I2CFSHCNT*	0x1C	I2C 快速速度模式时钟高电平计数寄存器	0x0000004B
I2CHSLCNT*	0x20	I2C 高速速度模式时钟低电平计数寄存器	0x00000014
I2CHSHCNT*	0x24	I2C 高速速度模式时钟高电平计数寄存器	0x00000008
I2CFSSPKLEN*	0x28	I2C 快速模式尖峰抑制阈值寄存器	0x00000006
I2CHSSPKLEN*	0x2C	I2C 高速模式尖峰抑制阈值寄存器	0x00000002
I2CSDASETUP*	0x30	I2C 数据建立时间寄存器	0x00000064
I2CSDAHOLD*	0x34	I2C 数据保持时间寄存器	0x00000001
I2CTXLVL	0x38	I2C 发送 FIFO 当前长度寄存器	0x00000000
I2CRXLVL	0x3C	I2C 接收 FIFO 当前长度寄存器	0x00000000
I2CTXTH*	0x40	I2C 发送 FIFO 长度阈值寄存器	0x00000000
I2CRXTH*	0x44	I2C 接收 FIFO 长度阈值寄存器	0x00000000
I2CSTS	0x54	I2C 状态寄存器	0x00000018
I2CTXABRTSRC	0x58	I2C 发送中止源寄存器	0x00000000
I2CRAWIF	0x5C	I2C 中断原始事件标志寄存器	0x00000000
I2CIF	0x60	I2C 中断标志寄存器	0x00000000
I2CIE*	0x64	I2C 中断使能寄存器	0x000010FF
I2CIC	0x68	I2C 中断清除寄存器	0x00000000
I2CREGKEY	0x6C	I2C 模块写使能寄存器	0x1ACCE551

注意： 由*标记的寄存器仅当 I2CREGKEY=0x1ACCE551 时才可以改写。

©2025 旋智电子科技（上海）有限公司

21.6.2 I2C 寄存器

表 21-4：I2C 控制寄存器（I2CCTL）位段定义

I2CCTL (I2C Control Register) Offset：0x0 Default：0x00000062 Access：I2Cx->I2CCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CLKSTRETCH	6 ACKGC	5 RESTARTEN	4 ADDRMODE	3 SPEED	2	1 MODE	0 EN

表 21-5：I2C 控制寄存器（I2CCTL）位段描述

Bits	位段名	Type	Reset	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	CLKSTRETCH	RW	0x0	当 RX FIFO 满时时钟拉伸 0：禁用 1：启用
6	ACKGC	RW	0x1	对于总线上的总线调用进行回应。此位仅适用于从模式 0：使用 NACK 对总线调用进行回应 1：使用 ACK 对总线调用进行回应
5	RESTARTEN	RW	0x1	确定作为主设备时是否可以发送 RESTART 条件。某些较旧的从设备不支持处理 RESTART 条件；然而，RESTART 条件在几种 I2C 操作中使用。 当禁用 RESTART 时，主设备禁止执行以下功能： - 在传输中改变方向（分割） -发送 START 字节 -高速模式操作 - 7 位寻址模式下的组合格式传输 -使用 10 位地址进行读操作 每次传输发送多个字节。通过替换 RESTART 条件后跟 STOP 和随后的 START 条件，分割操作会被拆分为多个 I2C 传输。如果执行上述操作，将导致设置 I2CRAWIF 寄存器的第 6 位（TXABRT）。 0：禁用 1：启用

©2025 旋智电子科技（上海）有限公司

Bits	位段名	Type	Reset	描述
4	ADDRMODE	RW	0x0	寻址模式 0：7 位寻址 I2C 会忽略涉及 10 位寻址的事务；对于 7 位寻址，仅比较 I2CSLVADDR 寄存器的低 7 位。 1：10 位寻址 I2C 仅对与 I2CSLVADDR 寄存器的完整 10 位匹配的 10 位寻址传输进行回应。
3:2	SPEED	RW	0x0	主模式下的速度 00：标准模式（100 kbit/s） 01：快速模式（400 kbit/s） 10：高速模式（3.4 Mbit/s）
1	MODE	RW	0x1	工作模式 0：主机模式 1：从机模式
0	EN	RW	0x0	I2C 使能（启用或禁用 I2C 时存在两个 I2C 时钟延迟）。 当禁用 I2C 时，会发生以下情况： - TX FIFO 和 RX FIFO 被清空。 - I2CIF 寄存器中的状态位仍然处于活动状态，直到 I2C 进入空闲状态。 -如果模块正在发送，它会在当前传输完成后停止并删除传输 FIFO 的内容。 -如果模块正在接收，I2C 会在当前字节结束时停止当前传输，并且不会确认传输。 0：禁用 I2C（将 FIFO 保持在擦除状态）。软件可以在 I2C 处于活动状态时禁用 I2C。然而，重要的是要确保正确地禁用 I2C。 1：启用 I2C

表 21-6：I2C 主机寻址寄存器（I2CMADDR）位段定义

I2CMADDR (I2C Master Address Register) Offset：0x4 Default：0x00090055 Access：I2Cx->I2CMADDR							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
HSMCODE							
15	14	13	12	11	10	9	8
RESERVED				SPECIALCMD		TARADDR	
7	6	5	4	3	2	1	0
TARADDR							

SPINTROL

表 21-7：I2C 主机寻址寄存器（I2CMADDR）位段描述

Bits	位段名	Type	Reset	描述
31:24	RESERVED_31_24	RO	0x0	保留
23:16	HSMCODE	RW	0x9	8 位高速模式主机代码，格式为（0x00001xxx）。 只有当 I2C 接口被禁用（I2CCTL_EN=0）并且要写入的值处于有效格式（00001xxx）时，才能写入该寄存器。 其他时间的写操作没有效果。
15:12	RESERVED_15_12	RO	0x0	保留
11:10	SPECIALCMD	RW	0x0	特殊命令 00：禁用特殊命令 10：广播命令 发出广播命令后，只能进行写操作。尝试发出读命令会导致设置 I2CRAWIF 寄存器的第 6 位（TXABRT）。I2C 会保持在广播模式，直到 SPECIAL 位被清除。 11：起始字节
9:0	TARADDR	RW	0x55	用于任何主机事务的目标地址。 对于发出广播命令，这些位将被忽略。 要生成起始字节，CPU 只需将这些位写入一次。 它不应与 I2CSLVADDR 相同。

表 21-8：I2C 从机地址寄存器（I2CSADDR）位段定义

I2CSADDR (I2C Slave Address Register) Offset：0x8 Default：0x00000055 Access：I2Cx->I2CSADDR							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						VAL	
7	6	5	4	3	2	1	0
VAL							

©2025 旋智电子科技（上海）有限公司

表 21-9：I2C 从机地址寄存器（I2CSADDR）位段描述

Bits	位段名	Type	Reset	描述
31:10	RESERVED_31_10	RO	0x0	保留
9:0	VAL	RW	0x55	当 I2C 作为从设备时的从设备地址。 只有当 I2CCTL_EN=0 时，才能写入该寄存器。其他时间的写操作没有效果。 注意：默认值不能是保留的地址位置，即 0x00 到 0x07 或 0x78 到 0x7f。如果将 I2CSLVADDR 或 I2CMASTERADDR 编程为保留值，设备的正确操作不能得到保证。

表 21-10：I2C 数据和命令寄存器（I2CDATACMD）位段定义

I2CDATACMD (I2C Data and Command Register)　Offset：0xC　Default：0x00000000 Access：I2Cx->I2CDATACMD							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							CMD
7	6	5	4	3	2	1	0
DATA							

表 21-11：I2C Data and Command Register (I2CDATACMD) 描述

Bits	位段名	Type	Reset	描述
31:9	RESERVED_31_9	RO	0x0	保留
8	CMD	RW	0x0	读写控制 0：写 1：读
7:0	DATA	RW	0x0	I2C 总线上发送或者接受的数据 写本寄存器往发送 FIFO 放入数据； 读本寄存器从接收 FIFO 取出数据。

SPINTROL

表 21-12：I2C 标准速度模式时钟低电平计数寄存器（I2CSSLCNT）位段定义

I2CSSLCNT (Standard Speed I2C Clock SCL Low Count Register) Offset：0x10 Default：0x0000024C Access：I2Cx->I2CSSLCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 21-13：I2C 标准速度模式时钟低电平计数寄存器（I2CSSLCNT）位段描述

Bits	位段名	Type	Reset	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x24C	标准速度模式下 SCL 低电平的持续时间。 即 SCL 低电平持续的 I2CSSLCNT 时钟周期。 最小有效值为 4。写入小于 4 的值将被硬件重置为 4。 只有当 I2CCTL_EN=0 时，才能写入该寄存器。 其他时间的写操作没有效果。

表 21-14：I2C 标准速度模式时钟高电平计数寄存器（I2CSSHCNT）位段定义

I2CSSHCNT (Standard Speed I2C Clock SCL High Count Register) Offset：0x14 Default：0x000001F4 Access：I2Cx->I2CSSHCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

©2025 旋智电子科技（上海）有限公司

表 21-15：I2C 标准速度模式时钟高电平计数寄存器（I2CSSHCNT）位段

Bits	位段名	Type	Reset	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x1F4	标准速度模式下 SCL 高电平的持续时间。 即 SCL 高电平持续的 I2CSSHCNT 时钟周期。 最小有效值为 5。写入小于 5 的值将被硬件重置为 5。 只有当 I2CCTL_EN=0 时，才能写入该寄存器。其他时间的写操作没有效果。

表 21-16：I2C 快速度模式时钟低电平计数寄存器（I2CFSLCNT）位段定义

I2CFSLCNT (Fast Speed I2C Clock SCL Low Count Register)　Offset：0x18　Default：0x000000A3
Access：I2Cx->I2CFSLCNT

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 21-17：I2C 快速度模式时钟低电平计数寄存器（I2CFSLCNT）位段描述

Bits	位段名	Type	Reset	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0xA3	快速速度模式下 SCL 低电平的持续时间。 即 SCL 低电平持续的 I2CFSLCNT 时钟周期。 最小有效值为 4。写入小于 4 的值将被硬件重置为 4。 只有当 I2CCTL_EN=0 时，才能写入该寄存器。其他时间的写操作没有效果。

©2025 旋智电子科技（上海）有限公司

表 21-18：I2C 快速度模式时钟高电平计数寄存器（I2CFSHCNT）位段定义

I2CFSHCNT (Fast Speed I2C Clock SCL High Count Register) Offset：0x1C Default：0x0000004B Access：I2Cx->I2CFSHCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 21-19：I2C 快速度模式时钟高电平计数寄存器（I2CFSHCNT）位段描述

Bits	位段名	Type	Reset	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x4B	快速速度模式下 SCL 高电平的持续时间。 即 SCL 高电平持续的 I2CFSHCNT 时钟周期。 最小有效值为 5。写入小于 5 的值将被硬件重置为 5。 只有当 I2CCTL_EN=0 时，才能写入该寄存器。 其他时间的写操作没有效果。

表 21-20：I2C 高速度模式时钟低电平计数寄存器（I2CHSLCNT）位段定义

I2CHSLCNT (High Speed I2C Clock SCL Low Count Register) Offset：0x20 Default：0x00000014 Access：I2Cx->I2CHSLCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

©2025 旋智电子科技（上海）有限公司

表 21-21：I2C 高速度模式时钟低电平计数寄存器（I2CHSLCNT）位段描述

Bits	位段名	Type	Reset	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x14	高速模式下 SCL 低电平的持续时间。 即 SCL 低电平持续的 I2CHSLCNT 时钟周期。 最小有效值为 4。写入小于 4 的值将被硬件重置为 4。 只有当 I2CCTL_EN=0 时，才能写入该寄存器。 其他时间的写操作没有效果。

表 21-22：I2C 高速度模式时钟高电平计数寄存器（I2CHSHCNT）位段定义

I2CHSHCNT (High Speed I2C Clock SCL High Count Register)　Offset：0x24　Default：0x00000008
Access：I2Cx->I2CHSHCNT

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 21-23：I2C 高速度模式时钟高电平计数寄存器（I2CHSHCNT）位段描述

Bits	位段名	Type	Reset	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x8	高速模式下 SCL 高电平的持续时间。 即 SCL 高电平持续的 I2CHSHCNT 时钟周期。 最小有效值为 1。写入 0 将被硬件重置为 1。 只有当 I2CCTL_EN=0 时，才能写入该寄存器。 其他时间的写操作没有效果。

SPINTROL

表 21-24：I2C 快速模式尖峰抑制阈值寄存器（I2CFSSPKLEN）位段定义

I2CFSSPKLEN (Fast Speed I2C Spike Suppresion Limit Register) Offset：0x28 Default：0x00000006 Access：I2Cx->I2CFSSPKLEN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
VAL							

表 21-25：I2C 快速模式尖峰抑制阈值寄存器（I2CFSSPKLEN）位段描述

Bits	位段名	Type	Reset	描述
31:8	RESERVED_31_8	RO	0x0	保留
7:0	VAL	RW	0x6	在快速模式下，由抑制尖峰逻辑过滤掉的 SCL 或 SDA 线上最长尖峰的持续时间。 最小值为 1，写入 0 将被硬件重置为 1。 只有当 I2CCTL_EN=0 时，才能写入该寄存器。其他时间的写操作没有效果。

表 21-26：I2C 高速模式尖峰抑制阈值寄存器（I2CHSSPKLEN）位段定义

I2CHSSPKLEN (High Speed I2C Spike Suppresion Limit Register) Offset：0x2C Default：0x00000002 Access：I2Cx->I2CHSSPKLEN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
VAL							

©2025 旋智电子科技（上海）有限公司

表 21-27：I2C 高速模式尖峰抑制阈值寄存器（I2CHSSPKLEN）位段描述

Bits	位段名	Type	Reset	描述
31:8	RESERVED_31_8	RO	0x0	保留
7:0	VAL	RW	0x2	在高速模式下，由抑制尖峰逻辑过滤掉的 SCL 或 SDA 线上最长尖峰的持续时间。 最小值为 1，写入 0 将被硬件重置为 1。 只有当 I2CCTL_EN=0 时，才能写入该寄存器。其他时间的写操作没有效果。

表 21-28：I2C 数据建立时间寄存器（I2CSDASETUP）位段定义

I2CSDASETUP (I2C SDA Setup Time Register)　Offset：0x30　Default：0x00000064
Access：I2Cx->I2CSDASETUP

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
VAL							

表 21-29：I2C 数据建立时间寄存器（I2CSDASETUP）位段描述

Bits	位段名	Type	Reset	描述
31:8	RESERVED_31_8	RO	0x0	保留
7:0	VAL	RW	0x64	将 SDA 设置时间配置为 VAL CLK_I2C 周期 即，如果所需的设置延迟为 1000 纳秒，则在 CLK_I2C 频率为 10 MHz 的情况下，建议将该寄存器设置为 11。 最小值为 2，写入小于 2 的值将被硬件重置为 2。 只有当 I2CCTL_EN=0 时，才能写入该寄存器。其他时间的写操作没有效果。

SPINTROL

表 21-30：I2C 数据保持时间寄存器（I2CSDAHOLD）位段定义

I2CSDAHOLD (I2C SDA Hold Time Register) Offset：0x34 Default：0x00000001 Access：I2Cx->I2CSDAHOLD							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 21-31：I2C 数据保持时间寄存器（I2CSDAHOLD）位段描述

Bits	位段名	Type	Reset	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	VAL	RW	0x1	将 SDA 保持时间配置为 VAL CLK_I2C 周期 即，如果所需的保持延迟为 1000 纳秒，则在 CLK_I2C 频率为 10 MHz 的情况下，建议将该寄存器设置为 11。 只有当 I2CCTL_EN=0 时，才能写入该寄存器。其他时间的写操作没有效果。

表 21-32：I2C 发送 FIFO 当前长度寄存器（I2CTXLVL）位段定义

I2CTXLVL (I2C Transmit FIFO Level Register) Offset：0x38 Default：0x00000000 Access：I2Cx->I2CTXLVL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED			VAL				

©2025 旋智电子科技（上海）有限公司

表 21-33：I2C 发送 FIFO 当前长度寄存器（I2CTXLVL）位段描述

Bits	位段名	Type	Reset	描述
31:5	RESERVED_31_5	RO	0x0	保留
4:0	VAL	RO	0x0	传输 FIFO 中有效数据条目的数量。

表 21-34：I2C 接收 FIFO 当前长度寄存器（I2CRXLVL）位段定义

I2CRXLVL (I2C Receive FIFO Level Register)　Offset：0x3C　Default：0x00000000
Access：I2Cx->I2CRXLVL

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED			VAL				

表 21-35：I2C 接收 FIFO 当前长度寄存器（I2CRXLVL）位段描述

Bits	位段名	Type	Reset	描述
31:5	RESERVED_31_5	RO	0x0	保留
4:0	VAL	RO	0x0	接收 FIFO 中有效数据条目的数量。

表 21-36：I2C 发送 FIFO 长度阈值寄存器（I2CTXTH）位段定义

I2CTXTH (I2C Transmit FIFO Threshold Register)　Offset：0x40　Default：0x00000000
Access：I2Cx->I2CTXTH

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				VAL			

©2025 旋智电子科技（上海）有限公司

表 21-37：I2C 发送 FIFO 长度阈值寄存器（I2CTXTH）位段描述

Bits	位段名	Type	Reset	描述
31:4	RESERVED_31_4	RO	0x0	保留
3:0	VAL	RW	0x0	触发 TFEMPTY 中断的传输 FIFO 阈值级别。

表 21-38：I2C 接收 FIFO 长度阈值寄存器（I2CRXTH）位段定义

I2CRXTH (I2C Receive FIFO Threshold Register)　Offset：0x44　Default：0x00000000
Access：I2Cx->I2CRXTH

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				VAL			

表 21-39：I2C 接收 FIFO 长度阈值寄存器（I2CRXTH）位段描述

Bits	位段名	Type	Reset	描述
31:4	RESERVED_31_4	RO	0x0	保留
3:0	VAL	RW	0x0	触发 RFFULL 中断的接收 FIFO 阈值级别。

表 21-40：I2C 状态寄存器（I2CSTS）位段定义

I2CSTS (I2C Status Register)　Offset：0x54　Default：0x00000018
Access：I2Cx->I2CSTS

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						SRXDLOST	SRXABRT
7	6	5	4	3	2	1	0
EN	RXFULL	RXNE	TXNF	TXEMPTY	SACT	MACT	ACT

©2025 旋智电子科技（上海）有限公司

表 21-41：I2C 状态寄存器（I2CSTS）位段描述

Bits	位段名	Type	Reset	描述
31:10	RESERVED_31_10	RO	0x0	保留
9	SRXDLOST	RO	0x0	禁用 I2C 时，从设备接收的数据丢失情况： 0：在将 I2CENABLE 从 1 更改为 0 之前，从设备的接收操作已完成。 1：在将 I2CENABLE 从 1 更改为 0 时，从设备的接收操作被中止，并且至少接收到一个来自 I2C 传输的数据字节。
8	SRXABRT	RO	0x0	在从设备忙时禁用从设备： 0：在从设备操作期间保持 I2CENABLE 不变。 1：在从设备接收来自远程主设备的数据期间，将 I2CENABLE 从 1 更改为 0。
7	EN	RO	0x0	I2C 的启用状态： 0：I2C 被认为完全处于非活动状态。 可以安全地读取 SLVRDLOST 和 SLVDISONBUSY 位。 1：I2C 被认为处于启用状态。
6	RXFULL	RO	0x0	接收 FIFO 完全满： 0：接收 FIFO 为空。 1：接收 FIFO 已完全填满。
5	RXNE	RO	0x0	接收 FIFO 非空（因此可以从中读取数据）： 0：接收 FIFO 为空。 1：接收 FIFO 包含一个或多个数据项。
4	TXNF	RO	0x1	传输 FIFO 非满（因此可以向其中写入数据）： 0：传输 FIFO 已满。 1：传输 FIFO 包含空位置。
3	TXEMPTY	RO	0x1	传输 FIFO 完全空： 0：传输 FIFO 包含一个或多个数据项。 1：传输 FIFO 为空。
2	SACT	RO	0x0	从设备 FSM（有限状态机）活动状态： 0：从设备 FSM 处于空闲状态（非活动状态）。 1：从设备 FSM 不处于空闲状态（活动状态）。
1	MACT	RO	0x0	主设备 FSM 活动状态： 0：主设备 FSM 处于空闲状态（非活动状态）。 1：主设备 FSM 不处于空闲状态（活动状态）。

SPINTROL

Bits	位段名	Type	Reset	描述
0	ACT	RO	0x0	I2C 活动状态（SACT 和 MACT 的逻辑或）： 0：主设备和从设备 FSM 都处于空闲状态。 1：主设备或从设备 FSM 处于活动状态。

表 21-42：I2C 发送中止源寄存器（I2CTXABRTSRC）位段定义

I2CTXABRTSRC (I2C Transmit Abort Source Register) Offset：0x58 Default：0x00000000 Access：I2Cx->I2CTXABRTSRC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 SCMDERR	14 SARBLOST	13 STFIFOCLR	12 MARBLOST	11 MDISABLED	10 RD10BNORESTART	9 STARTNORESTART	8 HSNORESTART
7 STARTACK	6 HSMCACK	5 GCREAD	4 GCNACK	3 TXDNACK	2 ADDR10B2NACK	1 ADDR10B1NACK	0 ADDR7BNACK

表 21-43：I2C 发送中止源寄存器（I2CTXABRTSRC）位段描述

Bits	位段名	Type	Reset	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	SCMDERR	RO	0x0	I2C 的角色：从设备发送器 当处理器端响应从设备模式的数据传输请求，并且用户在 I2CDATACMD 寄存器的 CMD 位（第 8 位）中写入 1 时，设置此位。 0：事件未发生 1：事件发生
14	SARBLOST	RO	0x0	I2C 的角色：从设备发送器 当从设备在向远程主设备传输数据时丢失总线时，设置此位。同时设置 I2CTXABRTSRC.MARBLOST 位。 注意：尽管从设备从不“拥有”总线，但总线上可能出现问题。这是一种安全检查。例如，在 SCL 的低到高转换期间，如果数据总线上的内容与应该传输的内容不符，则 I2C 将不再拥有总线。 0：事件未发生 1：事件发生
13	STFIFOCLR	RO	0x0	I2C 的角色：从设备发送器

©2025 旋智电子科技（上海）有限公司

Bits	位段名	Type	Reset	描述
				当从设备从外部主设备接收到 NACK，并且 TX FIFO 不为空时，从设备发出 TXABRT 中断以清除 TX FIFO 中的旧数据时，设置此位。 0：事件未发生 1：事件发生
12	MARBLOST	RO	0x0	I2C 的角色：主/从设备发送器 当主设备丧失仲裁，或者如果同时设置了 I2CTXABRTSRC.SARBLOST，则从设备发送器丧失了仲裁时，设置此位。 0：事件未发生 1：事件发生
11	MDISABLED	RO	0x0	I2C 的角色：从设备发送器/接收器 当在没有主设备的请求下向 TX FIFO 写入数据时设置此位。即设备本身是从设备而不是主设备。并且它还没有从外部主设备接收到读取请求。 0：事件未发生 1：事件发生
10	RD10BNORESTART	RO	0x0	I2C 的角色：主设备接收器 当禁用重启（I2CCTL.RESTARTEN=0）并且主设备以 10 位寻址模式发送读命令时设置此位。 0：事件未发生 1：事件发生
9	STARTNORESTART	RO	0x0	I2C 的角色：主设备发送器/接收器 当禁用重启（I2CCTL.RESTARTEN=0）并且用户尝试发送 START 字节时设置此位。 遵循以下配置以避免此错误： (1) I2CCTL.RESTARTEN=1 (2) I2CMASTERADDR.SPECIAL=0 或 I2CMASTERADDR.GCORSTART=0 否则，此位在清除 I2CIC_TXABRT 时将在下一个时钟周期再次设置为 1。 0：事件未发生 1：事件发生
8	HSNORESTART	RO	0x0	I2C 的角色：主设备发送器/接收器 当禁用重启（I2CCTL.RESTARTEN=0）并且用户尝试在高速模式下传输数据时设置此位。 0：事件未发生 1：事件发生
7	STARTACK	RO	0x0	I2C 的角色：主设备发送器/接收器 当主设备发送了 START 字节并且 START 字节被确认（错误行为）时设置此位。 0：事件未发生

SPINTROL

Bits	位段名	Type	Reset	描述
				1：事件发生
6	HSMCACK	RO	0x0	I2C 的角色：主设备发送器/接收器 当主设备处于高速模式并且高速主设备代码被确认（错误行为）时设置此位。 0：事件未发生 1：事件发生
5	GCREAD	RO	0x0	I2C 的角色：主设备发送器 当主设备发送了广播通信（General Call），但用户将广播通信后的字节编程为从总线上读取（I2CDATACMD_CMD=1）时设置此位。 0：事件未发生 1：事件发生
4	GCNACK	RO	0x0	I2C 的角色：主设备发送器 当主设备发送了广播通信（General Call），但总线上没有任何从设备确认广播通信时设置此位。 0：事件未发生 1：事件发生
3	TXDNACK	RO	0x0	I2C 的角色：主设备发送器 当主设备已经收到地址的确认，但在发送地址后的数据字节时，没有从远程从设备接收到确认时设置此位。 0：事件未发生 1：事件发生
2	ADDR10B2NACK	RO	0x0	I2C 的角色：主设备发送器/接收器 当主设备处于 10 位地址模式，并且 10 位地址的第二个地址字节未被任何从设备确认时设置此位。 0：事件未发生 1：事件发生
1	ADDR10B1NACK	RO	0x0	I2C 的角色：主设备发送器/接收器 当主设备处于 10 位地址模式，并且第一个 10 位地址字节未被任何从设备确认时设置此位。 0：事件未发生 1：事件发生
0	ADDR7BNACK	RO	0x0	I2C 的角色：主设备发送器/接收器 当主设备处于 7 位寻址模式并且发送的地址未被任何从设备确认时设置此位。 0：事件未发生 1：事件发生

©2025 旋智电子科技（上海）有限公司

表 21-44：I2C 中断原始事件标志寄存器（I2CRAWIF）位段定义

I2CRAWIF (I2C Interrupt Raw Flag Register) Offset：0x5C Default：0x00000000 Access：I2Cx->I2CRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED			GENCALL	STARTDET	STOPDET	ACKDET	ACT
7	6	5	4	3	2	1	0
RXDONE	TXABRT	RDREQ	TXDREQ	TXOVF	RXDAV	RXOVF	RXUDF

表 21-45：I2C 中断原始事件标志寄存器（I2CRAWIF）位段描述

Bits	位段名	Type	Reset	描述
31:13	RESERVED_31_13	RO	0x0	保留
12	GENCALL	RO	0x0	General Call 地址接收和确认标志 0：未发生 1：发生
11	STARTDET	RO	0x0	START 或 RESTART 条件标志 0：未检测到条件 1：检测到条件
10	STOPDET	RO	0x0	STOP 条件标志 0：未检测到条件 1：检测到条件
9	ACKDET	RO	0x0	外部接收器检测到的 ACK 锁存标志 0：未检测到 ACK 1：检测到 ACK
8	ACT	RO	0x0	锁存的 I2C 活动状态标志 0：在 I2C 接口上未检测到活动 1：在 I2C 接口上检测到活动
7	RXDONE	RO	0x0	I2C 作为从设备发送器时指示外部主设备接收完成的标志 0：从设备传输被其他主设备确认，继续接收数据。 1：从设备传输未被其他主设备确认，完成数据接收。
6	TXABRT	RO	0x0	锁存的传输中止标志 0：未发生传输中止 1：发生传输中止，并且 TX FIFO 保持清空状态，直到向 I2CIC_TXABRT 写入 1

SPINTROL

Bits	位段名	Type	Reset	描述
5	RDREQ	RO	0x0	I2C 作为从设备时锁存的外部主设备读取请求标志 0：未收到来自另一个主设备的读取请求 1：收到来自另一个主设备的读取请求。 处理器必须响应此中断，然后将请求的数据写入 I2CDATACMD 寄存器。
4	TXDREQ	RO	0x0	指示传输 FIFO 中没有足够数据并请求更多数据的标志 0：I2CTXLVL > I2CTXTH 1：I2CTXLVL <= I2CTXTH 当 I2CTXLVL > I2CTXTH 时，此位自动清除。
3	TXOVF	RO	0x0	指示在传输 FIFO 已满时尝试写入 I2CDATACMD 寄存器的标志 0：传输 FIFO 不溢出 1：传输 FIFO 溢出
2	RXDAV	RO	0x0	指示接收 FIFO 中有足够数据可供读取的标志 0：I2CRXLVL <= I2CRXTH 1：I2CRXLVL > I2CRXTH 当 I2CRXLVL <= I2CRXTH 时，此位自动清除。
1	RXOVF	RO	0x0	指示接收 FIFO 完全填满到 16，并从外部 I2C 设备接收到额外字节。I2C 确认此情况，但在 FIFO 已满后接收的任何数据字节将丢失。 0：接收 FIFO 未溢出 1：接收 FIFO 溢出
0	RXUDF	RO	0x0	指示处理器尝试通过 I2CDATACMD 寄存器读取接收 FIFO 时，接收 FIFO 为空的标志 0：接收 FIFO 未下溢 1：接收 FIFO 下溢

表 21-46：I2C 中断标志寄存器（I2CIF）位段定义

I2CIF (I2C Interrupt Flag Register) Offset：0x60 Default：0x00000000 Access：I2Cx->I2CIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED		INT	GENCALL	STARTDET	STOPDET	ACKDET	ACT
7	6	5	4	3	2	1	0
RXDONE	TXABRT	RDREQ	TXDREQ	TXOVF	RXDAV	RXOVF	RXUDF

©2025 旋智电子科技（上海）有限公司

表 21-47：I2C 中断标志寄存器（I2CIF）位段描述

Bits	位段名	Type	Reset	描述
31:14	RESERVED_31_14	RO	0x0	保留
13	INT	RO	0x0	全局中断标志： 0：未发生中断 1：发生中断
12	GENCALL	RO	0x0	收到和确认的广播地址标志： 0：未收到或确认广播地址 1：收到并确认广播地址
11	STARTDET	RO	0x0	START 或 RESTART 条件标志： 0：未检测到 START 或 RESTART 条件 1：检测到 START 或 RESTART 条件
10	STOPDET	RO	0x0	STOP 条件标志： 0：未检测到 STOP 条件 1：检测到 STOP 条件
9	ACKDET	RO	0x0	外部接收器检测到的 ACK 标志： 0：未检测到 ACK 1：检测到 ACK
8	ACT	RO	0x0	锁定的 I2C 活动状态标志： 0：未检测到 I2C 接口上的活动 1：检测到 I2C 接口上的活动
7	RXDONE	RO	0x0	当 I2C 作为从机发送器时，外部主机接收完成标志： 0：从机传输被其他主机确认，继续接收数据 1：从机传输未被其他主机确认，结束数据接收
6	TXABRT	RO	0x0	锁定的传输中止标志： 0：未发生传输中止 1：传输中止，TX FIFO 保持刷新状态，直到向 I2CIC_TXABRT 写入 1
5	RDREQ	RO	0x0	当 I2C 作为从机时，锁定的外部主机读取请求标志： 0：未收到其他主机的读取请求 1：收到其他主机的读取请求。 处理器必须响应此中断，然后将请求的数据写入 I2CDATACMD 寄存器。
4	TXDREQ	RO	0x0	标志，指示传输 FIFO 中没有足够的数据，并请求更多数据： 0：I2CTXLVL > I2CTXTH 1：I2CTXLVL <= I2CTXTH 当 I2CTXLVL > I2CTXTH 时，此位会自动清除。
3	TXOVF	RO	0x0	标志，指示尝试在传输 FIFO 已满时写入 I2CDATACMD 寄存器：

©2025 旋智电子科技（上海）有限公司

Bits	位段名	Type	Reset	描述
				0：传输 FIFO 未溢出 1：传输 FIFO 溢出
2	RXDAV	RO	0x0	标志，指示接收 FIFO 中有足够的数据可供读取： 0：I2CRXLVL <= I2CRXTH 1：I2CRXLVL > I2CRXTH 当 I2CRXLVL <= I2CRXTH 时，此位会自动清除。
1	RXOVF	RO	0x0	标志，指示接收 FIFO 已完全填满为 16 个字节，并从外部 I2C 设备接收到一个附加字节。I2C 会确认这一点，但 FIFO 已满后接收的任何数据字节都将丢失。 0：接收 FIFO 未溢出 1：接收 FIFO 溢出
0	RXUDF	RO	0x0	标志，指示处理器在接收 FIFO 为空时尝试通过 I2CDATACMD 寄存器读取数据： 0：接收 FIFO 未下溢 1：接收 FIFO 下溢

表 21-48：I2C 中断使能寄存器（I2CIE）位段定义

I2CIE (I2C Interrupt Enable Register) Offset：0x64 Default：0x000010FF Access：I2Cx->I2CIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED			GENCALL	STARTDET	STOPDET	ACKDET	ACT
7	6	5	4	3	2	1	0
RXDONE	TXABRT	RDREQ	TXDREQ	TXOVF	RXDAV	RXOVF	RXUDF

表 21-49：I2C 中断使能寄存器（I2CIE）位段描述

Bits	位段名	Type	Reset	描述
31:13	RESERVED_31_13	RO	0x0	保留
12	GENCALL	RW	0x1	启用/禁用 GENCALL 中断 0：禁用 GENCALL 中断 1：启用 GENCALL 中断
11	STARTDET	RW	0x0	启用/禁用 STARTDET 中断 0：禁用 STARTDET 中断 1：启用 STARTDET 中断

©2025 旋智电子科技（上海）有限公司

Bits	位段名	Type	Reset	描述
10	STOPDET	RW	0x0	启用/禁用 STOPDET 中断 0：禁用 STOPDET 中断 1：启用 STOPDET 中断
9	ACKDET	RW	0x0	启用/禁用 ACKDET 中断 0：禁用 ACKDET 中断 1：启用 ACKDET 中断
8	ACT	RW	0x0	启用/禁用 ACTIVITY 中断 0：禁用 ACTIVITY 中断 1：启用 ACTIVITY 中断
7	RXDONE	RW	0x1	启用/禁用 RXDONE 中断 0：禁用 RXDONE 中断 1：启用 RXDONE 中断
6	TXABRT	RW	0x1	启用/禁用 TXABRT 中断 0：禁用 TXABRT 中断 1：启用 TXABRT 中断
5	RDREQ	RW	0x1	启用/禁用 RDREQ 中断 0：禁用 RDREQ 中断 1：启用 RDREQ 中断
4	TXDREQ	RW	0x1	启用/禁用 TXDREQ 中断 0：禁用 TXDREQ 中断 1：启用 TXDREQ 中断
3	TXOVF	RW	0x1	启用/禁用 TXOVF 中断 0：禁用 TXOVF 中断 1：启用 TXOVF 中断
2	RXDAV	RW	0x1	启用/禁用 RXDAV 中断 0：禁用 RXDAV 中断 1：启用 RXDAV 中断
1	RXOVF	RW	0x1	启用/禁用 RXOVF 中断 0：禁用 RXOVF 中断 1：启用 RXOVF 中断
0	RXUDF	RW	0x1	启用/禁用 RXUDF 中断 0：禁用 RXUDF 中断 1：启用 RXUDF 中断

表 21-50：I2C 中断清除寄存器（I2CIC）位段定义

I2CIC (I2C Interrupt Enable Register)　Offset：0x68　Default：0x00000000 Access：I2Cx->I2CIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8

©2025 旋智电子科技（上海）有限公司

RESERVED		INT	GENCALL	STARTDET	STOPDET	ACKDET	ACT
7 RXDONE	6 TXABRT	5 RDREQ	4 TXDREQ	3 TXOVF	2 RXDAV	1 RXOVF	0 RXUDF

表 21-51：I2C 中断清除寄存器（I2CIC）位段描述

Bits	位段名	Type	Reset	描述
31:14	RESERVED_31_14	RO	0x0	保留
13	INT	W1C	0x0	清除全局组合中断 0：写入 0 没有任何效果，并且始终返回 0 1：写入 1 清除中断标志。 此位自动清除为 0
12	GENCALL	W1C	0x0	清除 GENCALL 中断 0：写入 0 没有任何效果，并且始终返回 0 1：写入 1 清除中断标志。 此位自动清除为 0
11	STARTDET	W1C	0x0	清除 STARTDET 中断 0：写入 0 没有任何效果，并且始终返回 0 1：写入 1 清除中断标志。 此位自动清除为 0
10	STOPDET	W1C	0x0	清除 STOPDET 中断 0：写入 0 没有任何效果，并且始终返回 0 1：写入 1 清除中断标志。 此位自动清除为 0
9	ACKDET	W1C	0x0	清除 ACKDET 中断 0：写入 0 没有任何效果，并且始终返回 0 1：写入 1 清除中断标志。 此位自动清除为 0
8	ACT	W1C	0x0	清除 ACTIVITY 中断 0：写入 0 没有任何效果，并且始终返回 0 1：写入 1 清除中断标志。 此位自动清除为 0
7	RXDONE	W1C	0x0	清除 RXDONE 中断 0：写入 0 没有任何效果，并且始终返回 0 1：写入 1 清除中断标志。 此位自动清除为 0
6	TXABRT	W1C	0x0	清除 TXABRT 中断 0：写入 0 没有任何效果，并且始终返回 0 1：写入 1 清除中断标志。 此位自动清除为 0

©2025 旋智电子科技（上海）有限公司

Bits	位段名	Type	Reset	描述
5	RDREQ	W1C	0x0	清除 RDREQ 中断 0：写入 0 没有任何效果，并且始终返回 0 1：写入 1 清除中断标志。 此位自动清除为 0
4	TXDREQ	W1C	0x0	清除 TXDREQ 中断 0：写入 0 没有任何效果，并且始终返回 0 1：写入 1 清除中断标志。 此位自动清除为 0
3	TXOVF	W1C	0x0	清除 TXOVF 中断 0：写入 0 没有任何效果，并且始终返回 0 1：写入 1 清除中断标志。 此位自动清除为 0
2	RXDAV	W1C	0x0	清除 RXDAV 中断 0：写入 0 没有任何效果，并且始终返回 0 1：写入 1 清除中断标志。 此位自动清除为 0
1	RXOVF	W1C	0x0	清除 RXOVF 中断 0：写入 0 没有任何效果，并且始终返回 0 1：写入 1 清除中断标志。 此位自动清除为 0
0	RXUDF	W1C	0x0	清除 RXUDF 中断 0：写入 0 没有任何效果，并且始终返回 0 1：写入 1 清除中断标志。 此位自动清除为 0

表 21-52：I2C 模块写使能寄存器（I2CREGKEY）位段定义

I2CREGKEY (I2C Register Write-Allow Key Register)　Offset：0x6C　Default：0x1ACCE551 Access：I2Cx->I2CREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

表 21-53：I2C 模块写使能寄存器（I2CREGKEY）位段描述

SPINTROL

Bits	位段名	Type	Reset	描述
31:0	KEY	RW	0x1ACCE551	将 0x1ACCE551 写入以解锁受保护的 I2C 寄存器。 此寄存器的 28 个最高有效位（MSBs）不能为 0x5555555。

SPINTROL

©2025 旋智电子科技（上海）有限公司

22 控制器局域网络（CAN）

22.1 概述

SPC1169 提供 1 个 CAN 模块，根据 CAN 协议通信，同时需要外部 CAN 收发器芯片将其连接到物理层（CAN 总线）。CAN 模块特性如下：

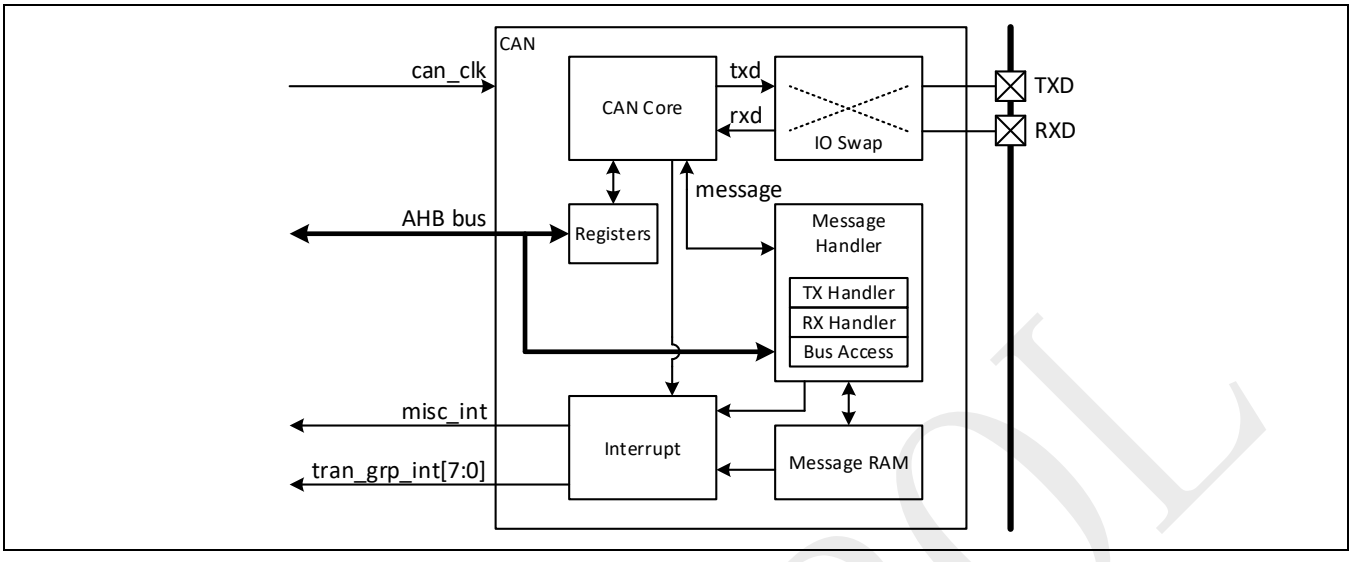
- 兼容 ISO-11898-1:2015 标准，相应的帧格式称之为经典 CAN 和 ISO-FDCAN；
- 兼容 BOSCH CAN2.0B 标准；
  - 其帧格式与 ISO-11898-1:2015 标准经典 CAN 兼容；
- 兼容 BOSCH FDCAN V1.0 标准，相应的帧格式称之为 NONISO-FDCAN；
  - 其帧格式与 ISO-11898-1:2015 标准 ISO-FDCAN 不兼容；
- 支持 CAN 总线关闭（Bus Off）恢复；
- 支持发送失败时的自动重传；
- 支持限制（Restricted）模式；
- 支持监视模式；
- 支持测试模式；
- 支持 64 个信箱（Mailbox），每个有独立的 ID 过滤规则；
- 支持远程帧的自动响应；
- 支持 CAN 总线错误的记录；
- 支持 32 位时间戳。

