匠芯创科技
ArtInChip

D12x 用户指南

Version 1.71

修订日期： 2025-07-04

版权声明

本文档是匠芯创科技（“ArtInChip”）的原创作品，匠芯创科技拥有该文档的全部版权。全部或部分复制必须获得匠芯创科技的书面批准，并向版权所有人明确确认。凡侵犯本公司版权等知识产权的，本公司将保留依法追究其法律责任的权利。

在法律允许的范围内，在此声明：使用前请仔细阅读合同条款和条件以及相关说明，并严格遵守本文档中的说明。匠芯创科技不对不当行为的后果（包括但不限于电压过高、超频或温度过高）承担任何责任。

匠芯创科技提供的信息仅作为参考或典型应用，本文档中的所有声明、信息和建议不构成任何明示或暗示的担保。匠芯创科技保留随时更改电路设计和/或规格的权利，恕不另行通知。

客户应全权负责获得实施解决方案/产品可能需要的第三方许可，匠芯创科技不承担任何与第三方许可相关的许可费或特许权使用费。对于任何要求的第三方许可证所涵盖的事项，匠芯创科技不承担任何保证、赔偿或其它义务。

凡以任何方式直接或间接使用本文档资料者，视为自愿接受本文档声明的约束。

修订记录

表 0–1 修订记录

版本	章节	修订说明
V1.71	System Configuration (SYSCFG)	在寄存器描述中添加了如下修改： • 修改了 LDO1x_SOFT_EN 位的配置描述。 • 在 LDO1x_EN 位，新增仅在 LDO1x_SOFT_EN 配置为1 时此位才有效的说明。
V1.70	–	对以下章节新增了编辑性修改，包括但不限于格式调整和内容优化。 • Inter Integrated Circuit (I2C) • Consumer Infrared Radiation (CIR) • Resistance Touch Panel (RTP) • Thermal Sensor (THS) • Audio

ArtInChip

内容

版权声明 ii
修订记录 iii
1. 文档说明 8
1.1. 适用范围 8
1.2. 内容组成 8
1.3. 寄存器属性 8
1.4. 数值定义 8
1.5. 术语、缩写词 9
2. 简介 13
2.1. 功能特性 14
3. D12x 地址映射 17
4. 引脚复用 20
4.1. D12x 功能复用 20
4.2. D12x QFN68 封装引脚说明 22
4.3. D121 QFN68 封装功能复用表 22
4.4. D123 QFN68 封装功能复用表 25
5. 处理器和总线 28
5.1. Central Process Unit (CPU) 28
5.1.1. 特性说明 28
5.1.2. 中断号 28
5.2. BUS 30
5.3. AXI Configuration (AXICFG) 31
5.3.1. 特性说明 31
5.3.2. 功能描述 31
5.3.3. 寄存器列表 32
5.3.4. 寄存器描述 32
5.4. Memory Top (MTOP) 33
5.4.1. 特性说明 33
5.4.2. 原理框图 34
5.4.3. 功能描述 34
5.4.4. 寄存器列表 35
5.4.5. 寄存器描述 36
5.5. Direct Memory Access (DMA) 38
5.5.1. 特性说明 39
5.5.2. 原理框图 39
5.5.3. 功能描述 39
5.5.4. 寄存器列表 41
5.5.5. 寄存器描述 42
6. 安全 57
6.1. Serial Peripheral Interface Encryption (SPI ENC) 57
6.1.1. 特性说明 57
6.1.2. 原理框图 58
6.1.3. 功能描述 58
6.1.4. 编程指南 61
6.1.5. 寄存器列表 62
6.1.6. 寄存器描述 63
6.2. Secure ID (SID) 67
6.2.1. 特性说明 67
6.2.2. 原理框图 68
6.2.3. 功能描述 68
6.2.4. 编程指南 72
6.2.5. 寄存器列表 73
6.2.6. 寄存器描述 73
6.3. Data Check Engine (DCE) 77
6.3.1. 特性说明 77
6.3.2. 原理框图 78
6.3.3. 功能描述 78

AriInChip

6.3.4. 寄存器列表 82
6.3.5. 寄存器描述 82
7. 启动 88
7.1. 模块概述 88
7.2. 选择启动类型 88
7.2.1. 启动流程 89
7.3. 正常启动模式 89
7.3.1. 启动介质选择 89
7.3.2. 使用 eFuse 配置启动介质 90
7.3.3. eFuse 中的 BROM 配置区域 91
7.3.4. SD/eMMC 启动 92
7.3.5. SPI NAND 启动 100
7.3.6. SPI NOR 启动 104
7.4. 烧录和升级模式 105
7.4.1. SiP Flash 烧录 106
7.4.2. 在线升级模式 106
7.5. 时钟配置 116
8. 时钟和电源 117
8.1. Clock Manage Unit (CMU) 117
8.1.1. 特性说明 117
8.1.2. 原理框图 118
8.1.3. 功能描述 118
8.1.4. 寄存器列表 124
8.1.5. 寄存器描述 126
8.2. System Configuration (SYSCFG) 166
8.2.1. 特性说明 167
8.2.2. 原理框图 167
8.2.3. 功能描述 168
8.2.4. 寄存器列表 168
8.2.5. 寄存器描述 169
8.3. Warm Reset Information (WRI) 179
8.3.1. 特性说明 179
8.3.2. 原理框图 180
8.3.3. 功能描述 180
8.3.4. 寄存器列表 180
8.3.5. 寄存器描述 181
9. 存储 184
9.1. Secure Digital Host Controller (SDMC) 184
9.1.1. 功能特性 184
9.1.2. 原理框图 185
9.1.3. 功能描述 185
9.1.4. 编程指南 196
9.1.5. 寄存器列表 200
9.1.6. 寄存器描述 202
9.2. eXpanded Serial Peripheral Interface (xSPI) 227
9.2.1. 功能特性 227
9.2.2. 原理框图 228
9.2.3. 功能描述 228
9.2.4. 寄存器列表 248
9.2.5. 寄存器描述 250
10. 多媒体 292
10.1. Display Engine (DE) 292
10.1.1. 特性说明 292
10.1.2. 原理框图 292
10.1.3. 功能描述 293
10.1.4. 寄存器列表 295
10.1.5. 寄存器描述 298
10.2. Graphics Engine (GE) 312
10.2.1. 特性说明 313
10.2.2. 原理框图 313
10.2.3. 功能描述 313

ArtInChip

10.2.4. 寄存器列表 323
10.2.5. 寄存器描述 325
10.3. Video Engine (VE) 345
10.3.1. 特性说明 345
10.3.2. JPEG 模块 346
10.3.3. PNG 模块 349
10.3.4. 寄存器列表 355
10.3.5. 寄存器描述 359
10.4. LCD 393
10.4.1. 特性说明 393
10.4.2. 原理框图 394
10.4.3. PRGB 接口时序 395
10.4.4. SRGB 接口时序 398
10.4.5. I8080 接口时序 399
10.4.6. 6800 接口时序 400
10.4.7. SPI 接口时序 401
10.4.8. LCD 模式切换功能 406
10.4.9. 像素时钟 (pixelclk) 与串行时钟 (sclk) 比例关系 406
10.4.10. 寄存器列表 407
10.4.11. 寄存器描述 409
10.5. Audio 429
10.5.1. 特性说明 430
10.5.2. 原理框图 430
10.5.3. 功能描述 430
10.5.4. 编程指南 433
10.5.5. 寄存器列表 434
10.5.6. 寄存器描述 435
11. 计时器 448
11.1. Generic Timer Controller (GTC) 448
11.1.1. 特性说明 448
11.1.2. 原理框图 448
11.1.3. 寄存器列表 449
11.1.4. 寄存器描述 449
11.2. Watch Dog (WDOG) 453
11.2.1. 特性说明 453
11.2.2. 原理框图 453
11.2.3. 功能描述 453
11.2.4. 寄存器列表 456
11.2.5. 寄存器描述 456
11.3. Pulse Width Modulation (PWM) 459
11.3.1. 特性说明 459
11.3.2. 原理框图 461
11.3.3. 功能描述 462
11.3.4. 寄存器列表 468
11.3.5. 寄存器描述 469
12. 接口 483
12.1. Quad Serial Peripheral Interface (QSPI) 483
12.1.1. 功能特性 483
12.1.2. 原理框图 484
12.1.3. 功能描述 484
12.1.4. 编程指南 494
12.1.5. 寄存器列表 500
12.1.6. 寄存器描述 502
12.2. Universal Asynchronous Receiver/Transmitter (UART) 529
12.2.1. 特性说明 529
12.2.2. 原理框图 530
12.2.3. 功能描述 530
12.2.4. 编程指南 537
12.2.5. 寄存器列表 538
12.2.6. 寄存器描述 540
12.3. Inter Integrated Circuit (I2C) 560

AllWinner

12.3.1. 特性说明 560
12.3.2. 原理框图 561
12.3.3. 功能描述 561
12.3.4. 编程指南 570
12.3.5. 寄存器列表 571
12.3.6. 寄存器描述 572
12.4. Controller Area Network (CAN) 593
12.4.1. 特性说明 593
12.4.2. 原理框图 593
12.4.3. 典型应用 594
12.4.4. 寄存器列表 606
12.4.5. 寄存器描述 608
12.5. Consumer Infrared Radiation (CIR) 627
12.5.1. 特性说明 627
12.5.2. 原理框图 628
12.5.3. 功能描述 628
12.5.4. 编程指南 634
12.5.5. 寄存器列表 636
12.5.6. 寄存器描述 636
12.6. General Purpose Input Output (GPIO) 645
12.6.1. 特性说明 645
12.6.2. 原理框图 646
12.6.3. 功能描述 646
12.6.4. 寄存器列表 648
12.6.5. 寄存器描述 649
13. 模拟 655
13.1. ADC Interface Module (ADCIM) 655
13.1.1. 特性说明 655
13.1.2. 原理框图 655
13.1.3. 功能描述 656
13.1.4. 编程指南 660
13.1.5. 寄存器列表 660
13.1.6. 寄存器描述 661
13.2. General Purpose Analog Input (GPAI) 663
13.2.1. 特性说明 663
13.2.2. 原理框图 664
13.2.3. 功能描述 664
13.2.4. 编程指南 667
13.2.5. 寄存器列表 669
13.2.6. 寄存器描述 669
13.3. Resistance Touch Panel (RTP) 681
13.3.1. 特性说明 682
13.3.2. 原理框图 682
13.3.3. 功能描述 683
13.3.4. 寄存器列表 693
13.3.5. 寄存器描述 694
13.4. Thermal Sensor (THS) 708
13.4.1. 特性说明 709
13.4.2. 原理框图 709
13.4.3. 功能描述 709
13.4.4. 编程指南 711
13.4.5. 寄存器列表 712
13.4.6. 寄存器描述 713

ArtInChip

1. 文档说明

1.1. 适用范围

用户手册适用于板级产品设计人员和产品软件开发商。 本手册假定读者具有计算机背景工程和/或软件工程，并了解数字化的概念、系统设计、 微处理器架构、输入/输出 (I/O) 设备、行业标准通信和设备接口协议。

1.2. 内容组成

用户手册在系统级别描述了芯片，并提供了架构概述。 它还描述了系统内存映射、系统级中断事件、外部引脚和引脚复用、外部存储器、 系统调试、系统启动、多媒体子系统、电源管理和系统安全。