注意： SPC1169 的 CAN 模块不支持唤醒功能。用户需要选用支持唤醒功能的 CAN 收发器，并将其唤醒信号连到 SPC1169 的 GPIO 引脚，来实现 CAN 唤醒功能。

经典 CAN 和 Bosch CAN2.0B 规范的帧格式是兼容的，而 ISO-FDCAN 与 NONISO-FDCAN 由于帧格式和 CRC 计算方法的差异而不兼容。

SPINTROL

图 22-1：CAN 框图



如图 22-1 所示，CAN 模块主要包括：寄存器控制接口、引脚选择逻辑、核心控制、消息处理、信箱报文存储和中断控制。

核心控制子模块控制 CAN 帧的传输和 CAN 总线状态的检测。

CAN 网络上通信的报文对象可以单独配置。信箱用来存储报文对象，包括其报文标识符掩码。它支持 32 位写访问和 8 位、16 位、32 位读访问，并提供 ECC 以确保数据完整性。

报文处理程序控制信箱访问并自动仲裁通过发送处理程序、接收处理程序、核心控制和 AHB 进行访问的优先级。

使能 CAN 模块后，发送处理器从第 1 个信箱开始逐个检查所有信箱。如果有信箱请求报文传输，发送处理器通知核心控制模块读取信箱内的报文内容并传输。当前报文传输成功或用户取消后，将开始检查下一个信箱。

当接收到报文标识符时，接收处理器总是从第一个信箱开始搜索。一旦找到一个有效的信箱能够存储数据，就会通知核心控制模块将所有接收到的报文数据保存到这个信箱中。

AHB 可以随时写入任何信箱，除非要写入的信箱当前正用于消息传输。在这种情况下，AHB 对该信箱的写访问被阻止。若要配置信箱，用户应首先禁用信箱并回读状态以检查是否成功禁用。如果仍然启用，用户不应该配置该信箱，而是等到当前消息传输完成。

引脚选择逻辑支持 TXD 和 RXD 之间的引脚功能交换。

每 8 个连续的信箱共享一个报文传输完成中断。因此，CAN 模块有 8 个报文传输完成中断。诸如 RAM ECC 错误或 CAN 总线错误等事件，共享一个杂项中断。

注意： 允许通过 DMA 访问信箱报文内存。

©2025 旋智电子科技（上海）有限公司

22.2 CAN 端口信号

表 22-1 罗列并描述了外部端口信号 TXD 和 RXD 如何连接到 CAN 模块。它们都是数字 CMOS 电平信号，采用通道复用最终连到 GPIO 引脚。参考章节 4.2 配置引脚通道功能。

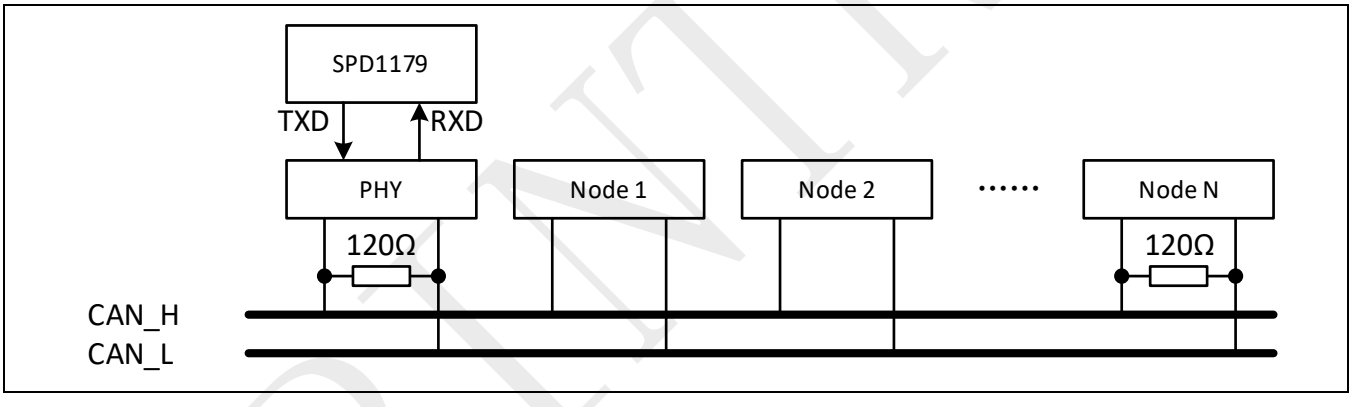
表 22-1：CAN 端口信号

信号名	类型	描述
RXD	输入	**串行输入** 送入接收移位寄存器的串行数据，连接到 CAN 收发器的 RXD 引脚。
TXD	输出	**串行输出** 发送的串行数据，连接到 CAN 收发器的 TXD 引脚，复位时该信号为 1。

22.2.1 CAN 设备连接

SPC1169 上的 CAN 模块应当按照如图 22-2 所示的方式通过 CAN 收发器连接到网络上的其他 CAN 设备。其中 CAN_H 和 CAN_L 是一对双绞线。

图 22-2：CAN 设备连接



22.3 CAN 帧格式

22.3.1 ISO 经典 CAN

如

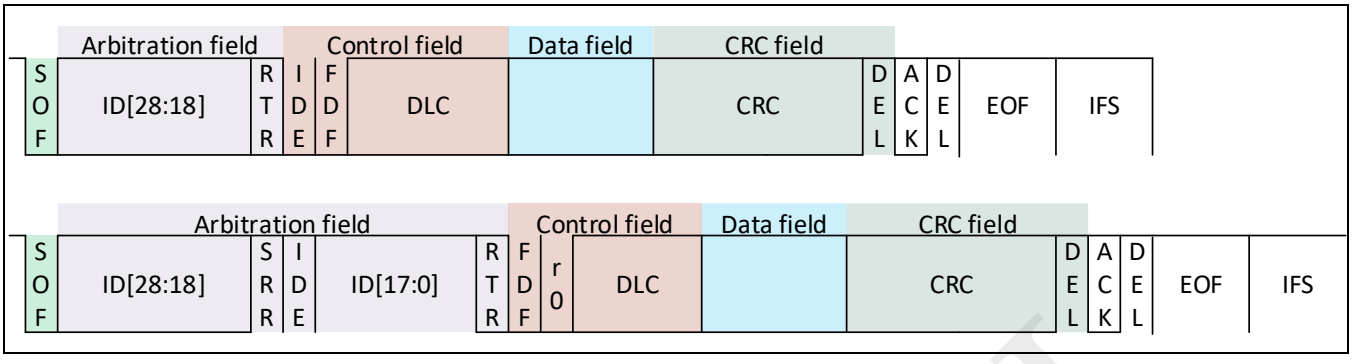
SPINTROL

图 22-3 所示，ISO 经典 CAN 支持标准帧和扩展帧两种帧格式，其标识符分别是 11 位和 29 位。两种帧的 CRC 位段都是 15 位。更多细节可以参见 ISO-11898-1:2015 标准。

SPINTROL

©2025 旋智电子科技（上海）有限公司

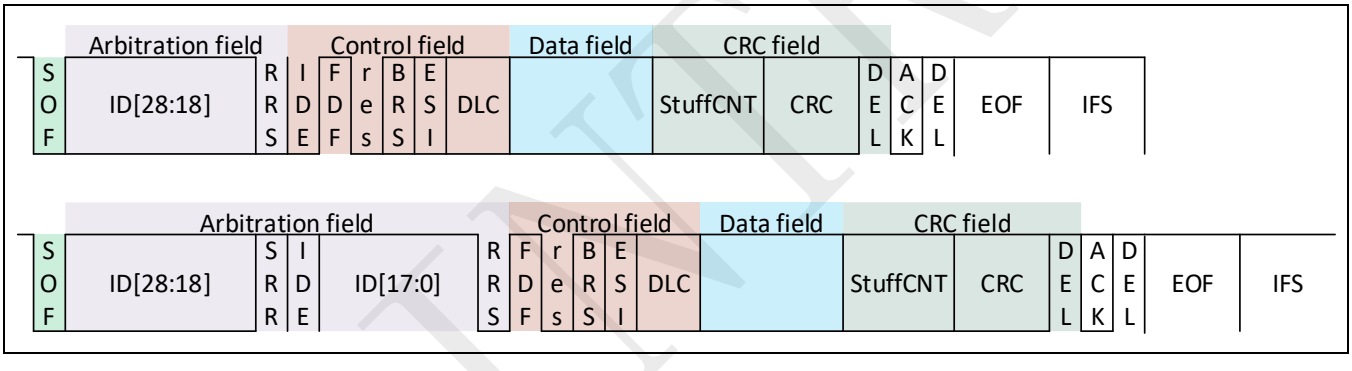
图 22-3：ISO 经典 CAN 帧格式



22.3.2 ISO FDCAN

如图 22-4 所示，ISO FDCAN 支持两种帧格式，且仅支持数据帧的传输。更多细节可以参见 ISO-11898-1:2015 标准。

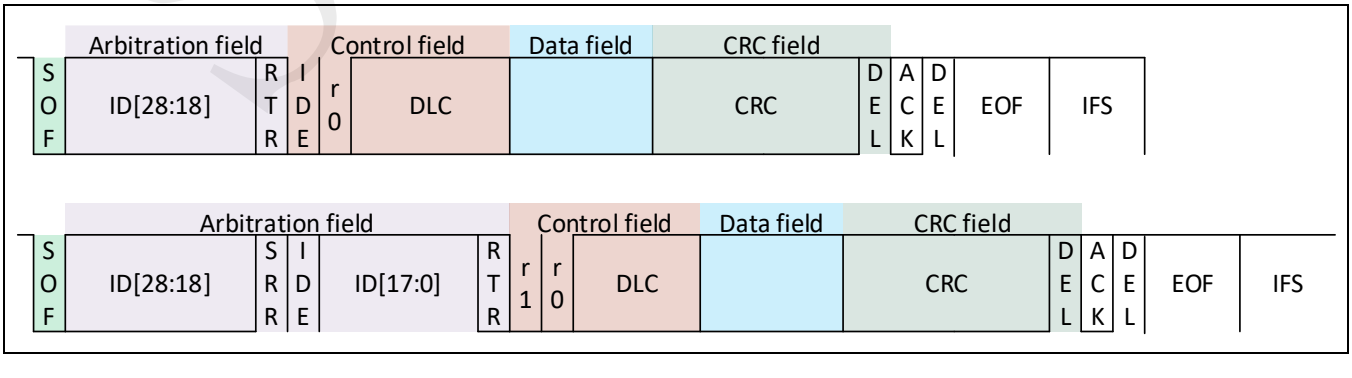
图 22-4：ISO FDCAN 帧格式



22.3.3 BOSCH CAN2.0B

如图 22-5 所示，BOSCH CAN2.0B 支持两种帧格式，与 ISO 经典 CAN 帧格式相同。更多细节可以参见 BOSCH CAN2.0B 标准。

图 22-5：BOSCH CAN2.0B 帧格式

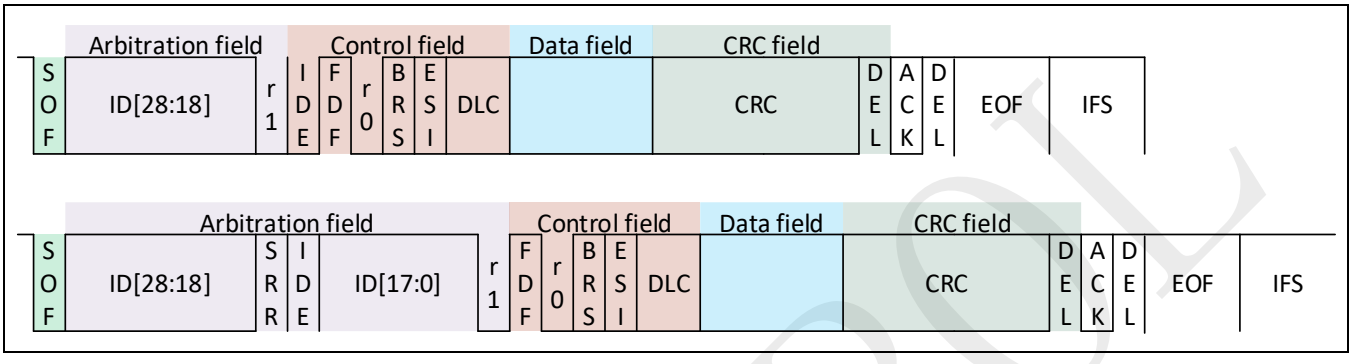


©2025 旋智电子科技（上海）有限公司

22.3.4 NONISO FDCAN

如图 22-6 所示，NONISO FDCAN 支持两种帧格式，且仅支持数据帧的传输。与 ISO FDCAN 相比，NONISO FDCAN 的 CRC 位段没有填充位数（stuff count），且采用了不同的 CRC 算法。更多细节可以参见 BOSCH FDCAN V1.0 标准。

图 22-6：NONISO FDCAN 帧格式

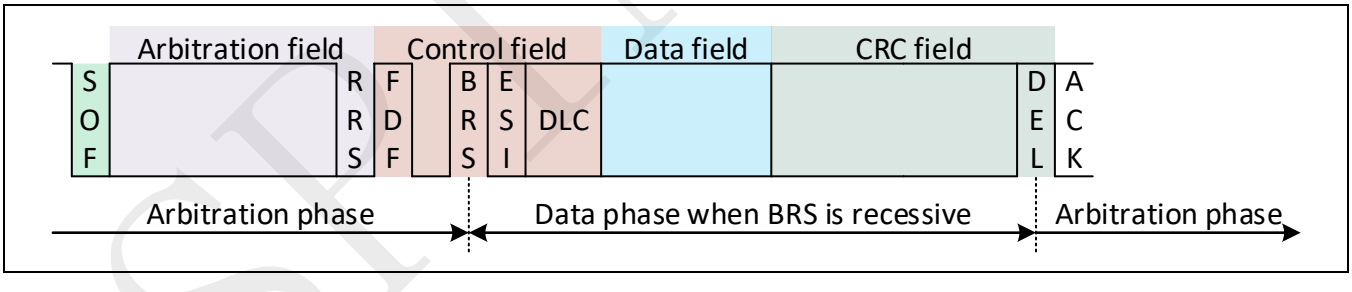


22.3.5 仲裁阶段和数据阶段

对于启用了速率切换的 ISO-FDCAN 或 NONISO-FDCAN 帧传输，有两个具有独立速率配置的阶段。如图 22-7 所示，BRS 位采样点之后到 CRC 定界符采样点之前的传输称为数据阶段。所有其他位传输称为仲裁阶段。数据阶段的速率通常比仲裁阶段的速率快。

对于禁用速率切换的 FDCAN 帧或经典 CAN 帧，只有仲裁阶段可用。

图 22-7：仲裁阶段和数据阶段

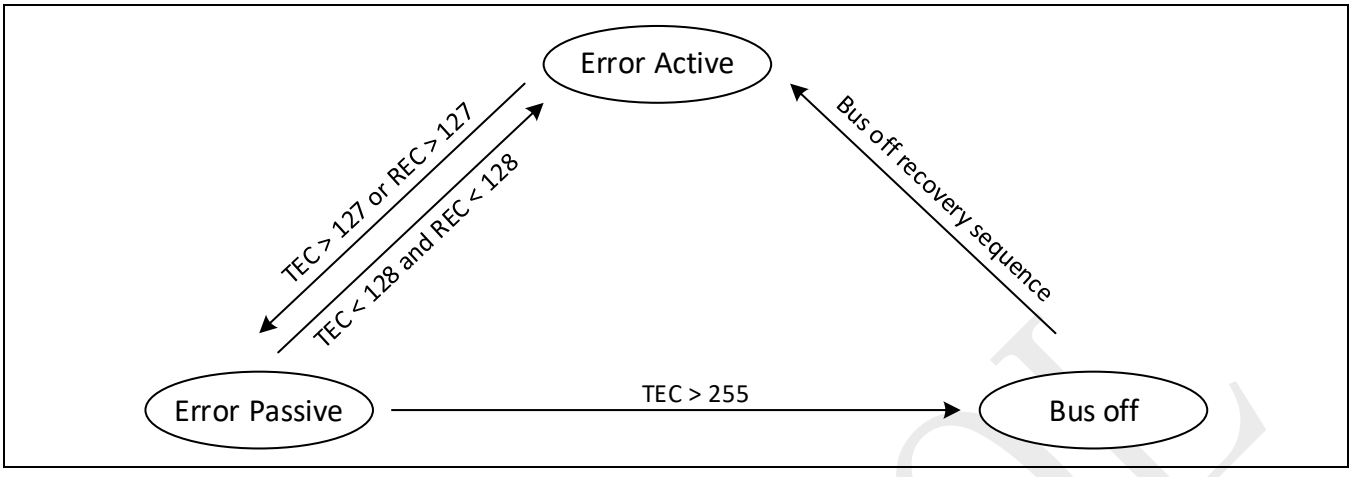


22.4 CAN 错误状态

CAN 节点的错误状态如图 22-8 所示。TEC 是发送器错误计数，REC 是接收器错误计数。相应规范中定义了为各种 CAN 模式更新 TEC 和 REC 的规则。

©2025 旋智电子科技（上海）有限公司

图 22-8：CAN 节点错误状态转换



22.4.1 错误记录

CAN 错误包括位错误、填充错误、CRC 错误、格式错误和 ACK 错误。仲裁阶段发生的错误记录在 CANLEC.LEC 中，数据阶段发生的错误记录在 CANLEC.DLEC 中。

22.4.2 主动错误状态

当 TEC 和 REC 均低于 128 时，CAN 节点处于主动错误（Error Active）状态，并在检测到错误事件时发送主动错误标志。一旦 TEC 或 REC 增加超过 CANECCTL.ERRWARN 中定义的阈值，将生成错误警告中断。

22.4.3 被动错误状态

当 TEC 或 REC 大于 127 时，CAN 节点处于被动错误（Error Passive）状态，并在检测到错误事件时发送被动错误标志。当节点进入被动错误状态时，会产生被动错误中断。当 TEC 和 REC 都下降到 128 以下时，节点再次进入主动错误状态。

如果 CAN 节点由于 REC 增加到 127 以上而 TEC 保持在 128 以下而进入被动错误状态，则 REC 将在报文成功接收后加载为 CANECCTL.RECLD。

22.4.4 总线关闭恢复

当 TEC 增加到 255 以上时，CAN 节点进入总线关闭状态并生成总线关闭中断。如果启用总线关闭自动恢复，则总线关闭恢复序列将在指定的延迟后启动。用户还可以通过设置 CANCTL.RECOVEN 立即启动总线关闭恢复序列。

在总线关闭恢复序列期间，每检测到 11 个连续的隐性位，REC 将减 1。同时，产生一个位 0 错误中断。当 REC 达到 0 时，总线关闭恢复序列完成，CAN 节点已经准备好再次正常传输报文。

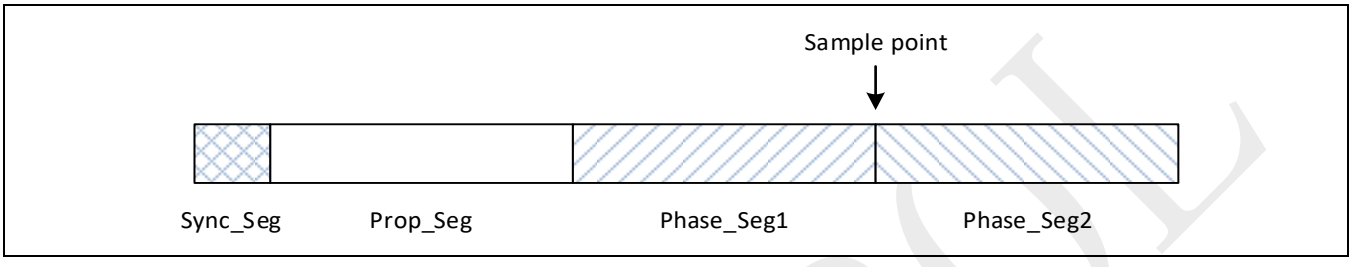
©2025 旋智电子科技（上海）有限公司

22.5 CAN 操作

22.5.1 速率配置

CAN 帧中的每一位可分为 4 段，如图 22-9 所示。每个持续时间均以模块时钟预分频器生成的特定时间量子（Time-quantum）表示。

图 22-9：位时序



Sync_Seg 称为同步段，用于同步总线上的各种 CAN 节点。预期将在此段内检测到下降沿。

Prop_Seg 称为传播时间段，用于补偿网络内的物理延迟，包括总线上的信号传播时间和 CAN 节点的内部延迟。

Phase_Seg1 和 Phase_Seg2 称为相位缓冲段 1 和 2，它们用于补偿边缘相位误差。这些段可能会因重新同步而延长或缩短。采样点正好在 Phase_Seg1 和 Phase_Seg2 之间的边界处。

仲裁阶段的速率称为标称速率，通过 CANNBTQ 寄存器配置，可以计算为：

f_{CLK_CAN}/(NBRP+1)/(1+NSEG1+1+NSEG2+1)。

数据阶段的速率称为数据速率，通过 CANDBTQ 寄存器配置，可以计算为：

f_{CLK_CAN}/(DBRP+1)/(1+DSEG1+1+DSEG2+1)。

Prop_Seg 和 Phase_Seg1 合并在一起，总持续时间通过 CANNBTQ 或 CANDBTQ 寄存器中的 SEG1 字段配置。

注意：	CANNBTQ.NBRP 和 CANDBTQ.DBRP 必须相等，否则可能导致传输异常。

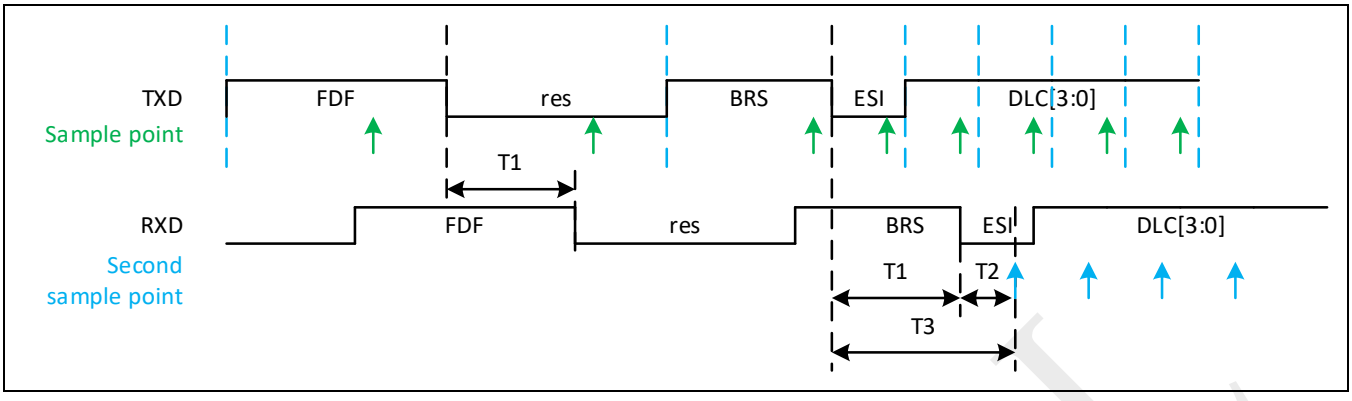
在启用 BRS 的 FDCAN 帧传输期间，支持发送器延迟补偿以生成第二个采样点。它可以检查采样数据是否与传输数据匹配。如图 22-10 所示：

- T1 是从 TXD 发送低电平到 RXD 检测到低电平的测量延迟。它必须长于通过 CANTDC.TDCW 指定的延迟滤波窗口。否则，指定的延迟窗口将用作测量的延迟；
- T2 是由 CANTDC.TDCO 设置的发送器延迟偏移；
- T3 是由 T1 和 T2 相加的发送器总延迟。它可以通过 CANTDC.TDCV 读取。

在 BRS 位采样点之后，CAN 节点从标称速率变为数据速率。第二个采样点是在每个 TXD 位的总发送器延迟之后生成的。

©2025 旋智电子科技（上海）有限公司

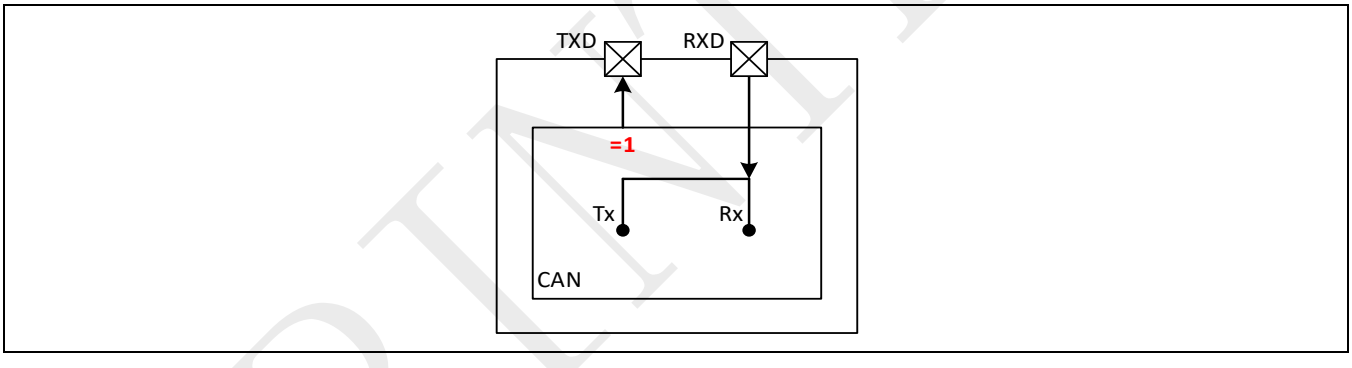
图 22-10：发送器延时补偿



22.5.2 监控模式

如图 22-11 所示，在监控模式下（CANCTL.MONMODE=1），CAN 节点不能发送帧，只能接收有效数据帧或远程帧。TXD 引脚保持隐性。内部 txd 位直接环回以生成 ACK 位、过载标志和主动错误标志等情况所需的显性位。

图 22-11：监控模式



22.5.3 限制模式

在限制模式下（CANCTL.RESMODE=1），CAN 节点能够接收数据帧、远程帧并确认有效帧。但是，它不能发送数据帧、远程帧、主动错误帧和过载帧。在发生错误事件或过载事件时，节点进入集成状态（integration state）并在检测到空闲条件（11 个连续的隐性采样位）时恢复。TEC 和 REC 在限制模式下不受影响，而错误事件检测仍处于活动状态。

限制模式不能与监控模式或任何环回模式一起使用。

22.5.4 测试模式

在测试模式下（CANCTL.TESTMODE=1），用户可以启用内部环回模式（CANTEST.ILPBK）或外部环回模式（CANTEST.ELPBK）。用户还可以通过 CANTEST.TXMODE 寄存器位将 TXD 输出设置为始终为低、始终为高或采样点模式。当 CANCTL.TESTMODE 位被清除时，寄存器 CANTEST 中的 ILPBK、ELPBK 和 TXMODE 将被重置为默认值。

©2025 旋智电子科技（上海）有限公司

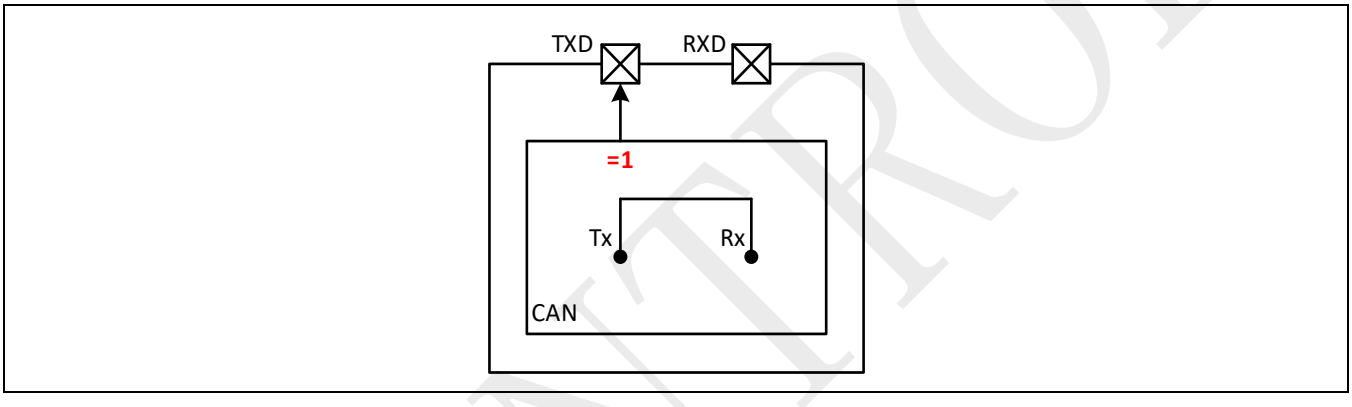
当寄存器 CANTEST 中的 ILPBK、ELPBK 和 TXMODE 位保持默认值时，CAN 节点可以在测试模式下正常使用。

环回模式不能与受限模式或监控模式一起使用，并且内部环回模式和外部环回模式不能同时启用。

22.5.4.1 内部环回

内部环回模式的连接如图 22-12 所示。TXD 总是隐性的。内部 rxd 从 RXD 引脚断开。在帧传输期间，当需要来自其他节点的 ACK 位时，内部逻辑将生成一个 ACK 位到 rxd。这样环回测试就不会产生 ACK 错误。

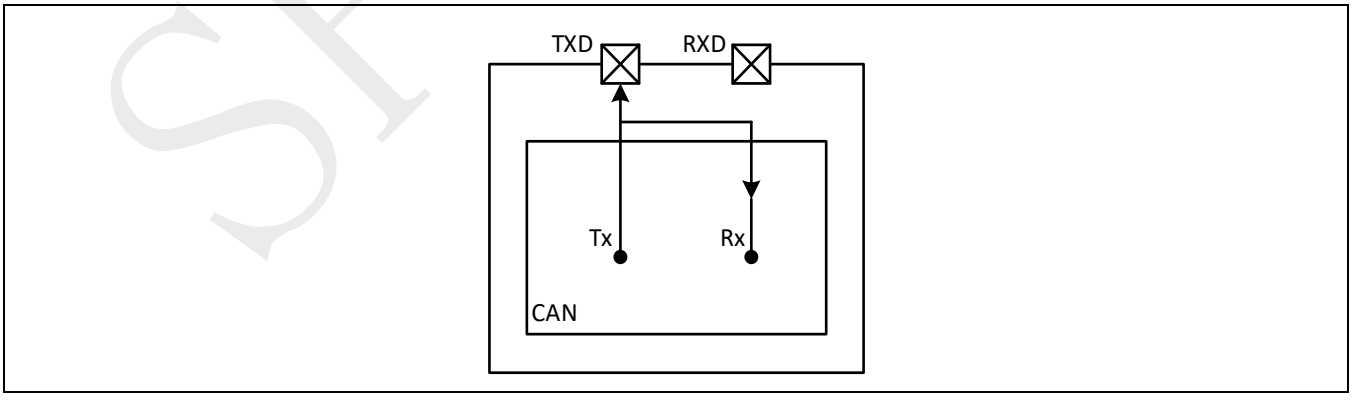
图 22-12：内部环回模式



22.5.4.2 外部环回

如图 22-13 所示，在外部环回模式下，CAN 节点的行为类似于内部环回模式。但是 TXD 引脚是实际发送的数据，并连接到 rxd。

图 22-13：外部环回模式



22.5.4.3 测试 TXD 输出

用户可以通过寄存器 CANTEST.TXMODE 将 TXD 输出设置为始终为低、始终为高或采样点模式。对于采样点模式，TXD 在采样点输出显性，在驱动点输出隐性。

©2025 旋智电子科技（上海）有限公司

22.6 信箱控制

信箱的报文存放于 RAM 内，该 RAM 在模块使能前必须初始化成全 0，从而避免因为上电后 RAM 内容的不确定性导致异常。

22.6.1 信箱信息

表 22-2：信箱控制位

EN	DIR	RMTEN	NEW	LOST	TXREQ	IF	IE	EOB	OVWR	LOCK
0	0	0	0	0	0	0	0	0	0	0

信箱的控制位如表 22-2 所示。

- EN：使能当前信箱。
- DIR：控制报文数据传输方向。
  - 0：信箱当前用于接收报文数据。
  - 1：信箱当前用于发送报文数据。
- RMTEN：使能自动应答接收到的远程帧，或自动发送远程帧以请求数据。
  - 当 DIR = 0 时，先发送远程帧到其他节点，然后等待并接收其他节点应答的数据。
  - 当 DIR = 1 时，发送数据帧来应答接收到的来自其他节点的远程帧。
- NEW：新报文标志。
  - 当 DIR = 0 时，使能信箱前应先清除本标志，硬件在成功收到报文后会将本标志置位。注意，若使能了 EOB 和 OVWR 控制位段，且 NEW 已经置位，则接收到标识符匹配的新报文时，硬件会清除 NEW 标志，同时将 LOST 置位。当整个报文接收完成后硬件将再次将 NEW 标志置位。
  - 当 DIR = 1 时，使能信箱前应先置位本标志。
    若使能了自动重发，本标志在报文开始发送时自动清零；
    若禁用了自动重发，本标志在报文完成发送时自动清零。
- LOST：报文数据丢失标志。
  - 当 DIR 为 0 且 NEW 已经置位，本标志在接收到标识符匹配的新报文时被置位。
  - 当 DIR 为 1 时，本标志无用。
- TXREQ：帧发送请求。
  - 当 DIR = 0：
    若 RMTEN = 0，应当在使能信箱前清除 TXREQ。
    若 RMTEN = 1，应当在使能信箱前将 TXREQ 置位。如果禁用了自动重发，TXREQ 在本信箱开始发送远程帧时自动清零，否则在完成远程帧发送时自动清零。
  - 当 DIR = 1：
    若 RMTEN = 0，应当在使能本信箱前将 TXREQ 置位。如果禁用了自动重发，TXREQ 在本信箱开始发送报文时自动清零，否则在完成报文发送时自动清零。
    若 RMTEN = 1，应当在使能信箱前清除 TXREQ。TXREQ 在接收到标识符匹配的远程帧请求时自动置位。如果禁用了自动重发，TXREQ 在开始发送应答的数据帧时自动清零，否则

©2025 旋智电子科技（上海）有限公司

在完成应答的数据帧发送时自动清零。

- IF：报文传输完成中断标志。
- IE：报文传输完成中断使能。
- EOB：编号连续的信箱可以分组为一个块。该位将当前信箱设置为块的最后一个信箱。同一个块中的所有信箱都可以接收报文，并且可能具有相似的标识符掩码过滤条件。信箱用于报文发送时，该位无用。
- OVWR：使能用新收到的报文覆盖信箱里的旧内容。本位段仅对于使能了 EOB 的信箱有用。
- LOCK：信箱在传输报文的过程中，该位自动置位，从而禁止 AHB 总线对信箱的写操作。传输完成后，该位自动清零，恢复 AHB 总线对信箱的写权限。

注意： 报文总是存放于同时符合标识符匹配且未被使用两个条件的第一个信箱里。用户应当先处理好存放于同一个信箱块里的报文，再使能下一个信箱块的报文接收。

22.6.2 非自动应答远程帧时的报文发送

表 22-3：用于普通报文发送的信箱控制位设置

EN	DIR	RMTEN	NEW	LOST	TXREQ	IF	IE	EOB	OVWR	LOCK
1	1	0	1	0	0	0	0/1	0	0	0

用户可以按照如下步骤采用轮询模式发送报文：

1. 等待硬件清除 NEW 和 TXREQ，表明信箱准备就绪，可以用于报文发送。
2. 通过 FID、FDF、BRS、ESICTL、ESI、DLC、FDW[0]到 FDW[15]寄存器配置报文信息。推荐将报文超出 DLC 范围的数据段填充 0xCC。
3. 按照表 22-3 配置信箱控制位。
4. 等待硬件清除 NEW 和 TXREQ 位，表明报文已经成功发送。
5. 在信箱真正开始工作前可以取消报文传输。用户可以通过清除 MCTL.EN 位来禁用信箱，并且必须回读该位以检查它是否被成功清除，然后可以重新配置此信箱用于其他用途。如果无法清除 MCTL.EN，则表明此信箱中的报文传输正在进行中，用户必须等到它完成。

22.6.3 自动应答远程帧时的报文发送

表 22-4：用于自动应答远程帧时的报文发送信箱控制位设置

EN	DIR	RMTEN	NEW	LOST	TXREQ	IF	IE	EOB	OVWR	LOCK
1	1	1	1	0	0	0	0/1	0	0	0

©2025 旋智电子科技（上海）有限公司

用户可以按照如下步骤采用轮询模式发送报文：

1. 等待硬件清除 NEW 和 TXREQ，表明信箱准备就绪，可以用于报文发送。
2. 使用 FID 和 FMSK 寄存器配置远程帧标识符掩码过滤器。RTR 位掩码被接收处理程序忽略，接收到的 RTR 必须为 1。FID 在接收到的标识符比较期间用作参考标识符。当接收到的标识符和 FMSK 做与运算后，其结果和（FID & FMSK）相同时，可以将帧保存到当前信箱。
3. 通过 ESICTL、FDW[0]到 FDW[15]寄存器配置应答的报文信息。推荐将报文超出 DLC 范围的数据段填充 0xCC。
4. 按照表 22-4 配置信箱控制位。
5. 当接收到标识符匹配的远程帧时，硬件自动将 TXREQ 置位，并按照接收到的远程帧更新 FID（RTR 位不变）、FDF、BRS、ESI、DLC 寄存器，然后信箱开始发送数据帧来应答远程节点的请求。
6. 等待硬件清除 NEW 和 TXREQ 位，表明报文已经成功发送。
7. 在信箱真正开始工作前可以取消报文传输。用户可以通过清除 MCTL.EN 位来禁用信箱，并且必须回读该位以检查它是否被成功清除，然后可以重新配置此信箱用于其他用途。如果无法清除 MCTL.EN，则表明此信箱中的报文传输正在进行中，用户必须等到它完成。

22.6.4 未对其他节点发出远程帧数据请求时的报文接收

表 22-5：用于普通报文接收的信箱控制位设置

EN	DIR	RMTEN	NEW	LOST	TXREQ	IF	IE	EOB	OVWR	LOCK
1	0	0	0	0	0	0	0/1	0	0	0

用户可以按照如下步骤采用轮询模式接收报文：

1. 使用 FID 和 FMSK 寄存器配置远程帧标识符掩码过滤器。FID 在接收到的标识符比较期间用作参考标识符。当接收到的标识符和 FMSK 做与运算后，其结果和（FID & FMSK）相同时，可以将帧保存到当前信箱。
2. 按照表 22-5 配置信箱控制位。
3. 等待 NEW 被自动置位，表明接收到了标识符匹配的报文。
4. 在信箱真正开始工作前可以取消报文传输。用户可以通过清除 MCTL.EN 位来禁用信箱，并且必须回读该位以检查它是否被成功清除，然后可以重新配置此信箱用于其他用途。如果无法清除 MCTL.EN，则表明此信箱中的报文传输正在进行中，用户必须等到它完成。

©2025 旋智电子科技（上海）有限公司

22.6.5 对其他节点发出远程帧数据请求时的报文接收

表 22-6：用于接收其他节点应答远程帧的报文的信箱控制位设置

EN	DIR	RMTEN	NEW	LOST	TXREQ	IF	IE	EOB	OVWR	LOCK
1	0	1	0	0	1	0	0/1	0	0	0

用户可以按照如下步骤采用轮询模式接收报文：

1. 通过 FID、FDF、BRS、ESICTL、ESI、DLC 寄存器配置报文信息。
2. 使用 FID 和 FMSK 寄存器配置远程帧标识符掩码过滤器。RTR 位掩码应当清零，它被发送处理程序忽略并总是将 RTR 发送为 1。FID 在接收到的标识符比较期间用作参考标识符。当接收到的标识符和 FMSK 做与运算后，其结果和（FID & FMSK）相同时，可以将帧保存到当前信箱。FID 同时作为远程帧标识符以发送数据请求给其他节点。
3. 按照表 22-6 配置信箱控制位。
4. 远程帧发送后 TXREQ 被自动清零，然后信箱开始等待其他节点的应答报文。
5. 等待 NEW 被自动置位，表明从其他节点接收到了标识符匹配的应答报文。
6. 在信箱真正开始工作前可以取消报文传输。用户可以通过清除 MCTL.EN 位来禁用信箱，并且必须回读该位以检查它是否被成功清除，然后可以重新配置此信箱用于其他用途。如果无法清除 MCTL.EN，则表明此信箱中的报文传输正在进行中，用户必须等到它完成。