1.3. 寄存器属性

术语	定义/寄存器类型
RO	只读（Read-only）
WO	只写（Write-only）
R/W	可读可写（Read/Write）
R/W1C	可读并写‘1’清零（Read/Write-‘1’-clear），常见于中断标志寄存器
R/W1T	可读并写‘1’反转（Read/Write-‘1’-toggle）
R/WAC	可读并写‘1’后自动清零（Read/Write-auto-clear），常见于模块的软复位寄存器
-	表示保留（Reserved），通常是不可写并且读返回‘0’
‘u’	寄存器值不确定，例如， 0x987a_bu65

1.4. 数值定义

类型	单位	大小
容量	1K	1024
	1M	1,048,576
	1G	1,073,741,824

Copyright © 2023-2025 ArtInChip Technology Co., Ltd.
All rights reserved

类型	单位	大小
频率	1K	1000
	1M	1000,000
	1G	1000,000,000

1.5. 术语、缩写词

术语	全称	中文说明
ADC	Analog Digital Converter	模拟数字转换器
ADCIM	ADC Interface Management	模数转换管理模块
AES	Advanced Encryption Standard	一种对称分组密钥算法
API	Application Programming Interface	应用程序接口
BCLK	Bit Clock	比特时钟，每个时钟周期传输 1bit 音频数据
blit	Bit Block Transfer	位块搬移
BROM	Boot ROM	固化在芯片中的启动程序
CAN	Controller Area Network	控制器局域网络
CBC	Cipher Block Chaining	密码分组链模式
CE	Crypto Engine	硬件加解密算法引擎
CIR	consumer IR	消费者红外
CMA	Contiguous Memory Allocator	连续内存分配器
CMD	Command	SD/eMMC 协议的命令
CMU	Clock Manage Unit	时钟管理单元
CPHA	Clock Phase	时钟相位
CPOL	Clock Polarity	时钟极性
CTR	Counter	计数器模式

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

术语	全称	中文说明
CTS	Ciphertext Stealing	一种对称分组密码算法模式
DE	Display Engine	显示引擎
DES	Data Encryption Standard	一种对称分组密码算法
DF	Decimation Filter	抽值滤波器
DI	Display Interface	显示接口，包括 RGB/ LVDS/ DSI 等
DMA	Direct Memory Access	直接存储器访问
DMIC	Digital Microphone	数字麦克风
DRQ	DMA Request	指 DMA 请求的端口号
Dual SPI	Dual Serial Peripheral Interface	双路 SPI
DVP	Digital Video Port	数字视频接收端口
ECB	Electronic Code Book	电子密码本模式
EFF	Extended Frame Format	扩展帧
eFuse	Electric Fuse	电子熔丝
eMMC	Embedded Multi Media Card	内嵌（在板卡上）式多媒体卡
ESC	EtherCAT Slave Controller	一种实时工业以太网从机控制器
GE	Graphics Engine	2D 图形加速引擎
GIC	Generic Interrupt Controller	通用中断控制器
GPAI	General Purpose Analog Input Interface	通用模拟输入接口
GPIO	General Purpose Input Output	通用输入输出
HUK	Hardware Unique Key	硬件唯一密钥
I8080	Intel 8080 architecture	因特尔 8080 总线标准
ISP	Image Signal Processing	图像信号处理
LRCK	left right clock	左右声道时钟

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

术语	全称	中文说明
LVDS	Low Voltage Differential Signaling	低压差分信号的显示接口标准
MAC	Media Access Control	媒体访问控制，Ethernet 中的一个组成模块
MD5	Message Digest Algorithm	消息摘要算法
MDIO	Management Data Input/Output	PHY 的管理接口，用来读/写 PHY 的寄存器
MIPI DSI	Mipi Display Serial Interface	Mipi 组织定义的一个显示接口标准
MMC	Multi Media Card	多媒体卡
MPP	Media Process Platform	通用多媒体处理软件平台
MPP	Media Prosess Platform	多媒体处理平台
NEC	–	一种红外编码协议
PBUS	Parallel Bus	并行总线
PCM	Pulse Code Modulation	脉冲编码调制
PDM	Pulse Density Modulation	脉冲密度调制
PHY	Physic	物理接口收发器
PLL	Phase Locked Loop	锁相环时钟
PNK	Part Number Key	芯片型号密钥
PSK	Partner Secret Key	合作伙伴密钥
PWM	Pulse-Width Modulation	脉冲宽度控制，简称脉宽调制
QSPI	Quad Serial Peripheral Interface	四路 SPI，又称 QSPI
RC5	–	一种红外编码协议
RC6	–	一种红外编码协议
RGB	Red Green Blue	按红绿蓝三原色编码的显示接口标准
RLC	Run-Length Code	游程编码
RMII	Reduced MII	简化的媒体独立接口，使用 2 根线来传输数据

Copyright © 2023-2025 ArtInChip Technology Co., Ltd.
All rights reserved

术语	全称	中文说明
RSA	Rivest–Shamir–Adleman	一种非对称密钥算法
RTC	Real Time Clock	实时时钟
RTR	Remote Transmission Request	远程请求帧
SD	Secure Digital	Flash 存储卡的一种标准，即常见的 SD 卡
SDCard	Secure Digital Memory Card	安全数码卡，同“SD 卡”
SDIO	Secure Digital Input and Output	安全数字输入输出接口
SDM	Sigma Delta Modulation	西格玛三角洲调解器
SDMC	SDCard & SDIO & eMMC Host Contollor	SD 卡/eMMC 主控制器
SHA	Secure Hash Algorithm	安全散列算法
SID	Secure ID	安全 ID 模块
SPI	Serial Peripheral Interface	串行外设接口
SSK	Symmetric Secret Key	对称密钥
TBC	Time Base Counter	时基计数器，用于产生锯齿波
TRNG	True Random Number Generator	真随机数生成器
TSensor/TSen	Thermal Sensor	温度传感器
V4L2	Video For Linux Two	Linux 中的视频框架第二版
VBI	Vertical Blanking Interval	垂直消隐期
VE	Video Engine	视频加速引擎
XTS	XEX Tweakable Block Ciphertext with Ciphertext Stealing	一种对称分组密码算法模式

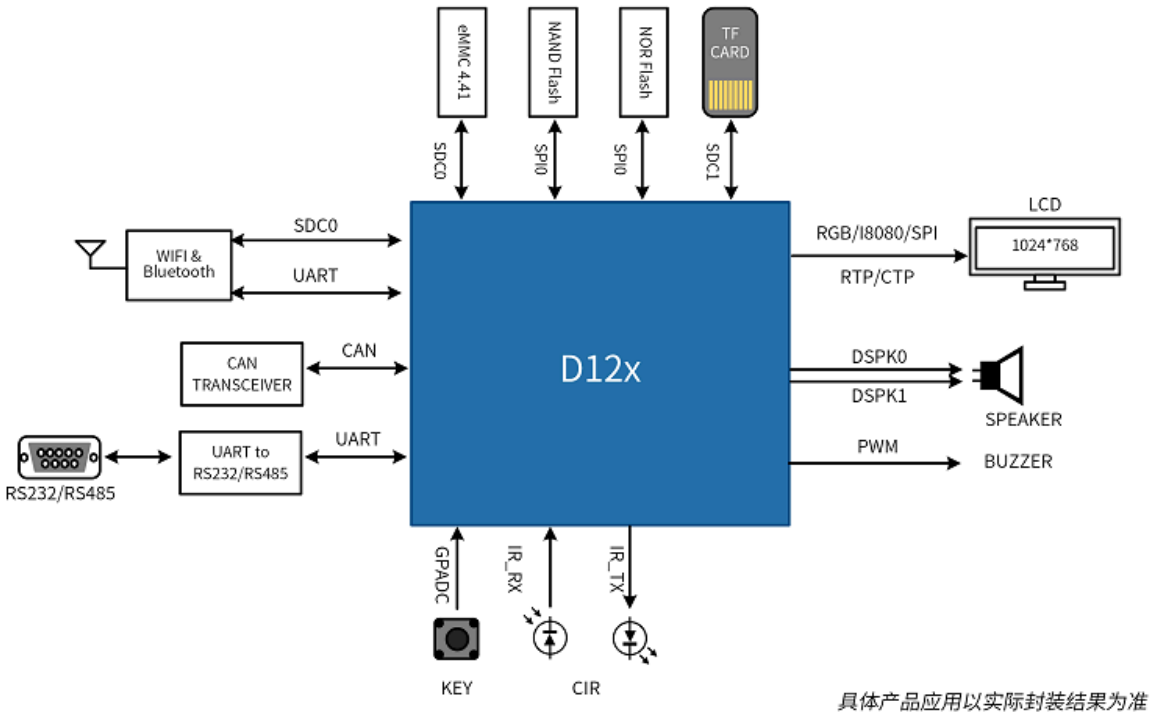
Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

2. 简介

D12x 是一款基于 RISC-V 的高性能、国产自主、工业级高清显示与智能控制 MCU，配备强大的 2D 图形加速处理器、PNG/ JPEG 解码引擎、丰富的接口，支持工业宽温，具有高可靠性、高开放性，可广泛应用于工业自动化控制、串口屏等智慧工业和智慧家居领域。

Display out	Processor	Memory
RGB 1024x768	RISC-V 32-bit CPU	128/ 64/ 32 Mb PSRAM
I8080	I-Cache 32KB	eMMC I/F
QSPI	D-Cache 16KB	QSPI I/F

Multimedia	System	Connectivity	
Display Engine 1024x768@60fps	SPI ENC	DSPK 2ch	SDMC x 2
2D Graphics Engine	DMA	UART x 4	QSPI x 2
JPEG Decoder	Temp. Sensor	CAN x 2	CIR
PNG Decoder	eFuse	RTP	I2C x 2
	Watchdog	ADC 6ch	PWM 4ch



具体产品应用以实际封装结果为准

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

2.1. 功能特性

- **CPU 内核**
  - 单核 E907，RV32IMAFC 指令架构，400 MHz@1.1V
  - 一级指令缓存 32KB，一级数据缓存 16KB
  - 物理内存保护 (PMP)
  - 核内中断 CLINT 和中断控制器 CLIC，支持中断嵌套
  - 机器模式和用户模式
  - 标准 2 线 JTAG 调试接口
- **系统启动**
  - 默认按 SD Card (SDMC1) → NOR (SPI0) → NAND (SPI0) → eMMC (SDMC0) 顺序扫描启动
  - 可通过烧写 eFuse 改变并固定启动介质
- **系统安全**
  - SPI 总线加密模块 (SPI ENC)，支持 SPI NAND/ SPI NOR 在线加解密
  - 数据校验引擎 (DCE)，支持 CRC-32 多项式和累加和两种方式
  - 内置 eFuse 512 bits ，其中 128 bits 供用户自定义使用
- **片内存储**
  - 32 KB BROM
  - 32 KB SRAM
  - PSRAM 以下规格可选：
    - 128 Mb，16-bit 位宽，最高频率 200 MHz DDR
    - 64 Mb，16-bit 位宽，最高频率 200 MHz DDR
    - 32 Mb，8-bit 位宽，最高频率 200 MHz DDR
  - PSRAM 时钟支持展频功能
- **存储接口**
  - QSPI 支持 SPI NOR Flash / SPI NAND Flash
    - 支持单/ 双/ 四数据线
    - 支持最大速率 SDR 100 MHz
  - eMMC 4.41/ SD 2.0/ SDIO 2.0，总共两套
    - eMMC 4 数据线，支持 SDR25/ SDR50/ DDR50 模式
    - SD 卡，支持 SDR25/ SDR50 模式
    - 支持最大速率 DDR 50 MHz，仅支持 3.3 V IO 电压
- **图像引擎**
  - DE 显示引擎：
    - 支持一个 UI 图层，最高性能 1024x768@60fps
    - 支持抖动、伽马及色彩矩阵调整
  - GE 图像引擎：
    - 支持 2D 图形加速，最大性能 1080P@60fps
    - 支持水平和垂直翻转，90/ 180/ 270 度旋转
    - 支持 RGB 格式任意角度旋转和不同扫描顺序
    - 支持 1/16x ~ 16x 缩放，采用 bilinear 滤波算法
    - 支持命令队列
  - VE 视频编解码：
    - JPEG 解码器，最高性能 720P@60fps
    - PNG 解码器，最高性能 720P@60fps
- **显示接口**

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 支持 24-bit 并口 RGB，最高性能 1024 x 768@60fps
- 支持 SRGB/ I8080/ QSPI 屏接口
- 支持展频功能

• 音频接口

- 两路数字 PWM 输出 (DSPK)，支持两路单端输出左右声道，支持一路差分输出单声道

• 通用接口

- 两路 QSPI，支持 单/ 双/ 四数据线，支持 Master
- 四路 UART，支持 2 线/ 3 线/ 4 线接口，兼容工业标准 16550
- 两路 I2C，支持 7 bits 和 10 bits 寻址，最高速率 400 Kb/s
- 两路 CAN，支持 CAN2.0A 和 CAN2.0B，可编程通信速率最高 1 Mbps
- 一组 CIR，支持红外输入和红外输出
- 五组 GPIO 总共 60 个 IO，支持每个 IO 独立配置：
  - 可选无上下拉/上拉 33 KΩ/ 下拉 33 KΩ
  - 输出驱动八个档位可调
  - 输入支持二级去抖和中断
  - 数据输出寄存器支持位操作

• 计数器

- GTC 通用计时器
  - 支持 52 位计时器，提供系统心跳时钟，计时周期大于 35 年
  - 支持调试模式下可配置为暂停计时或继续计时
- WDOG 看门狗
  - 支持中断和复位，超时时间 1 毫秒 ~ 37 小时可配置
  - 支持调试模式下可配置为暂停计时或继续计时
  - 支持硬件写保护机制
- PWM
  - 内置 16-bit 计数器
  - 最大可支持四路独立 PWM 或两路互补 PWM

• 模拟

- 内置 6 通道 12-bit GPADC，采样速率为 1 Msps
- 集成 RTP 电阻触摸屏接口，若不使用 RTP 功能，可当 ADC 功能使用

• 时钟和电源管理

- 芯片时钟源
  - 支持无晶体方案，使用内置 OSC24M，精度 ±2%
  - 支持外挂晶体方案，使用外置 24M 晶体，精度取决于晶体
- CMU 内置四个 PLL：
  - PLL_INT0 用于 CPU 单独使用
  - PLL_INT1 用于总线，内部模块，及低速接口模块使用
  - PLL_FRA0 用于存储接口模块使用，支持展频
  - PLL_FRA2 用于屏输出模块使用，支持展频
- SYSCFG 内置三个 LDO：

Copyright © 2023-2025 ArtInChip Technology Co., Ltd.
All rights reserved

- LDO25 (2.5 V 100 mA)，用于系统复位启动、ADC 供电、eFuse 供电
- LDO18 (1.8 V 100 mA)，可用于 PSRAM IO 和 PSRAM 颗粒供电
- LDO1x (1.0 ~ 1.375 V 200 mA，每档 25 mV)，可用于 VDD_SYS 供电

◦ 内置 THS 温度传感器，支持高低温中断报警和过温复位芯片

ArtInChip

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

3. D12x 地址映射

表 3–1 D12x 地址映射

模块名称	起始地址	结束地址	空间大小
DMA	0x10000000	0x1000FFFF	64 KB
DCE	0x10010000	0x10010FFF	4 KB
XSPI	0x10300000	0x10300FFF	4 KB
SPI0	0x10400000	0x1040FFFF	64 KB
SPI1	0x10410000	0x1041FFFF	64 KB
SDMC0	0x10440000	0x1044FFFF	64 KB
SDMC1	0x10450000	0x1045FFFF	64 KB
AHBCFG	0x104FE000	0x104FEFFF	4 KB
SYSCFG	0x18000000	0x18000FFF	4 KB
CMU	0x18020000	0x18020FFF	64 KB
SPI_ENC	0x18100000	0x18100FFF	4 KB
AXICFG	0x184FE000	0x184FEFFF	4 KB
MTOP	0x184FF000	0x184FFFFF	4 KB
AUDIO	0x18610000	0x18610FFF	4 KB
GPIO	0x18700000	0x18700FFF	4 KB
UART0	0x18710000	0x18710FFF	4 KB
UART1	0x18711000	0x18711FFF	4 KB
UART2	0x18712000	0x18712FFF	4 KB
UART3	0x18713000	0x18713FFF	4 KB
LCD	0x18800000	0x18800FFF	4 KB

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 3–1 D12x 地址映射 (续)

模块名称	起始地址	结束地址	空间大小
DE	0x18A00000	0x18AFFFFF	1 MB
GE	0x18B00000	0x18BFFFFF	1 MB
VE	0x18C00000	0x18CFFFFF	1 MB
WDOG	0x19000000	0x19000FFF	4 KB
WRI	0x1900F000	0x1900FFFF	4 KB
SID	0x19010000	0x19010FFF	4 KB
GTC	0x19050000	0x19051FFF	8KB
I2C0	0x19220000	0x19220FFF	4 KB
I2C1	0x19221000	0x19221FFF	4 KB
CAN0	0x19230000	0x19230FFF	4 KB
CAN1	0x19231000	0x19231FFF	4 KB
PWM	0x19240000	0x19240FFF	4 KB
ADCIM	0x19250000	0x19250FFF	4 KB
GPAI	0x19251000	0x19251FFF	4 KB
RTP	0x19252000	0x19252FFF	4 KB
THS	0x19253000	0x19253FFF	4 KB
CIR	0x19260000	0x19260FFF	4 KB
E907_CLINT	0x20000000	0x2000FFFF	64 KB
E907_CLIC	0x20800000	0x2080FFFF	64 KB
BROM	0x30000000	0x30007FFF	32 KB
SRAM_SYS	0x30040000	0x30047FFF	32 KB
SRAM_VE	0x30048000	0x3004BFFF	16 KB

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 3–1 D12x 地址映射 (续)

模块名称	起始地址	结束地址	空间大小
PSRAM	0x40000000	0x5FFFFFFF	512 MB
FLASH_XIP	0x60000000	0x7FFFFFFF	512 MB
RESERVED	0x80000000	0xFFFFFFFF	2 GB

ArtInChip

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

4. 引脚复用

4.1. D12x 功能复用

表 4–1 D12x 功能复用表

引脚	功能 2	功能 3	功能 4	功能 5	功能 6	功能 8
PA0	GPADC0	IR_TX	I2C0_SCL	UART0_TX	–	CPU_NMI
PA1	GPADC1	IR_RX	I2C0_SDA	UART0_RX	–	DE_TE
PA2	GPADC2	CAN1_TX	I2C1_SCL	UART1_TX	–	UART2_CTS
PA3	GPADC3	CAN1_RX	I2C1_SDA	UART1_RX	–	UART2_RTS
PA4	GPADC4	–	CAN0_TX	UART2_TX	–	–
PA5	GPADC5	–	CAN0_RX	UART2_RX	–	–
PA8	RTP_XP	–	I2C0_SCL	–	–	–
PA9	RTP_YP	–	I2C0_SDA	–	–	–
PA10	RTP_XN	IR_RX	–	–	–	JTAG_MS
PA11	RTP_YN	IR_TX	–	–	–	JTAG_CK
PB0	SPI0_WP	SPI1_WP	–	UART0_TX	–	–
PB1	SPI0_MISO	SPI1_MISO	–	UART2_TX	–	–
PB2	SPI0_CS0	SPI1_CS	–	UART2_RX	–	–
PB3	SPI0_HOLD	SPI1_HOLD	–	UART0_RX	–	–
PB4	SPI0_CLK	SPI1_CLK	–	UART2_RTS	–	–
PB5	SPI0_MOSI	SPI1_MOSI	–	UART0_RTS	UART2_CTS	–
PB6	SDC0_CMD	SPI1_CS	–	UART1_TX	–	–
PB7	SDC0_CLK	SPI1_MISO	–	UART1_RX	–	–
PB8	SDC0_D3	SPI1_MOSI	–	UART1_RTS	UART3_CTS	–
PB9	SDC0_D0	SPI1_CLK	–	UART3_RTS	–	–
PB10	SDC0_D1	SPI1_HOLD	–	UART3_TX	–	–
PB11	SDC0_D2	SPI1_WP	–	UART3_RX	–	–
PC0	SDC1_D1	–	I2C0_SCL	UART3_RTS	–	JTAG_MS
PC1	SDC1_D0	–	–	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 4–1 D12x 功能复用表 (续)

引脚	功能 2	功能 3	功能 4	功能 5	功能 6	功能 8
PC2	SDC1_CLK	–	–	–	–	UART0_TX
PC3	SDC1_CMD	–	–	–	–	–
PC4	SDC1_D3	PWM0_A	I2C1_SCL	UART3_TX	–	UART0_RX
PC5	SDC1_D2	PWM0_B	I2C1_SDA	UART3_RX	–	JTAG_CK
PC6	SDC1_DET	PWM1_A	I2C0_SDA	UART3_CTS	DE_TE	IR_RX
PC7	–	PWM1_B	–	–	–	IR_TX
PD0	LCD_D0	CAN0_TX	I2C0_SCL	UART0_TX	–	–
PD1	LCD_D1	CAN0_RX	I2C0_SDA	UART0_RX	–	–
PD2	LCD_D2	CAN1_TX	I2C1_SCL	UART1_TX	–	–
PD3	LCD_D3	CAN1_RX	I2C1_SDA	UART1_RX	–	–
PD4	LCD_D4	–	I2C1_SCL	UART2_TX	–	–
PD5	LCD_D5	–	I2C1_SDA	UART2_RX	–	–
PD6	LCD_D6	–	PWM0_A	DSPK0	–	–
PD7	LCD_D7	–	PWM0_B	DSPK1	–	–
PD8	LCD_D8	–	PWM1_A	–	–	–
PD9	LCD_D9	–	–	–	–	–
PD10	LCD_D10	–	–	–	–	–
PD11	LCD_D11	–	–	–	–	–
PD12	LCD_D12	–	–	–	–	–
PD13	LCD_D13	–	–	–	–	–
PD14	LCD_D14	–	–	–	–	–
PD15	LCD_D15	–	–	–	–	–
PD16	LCD_D16	–	–	–	–	–
PD17	LCD_D17	–	–	–	–	–
PD18	LCD_D18	–	–	–	–	–
PD19	LCD_D19	–	–	–	–	–
PD20	LCD_D20	–	–	–	–	–
PD21	LCD_D21	–	–	–	–	–
PD22	LCD_D22	–	–	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 4–1 D12x 功能复用表 (续)