SPINTROL

©2025 旋智电子科技（上海）有限公司

22.7 寄存器

22.7.1 CAN 寄存器列表

表 22-7：CAN 模块基地址

外设模块	基地址
CAN	0x40005400

表 22-8：CAN 寄存器列表

寄存器	偏移地址	描述	复位值
CANCTL*	0x00	CAN 控制寄存器	0x00000060
CANNBTQ*	0x04	CAN 标称位时间量子寄存器	0x00090403
CANDBTQ*	0x08	CAN 数据位时间量子寄存器	0x00090403
CANTDC*	0x0C	CAN 发送器延时补偿寄存器	0x00000000
CANBRDLY*	0x10	CAN 总线关闭恢复延时寄存器	0x00000000
CANSTS	0x14	CAN 状态寄存器	0x00000000
CANLEC	0x18	CAN 最新错误码寄存器	0x00000000
CANMIE*	0x1C	CAN 杂项中断使能寄存器	0x00000000
CANMRAWIF	0x20	CAN 杂项中断原始事件标志寄存器	0x00000000
CANMIF	0x24	CAN 杂项中断标志寄存器	0x00000000
CANMIC	0x28	CAN 杂项中断清除寄存器	0x00000000
CANTGIF	0x2C	CAN 信箱组传输完成中断标志寄存器	0x00000000
CANTGIC	0x30	CAN 信箱组传输完成中断清除寄存器	0x00000000
CANMEMERRADDR	0x34	CAN 内存错误地址寄存器	0x00000000
CANTEST*	0x38	CAN 测试控制寄存器	0x00000030
CANECCTL*	0x3C	CAN 错误计数控制寄存器	0x0000307F
CANTXREQ	0x40	CAN 发送请求寄存器 n([0:1])	0x00000000
CANMSGNEW	0x50	CAN 新报文标志寄存器 n([0:1])	0x00000000
CANMSGIF	0x60	CAN 信箱传输完成中断标志寄存器 n([0:1])	0x00000000
CANMBEN	0x70	CAN 信箱使能寄存器 n([0:1])	0x00000000
CANTSCTL*	0x80	CAN 时间戳控制寄存器	0x00000000
CANTSCNT*	0x84	CAN 时间戳计数寄存器	0x00000000
CANREGKEY	0x8C	CAN 内存错误注入样式寄存器 n([0:1])	0x1ACCE551

注意： 由*标记的寄存器仅当 CANREGKEY=0x1ACCE551 时才可以改写。

©2025 旋智电子科技（上海）有限公司

22.7.2 CAN 寄存器

表 22-9：CAN 控制寄存器（CANCTL）位段定义

CANCTL (CAN Control Register) Offset：0x00 Default：0x00000060 Access：CAN->CANCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							ECCEN
15	14	13	12	11	10	9	8
NONISO	FDEN	TSTMODE	MONMODE	RESMODE	EDGEFLT	TXPAUSE	PXEN
7	6	5	4	3	2	1	0
IOSWAP	RETX	AUTOON	RECOVEN	RST	RESERVED		EN

表 22-10：CAN 控制寄存器（CANCTL）位段描述

位段	位段名	属性	复位值	描述
31:17	RESERVED_31_17	RO	0x0	保留
16	ECCEN	RW	0x0	报文内存 ECC 使能 0：关闭 1：使能
15	NONISO	RW	0x0	NON-ISO CAN 格式模式 本位段仅当 CANSTS.OP = 0 时允许更改 0：ISO11898-1:2015 定义的格式 1：BOSCH CAN2.0B 和 FDCAN_V1.0 定义的格式
14	FDEN	RW	0x0	FDCAN 使能 本位段仅当 CANSTS.OP = 0 时允许更改 0：关闭 1：使能
13	TSTMODE	RW	0x0	测试模式使能 本位段仅当 CANSTS.OP = 0 时允许更改 0：关闭 1：使能
12	MONMODE	RW	0x0	监控模式使能 本位段仅当 CANSTS.OP = 0 时允许写 1 任何时候都允许写 0 0：关闭 1：使能
11	RESMODE	RW	0x0	限制模式使能 本位段仅当 CANSTS.OP = 0 时允许写 1 任何时候都允许写 0

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				0：关闭 1：使能
10	EDGEFLT	RW	0x0	集成（Integration）状态下的边沿滤波使能 0：关闭 1：使能，隐性位后出现两个连续的显性位才判定为下降沿
9	TXPAUSE	RW	0x0	在连续发送两个帧之间插入 2 个标称位 0：关闭 1：使能
8	PXEN	RW	0x0	协议异常使能 本位段仅当 CANSTS.OP = 0 时允许更改 0：关闭 1：使能
7	IOSWAP	RW	0x0	使能 TXD/RXD 引脚交换 本位段仅当 CANSTS.OP = 0 时允许更改 0：关闭 1：使能
6	RETX	RW	0x1	使能自动重发未成功发送的帧 本位段仅当 CANSTS.OP = 0 时允许更改 0：关闭 1：使能
5	AUTOON	RW	0x1	使能总线自动恢复，从而总线关闭后经过由 CANBRDLY 指定的延时，开始自动恢复 调试模式下不支持本模式。 0：关闭 1：使能
4	RECOVEN	W1C	0x0	使能总线关闭恢复 本位段不受 CANREGKEY 保护 0：写 0 无效，读总是 0 1：写 1 立即启动总线关闭恢复序列，本位段自动清零
3	RST	W1C	0x0	软件复位 用户应当先往 CANCTL.EN 写 0 关闭 CAN 模块后再发起软件复位. 直到 CAN 进入空闲状态时才会真正执行复位 0：写 0 无效 1：写 1 请求复位，复位结束后位段自动清零
2:1	RESERVED_2_1	RO	0x0	保留
0	EN	RW	0x0	CAN 模块使能

SPINTROL

位段	位段名	属性	复位值	描述
				0：关闭 1：使能

表 22-11：CAN 标称位时间量子寄存器（CANNBTQ）位段定义

CANNBTQ (CAN Nominal Bit Timing Quantum Register) Offset：0x04 Default：0x00090403 Access：CAN->CANNBTQ							
31 RESERVED	30	29	28	27 NSJW	26	25	24
23 RESERVED	22	21	20	19 NSEG2	18	17	16
15	14	13	12	11 NSEG1	10	9	8
7	6	5	4	3 NBRP	2	1	0

表 22-12：CAN 标称位时间量子寄存器（CANNBTQ）位段描述

位段	位段名	属性	复位值	描述
31	RESERVED_31	RO	0x0	保留
30:24	NSJW	RW	0x0	设定标称速率下的同步跳转宽度为(NSJW+1) NTq 本位段仅当 CANSTS.OP = 0 时允许更改
23	RESERVED_23	RO	0x0	保留
22:16	NSEG2	RW	0x9	设定标称速率下采样点后的时段宽度为(NSEG2+1) NTq NSEG2 不能设为 0 本位段仅当 CANSTS.OP = 0 时允许更改
15:8	NSEG1	RW	0x4	设定标称速率下采样点前的时段宽度为(NSEG1+1) NTq NSEG1 不能设为 0 本位段仅当 CANSTS.OP = 0 时允许更改
7:0	NBRP	RW	0x3	标称速率预分频值 仲裁阶段的时间量子 NTq 被设为(NBRP+1)个 CAN 模块时钟周期 本位段仅当 CANSTS.OP = 0 时允许更改

©2025 旋智电子科技（上海）有限公司

表 22-13：CAN 数据位时间量子寄存器（CANDBTQ）位段定义

CANDBTQ (CAN Data Bit Timing Quantum Register) Offset：0x08 Default：0x00090403
Access：CAN->CANDBTQ

31	30	29	28	27	26	25	24
TDC	RESERVED			DSJW			
23	22	21	20	19	18	17	16
RESERVED				DSEG2			
15	14	13	12	11	10	9	8
RESERVED			DSEG1				
7	6	5	4	3	2	1	0
DBRP							

表 22-14：CAN 数据位时间量子寄存器（CANDBTQ）位段描述

位段	位段名	属性	复位值	描述
31	TDC	RW	0x0	发送器延时补偿使能 本位段仅当 CANSTS.OP = 0 时允许更改 0：关闭 1：使能
30:28	RESERVED_30_28	RO	0x0	保留
27:24	DSJW	RW	0x0	设定数据速率下的同步跳转宽度为(DSJW+1) DTq 本位段仅当 CANSTS.OP = 0 时允许更改
23:20	RESERVED_23_20	RO	0x0	保留
19:16	DSEG2	RW	0x9	设定数据速率下采样点后的时段宽度为(DSEG2+1) DTq DSEG2 不能设为 0 本位段仅当 CANSTS.OP = 0 时允许更改
15:13	RESERVED_15_13	RO	0x0	保留
12:8	DSEG1	RW	0x4	设定数据速率下采样点前的时段宽度为(DSEG1+1) DTq DSEG1 不能设为 0 本位段仅当 CANSTS.OP = 0 时允许更改
7:0	DBRP	RW	0x3	数据速率预分频值 数据阶段的时间量子 DTq 被设为(DBRP+1)个 CAN 模块时钟周期 本位段仅当 CANSTS.OP = 0 时允许更改 注意：DBRP 应当设定为与 NBRP 相同的值

SPINTROL

表 22-15：CAN 发送器延时补偿寄存器（CANTDC）位段定义

CANTDC (CAN Transmitter Delay Compensation Register) Offset：0x0C Default：0x00000000 Access：CAN->CANTDC							
31	30	29	28	27	26	25	24
RESERVED		TDCV					
23	22	21	20	19	18	17	16
TDCV				TDCW			
15	14	13	12	11	10	9	8
TDCW						TDCO	
7	6	5	4	3	2	1	0
TDCO							

表 22-16：CAN 发送器延时补偿寄存器（CANTDC）位段描述

位段	位段名	属性	复位值	描述
31:30	RESERVED_31_30	RO	0x0	保留
29:20	TDCV	RO	0x0	以 DTq 计算的发送器延时补偿值 它是 TDCO 和测量得到的延时的和
19:10	TDCW	RW	0x0	设定发送器延时补偿滤波窗口大小为 TDCW DTq
9:0	TDCO	RW	0x0	设定发送器延时补偿第二采样点偏移量为 TDCO DTq

表 22-17：CAN 总线关闭恢复延时寄存器（CANBRDLY）位段定义

CANBRDLY (CAN Bus-Off Recovery Delay Register) Offset：0x10 Default：0x00000000 Access：CAN->CANBRDLY							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
DLY							
7	6	5	4	3	2	1	0
DLY							

表 22-18：CAN 总线关闭恢复延时寄存器（CANBRDLY）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15:0	DLY	RW	0x0	总线关闭恢复的延时 若 CANCTL.AUTOON=1，则总线关闭后经过(DLY+1)个标称位时间后自动开始恢复。

©2025 旋智电子科技（上海）有限公司

表 22-19：CAN 状态（CANSTS）位段定义

CANSTS (CAN Status Register) Offset：0x14 Default：0x00000000 Access：CAN->CANSTS							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED				OPMODE			ERRWARN
15	14	13	12	11	10	9	8
ERRPASSIVE	REC						
7	6	5	4	3	2	1	0
TEC							

表 22-20：CAN 状态（CANSTS）位段描述

位段	位段名	属性	复位值	描述
31:20	RESERVED_31_20	RO	0x0	保留
19:17	OPMODE	RO	0x0	工作模式 0：停止状态 1：集成状态 2：空闲状态 3：忙于发送报文 4：忙于接收报文 6：总线关闭
16	ERRWARN	RO	0x0	错误警告 0：TEC 和 REC 均低于 CANECCTL.ERRWARN 1：TEC 或 REC 不低于 CANECCTL.ERRWARN
15	ERRPASSIVE	RO	0x0	被动错误状态 0：总线错误时发送主动错误帧 1：总线错误时发送被动错误帧
14:8	REC	RO	0x0	接收器错误计数值，其范围为[0, 128] 注意：总线关闭恢复过程中每检测到 11 个 连续的隐性位，计数值减 1
7:0	TEC	RO	0x0	发送器错误计数值，其范围为[0, 255]

SPINTROL

表 22-21：CAN 最新错误码寄存器（CANLEC）位段定义

CANLEC (CAN Last Error Code Register) Offset：0x18 Default：0x00000000 Access：CAN->CANLEC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							WRLCK
15	14	13	12	11	10	9	8
RESERVED					DLEC		
7	6	5	4	3	2	1	0
RESERVED					LEC		

表 22-22：CAN 最新错误码寄存器（CANLEC）位段描述

位段	位段名	属性	复位值	描述
31:17	RESERVED_31_17	RO	0x0	保留
16	WRLCK	RC	0x0	对锁定的信箱（正在传输数据）进行写操作（会被忽略） 推荐按字节读取本位段 0：事件未发生 1：事件已发生
15:11	RESERVED_15_11	RO	0x0	保留
10:8	DLEC	RS	0x0	数据阶段的错误码，推荐按字节读取本位段 0：默认为 0，报文传输完成后本位段清零 1：填充位错误 2：格式错误 4：位 1 错误（发送 1 但总线回读 0） 5：位 0 错误（发送 0 但总线回读 1） 7：从上次位段读取后未发生错误 每次本位段被读取后值自动恢复为 7
7:3	RESERVED_7_3	RO	0x0	保留
2:0	LEC	RS	0x0	仲裁阶段的错误码，推荐按字节读取本位段 0：默认为 0，报文传输完成后本位段清零 1：填充位错误 2：格式错误 3：发送的报文未收到其他节点的回应 4：位 1 错误（发送 1 但总线回读 0） 5：位 0 错误（发送 0 但总线回读 1） 总线关闭恢复过程中，每检测到 11 个隐性位本位段就会被设为 5，由此 CPU 可以检测恢复的进程（总线未被卡在显性位） 6：接收到的报文有 CRC 错误

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				7：每次本位段被读取后值自动恢复为 7

表 22-23：CAN 杂项中断使能寄存器（CANMIE）位段定义

CANMIE (CAN Miscellaneous Interrupt Enable Register)　Offset：0x1C　Default：0x00000000
Access：CAN->CANMIE

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						MSGLOST	TSOV
7	6	5	4	3	2	1	0
MEMMBE	MEMSBE	PXEVT	FLTOK	BUSOFF	ERRPASSIVE	ERRWARN	TRANERR

表 22-24：CAN 杂项中断使能寄存器（CANMIE）位段描述

位段	位段名	属性	复位值	描述
31:10	RESERVED_31_10	RO	0x0	保留
9	MSGLOST	RW	0x0	报文丢失中断使能 0：关闭 1：使能
8	TSOV	RW	0x0	时间戳计数溢出中断使能 0：关闭 1：使能
7	MEMMBE	RW	0x0	内存多比特错误中断使能 0：关闭 1：使能
6	MEMSBE	RW	0x0	内存单比特错误中断使能 0：关闭 1：使能
5	PXEVT	RW	0x0	协议异常事件中断使能 0：关闭 1：使能
4	FLTOK	RW	0x0	接收到标识符匹配的报文中断使能 0：关闭 1：使能
3	BUSOFF	RW	0x0	进入总线关闭状态中断使能

SPINTROL

位段	位段名	属性	复位值	描述
				0：关闭 1：使能
2	ERRPASSIVE	RW	0x0	进入被动错误状态中断使能 0：关闭 1：使能
1	ERRWARN	RW	0x0	错误警告中断使能 0：关闭 1：使能
0	TRANERR	RW	0x0	报文传输错误中断使能 0：关闭 1：使能

表 22-25：CAN 杂项中断原始事件标志寄存器（CANMRAWIF）位段定义

CANMRAWIF (CAN Miscellaneous Interrupt Raw Flag Register) Offset：0x20 Default：0x00000000 Access：CAN->CANMRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						MSGLOST	TSOV
7	6	5	4	3	2	1	0
MEMMBE	MEMSBE	PXEVT	FLTOK	BUSOFF	ERRPASSIVE	ERRWARN	TRANERR

表 22-26：CAN 杂项中断原始事件标志寄存器（CANMRAWIF）位段描述

位段	位段名	属性	复位值	描述
31:10	RESERVED_31_10	RO	0x0	保留
9	MSGLOST	RO	0x0	报文丢失事件标志 0：事件未发生 1：事件已发生
8	TSOV	RO	0x0	时间戳计数溢出事件标志 0：事件未发生 1：事件已发生
7	MEMMBE	RO	0x0	内存多比特错误事件标志 0：事件未发生 1：事件已发生
6	MEMSBE	RO	0x0	内存单比特错误事件标志

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				0：事件未发生 1：事件已发生
5	PXEVT	RO	0x0	协议异常事件标志 0：事件未发生 1：事件已发生
4	FLTOK	RO	0x0	接收到标识符匹配的报文事件标志 0：事件未发生 1：事件已发生
3	BUSOFF	RO	0x0	进入总线关闭状态事件标志 0：事件未发生 1：事件已发生
2	ERRPASSIVE	RO	0x0	进入被动错误状态事件标志 0：事件未发生 1：事件已发生
1	ERRWARN	RO	0x0	错误警告事件标志 0：事件未发生 1：事件已发生
0	TRANERR	RO	0x0	报文传输错误事件标志 0：事件未发生 1：事件已发生

表 22-27：CAN 杂项中断标志寄存器（CANMIF）位段定义

CANMIF (CAN Miscellaneous Interrupt Flag Register)　Offset：0x24　Default：0x00000000
Access：CAN->CANMIF

31	30	29	28	27	26	25	24
INT	RESERVED						
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						MSGLOST	TSOV
7	6	5	4	3	2	1	0
MEMMBE	MEMSBE	PXEVT	FLTOK	BUSOFF	ERRPASSIVE	ERRWARN	TRANERR

SPINTROL

表 22-28：CAN 杂项中断标志寄存器（CANMIF）位段描述

位段	位段名	属性	复位值	描述
31	INT	RO	0x0	CAN 全局杂项中断标志 0：中断未发生 1：中断已产生，并已经通知 CPU 本位段被清除之前不会产生新的杂项中断
30:10	RESERVED_30_10	RO	0x0	保留
9	MSGLOST	RO	0x0	报文丢失中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
8	TSOV	RO	0x0	时间戳计数溢出中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
7	MEMMBE	RO	0x0	内存多比特错误中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
6	MEMSBE	RO	0x0	内存单比特错误中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
5	PXEVT	RO	0x0	协议异常中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
4	FLTOK	RO	0x0	接收到标识符匹配的报文中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
3	BUSOFF	RO	0x0	进入总线关闭状态中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
2	ERRPASSIVE	RO	0x0	进入被动错误状态中断标志 0：中断未发生

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				1：中断已发生 本位段被清除前新事件不再触发中断
1	ERRWARN	RO	0x0	错误警告中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
0	TRANERR	RO	0x0	报文传输错误中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断

表 22-29：CAN 杂项中断清除寄存器（CANMIC）位段定义

CANMIC (CAN Miscellaneous Interrupt Flag Clear Register) Offset：0x28 Default：0x00000000
Access：CAN->CANMIC

31	30	29	28	27	26	25	24
INT	RESERVED						
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						MSGLOST	TSOV
7	6	5	4	3	2	1	0
MEMMBE	MEMSBE	PXEVT	FLTOK	BUSOFF	ERRPASSIVE	ERRWARN	TRANERR

表 22-30：CAN 杂项中断清除寄存器（CANMIC）位段描述

位段	位段名	属性	复位值	描述
31	INT	W1C	0x0	CAN 全局杂项中断清除 0：写 0 无效，读总是 0 1：写 1 清除中断标志，本位段自动清零
30:10	RESERVED_30_10	RO	0x0	保留
9	MSGLOST	W1C	0x0	报文丢失中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，位段自动清零
8	TSOV	W1C	0x0	时间戳计数溢出中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
7	MEMMBE	W1C	0x0	内存多比特错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，位段自动清零
6	MEMSBE	W1C	0x0	内存单比特错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，位段自动清零
5	PXEVT	W1C	0x0	协议异常中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，位段自动清零
4	FLTOK	W1C	0x0	接收到标识符匹配的报文中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，位段自动清零
3	BUSOFF	W1C	0x0	进入总线关闭状态中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，位段自动清零
2	ERRPASSIVE	W1C	0x0	进入被动错误状态中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，位段自动清零
1	ERRWARN	W1C	0x0	错误警告中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，位段自动清零
0	TRANERR	W1C	0x0	报文传输错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，位段自动清零

表 22-31：CAN 信箱组传输完成中断标志寄存器（CANTGIF）位段定义

CANTGIF (CAN Mailbox Group Transfer Done Interrupt Flag Register) Offset：0x2C Default：0x00000000 Access：CAN->CANTGIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 GRP7	6 GRP6	5 GRP5	4 GRP4	3 GRP3	2 GRP2	1 GRP1	0 GRP0

©2025 旋智电子科技（上海）有限公司

表 22-32：CAN 信箱组传输完成中断标志寄存器（CANTGIF）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	GRP7	RO	0x0	信箱组 7 传输完成中断标志 信箱 56~63 中任意一个完成传输后本位段置位 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
6	GRP6	RO	0x0	信箱组 6 传输完成中断标志 信箱 48~55 中任意一个完成传输后本位段置位 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
5	GRP5	RO	0x0	信箱组 5 传输完成中断标志 信箱 40~47 中任意一个完成传输后本位段置位 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
4	GRP4	RO	0x0	信箱组 4 传输完成中断标志 信箱 32~39 中任意一个完成传输后本位段置位 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
3	GRP3	RO	0x0	信箱组 3 传输完成中断标志 信箱 24~31 中任意一个完成传输后本位段置位 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
2	GRP2	RO	0x0	信箱组 2 传输完成中断标志 信箱 16~23 中任意一个完成传输后本位段置位 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
1	GRP1	RO	0x0	信箱组 1 传输完成中断标志 信箱 8~15 中任意一个完成传输后本位段置位 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断

SPINTROL

位段	位段名	属性	复位值	描述
0	GRP0	RO	0x0	信箱组 0 传输完成中断标志 信箱 0~7 中任意一个完成传输后本位段置位 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断

表 22-33：CAN 信箱组传输完成中断清除寄存器（CANTGIC）位段定义

CANTGIC (CAN Mailbox Group Transfer Done Interrupt Clear Register) Offset：0x30 Default：0x00000000 Access：CAN->CANTGIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 GRP7	6 GRP6	5 GRP5	4 GRP4	3 GRP3	2 GRP2	1 GRP1	0 GRP0

表 22-34：CAN 信箱组传输完成中断清除寄存器（CANTGIC）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	GRP7	W1C	0x0	信箱组 7 传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，位段自动清零
6	GRP6	W1C	0x0	信箱组 6 传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，位段自动清零
5	GRP5	W1C	0x0	信箱组 5 传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，位段自动清零
4	GRP4	W1C	0x0	信箱组 4 传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，位段自动清零
3	GRP3	W1C	0x0	信箱组 3 传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，位段自动清零
2	GRP2	W1C	0x0	信箱组 2 传输完成中断清除

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，位段自动清零
1	GRP1	W1C	0x0	信箱组 1 传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，位段自动清零
0	GRP0	W1C	0x0	信箱组 0 传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，位段自动清零

表 22-35：CAN 内存错误地址寄存器（CANMEMERRADDR）位段定义

CANMEMERRADDR (CAN Memory Error Address Register) Offset：0x34 Default：0x00000000 Access：CAN->CANMEMERRADDR							
31	30	29	28	27	26	25	24
ADDR							
23	22	21	20	19	18	17	16
ADDR							
15	14	13	12	11	10	9	8
ADDR							
7	6	5	4	3	2	1	0
ADDR							

表 22-36：CAN 内存错误地址寄存器（CANMEMERRADDR）位段描述

位段	位段名	属性	复位值	描述
31:0	ADDR	RO	0x0	内存访问发生 ECC 错误的地址

表 22-37：CAN 测试控制寄存器（CANTEST）位段定义

CANTEST (CAN Test Control Register) Offset：0x38 Default：0x00000030 Access：CAN->CANTEST							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 RESERVED	6 RESERVED	5 RXD	4 TXD	3	2 TXMODE	1 ELOOP	0 LOOP

SPINTROL

表 22-38：CAN 测试控制寄存器（CANTEST）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6	RESERVED_6	RW	0x0	保留
5	RXD	RO	0x1	RXD 引脚监测到的电平 0：低 1：高
4	TXD	RO	0x1	TXD 引脚监测到的电平 0：低 1：高
3:2	TXMODE	RW	0x0	TXD 引脚控制 关闭 CANCTL.TESTMODE 时本位段复位 0：TXD 引脚由 CAN 协议引擎驱动 1：采样点处输出低，驱动点处输出高 由此可以通过 TXD 引脚观测采样点位置 2：输出低 3：输出高
1	ELOOP	RW	0x0	外部环回模式使能 关闭 CANCTL.TESTMODE 时本位段复位 0：关闭 1：使能
0	LOOP	RW	0x0	内部环回模式使能 关闭 CANCTL.TESTMODE 时本位段复位 0：关闭 1：使能

表 22-39：CAN 错误计数控制寄存器（CANECCTL）位段定义

CANECCTL (CAN Error Counter Control Register)　Offset：0x3C　Default：0x0000307F
Access：CAN->CANECCTL

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED		ERRWARN					
7	6	5	4	3	2	1	0
ERRWARN	RECLD						

©2025 旋智电子科技（上海）有限公司

表 22-40：CAN 错误计数控制寄存器（CANECCTL）位段描述

位段	位段名	属性	复位值	描述
31:14	RESERVED_31_14	RO	0x0	保留
13:7	ERRWARN	RW	0x60	错误警告阈值，取值范围为[80, 127]
6:0	RECLD	RW	0x7F	REC 大于 127 时帧接收成功后 REC 的复位值 取值范围为[119, 127]

表 22-41：CAN 发送请求寄存器 n ([0:1])（CANTXREQ）位段定义

CANTXREQ (CAN Transmission Request Register) Offset：0x40 Default：0x00000000 Access：CAN->CANTXREQ							
31	30	29	28	27	26	25	24
			VAL				
23	22	21	20	19	18	17	16
			VAL				
15	14	13	12	11	10	9	8
			VAL				
7	6	5	4	3	2	1	0
			VAL				

表 22-42：CAN 发送请求寄存器 n ([0:1])（CANTXREQ）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	每位对应的信箱（0~63）的发送请求 0：该信箱未请求发送 1：该信箱有发送请求正在等待或处理中

表 22-43：CAN 新报文标志寄存器 n ([0:1])（CANMSGNEW）位段定义

CANMSGNEW (CAN New Message Register) Offset：0x50 Default：0x00000000 Access：CAN->CANMSGNEW							
31	30	29	28	27	26	25	24
			VAL				
23	22	21	20	19	18	17	16
			VAL				
15	14	13	12	11	10	9	8
			VAL				
7	6	5	4	3	2	1	0
			VAL				

©2025 旋智电子科技（上海）有限公司

表 22-44：CAN 新报文标志寄存器 n ([0:1])（CANMSGNEW）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	每位对应的信箱（0~63）的新报文标志 0：该信箱未写入新报文 1：该信箱有新报文写入

表 22-45：CAN 信箱传输完成中断标志寄存器 n ([0:1])（CANMSGIF）位段定义

CANMSGIF (CAN Mailbox Transfer Done Interrupt Flag Register) Offset：0x60 Default：0x00000000 Access：CAN->CANMSGIF							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 22-46：CAN 信箱传输完成中断标志寄存器 n ([0:1])（CANMSGIF）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	每位对应的信箱（0~63）的传输完成中断标志 0：该信箱未触发传输完成中断 1：该信箱已触发传输完成中断

表 22-47：CAN 信箱使能寄存器 n ([0:1])（CANMBEN）位段定义

CANMBEN (CAN Mailbox Enable Register) Offset：0x70 Default：0x00000000 Access：CAN->CANMBEN							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

©2025 旋智电子科技（上海）有限公司

表 22-48：CAN 信箱使能寄存器 n ([0:1])（CANMBEN）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	使能每位对应的信箱（0~63） 0：关闭该信箱，不用于报文传输 1：使能该信箱，用于报文传输

表 22-49：CAN 时间戳控制寄存器（CANTSCTL）位段定义

CANTSCTL (CAN Timestamp Control Register) Offset：0x80 Default：0x00000000 Access：CAN->CANTSCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							TSEN

表 22-50：CAN 时间戳控制寄存器（CANTSCTL）位段描述

位段	位段名	属性	复位值	描述
31:1	RESERVED_31_1	RO	0x0	保留
0	TSEN	RW	0x0	时间戳使能 0：关闭 1：使能

表 22-51：CAN 时间戳计数寄存器（CANTSCNT）位段定义

CANTSCNT (CAN Timestamp Counter Register) Offset：0x84 Default：0x00000000 Access：CAN->CANTSCNT							
31	30	29	28	27	26	25	24
TSCNT							
23	22	21	20	19	18	17	16
TSCNT							
15	14	13	12	11	10	9	8
TSCNT							
7	6	5	4	3	2	1	0
TSCNT							

©2025 旋智电子科技（上海）有限公司

表 22-52：CAN 时间戳计数寄存器（CANTSCNT）位段描述

位段	位段名	属性	复位值	描述
31:0	TSCNT	RW	0x0	时间戳计数器在采样点向上计数 读本寄存器返回当前计数值 写本寄存器更新计数值，计数器在下一个采样点基于更新后的值继续向上计数

表 22-53：CAN 模块写使能寄存器（CANREGKEY）位段定义

CANREGKEY (CAN Register Write-Allow Key Register) Offset：0x8C Default：0x1ACCE551
Access：CAN->CANREGKEY

31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

表 22-54：CAN 模块写使能寄存器（CANREGKEY）位段描述

位段	位段名	属性	复位值	描述
31:0	KEY	RW	0x1ACCE551	写入 0x1ACCE551 解锁受保护的 CAN 寄存器

SPINTROL

©2025 旋智电子科技（上海）有限公司

22.7.3 CANMBOX 寄存器列表

表 22-55：CANMBOX 模块基地址

外设模块	基地址
CAN->CANCANMBOX	0x40004000

表 22-56：CANMBOX 寄存器列表

寄存器	偏移地址	描述	复位值
CANMBOXFDW	0x00	CAN 信箱报文数据字 n ([0:15])寄存器	0x00000000
CANMBOXFMSK	0x40	CAN 信箱报文标识符掩码寄存器	0x00000000
CANMBOXFID	0x44	CAN 信箱报文标识符寄存器	0x00000000
CANMBOXMTS	0x48	CAN 信箱报文时间戳寄存器	0x00000000
CANMBOXMCTL	0x4C	CAN 信箱报文控制寄存器	0x00000000

SPINTROL

22.7.4 CANMBOX 寄存器

表 22-57：CAN 信箱报文数据字 n ([0:15])寄存器（CANMBOXFDW）位段定义

CANMBOXFDW (CAN Mailbox Data Word Register n) Offset：0x00 Default：0x00000000 Access：CAN->CANMBOX[m].CANMBOXFDW[n] (m is from 0 to 63, n is from 0 to 15)							
31	30	29	28	27	26	25	24
BYTE3							
23	22	21	20	19	18	17	16
BYTE2							
15	14	13	12	11	10	9	8
BYTE1							
7	6	5	4	3	2	1	0
BYTE0							

表 22-58：CAN 信箱报文数据字 n ([0:15])寄存器（CANMBOXFDW）位段描述

位段	位段名	属性	复位值	描述
31:24	BYTE3	RW	0x0	报文数据第(4n + 3)字节 推荐将不用的字节填充为 0xCC
23:16	BYTE2	RW	0x0	报文数据第(4n + 2)字节 推荐将不用的字节填充为 0xCC
15:8	BYTE1	RW	0x0	报文数据第(4n + 1)字节 推荐将不用的字节填充为 0xCC
7:0	BYTE0	RW	0x0	报文数据第(4n + 0)字节 推荐将不用的字节填充为 0xCC

表 22-59：CAN 信箱报文标识符掩码寄存器（CANMBOXFMSK）位段定义

CANMBOXFMSK (CAN Mailbox Frame Mask Register) Offset：0x40 Default：0x00000000 Access：CAN->CANMBOX[m].CANMBOXFMSK (m is from 0 to 63)							
31 RESERVED	30 RTR	29 IDE	28	27	26 ID	25	24
23	22	21	20	19	18	17	16
ID							
15	14	13	12	11	10	9	8
ID							
7	6	5	4	3	2	1	0
ID							

©2025 旋智电子科技（上海）有限公司

表 22-60：CAN 信箱报文标识符掩码寄存器（CANMBOXFMSK）位段描述

位段	位段名	属性	复位值	描述
31	RESERVED_31	RO	0x0	保留
30	RTR	RW	0x0	远程帧请求位掩码 0：标识符过滤匹配时 RTR 位不参与比较 1：标识符过滤匹配时 RTR 位参与比较
29	IDE	RW	0x0	扩展帧位掩码 0：标识符过滤匹配时 IDE 位不参与比较 1：标识符过滤匹配时 IDE 位参与比较
28:0	ID	RW	0x0	标识符掩码 接收到的 IDE 为 0 时只有 ID[28:18]有效 0：标识符过滤匹配时对应位不参与比较 1：标识符过滤匹配时对应位参与比较

表 22-61：CAN 信箱报文标识符寄存器（CANMBOXFID）位段定义

CANMBOXFID (CAN Mailbox Frame Identifier Register)　Offset：0x44　Default：0x00000000
Access：CAN->CANMBOX[m].CANMBOXFID (m is from 0 to 63)

31	30	29	28	27	26	25	24
RESERVED	RTR	IDE	ID				
23	22	21	20	19	18	17	16
ID							
15	14	13	12	11	10	9	8
ID							
7	6	5	4	3	2	1	0
ID							

表 22-62：CAN 信箱报文标识符寄存器（CANMBOXFID）位段描述

位段	位段名	属性	复位值	描述
31	RESERVED_31	RO	0x0	保留
30	RTR	RW	0x0	远程帧请求 0：报文为数据帧 1：报文为远程帧
29	IDE	RW	0x0	扩展帧标识 0：报文为标准帧 1：报文为远程帧
28:0	ID	RW	0x0	报文标识符 IDE = 0：标识符为 ID[28:18] IDE = 1：标识符为 ID[28:0]

©2025 旋智电子科技（上海）有限公司

表 22-63：CAN 信箱报文时间戳寄存器（CANMBOXMTS）位段定义

CANMBOXMTS (CAN Mailbox Message Timestamp Register) Offset：0x48 Default：0x00000000 Access：CAN->CANMBOX[m].CANMBOXMTS (m is from 0 to 63)							
31	30	29	28	27	26	25	24
TS							
23	22	21	20	19	18	17	16
TS							
15	14	13	12	11	10	9	8
TS							
7	6	5	4	3	2	1	0
TS							

表 22-64：CAN 信箱报文时间戳寄存器（CANMBOXMTS）位段描述

位段	位段名	属性	复位值	描述
31:0	TS	RW	0x0	时间戳 用户应当只在初始化时写本寄存器，传输完成后可以回读硬件更新后的有效值

表 22-65：CAN 信箱报文控制寄存器（CANMBOXMCTL）位段定义

CANMBOXMCTL (CAN Mailbox Message Control Register) Offset：0x4C Default：0x00000000 Access：CAN->CANMBOX[m].CANMBOXMCTL (m is from 0 to 63)							
31 EN	30 DIR	29 RMTEN	28 NEW	27 LOST	26 TXREQ	25 IF	24 IE
23 EOB	22 OVWR	21 LOCK	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 ESICTL	6 FDF	5 BRS	4 ESI	3	2	1	0
DLC							

表 22-66：CAN 信箱报文控制寄存器（CANMBOXMCTL）位段描述

位段	位段名	属性	复位值	描述
31	EN	RW	0x0	信箱使能 0：关闭 1：使能
30	DIR	RW	0x0	报文传输方向 0：接收 1：发送

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
29	RMTEN	RW	0x0	远程帧自动应答使能 0：关闭 1：使能
28	NEW	RW	0x0	新报文标志 0：旧报文 1：新报文
27	LOST	RW	0x0	接收到的报文丢失 用户不应该往本位写 1 0：没有报文丢失 1：有接收到的报文丢失
26	TXREQ	RW	0x0	发送请求 0：关闭 1：使能
25	IF	RW	0x0	中断标志 用户不应该往本位写 1 0：中断未发生 1：中断发送
24	IE	RW	0x0	信箱传输完成中断使能 0：关闭 1：使能
23	EOB	RW	0x0	信箱分组块最后一个信箱标记 0：本信箱未用作信箱分组块最后一个信箱 1：本信箱用作信箱分组块最后一个信箱
22	OVWR	RW	0x0	信箱分组块最后一个信箱改写使能 0：关闭 1：使能
21	LOCK	RW	0x0	信箱锁定 用户不应该往本位写 1 0：信箱未被锁定（允许 AHB 写访问） 1：信箱被锁定（禁止 AHB 写访问）
20:8	RESERVED_20_8	RW	0x0	保留
7	ESICTL	RW	0x0	错误状态软件改写使能 0：ESI 由 CAN 协议硬件控制 1：ESI 由 CANMBOXMCTL.ESI 寄存器位段控制
6	FDF	RW	0x0	FDCAN 格式使能 0：关闭 1：使能

SPINTROL

位段	位段名	属性	复位值	描述
5	BRS	RW	0x0	速率切换 0：关闭 1：使能
4	ESI	RW	0x0	错误状态 ESICTL = 0 时写本位段无意义 0：主动错误 1：被动错误
3:0	DLC	RW	0x0	数据长度码 对于非 FDCAN 格式，若本位段写入值大于 8，则实际传输的报文数据为 8 字节 0： 0 字节 1： 1 字节 2： 2 字节 3： 3 字节 4： 4 字节 5： 5 字节 6： 6 字节 7： 7 字节 8： 8 字节 9：12 字节 10：16 字节 11：20 字节 12：24 字节 13：32 字节 14：48 字节 15：64 字节

SPINTROL

©2025 旋智电子科技（上海）有限公司

23 循环冗余校验（CRC）

23.1 概述

循环冗余校验（Cyclic Redundancy Check，CRC）或者叫做多项式码校验是一种哈希函数，被设计用于检查原始计算数据的意外变化。因此，CRC 被用于数据传输或者存储过程中的错误检测方面。CRC 校验码最多为 32 比特。

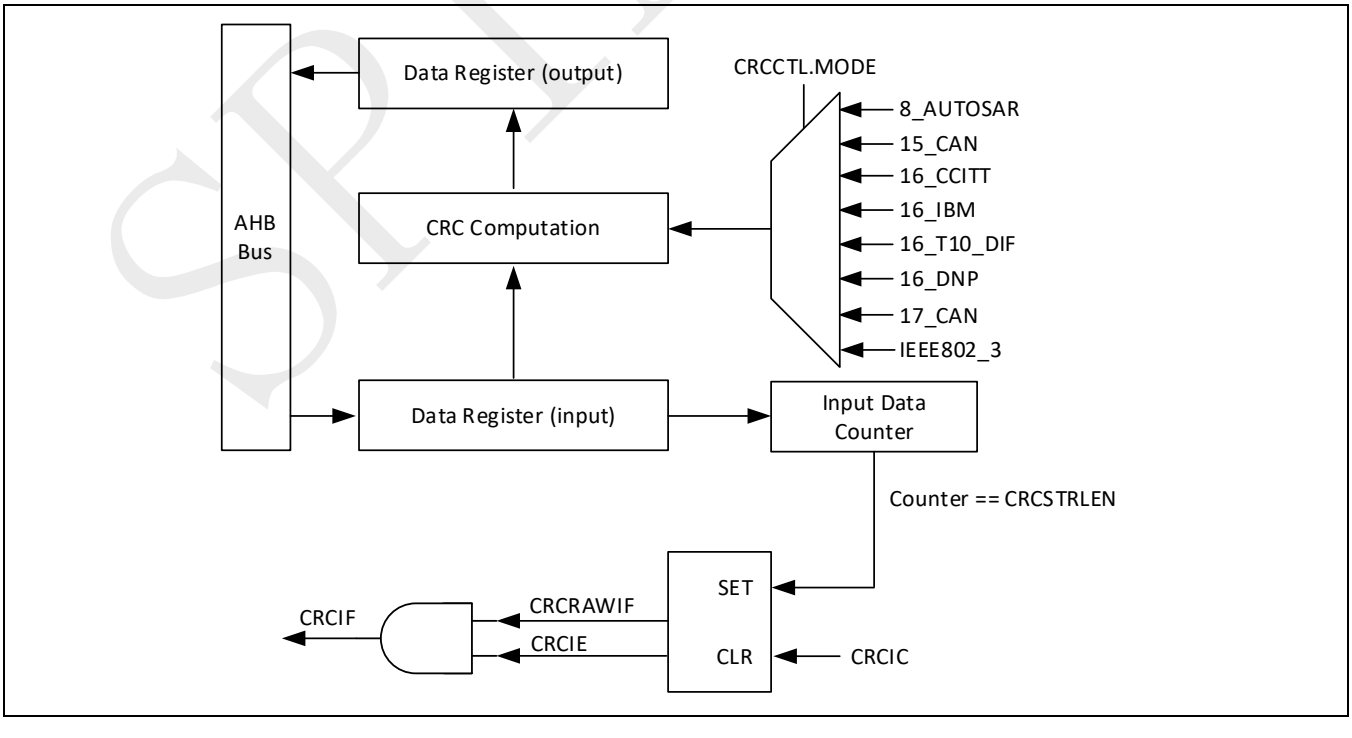
23.2 主要特性

SPC1169 提供 1 个 CRC 模块，其特性如下：

- 支持 32 位并行数据比特流输入，支持多达 32 位 CRC 值输出
- 支持多达 2^{32}（4294967296）字节长度的数据的 CRC 计算
- 支持以下 CRC 标准
  - CRC-8-AUTOSAR：$x^8+x^5+x^3+x^2+x+1$
  - CRC-15-CAN：$x^{15}+x^{14}+x^{10}+x^8+x^7+x^4+x^3+1$
  - CRC-16-CCITT：$x^{16}+x^{12}+x^5+1$
  - CRC-16-IBM：$x^{16}+x^{15}+x^2+1$
  - CRC-16-T10-DIF：$x^{16}+x^{15}+x^{11}+x^9+x^8+x^7+x^5+x^4+x^2+1$
  - CRC-16-DNP：$x^{16}+x^{13}+x^{12}+x^{11}+x^{10}+x^8+x^6+x^5+x^2+1$
  - CRC-17-CAN：$x^{17}+x^{16}+x^{14}+x^{13}+x^{11}+x^6+x^4+x^3+x+1$
  - CRC-32-IEEE 802.3：$X^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$