引脚	功能 2	功能 3	功能 4	功能 5	功能 6	功能 8
PD23	LCD_D23	–	–	–	–	–
PD24	LCD_DCLK	–	–	–	–	–
PD25	LCD_HS	–	–	–	PWM0_B	–
PD26	LCD_VS	–	–	–	PWM1_A	–
PD27	LCD_DE	–	–	–	PWM1_B	–
PE12	–	PWM1_B	–	DSPK1	–	–
PE13	–	PWM0_A	–	DSPK0	–	–

4.2. D12x QFN68 封装引脚说明

表 4–2 D12x QFN68 封装引脚说明

引脚	定义	类型	功能	备注说明
SYSTEM				
52	RESETN	INPUT	系统复位	内置约 30 KΩ 上拉电阻和去抖滤波，不使用可直接悬空，若外挂电容建议不超过 4.7 uF。
POWER				
12, 29, 54	VCC33_IO	POWER	芯片 IO 电压	3.3 V 供电
55	LDO25	POWER	内置 LDO 输出	内部模拟模块使用，外部接 1 uF 旁路电容。
28	LDO18	POWER	内置 LDO 输出	供内部 PSRAM 使用，若使用需做好芯片散热，外部接 1 uF 旁路电容。
27, 53	VDD11_SYS	POWER	芯片 Core 电压	1.1 V 供电，若使用内置 LDO1x，必须做好芯片散热，53 引脚为 LDO1x 输出。
69	GND	POWER	–	GND 铜皮全连接，需多加过孔散热。

4.3. D121 QFN68 封装功能复用表

表 4–3 D121 QFN68 封装功能复用表

引脚	功能 1	功能 2	功能 3	功能 4	功能 5	功能 6	功能 8
GPIO A							
56	PA0	GPADC0	IR_TX	I2C0_SCL	UART0_TX	–	CPU_NMI
57	PA1	GPADC1	IR_RX	I2C0_SDA	UART0_RX	–	DE_TE
58	PA2	GPADC2	–	I2C1_SCL	UART1_TX	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 4–3 D121 QFN68 封装功能复用表 (续)

引脚	功能 1	功能 2	功能 3	功能 4	功能 5	功能 6	功能 8
59	PA3	GPADC3	–	I2C1_SDA	UART1_RX	–	–
60	PA4	GPADC4	–	–	UART2_TX	–	–
61	PA5	GPADC5	–	–	UART2_RX	–	–
62	PA8	RTP_XP	–	I2C0_SCL	–	–	–
63	PA9	RTP_YP	–	I2C0_SDA	–	–	–
64	PA10	RTP_XN	IR_RX	–	–	–	JTAG_MS
65	PA11	RTP_YN	IR_TX	–	–	–	JTAG_CK
GPIO B							
40	PB0	SPI0_WP	SPI1_WP	–	UART0_TX	–	–
41	PB1	SPI0_MISO	SPI1_MISO	–	UART2_TX	–	–
42	PB2	SPI0_CS0	SPI1_CS	–	UART2_RX	–	–
43	PB3	SPI0_HOLD	SPI1_HOLD	–	UART0_RX	–	–
44	PB4	SPI0_CLK	SPI1_CLK	–	UART2_RTS	–	–
45	PB5	SPI0_MOSI	SPI1_MOSI	–	UART0_RTS	UART2_CTS	–
46	PB6	SDC0_CMD	SPI1_CS	–	UART1_TX	–	–
47	PB7	SDC0_CLK	SPI1_MISO	–	UART1_RX	–	–
48	PB8	SDC0_D3	SPI1_MOSI	–	UART1_RTS	UART3_CTS	–
49	PB9	SDC0_D0	SPI1_CLK	–	UART3_RTS	–	–
50	PB10	SDC0_D1	SPI1_HOLD	–	UART3_TX	–	–
51	PB11	SDC0_D2	SPI1_WP	–	UART3_RX	–	–
GPIO C							
66	PC0	SDC1_D1	–	I2C0_SCL	UART3_RTS	–	JTAG_MS
67	PC1	SDC1_D0	–	–	–	–	–
68	PC2	SDC1_CLK	–	–	–	–	UART0_TX
1	PC3	SDC1_CMD	–	–	–	–	–
2	PC4	SDC1_D3	PWM0_A	I2C1_SCL	UART3_TX	–	UART0_RX
3	PC5	SDC1_D2	PWM0_B	I2C1_SDA	UART3_RX	–	JTAG_CK
4	PC6	SDC1_DET	PWM1_A	I2C0_SDA	UART3_CTS	DE_TE	IR_RX
5	PC7	–	PWM1_B	–	–	–	IR_TX

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 4–3 D121 QFN68 封装功能复用表 (续)

引脚	功能 1	功能 2	功能 3	功能 4	功能 5	功能 6	功能 8
GPIO D							
37	PD0	LCD_D0	–	I2C0_SCL	UART0_TX	–	–
36	PD1	LCD_D1	–	I2C0_SDA	UART0_RX	–	–
35	PD2	LCD_D2	–	I2C1_SCL	UART1_TX	–	–
34	PD3	LCD_D3	–	I2C1_SDA	UART1_RX	–	–
33	PD4	LCD_D4	–	I2C1_SCL	UART2_TX	–	–
32	PD5	LCD_D5	–	I2C1_SDA	UART2_RX	–	–
31	PD6	LCD_D6	–	PWM0_A	DSPK0	–	–
30	PD7	LCD_D7	–	PWM0_B	DSPK1	–	–
26	PD8	LCD_D8	–	PWM1_A	–	–	–
25	PD9	LCD_D9	–	–	–	–	–
24	PD10	LCD_D10	–	–	–	–	–
23	PD11	LCD_D11	–	–	–	–	–
22	PD12	LCD_D12	–	–	–	–	–
21	PD13	LCD_D13	–	–	–	–	–
20	PD14	LCD_D14	–	–	–	–	–
19	PD15	LCD_D15	–	–	–	–	–
18	PD16	LCD_D16	–	–	–	–	–
17	PD17	LCD_D17	–	–	–	–	–
16	PD18	LCD_D18	–	–	–	–	–
15	PD19	LCD_D19	–	–	–	–	–
14	PD20	LCD_D20	–	–	–	–	–
13	PD21	LCD_D21	–	–	–	–	–
11	PD22	LCD_D22	–	–	–	–	–
10	PD23	LCD_D23	–	–	–	–	–
9	PD24	LCD_DCLK	–	–	–	–	–
8	PD25	LCD_HS	–	–	–	PWM0_B	–
7	PD26	LCD_VS	–	–	–	PWM1_A	–
6	PD27	LCD_DE	–	–	–	PWM1_B	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 4–3 D121 QFN68 封装功能复用表 (续)

引脚	功能 1	功能 2	功能 3	功能 4	功能 5	功能 6	功能 8
GPIO E							
38	PE12	–	PWM1_B	–	DSPK1	–	–
39	PE13	–	PWM0_A	–	DSPK0	–	–

4.4. D123 QFN68 封装功能复用表

表 4–4 D123 QFN68 封装功能复用表

引脚	功能 1	功能 2	功能 3	功能 4	功能 5	功能 6	功能 8
GPIO A							
56	PA0	GPADC0	IR_TX	I2C0_SCL	UART0_TX	–	CPU_NMI
57	PA1	GPADC1	IR_RX	I2C0_SDA	UART0_RX	–	DE_TE
58	PLL_XI	–	–	–	–	–	–
59	PLL_XO	–	–	–	–	–	–
60	PA4	GPADC4	–	CAN0_TX	UART2_TX	–	–
61	PA5	GPADC5	–	CAN0_RX	UART2_RX	–	–
62	PA8	RTP_XP	–	I2C0_SCL	–	–	–
63	PA9	RTP_YP	–	I2C0_SDA	–	–	–
64	PA10	RTP_XN	IR_RX	–	–	–	JTAG_MS
65	PA11	RTP_YN	IR_TX	–	–	–	JTAG_CK
GPIO B							
40	PB0	SPI0_WP	SPI1_WP	–	UART0_TX	–	–
41	PB1	SPI0_MISO	SPI1_MISO	–	UART2_TX	–	–
42	PB2	SPI0_CS0	SPI1_CS	–	UART2_RX	–	–
43	PB3	SPI0_HOLD	SPI1_HOLD	–	UART0_RX	–	–
44	PB4	SPI0_CLK	SPI1_CLK	–	UART2_RTS	–	–
45	PB5	SPI0_MOSI	SPI1_MOSI	–	UART0_RTS	UART2_CTS	–
46	PB6	SDC0_CMD	SPI1_CS	–	UART1_TX	–	–
47	PB7	SDC0_CLK	SPI1_MISO	–	UART1_RX	–	–
48	PB8	SDC0_D3	SPI1_MOSI	–	UART1_RTS	UART3_CTS	–
49	PB9	SDC0_D0	SPI1_CLK	–	UART3_RTS	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 4–4 D123 QFN68 封装功能复用表 (续)

引脚	功能 1	功能 2	功能 3	功能 4	功能 5	功能 6	功能 8
50	PB10	SDC0_D1	SPI1_HOLD	–	UART3_TX	–	–
51	PB11	SDC0_D2	SPI1_WP	–	UART3_RX	–	–
GPIO C							
66	PC0	SDC1_D1	–	I2C0_SCL	UART3_RTS	–	JTAG_MS
67	PC1	SDC1_D0	–	–	–	–	–
68	PC2	SDC1_CLK	–	–	–	–	UART0_TX
1	PC3	SDC1_CMD	–	–	–	–	–
2	PC4	SDC1_D3	PWM0_A	I2C1_SCL	UART3_TX	–	UART0_RX
3	PC5	SDC1_D2	PWM0_B	I2C1_SDA	UART3_RX	–	JTAG_CK
4	PC6	SDC1_DET	PWM1_A	I2C0_SDA	UART3_CTS	DE_TE	IR_RX
5	PC7	–	PWM1_B	–	–	–	IR_TX
GPIO D							
37	PD0	LCD_D0	CAN0_TX	I2C0_SCL	UART0_TX	–	–
36	PD1	LCD_D1	CAN0_RX	I2C0_SDA	UART0_RX	–	–
35	PD2	LCD_D2	CAN1_TX	I2C1_SCL	UART1_TX	–	–
34	PD3	LCD_D3	CAN1_RX	I2C1_SDA	UART1_RX	–	–
33	PD4	LCD_D4	–	I2C1_SCL	UART2_TX	–	–
32	PD5	LCD_D5	–	I2C1_SDA	UART2_RX	–	–
31	PD6	LCD_D6	–	PWM0_A	DSPK0	–	–
30	PD7	LCD_D7	–	PWM0_B	DSPK1	–	–
26	PD8	LCD_D8	–	PWM1_A	–	–	–
25	PD9	LCD_D9	–	–	–	–	–
24	PD10	LCD_D10	–	–	–	–	–
23	PD11	LCD_D11	–	–	–	–	–
22	PD12	LCD_D12	–	–	–	–	–
21	PD13	LCD_D13	–	–	–	–	–
20	PD14	LCD_D14	–	–	–	–	–
19	PD15	LCD_D15	–	–	–	–	–
18	PD16	LCD_D16	–	–	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 4–4 D123 QFN68 封装功能复用表 (续)

引脚	功能 1	功能 2	功能 3	功能 4	功能 5	功能 6	功能 8
17	PD17	LCD_D17	–	–	–	–	–
16	PD18	LCD_D18	–	–	–	–	–
15	PD19	LCD_D19	–	–	–	–	–
14	PD20	LCD_D20	–	–	–	–	–
13	PD21	LCD_D21	–	–	–	–	–
11	PD22	LCD_D22	–	–	–	–	–
10	PD23	LCD_D23	–	–	–	–	–
9	PD24	LCD_DCLK	–	–	–	–	–
8	PD25	LCD_HS	–	–	–	PWM0_B	–
7	PD26	LCD_VS	–	–	–	PWM1_A	–
6	PD27	LCD_DE	–	–	–	PWM1_B	–
GPIO E							
38	PE12	–	PWM1_B	–	DSPK1	–	–
39	PE13	–	PWM0_A	–	DSPK0	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

5. 处理器和总线

5.1. Central Process Unit (CPU)

CPU 内核采用了平头哥 E907，是一款基于 RISC-V 指令集的高性能嵌入式处理器。

5.1.1. 特性说明

- RV32IMAFC 指令集
- 五级单发按序执行流水线
- 一级哈佛结构，ICACHE 32 KB，DCACHE 16 KB，缓存行大小为 32 字节
- 物理内存保护 (PMP)
- 核内中断 CLINT 和中断控制器 CLIC，支持中断嵌套
- 支持平头哥 MCU 特性扩展技术，包括中断处理加速技术、MCU 扩展特性
- 机器模式和用户模式
- 标准 2 线 JTAG 调试接口

5.1.2. 中断号

中断号	模块
7	CPU_TIMER
31	DCE
32	DMA
41	SPI_ENC
44	QSPI0
45	QSPI1
46	SDMC0
47	SDMC1
49	XSPI

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

中断号	模块
51	MTOP
54	AUDIO
55	LCD
59	DE
60	GE
61	VE
64	WDOG
68	GPIO_GRP_A
69	GPIO_GRP_B
70	GPIO_GRP_C
71	GPIO_GRP_D
72	GPIO_GRP_E
76	UART0
77	UART1
78	UART2
79	UART3
84	I2C0
85	I2C1
88	CAN0
89	CAN1
90	PWM
92	GPAI
93	RTP

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

中断号	模块
94	THS
95	CIR

5.2. BUS

图 5–1 : 总线框图如下所示：

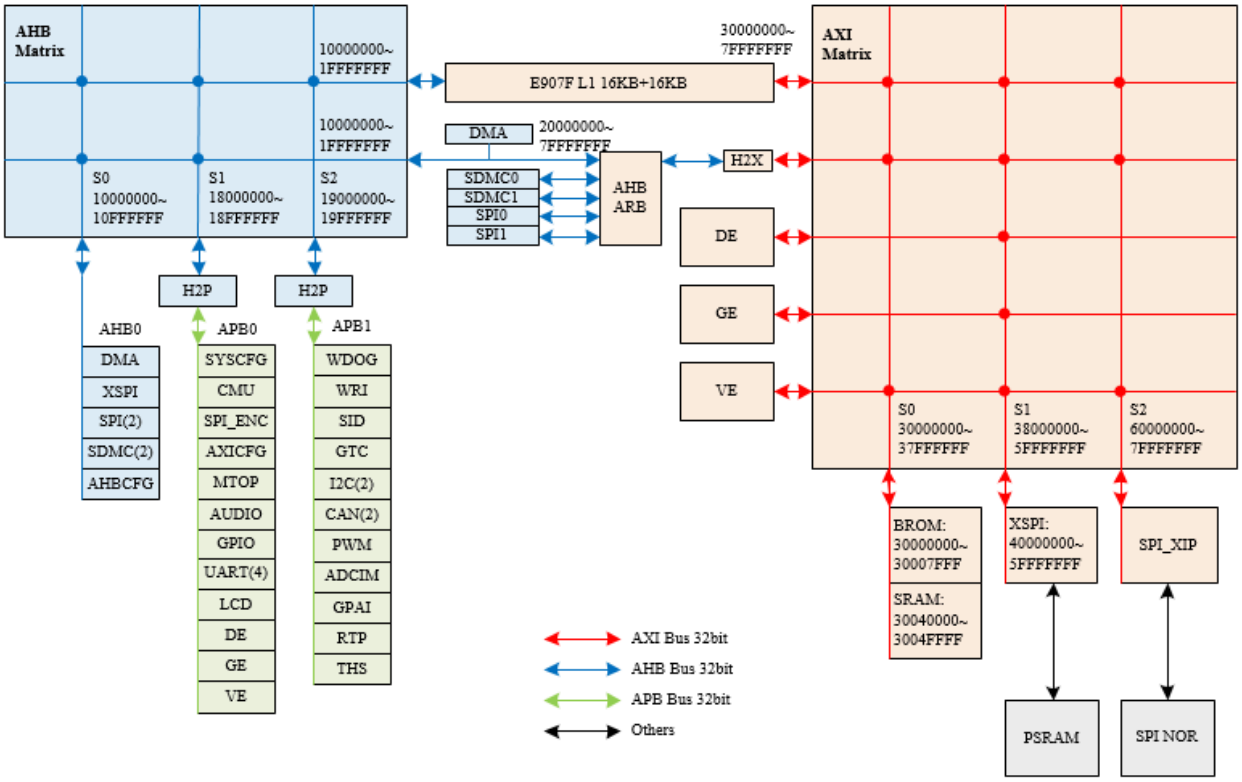


图 5–1 总线框图

总线空间分配及内存属性如下表所示。

表 5–1 总线空间分配及内存属性

地址空间定义				CPU SYSMAP 硬件定义（机器模式内存属性）				
空间段	起始地址	结束地址	大小	变量	ADDR	SO	C	B
RESERVE0	0x00000000	0x0FFFFFFF	256 MB	SYSMAP_BASE_ADDR0	28'h00100	1	0	0
DEVICE	0x10000000	0x1FFFFFFF	256 MB	SYSMAP_BASE_ADDR1	28'h00200	1	0	0
E907	0x20000000	0x2FFFFFFF	256 MB	SYSMAP_BASE_ADDR2	28'h00300	1	0	0
BROM	0x30000000	0x3003FFFF	256 KB	SYSMAP_BASE_ADDR3	28'h30040	0	1	1
SRAM0	0x30040000	0x37FFFFFF	127.7 MB	SYSMAP_BASE_ADDR4	28'h38000	0	1	1

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 5–1 总线空间分配及内存属性 (续)

地址空间定义				CPU SYSMAP 硬件定义（机器模式内存属性）				
空间段	起始地址	结束地址	大小	变量	ADDR	SO	C	B
SRAM1	0x38000000	0x3FFFFFFF	128 MB	SYSMAP_BASE_ADDR5	28'h40000	0	1	1
PSRAM	0x40000000	0x5FFFFFFF	512 MB	SYSMAP_BASE_ADDR6	28'h60000	0	1	1
FLASH	0x60000000	0x7FFFFFFF	512 MB	SYSMAP_BASE_ADDR7	28'h80000	0	1	1

5.3. AXI Configuration (AXICFG)

AXI Configuration (AXICFG) 用于配置 AXI Matrix 主从设备，进行优先级配置。

5.3.1. 特性说明

- 针对存储控制器的 AXI 端口分为多个组 (Group)，每个组内支持多个端口 (Port)
- 支持对每个主设备进行优先级调整

5.3.2. 功能描述

5.3.2.1. 优先级配置

主设备优先级配置列出了 AXI Mattrix 主设备优先级配置推荐。

表 5–2 主设备优先级配置

主设备	优先级
CPU	Int_QoS = 6 (1)
AHB	Int_QoS = a
DE	Ext_QoS = 8 (low) 13 (high)
GE	Int_QoS = 5
VE	Int_QoS = 5

(1) QoS 范围为 0~15，值越大优先级越高

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

5.3.3. 寄存器列表

表 5–3 AXICFG 寄存器列表

偏移地址	缩写	寄存器名称	章节
0x000	AXI_CFG0	AXI 配置 0	0x000+W*0x10+P*0x20(P=0~4) AXI_CFG0
0xFFC	AXICFG_VER	AXICFG 版本	0xFFC AXICFG_VER

5.3.4. 寄存器描述

P 为主设备端口，R 为读写通道，对应的定义如下表：

端口号 P	设备
0	CPU
1	AHB
2	DE
3	GE
4	VE

读写通道 R	通道
0	写通道
1	读通道

5.3.4.1. 0x000+W*0x10+P*0x20(P=0~4) AXI_CFG0

P 为主设备端口号。R=0 写端口，R=1 读端口

默认值：0x00000000			AXI 配置 0 (AXI Configuration0)
位域	类型	默认值	描述
31:28	–	–	–
27:24	R/W	0x0	QOS_VAL QOS 值设置，仅在 QOS_SEL = 1 时才有效
23:17	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			AXI 配置 0 (AXI Configuration0)
位域	类型	默认值	描述
16	R/W	0x0	QOS_SEL QOS 选择 • 0：EXT_QOS，QoS 来源于主设备 • 1：INT_QOS，QoS 来源于 QOS_VAL
15:2	–	–	–
1	R/W	0x0	SLV_READY 从设备就绪使能 • 0：关闭 • 1：开启
0	R/W	0x0	–

5.3.4.2. 0xFFC AXICFG_VER

默认值：0x00000101			AXICFG 版本 (AXICFG Version)
位域	类型	默认值	描述
31:0	RO	0x00000101	VERSION AXICFG 版本，V1.1

5.4. Memory Top (MTOP)

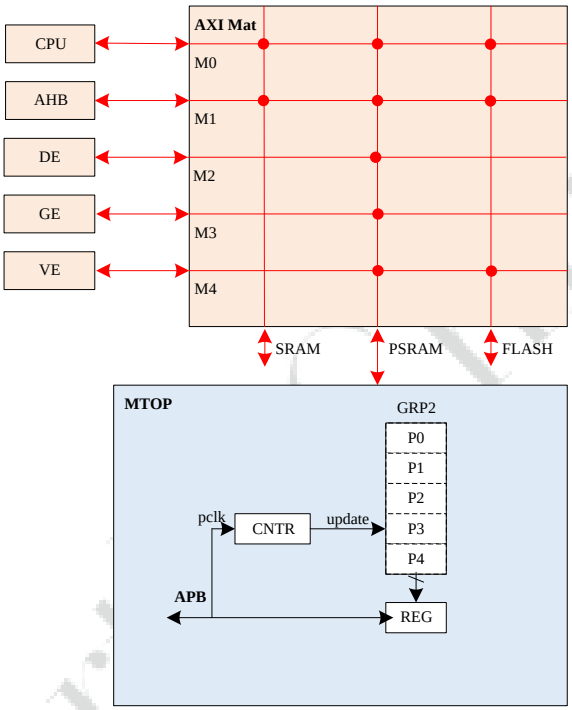
Memory Top (MTOP) 用于针对系统总线端口进行带宽监控。

5.4.1. 特性说明

- 支持自动循环和手动触发两种模式
- 自动循环模式可配置最大周期为 2^{32} 个 APB 时钟周期
- PSRAM 各个端口进行带宽统计
- DRAM 四个 AXI 端口和 AHB 端口进行带宽统计
- 每个 AXI 端口单独统计，读写分开监控