CRC 模块框图如图 23-1 所示。

图 23-1：CRC 模块框图



SPINTROL

23.3 CRC 工作流程

用户可以根据以下步骤操作 CRC 模块：

步骤 1：关闭 CRC（设置 CRCCTL.EN 位为 0）

步骤 2：关闭中断（设置 CRCIE.DONE 位为 0）

步骤 3：清除中断标志（设置 CRCIC.DONE 为 1）

步骤 4：配置数据流长度（设置 CRCSTRLEN 寄存器）

步骤 5：配置 CRC 工作模式（设置 CRCCTL.MODE 位段)

步骤 6：使能中断（设置 CRCIE.DONE 位为 1)

步骤 7：使能 CRC（设置 CRCCTL.EN 位为 1)

步骤 8：写入数据流，等待中断发生（如果中断发生，转到步骤 9）

步骤 9：获取 CRC 计算结果（读 CRCRESULT 寄存器）

步骤 10：CRC 操作完成

注意 CRC 数据流输入寄存器一次接受一个字（32 位）数据。如果输入数据不是 4 字节对齐的，则会在数据流的开始填充 0。例如，如果数据流由 5 个字节构成，从低地址开始依次为：0xA1，0xA2，0xA3，0xA4，0xA5，那么下面两个字数据应该被写入数据流输入寄存器：

- 0xA1000000
- 0xA5A4A3A2

CRC 结果的位序为：

- 16 位 CRC：x_0~x_{15} [MSB->LSB]
- 32 位 CRC：x_0~x_{31} [MSB->LSB]

SPINTROL

©2025 旋智电子科技（上海）有限公司

23.4 寄存器

23.4.1 CRC 寄存器列表

表 23-1：CRC 模块基地址

外设模块	基地址
CRC	0x4000 3600

表 23-2：CRC 寄存器列表

寄存器	偏移地址	描述	复位值
CRCCTL*	0x00	CRC 控制寄存器	0x00000000
CRCSTRLEN*	0x04	CRC 数据流长度寄存器	0x00000000
CRCSEED*	0x08	CRC 初值种子寄存器	0x00000000
CRCSTRIN	0x0C	CRC 数据流输入寄存器	0x00000000
CRCRESULT	0x10	CRC 结果寄存器	0x00000000
CRCRAWIF	0x14	CRC 中断原始事件标志寄存器	0x00000000
CRCIF	0x18	CRC 中断标志寄存器	0x00000000
CRCIC	0x1C	CRC 中断清除寄存器	0x00000000
CRCIE*	0x20	CRC 中断使能寄存器	0x00000000
CRCREGKEY	0x24	CRC 模块写使能寄存器	0x1ACCE551

注意： 由*标记的寄存器仅当 CRCREGKEY=0x1ACCE551 时才可以改写。

SPINTROL

23.4.2 CRC 寄存器

表 23-3：CRC 控制寄存器（CRCCTL）位段定义

CRCCTL (CRC Control Register) Access：CRCx->CRCCTL Offset：0x00 Default：0x00000000							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						PERBYTE	INVOUT
7	6	5	4	3	2	1	0
INVIN	ENDIANOUT	ENDIANIN	CONTINUOUS	MODE			EN

表 23-4：CRC 控制寄存器（CRCCTL）位段描述

位段	位段名	属性	复位值	描述
31:10	RESERVED_31_10	RO	0x0	保留
9	PERBYTE	RW	0x0	设置写 CRCSTRIN 的数据格式 0：往 CRCSTRIN 单次写入 4 个字节 1：往 CRCSTRIN 单次写入 1 个字节
8	INVOUT	RW	0x0	对输出结果取反 0：输出线性反馈移位寄存器原始的结果 1：输出线性反馈移位寄存器取反后的结果
7	INVIN	RW	0x0	对输入数据取反 0：将原始输入数据流放入线性反馈移位寄存器 1：将输入数据取反后放入线性反馈移位寄存器
6	ENDIANOUT	RW	0x0	输出结果大小端方式 0：最低位是线性反馈移位寄存器的最后一级 1：最低位是线性反馈移位寄存器的第一级
5	ENDIANIN	RW	0x0	输入数据按字节大小端方式 0：字节最低位先送入线性反馈移位寄存器 1：字节最高位先送入线性反馈移位寄存器
4	CONTINUOUS	RW	0x0	数据流级联运算 0：设置 CRCCTL.EN=0 关闭模块时，内部线性反馈移位寄存器清零，因此下次重新开启时是从全新状态开始 1：设置 CRCCTL.EN=0 关闭模块时，内部线性反馈移位寄存器保持不变，因此下次重新开启时是

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				接着当前帧继续运算（等效于计算一个合并后的长数据流）
3:1	MODE	RW	0x0	CRC 模式选择 0：CRC-8-AUTOSAR 1：CRC-15-CAN 2：CRC-16-CCITT 3：CRC-16-IBM 4：CRC-16-T10-DIF 5：CRC-16-DNP 6：CRC-17-CAN 7：CRC-32-IEEE802.3
0	EN	RW	0x0	CRC 使能 0：关闭 1：使能

表 23-5：CRC 数据流长度寄存器（CRCSTRLEN）位段定义

CRCSTRLEN (CRC Stream Length Register) Access：CRCx->CRCSTRLEN Offset：0x04 Default：0x00000000							
31	30	29	28	27	26	25	24
				VAL			
23	22	21	20	19	18	17	16
				VAL			
15	14	13	12	11	10	9	8
				VAL			
7	6	5	4	3	2	1	0
				VAL			

表 23-6：CRC 数据流长度寄存器（CRCSTRLEN）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	设置输入数据流为(CRCSTRLEN+1)个字节

©2025 旋智电子科技（上海）有限公司

表 23-7：CRC 初值种子寄存器（CRCSEED）位段定义

CRCSEED (CRC Seed Register) Access：CRCx->CRCSEED Offset：0x08 Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 23-8：CRC 初值种子寄存器（CRCSEED）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	线性反馈移位寄存器初值

表 23-9：CRC 数据流输入寄存器（CRCSTRIN）位段定义

CRCSTRIN (CRC Stream Input Register) Access：CRCx->CRCSTRIN Offset：0x0C Default：0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 23-10：CRC 数据流输入寄存器（CRCSTRIN）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	数据流输入 CRCCTLEN.PERBYTE=1 时高 24 位回读总是 0

©2025 旋智电子科技（上海）有限公司

表 23-11：CRC 结果寄存器（CRCRESULT）位段定义

CRCRESULT (CRC Result Register) Offset：0x10 Default：0x00000000 Access：CRCx->CRCRESULT							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 23-12：CRC 结果寄存器（CRCRESULT）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	CRC 计算结果

表 23-13：CRC 中断原始事件标志寄存器（CRCRAWIF）位段定义

CRCRAWIF (CRC Interrupt Raw Flag Register) Access：CRCx->CRCRAWIF Offset：0x14 Default：0x00000000							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						STROVF	DONE

表 23-14：CRC 中断原始事件标志寄存器（CRCRAWIF）位段描述

位段	位段名	属性	复位值	描述
31:2	RESERVED_31_2	RO	0x0	保留
1	STROVF	RO	0x0	CRC 数据流上溢事件标志 0：数据流输入长度未超出设定值 1：数据流输入长度超出了设定值
0	DONE	RO	0x0	CRC 运算结束事件标志 0：CRC 运算未完成 1：CRC 运算已完成

©2025 旋智电子科技（上海）有限公司

表 23-15：CRC 中断标志寄存器（CRCIF）位段定义

CRCIF (CRC Interrupt Flag Register) Access：CRCx->CRCIF Offset：0x18 Default：0x00000000							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED					INT	STROVF	DONE

表 23-16：CRC 中断标志寄存器（CRCIF）位段描述

位段	位段名	属性	复位值	描述
31:3	RESERVED_31_3	RO	0x0	保留
2	INT	RO	0x0	CRC 全局中断标志 0：中断未发生 1：中断已产生，并已经通知 CPU 本位段被清除之前不会产生新中断
1	STROVF	RO	0x0	CRC 数据流上溢中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
0	DONE	RO	0x0	CRC 运算结束中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断

表 23-17：CRC 中断清除寄存器（CRCIC）位段定义

CRCIC (CRC Interrupt Clear Register) Access：CRCx->CRCIC Offset：0x1C Default：0x00000000							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED					INT	STROVF	DONE

©2025 旋智电子科技（上海）有限公司

表 23-18：CRC 中断清除寄存器（CRCIC）位段描述

位段	位段名	属性	复位值	描述
31:3	RESERVED_31_3	RO	0x0	保留
2	INT	W1C	0x0	全局中断清除 0：写 0 无效，读总是 0 1：写 1 清除中断标志，本位段自动清零
1	STROVF	W1C	0x0	CRC 数据流上溢中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
0	DONE	W1C	0x0	CRC 运算结束中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

表 23-19：CRC 中断使能寄存器（CRCIE）位段定义

CRCIE (CRC Interrupt Enable Register)
Access：CRCx->CRCIE　　Offset：0x20　　Default：0x00000000

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						STROVF	DONE

表 23-20：CRC 中断使能寄存器（CRCIE）位段描述

位段	位段名	属性	复位值	描述
31:2	RESERVED_31_2	RO	0x0	保留
1	STROVF	RW	0x0	CRC 数据流上溢中断使能 0：关闭 1：使能
0	DONE	RW	0x0	CRC 运算结束中断使能 0：关闭 1：使能

SPINTROL

表 23-21：CRC 模块写使能寄存器（CRCREGKEY）位段定义

CRCREGKEY (CRC Register Write-Allow Key Register) Access：CRCx->CRCREGKEY Offset：0x24 Default：0x1ACCE551							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

表 23-22：CRC 模块写使能寄存器（CRCREGKEY）位段描述

位段	位段名	属性	复位值	描述
31:0	KEY	RW	0x1ACCE551	写入 0x1ACCE551 解锁受保护的 CRC 寄存器

SPINTROL

©2025 旋智电子科技（上海）有限公司

24 Flash 存储器

24.1 特性

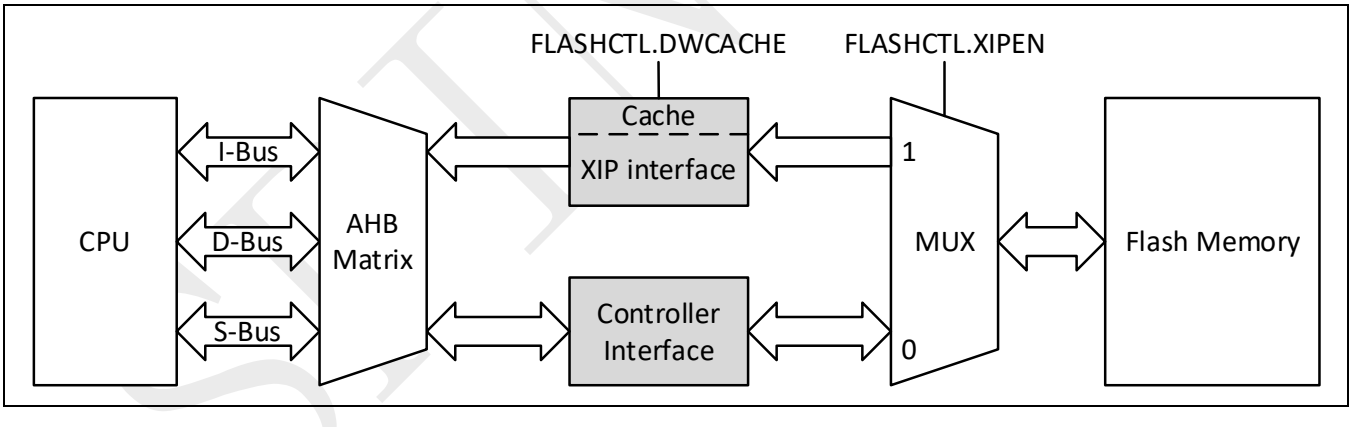
SPC1169 中的 Flash 存储器具有以下主要特性：

- 存储器最高容量可达 128KB
- 存储器组织
  - 主存储区块：包含 32 个扇区，每个扇区包含 512x64 比特位；
  - NVR 存储区块：包含 3 个扇区，每个扇区 512x64 比特位；
- 擦写寿命：最少 100,000 次擦写周期@T_J=85°C
- 数据保存时间：超过 20 年@T_J=85°C
- ECC（错误检查和纠正）保护
- 符合 AEC-Q100 1 级认证要求

如图 24-1 所示，闪存存储器接口具有以下特点：

- XIP（Execute-In-Place）读接口，支持直接使用 64 位缓存进行 AHB 读取访问；
- 基于状态机的读取/编程/擦除操作的控制器接口；
- 读出/写入保护功能；

图 24-1：Flash 存储器访问接口



24.2 Flash 架构

Flash 存储器由一个主存储区和一个 NVR 存储区构成。主存储区由 32 个扇区构成，每个扇区包含 512 个基本存储器单元（64-bit）；NVR 存储区由 3 个扇区构成，每个扇区包含 512 个基本存储器单元（64-bit）。具体信息如图 24-2 和表 24-1 所示。

NVR 存储器阵列可以用作：

- 数据存储器：3 个 NVR 扇区仅用作普通闪存存储器；

对主存储区和 NVR 存储区的写操作由控制器接口管理，如图 24-1 所示，写入/擦除操作所需的高电压由内部生成。

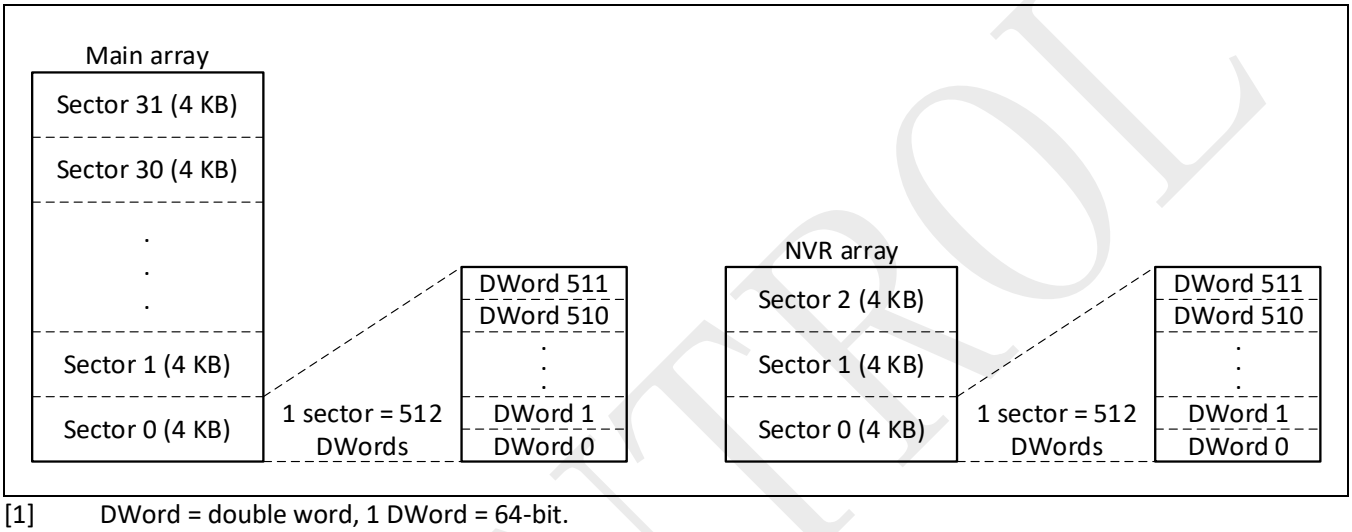
©2025 旋智电子科技（上海）有限公司

为避免不同类型的非法访问，可以对主存储区进行以下保护：

- 扇区写入保护；
- 读出保护；

注意： 通过控制器接口对闪存进行操作时，应首先禁用 XIP 接口（FLASHCTL.XIPEN=0）。

图 24-2：Flash 模块逻辑结构



[1] DWord = double word, 1 DWord = 64-bit.

表 24-1：Flash 模块构成

区块	名称	起始地址	大小
主存储区	扇区 0	0x1000 0000	4096 字节
	扇区 1	0x1000 1000	4096 字节
	扇区 2	0x1000 2000	4096 字节
	扇区 3	0x1000 3000	4096 字节
	……	……	……
	扇区 31	0x1001 F000	4096 字节
NVR 存储区	扇区 0	0x1100 2000	4096 字节
	扇区 1	0x1100 3000	4096 字节
	扇区 2	0x1100 4000	4096 字节

©2025 旋智电子科技（上海）有限公司

24.3 读操作

CPU 可以通过 XIP 接口或控制器接口从嵌入式 Flash 模块中读取数据，读取的粒度与存储单元的大小完全相同，即 1 个 DWord。

24.3.1 XIP 读接口读操作

通过 XIP 接口，Flash 存储器可以被看做一个普通的存储空间，能够以字节、半字或者字的方式直接通过地址访问。XIP 接口的主要任务就是产生读 Flash 存储器所需要的控制信号。因此，CPU 可以使用它来从 I 总线中获取指令，并从 D 总线中获取数据。

读访问操作可能会存在延迟（Latency）的情况。这个延迟是指一个正在进行的读操作所需要的等待状态（Wait state）数。等待状态数是系统时钟 SYSCLK 周期和 Flash 存储器访问时间的比值。具体信息如表 24-2 所示。

读取的粒度为 64 位，而 CPU 每次访问最多为 32 位。为此，提供了一个 64 比特位的缓存来加速读取速度，当启用缓存时，CPU 当前拍将下一拍的数据存放如缓存中，当下一拍取数据时，可直接从缓存中取数据，而不需要从 flash 中读取。

表 24-2：Flash 存储器读访问操作延迟

系统时钟频率	等待状态数
0 < SYSCLK ≤ 25 MHz	0
25 MHz < SYSCLK ≤ 50 MHz	1
50 MHz < SYSCLK ≤ 75 MHz	2
75 MHz < SYSCLK ≤ 100 MHz	3

24.3.2 Flash 控制器接口读操作

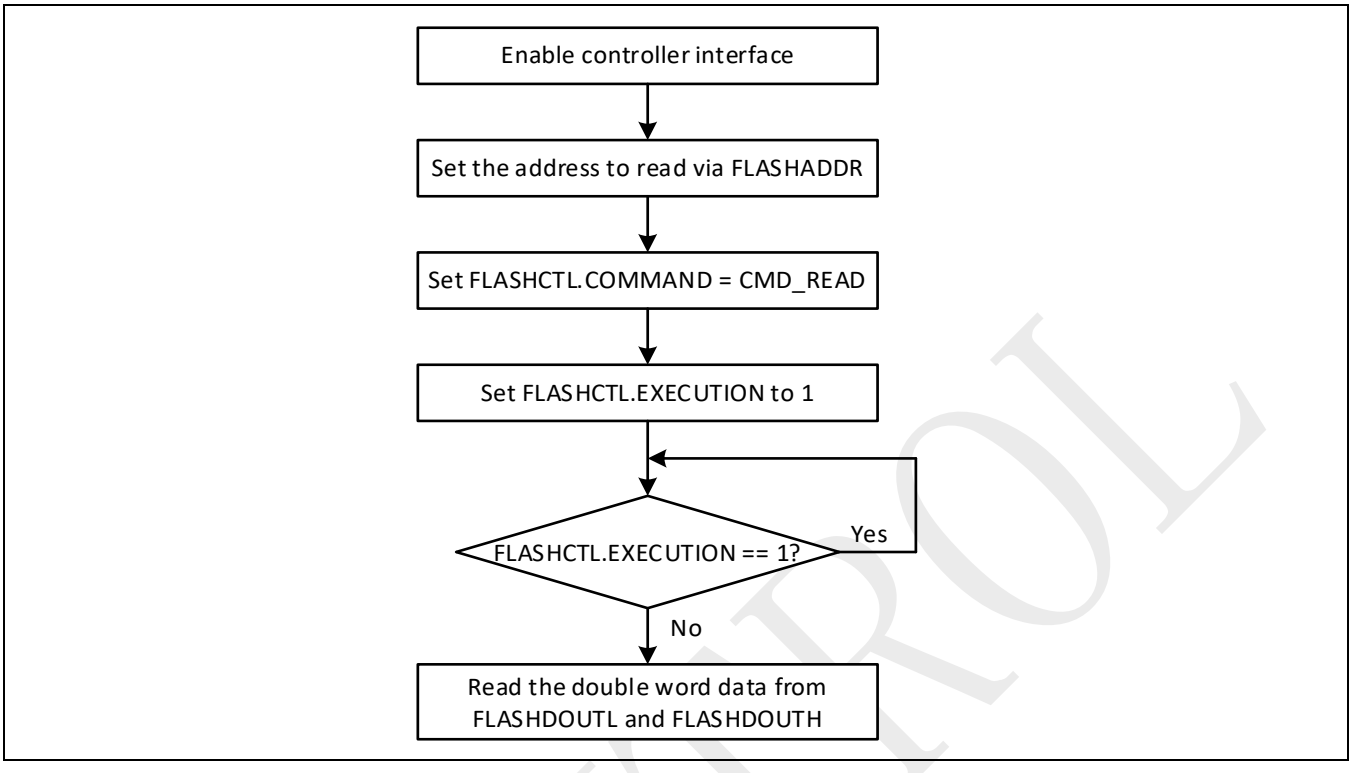
CPU 还可以使用控制器接口从 Flash 读取一个 DWord 数据，读取的顺序如下：

- 将 FLASHCREGKEY 写入 0x1ACCE551 以启用对其寄存器的写入操作；
- 将 FLASHCTL.XIPEN 写入 0 以禁用 XIP 接口并激活控制器接口；
- 使用 FLASHADDR 寄存器选择要读取的存储单元；
- 在 FLASHCTL 寄存器中设置 COMMAND 位为 CMD_READ；
- 在 FLASHCTL 寄存器中设置 EXECUTION 位；
- 等待硬件将 EXECUTION 位清除；
- 从 FLASHDOUTL 和 FLASHDOUTH 寄存器中读取双字数据；
- 将 FLASHCTL.XIPEN 写入 1 以切换回 XIP 接口；
- 将 FLASHCREGKEY 写入 0x0（任何值，除了 0x1ACCE551）以启用寄存器写保护；

用户还可以调用 pHWLIB->FLASHC_Read()函数通过控制器接口进行读取。有关详细信息，请参考 SDK 中的描述。

©2025 旋智电子科技（上海）有限公司

图 24-3：Flash 存储器读操作流程



24.4 编程和擦除操作

Flash 存储器的编程和擦除操作通过 Flash 控制器接口实现。包含 8 个 32 位寄存器：

- Flash 控制寄存器（FLASHCTL）
- Flash 地址寄存器（FLASHADDR）
- Flash 输入数据低字寄存器（FLASHDINL）
- Flash 输入数据高字寄存器（FLASHDINH）
- Flash 输出数据低字寄存器（FLASHDOUTL）
- Flash 输出数据高字寄存器（FLASHDOUTH）
- Flash 写保护寄存器（FLASHWP）
- Flash 寄存器写入允许键寄存器（FLASHCREGKEY）

24.4.1 使能 Flash 控制器

在复位后，Flash 控制器接口处于禁用状态。可以通过将 FLASHCTL.XIPEN 设置为 0 使能 Flash 控制器。然而，FLASHCTL 寄存器是写保护的，必须向 FLASHCREGKEY 寄存器写入解锁密钥值（0x1ACCE551）以使能对受保护寄存器的写访问。

©2025 旋智电子科技（上海）有限公司

24.4.2 Flash 存储器编程操作

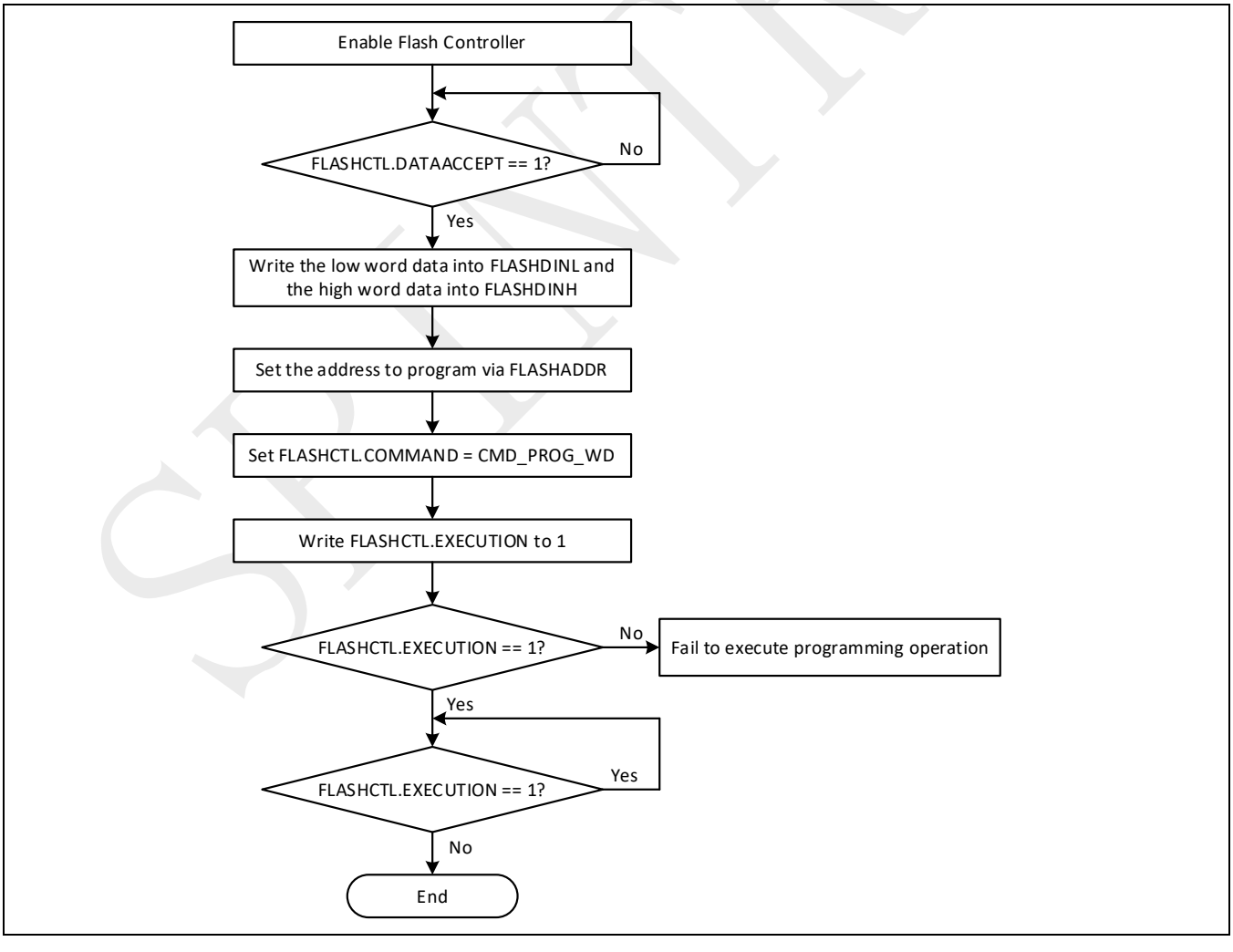
闪存编程的粒度与内存单元的大小相同，即 1 个 DWord。闪存编程的顺序如下：

- 等待 FLASHCTL 寄存器中的 DATAACCEPT 位置位；
- 将低位数据写入 FLASHDINL 寄存器，将高位数据写入 FLASHDINH 寄存器；
- 设置 FLASHADDR 寄存器以选择要编程的内存单元；
- 在 FLASHCTL 寄存器中使用 CMD_PROG_DWORD 命令位；
- 设置 FLASHCTL 寄存器中的 EXECUTION 位，并检查该位的值是否被设置；
- 等待 EXECUTION 位被清除；

如果所选择的 Flash 存储位置由 FLASHWP 寄存器写保护，则 EXECUTION 位无法被设置，编程操作将被跳过。

用户还可以通过调用 pHWLIB->FLASHC_ProgramDWord()或 pHWLIB->FLASHC_Program()函数通过控制器接口进行编程。有关详细信息，请参阅 SDK 中的描述。

图 24-4：Flash 存储器编程操作流程



SPINTROL

24.4.3 Flash 存储器擦除操作

可以使用 Flash 控制器接口按扇区或整片擦除主闪存内存，而 NVR 内存只能按扇区擦除。

扇区擦除：

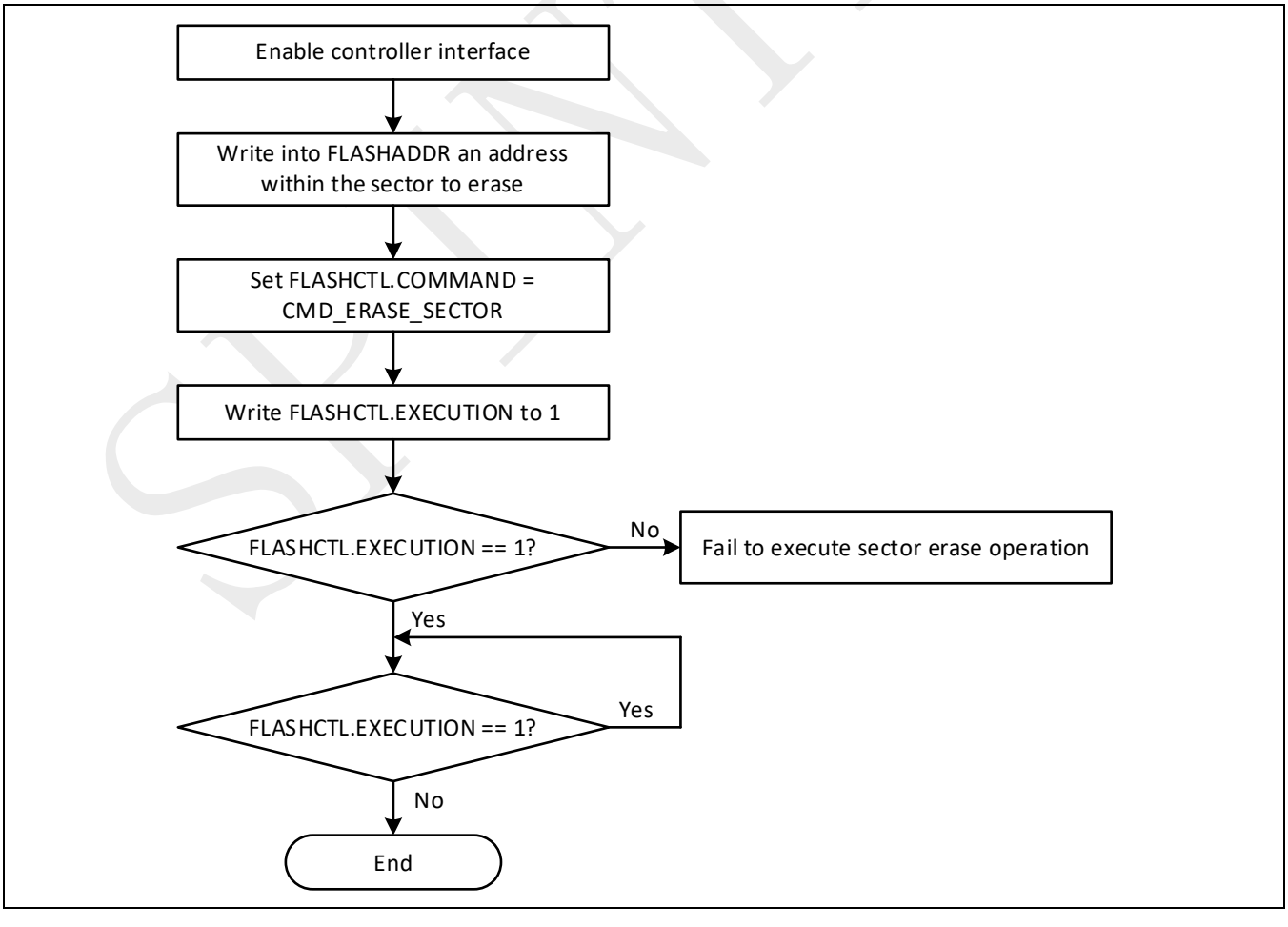
Flash 存储器的扇区擦除可以通过 Flash 控制器的擦除特性来实现，要擦除一个扇区的操作过程如下：

- 设置 FLASHADDR 寄存器以选择要擦除的扇区；
- 在 FLASHCTL 寄存器中使用 CMD_ERASE_SECTOR 命令位；
- 设置 FLASHCTL 寄存器中的 EXECUTION 位，并检查该位的值是否被设置；
- 等待 EXECUTION 位被清除；

如果所选择的主存储位置由 FLASHWP 寄存器写保护，则 EXECUTION 位无法被设置，扇区擦除操作将被跳过。

用户还可以通过调用 pHWLIB->FLASHC_EraseSector()函数通过控制器接口擦除扇区，并通过调用 pHWLIB->FLASHC_VerifyErase()函数检查擦除后内容是否全为 1。有关详细信息，请参阅 SDK 中的描述。

图 24-5：Flash 存储器扇区擦除操作流程



©2025 旋智电子科技（上海）有限公司

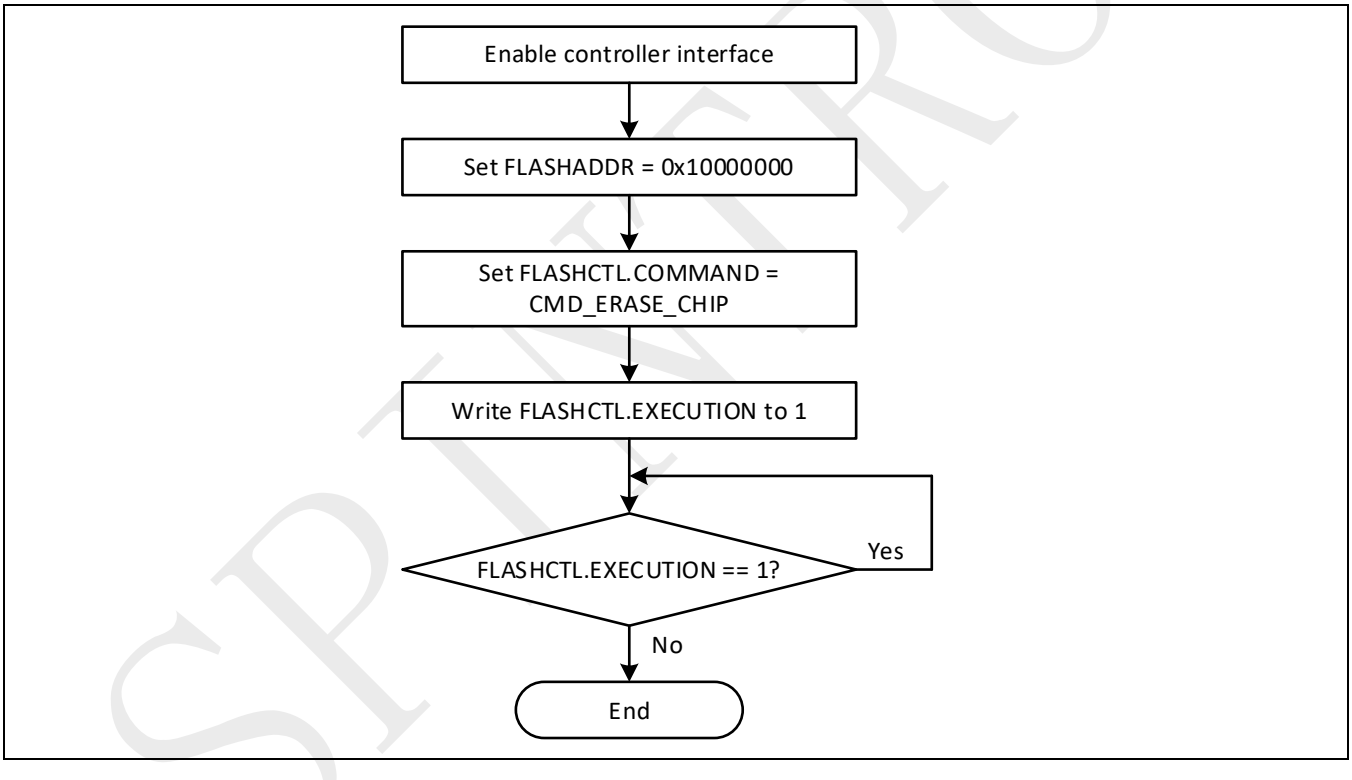
整片擦除

Flash 存储器的整片擦除可以通过 Flash 控制器的整片擦除指令来实现，要擦除整片的操作过程如下：

- 使用主闪存的起始地址设置 FLASHADDR 寄存器；
- 在 FLASHCTL 寄存器中使用 CMD_ERASE_CHIP 命令位；
- 设置 FLASHCTL 寄存器中的 EXECUTION 位；
- 等待 EXECUTION 位被重置；

用户还可以通过调用 pHWLIB->FLASHC_EraseChip()函数通过闪存控制器擦除整个芯片，并通过调用 pHWLIB->FLASHC_VerifyErase()函数检查擦除后内容是否全为 1。有关详细信息，请参阅 SDK 中的描述。

图 24-6：Flash 擦除流程图



SPINTROL

24.5 保护机制

24.5.1 分区保护

SPC1169 提供了对 Flash 存储区的读取保护功能，可以防止通过调试接口读取和修改存储区的数据，该保护在设备的所有操作模式下都有效。

要激活读取保护，需要设置配置字区域中的 CHIP_SECURITY 字段，并通过系统复位重新加载新的配置。

要禁用读取保护，可以按以下步骤操作：

- 擦除整个芯片，此时读取保护仍然是启用状态；
- 通过复位设备以重新加载配置，芯片将禁用读取保护功能；

24.5.2 写保护

对于 Flash 主存储区的扇区，可以设置写入保护，以防止由于程序计数器上下文丢失而导致的意外写入。写入保护以每次一个扇区的粒度实现，如果在受保护的扇区上执行程序或擦除操作，该操作将被跳过。

通过设置 FLASHWP 寄存器的位来激活写入保护，重置寄存器位将禁用写入保护。

24.6 Flash 时序设置

为了避免故障，Flash 访问有一系列物理时序要求，这是一系列事件的时序关系的绝对值列表。在 XIP 接口和闪存控制器接口中都设计了有限状态机（FSM），它们基于计数时钟周期和一系列超时计数来保证时序。为了方便使用，用户可以调用 pHWLIB->FLASHC_SetTiming()函数。该函数以 CPU 时钟频率作为输入参数，并自动设置所有超时计数。

每当用户计划更改 CPU 时钟频率时，应在频率更改之前调用 FLASHC_RelaxXIPTiming()函数，以放宽对闪存访问的时序要求。在频率更改后，应重新调用 pHWLIB->FLASHC_SetTiming()函数来设置优化的时序参数。

©2025 旋智电子科技（上海）有限公司

24.7 配置字（Configuration words）

系统设置有两个配置字，根据应用需求由最终用户进行编程。这些配置字位于 Flash 主存储区的末尾（最后一个双字），地址为 0x1001FFF8。

表 24-3 中显示了这些配置字的描述。在系统上电复位后，新编程的配置字将被加载并激活。

表 24-3：配置字的构成及其描述

地址	配置字名称	配置字描述
0x1001FFF8	WDT_ENABLE	看门狗使能字 0xFFFFFFFF：芯片启动时禁用看门狗 其他数值：芯片启动时启用看门狗
0x1001FFFC	CHIP_SECURITY	芯片调试接口锁定字 0xFFFFFFFF：芯片调试接口将不被锁定 其他数值：芯片调试接口将被锁定

SPINTROL

24.8 寄存器

24.8.1 Flash 寄存器列表

表 24-4：Flash 模块基地址

外设模块	基地址
FLASHC	0x4000 1B00

表 24-5：FLASHC 寄存器列表

寄存器	偏移地址	描述	复位值
FLASHCTL*	0x00	Flash 控制寄存器	0x00000010
FLASHADDR	0x08	Flash 地址寄存器	0x00000000
FLASHDINL	0x10	Flash 输入数据低 32 位寄存器	0x00000000
FLASHDINH	0x14	Flash 输入数据高 32 位寄存器	0x00000000
FLASHDINE	0x18	Flash 输入数据 ECC 字节寄存器	0x00000000
FLASHDINP	0x1C	Flash 数据输入模式寄存器	0x00000000
FLASHDOUTL	0x20	Flash 输出数据低 32 位寄存器	0x00000000
FLASHDOUTH	0x24	Flash 输出数据高 32 位寄存器	0x00000000
FLASHDOUTE	0x28	Flash 输出数据 ECC 字节寄存器	0x00000000
FLASHDOUTP	0x2C	Flash 数据输出模式寄存器	0x00000000
FLASHWP*	0x58	Flash 写保护寄存器	0x00000000
FLASHCREGKEY	0x60	Flash 控制器写使能寄存器	0x00000000

注意： 由*标记的寄存器仅当 AESREGKEY=0x1ACCE551 时才可以改写。

©2025 旋智电子科技（上海）有限公司

24.8.2 Flash 寄存器

表 24-6：Flash 控制寄存器（FLASHCTL）位段定义

FLASHCTL (Flash Control Register) Offset：0x0 Default：0x00000010 Access：FLASHC->FLASHCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							DWCACHEWE
7	6	5	4	3	2	1	0
DWCACHE	XIPEN	EXECUTION	DATAACCEPT	COMMAND			

表 24-7：Flash 控制寄存器（FLASHCTL）位段描述

位段	位段名	属性	复位值	描述
31:9	RESERVED_31_9	RO	0x0	保留
8	DWCACHEWE	W1S	0x0	DWCACHE 位写使能。 在写入 DWCACHE 位时需要同时写入 1 0：写入 0 无效，始终读回 0 1：写入 1 启用对 DWCACHE 的并发写入。该位会自动清零。
7	DWCACHE	RW	0x0	如果新地址与上次地址位于同一双字中，则启用简单缓存 0：禁用双字缓存 1：启用双字缓存
6	XIPEN	RW	0x0	XIP 模块使能 0：禁用 XIP 模块 1：启用 XIP 模块
5	EXECUTION	W1C	0x0	命令正在执行 0：写入 0 无效 读取 0 表示无命令执行 1：写入 1 启动新的命令执行周期。 读取 1 表示命令仍在执行。执行完成后将变为 0。
4	DATAACCEPT	RO	0x1	当该位为 0 时，无法更改 FLASHDIN。
3:0	COMMAND	RW	0x0	Flash 命令 0000：上电 0001：读取 0010：编程字 0011：编程页

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				0100：擦除扇区 0101：擦除芯片 0110：召回读取（仅在读取 NVR 扇区时使用） 0111：设置配置 1110：验证读取 1111：掉电

表 24-8：Flash 地址寄存器（FLASHADDR）位段定义

FLASHADDR (Flash Address Register) Offset：0x8 Default：0x00000000 Access：FLASHC->FLASHADDR							
31	30	29	28	27	26	25	24
ADDR							
23	22	21	20	19	18	17	16
ADDR							
15	14	13	12	11	10	9	8
ADDR							
7	6	5	4	3	2	1	0
ADDR							

表 24-9：Flash 地址寄存器（FLASHADDR）位段描述

位段	位段名	属性	复位值	描述
31:0	ADDR	RW	0x0	Flash 地址

表 24-10：Flash 输入数据低 32 位寄存器（FLASHDINL）位段定义

FLASHDINL (Flash Data Input Low Word Register) Offset：0x10 Default：0x00000000 Access：FLASHC->FLASHDINL							
31	30	29	28	27	26	25	24
DINL							
23	22	21	20	19	18	17	16
DINL							
15	14	13	12	11	10	9	8
DINL							
7	6	5	4	3	2	1	0
DINL							

表 24-11：Flash 输入数据低 32 位寄存器（FLASHDINL）位段描述

位段	位段名	属性	复位值	描述
31:0	DINL	RW	0x0	Flash 输入字数据的低 32 位

©2025 旋智电子科技（上海）有限公司

表 24-12：Flash 输入数据高 32 位寄存器（FLASHDINH）位段定义

FLASHDINH (Flash Data Input High Word Register) Offset：0x14 Default：0x00000000 Access：FLASHC->FLASHDINH							
31	30	29	28	27	26	25	24
DINH							
23	22	21	20	19	18	17	16
DINH							
15	14	13	12	11	10	9	8
DINH							
7	6	5	4	3	2	1	0
DINH							

表 24-13：Flash 输入数据高 32 位寄存器（FLASHDINH）位段描述

位段	位段名	属性	复位值	描述
31:0	DINH	RW	0x0	Flash 输入字数据的高 32 位

表 24-14：Flash 输入数据 ECC 字节寄存器（FLASHDINE）位段定义

FLASHDINE (Flash Data Input ECC Byte Register) Offset：0x18 Default：0x00000000 Access：FLASHC->FLASHDINE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
DINE							

表 24-15：Flash 输入数据 ECC 字节寄存器（FLASHDINE）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7:0	DINE	RW	0x0	Flash 输入的 ECC 字节数据 仅在硬件 ECC 禁用时有效。

©2025 旋智电子科技（上海）有限公司

表 24-16：Flash 数据输入模式寄存器（FLASHDINP）位段定义

FLASHDINP (Flash Data Input Pattern Register) Offset：0x1C Default：0x00000000 Access：FLASHC->FLASHDINP							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED					PATTERN		

表 24-17：Flash 数据输入模式寄存器（FLASHDINP）位段描述

位段	位段名	属性	复位值	描述
31:3	RESERVED_31_3	RO	0x0	保留
2:0	PATTERN	RW	0x0	Flash 原始输入模式（不进行 ECC 纠正） 000：由 FLASHDIN {E|H|L}设置 100：全为 0 101：全为 0x55 的棋盘格字节 110：全为 0xAA 的棋盘格字节 111：全为 1

表 24-18：Flash 输出数据低 32 位寄存器（FLASHDOUTL）位段定义

FLASHDOUTL (Flash Data Output Low Word Register) Offset：0x20 Default：0x00000000 Access：FLASHC->FLASHDOUTL							
31	30	29	28	27	26	25	24
DOUTL							
23	22	21	20	19	18	17	16
DOUTL							
15	14	13	12	11	10	9	8
DOUTL							
7	6	5	4	3	2	1	0
DOUTL							

表 24-19：Flash 输出数据低 32 位寄存器（FLASHDOUTL）位段描述

位段	位段名	属性	复位值	描述
31:0	DOUTL	RO	0x0	Flash 输出字数据的低 32 位

©2025 旋智电子科技（上海）有限公司

表 24-20：Flash 输出数据高 32 位寄存器（FLASHDOUTH）位段定义

FLASHDOUTH (Flash Data Output High Word Register) Offset：0x24 Default：0x00000000 Access：FLASHC->FLASHDOUTH							
31	30	29	28	27	26	25	24
DOUTH							
23	22	21	20	19	18	17	16
DOUTH							
15	14	13	12	11	10	9	8
DOUTH							
7	6	5	4	3	2	1	0
DOUTH							

表 24-21：Flash 输出数据高 32 位寄存器（FLASHDOUTH）位段描述

位段	位段名	属性	复位值	描述
31:0	DOUTH	RO	0x0	Flash 输出字数据的高 32 位

表 24-22：Flash 输出数据 ECC 字节寄存器（FLASHDOUTE）位段定义

FLASHDOUTE (Flash Data Output ECC Byte Register) Offset：0x28 Default：0x00000000 Access：FLASHC->FLASHDOUTE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
DOUTE							

表 24-23：Flash 输出数据 ECC 字节寄存器（FLASHDOUTE）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7:0	DOUTE	RO	0x0	Flash 输出的 ECC 字节数据

©2025 旋智电子科技（上海）有限公司

表 24-24：Flash 数据输出模式寄存器（FLASHDOUTP）位段定义

FLASHDOUTP (Flash Data Output Pattern Register) Offset：0x2C Default：0x00000000 Access：FLASHC->FLASHDOUTP							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED					PATTERN		

表 24-25：Flash 数据输出模式寄存器（FLASHDOUTP）位段描述

位段	位段名	属性	复位值	描述
31:3	RESERVED_31_3	RO	0x0	保留
2:0	PATTERN	RO	0x0	Flash 输出的原始模式，不进行 ECC 校正： 000：没有特殊模式识别 100：全部为 0 101：全部为 0x55 交错字节 110：全部为 0xAA 交错字节 111：全部为 1

表 24-26：Flash 写保护寄存器（FLASHWP）位段定义

FLASHWP (Flash Write Protect Register) Offset：0x58 Default：0x00000000 Access：FLASHC->FLASHWP							
31 SECTOR31	30 SECTOR30	29 SECTOR29	28 SECTOR28	27 SECTOR27	26 SECTOR26	25 SECTOR25	24 SECTOR24
23 SECTOR23	22 SECTOR22	21 SECTOR21	20 SECTOR20	19 SECTOR19	18 SECTOR18	17 SECTOR17	16 SECTOR16
15 SECTOR15	14 SECTOR14	13 SECTOR13	12 SECTOR12	11 SECTOR11	10 SECTOR10	9 SECTOR9	8 SECTOR8
7 SECTOR7	6 SECTOR6	5 SECTOR5	4 SECTOR4	3 SECTOR3	2 SECTOR2	1 SECTOR1	0 SECTOR0

表 24-27：Flash 写保护寄存器（FLASHWP）位段描述

位段	位段名	属性	复位值	描述
31	SECTOR31	RW	0x0	闪存扇区 31 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
30	SECTOR30	RW	0x0	闪存扇区 30 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
29	SECTOR29	RW	0x0	闪存扇区 29 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
28	SECTOR28	RW	0x0	闪存扇区 28 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
27	SECTOR27	RW	0x0	闪存扇区 27 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
26	SECTOR26	RW	0x0	闪存扇区 26 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
25	SECTOR25	RW	0x0	闪存扇区 25 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
24	SECTOR24	RW	0x0	闪存扇区 24 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
23	SECTOR23	RW	0x0	闪存扇区 23 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
22	SECTOR22	RW	0x0	闪存扇区 22 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
21	SECTOR21	RW	0x0	闪存扇区 21 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
20	SECTOR20	RW	0x0	闪存扇区 20 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
19	SECTOR19	RW	0x0	闪存扇区 19 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
18	SECTOR18	RW	0x0	闪存扇区 18 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
17	SECTOR17	RW	0x0	闪存扇区 17 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区

SPINTROL

位段	位段名	属性	复位值	描述
16	SECTOR16	RW	0x0	闪存扇区 16 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
15	SECTOR15	RW	0x0	闪存扇区 15 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
14	SECTOR14	RW	0x0	闪存扇区 14 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
13	SECTOR13	RW	0x0	闪存扇区 13 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
12	SECTOR12	RW	0x0	闪存扇区 12 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
11	SECTOR11	RW	0x0	闪存扇区 11 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
10	SECTOR10	RW	0x0	闪存扇区 10 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
9	SECTOR9	RW	0x0	闪存扇区 9 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
8	SECTOR8	RW	0x0	闪存扇区 8 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
7	SECTOR7	RW	0x0	闪存扇区 7 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
6	SECTOR6	RW	0x0	闪存扇区 6 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
5	SECTOR5	RW	0x0	闪存扇区 5 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
4	SECTOR4	RW	0x0	闪存扇区 4 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
3	SECTOR3	RW	0x0	闪存扇区 3 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
2	SECTOR2	RW	0x0	闪存扇区 2 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
1	SECTOR1	RW	0x0	闪存扇区 1 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区
0	SECTOR0	RW	0x0	闪存扇区 0 的写保护: 0：可以编程或擦除 1：无法编程或擦除该扇区

表 24-28：Flash 控制器写使能寄存器（FLASHCREGKEY）位段定义

FLASHCREGKEY (Flash Controller Register Write-Allow Key Register) Offset：0x60 Default：0x00000000 Access：FLASHC->FLASHCREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

表 24-29：Flash 控制器写使能寄存器（FLASHCREGKEY）位段描述

位段	位段名	属性	复位值	描述
31:0	KEY	RW	0x0	写入 0x1ACCE551 以解锁受保护的闪存寄存器。

SPINTROL

25 系统控制（SYSTEM）

25.1 概述

系统控制模块提供以下功能：

- 64 位产品 ID 识别号；
- 系统监控控制；
- 用于低功耗模式唤醒的 GPIO 引脚配置；
- 存储器 ECC 错误地址；
- 非屏蔽中断（NMI）事件控制；
- 复位事件控制；
- 系统配置信息；
- 启动信息；

25.2 64 位产品识别号

64 位的产品识别号在芯片的制造时确定，它提供了一个在任何设备和任何环境下都是唯一的参考号码，且不能由用户更改。它们的理想用途包括：

- 用作序列号
- 用作安全密钥，以增加闪存中代码的安全性
  - 通过在编程内部 Flash 之前，使用和结合该唯一标识符与软件加密协议来实现；
- 用于激活安全启动流程等

25.3 系统监控寄存器

这些寄存器用于启用系统电压或温度监控。

25.4 GPIO 唤醒控制寄存器

该寄存器使选择的 GPIO 引脚能够唤醒睡眠模式或停止模式中的 MCU。

25.5 存储器 ECC 错误地址寄存器

这些寄存器存储了内存 ECC 校验出错时的内存地址。

SPINTROL

©2025 旋智电子科技（上海）有限公司

25.6 NMI 事件控制寄存器

这些寄存器用于使能、标记、清除 NMI 事件。

25.7 复位事件控制寄存器

这些寄存器用于使能、标记、清除复位事件。

25.8 系统信息寄存器

该寄存器提供了闪存大小、RAM 大小、ADC 有效位数以及时钟频率选项的信息。

25.9 启动信息寄存器

该寄存器提供了 RAM 内建的自检结果状态、Flash 初始化状态、关键寄存器默认值检查状态以及芯片上电后启动次数的计数等信息，RAM 内建的自检动作仅在上电复位（POR）后执行。

SPINTROL

©2025 旋智电子科技（上海）有限公司

25.10 寄存器

25.10.1 SYSTEM 寄存器列表

表 25-1：SYSTEM 模块及地址

外设模块	基地址
SYSTEM	0x4000 0000

表 25-2：SYSTEM 寄存器列表

寄存器	偏移地址	描述	复位值
CID0	0x00	产品识别号寄存器 0	0x00000000
CID1	0x04	产品识别号寄存器 1	0x00000000
UID0	0x08	唯一设备识别号寄存器 0	0x00000000
UID1	0x0C	唯一设备识别号寄存器 1	0x00000000
RND0	0x10	随机数寄存器 0	0x00000000
RND1	0x14	随机数寄存器 1	0x00000000
REV0	0x18	版本信息寄存器 0	0x63681010
REV1	0x1C	版本信息寄存器 1	0x20200200
SYSMONCTL*	0x20	系统监测控制寄存器	0xF800FFFF
TMCUTH*	0x40	MCU 温度监测阈值寄存器	0x0FFF0000
VDD33TH*	0x44	VDD33 监测阈值寄存器	0x0FFF0000
VDD12TH*	0x48	VDD12 监测阈值寄存器	0x0FFF0000
VADCREFTH*	0x4C	ADC 参考电压监测阈值寄存器	0x0FFF0000
TMCUCODE	0x6C	MCU 温度监测码值寄存器	0x00000000
VDD33CODE	0x70	VDD33 监测码值寄存器	0x00000000
VDD12CODE	0x74	VDD12 监测码值寄存器	0x00000000
VADCREFCODE	0x78	13-bit ADC 参考电压监测码值寄存器	0x00000000
GPIOWKUPCTL*	0x7C	GPIO 唤醒控制寄存器	0x00010001
GPIONMICTL*	0x80	GPIO NMI 中断控制寄存器	0x00000001
NMRAWIF	0x84	NMI 中断原始标志寄存器	0x00000000
NMIF	0x88	NMI 中断标志寄存器	0x00000000
NMIC	0x8C	NMI 中断清除寄存器	0x00000000
NMIE*	0x90	不可屏蔽中断使能寄存器	0x00000400
MEMRAWIF	0x94	内存错误原始标志寄存器	0x00000000
MEMIF	0x98	内存错误标志寄存器	0x00000000
MEMIC	0x9C	内存错误清除寄存器	0x00000000
MEMIE*	0xA0	内存错误使能寄存器	0x00000000
RSTEVTFLG	0xA4	复位事件标志寄存器	0x00000000
RSTEVTCLR	0xA8	复位事件清除寄存器	0x00000000
RSTEVTEN*	0xAC	复位事件使能寄存器	0x00000C00
SYSINFO	0xB0	系统信息寄存器	0x000001FF

©2025 旋智电子科技（上海）有限公司

寄存器	偏移地址	描述	复位值
ROMERRADDR	0xB4	ROM 访问 ECC 错误地址寄存器	0xFFFFFFFF
FLASHERRADDR	0xB8	Flash 访问 ECC 错误地址寄存器	0xFFFFFFFF
RAM0ERRADDR	0xBC	RAM0 访问 ECC 错误地址寄存器	0xFFFFFFFF
RAM1ERRADDR	0xC0	RAM1 访问 ECC 错误地址寄存器	0xFFFFFFFF
DBGIFCTL	0xC8	调试接口控制寄存器	0x00000000
BOOTINFO	0xCC	引导信息寄存器	0x00000000
AHBMBEFLG	0xD0	AHB 多比特 ECC 错误事件标志寄存器	0x00000000
AHBMBECLR	0xD4	AHB 多比特 ECC 错误事件清除寄存器	0x00000000
AHBSBEFLG	0xD8	AHB 单比特 ECC 错误事件标志寄存器	0x00000000
AHBSBECLR	0xDC	AHB 单比特 ECC 错误事件清除寄存器	0x00000000
PERIRSTFLG	0xE0	外设复位事件标志寄存器	0x0000FFFF
PERIRSTCLR	0xE4	外设复位事件清除寄存器	0x00000000
NVREG0	0xF0	上电永久寄存器 0	0x00000000
NVREG1	0xF4	上电永久寄存器 1	0x00000000
NVREG2	0xF8	上电永久寄存器 2	0x00000000
SYSREGKEY	0xFC	SYSTEM 模块写使能寄存器	0x1ACCE551

注意： 由*标记的寄存器仅当 SYSREGKEY=0x1ACCE551 时才可以改写。

SPINTROL

25.10.2 SYSTEM 寄存器

表 25-3：产品识别号寄存器 0（CID0）位段定义

CID0 (Chip ID Register 0) Offset：0x0 Default：0x00000000 Access：SYSTEM->CID0							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 25-4：产品识别号寄存器 0（CID0）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	64 位产品识别号的低 32 位

表 25-5：产品识别号寄存器 1（CID1）位段描述

CID1 (Chip ID Register 1) Offset：0x4 Default：0x00000000 Access：SYSTEM->CID1							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 25-6：产品识别号寄存器 1（CID1）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	64 位产品识别号的高 32 位

©2025 旋智电子科技（上海）有限公司

表 25-7：唯一设备识别号寄存器 0（UID0）位段定义

UID0 (Unique ID Register 0) Offset：0x8 Default：0x00000000 Access：SYSTEM->UID0							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 25-8：唯一设备识别号寄存器 0（UID0）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	64 位唯一设备识别号的低 32 位 以“HHmmSSsn”的格式记录生产时间信息，其中：HH 为小时，mm 为分钟，SS 为秒，sn 为生产测试时产品所在的测试槽

表 25-9：唯一设备识别号寄存器 1（UID1）位段定义

UID1 (Unique ID Register 1) Offset：0xC Default：0x00000000 Access：SYSTEM->UID1							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 25-10：唯一设备识别号寄存器 1（UID1）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	64 位唯一设备识别号的高 32 位 以“YYYYMMDD”的格式记录生产时间信息，其中：YYYY 为年份，MM 为月，DD 为天

SPINTROL

表 25-11：随机数寄存器 0（RND0）位段定义

RND0 (Random Number Register 0) Offset：0x10 Default：0x00000000 Access：SYSTEM->RND0							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 25-12：随机数寄存器 0（RND0）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	64 位随机数的低 32 位

表 25-13：随机数寄存器 1（RND1）位段定义

RND1 (Random Number Register 1) Offset：0x14 Default：0x00000000 Access：SYSTEM->RND1							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 25-14：随机数寄存器 1（RND1）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x0	64 位随机数的高 32 位

©2025 旋智电子科技（上海）有限公司

表 25-15：版本信息寄存器 0（REV0）位段定义

REV0 (Revision Information Register 0) Offset：0x18 Default：0x63681010 Access：SYSTEM->REV0							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 25-16：版本信息寄存器 0（REV0）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x63681010	64 位版本信息的低 32 位

表 25-17：版本信息寄存器 1（REV1）位段定义

REV1 (Revision Information Register 1) Offset：0x1C Default：0x20200200 Access：SYSTEM->REV1							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 25-18：版本信息寄存器 1（REV1）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0x20200200	64 位版本信息的高 32 位

SPINTROL

表 25-19：系统监控寄存器（SYSMONVTL）位段定义

SYSMONCTL (System Monitor Control Register) Offset：0x20 Default：0xF800FFFF Access：SYSTEM->SYSMONCTL							
31	30	29	28	27	26	25	24
SCWIN					VBATEN	VCPEN	VMONEN
23	22	21	20	19	18	17	16
VDD5EN	VBGEN	TPMUEN	TLINEN	TMCUEN	VDD33EN	VDD12EN	VADCREFEN
15	14	13	12	11	10	9	8
MONPRD							
7	6	5	4	3	2	1	0
MONPRD							

表 25-20：系统监控寄存器（SYSMONVTL）位段描述

位段	位段名	属性	复位值	描述
31:27	SCWIN	RW	0x1F	采样和转换时间都设置为（SCWIN + 1）*4 个 HCLK 时钟周期
26	VBATEN	RW	0x0	VBAT 监测使能 0:关闭 1:使能
25	VCPEN	RW	0x0	VCP 监测使能 0:关闭 1:使能
24	VMONEN	RW	0x0	VMON 监测使能 0:关闭 1:使能
23	VDD5EN	RW	0x0	VDD5 监测使能 0:关闭 1:使能
22	VBGEN	RW	0x0	Power bandgap 监测使能 0:关闭 1:使能
21	TPMUEN	RW	0x0	Power 温度监测使能 0:关闭 1:使能
20	TLINEN	RW	0x0	LIN PHY 温度监测使能 0:关闭 1:使能
19	TMCUEN	RW	0x0	MCU 温度监测使能 0:关闭 1:使能

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
18	VDD33EN	RW	0x0	VDD33 监测使能 0:关闭 1:使能
17	VDD12EN	RW	0x0	VDD12 监测使能 0:关闭 1:使能
16	VADCREFEN	RW	0x0	13-bit ADC 参考电压监测使能 0:关闭 1:使能
15:0	MONPRD	RW	0xFFFF	系统监测间隔设置，例如，等待（MONPRD*8+1）个 HCLK 时钟周期再测量下一个项目

表 25-21：VBAT 监测阈值寄存器（VBATTH）位段定义

VBATTH (VBAT Monitor Threshold Register)　Offset：0x24　Default：0x0FFF0000
Access：SYSTEM->VBATTH

31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

表 25-22：VBAT 监测阈值寄存器（VBATTH）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:16	MAX	RW	0xFFF	监测允许的最大 VBAT 码值
15:12	RESERVED_15_12	RO	0x0	保留
11:0	MIN	RW	0x0	监测允许的最小 VBAT 码值

©2025 旋智电子科技（上海）有限公司

表 25-23：VCP 监测阈值寄存器（VCPTH）位段定义

VCPTH (VCP Monitor Threshold Register) Offset：0x28 Default：0x0FFF0000 Access：SYSTEM->VCPTH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

表 25-24：VCP 监测阈值寄存器（VCPTH）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:16	MAX	RW	0xFFF	监测允许的最大 VCAP 码值
15:12	RESERVED_15_12	RO	0x0	保留
11:0	MIN	RW	0x0	监测允许的最小 VCAP 码值

表 25-25：VMON 监测阈值寄存器（VMONTH）位段定义

VMONTH (VMON Monitor Threshold Register) Offset：0x2C Default：0x0FFF0000 Access：SYSTEM->VMONTH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

表 25-26：VMON 监测阈值寄存器（VMONTH）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:16	MAX	RW	0xFFF	监测允许的最大 VMON 码值
15:12	RESERVED_15_12	RO	0x0	保留
11:0	MIN	RW	0x0	监测允许的最小 VMON 码值

©2025 旋智电子科技（上海）有限公司

表 25-27：VDD5 监测阈值寄存器（VDD5TH）位段定义

VDD5TH (VDD5 Monitor Threshold Register) Offset：0x30 Default：0x0FFF0000 Access：SYSTEM->VDD5TH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

表 25-28：VDD5 监测阈值寄存器（VDD5TH）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:16	MAX	RW	0xFFF	监测允许的最大 VDD5 码值
15:12	RESERVED_15_12	RO	0x0	保留
11:0	MIN	RW	0x0	监测允许的最小 VDD5 码值

表 25-29：POWER Bandgap 监测阈值寄存器（VBGTH）位段定义

VBGTH (Power Bandgap Monitor Threshold Register) Offset：0x34 Default：0x0FFF0000 Access：SYSTEM->VBGTH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

表 25-30：POWER Bandgap 监测阈值寄存器（VBGTH）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:16	MAX	RW	0xFFF	监测允许的最大 Power bandgap 码值
15:12	RESERVED_15_12	RO	0x0	保留
11:0	MIN	RW	0x0	监测允许的最小 Power bandgap 码值

©2025 旋智电子科技（上海）有限公司

表 25-31：POWER 模块温度监测阈值寄存器（TPMUTH）位段定义

TPMUTH (Power Module Temperature Monitor Threshold Register) Offset：0x38 Default：0x0FFF0000 Access：SYSTEM->TPMUTH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

表 25-32：POWER 模块温度监测阈值寄存器（TPMUTH）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:16	MAX	RW	0xFFF	监测允许的最大 Power 模块温度码值
15:12	RESERVED_15_12	RO	0x0	保留
11:0	MIN	RW	0x0	监测允许的最小 Power 模块温度码值

表 25-33：LIN PHY 温度监测阈值寄存器（TLINTH）位段定义

TLINTH (LIN PHY Temperature Monitor Threshold Register) Offset：0x3C Default：0x0FFF0000 Access：SYSTEM->TLINTH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

表 25-34：LIN PHY 温度监测阈值寄存器（TLINTH）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:16	MAX	RW	0xFFF	监测允许的最大 LIN PHY 模块温度码值
15:12	RESERVED_15_12	RO	0x0	保留
11:0	MIN	RW	0x0	监测允许的最小 LIN PHY 模块温度码值

©2025 旋智电子科技（上海）有限公司

表 25-35：MCU 温度监测阈值寄存器（TMCUTH）位段定义

TMCUTH (Controller Temperature Monitor Threshold Register) Offset：0x40 Default：0x0FFF0000 Access：SYSTEM->TMCUTH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

表 25-36：MCU 温度监测阈值寄存器（TMCUTH）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:16	MAX	RW	0xFFF	监测允许的最大 MCU 模块温度码值
15:12	RESERVED_15_12	RO	0x0	保留
11:0	MIN	RW	0x0	监测允许的最小 MCU 模块温度码值

表 25-37：VDD33 监测阈值寄存器（VDD33TH）位段定义

VDD33TH (VDD33 Monitor Threshold Register) Offset：0x44 Default：0x0FFF0000 Access：SYSTEM->VDD33TH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

表 25-38：VDD33 监测阈值寄存器（VDD33TH）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:16	MAX	RW	0xFFF	监测允许的最大 VDD33 码值
15:12	RESERVED_15_12	RO	0x0	保留
11:0	MIN	RW	0x0	监测允许的最小 VDD33 码值

©2025 旋智电子科技（上海）有限公司

表 25-39：VDD12 监测阈值寄存器（VDD12TH）位段定义

VDD12TH (VDD12 Monitor Threshold Register) Offset：0x48 Default：0x0FFF0000 Access：SYSTEM->VDD12TH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

表 25-40：VDD12 监测阈值寄存器（VDD12TH）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:16	MAX	RW	0xFFF	监测允许的最大 VDD12 码值
15:12	RESERVED_15_12	RO	0x0	保留
11:0	MIN	RW	0x0	监测允许的最小 VDD12 码值

表 25-41：ADC 参考电压监测阈值寄存器（VADCREFTH）位段定义

VADCREFTH (13-bit ADC Reference Voltage Monitor Threshold Register) Offset：0x4C Default：0x0FFF0000 Access：SYSTEM->VADCREFTH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

表 25-42：ADC 参考电压监测阈值寄存器（VADCREFTH）位段描述

位段	位段名	属性	复位值	描述
31:28	RESERVED_31_28	RO	0x0	保留
27:16	MAX	RW	0xFFF	监测允许的最大 13 位 ADC 参考电压码值
15:12	RESERVED_15_12	RO	0x0	保留
11:0	MIN	RW	0x0	监测允许的最小 13 位 ADC 参考电压码值

©2025 旋智电子科技（上海）有限公司

表 25-43：VBAT 监测码值寄存器（VBATCODE）位段定义

VBATCODE (VBAT Monitor Code Register) Offset：0x50 Default：0x00000000 Access：SYSTEM->VBATCODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

表 25-44：VBAT 监测码值寄存器（VBATCODE）位段描述

位段	位段名	属性	复位值	描述
31:0	CODE	RO	0x0	监测到的 VBAT 码值

表 25-45：VCP 监测码值寄存器（VCPCODE）位段定义

VCPCODE (VCP Monitor Code Register) Offset：0x54 Default：0x00000000 Access：SYSTEM->VCPCODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

表 25-46：VCP 监测码值寄存器（VCPCODE）位段描述

位段	位段名	属性	复位值	描述
31:0	CODE	RO	0x0	监测到的 VCP 码值

©2025 旋智电子科技（上海）有限公司

表 25-47：VMON 监测码值寄存器（VCPCODE）位段定义

VMONCODE (VMON Monitor Code Register) Offset：0x58 Default：0x00000000 Access：SYSTEM->VMONCODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

表 25-48：VMON 监测码值寄存器（VCPCODE）位段描述

位段	位段名	属性	复位值	描述
31:0	CODE	RO	0x0	监测到的 VMON 码值

表 25-49：VDD5 监测码值寄存器（VDD5CODE）位段定义

VDD5CODE (VDD5 Monitor Code Register) Offset：0x5C Default：0x00000000 Access：SYSTEM->VDD5CODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

表 25-50：VDD5 监测码值寄存器（VDD5CODE）位段描述

位段	位段名	属性	复位值	描述
31:0	CODE	RO	0x0	监测到的 VDD5 码值

©2025 旋智电子科技（上海）有限公司

表 25-51：POWER Bandgap 监测码值寄存器（VBGCODE）位段定义

VBGCODE (Power Bandgap Monitor Code Register) Offset：0x60 Default：0x00000000 Access：SYSTEM->VBGCODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

表 25-52：POWER Bandgap 监测码值寄存器（VBGCODE）位段描述

位段	位段名	属性	复位值	描述
31:0	CODE	RO	0x0	监测到的 POWER Bnadgap 码值

表 25-53：POWER 模块温度监测码值寄存器（TPMUCODE）位段定义

TPMUCODE (Power Module Temperature Monitor Code Register) Offset：0x64 Default：0x00000000 Access：SYSTEM->TPMUCODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

表 25-54：POWER 模块温度监测码值寄存器（TPMUCODE）位段描述

位段	位段名	属性	复位值	描述
31:0	CODE	RO	0x0	监测到的 POWER 模块温度码值

SPINTROL

表 25-55：LIN PHY 温度监测码值寄存器（TLINCODE）位段定义

TLINCODE (LIN PHY Temperature Monitor Code Register) Offset：0x68 Default：0x00000000 Access：SYSTEM->TLINCODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

表 25-56：LIN PHY 温度监测码值寄存器（TLINCODE）位段描述

位段	位段名	属性	复位值	描述
31:0	CODE	RO	0x0	监测到的 LIN PHY 温度码值

表 25-57：MCU 温度监测码值寄存器（TMCUCODE）位段定义

TMCUCODE (Controller Temperature Monitor Code Register) Offset：0x6C Default：0x00000000 Access：SYSTEM->TMCUCODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

表 25-58：MCU 温度监测码值寄存器（TMCUCODE）位段描述

位段	位段名	属性	复位值	描述
31:0	CODE	RO	0x0	监测到的 MCU 温度码值

©2025 旋智电子科技（上海）有限公司

表 25-59：VDD33 监测码值寄存器（VDD33CODE）位段定义

VDD33CODE (VDD33 Monitor Code Register) Offset：0x70 Default：0x00000000 Access：SYSTEM->VDD33CODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

表 25-60：VDD33 监测码值寄存器（VDD33CODE）位段描述

位段	位段名	属性	复位值	描述
31:0	CODE	RO	0x0	监测到的 VDD33 码值

表 25-61：VDD12 监测码值寄存器（VDD12CODE）位段定义

VDD12CODE (VDD12 Monitor Threshold Register) Offset：0x74 Default：0x00000000 Access：SYSTEM->VDD12CODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

表 25-62：VDD12 监测码值寄存器（VDD12CODE）位段描述

位段	位段名	属性	复位值	描述
31:0	CODE	RO	0x0	监测到的 VDD12 码值

SPINTROL

表 25-63：13 位 ADC 参考电压监测码值寄存器（VADCREFCODE）位段定义

VADCREFCODE (13-bit ADC Reference Voltage Monitor Code Register) Offset：0x78 Default：0x00000000 Access：SYSTEM->VADCREFCODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

表 25-64：13 位 ADC 参考电压监测码值寄存器（VADCREFCODE）位段描述

位段	位段名	属性	复位值	描述
31:0	CODE	RO	0x0	监测到的 13 位 ADC 参考电压码值

表 25-65：GPIO 唤醒控制寄存器（GPIOWKUPCTL）位段定义

GPIOWKUPCTL (GPIO Wake-Up Control Register) Offset：0x7C Default：0x00010001 Access：SYSTEM->GPIOWKUPCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED					SLEEPWKUPWE	SLEEPWKUPIO	SLEEPWKUPPOL
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
STOPWKUPWE	STOPWKUPIO						STOPWKUPPOL

©2025 旋智电子科技（上海）有限公司

表 25-66：GPIO 唤醒控制寄存器（GPIOWKUPCTL）位段描述

位段	位段名	属性	复位值	描述
31:19	RESERVED_31_19	RO	0x0	保留
18	SLEEPWKUPWE	W1S	0x0	允许从睡眠状态唤醒的 GPIO 写使能 在写入 SLEEPWKUPIO 和 SLEEPWKUPPOL 位段时应同时将这个位段写 1 0:关闭 1:使能
17	SLEEPWKUPIO	RW	0x0	从睡眠状态唤醒所使用的 GPIO 号 0:使用 GPIO28 将 MCU 从睡眠状态唤醒 1：使用 GPIO4 将 MCU 从睡眠状态唤醒
16	SLEEPWKUPPOL	RW	0x1	从睡眠状态唤醒所使用的 GPIO 电平 0:使用低电平将 MCU 从睡眠状态唤醒 1:使用高电平将 MCU 从睡眠状态唤醒
15:8	RESERVED_15_8	RO	0x0	保留
7	STOPWKUPWE	W1S	0x0	允许从停止状态唤醒的 GPIO 写使能 在写入 STOPWKUPIO 和 STOPWKUPPOL 位段时应同时将这个位段写 1 0:关闭 1:使能
6:1	STOPWKUPIO	RW	0x0	从停止状态唤醒所使用的 GPIO 号
0	STOPWKUPPOL	RW	0x1	从停止状态唤醒所使用的 GPIO 电平 0:使用低电平将 MCU 从停止状态唤醒 1:使用高电平将 MCU 从停止状态唤醒

表 25-67：GPIO NMI 中断控制寄存器（GPIONMICTL）位段定义

GPIONMICTL (GPIO Non-Maskable Interrupt Control Register)　Offset：0x80　Default：0x00000001

Access：SYSTEM->GPIONMICTL

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	IOSEL						POL

SPINTROL

表 25-68：GPIO NMI 中断控制寄存器（GPIONMICTL）位段描述

位段	位段名	属性	复位值	描述
31:7	RESERVED_31_7	RO	0x0	保留
6:1	IOSEL	RW	0x0	触发 NMI 中断的 GPIO 号
0	POL	RW	0x1	GPIO 触发 NMI 中断的电平 0:下降沿 1:上升沿

表 25-69：NMI 中断原始标志寄存器（NMRAWIF）位段定义

NMRAWIF (Non-Maskable Interrupt Raw Flag Register)　Offset：0x84　Default：0x00000000
Access：SYSTEM->NMRAWIF

31	30	29	28	27	26	25	24
RESERVED						PTM	DEBUG
23	22	21	20	19	18	17	16
AHBMBE	VBATERR	VCPERR	VMONERR	VDD5ERR	VBGERR	TPMUERR	TLINERR
15	14	13	12	11	10	9	8
TMCUERR	VDD33ERR	VDD12ERR	VADCREFERR	GPIONMI	WDT0	RAM1MBE	RAM0MBE
7	6	5	4	3	2	1	0
FLASHMBE	ROMMBE	CLKDETERR	PLLUNLOCK	VDD33OV	VDD33UV	VDD12OV	VDD12UV

表 25-70：NMI 中断原始标志寄存器（NMRAWIF）位段描述

位段	位段名	属性	复位值	描述
31:26	RESERVED_31_26	RO	0x0	保留
25	PTM	RO	0x0	进入测试模式标志 0：事件未发生 1：事件已发生
24	DEBUG	RO	0x0	进入调试模式标志 0：事件未发生 1：事件已发生
23	AHBMBE	RO	0x0	AHB 地址或者数据 ECC 多比特错误事件标志 0：事件未发生 1：事件已发生
22	VBATERR	RO	0x0	VBAT 电压异常事件标志 0：事件未发生 1：事件已发生
21	VCPERR	RO	0x0	VCP 电压异常事件标志 0：事件未发生 1：事件已发生

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
20	VMONERR	RO	0x0	VMON 电压异常事件标志 0：事件未发生 1：事件已发生
19	VDD5ERR	RO	0x0	VDD5 电压异常事件标志 0：事件未发生 1：事件已发生
18	VBGERR	RO	0x0	POWER Bandgap 电压异常事件标志 0：事件未发生 1：事件已发生
17	TPMUERR	RO	0x0	POWER 模块温度异常事件标志 0：事件未发生 1：事件已发生
16	TLINERR	RO	0x0	LIN PHY 温度异常事件标志 0：事件未发生 1：事件已发生
15	TMCUERR	RO	0x0	MCU 温度异常事件标志 0：事件未发生 1：事件已发生
14	VDD33ERR	RO	0x0	VDD33 电压异常事件标志 0：事件未发生 1：事件已发生
13	VDD12ERR	RO	0x0	VDD12 电压异常事件标志 0：事件未发生 1：事件已发生
12	VADCREFERR	RO	0x0	13 位 ADC 参考电压异常事件标志 0：事件未发生 1：事件已发生
11	GPIONMI	RO	0x0	GPIO 触发 NMI 中断事件标志 0：事件未发生 1：事件已发生
10	WDT0	RO	0x0	WDT0 超时事件标志 0：事件未发生 1：事件已发生
9	RAM1MBE	RO	0x0	RAM1 ECC 多比特错误事件标志 0：事件未发生 1：事件已发生
8	RAM0MBE	RO	0x0	RAM0 ECC 多比特错误事件标志 0：事件未发生 1：事件已发生

SPINTROL

位段	位段名	属性	复位值	描述
7	FLASHMBE	RO	0x0	FLASH ECC 多比特错误事件标志 0：事件未发生 1：事件已发生
6	ROMMBE	RO	0x0	ROM ECC 多比特错误事件标志 0：事件未发生 1：事件已发生
5	CLKDETERR	RO	0x0	CLK 探测错误事件标志 这意味着 CLKDETCNT 超出 CLKDETCTH 的值 0：事件未发生 1：事件已发生
4	PLLUNLOCK	RO	0x0	PLL 失锁事件标志 详细原因请参考 CLKSTS[PLLFAST]和 CLKSTS[PLLSLOW] 0：事件未发生 1：事件已发生
3	VDD33OV	RO	0x0	VDD33 过压事件标志 0：事件未发生 1：事件已发生
2	VDD33UV	RO	0x0	VDD33 欠压事件标志 0：事件未发生 1：事件已发生
1	VDD12OV	RO	0x0	VDD12 过压事件标志 0：事件未发生 1：事件已发生
0	VDD12UV	RO	0x0	VDD12 欠压事件标志 0：事件未发生 1：事件已发生

表 25-71：不可屏蔽中断标志寄存器（NMIF）位段定义

NMIF (Non-Maskable Interrupt Flag Register)　Offset：0x88　Default：0x00000000
Access：SYSTEM->NMIF

31	30	29	28	27	26	25	24
RESERVED					INT	PTM	DEBUG
23	22	21	20	19	18	17	16
AHBMBE	VBATERR	VCPERR	VMONERR	VDD5ERR	VBGERR	TPMUERR	TLINERR
15	14	13	12	11	10	9	8
TMCUERR	VDD33ERR	VDD12ERR	VADCREFERR	GPIONMI	WDT0	RAM1MBE	RAM0MBE
7	6	5	4	3	2	1	0
FLASHMBE	ROMMBE	CLKDETERR	PLLUNLOCK	VDD33OV	VDD33UV	VDD12OV	VDD12UV

©2025 旋智电子科技（上海）有限公司

表 25-72：不可屏蔽中断标志寄存器（NMIF）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26	INT	RO	0x0	不可屏蔽中断全局中断标志 0：中断未发生 1：中断已产生，并已经通知 CPU 本位段被清除之前不会产生新中断
25	PTM	RO	0x0	进入测试模式中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
24	DEBUG	RO	0x0	进入调试模式中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
23	AHBMBE	RO	0x0	AHB 地址或者数据 ECC 多比特错误中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
22	VBATERR	RO	0x0	VBAT 电压异常中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
21	VCPERR	RO	0x0	VCP 电压异常中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
20	VMONERR	RO	0x0	VMON 电压异常中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
19	VDD5ERR	RO	0x0	VDD5 电压异常中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
18	VBGERR	RO	0x0	POWER Bandgap 电压异常中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
17	TPMUERR	RO	0x0	POWER 模块温度异常异常中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
16	TLINERR	RO	0x0	LIN PHY 温度异常中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
15	TMCUERR	RO	0x0	MCU 温度异常中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
14	VDD33ERR	RO	0x0	VDD33 电压异常中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
13	VDD12ERR	RO	0x0	VDD12 电压异常中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
12	VADCREFERR	RO	0x0	13 位 ADC 参考电压异常中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
11	GPIONMI	RO	0x0	GPIO 触发 NMI 中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
10	WDT0	RO	0x0	WDT0 中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
9	RAM1MBE	RO	0x0	RAM1 ECC 多比特错误中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
8	RAM0MBE	RO	0x0	RAM0 ECC 多比特错误中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
7	FLASHMBE	RO	0x0	FLASH ECC 多比特错误中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
6	ROMMBE	RO	0x0	ROM ECC 多比特错误中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
5	CLKDETERR	RO	0x0	时钟交叉检测模块报错中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
4	PLLUNLOCK	RO	0x0	PLL 失锁中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
3	VDD33OV	RO	0x0	3.3V 电源过压中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
2	VDD33UV	RO	0x0	3.3V 电源欠压中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
1	VDD12OV	RO	0x0	1.2V 电源过压中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
0	VDD12UV	RO	0x0	1.2V 电源欠压中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断