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

5.4.2. 原理框图



5.4.3. 功能描述

5.4.3.1. MTOP 工作模式

MTOP 模块可选两种工作模式：

> **注：**
>
> 关于工作模式的配置，可查看 0x000 MTOP_CTL。

- 循环模式：硬件根据设置的 循环模式周期时间，统计最近一次循环周期内总线上的吞吐量：
  - 通过读取寄存器 0x100+0x100*G+0x20*P AXI_WCNT 可以获取总线写行为的吞吐量。
  - 通过读取寄存器 0x104+0x100*G+0x20*P AXI_RCNT 可以获取总线读行为的吞吐量。

  下图所示为循环模式的工作机制：
  - 蓝色和绿色分别表示相同循环周期的不同时间段。
  - PER_TIME 为设置的循环周期时间。
  - T_{pclk} 表示 APB 总线时钟的周期时间。
  - IRQ_STA 为中断状态寄存器，置位为 1 表示完成一次加载带宽数据。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

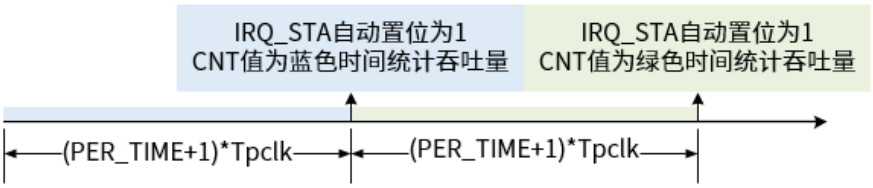


图 5–2 MTOP 循环模式

- 触发模式：需要手动配置 MTOP_TRIG 位域，配置从 0 到 1 启动一次触发。 硬件根据两次触发的间隔，读取下列吞吐量信息：
  - 通过读取寄存器 0x100+0x100*G+0x20*P AXI_WCNT 可以获取总线写行为的吞吐量。
  - 通过读取寄存器 0x104+0x100*G+0x20*P AXI_RCNT 可以获取总线读行为的吞吐量。

  下图所示触发模式工作机制：
  - MTOP_TRIG 为强制触发。
  - 蓝色和绿色分别表示一次强制触发时间段。
  - IRQ_STA 为中断状态寄存器，置位为 1 表示完成一次加载带宽数据。

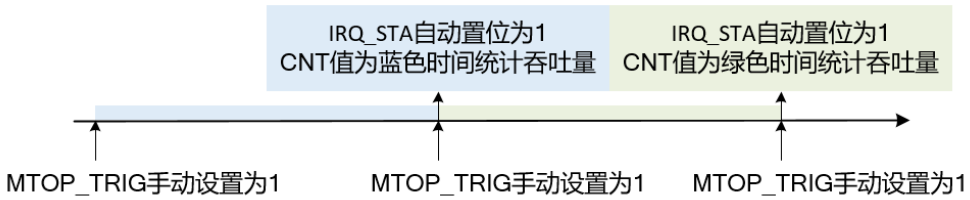


图 5–3 MTOP 触发模式

5.4.4. 寄存器列表

表 5–4 MTOP 寄存器列表

偏移地址	缩写	寄存器名称	章节
0x000	MTOP_CTL	MTOP 控制	0x000 MTOP_CTL
0x004	PER_TIME	周期时间	0x004 PER_TIME
0x008	IRQ_CTL	中断控制	0x008 IRQ_CTL
0x00C	IRQ_STA	中断状态	0x00C IRQ_STA
0x100	AXI_WCNT	AXI 写计数	0x100+0x100*G+0x20*P AXI_WCNT
0x104	AXI_RCNT	AXI 读计数	0x104+0x100*G+0x20*P AXI_RCNT

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 5–4 MTOP 寄存器列表 (续)

偏移地址	缩写	寄存器名称	章节
0xFFC	MTOP_VER	MTOP 版本	0xFFC MTOP_VER

MTOP 统计通道分组及端口如下

分组	偏移地址	G 取值	P 有效值	P 值对应端口
PSRAM	0x300	2	0/ 1/ 2 /3/ 4/ 5	CPU/ DMA /DE/ GE/ VE

5.4.5. 寄存器描述

5.4.5.1. 0x000 MTOP_CTL

默认值：0x00000000			MTOP 控制 (MTOP Control)
位域	类型	默认值	描述
31:30	–	–	–
29	R/W1C	0	MTOP_TRIG 强制触发 • 只在触发模式下有效。 • 配置从 0 到 1 启动一次触发。
28	R/W	0	MTOP_MODE 带宽监控模式 • 0：循环模式。周期循环统计并输出最近周期的统计结果。 • 1：触发模式。需要配置 bit[29] 触发。每触发一次输出最近两次触发的统计结果。
27:1	–	–	–
0	R/W	0	MTOP_EN 带宽监控使能

5.4.5.2. 0x004 PER_TIME

默认值：0x05F510FF			周期时间 (Period Time)
位域	类型	默认值	描述
31:0	R/W	0x05f510ff	PER_TIME

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x05F510FF			周期时间 (Period Time)
位域	类型	默认值	描述
			循环模式的周期时间 以 APB 时钟周期为单位计数，只在循环模式下有效。

5.4.5.3. 0x008 IRQ_CTL

默认值：0x00000000			中断控制 (Interrupt Request Control)
位域	类型	默认值	描述
31:2	–	–	–
1	R/W	0	MTOP_OVERFLOW_IRQ_EN MTOP 溢出中断使能 • 0：关闭 • 1：使能
0	R/W	0	MTOP_UPDATE_IRQ_EN MTOP 更新中断使能 • 0：关闭 • 1：使能

5.4.5.4. 0x00C IRQ_STA

默认值：0x00000000			中断状态 (Interrupt Request Status)
位域	类型	默认值	描述
31:2	–	–	–
1	R/W1C	0	MTOP_OVERFLOW_IRQ_STA MTOP 溢出中断状态 • 0：未发生过中断 • 1：发生过中断
0	R/W1C	0	MTOP_UPDATE_IRQ_STA MTOP 更新中断状态 • 0：未发生过中断 • 1：发生过中断

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			中断状态 (Interrupt Request Status)
位域	类型	默认值	描述
			该寄存器置位表示完成一次加载带宽数据，读取带宽数据前，先查看该位是否置位，在周期模式和触发模式下均有效。

5.4.5.5. 0x100+0x100*G+0x20*P AXI_WCNT

默认值：0x00000000			AXI 写计数 (AXI Write Counter)
位域	类型	默认值	描述
31:0	RO	0x0	AXI_WCNT AXI 写计数值 以字节为单位，每次中断更新最近写周期的吞吐量。

5.4.5.6. 0x104+0x100*G+0x20*P AXI_RCNT

默认值：0x00000000			AXI 读计数 (AXI Read Counter)
位域	类型	默认值	描述
31:0	RO	0x0	AXI_RCNT AXI 读计数值 以字节为单位，每次中断更新最近读周期的吞吐量。

5.4.5.7. 0xFFC MTOP_VER

默认值：0x00000200			MTOP 版本 (MTOP Version)
位域	类型	默认值	描述
31:0	RO	0x00000200	MTOP_VER MTOP 版本，V2.0

5.5. Direct Memory Access (DMA)

Direction memory access (DMA) 模块提供一种高效的数据传输技术，能够在不受 CPU 干预的情况下进行设备间的数据传输，从而减少 CPU 负荷。CPU 使能 DMA 之后，由 DMA 独立自主控制数据传输。此外，DMA 提供的数据交换具有高带宽和低延迟的优点。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

5.5.1. 特性说明

- 四通道 DMA，每通道有 32 个源端和 32 个终端可选
- 采用链表配置方式，寄存器描述通道状态
- 设备位宽支持 8/ 16/ 32/ 64 位，突发长度支持 1/ 4/ 8/ 16 个
- DMA 源端、终端地址 8 bytes 对齐

5.5.2. 原理框图

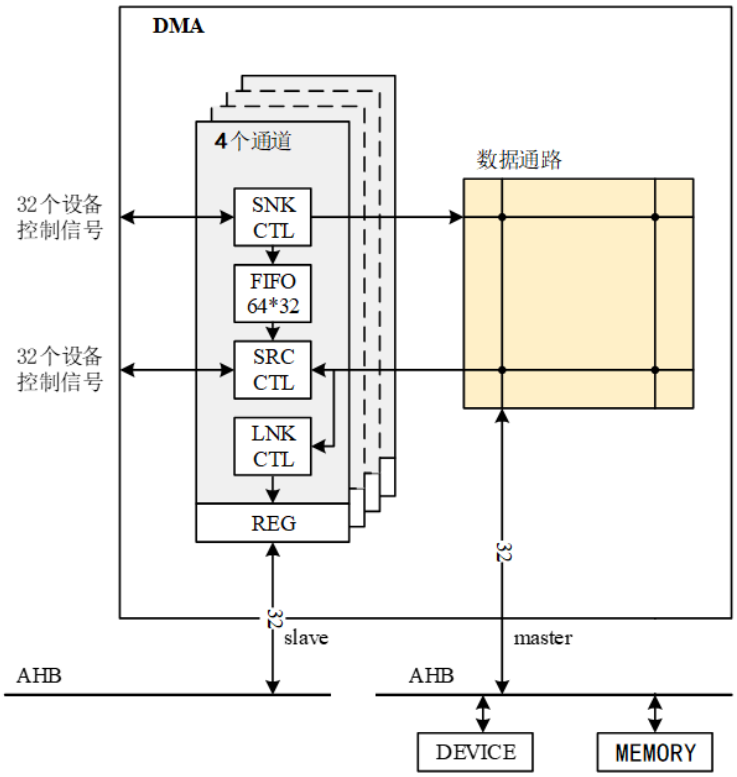


图 5–4 原理框图

5.5.3. 功能描述

5.5.3.1. 设备

DMA 相关设备推荐配置中，单次 DMA 任务传输数据量为位宽 * BURST。

表 5–5 DMA 相关设备推荐配置

存储模块	DMA_ID	位宽（bit）	Burst	地址
SRAM/ DRAM/ XIP	0	32	16	Linear

存储模块与 DMA 采用总线握手方式，无需单独的 REQ/ ACK 机制，DMA 模式选择等待，即只能被动发送或接收数据，若源终端皆为存储模块则实时响应传输。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

读取 XIP 数据时，起始地址和长度需要 64 Byte 对齐。

设备模块	DMA_ID	位宽（bit）	Burst	地址
DMA_ID_SPI0	10	8	16	IO
DMA_ID_SPI1	11	8	16	IO
DMA_ID_AUDIO	14	32	1	IO
DMA_ID_UART0	16	8	1	IO
DMA_ID_UART1	17	8	1	IO
DMA_ID_UART2	18	8	1	IO
DMA_ID_UART3	19	8	1	IO
DMA_ID_XSPI	24	8	16	IO

外设（非存储）设备与 DMA 采用单独的 REQ/ACK 机制握手方式，DMA 模式选择握手，即需主动发起接收或发送请求。外设（非存储）设备传输字节数需要与位宽对齐。