SPINTROL

表 25-73：不可屏蔽中断清除寄存器（NMIC）位段定义

NMIC (Non-Maskable Interrupt Clear Register) Offset：0x8C Default：0x00000000 Access：SYSTEM->NMIC							
31	30	29 RESERVED	28	27	26 INT	25 PTM	24 DEBUG
23 AHBMBE	22 VBATERR	21 VCPERR	20 VMONERR	19 VDD5ERR	18 VBGERR	17 TPMUERR	16 TLINERR
15 TMCUERR	14 VDD33ERR	13 VDD12ERR	12 VADCREFERR	11 GPIONMI	10 WDT0	9 RAM1MBE	8 RAM0MBE
7 FLASHMBE	6 ROMMBE	5 CLKDETER	4 PLLUNLOCK	3 VDD33OV	2 VDD33UV	1 VDD12OV	0 VDD12UV

表 25-74 不可屏蔽中断清除寄存器（NMIC）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26	INT	W1C	0x0	不可屏蔽中断全局中断清除 0：写 0 无效，读总是 0 1：写 1 清除中断标志，本位段自动清零
25	PTM	W1C	0x0	进入测试模式中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
24	DEBUG	RW	0x0	进入调试模式中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
23	AHBMBE	W1C	0x0	AHB 地址或者数据 ECC 多比特错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
22	VBATERR	W1C	0x0	VBAT 电压异常中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
21	VCPERR	W1C	0x0	VCP 电压异常中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
20	VMONERR	W1C	0x0	VMON 电压异常中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
19	VDD5ERR	W1C	0x0	VDD5 电压异常中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
18	VBGERR	W1C	0x0	POWER Bandgap 电压异常中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
17	TPMUERR	W1C	0x0	POWER 模块温度异常异常中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
16	TLINERR	W1C	0x0	LIN PHY 温度异常中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
15	TMCUERR	W1C	0x0	MCU 温度异常中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
14	VDD33ERR	W1C	0x0	VDD33 电压异常中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
13	VDD12ERR	W1C	0x0	VDD12 电压异常中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
12	VADCREFERR	W1C	0x0	13 位 ADC 参考电压异常中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
11	GPIONMI	W1C	0x0	GPIO 触发 NMI 中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

SPINTROL

位段	位段名	属性	复位值	描述
10	WDT0	W1C	0x0	WDT0 中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
9	RAM1MBE	W1C	0x0	RAM1 ECC 多比特错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
8	RAM0MBE	W1C	0x0	RAM0 ECC 多比特错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
7	FLASHMBE	W1C	0x0	FLASH ECC 多比特错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
6	ROMMBE	W1C	0x0	ROM ECC 多比特错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
5	CLKDETERR	W1C	0x0	时钟交叉检测模块报错中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
4	PLLUNLOCK	W1C	0x0	PLL 失锁中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
3	VDD33OV	W1C	0x0	3.3V 电源过压中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
2	VDD33UV	W1C	0x0	3.3V 电源欠压中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
1	VDD12OV	W1C	0x0	1.2V 电源过压中断清除

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
0	VDD12UV	W1C	0x0	1.2V 电源欠压中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

表 25-75：不可屏蔽中断使能寄存器（NMIE）位段定义

NMIE (Non-Maskable Interrupt Enable Register) Offset：0x90 Default：0x00000400 Access：SYSTEM->NMIE							
31	30	29	28	27	26	25	24
RESERVED						PTM	DEBUG
23	22	21	20	19	18	17	16
AHBMBE	VBATERR	VCPERR	VMONERR	VDD5ERR	VBGERR	TPMUERR	TLINERR
15	14	13	12	11	10	9	8
TMCUERR	VDD33ERR	VDD12ERR	VADCREFERR	GPIONMI	WDT0	RAM1MBE	RAM0MBE
7	6	5	4	3	2	1	0
FLASHMBE	ROMMBE	CLKDETERR	PLLUNLOCK	VDD33OV	VDD33UV	VDD12OV	VDD12UV

SPINTROL

表 25-76：不可屏蔽中断使能寄存器（NMIE）位段描述

位段	位段名	属性	复位值	描述
31:26	RESERVED_31_26	RO	0x0	保留
25	PTM	RW	0x0	进入生产测试模式中断使能 0：关闭 1：使能
24	DEBUG	RW	0x0	进入调试模式中断清除 0：关闭 1：使能
23	AHBMBE	RW	0x0	AHB 地址或者数据 ECC 多比特错误中断使能 0：关闭 1：使能
22	VBATERR	RW	0x0	VBAT 电压异常中断使能 0：关闭 1：使能
21	VCPERR	RW	0x0	VCP 电压异常中断使能 0：关闭 1：使能
20	VMONERR	RW	0x0	VMON 电压异常中断使能 0：关闭 1：使能
19	VDD5ERR	RW	0x0	VDD5 电压异常中断使能 0：关闭 1：使能
18	VBGERR	RW	0x0	POWER Bandgap 电压异常中断使能 0：关闭 1：使能
17	TPMUERR	RW	0x0	POWER 模块温度异常中断清除 0：关闭 1：使能
16	TLINERR	RW	0x0	LIN PHY 温度异常中断使能 0：关闭 1：使能
15	TMCUERR	RW	0x0	MCU 温度异常中断使能 0：关闭 1：使能

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
14	VDD33ERR	RW	0x0	VDD33 电压异常中断使能 0：关闭 1：使能
13	VDD12ERR	RW	0x0	VDD12 电压异常中断使能 0：关闭 1：使能
12	VADCREFERR	RW	0x0	13 位 ADC 参考电压异常中断使能 0：关闭 1：使能
11	GPIONMI	RW	0x0	GPIO 触发 NMI 中断清除 00：关闭 1：使能
10	WDT0	RW	0x1	WDT0 中断使能 0：关闭 1：使能
9	RAM1MBE	RW	0x0	RAM1 ECC 多比特错误中断使能 0：关闭 1：使能
8	RAM0MBE	RW	0x0	RAM0 ECC 多比特错误中断使能 0：关闭 1：使能
7	FLASHMBE	RW	0x0	FLASH ECC 多比特错误中断使能 0：关闭 1：使能
6	ROMMBE	RW	0x0	ROM ECC 多比特错误中断使能 0：关闭 1：使能
5	CLKDETERR	RW	0x0	时钟交叉检测模块报错中断使能 0：关闭 1：使能
4	PLLUNLOCK	RW	0x0	PLL 失锁中断使能 0：关闭 1：使能
3	VDD33OV	RW	0x0	3.3V 电源过压中断使能 0：关闭 1：使能

SPINTROL

位段	位段名	属性	复位值	描述
2	VDD33UV	RW	0x0	3.3V 电源欠压中断使能 0：关闭 1：使能
1	VDD12OV	RW	0x0	1.2V 电源过压中断使能 0：关闭 1：使能
0	VDD12UV	RW	0x0	1.2V 电源欠压中断使能 0：关闭 1：使能

表 25-77：内存 ECC 错误原始标志寄存器（MEMRAWIF）位段定义

MEMRAWIF (Memory Interrupt Raw Flag Register) Offset：0x94 Default：0x00000000 Access：SYSTEM->MEMRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				RAM1SBE	RAM0SBE	FLASHSBE	ROMSBE

表 25-78：内存 ECC 错误原始标志寄存器（MEMRAWIF）位段描述

位段	位段名	属性	复位值	描述
31:4	RESERVED_31_4	RO	0x0	保留
3	RAM1SBE	RO	0x0	RAM1 单比特错误事件标志 0：事件未发生 1：事件已发生
2	RAM0SBE	RO	0x0	RAM0 单比特错误事件标志 0：事件未发生 1：事件已发生
1	FLASHSBE	RO	0x0	FLASH 单比特错误事件标志 0：事件未发生 1：事件已发生
0	ROMSBE	RO	0x0	ROM 单比特错误事件标志 0：事件未发生 1：事件已发生

©2025 旋智电子科技（上海）有限公司

表 25-79：内存 ECC 错误标志寄存器（MEMIF）位段定义

MEMIF (Memory Interrupt Flag Register) Offset：0x98 Default：0x00000000 Access：SYSTEM->MEMIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED			INT	RAM1SBE	RAM0SBE	FLASHSBE	ROMSBE

表 25-80：内存 ECC 错误标志寄存器（MEMIF）位段描述

位段	位段名	属性	复位值	描述
31:5	RESERVED_31_5	RO	0x0	保留
4	INT	RO	0x0	全局内存错误中断标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
3	RAM1SBE	RO	0x0	RAM1 ECC 单比特错误标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
2	RAM0SBE	RO	0x0	RAM0 ECC 单比特错误标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
1	FLASHSBE	RO	0x0	FLASH ECC 单比特错误标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断
0	ROMSBE	RO	0x0	ROM ECC 单比特错误标志 0：中断未发生 1：中断已发生 本位段被清除前新事件不再触发中断

SPINTROL

表 25-81：内存 ECC 错误清除寄存器（MEMIC）位段定义

MEMIC (Memory Interrupt Clear Register) Offset：0x9C Default：0x00000000 Access：SYSTEM->MEMIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED			INT	RAM1SBE	RAM0SBE	FLASHSBE	ROMSBE

表 25-82：内存 ECC 错误清除寄存器（MEMIC）位段描述

位段	位段名	属性	复位值	描述
31:5	RESERVED_31_5	RO	0x0	保留
4	INT	W1C	0x0	全局内存错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
3	RAM1SBE	W1C	0x0	RAM1 ECC 单比特错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
2	RAM0SBE	W1C	0x0	RAM0 ECC 单比特错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
1	FLASHSBE	W1C	0x0	FLASH ECC 单比特错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
0	ROMSBE	W1C	0x0	ROM ECC 单比特错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

©2025 旋智电子科技（上海）有限公司

表 25-83：内存 ECC 错误使能寄存器（MEMIE）位段定义

MEMIE (Memory Interrupt Enable Register) Offset：0xA0 Default：0x00000000 Access：SYSTEM->MEMIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				RAM1SBE	RAM0SBE	FLASHSBE	ROMSBE

表 25-84：内存 ECC 错误使能寄存器（MEMIE）位段描述

位段	位段名	属性	复位值	描述
31:4	RESERVED_31_4	RO	0x0	保留
3	RAM1SBE	RW	0x0	RAM1 ECC 单比特错误中断使能 0：关闭 1：使能
2	RAM0SBE	RW	0x0	RAM0 ECC 单比特错误中断使能 0：关闭 1：使能
1	FLASHSBE	RW	0x0	FLASH ECC 单比特错误中断使能 0：关闭 1：使能
0	ROMSBE	RW	0x0	ROM ECC 单比特错误中断使能 0：关闭 1：使能

表 25-85：复位事件标志寄存器（RSTEVTFLG）位段定义

RSTEVTFLG (Reset Event Flag Register) Offset：0xA4 Default：0x00000000 Access：SYSTEM->RSTEVTFLG							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED		PINRST	SYSRST	WDT1RST	WDT0RST	RAM1MBE	RAM0MBE
7	6	5	4	3	2	1	0
FLASHMBE	ROMMBE	CLKDETERR	PLLUNLOCK	VDD33OV	VDD33UV	VDD12OV	VDD12UV

©2025 旋智电子科技（上海）有限公司

表 25-86：复位事件标志寄存器（RSTEVTFLG）位段描述

位段	位段名	属性	复位值	描述
31:14	RESERVED_31_14	RO	0x0	保留
13	PINRST	RO	0x0	XRSTn 引脚复位事件标志 0：事件未发生 1：事件已发送
12	SYSRST	RO	0x0	SYSRESET 复位事件标志 0：事件未发生 1：事件已发送
11	WDT1RST	RO	0x0	WDT1 复位事件标志 0：事件未发生 1：事件已发送
10	WDT0RST	RO	0x0	WDT0 复位事件标志 0：事件未发生 1：事件已发送
9	RAM1MBE	RO	0x0	RAM1 读取时 ECC 多比特错误复位事件标志 0：事件未发生 1：事件已发送
8	RAM0MBE	RO	0x0	RAM0 读取时 ECC 多比特错误复位事件标志 0：事件未发生 1：事件已发送
7	FLASHMBE	RO	0x0	Flash 读取时 ECC 多比特错误复位事件标志 0：事件未发生 1：事件已发送
6	ROMMBE	RO	0x0	ROM 读取时 ECC 多比特错误复位事件标志 0：事件未发生 1：事件已发送
5	CLKDETERR	RO	0x0	时钟交叉检测错误复位事件标志 0：事件未发生 1：事件已发送
4	PLLUNLOCK	RO	0x0	PLL 失锁复位事件标志 0：事件未发生 1：事件已发送
3	VDD33OV	RO	0x0	3.3V 电源过压复位事件标志 0：事件未发生 1：事件已发送
2	VDD33UV	RO	0x0	3.3V 电源欠压复位事件标志

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				0：事件未发生 1：事件已发送
1	VDD12OV	RO	0x0	1.2V 电源过压复位事件标志 0：事件未发生 1：事件已发送
0	VDD12UV	RO	0x0	1.2V 电源欠压复位事件标志 0：事件未发生 1：事件已发送

表 25-87：复位事件清除寄存器（RSTEVTCLR）位段定义

RSTEVTCLR (Reset Event Flag Clear Register)　Offset：0xA8　Default：0x00000000
Access：SYSTEM->RSTEVTCLR

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED		PINRST	SYSRST	WDT1RST	WDT0RST	RAM1MBE	RAM0MBE
7	6	5	4	3	2	1	0
FLASHMBE	ROMMBE	CLKDETERR	PLLUNLOCK	VDD33OV	VDD33UV	VDD12OV	VDD12UV

表 25-88：复位事件清除寄存器（RSTEVTCLR）位段描述

位段	位段名	属性	复位值	描述
31:14	RESERVED_31_14	RO	0x0	保留
13	PINRST	W1C	0x0	XRSTn 引脚复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除复位事件标志，本位段自动清零
12	SYSRST	W1C	0x0	SYSRESET 复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除复位事件标志，本位段自动清零
11	WDT1RST	W1C	0x0	WDT1 复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除复位事件标志，本位段自动清零
10	WDT0RST	W1C	0x0	WDT0 复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除复位事件标志，本位段自动清零

SPINTROL

位段	位段名	属性	复位值	描述
9	RAM1MBE	W1C	0x0	RAM1 读取时 ECC 多比特错误复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除复位事件标志，本位段自动清零
8	RAM0MBE	W1C	0x0	RAM0 读取时 ECC 多比特错误复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除复位事件标志，本位段自动清零
7	FLASHMBE	W1C	0x0	Flash 读取时 ECC 多比特错误复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除复位事件标志，本位段自动清零
6	ROMMBE	W1C	0x0	ROM 读取时 ECC 多比特错误复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除复位事件标志，本位段自动清零
5	CLKDETERR	W1C	0x0	时钟交叉检测错误复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除复位事件标志，本位段自动清零
4	PLLUNLOCK	W1C	0x0	PLL 失锁复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除复位事件标志，本位段自动清零
3	VDD33OV	W1C	0x0	3.3V 电源过压复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除复位事件标志，本位段自动清零
2	VDD33UV	W1C	0x0	3.3V 电源欠压复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除复位事件标志，本位段自动清零
1	VDD12OV	W1C	0x0	1.2V 电源过压复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除复位事件标志，本位段自动清零
0	VDD12UV	W1C	0x0	1.2V 电源欠压复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除复位事件标志，本位段自动清零

©2025 旋智电子科技（上海）有限公司

表 25-89：复位事件使能寄存器（RSTEVTEN）位段定义

RSTEVTEN (Reset Event Enable Register) Offset：0xAC Default：0x00000C00 Access：SYSTEM->RSTEVTEN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED				WDT1RST	WDT0RST	RAM1MBE	RAM0MBE
7	6	5	4	3	2	1	0
FLASHMBE	ROMMBE	CLKDETERR	PLLUNLOCK	VDD33OV	VDD33UV	VDD12OV	VDD12UV

表 25-90：复位事件使能寄存器（RSTEVTEN）位段描述

位段	位段名	属性	复位值	描述
31:12	RESERVED_31_12	RO	0x0	保留
11	WDT1RST	RW	0x1	WDT1 复位事件使能 0：关闭 1：使能
10	WDT0RST	RW	0x1	WDT0 复位事件使能 0：关闭 1：使能
9	RAM1MBE	RW	0x0	RAM1 读取时 ECC 多比特错误复位事件使能 0：关闭 1：使能
8	RAM0MBE	RW	0x0	RAM0 读取时 ECC 多比特错误复位事件使能 0：关闭 1：使能
7	FLASHMBE	RW	0x0	Flash 读取时 ECC 多比特错误复位事件使能 0：关闭 1：使能
6	ROMMBE	RW	0x0	ROM 读取时 ECC 多比特错误复位事件使能 0：关闭 1：使能
5	CLKDETERR	RW	0x0	时钟交叉检测错误复位事件使能 0：关闭 1：使能
4	PLLUNLOCK	RW	0x0	PLL 失锁复位事件使能 0：关闭 1：使能

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
3	VDD33OV	RW	0x0	3.3V 电源过压复位事件使能 0：关闭 1：使能
2	VDD33UV	RW	0x0	3.3V 电源欠压复位事件使能 0：关闭 1：使能
1	VDD12OV	RW	0x0	1.2V 电源过压复位事件使能 0：关闭 1：使能
0	VDD12UV	RW	0x0	1.2V 电源欠压复位事件使能 0：关闭 1：使能

表 25-91：系统信息寄存器（SYSINFO）位段定义

SYSINFO (System Information Register) Offset：0xB0 Default：0x000001FF Access：SYSTEM->SYSINFO							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							ADCBIT
7	6	5	4	3	2	1	0
ADCBIT	CLKOPT		RAM0SIZE		RESERVED		ROMSIZE

表 25-92：系统信息寄存器（SYSINFO）位段描述

位段	位段名	属性	复位值	描述
31:9	RESERVED_31_9	RO	0x0	保留
8:7	ADCBIT	RO	0x3	ADC 解析度 00：10 bit 01：11 bit 10：12 bit 11：13 bit
6:5	CLKOPT	RO	0x3	时钟选项 00:PLL 输出最小分频为 12，不支持晶振 01：PLL 输出最小分频为 6，不支持晶振 10：PLL 输出最小分频为 12 11：PLL 输出最小分频为 6

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
4:3	RAM0SIZE	RO	0x3	RAM0 大小 00：从 0x1FFFB000 to 0x1FFFBFFF 的 4KB 或者从 0x20003000 to 0x20003FFF 的 4KB 01：从 0x1FFFA000 to 0x1FFFBFFF 的 8KB 或者从 0x20002000 to 0x20003FFF 的 8KB 10：从 0x1FFF9000 to 0x1FFFBFFF 的 12KB 或者从 0x20001000 to 0x20003FFF 的 12KB 11：从 0x1FFF8000 to 0x1FFFBFFF 的 16KB 或者从 0x20000000 to 0x20003FFF 的 16KB
2:1	RESERVED_2_1	RO	0x3	保留
0	ROMSIZE	RO	0x1	ROM 固化 API 函数支持 0：仅支持基本的 Flash 操作和电源模式函数 1：完整的函数支持（包括三角函数等）

表 25-93：ROM 访问 ECC 错误地址寄存器（ROMERRADDR）位段定义

ROMERRADDR (ROM ECC Error Address) Offset：0xB4 Default：0xFFFFFFFF Access：SYSTEM->ROMERRADDR							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 25-94：ROM 访问 ECC 错误地址寄存器（ROMERRADDR）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0xFFFFFFFF	ROM 读取时发生 ECC 错误的地址

SPINTROL

表 25-95：Flash 访问 ECC 错误地址寄存器（FLASHERRADDR）位段定义

FLASHERRADDR (FLASH ECC Error Address) Offset：0xB8 Default：0xFFFFFFFF Access：SYSTEM->FLASHERRADDR							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 25-96：Flash 访问 ECC 错误地址寄存器（FLASHERRADDR）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0xFFFFFFFF	Flash 读取时发生 ECC 错误的地址

表 25-97：RAM0 访问 ECC 错误地址寄存器（RAM0ERRADDR）位段定义

RAM0ERRADDR (RAM0 ECC Error Address) Offset：0xBC Default：0xFFFFFFFF Access：SYSTEM->RAM0ERRADDR							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 25-98：RAM0 访问 ECC 错误地址寄存器（RAM0ERRADDR）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0xFFFFFFFF	RAM0 读取时发生 ECC 错误的地址

©2025 旋智电子科技（上海）有限公司

表 25-99：RAM1 访问 ECC 错误地址寄存器（RAM1ERRADDR）位段定义

RAM1ERRADDR (RAM1 ECC Error Address) Offset：0xC0 Default：0xFFFFFFFF Access：SYSTEM->RAM1ERRADDR							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 25-100：RAM1 访问 ECC 错误地址寄存器（RAM1ERRADDR）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RO	0xFFFFFFFF	RAM1 读取时发生 ECC 错误的地址

表 25-101：调试接口控制寄存器（DBGIFCTL）位段定义

DBGIFCTL (Debug Interface Control Register) Offset：0xC8 Default：0x00000000 Access：SYSTEM->DBGIFCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							JTAGMODE

表 25-102：调试接口控制寄存器（DBGIFCTL）位段描述

位段	位段名	属性	复位值	描述
31:1	RESERVED_31_1	RO	0x0	保留
0	JTAGMODE	RW	0x0	JTAG 或者 SWD 调试接口选择 0：SWD 模式 1：JTAG 模式

©2025 旋智电子科技（上海）有限公司

表 25-103：引导信息寄存器（BOOTINFO）位段定义

BOOTINFO (Boot Information Register) Offset：0xCC Default：0x00000000 Access：SYSTEM->BOOTINFO							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED				RESERVED	RESERVED	FLASHINITERR	RAMBISTERR
15	14	13	12	11	10	9	8
BOOTCNT							
7	6	5	4	3	2	1	0
BOOTCNT							

表 25-104：引导信息寄存器（BOOTINFO）位段描述

位段	位段名	属性	复位值	描述
31:20	RESERVED_31_20	RO	0x0	保留
19	RESERVED	RW	0x0	保留
18	RESERVED	RO	0x0	保留
17	FLASHINITERR	RW	0x0	FLASH 初始化标志 0：FLASH 初始化通过 1：FALSH 初始化错误
16	RAMBISTERR	RW	0x0	RAM 测试标志 0：RAM 测试通过 1：RAM 测试错误
15:0	BOOTCNT	RW	0x0	Boot 计数器，每当启动引导代码被执行一次，计数数值加 1

表 25-105：AHB 多比特 ECC 错误事件寄存器（AHBMBEFLG）位段定义

AHBMBEFLG (AHB Multi-Bit ECC Error Event Flag Register) Offset：0xD0 Default：0x00000000 Access：SYSTEM->AHBMBEFLG							
31	30	29	28	27	26	25	24
RESERVED					CRC	ECAP	AFE
23	22	21	20	19	18	17	16
PWM	DMAC	FLASHC	TIMER2	TIMER1	TIMER0	CAN	I2C
15	14	13	12	11	10	9	8
SPI1	SPI0	UART1	UART0	GPIO	WDT1	WDT0	GLBREG
7	6	5	4	3	2	1	0
RAM1	RAM0	XIP	ROM	DMA	SBUS	IBUS	DBUS

©2025 旋智电子科技（上海）有限公司

表 25-106：AHB 多比特 ECC 错误事件寄存器（AHBMBEFLG）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26	CRC	RO	0x0	CRC 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
25	ECAP	RO	0x0	ECAP 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
24	AFE	RO	0x0	AFE 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
23	PWM	RO	0x0	PWM 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
22	DMAC	RO	0x0	DMAC 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
21	FLASHC	RO	0x0	FLASHC 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
20	TIMER2	RO	0x0	TIMER2 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
19	TIMER1	RO	0x0	TIMER1 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
18	TIMER0	RO	0x0	TIMER0 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
17	CAN	RO	0x0	CAN 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
16	I2C	RO	0x0	I2C 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
15	SPI1	RO	0x0	SPI1 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
14	SPI0	RO	0x0	SPI0 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
13	UART1	RO	0x0	UART1 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
12	UART0	RO	0x0	UART0 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
11	GPIO	RO	0x0	GPIO 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
10	WDT1	RO	0x0	WDT1 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
9	WDT0	RO	0x0	WDT0 在 AHB 端多比特 ECC 错误事件标志 0：Event does not occur 1：Event occurred
8	GLBREG	RO	0x0	Global control 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
7	RAM1	RO	0x0	RAM1 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
6	RAM0	RO	0x0	RAM0 在 AHB 端多比特 ECC 错误事件标志 0：Event does not occur 1：Event occurred
5	XIP	RO	0x0	XIP 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
4	ROM	RO	0x0	ROM 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
3	DMA	RO	0x0	DMA 在 AHB 端多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
2	SBUS	RO	0x0	S-bus 多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
1	IBUS	RO	0x0	I-bus 多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生

SPINTROL

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
0	DBUS	RO	0x0	D-bus 多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生

表 25-107：AHB 多比特 ECC 错误事件清除寄存器（AHBMBECLR）位段定义

AHBMBECLR (AHB Multi-bit ECC Error Event Flag Register) Offset：0xD4 Default：0x00000000
Access：SYSTEM->AHBMBECLR

31	30	29	28	27	26	25	24
RESERVED					CRC	ECAP	AFE
23	22	21	20	19	18	17	16
PWM	DMAC	FLASHC	TIMER2	TIMER1	TIMER0	CAN	I2C
15	14	13	12	11	10	9	8
SPI1	SPI0	UART1	UART0	GPIO	WDT1	WDT0	GLBREG
7	6	5	4	3	2	1	0
RAM1	RAM0	XIP	ROM	DMA	SBUS	IBUS	DBUS

表 25-108：AHB 多比特 ECC 错误事件清除寄存器（AHBMBECLR）位段定义

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26	CRC	W1C	0x0	CRC 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
25	ECAP	W1C	0x0	ECAP 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
24	AFE	W1C	0x0	AFE 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
23	PWM	W1C	0x0	PWM 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
22	DMAC	W1C	0x0	DMAC 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
21	FLASHC	W1C	0x0	FLASHC 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				1：写 1 清除事件和中断标志，本位段自动清零
20	TIMER2	W1C	0x0	TIMER2 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
19	TIMER1	W1C	0x0	TIMER1 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
18	TIMER0	W1C	0x0	TIMER0 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
17	CAN	W1C	0x0	CAN 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
16	I2C	W1C	0x0	I2C 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
15	SPI1	W1C	0x0	SPI1 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
14	SPI0	W1C	0x0	SPI0 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
13	UART1	W1C	0x0	UART1 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
12	UART0	W1C	0x0	UART0 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
11	GPIO	W1C	0x0	GPIO 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
10	WDT1	W1C	0x0	WDT1 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
9	WDT0	W1C	0x0	WDT0 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
8	GLBREG	W1C	0x0	Global control 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
7	RAM1	W1C	0x0	RAM1 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
6	RAM0	W1C	0x0	RAM0 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
5	XIP	W1C	0x0	XIP 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
4	ROM	W1C	0x0	ROM 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
3	DMA	W1C	0x0	DMA 在 AHB 端多比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
2	SBUS	W1C	0x0	S-bus 无法纠正的 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
1	IBUS	W1C	0x0	I-bus 无法纠正的错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
0	DBUS	W1C	0x0	D-bus 无法纠正的错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

表 25-109：AHB 单比特 ECC 错误事件标志寄存器（AHBSBEFLG）位段定义

AHBSBEFLG (AHB Single-Bit ECC Error Event Flag Register)　Offset：0xD8　Default：0x00000000
Access：SYSTEM->AHBSBEFLG

31	30	29	28	27	26	25	24
RESERVED					CRC	ECAP	AFE
23	22	21	20	19	18	17	16
PWM	DMAC	FLASHC	TIMER2	TIMER1	TIMER0	CAN	I2C
15	14	13	12	11	10	9	8
SPI1	SPI0	UART1	UART0	GPIO	WDT1	WDT0	GLBREG
7	6	5	4	3	2	1	0
RAM1	RAM0	XIP	ROM	DMA	SBUS	IBUS	DBUS

表 25-110：AHB 单比特 ECC 错误事件标志寄存器（AHBSBEFLG）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26	CRC	RO	0x0	CRC 在 AHB 端单比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
25	ECAP	RO	0x0	ECAP 在 AHB 端单比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
24	AFE	RO	0x0	AFE 在 AHB 端单比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
23	PWM	RO	0x0	PWM 在 AHB 端单比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
22	DMAC	RO	0x0	DMAC 在 AHB 端单比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
21	FLASHC	RO	0x0	FLASHC 在 AHB 端单比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
20	TIMER2	RO	0x0	TIMER2 在 AHB 端单比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
19	TIMER1	RO	0x0	TIMER1 在 AHB 端单比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
18	TIMER0	RO	0x0	TIMER0 在 AHB 端单比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
17	CAN	RO	0x0	CAN 在 AHB 端多单特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
16	I2C	RO	0x0	I2C 在 AHB 端多单特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
15	SPI1	RO	0x0	SPI1 在 AHB 端多单特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
14	SPI0	RO	0x0	SPI0 在 AHB 端多单特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
13	UART1	RO	0x0	UART1 在 AHB 端单比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
12	UART0	RO	0x0	UART0 在 AHB 端单比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
11	GPIO	RO	0x0	GPIO 在 AHB 端单比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
10	WDT1	RO	0x0	WDT1 在 AHB 端单比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
9	WDT0	RO	0x0	WDT0 在 AHB 端单比特 ECC 错误事件标志 0： Event does not occur 1： Event occurred
8	GLBREG	RO	0x0	Global control 在 AHB 端单比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
7	RAM1	RO	0x0	RAM1 在 AHB 端单比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生

SPINTROL

位段	位段名	属性	复位值	描述
6	RAM0	RO	0x0	RAM0 在 AHB 端单比特 ECC 错误事件标志 0：Event does not occur 1：Event occurred
5	XIP	RO	0x0	XIP 在 AHB 端单比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
4	ROM	RO	0x0	ROM 在 AHB 端单比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
3	DMA	RO	0x0	DMA 在 AHB 端单比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
2	SBUS	RO	0x0	S-bus 多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
1	IBUS	RO	0x0	I-bus 多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生
0	DBUS	RO	0x0	D-bus 多比特 ECC 错误事件标志 0:事件没有发生 1:事件已经发生

SPINTROL

©2025 旋智电子科技（上海）有限公司

表 25-111：AHB 单比特 ECC 错误事件清除寄存器（AHBSBECLR）位段定义

AHBSBECLR (AHB Single-bit ECC Error Event Flag Register) Offset：0xDC Default：0x00000000 Access：SYSTEM->AHBSBECLR							
31	30	29	28	27	26	25	24
RESERVED					CRC	ECAP	AFE
23	22	21	20	19	18	17	16
PWM	DMAC	FLASHC	TIMER2	TIMER1	TIMER0	CAN	I2C
15	14	13	12	11	10	9	8
SPI1	SPI0	UART1	UART0	GPIO	WDT1	WDT0	GLBREG
7	6	5	4	3	2	1	0
RAM1	RAM0	XIP	ROM	DMA	SBUS	IBUS	DBUS

表 25-112：AHB 单比特 ECC 错误事件清除寄存器（AHBSBECLR）位段描述

位段	位段名	属性	复位值	描述
31:27	RESERVED_31_27	RO	0x0	保留
26	CRC	W1C	0x0	CRC 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
25	ECAP	W1C	0x0	ECAP 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
24	AFE	W1C	0x0	AFE 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
23	PWM	W1C	0x0	PWM 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
22	DMAC	W1C	0x0	DMAC 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
21	FLASHC	W1C	0x0	FLASHC 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
20	TIMER2	W1C	0x0	TIMER2 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				1：写 1 清除事件和中断标志，本位段自动清零
19	TIMER1	W1C	0x0	TIMER1 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
18	TIMER0	W1C	0x0	TIMER0 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
17	CAN	W1C	0x0	CAN 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
16	I2C	W1C	0x0	I2C 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
15	SPI1	W1C	0x0	SPI1 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
14	SPI0	W1C	0x0	SPI0 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
13	UART1	W1C	0x0	UART1 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
12	UART0	W1C	0x0	UART0 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
11	GPIO	W1C	0x0	GPIO 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
10	WDT1	W1C	0x0	WDT1 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
9	WDT0	W1C	0x0	WDT0 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
8	GLBREG	W1C	0x0	Global control 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
7	RAM1	W1C	0x0	RAM1 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
6	RAM0	W1C	0x0	RAM0 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
5	XIP	W1C	0x0	XIP 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
4	ROM	W1C	0x0	ROM 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
3	DMA	W1C	0x0	DMA 在 AHB 端单比特 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
2	SBUS	W1C	0x0	S-bus 无法纠正的 ECC 错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
1	IBUS	W1C	0x0	I-bus 无法纠正的错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
0	DBUS	W1C	0x0	D-bus 无法纠正的错误事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

SPINTROL

表 25-113：外设复位事件标志寄存器（PERIRSTFLG）位段定义

PERIRSTFLG (Peripheral Reset Event Flag Register) Offset：0xE0 Default：0x0000FFFF Access：SYSTEM->PERIRSTFLG							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 WDT1	14 WDT0	13 DG	12 AHB	11 CAN	10 I2C	9 SPI1	8 SPI0
7 UART1	6 UART0	5 TIMER2	4 TIMER1	3 TIMER0	2 ECAP	1 PWM	0 ADC

表 25-114：外设复位事件标志寄存器（PERIRSTFLG）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	WDT1	RO	0x1	Watchdog 1 复位事件标志 0:事件没有发生 1:事件已经发生
14	WDT0	RO	0x1	Watchdog 0 复位事件标志 0:事件没有发生 1:事件已经发生
13	DG	RO	0x1	Deglitch 模块复位事件标志 0:事件没有发生 1:事件已经发生
12	AHB	RO	0x1	Bus 和 CPU 复位事件标志 0:事件没有发生 1:事件已经发生
11	CAN	RO	0x1	CAN 复位事件标志 0:事件没有发生 1:事件已经发生
10	I2C	RO	0x1	I2C 复位事件标志 0:事件没有发生 1:事件已经发生
9	SPI1	RO	0x1	SPI1 复位事件标志 0:事件没有发生 1:事件已经发生
8	SPI0	RO	0x1	SPI0 复位事件标志 0:事件没有发生 1:事件已经发生
7	UART1	RO	0x1	UART1 复位事件标志 0:事件没有发生 1:事件已经发生

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
6	UART0	RO	0x1	UART0 复位事件标志 0:事件没有发生 1:事件已经发生
5	TIMER2	RO	0x1	TIMER2 复位事件标志 0:事件没有发生 1:事件已经发生
4	TIMER1	RO	0x1	TIMER1 复位事件标志 0:事件没有发生 1:事件已经发生
3	TIMER0	RO	0x1	TIMER0 复位事件标志 0:事件没有发生 1:事件已经发生
2	ECAP	RO	0x1	ECAP 复位事件标志 0:事件没有发生 1:事件已经发生
1	PWM	RO	0x1	PWM 复位事件标志 0:事件没有发生 1:事件已经发生
0	ADC	RO	0x1	ADC 复位事件标志 0:事件没有发生 1:事件已经发生

SPINTROL

©2025 旋智电子科技（上海）有限公司

表 25-115：外设复位事件清除寄存器（PERIRSTCLR）位段定义

PERIRSTCLR (Peripheral Reset Event Clear Register) Offset：0xE4 Default：0x00000000 Access：SYSTEM->PERIRSTCLR							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 WDT1	14 WDT0	13 DG	12 AHB	11 CAN	10 I2C	9 SPI1	8 SPI0
7 UART1	6 UART0	5 TIMER2	4 TIMER1	3 TIMER0	2 ECAP	1 PWM	0 ADC

表 25-116：外设复位事件清除寄存器（PERIRSTCLR）描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	WDT1	W1C	0x0	Watchdog 1 复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
14	WDT0	W1C	0x0	Watchdog 0 复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
13	DG	W1C	0x0	Deglitch 模块复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
12	AHB	W1C	0x0	Bus 和 CPU 复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
11	CAN	W1C	0x0	CAN 复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
10	I2C	W1C	0x0	I2C 复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
9	SPI1	W1C	0x0	SPI1 复位事件清除 0：写 0 无效，读总是 0

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				1：写 1 清除事件和中断标志，本位段自动清零
8	SPI0	W1C	0x0	SPI0 复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
7	UART1	W1C	0x0	UART1 复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
6	UART0	W1C	0x0	UART0 复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
5	TIMER2	W1C	0x0	TIMER2 复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
4	TIMER1	W1C	0x0	TIMER1 复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
3	TIMER0	W1C	0x0	TIMER0 复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
2	ECAP	W1C	0x0	ECAP 复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
1	PWM	W1C	0x0	PWM 复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
0	ADC	W1C	0x0	ADC 复位事件清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

表 25-117：上电永久寄存器 0（NVREG0）位段定义

NVREG0 (Non-Volatile Register 0) Offset：0xF0 Default：0x00000000 Access：SYSTEM->NVREG0							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 25-118：上电永久寄存器 0（NVREG0）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	本寄存器仅在上电时会复位

表 25-119：上电永久寄存器 1（NVREG1）位段定义

NVREG1 (Non-Volatile Register 1) Offset：0xF4 Default：0x00000000 Access：SYSTEM->NVREG1							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 25-120：上电永久寄存器 1（NVREG1）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	本寄存器仅在上电时会复位

表 25-121：上电永久寄存器 2（NVREG2）位段定义

NVREG2 (Non-Volatile Register 2) Offset：0xF8 Default：0x00000000 Access：SYSTEM->NVREG2							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16

©2025 旋智电子科技（上海）有限公司

VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 25-122：上电永久寄存器 2（NVREG2）位段描述

位段	位段名	属性	复位值	描述
31:0	VAL	RW	0x0	本寄存器仅在上电时会复位

SPINTROL

表 25-123：SYSTEM 模块写使能寄存器（SYSREGKEY）位段定义

SYSREGKEY (System Register Write-Allow Key Register) Offset：0xFC Default：0x1ACCE551 Access：SYSTEM->SYSREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

表 25-124：SYSTEM 模块写使能寄存器（SYSREGKEY）位段描述

位段	位段名	属性	复位值	描述
31:0	KEY	RW	0x1ACCE551	写入 0x1ACCE551 解锁受保护的 SYSTEM 寄存器

SPINTROL

©2025 旋智电子科技（上海）有限公司

26 直接存储器访问控制器（DMAC）

26.1 概述

直接存储器访问（DMA）用于提供存储器之间、外设之间以及存储器和外设之间的数据高速传输。数据通过 DMA 控制器搬移而不需要 CPU 的参与，从而可以释放 CPU 资源给其他应用。

SPC1169 中的 DMA 控制器总共有 8 个通道，用于管理来自一个或多个外设的存储器访问请求。DMA 控制器通过总线矩阵与 RAM、ADC、UART 和 SPI 相互连接，其特性如下：

26.2 DMA 特性

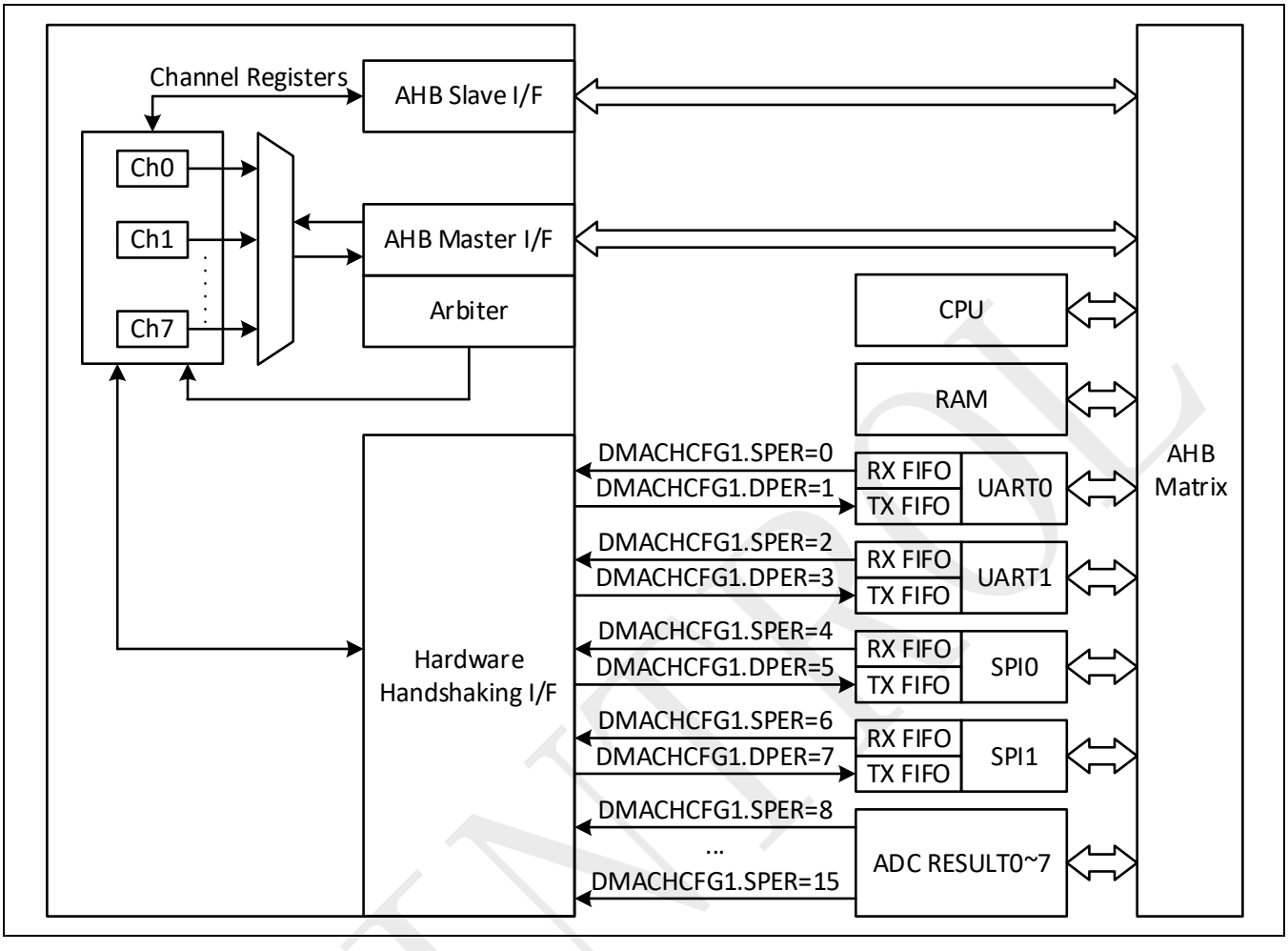
SPC1169 DMA 特性如下：

- 支持 8 个独立的通道，每个通道配备大小为 16 x 32 位的 FIFO；
- AHB 总线上的外设可以通过硬件握手接口发起 DMA 传输请求；
  - 支持硬件握手的外设包括：ADC、UART、SPI；
- 通道优先级软件可配；
- 支持存储器之间、存储器到外设或者外设到存储器、外设之间的传输；
- 支持单块传输，块大小可配，最大为 4095 字节；
- 支持源和目标突发传输，大小可配；
- 源和目的地址步进大小可配；
  - 增加到下一个传输位宽对齐的地址；
  - 减小到上一个传输位宽对齐的地址；
  - 保持不变；
- 支持 2 个中断源；
  - 传输完成触发中断；
  - 传输错误触发中断。

图 26-1 给出了 DMA 控制器框图。

SPINTROL

图 26-1：DMA 控制器框图



26.3 DMA 工作过程

26.3.1 基本定义

本章节涉及到的 DMA 相关概念定义和术语如下：

- **源**：指 DMA 从其读取数据的设备，DMA 控制器从源设备读取数据并存入通道 FIFO。
- **目的地**：指 DMA 将写入数据（之前从源设备读取并存放于 FIFO 中的数据）的设备。
- **通道**：连接源设备和目的设备之间的读/写数据路径。
- **块**：DMA 数据块，其大小由块大小确定。在 DMA 与内存之间的传输中，块被直接分解成突发传输和单个传输的序列。在 DMA 与外设之间的传输中，块被分解成单个和突发 DMA 事务的序列。
- **事务**：DMA 传输的基本单元。仅当设备为非内存设备时，事务才与 DMA 和源或目的设备之间的传输相关。有两种类型的事务：
  - 单个事务：单个事务的长度始终为 1，并转换为单个 AHB 传输；
  - 突发事务：突发事务的长度可配置。突发事务被转换为一系列突发和 AHB 单个传输。

©2025 旋智电子科技（上海）有限公司

26.3.2 硬件握手接口

DMA 硬件握手接口用于在事务级别上控制单个或突发事务的流动。当 DMA 通道和外设同时启用时，硬件握手接口可以自动完成数据传输。在 SPC1169 中，DMA 硬件握手接口分配给 UART0、UART1、SPI0、SPI1 和 ADC 模块，如表 26-1 所示。

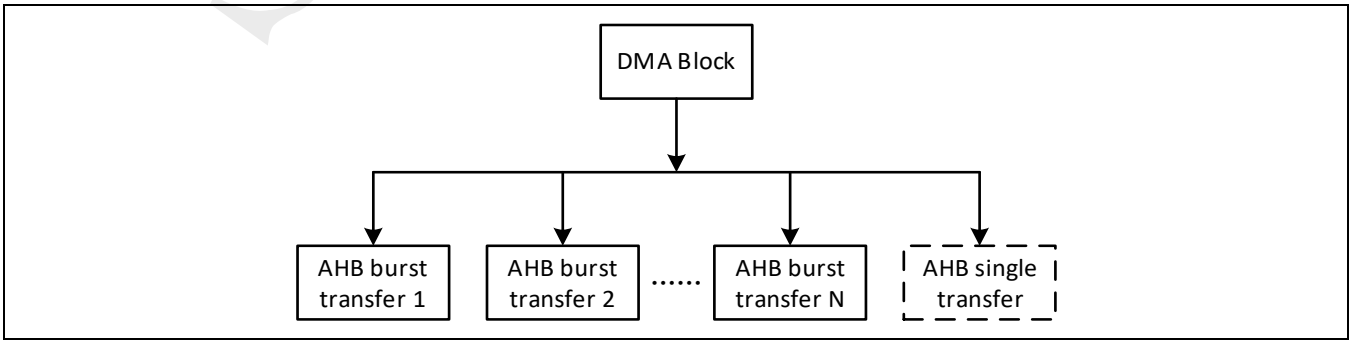
表 26-1：DMAC 硬件握手接口分配

DMACHx->DMACHCFG1.DPER/SPER	硬件握手接口
0	UART0_RX
1	UART0_TX
2	UART1_RX
3	UART1_TX
4	SPI0_RX
5	SPI0_TX
6	SPI1_RX
7	SPI1_TX
8	ADC_RESULT0
9	ADC_RESULT1
10	ADC_RESULT2
11	ADC_RESULT3
12	ADC_RESULT4
13	ADC_RESULT5
14	ADC_RESULT6
15	ADC_RESULT7

26.3.3 Memory-to-memory transfer operation

DMA 仅支持单个块传输，块的数量可以通过编程 DMACHCTL1.BLKTS 进行配置。对于 DMA 和内存之间的传输，块数据传输会直接分解为一系列的突发传输和单个传输，如图 26-2 所示。当编程到 DMA 通道中的块大小不是突发传输长度的倍数时，如图 26-2 所示，需要一系列的突发传输后跟单个传输来完成块传输。

图 26-2：DMA 内存传输的分解



©2025 旋智电子科技（上海）有限公司

示例：DMA 在两个不同内存区域之间的传输

请参照下面的步骤使用 DMA 通道 0 将 12 字节数据从 0x1FFFE000 传输到 0x1FFFF000：

- 通过将 DMAEN 设置为 1 来启用 DMA 模块；
- 配置通道 0
  - 将源地址 DMACHSA 设置为 0x1FFFE000；
  - 将目的地址 DMACHDA 设置为 0x1FFFF000；
  - 通过将 DMACHCTL0.SINC 设置为 0 来设置源地址为递增步进；
  - 通过将 DMACHCTL0.DINC 设置为 0 来设置目标地址为递增步进；
  - 通过将 DMACHCTL0.TT 设置为 0 将传输类型设置为内存到内存；
  - 通过将 DMACHCTL0.IE 设置为 1 来启用通道中断；
  - 通过将 DMACHCTL0.SWIDTH 和 DMACHCTL0.DWIDTH 设置为 0 将传输位段设置为字节；
  - 通过将 DMACHCTL1.BLKTS 设置为 12 来设置总块传输大小；
- 通过将 0x101 写入 DMACHIE 启用通道 0 传输完成中断；
- 通过将 0x101 写入 DMACHEN 启动传输。

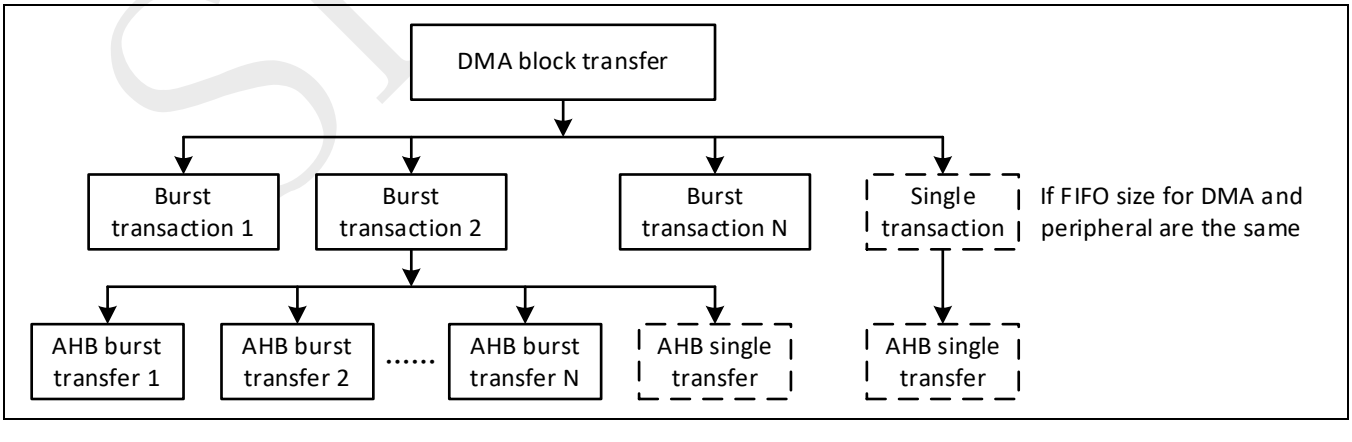
26.3.4 内存到外设和外设到内存传输

内存到外设和外设到内存是非内存传输。如图 26-3 所示，对于 DMA 和外设之间的传输，块被分成多个事务（单次或突发），每个事务由外设发起请求。然后，每个事务都转换为 AHB 传输（单个或突发）到非内存外设。

DMA 控制器的突发事务长度由源和目的地的 DMACHCTL0.SSIZE / DSIZE 定义。它是每个 DMA 请求要传输的数据项（例如 FIFO 条目）的数量。应考虑 DMA 和外设的 FIFO 大小来确定突发事务长度，以避免 FIFO 上溢或下溢。

在 SPC1169 中，DMA 传输基于内存到外设操作完成 UART 传输和 SPI 传输。对于 UART 接收、SPI 接收和 ADC 结果读取，DMA 传输基于外设到内存操作完成。

图 26-3：DMA 非内存传输的分解



©2025 旋智电子科技（上海）有限公司

以下是在内存和 UART0/SPI0 之间使用 DMA 进行 16 个字数据传输的步骤：

- 将 DMA 模块使能，设置 DMAEN 为 1；
- 配置 UART0 以支持 DMA 传输；
  - 通过 UARTCTL 寄存器使能传输、DMA 接口和 32 位总线；
  - 通过设置 UARTRXTH 为 8 将 RX FIFO 阈值设置为 8 个字；
  - 通过设置 UARTTXTH 为 0 将 TX FIFO 阈值设置为 0；
- 配置通道 0 用于 UART0 数据接收；
  - 将源地址 DMACHSA 设置为 UART0 FIFO 接口的地址（即 UARTDAT 寄存器）；
  - 将目标地址 DMACHDA 设置为内存区域 0 的起始地址；
  - 通过设置 DMACHCTL0.SINC 为 0，将源地址设置为冻结；
  - 通过设置 DMACHCTL0.DINC 为 0，将目标地址设置为递增；
  - 通过设置 DMACHCTL0.TT 为 0，将传输类型设置为从外设到内存；
  - 通过设置 DMACHCTL0.IE 为 1，使能通道中断；
  - 通过将 DMACHCTL0.SWIDTH 和 DMACHCTL0.DWIDTH 设置为 2，将传输设置为字节；
  - 通过设置 DMACHCTL1.BLKTS 为 16，设置整体块传输大小为 16；
  - 通过设置 DMACHCFG0.SSWHS 为 0，启用硬件握手；
  - 通过设置 DMACHCFG0.SHSPOL 为 1，将硬件握手极性设置为高电平活动；
  - 通过设置 DMACHCFG1.SPER 为 0，将硬件握手接口设置为 UART0 RX；
- 配置通道 1 用于 UART0 数据发送；
  - 将源地址 DMACHSA 设置为内存区域 1 的起始地址；
  - 将目标地址 DMACHDA 设置为 UART0 FIFO 接口的地址（即 UARTDAT 寄存器）；
  - 通过设置 DMACHCTL0.SINC 为 0，将源地址设置为递增；
  - 通过设置 DMACHCTL0.DINC 为 0，将目标地址设置为冻结；
  - 通过设置 DMACHCTL0.TT 为 1，将传输类型设置为从内存到外设；
  - 通过设置 DMACHCTL0.IE 为 1，使能通道中断；
  - 通过将 DMACHCTL0.SWIDTH 和 DMACHCTL0.DWIDTH 设置为 2，将传输设置为字节；
  - 通过设置 DMACHCTL1.BLKTS 为 16，设置整体块传输大小为 16；
  - 通过设置 DMACHCFG0.DSWHS 为 0，启用硬件握手；
  - 通过设置 DMACHCFG0.DHSPOL 为 1，将硬件握手极性设置为高电平活动；
  - 通过设置 DMACHCFG1.DPER 为 0，将硬件握手接口设置为 UART0 TX；
- 配置 SPI0 以支持 DMA 传输
  - 通过设置 SPICTL0.DMATXREQ 和 SPICTL0.DMARXREQ 为 1，使能 DMA 接口；

SPINTROL

- 通过设置 SPIRFTH 为 8，将 RX FIFO 阈值设置为 8 个字；
- 通过设置 SPITFTH 为 8，将 TX FIFO 阈值设置为 8 个字；

- 配置通道 2 用于 SPI0 数据接收
  - 将源地址 DMACHSA 设置为 SPI0 FIFO 接口的地址（即 SPIDAT 寄存器）；
  - 将目标地址 DMACHDA 设置为内存区域 2 的起始地址；
  - 通过设置 DMACHCTL0.SINC 为 0，将源地址设置为冻结；
  - 通过设置 DMACHCTL0.DINC 为 0，将目标地址设置为递增；
  - 通过设置 DMACHCTL0.TT 为 0，将传输类型设置为从外设到内存；
  - 通过设置 DMACHCTL0.IE 为 1，使能通道中断；
  - 通过将 DMACHCTL0.SWIDTH 和 DMACHCTL0.DWIDTH 设置为 2，将传输设置为字节；
  - 通过设置 DMACHCTL1.BLKTS 为 16，设置整体块传输大小为 16；
  - 通过设置 DMACHCFG0.SSWHS 为 0，启用硬件握手；
  - 通过设置 DMACHCFG0.SHSPOL 为 1，将硬件握手极性设置为高电平活动；
  - 通过设置 DMACHCFG1.SPER 为 4，将硬件握手接口设置为 SPI0 RX；
- 配置通道 3 用于 SPI0 数据发送
  - 将源地址 DMACHSA 设置为内存区域 3 的起始地址；
  - 将目标地址 DMACHDA 设置为 SPI0 FIFO 接口的地址（即 SPIDAT 寄存器）；
  - 通过设置 DMACHCTL0.SINC 为 0，将源地址设置为递增；
  - 通过设置 DMACHCTL0.DINC 为 0，将目标地址设置为冻结；；
  - 通过设置 DMACHCTL0.TT 为 1，将传输类型设置为从内存到外设
  - 通过设置 DMACHCTL0.IE 为 1，使能通道中断；
  - 通过将 DMACHCTL0.SWIDTH 和 DMACHCTL0.DWIDTH 设置为 2，将传输设置为字节；
  - 通过设置 DMACHCTL1.BLKTS 为 16，设置整体块传输大小为 16；
  - 通过设置 DMACHCFG0.DSWHS 为 0，启用硬件握手；
  - 通过设置 DMACHCFG0.DHSPOL 为 1，将硬件握手极性设置为高电平活动；
  - 通过设置 DMACHCFG1.DPER 为 5，将硬件握手接口设置为 SPI0 TX；
- 通过将 0xF0F 写入 DMACHIE，使能通道 0-3 的传输完成中断；
- 将 DMA 模块使能，设置 DMAEN 为 1；
- 通过将 0xF0F 写入 DMACHEN，启动传输；

©2025 旋智电子科技（上海）有限公司

26.3.5 外设到外设的传输

SPC1169 中的外设到外设传输提供了 UART0/1 和 SPI0/1 之间的物理通信通道，例如从 UART0_RX 到 UART1_TX，从 UART0_RX 到 SPI0_TX，从 SPI0_RX 到 UART1_TX 等。

示例：从 SPI0 接收到 UART0 传输的 DMA 传输

请参考以下步骤，使用 DMA 将 16 个字数据从 SPI0 接收传输到 UART0。

- 将 DMA 模块使能，将 DMAEN 设置为 1；
- 配置 UART0 以支持 DMA 传输的发送；
  - 通过 UARTCTL 寄存器启用传输、DMA 接口和 32 位总线；
  - 通过设置 UARTTXTH 为 0，将 TX FIFO 阈值设置为 0；
- 配置 SPI0 以支持 DMA 传输的接收；
  - 通过设置 SPICTL0.DMARXREQ 为 1，启用 DMA 接口；
  - 通过设置 SPIRFTH 为 8，将 RX FIFO 阈值设置为 8 个字；
- 配置通道 0，将从 SPI0 接收到的数据传输到 UART0 进行发送；
  - 将源地址 DMACHSA 设置为 SPI0 FIFO 接口的地址（即 SPIDAT 寄存器的地址）；
  - 将目标地址 DMACHDA 设置为 UART0 FIFO 接口的地址（即 UARTDAT 寄存器的地址）；
  - 通过将 DMACHCTL0.SINC 设置为 2，将源地址设置为冻结；
  - 通过将 DMACHCTL0.DINC 设置为 2，将目标地址设置为冻结；
  - 通过将 DMACHCTL0.TT 设置为 3，将传输类型设置为从外设到外设；
  - 通过将 DMACHCTL0.IE 设置为 1，启用通道中断；
  - 通过将 DMACHCTL0.SWIDTH 和 DMACHCTL0.DWIDTH 设置为 2，将传输设置为字传输；
  - 通过将 DMACHCTL1.BLKTS 设置为 16，设置整体块传输大小；
  - 通过将 DMACHCFG0.SSWHS 和 DMACHCFG0.DSWHS 设置为 0，启用硬件握手；
  - 通过将 DMACHCFG0.SHSPOL 和 DMACHCFG0.DHSPOL 设置为 1，设置硬件握手极性为高电平有效；
  - 通过将 DMACHCFG1.SPER 设置为 4，将源硬件握手接口设置为 SPI0 RX；
  - 通过将 DMACHCFG1.DPER 设置为 1，将目标硬件握手接口设置为 UART0 TX；
- 通过将 0x101 写入 DMACHIE，使能通道 0 传输完成中断；
- 将 DMA 模块使能，将 DMAEN 设置为 1；
- 通过将 0x101 写入 DMACHEN，启动传输；

SPINTROL

26.4 AHB 传输错误处理

发生 AHB 传输错误（例如访问了不可读写的地址，使得 AHB 从设备返回 ERROR 响应）时，会发生以下操作：

- 正在进行的 DMA 传输会立即停止；
- 相关通道被禁用；
- 如果启用了中断，将会发出一个中断请求。

如果多个通道被启用，仅有检测到AHB错误的通道会被禁用。FIFO的内容将变得不可访问，并在重新启用通道以开始新的传输时被覆盖。

硬件上没有自动从错误发生点恢复传输的支持，用户必须重新初始化完整的块传输才能成功完成数据传输。DMA 控制器不使用硬件握手接口来指示错误发生或传输结束。因此，如果外设的请求在发生错误时处于活动状态，则 DMA 不会自动将其停用。同理，硬件握手接口在出现错误中断时也需要由软件重置或重新初始化。换言之，在重新启用通道之前，用户需要通过关闭并重启外设 DMA 接口等方法让 DMA 硬件握手请求信号变低。

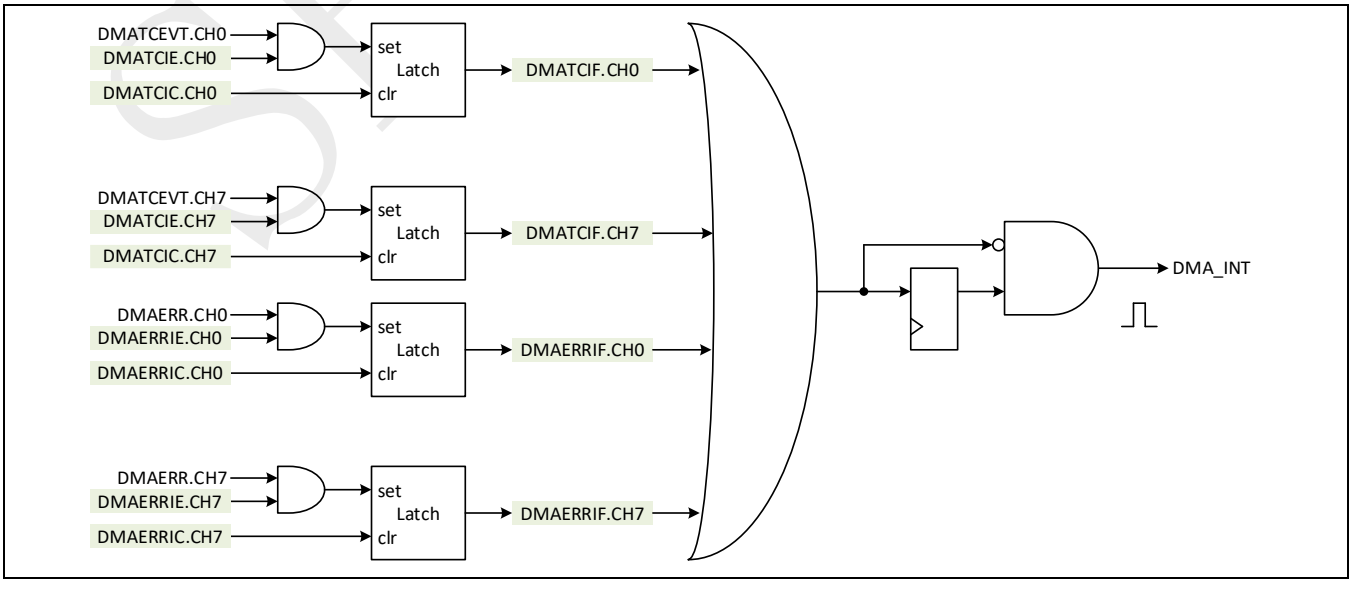
26.5 中断处理

当通道 x 的 DMA 传输完成时，事件会锁存到 DMATCRAWIF.CHx。如果通过将 DMATCIE.CHx 设置为 1 启用，可以生成中断请求并锁存到 DMATCIF.CHx。写入 1 到 DMATCIC 会清除 DMATCRAWIF.CHx 和 DMATCIF.CHx。

在 DMA 通道 x 传输期间，如果从 AHB 从设备收到 ERROR 响应时，DMA 传输将被取消并禁用该通道。该事件会锁存到 DMAERRRAWIF.CHx。如果通过将 DMAERRIE.CHx 设置为 1 启用，可以生成中断请求并锁存到 DMAERRIF.CHx。写入 1 到 DMAERRIC 会清除 DMAERRRAWIF.CHx 和 DMAERRIF.CHx。

如图 26-4 所示，中断标志位通过或运算后，合并成一个中断信号送给 CPU。

图 26-4：DMA 中断框图



©2025 旋智电子科技（上海）有限公司

根据图 26-4 所示，中断标志位被组合在一起以生成向 CPU 发送的最终单个中断。可以看出，中断标志位会阻止中断发送到 CPU，直到它们全部清除。

根据该结构，建议将中断服务例程（ISR）实现为图 26-5 所示的伪代码形式。

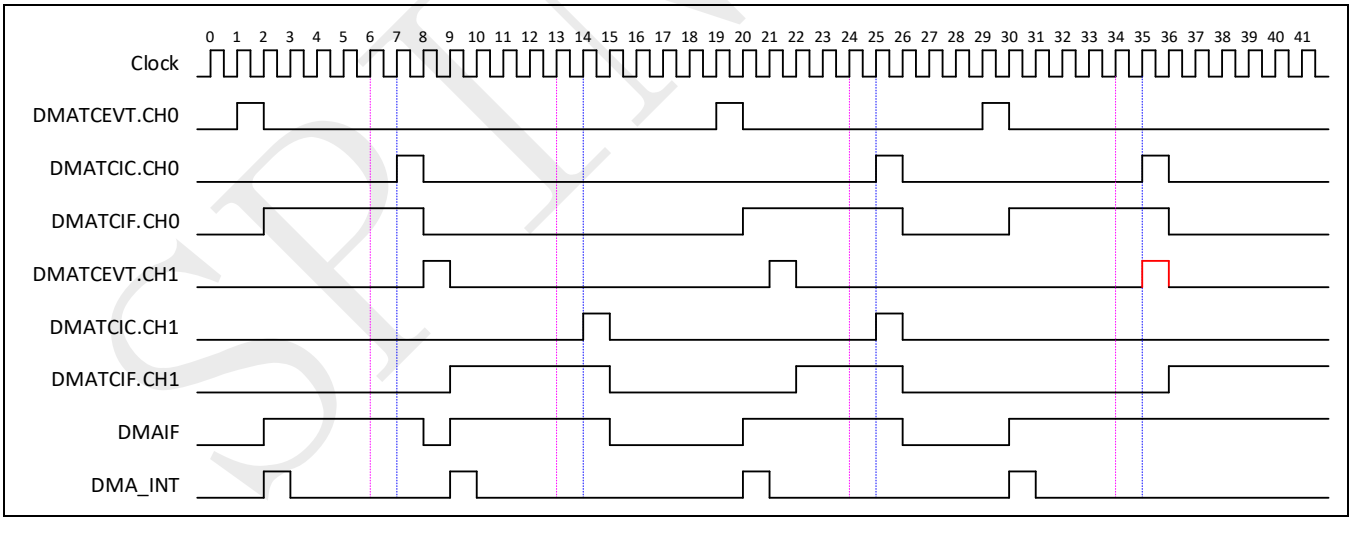
图 26-5：DMA 中断服务程序伪代码

```
u32TCFlag = DMAC->DMATCIF;
DMAC->DMATCIC = u32TCFlag; // Make sure it will not be interrupt here by ISR with higher priority
u32ErrFlag = DMAC->DMAERRIF;
DMAC->DMAERRIC = u32ErrFlag; // Make sure it will not be interrupt here by ISR with higher priority
if ((u32TCFlag & 1U) !=0) { // Code to handle channel 0 transfer completion }
if ((u32TCFlag & 2U) !=0) { // Code to handle channel 1 transfer completion }
…
if ((u32ErrFlag & 1U) !=0) { // Code to handle channel 0 transfer error }
if ((u32ErrFlag & 2U) !=0) { // Code to handle channel 1 transfer error }
```

26.5.1 中断丢失和阻塞的风险缓解

图 26-6 说明了 DMA 多通道中断事件的一种场景。

图 26-6：DMA 多通道中断事件



- ISR 的进入由每个通道的请求独立触发，因此，不存在中断丢失或重复服务的情况；
  - 在 t1 时，通道 0 的传输完成并触发中断请求，相应地设置 DMATCIF.CH0；
  - 在 ISR 中的 t6 时，软件读取 DMATCIF，并发现只有 DMATCIF.CH0 被设置。然后在 t7 时清除该标志以启用稍后到来的新中断。之后，它会处理来自通道 0 的请求；
  - 在 t8 时，通道 1 的传输完成并触发中断请求。相应地设置 DMATCIF.CH1。

©2025 旋智电子科技（上海）有限公司

- 由于 DMATCIF.CH0 和 DMATCIF.CH1 之间没有重叠，可以在组合的 DMAIF 信号中识别到上升沿，并向 CPU 发出新的中断脉冲。因此，在软件退出 ISR 时，它将再次进入 ISR；
- 在 ISR 中的 t13 时，软件读取 DMATCIF，并发现只有 DMATCIF.CH1 被设置。然后在 t14 时清除该标志以启用稍后到来的新中断。在处理完来自通道 1 的请求后，它退出 ISR；

– ISR 的进入由一个通道的请求触发，但在 ISR 内部处理多个请求。因此，不存在中断丢失或重复服务的情况；

- 在 t19 时，通道 0 的传输完成并触发中断请求。相应地设置 DMATCIF.CH0；
- 在 ISR 中的 t24 时，软件读取 DMATCIF 之前，通道 1 的传输完成于 t21。因此，软件将在 t24 时发现 DMATCIF.CH0 和 DMATCIF.CH1 都被设置，并在 t25 时清除它们以启用稍后到来的新中断。在处理完来自通道 0 和通道 1 的请求后，退出 ISR；

– 在 ISR 内的中断标志读取之后但中断清除之前出现新的中断请求，导致中断丢失并阻塞后续的中断；

- 在 t29 时，通道 0 的传输完成并触发 DMA 中断。相应地设置 DMATCIF.CH0；
- 在 ISR 中的 t34 时，软件读取 DMATCIF，并发现只有 DMATCIF.CH0 被设置。然后在 t35 时清除该标志以启用稍后到来的新中断，并开始处理通道 0；
- 在 t35 时，通道 1 的传输完成并触发中断请求。相应地设置 DMATCIF.CH1；
- 由于 DMATCIF.CH0 在 DMATCIF.CH1 被设置的同时被清除，组合的 DMAIF 信号中将不会出现上升沿。因此，不会向 CPU 发出新的中断脉冲；
- 由于 CPU 对此情况无感知，所以没有清除 DMATCIF.CH1 的机会；

根据上述分析，在标志位读取和标志位清除之间出现新的中断请求时会导致问题。为了最小化风险，这个时间窗口应尽可能小。图 26-5 所示的代码将窗口缩小到仅为 1 个时钟周期，但仍存在潜在风险，如图 26-6 所示。为了完全解决这个问题，用户需要定期检查 DMA 中断标志位（例如，在定时器的 ISR 中）并采取相应措施。

©2025 旋智电子科技（上海）有限公司

26.6 寄存器

26.6.1 DMACH 寄存器列表

表 26-2：DMACH 模块基地址

外设模块	基地址
DMACH0	0x4000 1C00
DMACH1	0x4000 1C58
DMACH2	0x4000 1CB0
DMACH3	0x4000 1D08
DMACH4	0x4000 1D60
DMACH5	0x4000 1DB8
DMACH6	0x4000 1E10
DMACH7	0x4000 1E68

表 26-3：DMACH 寄存器列表

寄存器	偏移地址	描述	复位值
DMACHSA	0x0	DMA 通道源地址寄存器	0x00000000
DMACHDA	0x8	DMA 通道目的地址寄存器	0x00000000
DMACHCTL0	0x18	DMA 通道控制寄存器 0	0x00304801
DMACHCTL1	0x1C	DMA 通道控制寄存器 1	0x00000002
DMACHCFG0	0x40	DMA 通道配置寄存器 0	0x00000E00
DMACHCFG1	0x44	DMA 通道配置寄存器 1	0x00000004

SPINTROL

26.6.2 DMACH 寄存器

表 26-4：DMA 通道源地址寄存器（DMACHSA）位段定义

DMACHSA (DMA Channel Source Address Register) Offset：0x0 Default：0x00000000 Access：DMACHx->DMACHSA							
31	30	29	28	27	26	25	24
ADDR							
23	22	21	20	19	18	17	16
ADDR							
15	14	13	12	11	10	9	8
ADDR							
7	6	5	4	3	2	1	0
ADDR							

表 26-5：DMA 通道源地址寄存器（DMACHSA）位段描述

位段	位段名	Type	Reset	描述
31:0	ADDR	RW	0x0	当前 DMA 传输源地址 本寄存器只能写入一个 32 位对齐的地址 每次源端传输后按照 DMACHCTL0.SINC 更新

表 26-6：DMA 通道源地址寄存器（DMACHDA）位段定义

DMACHDA (DMA Channel Destination Address Register) Offset：0x8 Default：0x00000000 Access：DMACHx->DMACHDA							
31	30	29	28	27	26	25	24
ADDR							
23	22	21	20	19	18	17	16
ADDR							
15	14	13	12	11	10	9	8
ADDR							
7	6	5	4	3	2	1	0
ADDR							

表 26-7：DMA 通道源地址寄存器（DMACHDA）位段描述

位段	位段名	Type	Reset	描述
31:0	ADDR	RW	0x0	当前 DMA 传输目的地址 本寄存器只能写入一个 32 位对齐的地址 每次目的端传输后按照 DMACHCTL0.DINC 更新

©2025 旋智电子科技（上海）有限公司

表 26-8：DMA 通道控制寄存器 0（DMACHCTL0）位段定义

DMACHCTL0 (DMA Channel Control Register 0)　Offset：0x18　Default：0x00304801
Access：DMACHx->DMACHCTL0

31	30	29	28	27	26	25	24
RESERVED			RESERVED	RESERVED	RESERVED		RESERVED
23	22	21	20	19	18	17	16
RESERVED	TT			RESERVED	RESERVED	RESERVED	SSIZE
15	14	13	12	11	10	9	8
SSIZE		DSIZE			SINC		DINC
7	6	5	4	3	2	1	0
DINC	SWIDTH			DWIDTH			IE

表 26-9：DMA 通道控制寄存器 0（DMACHCTL0）位段描述

位段	位段名	Type	Reset	描述
31:29	RESERVED_31_29	RO	0x0	保留
28	RESERVED_28	RW	0x0	保留
27	RESERVED_27	RW	0x0	保留
26:25	RESERVED_26_25	RW	0x0	保留
24:23	RESERVED_24_23	RW	0x0	保留
22:20	TT	RW	0x3	传输类型 对于多数据块传输而言，在传输过程中应保持 TT 不变 000：内存到内存，DMAC 为流控制器 001：内存到外设，DMAC 为流控制器 010：外设到内存，DMAC 为流控制器 011：外设到外设，DMAC 为流控制器 100：外设到内存，源端外设为流控制器 101：外设到外设，源端外设为流控制器 110：内存到外设，目的端外设为流控制器 111：外设到外设，目的端外设为流控制器
19	RESERVED_19	RO	0x0	保留
18	RESERVED_18	RW	0x0	保留
17	RESERVED_17	RW	0x0	保留
16:14	SSIZE	RW	0x1	源突发事务长度 即每个源突发事务请求读取的宽度为 SWDITH 的数据个数，单位 word 注意：本位段与 AHB 主机 HBURST 信号无关 0：1 个 1：4 个 2：8 个 3：16 个 4：32 个

©2025 旋智电子科技（上海）有限公司

位段	位段名	Type	Reset	描述
				5：64 个 6：128 个 7：256 个
13:11	DSIZE	RW	0x1	目的突发事务长度 即每个目的突发事务请求写入的宽度为 DWDITH 的数据个数，单位 word 注意：本位段与 AHB 主机 HBURST 信号无关 0：1 个 1：4 个 2：8 个 3：16 个 4：32 个 5：64 个 6：128 个 7：256 个
10:9	SINC	RW	0x0	每次传输后的源地址步进大小 0：递增到下一个 SWIDTH 位宽对齐的地址 1：递减到上一个 SWIDTH 位宽对齐的地址 2：保持不变
8:7	DINC	RW	0x0	每次传输后的目的地址步进大小 0：递增到下一个 DWIDTH 位宽对齐的地址 1：递减到上一个 DWIDTH 位宽对齐的地址 2：保持不变
6:4	SWIDTH	RW	0x0	源传输位宽 对于非内存外设，通常设为 FIFO 接口位宽 000：字节(8 位) 001：半字(16 位) 010：字(32 位) 011：双字(64 位) 100：4 字(128 位) 101：8 字(256 位)
3:1	DWIDTH	RW	0x0	目的传输位宽 对于非内存外设，通常设为 FIFO 接口位宽 000：字节(8 位) 001：半字(16 位) 010：字(32 位) 011：双字(64 位) 100：4 字(128 位) 101：8 字(256 位)
0	IE	RW	0x1	中断使能 本位段不影响中断原始事件标志的产生

©2025 旋智电子科技（上海）有限公司

位段	位段名	Type	Reset	描述
				0：关闭 1：使能

表 26-10：DMA 通道控制寄存器 1（DMACHCTL1）位段定义

DMACHCTL1 (DMA Channel Control Register 1)　Offset：0x1C　Default：0x00000002
Access：DMACHx->DMACHCTL1

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED			RESERVED	BLKTS			
7	6	5	4	3	2	1	0
BLKTS							

表 26-11：DMA 通道控制寄存器 1（DMACHCTL1）位段描述

Bits	位段名	Type	Reset	描述
31:13	RESERVED_31_13	RO	0x0	保留
12	RESERVED_12	RW	0x0	保留
11:0	BLKTS	RW	0x2	块传输大小 即每个块传输所需的单次事务的总数 单次事务映射为单次 AMBA 总线操作，其位宽由 DMACHCTL0 寄存器 SWIDTH 或 DWIDTH 位段指定。 本位段必须在通道使能前配置。

表 26-12：DMA 通道配置寄存器 0（DMACHCFG0）位段定义

DMACHCFG0 (DMA Channel Configure Register 0)　Offset：0x40　Default：0x00000E00
Access：DMACHx->DMACHCFG0

31	30	29	28	27	26	25	24
DRLD	SRLD	RESERVED					
23	22	21	20	19	18	17	16
RESERVED				SHSPOL	DHSPOL	RESERVED	RESERVED
15	14	13	12	11	10	9	8
RESERVED		RESERVED		SSWHS	DSWHS	FIFOEMPTY	SUSPEND
7	6	5	4	3	2	1	0
PRIORITY			RESERVED				

SPINTROL

表 26-13：DMA 通道配置寄存器 0（DMACHCFG0）位段描述

Bits	位段名	Type	Reset	描述
31	DRLD	RW	0x0	自动目的地址重载 在每次数据块传输完毕时，DMACHDA 可以重载为初始数值。 0：关闭 1：使能
30	SRLD	RW	0x0	自动源地址重载 在每次数据块传输完毕时，DMACHDA 可以重载为初始数值。 0：关闭 1：使能
29:20	RESERVED_29_20	RW	0x0	保留
19	SHSPOL	RW	0x0	源硬件握手信号极性 0：低电平有效 1：高电平有效
18	DHSPOL	RW	0x0	目的硬件握手信号极性 0：低电平有效 1：高电平有效
17	RESERVED_17	RW	0x0	保留
16	RESERVED_16	RW	0x0	保留
15:14	RESERVED_15_14	RW	0x0	保留
13:12	RESERVED_13_12	RW	0x0	保留
11	SSWHS	RW	0x1	源握手方式，源为内存时忽略本位段 0：硬件握手 1：软件握手
10	DSWHS	RW	0x1	目的握手方式，目的为内存时忽略本位段 0：硬件握手 1：软件握手
9	FIFOEMPTY	RO	0x1	指示 FIFO 是否为空，与 SUSPEND 位段配合可以干净地关闭一个通道 0：FIFO 非空（有数据） 1：FIFO 空（没有数据）
8	SUSPEND	RW	0x0	通道暂停 暂停 DMA 数据传输，直到该位被清除。暂停操作无法保证当前事务会完成。可以与 FIFOEMPTY 位结合使用，以在不丢失任何数据的情况下完全禁用通道。 0：运行 1：暂停

©2025 旋智电子科技（上海）有限公司

Bits	位段名	Type	Reset	描述
7:5	PRIORITY	RW	0x0	通道优先级：值越大，优先级越高
4:0	RESERVED_4_0	RO	0x0	保留

表 26-14：DMA 通道配置寄存器 1（DMACHCFG1）位段定义

DMACHCFG1 (DMA Channel Configure Register 1)　Offset：0x44　Default：0x00000004
Access：DMACHx->DMACHCFG1

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED	DPER				SPER		
7	6	5	4	3	2	1	0
SPER	RESERVED	RESERVED	RESERVED			FIFOTH	RESERVED

表 26-15：DMA 通道配置寄存器 1（DMACHCFG1）位段描述

Bits	位段名	Type	Reset	描述
31:15	RESERVED_31_15	RO	0x0	保留
14:11	DPER	RW	0x0	通过硬件握手接口交互的目的外设 DMACHCFG0.DSWHS=1 时忽略本位段 具体选项详见章节 26.3.2
10:7	SPER	RW	0x0	通过硬件握手接口交互的源外设 DMACHCFG0.SSWHS=1 时忽略本位段 具体选项详见章节 26.3.2
6	RESERVED_6	RW	0x0	保留
5	RESERVED_5	RW	0x0	保留
4:2	RESERVED_4_2	RW	0x1	保留
1	FIFOTH	RW	0x0	设定服务突发事务时 FIFO 内的数据个数 0：只要 FIFO 里有指定位宽的单次 AHB 传输所需的数据（目的事务）或者空位（源事务）就开始传输 1：FIFO 中用于目的传输的数据不小于 FIFO 容量的一半或者 FIFO 中用于源传输的空位大于 FIFO 容量的一半，除非是突发事务或者块传输的最后一次传输。
0	RESERVED_0	RW	0x0	保留