5.5.3.2. 任务链表

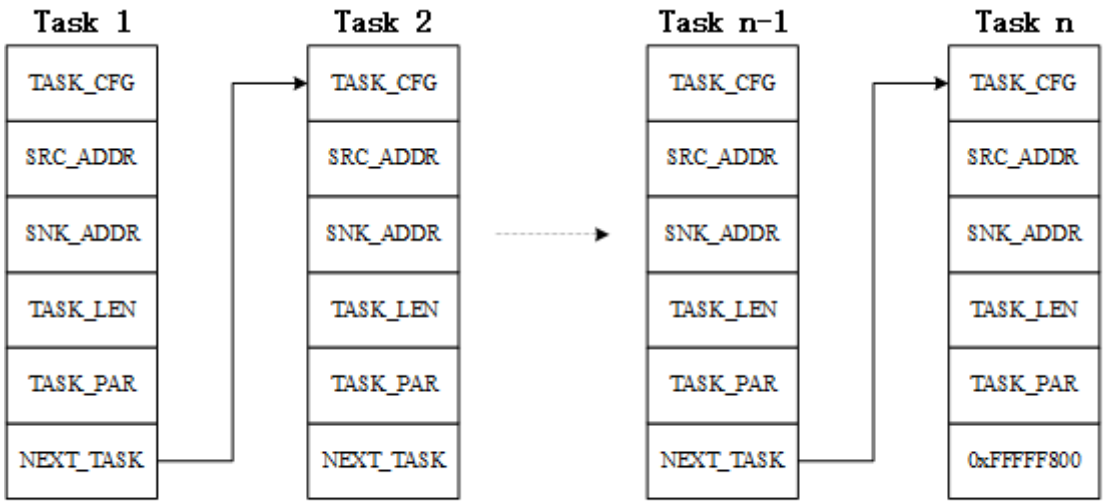


图 5–5 任务链表

多个任务组成任务链表，每个任务单元为 32 位，从低地址往高地址存放，具体定义如下：

- TASK_CFG1：当前任务配置，包括源端终端的 BURST 和 WIDTH 等（外设模块的 WIDTH 需要与自身 FIFO 寄存器位宽对齐）
- SRC_ADDR：当前任务源端起始地址，注意需要 8 byte 对齐

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- SNK_ADDR：当前任务终端低 32 bits 起始地址，注意需要 8 byte 对齐
- TASK_LENGTH：当前任务传输字节数，最大可设置为 32M–1 字节（总传输长度需要与 WIDTH 对齐）
- TASK_PAR：当前任务参数，设置两个任务之间传输的延迟
- NEXT_TASK：下一任务地址，该单元配置为 “0xFFFFF800” 表示 DMA 传输结束

DMA 在工作时会读出任务相关参数更新到对应寄存器中（软件可见）。

若需提高 DMA 带宽，建议配置起始地址和长度进行 64 bytes 对齐。

5.5.3.3. FIFO

外设 FIFO 大小推荐 FIFO_DEPTH ≥（BRT × DW ÷ 8）+（L ÷ S) BYTE

- BRT：突发 BRUST 长度。
- DW：为外设使用的 DMA 数据宽度。
- L：系统时延。
- S：外设吞吐速度。

实际应用则把 REQ 触发条件设置为（BRT × DW ÷ 8）BYTE，满足 BURST 传输的条件。（L ÷ S) 用于解决系统时延影响，保证外设可以正常吞吐。

- AHB 总线上的外设，建议 BRT 选择 16，APB 总线上的模块，建议 BRT 选择 1。
- 存储器设置地址递增，外设设置地址保持
- 源端位宽：从总位宽取有效数据值
- 终端位宽：从有效数据值，补全总位宽。
- SPI：AHB 设备，可以与 DRAM 或 SRAM 传输数据
- UART：APB 设备，可以与 DRAM 或 SRAM 传输数据
- 外设 DMA WIDTH 设置需与外设模块寄存器位宽对齐。
- 外设 DMA LENGTH 设置需要与 WIDTH 对齐

5.5.4. 寄存器列表

表 5–6 DMA 地址映射表

Module Name	Mapping Size	Offset
DMA	1 KB	0x000

表 5–7 DMA 寄存器列表

偏移地址	缩写	寄存器名称	章节
0x000	DMA_IRQ_EN	DMA 中断控制	0x000 DMA_IRQ_EN

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 5–7 DMA 寄存器列表 (续)

偏移地址	缩写	寄存器名称	章节
0x010	DMA_IRQ_STA	DMA 中断状态	0x010 DMA_IRQ_STA
0x020	DMA_MEM_CFG	DMA 存储模块配置	0x020 DMA_MEM_CFG
0x028	DMA_GATE	DMA 锁定配置	0x028 DMA_GATE
0x030	DMA_CH_STA	DMA 通道状态	0x030 DMA_CH_STA
0x100	DMA_CH_EN	DMA 通道使能	0x100+N*0x40(N=0~3) DMA_CH_EN
0x104	DMA_CH_PAUSE	DMA 通道暂定	0x104+N*0x40(N=0~3) DMA_CH_PAUSE
0x108	DMA_CH_TASK	DMA 通道任务地址	0x108+N*0x40(N=0~3) DMA_CH_TASK
0x10C	DMA_CH_CFG	DMA 通道配置	0x10C+N*0x40(N=0~3) DMA_CH_CFG
0x110	DMA_SRC_ADDR	DMA 源端地址	0x110+N*0x40(N=0~3) DMA_SRC_ADDR
0x114	DMA_SINK_ADDR	DMA 终端地址	0x114+N*0x40(N=0~3) DMA_SINK_ADDR
0x118	DMA_BCNT_LEFT	DMA 传输剩余字节	0x118+N*0x40(N=0~3) DMA_BCNT_LEFT
0x128	DMA_MODE	DMA 模式	0x128+N*0x40(N=0~3) DMA_MODE
0x12C	DMA_FDES_ADDR	DMA 前一任务地址	0x12C+N*0x40(N=0~3) DMA_FDES_ADDR
0x130	DMA_PKG_NUM	DMA 已传任务数量	0x130+N*0x40(N=0~3) DMA_PKG_NUM
0x134	DMA_MEM_SET	DMA 内存设置	0x134+N*0x40(N=0~3) DMA_MEM_SET

5.5.5. 寄存器描述

5.5.5.1. 0x000 DMA_IRQ_EN

默认值：0x00000000			DMA 中断控制 (DMA Interrupt Request Enable)
位域	类型	默认值	描述
31:16	–	–	–
15	R/W	0x0	CH3_ADDR_REQ_ERR_EN 通道 3 地址请求错误中断使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			DMA 中断控制 (DMA Interrupt Request Enable)
位域	类型	默认值	描述
			• 0：关闭中断 • 1：打开中断
14	R/W	0x0	CH3_ALL_FINISH_IRQ_EN 通道 3 完成全部任务中断使能 • 0：关闭中断 • 1：打开中断
13	R/W	0x0	CH3_TASK_FINISH_IRQ_EN 通道 3 完成一个任务中断使能 • 0：关闭中断 • 1：打开中断
12	R/W	0x0	CH3_TASK_HALF_IRQ_EN 通道 3 完成半个任务中断使能 • 0：关闭中断 • 1：打开中断
11	R/W	0x0	CH2_ADDR_REQ_ERR_EN 通道 2 地址请求错误中断使能 • 0：关闭中断 • 1：打开中断
10	R/W	0x0	CH2_ALL_FINISH_IRQ_EN 通道 2 完成全部任务中断使能 • 0：关闭中断 • 1：打开中断
9	R/W	0x0	CH2_TASK_FINISH_IRQ_EN 通道 2 完成一个任务中断使能 0: 关闭中断 1: 打开中断
8	R/W	0x0	CH2_TASK_HALF_IRQ_EN 通道 2 完成半个任务中断使能 • 0：关闭中断 • 1：打开中断
7	R/W	0x0	CH1_ADDR_REQ_ERR_EN 通道 1 地址请求错误中断使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			DMA 中断控制 (DMA Interrupt Request Enable)
位域	类型	默认值	描述
			• 0：关闭中断 • 1：打开中断
6	R/W	0x0	CH1_ALL_FINISH_IRQ_EN 通道 1 完成全部任务中断使能 • 0：关闭中断 • 1：打开中断
5	R/W	0x0	CH1_TASK_FINISH_IRQ_EN 通道 1 完成一个任务中断使能 • 0：关闭中断 • 1：打开中断
4	R/W	0x0	CH1_TASK_HALF_IRQ_EN 通道 1 完成半个任务中断使能 • 0：关闭中断 • 1：打开中断
3	R/W	0x0	CH0_ADDR_REQ_ERR_EN 通道 0 地址请求错误中断使能 • 0：关闭中断 • 1：打开中断
2	R/W	0x0	CH0_ALL_FINISH_IRQ_EN 通道 0 完成全部任务中断使能 • 0：关闭中断 • 1：打开中断
1	R/W	0x0	CH0_TASK_FINISH_IRQ_EN 通道 0 完成一个任务中断使能 • 0：关闭中断 • 1：打开中断
0	R/W	0x0	CH0_TASK_HALF_IRQ_EN 通道 0 完成半个任务中断使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

5.5.5.2. 0x010 DMA_IRQ_STA

默认值：0x00000000			DMA 中断状态 (DMA Interrupt States)
位域	类型	默认值	描述
31	R/W	0x0	CH7_ADDR_REQ_ERR_STA 通道 7 地址请求错误中断状态 • 0: 无中断 • 1: 有中断
30	R/W	0x0	CH7_ALL_FINISH_IRQ_STA 通道 7 完成全部任务中断状态 • 0: 无中断 • 1: 有中断
29	R/W	0x0	CH7_TASK_FINISH_IRQ_STA 通道 7 完成一个任务中断状态 • 0: 无中断 • 1: 有中断
28	R/W	0x0	CH7_TASK_HALF_IRQ_STA 通道 7 完成半个任务中断状态 • 0: 无中断 • 1: 有中断
27	R/W	0x0	CH6_ADDR_REQ_ERR_STA 通道 6 地址请求错误中断状态 • 0: 无中断 • 1: 有中断
26	R/W	0x0	CH6_ALL_FINISH_IRQ_STA 通道 6 完成全部任务中断状态 • 0: 无中断 • 1: 有中断
25	R/W	0x0	CH6_TASK_FINISH_IRQ_STA 通道 6 完成一个任务中断状态 • 0: 无中断 • 1: 有中断

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			DMA 中断状态 (DMA Interrupt States)
位域	类型	默认值	描述
24	R/W	0x0	CH6_TASK_HALF_IRQ_STA 通道 6 完成半个任务中断状态 • 0: 无中断 • 1: 有中断
23	R/W	0x0	CH5_ADDR_REQ_ERR_STA 通道 5 地址请求错误中断状态 • 0: 无中断 • 1: 有中断
22	R/W	0x0	CH5_ALL_FINISH_IRQ_STA 通道 5 完成全部任务中断状态 • 0: 无中断 • 1: 有中断
21	R/W	0x0	CH5_TASK_FINISH_IRQ_STA 通道 5 完成一个任务中断状态 • 0: 无中断 • 1: 有中断
20	R/W	0x0	CH5_TASK_HALF_IRQ_STA 通道 5 完成半个任务中断状态 • 0: 无中断 • 1: 有中断
19	R/W	0x0	CH4_ADDR_REQ_ERR_STA 通道 4 地址请求错误中断状态 • 0: 无中断 • 1: 有中断
18	R/W	0x0	CH4_ALL_FINISH_IRQ_STA 通道 4 完成全部任务中断状态 • 0: 无中断 • 1: 有中断
17	R/W	0x0	CH4_TASK_FINISH_IRQ_STA 通道 4 完成一个任务中断状态