SPINTROL

26.6.3 DMAC 寄存器列表

表 26-16：DMAC 模块基地址

外设模块	基地址
DMAC	0x4000 1C00

表 26-17：DMAC 寄存器列表

寄存器	偏移地址	描述	复位值
DMATCRAWIF	0x2C0	DMA 传输完成中断原始事件标志寄存器	0x00000000
DMABTCRAWIF	0x2C8	DMA 块传输完成中断原始事件标志寄存器	0x00000000
DMASTCRAWIF	0x2D0	DMA 源端传输完成中断原始事件标志寄存器	0x00000000
DMADTCRAWIF	0x2D8	DMA 目的端传输完成中断原始事件标志寄存器	0x00000000
DMAERRRAWIF	0x2E0	DMA 总线错误中断原始事件标志寄存器	0x00000000
DMATCIF	0x2E8	DMA 传输完成中断标志寄存器	0x00000000
DMABTCIF	0x2F0	DMA 块传输完成中断标志寄存器	0x00000000
DMASTCIF	0x2F8	DMA 源端传输完成中断标志寄存器	0x00000000
DMADTCIF	0x300	DMA 目的端传输完成中断标志寄存器	0x00000000
DMAERRIF	0x308	DMA 总线错误中断标志寄存器	0x00000000
DMATCIE	0x310	DMA 传输完成中断使能寄存器	0x00000000
DMABTCIE	0x318	DMA 块传输完成中断使能寄存器	0x00000000
DMASTCIE	0x320	DMA 源端传输完成中断使能寄存器	0x00000000
DMADTCIE	0x328	DMA 目的端传输完成中断使能寄存器	0x00000000
DMAERRIE	0x330	DMA 总线错误中断使能寄存器	0x00000000
DMATCIC	0x338	DMA 传输完成中断清除寄存器	0x00000000
DMABTCIC	0x340	DMA 块传输完成中断清除寄存器	0x00000000
DMASTCIC	0x348	DMA 源端传输完成中断清除寄存器	0x00000000
DMADTCIC	0x350	DMA 目的端传输完成中断清除寄存器	0x00000000
DMAERRIC	0x358	DMA 总线错误中断清除寄存器	0x00000000
DMAIF	0x360	DMA 组合中断标志寄存器	0x00000000
DMASSREQ	0x368	DMA 源端软件事务请求寄存器	0x00000000
DMADSREQ	0x370	DMA 目的端软件事务请求寄存器	0x00000000
DMASSSREQ	0x378	DMA 源端软件事务单次请求寄存器	0x00000000
DMADSSREQ	0x380	DMA 目的端软件事务单次请求寄存器	0x00000000
DMASSLREQ	0x388	DMA 源端软件事务末次请求寄存器	0x00000000
DMADSLREQ	0x390	DMA 目的端软件事务末次请求寄存器	0x00000000
DMAEN	0x398	DMA 使能寄存器	0x00000000
DMACHEN	0x3A0	DMA 通道使能寄存器	0x00000000

©2025 旋智电子科技（上海）有限公司

26.6.4 DMAC 寄存器

表 26-18：DMA 传输完成中断原始事件标志寄存器（DMATCRAWIF）位段定义

DMATCRAWIF (DMA Transfer Complete Interrupt Raw Flag Register) Offset：0x2C0 Default：0x00000000 Access：DMACx->DMATCRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-19：DMA 传输完成中断原始事件标志寄存器（DMATCRAWIF）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	CH7	RO	0x0	通道 7 传输完成事件标志 0：传输未完成 1：传输完成
6	CH6	RO	0x0	通道 6 传输完成事件标志 0：传输未完成 1：传输完成
5	CH5	RO	0x0	通道 5 传输完成事件标志 0：传输未完成 1：传输完成
4	CH4	RO	0x0	通道 4 传输完成事件标志 0：传输未完成 1：传输完成
3	CH3	RO	0x0	通道 3 传输完成事件标志 0：传输未完成 1：传输完成
2	CH2	RO	0x0	通道 2 传输完成事件标志 0：传输未完成 1：传输完成
1	CH1	RO	0x0	通道 1 传输完成事件标志 0：传输未完成 1：传输完成

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
0	CH0	RO	0x0	通道 0 传输完成事件标志 0：传输未完成 1：传输完成

表 26-20：DMA 块传输完成中断原始事件标志寄存器（DMABTCRAWIF）位段定义

DMABTCRAWIF (DMA Block Transfer Complete Interrupt Raw Flag Register) Offset：0x2C8 Default：0x00000000 Access：DMACx->DMABTCRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-21：DMA 块传输完成中断原始事件标志寄存器（DMABTCRAWIF）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	CH7	RO	0x0	通道 7 块传输完成事件标志 0：传输未完成 1：传输完成
6	CH6	RO	0x0	通道 6 块传输完成事件标志 0：传输未完成 1：传输完成
5	CH5	RO	0x0	通道 5 块传输完成事件标志 0：传输未完成 1：传输完成
4	CH4	RO	0x0	通道 4 块传输完成事件标志 0：传输未完成 1：传输完成
3	CH3	RO	0x0	通道 3 块传输完成事件标志 0：传输未完成 1：传输完成
2	CH2	RO	0x0	通道 2 块传输完成事件标志 0：传输未完成 1：传输完成

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
1	CH1	RO	0x0	通道 1 块传输完成事件标志 0：传输未完成 1：传输完成
0	CH0	RO	0x0	通道 0 块传输完成事件标志 0：传输未完成 1：传输完成

表 26-22：DMA 源端传输完成中断原始事件标志寄存器（DMASTCRAWIF）位段定义

DMASTCRAWIF (DMA Source Transfer Complete Interrupt Raw Flag Register) Offset：0x2D0 Default：0x00000000 Access：DMACx->DMASTCRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-23：DMA 源端传输完成中断原始事件标志寄存器（DMASTCRAWIF）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	CH7	RO	0x0	通道 7 源端传输完成事件标志 0：传输未完成 1：传输完成
6	CH6	RO	0x0	通道 6 源端传输完成事件标志 0：传输未完成 1：传输完成
5	CH5	RO	0x0	通道 5 源端传输完成事件标志 0：传输未完成 1：传输完成
4	CH4	RO	0x0	通道 4 源端传输完成事件标志 0：传输未完成 1：传输完成
3	CH3	RO	0x0	通道 3 源端传输完成事件标志 0：传输未完成 1：传输完成

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
2	CH2	RO	0x0	通道 2 源端传输完成事件标志 0：传输未完成 1：传输完成
1	CH1	RO	0x0	通道 1 源端传输完成事件标志 0：传输未完成 1：传输完成
0	CH0	RO	0x0	通道 0 源端传输完成事件标志 0：传输未完成 1：传输完成

表 26-24：DMA 目的端传输完成中断原始事件标志寄存器（DMADTCRAWIF）位段定义

DMADTCRAWIF (DMA Destination Transfer Complete Interrupt Raw Flag Register) Offset：0x2D8 Default：0x00000000 Access：DMACx->DMADTCRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-25：DMA 目的端传输完成中断原始事件标志寄存器（DMADTCRAWIF）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	CH7	RO	0x0	通道 7 目的端传输完成事件标志 0：传输未完成 1：传输完成
6	CH6	RO	0x0	通道 6 目的端传输完成事件标志 0：传输未完成 1：传输完成
5	CH5	RO	0x0	通道 5 目的端传输完成事件标志 0：传输未完成 1：传输完成
4	CH4	RO	0x0	通道 4 目的端传输完成事件标志 0：传输未完成 1：传输完成

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
3	CH3	RO	0x0	通道 3 目的端传输完成事件标志 0：传输未完成 1：传输完成
2	CH2	RO	0x0	通道 2 目的端传输完成事件标志 0：传输未完成 1：传输完成
1	CH1	RO	0x0	通道 1 目的端传输完成事件标志 0：传输未完成 1：传输完成
0	CH0	RO	0x0	通道 0 目的端传输完成事件标志 0：传输未完成 1：传输完成

表 26-26：DMA 总线访问错误原始标志寄存器（DMAERRRAWIF）位段定义

DMAERRRAWIF (DMA Error Interrupt Raw Flag Register) Offset：0x2E0 Default：0x00000000 Access：DMACx->DMAERRRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-27：DMA 总线访问错误原始标志寄存器（DMAERRRAWIF）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	CH7	RO	0x0	通道 7 总线访问错误事件标志 0：总线访问正常 1：总线访问错误
6	CH6	RO	0x0	通道 6 总线访问错误事件标志 0：总线访问正常 1：总线访问错误
5	CH5	RO	0x0	通道 5 总线访问错误事件标志 0：总线访问正常 1：总线访问错误
4	CH4	RO	0x0	通道 4 总线访问错误事件标志

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				0：总线访问正常 1：总线访问错误
3	CH3	RO	0x0	通道 3 总线访问错误事件标志 0：总线访问正常 1：总线访问错误
2	CH2	RO	0x0	通道 2 总线访问错误事件标志 0：总线访问正常 1：总线访问错误
1	CH1	RO	0x0	通道 1 总线访问错误事件标志 0：总线访问正常 1：总线访问错误
0	CH0	RO	0x0	通道 0 总线访问错误事件标志 0：总线访问正常 1：总线访问错误

表 26-28：DMA 传输完成中断标志寄存器（DMATCIF）位段定义

DMATCIF (DMA Transfer Complete Interrupt Flag Register) Offset：0x2E8 Default：0x00000000 Access：DMACx->DMATCIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-29：DMA 传输完成中断标志寄存器（DMATCIF）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	CH7	RO	0x0	通道 7 传输完成中断标志 0：中断未触发 1：中断已触发
6	CH6	RO	0x0	通道 6 传输完成中断标志 0：中断未触发 1：中断已触发
5	CH5	RO	0x0	通道 5 传输完成中断标志

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				0：中断未触发 1：中断已触发
4	CH4	RO	0x0	通道 4 传输完成中断标志 0：中断未触发 1：中断已触发
3	CH3	RO	0x0	通道 3 传输完成中断标志 0：中断未触发 1：中断已触发
2	CH2	RO	0x0	通道 2 传输完成中断标志 0：中断未触发 1：中断已触发
1	CH1	RO	0x0	通道 1 传输完成中断标志 0：中断未触发 1：中断已触发
0	CH0	RO	0x0	通道 0 传输完成中断标志 0：中断未触发 1：中断已触发

表 26-30：DMA 块传输完成中断标志寄存器（DMABTCIF）位段定义

DMABTCIF (DMA Block Transfer Complete Interrupt Flag Register) Offset：0x2F0 Default：0x00000000 Access：DMACx->DMABTCIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-31：DMA 块传输完成中断标志寄存器（DMABTCIF）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	CH7	RO	0x0	通道 7 块传输完成中断标志 0：中断未触发 1：中断已触发

SPINTROL

位段	位段名	属性	复位值	描述
6	CH6	RO	0x0	通道 6 块传输完成中断标志 0：中断未触发 1：中断已触发
5	CH5	RO	0x0	通道 5 块传输完成中断标志 0：中断未触发 1：中断已触发
4	CH4	RO	0x0	通道 4 块传输完成中断标志 0：中断未触发 1：中断已触发
3	CH3	RO	0x0	通道 3 块传输完成中断标志 0：中断未触发 1：中断已触发
2	CH2	RO	0x0	通道 2 块传输完成中断标志 0：中断未触发 1：中断已触发
1	CH1	RO	0x0	通道 1 块传输完成中断标志 0：中断未触发 1：中断已触发
0	CH0	RO	0x0	通道 0 块传输完成中断标志 0：中断未触发 1：中断已触发

表 26-32：DMA 源端传输完成中断标志寄存器（DMASTCIF）位段定义

DMASTCIF (DMA Source Transfer Complete Interrupt Flag Register) Offset：0x2F8 Default：0x00000000 Access：DMACx->DMASTCIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

©2025 旋智电子科技（上海）有限公司

表 26-33：DMA 源端传输完成中断标志寄存器（DMASTCIF）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	CH7	RO	0x0	通道 7 源端传输完成中断标志 0：中断未触发 1：中断已触发
6	CH6	RO	0x0	通道 6 源端传输完成中断标志 0：中断未触发 1：中断已触发
5	CH5	RO	0x0	通道 5 源端传输完成中断标志 0：中断未触发 1：中断已触发
4	CH4	RO	0x0	通道 4 源端传输完成中断标志 0：中断未触发 1：中断已触发
3	CH3	RO	0x0	通道 3 源端传输完成中断标志 0：中断未触发 1：中断已触发
2	CH2	RO	0x0	通道 2 源端传输完成中断标志 0：中断未触发 1：中断已触发
1	CH1	RO	0x0	通道 1 源端传输完成中断标志 0：中断未触发 1：中断已触发
0	CH0	RO	0x0	通道 0 源端传输完成中断标志 0：中断未触发 1：中断已触发

SPINTROL

表 26-34：DMA 目的端传输完成中断标志寄存器（DMADTCIF）位段定义

DMADTCIF (DMA Destination Transfer Complete Interrupt Flag Register) Offset：0x300 Default：0x00000000 Access：DMACx->DMADTCIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-35：DMA 目的端传输完成中断标志寄存器（DMADTCIF）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	CH7	RO	0x0	通道 7 目的端传输完成中断标志 0：中断未触发 1：中断已触发
6	CH6	RO	0x0	通道 6 目的端传输完成中断标志 0：中断未触发 1：中断已触发
5	CH5	RO	0x0	通道 5 目的端传输完成中断标志 0：中断未触发 1：中断已触发
4	CH4	RO	0x0	通道 4 目的端传输完成中断标志 0：中断未触发 1：中断已触发
3	CH3	RO	0x0	通道 3 目的端传输完成中断标志 0：中断未触发 1：中断已触发
2	CH2	RO	0x0	通道 2 目的端传输完成中断标志 0：中断未触发 1：中断已触发
1	CH1	RO	0x0	通道 1 目的端传输完成中断标志 0：中断未触发 1：中断已触发
0	CH0	RO	0x0	通道 0 目的端传输完成中断标志

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				0：中断未触发 1：中断已触发

表 26-36：DMA 总线访问错误中断标志寄存器（DMAERRIF）位段定义

DMAERRIF (DMA Error Interrupt Flag Register) Offset：0x308 Default：0x00000000 Access：DMACx->DMAERRIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-37：DMA 总线访问错误中断标志寄存器（DMAERRIF）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	CH7	RO	0x0	通道 7 总线访问错误中断标志 0：中断未触发 1：中断已触发
6	CH6	RO	0x0	通道 6 总线访问错误中断标志 0：中断未触发 1：中断已触发
5	CH5	RO	0x0	通道 5 总线访问错误中断标志 0：中断未触发 1：中断已触发
4	CH4	RO	0x0	通道 4 总线访问错误中断标志 0：中断未触发 1：中断已触发
3	CH3	RO	0x0	通道 3 总线访问错误中断标志 0：中断未触发 1：中断已触发
2	CH2	RO	0x0	通道 2 总线访问错误中断标志 0：中断未触发 1：中断已触发

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
1	CH1	RO	0x0	通道 1 总线访问错误中断标志 0：中断未触发 1：中断已触发
0	CH0	RO	0x0	通道 0 总线访问错误中断标志 0：中断未触发 1：中断已触发

表 26-38：DMA 传输完成中断使能寄存器（DMATCIE）位段定义

DMATCIE (DMA Transfer Complete Interrupt Enable Register) Offset：0x310 Default：0x00000000 Access：DMACx->DMATCIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-39：DMA 传输完成中断使能寄存器（DMATCIE）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	CH7WE	W1S	0x0	CH7 位段写使能（必须与 CH7 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
14	CH6WE	W1S	0x0	CH6 位段写使能（必须与 CH6 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
13	CH5WE	W1S	0x0	CH5 位段写使能（必须与 CH5 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
12	CH4WE	W1S	0x0	CH4 位段写使能（必须与 CH4 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
11	CH3WE	W1S	0x0	CH3 位段写使能（必须与 CH3 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
10	CH2WE	W1S	0x0	CH2 位段写使能（必须与 CH2 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
9	CH1WE	W1S	0x0	CH1 位段写使能（必须与 CH1 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
8	CH0WE	W1S	0x0	CH0 位段写使能（必须与 CH0 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
7	CH7	RW	0x0	通道 7 传输完成中断使能 0：关闭 1：使能
6	CH6	RW	0x0	通道 6 传输完成中断使能 0：关闭 1：使能
5	CH5	RW	0x0	通道 5 传输完成中断使能 0：关闭 1：使能
4	CH4	RW	0x0	通道 4 传输完成中断使能 0：关闭 1：使能
3	CH3	RW	0x0	通道 3 传输完成中断使能 0：关闭 1：使能
2	CH2	RW	0x0	通道 2 传输完成中断使能 0：关闭 1：使能
1	CH1	RW	0x0	通道 1 传输完成中断使能 0：关闭 1：使能
0	CH0	RW	0x0	通道 0 传输完成中断使能 0：关闭 1：使能

SPINTROL

表 26-40：DMA 块传输完成中断使能寄存器（DMABTCIE）位段定义

DMABTCIE (DMA Block Transfer Complete Interrupt Enable Register) Offset：0x318 Default：0x00000000 Access：DMACx->DMABTCIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-41：DMA 块传输完成中断使能寄存器（DMABTCIE）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	CH7WE	W1S	0x0	CH7 位段写使能（必须与 CH7 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
14	CH6WE	W1S	0x0	CH6 位段写使能（必须与 CH6 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
13	CH5WE	W1S	0x0	CH5 位段写使能（必须与 CH5 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
12	CH4WE	W1S	0x0	CH4 位段写使能（必须与 CH4 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
11	CH3WE	W1S	0x0	CH3 位段写使能（必须与 CH3 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
10	CH2WE	W1S	0x0	CH2 位段写使能（必须与 CH2 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
9	CH1WE	W1S	0x0	CH1 位段写使能（必须与 CH1 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
8	CH0WE	W1S	0x0	CH0 位段写使能（必须与 CH0 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
7	CH7	RW	0x0	通道 7 块传输完成中断使能 0：关闭 1：使能
6	CH6	RW	0x0	通道 6 块传输完成中断使能 0：关闭 1：使能
5	CH5	RW	0x0	通道 5 块传输完成中断使能 0：关闭 1：使能
4	CH4	RW	0x0	通道 4 块传输完成中断使能 0：关闭 1：使能
3	CH3	RW	0x0	通道 3 块传输完成中断使能 0：关闭 1：使能
2	CH2	RW	0x0	通道 2 块传输完成中断使能 0：关闭 1：使能
1	CH1	RW	0x0	通道 1 块传输完成中断使能 0：关闭 1：使能
0	CH0	RW	0x0	通道 0 块传输完成中断使能 0：关闭 1：使能

表 26-42：DMA 源端传输完成中断使能寄存器（DMASTCIE）位段定义

DMASTCIE (DMA Source Transfer Complete Interrupt Enable Register) Offset：0x320 Default：0x00000000 Access：DMACx->DMASTCIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

©2025 旋智电子科技（上海）有限公司

表 26-43：DMA 源端传输完成中断使能寄存器（DMASTCIE）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	CH7WE	W1S	0x0	CH7 位段写使能（必须与 CH7 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
14	CH6WE	W1S	0x0	CH6 位段写使能（必须与 CH6 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
13	CH5WE	W1S	0x0	CH5 位段写使能（必须与 CH5 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
12	CH4WE	W1S	0x0	CH4 位段写使能（必须与 CH4 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
11	CH3WE	W1S	0x0	CH3 位段写使能（必须与 CH3 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
10	CH2WE	W1S	0x0	CH2 位段写使能（必须与 CH2 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
9	CH1WE	W1S	0x0	CH1 位段写使能（必须与 CH1 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
8	CH0WE	W1S	0x0	CH0 位段写使能（必须与 CH0 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
7	CH7	RW	0x0	通道 7 源端传输完成中断使能 0：关闭 1：使能
6	CH6	RW	0x0	通道 6 源端传输完成中断使能 0：关闭 1：使能
5	CH5	RW	0x0	通道 5 源端传输完成中断使能 0：关闭 1：使能
4	CH4	RW	0x0	通道 4 源端传输完成中断使能 0：关闭 1：使能

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
3	CH3	RW	0x0	通道 3 源端传输完成中断使能 0：关闭 1：使能
2	CH2	RW	0x0	通道 2 源端传输完成中断使能 0：关闭 1：使能
1	CH1	RW	0x0	通道 1 源端传输完成中断使能 0：关闭 1：使能
0	CH0	RW	0x0	通道 0 源端传输完成中断使能 0：关闭 1：使能

表 26-44：DMA 目的端传输完成中断使能寄存器（DMADTCIE）位段定义

DMADTCIE (DMA Destination Transfer Complete Interrupt Enable Register)　Offset：0x328 Default：0x00000000 Access：DMACx->DMADTCIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-45：DMA 目的端传输完成中断使能寄存器（DMADTCIE）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	CH7WE	W1S	0x0	CH7 位段写使能（必须与 CH7 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
14	CH6WE	W1S	0x0	CH6 位段写使能（必须与 CH6 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
13	CH5WE	W1S	0x0	CH5 位段写使能（必须与 CH5 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
12	CH4WE	W1S	0x0	CH4 位段写使能（必须与 CH4 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
11	CH3WE	W1S	0x0	CH3 位段写使能（必须与 CH3 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
10	CH2WE	W1S	0x0	CH2 位段写使能（必须与 CH2 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
9	CH1WE	W1S	0x0	CH1 位段写使能（必须与 CH1 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
8	CH0WE	W1S	0x0	CH0 位段写使能（必须与 CH0 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
7	CH7	RW	0x0	通道 7 目的端传输完成中断使能 0：关闭 1：使能
6	CH6	RW	0x0	通道 6 目的端传输完成中断使能 0：关闭 1：使能
5	CH5	RW	0x0	通道 5 目的端传输完成中断使能 0：关闭 1：使能
4	CH4	RW	0x0	通道 4 目的端传输完成中断使能 0：关闭 1：使能
3	CH3	RW	0x0	通道 3 目的端传输完成中断使能 0：关闭 1：使能
2	CH2	RW	0x0	通道 2 目的端传输完成中断使能 0：关闭 1：使能
1	CH1	RW	0x0	通道 1 目的端传输完成中断使能 0：关闭 1：使能

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
0	CH0	RW	0x0	通道 0 目的端传输完成中断使能 0：关闭 1：使能

表 26-46：DMA 总线访问错误中断使能寄存器（DMAERRIE）位段定义

DMAERRIE (DMA Error Interrupt Enable Register) Offset：0x330 Default：0x00000000 Access：DMACx->DMAERRIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-47：DMA 总线访问错误中断使能寄存器（DMAERRIE）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	CH7WE	W1S	0x0	CH7 位段写使能（必须与 CH7 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
14	CH6WE	W1S	0x0	CH6 位段写使能（必须与 CH6 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
13	CH5WE	W1S	0x0	CH5 位段写使能（必须与 CH5 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
12	CH4WE	W1S	0x0	CH4 位段写使能（必须与 CH4 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
11	CH3WE	W1S	0x0	CH3 位段写使能（必须与 CH3 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
10	CH2WE	W1S	0x0	CH2 位段写使能（必须与 CH2 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零

SPINTROL

位段	位段名	属性	复位值	描述
9	CH1WE	W1S	0x0	CH1 位段写使能（必须与 CH1 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
8	CH0WE	W1S	0x0	CH0 位段写使能（必须与 CH0 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
7	CH7	RW	0x0	通道 7 总线访问错误中断使能 0：关闭 1：使能
6	CH6	RW	0x0	通道 6 总线访问错误中断使能 0：关闭 1：使能
5	CH5	RW	0x0	通道 5 总线访问错误中断使能 0：关闭 1：使能
4	CH4	RW	0x0	通道 4 总线访问错误中断使能 0：关闭 1：使能
3	CH3	RW	0x0	通道 3 总线访问错误中断使能 0：关闭 1：使能
2	CH2	RW	0x0	通道 2 总线访问错误中断使能 0：关闭 1：使能
1	CH1	RW	0x0	通道 1 总线访问错误中断使能 0：关闭 1：使能
0	CH0	RW	0x0	通道 0 总线访问错误中断使能 0：关闭 1：使能

©2025 旋智电子科技（上海）有限公司

表 26-48：DMA 传输完成中断清除寄存器（DMATCIC）位段定义

DMATCIC (DMA Transfer Complete Interrupt Clear Register) Offset：0x338 Default：0x00000000 Access：DMACx->DMATCIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-49：DMA 传输完成中断清除寄存器（DMATCIC）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	CH7	W1C	0x0	通道 7 传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
6	CH6	W1C	0x0	通道 6 传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
5	CH5	W1C	0x0	通道 5 传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
4	CH4	W1C	0x0	通道 4 传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
3	CH3	W1C	0x0	通道 3 传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
2	CH2	W1C	0x0	通道 2 传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
1	CH1	W1C	0x0	通道 1 传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
0	CH0	W1C	0x0	通道 0 传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

表 26-50：DMA 块传输完成中断清除寄存器（DMABTCIC）位段定义

DMABTCIC (DMA Block Transfer Complete Interrupt Clear Register) Offset：0x340 Default：0x00000000 Access：DMACx->DMABTCIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-51：DMA 块传输完成中断清除寄存器（DMABTCIC）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	CH7	W1C	0x0	通道 7 块传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
6	CH6	W1C	0x0	通道 6 块传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
5	CH5	W1C	0x0	通道 5 块传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
4	CH4	W1C	0x0	通道 4 块传输完成中断清除

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
3	CH3	W1C	0x0	通道 3 块传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
2	CH2	W1C	0x0	通道 2 块传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
1	CH1	W1C	0x0	通道 1 块传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
0	CH0	W1C	0x0	通道 0 块传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

表 26-52：DMA 源端传输完成中断清除寄存器（DMASTCIC）位段定义

DMASTCIC (DMA Source Transfer Complete Interrupt Clear Register) Offset：0x348 Default：0x00000000 Access：DMACx->DMASTCIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-53：DMA 源端传输完成中断清除寄存器（DMASTCIC）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	CH7	W1C	0x0	通道 7 源端传输完成中断清除

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
6	CH6	W1C	0x0	通道 6 源端传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
5	CH5	W1C	0x0	通道 5 源端传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
4	CH4	W1C	0x0	通道 4 源端传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
3	CH3	W1C	0x0	通道 3 源端传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
2	CH2	W1C	0x0	通道 2 源端传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
1	CH1	W1C	0x0	通道 1 源端传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
0	CH0	W1C	0x0	通道 0 源端传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

表 26-54：DMA 目的端传输完成中断清除寄存器（DMADTCIC）位段定义

DMADTCIC (DMA Destination Transfer Complete Interrupt Clear Register) Offset：0x350 Default：0x00000000 Access：DMACx->DMADTCIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16

©2025 旋智电子科技（上海）有限公司

RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-55：DMA 目的端传输完成中断清除寄存器（DMADTCIC）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	CH7	W1C	0x0	通道 7 目的端传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
6	CH6	W1C	0x0	通道 6 目的端传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
5	CH5	W1C	0x0	通道 5 目的端传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
4	CH4	W1C	0x0	通道 4 目的端传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
3	CH3	W1C	0x0	通道 3 目的端传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
2	CH2	W1C	0x0	通道 2 目的端传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
1	CH1	W1C	0x0	通道 1 目的端传输完成中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
0	CH0	W1C	0x0	通道 0 目的端传输完成中断清除

SPINTROL

位段	位段名	属性	复位值	描述
				0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

表 26-56：DMA 总线访问错误中断清除寄存器（DMAERRIC）位段定义

DMAERRIC (DMA Error Interrupt Clear Register) Offset：0x358 Default：0x00000000 Access： DMACx->DMAERRIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-57：DMA 总线访问错误中断清除寄存器（DMAERRIC）位段描述

位段	位段名	属性	复位值	描述
31:8	RESERVED_31_8	RO	0x0	保留
7	CH7	W1S	0x0	通道 7 总线访问错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
6	CH6	W1S	0x0	通道 6 总线访问错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
5	CH5	W1S	0x0	通道 5 总线访问错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
4	CH4	W1S	0x0	通道 4 总线访问错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
3	CH3	W1S	0x0	通道 3 总线访问错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
2	CH2	W1S	0x0	通道 2 总线访问错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
1	CH1	W1S	0x0	通道 1 总线访问错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零
0	CH0	W1S	0x0	通道 0 总线访问错误中断清除 0：写 0 无效，读总是 0 1：写 1 清除事件和中断标志，本位段自动清零

表 26-58：DMA 组合中断标志寄存器（DMAIF）位段定义

DMAIF (DMA Combined Interrupt Flag Register)　Offset：0x360　Default：0x00000000
Access：DMACx->DMAIF

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED			ERROR	DC	SC	BC	TC

表 26-59：DMA 组合中断标志寄存器（DMAIF）位段描述

位段	位段名	属性	复位值	描述
31:5	RESERVED_31_5	RO	0x0	保留
4	ERROR	RO	0x0	总线访问错误全局中断标志 本位段是 DMAERRIF 所有位段的逻辑或 0：无任何通道触发总线访问错误中断 1：至少有一个通道触发总线访问错误中断
3	DC	RO	0x0	目的端传输完成全局中断标志 本位段是 DMADTCIF 所有位段的逻辑或 0：无任何通道触发目的端传输完成中断 1：至少有一个通道触发目的端传输完成中断

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
2	SC	RO	0x0	源端传输完成全局中断标志 本位段是 DMASTCIF 所有位段的逻辑或 0：无任何通道触发源端传输完成中断 1：至少有一个通道触发源端传输完成中断
1	BC	RO	0x0	块传输完成全局中断标志 本位段是 DMABTCIF 所有位段的逻辑或 0：无任何通道触发块传输完成中断 1：至少有一个通道触发块传输完成中断
0	TC	RO	0x0	传输完成全局中断标志 本位段是 DMATCIF 所有位段的逻辑或 0：无任何通道触发传输完成中断 1：至少有一个通道触发传输完成中断

表 26-60：DMA 源端软件事务请求寄存器（DMASSREQ）位段定义

DMASSREQ (DMA Source Software Transaction Request Register) Offset：0x368 Default：0x00000000 Access：DMACx->DMASSREQ							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-61：DMA 源端软件事务请求寄存器（DMASSREQ）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	CH7WE	W1S	0x0	CH7 位段写使能（必须与 CH7 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
14	CH6WE	W1S	0x0	CH6 位段写使能（必须与 CH6 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
13	CH5WE	W1S	0x0	CH5 位段写使能（必须与 CH5 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
12	CH4WE	W1S	0x0	CH4 位段写使能（必须与 CH4 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
11	CH3WE	W1S	0x0	CH3 位段写使能（必须与 CH3 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
10	CH2WE	W1S	0x0	CH2 位段写使能（必须与 CH2 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
9	CH1WE	W1S	0x0	CH1 位段写使能（必须与 CH1 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
8	CH0WE	W1S	0x0	CH0 位段写使能（必须与 CH0 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
7	CH7	RW	0x0	通道 7 源端软件事务请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
6	CH6	RW	0x0	通道 6 源端软件事务请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
5	CH5	RW	0x0	通道 5 源端软件事务请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
4	CH4	RW	0x0	通道 4 源端软件事务请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
3	CH3	RW	0x0	通道 3 源端软件事务请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
2	CH2	RW	0x0	通道 2 源端软件事务请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
1	CH1	RW	0x0	通道 1 源端软件事务请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零

SPINTROL

位段	位段名	属性	复位值	描述
0	CH0	RW	0x0	通道 0 源端软件事务请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零

表 26-62：DMA 目的端软件事务请求寄存器（DMADSREQ）位段定义

DMADSREQ (DMA Destination Software Transaction Request Register) Offset：0x370 Default：0x00000000 Access：DMACx->DMADSREQ							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-63：DMA 目的端软件事务请求寄存器（DMADSREQ）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	CH7WE	W1S	0x0	CH7 位段写使能（必须与 CH7 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
14	CH6WE	W1S	0x0	CH6 位段写使能（必须与 CH6 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
13	CH5WE	W1S	0x0	CH5 位段写使能（必须与 CH5 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
12	CH4WE	W1S	0x0	CH4 位段写使能（必须与 CH4 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
11	CH3WE	W1S	0x0	CH3 位段写使能（必须与 CH3 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
10	CH2WE	W1S	0x0	CH2 位段写使能（必须与 CH2 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
9	CH1WE	W1S	0x0	CH1 位段写使能（必须与 CH1 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
8	CH0WE	W1S	0x0	CH0 位段写使能（必须与 CH0 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
7	CH7	RW	0x0	通道 7 目的端软件事务请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
6	CH6	RW	0x0	通道 6 目的端软件事务请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
5	CH5	RW	0x0	通道 5 目的端软件事务请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
4	CH4	RW	0x0	通道 4 目的端软件事务请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
3	CH3	RW	0x0	通道 3 目的端软件事务请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
2	CH2	RW	0x0	通道 2 目的端软件事务请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
1	CH1	RW	0x0	通道 1 目的端软件事务请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
0	CH0	RW	0x0	通道 0 目的端软件事务请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零

表 26-64：DMA 源端软件事务单次请求寄存器（DMASSSREQ）位段定义

DMASSSREQ (DMA Source Software Transaction Single Request Register)　Offset：0x378 Default：0x00000000 Access：DMACx->DMASSSREQ							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16

RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-65：DMA 源端软件事务单次请求寄存器（DMASSSREQ）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	CH7WE	W1S	0x0	CH7 位段写使能（必须与 CH7 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
14	CH6WE	W1S	0x0	CH6 位段写使能（必须与 CH6 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
13	CH5WE	W1S	0x0	CH5 位段写使能（必须与 CH5 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
12	CH4WE	W1S	0x0	CH4 位段写使能（必须与 CH4 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
11	CH3WE	W1S	0x0	CH3 位段写使能（必须与 CH3 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
10	CH2WE	W1S	0x0	CH2 位段写使能（必须与 CH2 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
9	CH1WE	W1S	0x0	CH1 位段写使能（必须与 CH1 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
8	CH0WE	W1S	0x0	CH0 位段写使能（必须与 CH0 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
7	CH7	RW	0x0	通道 7 源端软件事务单次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
6	CH6	RW	0x0	通道 6 源端软件事务单次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
5	CH5	RW	0x0	通道 5 源端软件事务单次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
4	CH4	RW	0x0	通道 4 源端软件事务单次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
3	CH3	RW	0x0	通道 3 源端软件事务单次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
2	CH2	RW	0x0	通道 2 源端软件事务单次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
1	CH1	RW	0x0	通道 1 源端软件事务单次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
0	CH0	RW	0x0	通道 0 源端软件事务单次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零

表 26-66：DMA 目的端软件事务单次请求寄存器（DMADSSREQ）位段定义

DMADSSREQ (DMA Destination Software Transaction Single Request Register)　Offset：0x380 Default：0x00000000 Access：DMACx->DMADSSREQ							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-67：DMA 目的端软件事务单次请求寄存器（DMADSSREQ）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
15	CH7WE	W1S	0x0	CH7 位段写使能（必须与 CH7 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
14	CH6WE	W1S	0x0	CH6 位段写使能（必须与 CH6 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
13	CH5WE	W1S	0x0	CH5 位段写使能（必须与 CH5 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
12	CH4WE	W1S	0x0	CH4 位段写使能（必须与 CH4 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
11	CH3WE	W1S	0x0	CH3 位段写使能（必须与 CH3 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
10	CH2WE	W1S	0x0	CH2 位段写使能（必须与 CH2 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
9	CH1WE	W1S	0x0	CH1 位段写使能（必须与 CH1 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
8	CH0WE	W1S	0x0	CH0 位段写使能（必须与 CH0 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
7	CH7	RW	0x0	通道 7 目的端软件事务单次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
6	CH6	RW	0x0	通道 6 目的端软件事务单次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
5	CH5	RW	0x0	通道 5 目的端软件事务单次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
4	CH4	RW	0x0	通道 4 目的端软件事务单次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
3	CH3	RW	0x0	通道 3 目的端软件事务单次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
2	CH2	RW	0x0	通道 2 目的端软件事务单次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
1	CH1	RW	0x0	通道 1 目的端软件事务单次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
0	CH0	RW	0x0	通道 0 目的端软件事务单次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零

表 26-68：DMA 源端软件事务末次请求寄存器（DMASSLREQ）位段定义

DMASSLREQ (DMA Source Software Transaction Last Request Register) Offset：0x388 Default：0x00000000 Access：DMACx->DMASSLREQ							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-69：DMA 源端软件事务末次请求寄存器（DMASSLREQ）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	CH7WE	W1S	0x0	CH7 位段写使能（必须与 CH7 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
14	CH6WE	W1S	0x0	CH6 位段写使能（必须与 CH6 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
13	CH5WE	W1S	0x0	CH5 位段写使能（必须与 CH5 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
12	CH4WE	W1S	0x0	CH4 位段写使能（必须与 CH4 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
11	CH3WE	W1S	0x0	CH3 位段写使能（必须与 CH3 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
10	CH2WE	W1S	0x0	CH2 位段写使能（必须与 CH2 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
9	CH1WE	W1S	0x0	CH1 位段写使能（必须与 CH1 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
8	CH0WE	W1S	0x0	CH0 位段写使能（必须与 CH0 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
7	CH7	RW	0x0	通道 7 源端软件事务末次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
6	CH6	RW	0x0	通道 6 源端软件事务末次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
5	CH5	RW	0x0	通道 5 源端软件事务末次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
4	CH4	RW	0x0	通道 4 源端软件事务末次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
3	CH3	RW	0x0	通道 3 源端软件事务末次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
2	CH2	RW	0x0	通道 2 源端软件事务末次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
1	CH1	RW	0x0	通道 1 源端软件事务末次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
0	CH0	RW	0x0	通道 0 源端软件事务末次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零

SPINTROL

表 26-70：DMA 目的端软件事务末次请求寄存器（DMADSLREQ）位段定义

DMADSLREQ (DMA Destination Software Transaction Last Request Register) Offset：0x390 Default：0x00000000 Access：DMACx->DMADSLREQ							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-71：DMA 目的端软件事务末次请求寄存器（DMADSLREQ）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	CH7WE	W1S	0x0	CH7 位段写使能（必须与 CH7 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
14	CH6WE	W1S	0x0	CH6 位段写使能（必须与 CH6 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
13	CH5WE	W1S	0x0	CH5 位段写使能（必须与 CH5 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
12	CH4WE	W1S	0x0	CH4 位段写使能（必须与 CH4 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
11	CH3WE	W1S	0x0	CH3 位段写使能（必须与 CH3 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
10	CH2WE	W1S	0x0	CH2 位段写使能（必须与 CH2 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
9	CH1WE	W1S	0x0	CH1 位段写使能（必须与 CH1 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
8	CH0WE	W1S	0x0	CH0 位段写使能（必须与 CH0 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
7	CH7	RW	0x0	通道 7 目的端软件事务末次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
6	CH6	RW	0x0	通道 6 目的端软件事务末次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
5	CH5	RW	0x0	通道 5 目的端软件事务末次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
4	CH4	RW	0x0	通道 4 目的端软件事务末次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
3	CH3	RW	0x0	通道 3 目的端软件事务末次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
2	CH2	RW	0x0	通道 2 目的端软件事务末次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
1	CH1	RW	0x0	通道 1 目的端软件事务末次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零
0	CH0	RW	0x0	通道 0 目的端软件事务末次请求 0：写 0 无效 1：写 1 发起请求，事务完成后位段自动清零

表 26-72：DMA 使能寄存器（DMAEN）位段定义

DMAEN (DMA Enable Register) Offset：0x398 Default：0x00000000 Access：DMACx->DMAEN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							EN

©2025 旋智电子科技（上海）有限公司

表 26-73：DMA 使能寄存器（DMAEN）位段描述

位段	位段名	属性	复位值	描述
31:1	RESERVED_31_1	RO	0x0	保留
0	EN	RW	0x0	DMA 使能 0：关闭 当本位段写 0 时仍有通道在传输数据，则回读值保持为 0，直到所有通道均完成数据传输，位段清零 1：使能

表 26-74：DMA 通道使能寄存器（DMACHEN）位段定义

DMACHEN (DMA Channel Enable Register)　Offset：0x3A0　Default：0x00000000
Access：DMACx->DMACHEN

31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

表 26-75：DMA 通道使能寄存器（DMACHEN）位段描述

位段	位段名	属性	复位值	描述
31:16	RESERVED_31_16	RO	0x0	保留
15	CH7WE	W1S	0x0	CH7 位段写使能（必须与 CH7 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
14	CH6WE	W1S	0x0	CH6 位段写使能（必须与 CH6 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
13	CH5WE	W1S	0x0	CH5 位段写使能（必须与 CH5 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
12	CH4WE	W1S	0x0	CH4 位段写使能（必须与 CH4 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
11	CH3WE	W1S	0x0	CH3 位段写使能（必须与 CH3 位段同时写）

©2025 旋智电子科技（上海）有限公司

位段	位段名	属性	复位值	描述
				0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
10	CH2WE	W1S	0x0	CH2 位段写使能（必须与 CH2 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
9	CH1WE	W1S	0x0	CH1 位段写使能（必须与 CH1 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
8	CH0WE	W1S	0x0	CH0 位段写使能（必须与 CH0 位段同时写） 0：写 0 禁止改写，读总是 0 1：写 1 使能改写，本位段自动清零
7	CH7	RW	0x0	通道 7 传输使能，传输完成后位段自动清零 0：关闭 1：使能
6	CH6	RW	0x0	通道 6 传输使能，传输完成后位段自动清零 0：关闭 1：使能
5	CH5	RW	0x0	通道 5 传输使能，传输完成后位段自动清零 0：关闭 1：使能
4	CH4	RW	0x0	通道 4 传输使能，传输完成后位段自动清零 0：关闭 1：使能
3	CH3	RW	0x0	通道 3 传输使能，传输完成后位段自动清零 0：关闭 1：使能
2	CH2	RW	0x0	通道 2 传输使能，传输完成后位段自动清零 0：关闭 1：使能
1	CH1	RW	0x0	通道 1 传输使能，传输完成后位段自动清零 0：关闭 1：使能
0	CH0	RW	0x0	通道 0 传输使能，传输完成后位段自动清零 0：关闭 1：使能

SPINTROL