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			DMA 中断状态 (DMA Interrupt States)
位域	类型	默认值	描述
			• 0: 无中断 • 1: 有中断
16	R/W	0x0	CH4_TASK_HALF_IRQ_STA 通道 4 完成半个任务中断状态 • 0: 无中断 • 1: 有中断
15	R/W	0x0	CH3_ADDR_REQ_ERR_STA 通道 3 地址请求错误中断状态 • 0: 无中断 • 1: 有中断
14	R/W	0x0	CH3_ALL_FINISH_IRQ_STA 通道 3 完成全部任务中断状态 • 0: 无中断 • 1: 有中断
13	R/W	0x0	CH3_TASK_FINISH_IRQ_STA 通道 3 完成一个任务中断状态 • 0: 无中断 • 1: 有中断
12	R/W	0x0	CH3_TASK_HALF_IRQ_STA 通道 3 完成半个任务中断状态 • 0: 无中断 • 1: 有中断
11	R/W	0x0	CH2_ADDR_REQ_ERR_STA 通道 2 地址请求错误中断状态 • 0: 无中断 • 1: 有中断
10	R/W	0x0	CH2_ALL_FINISH_IRQ_STA 通道 2 完成全部任务中断状态 • 0: 无中断 • 1: 有中断

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			DMA 中断状态 (DMA Interrupt States)
位域	**类型**	**默认值**	**描述**
9	R/W	0x0	CH2_TASK_FINISH_IRQ_STA 通道 2 完成一个任务中断状态 • 0: 无中断 • 1: 有中断
8	R/W	0x0	CH2_TASK_HALF_IRQ_STA 通道 2 完成半个任务中断状态 • 0: 无中断 • 1: 有中断
7	R/W	0x0	CH1_ADDR_REQ_ERR_STA 通道 1 地址请求错误中断状态 • 0: 无中断 • 1: 有中断
6	R/W	0x0	CH1_ALL_FINISH_IRQ_STA 通道 1 完成全部任务中断状态 • 0: 无中断 • 1: 有中断
5	R/W	0x0	CH1_TASK_FINISH_IRQ_STA 通道 1 完成一个任务中断状态 • 0: 无中断 • 1: 有中断
4	R/W	0x0	CH1_TASK_HALF_IRQ_STA 通道 1 完成半个任务中断状态 • 0: 无中断 • 1: 有中断
3	R/W	0x0	CH0_ADDR_REQ_ERR_STA 通道 0 地址请求错误中断状态 • 0: 无中断 • 1: 有中断
2	R/W	0x0	CH0_ALL_FINISH_IRQ_STA 通道 0 完成全部任务中断状态

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			DMA 中断状态 (DMA Interrupt States)
位域	类型	默认值	描述
			• 0: 无中断 • 1: 有中断
1	R/W	0x0	CH0_TASK_FINISH_IRQ_STA 通道 0 完成一个任务中断状态 • 0: 无中断 • 1: 有中断
0	R/W	0x0	CH0_TASK_HALF_IRQ_STA 通道 0 完成半个任务中断状态 • 0: 无中断 • 1: 有中断

5.5.5.3. 0x020 DMA_MEM_CFG

默认值：0x80000000			DMA 存储模块配置 (DMA Memory Config)
位域	类型	默认值	描述
31	R/W	0x1	MEM_BURST_SET 存储类型突发设置 • 0: BURST8 • 1: BURST16
30:0	–	–	–

5.5.5.4. 0x028 DMA_GATE

默认值：0x00000000			DMA 锁定配置 (DMA GATE)
位域	类型	默认值	描述
31:8	–	–	–
7:1	–	–	–
0	R/W	0x0	DMA_GATE_BYPASS DMA 锁定解除

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			DMA 锁定配置 (DMA GATE)
位域	类型	默认值	描述
			• 0: 关闭 • 1: 开启 一个通道对应 1BIT

5.5.5.5. 0x030 DMA_CH_STA

默认值：0x00000000			DMA 通道状态 (DMA Channel State)
位域	类型	默认值	描述
31:	_	_	_
7	R	0x0	DMA_CH7_STATUS DMA 通道 7 状态： • 0: 空闲中 • 1: 工作中
6	R	0x0	DMA_CH6_STATUS DMA 通道 6 状态 • 0: 空闲中 • 1: 工作中
5	R	0x0	DMA_CH5_STATUS DMA 通道 5 状态 • 0: 空闲中 • 1: 工作中
4	R	0x0	DMA_CH4_STATUS DMA 通道 4 状态 • 0: 空闲中 • 1: 工作中
3	R	0x0	DMA_CH3_STATUS DMA 通道 3 状态 • 0: 空闲中 • 1: 工作中
2	R	0x0	DMA_CH2_STATUS

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			DMA 通道状态 (DMA Channel State)
位域	类型	默认值	描述
			DMA 通道 2 状态 • 0: 空闲中 • 1: 工作中
1	R	0x0	DMA_CH1_STATUS DMA 通道 1 状态 • 0: 空闲中 • 1: 工作中
0	R	0x0	DMA_CH0_STATUS DMA 通道 0 状态 • 0: 空闲中 • 1: 工作中

5.5.5.6. 0x100+N*0x40(N=0~3) DMA_CH_EN

默认值：0x00000000			DMA 通道使能 (DMA Channel Enable)
位域	类型	默认值	描述
31:1	_	_	_
0	R/WAC	0x0	DMA_CH_ENDMA 通道使能 • 0: 关闭 • 1: 打开 注意 DMA 传输完成后，该位会自动清零，可以通过读取该位表示 DMA 当前状态

5.5.5.7. 0x104+N*0x40(N=0~3) DMA_CH_PAUSE

默认值：0x00000000			DMA 通道暂定 (DMA Channel Pause)
位域	类型	默认值	描述
31:6	_	_	_
5	R/WAC	0x0	MEM_BYTE_ENABLE 内存字节模式使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			DMA 通道暂定 (DMA Channel Pause)
位域	类型	默认值	描述
			• 0: 关闭 • 1: 打开 DMA 完成之后自动清零
4	R/WAC	0x0	MEM_SET_ENABLE 内存设置使能 • 0: 关闭 • 1: 打开 DMA 完成之后自动清零
3:1	–	–	–
0	R/W	0x0	DMA_CH_PAUSE DMA 通道暂停 • 0: 继续传输 • 1: 暂停传输

5.5.5.8. 0x108+N*0x40(N=0~3) DMA_CH_TASK

默认值：0x00000000			DMA 通道任务地址 (DMA Channel task Addr)
位域	类型	默认值	描述
31:0	R/W	0x0	DMA_TASK_ADDR DMA 任务地址

5.5.5.9. 0x10C+N*0x40(N=0~3) DMA_CH_CFG

默认值：0x00000000			DMA 通道配置 (DMA Channel Config)
位域	类型	默认值	描述
31:27	–	–	–
26:25	R	0x0	SNK_DATA_WIDTH DMA 终端数据位宽 • 0: 8 位 • 1: 16 位

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			DMA 通道配置 (DMA Channel Config)
位域	类型	默认值	描述
			• 2: 32 位 • 3: 64 位
24	R	0x0	SNK_ADDR_MODE DMA 终端地址模式 • 0: 递增 • 1: 保持
23:22	R	0x0	SNK_BURST DMA 终端突发数 • 0: 1 • 1: 4 • 2: 8 • 3: 16
21	–	–	–
21:16	R	0x0	SNK_DEV DMA 终端设备
15:11	–	–	–
10:9	R	0x0	SRC_DATA_WIDTH DMA 源端数据位宽（总线上有效数据的位宽） • 0: 8 位 • 1: 16 位 • 2: 32 位 • 3: 64 位
8	R	0x0	SRC_ADDR_MODE DMA 源端地址模式 • 0: 递增 • 1: 保持
7:6	R	0x0	SRC_BURST (Device IO BURST 配置 1) DMA 源端突发数 • 0: 1 • 1: 4

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			DMA 通道配置 (DMA Channel Config)
位域	类型	默认值	描述
			• 2: 8 • 3: 16
5	–	–	–
5:0	R	0x0	SRC_DEV DMA 源端设备

5.5.5.10. 0x110+N*0x40(N=0~3) DMA_SRC_ADDR

默认值：0x00000000			DMA 源端地址 (DMA Source Addr)
位域	类型	默认值	描述
31:0	R	0x0	DMA 任务源端地址[2:0]=3'b000(8byte 对齐)

5.5.5.11. 0x114+N*0x40(N=0~3) DMA_SINK_ADDR

默认值：0x00000000			DMA 终端地址 (DMA Sink Addr)
位域	类型	默认值	描述
31:0	R	0x0	DMA 任务终端地址[2:0]=3'b000(8byte 对齐)

5.5.5.12. 0x118+N*0x40(N=0~3) DMA_BCNT_LEFT

默认值：0x00000000			DMA 传输剩余字节 (DMA Byte Counter Left)
位域	类型	默认值	描述
31:25	–	–	–
24:0	R	0x0	DMA 任务传输剩余字节

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

5.5.5.13. 0x128+N*0x40(N=0~3) DMA_MODE

默认值：0x00000000			DMA 模式 (DMA Mode)
位域	类型	默认值	描述
31:7	_	_	_
6	R/W	0x0	SNK_HANDSHAKE_ENABLE DMA 终端使能 当终端模式为握手且 DMA_CH_CFG 寄存器中 SNK_BURST 不为 1 时需要使能该位 • 0: Bypass • 1: Enable
5	R/W	0x0	SRC_HANDSHAKE_ENABLE DMA 源端使能 当源端模式为握手且 DMA_CH_CFG 寄存器中 SRC_BURST 不为 1 时需要使能该位 • 0: Bypass • 1: Enable
4	R/W	0x0	HANDSHAKE_ENABLE 握手使能 当有源端或终端模式为握手且 CFG 寄存器中 BURST 设置不为 1 时需要使能该位 • 0：Bypass • 1：Enable
3	R/W	0x0	DMA_SNK_MODE DMA 终端模式 • 0: 等待 • 1: 握手
2	R/W	0x0	DMA_SRC_MODE DMA 源端模式 • 0: 等待 • 1: 握手
1:0	_	_	_

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

5.5.5.14. 0x12C+N*0x40(N=0~3) DMA_FDES_ADDR

默认值：0x00000000			DMA 前一任务地址 (DMA Former Descriptor Address)
位域	类型	默认值	描述
31:0	R	0x0	DMA 前一任务地址

5.5.5.15. 0x130+N*0x40(N=0~3) DMA_PKG_NUM

默认值：0x00000000			DMA 已传任务数量 (DMA Package Number)
位域	类型	默认值	描述
31:0	R	0x0	DMA 已传任务数量

5.5.5.16. 0x134+N*0x40(N=0~3) DMA_MEM_SET

默认值：0x00000000			DMA 内存设置 (DMA Memory Set)
位域	类型	默认值	描述
31:0	R/W	0x0	存放内存设置具体信息，只有当 DMA_CH_PAUSE[4] 打开后有效。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

6. 安全

6.1. Serial Peripheral Interface Encryption (SPI ENC)

Serial Peripheral Interface Encryption (SPI ENC) 模块用于对 SPI 总线上的数据进行在线加密和解密，实现对 CPU 透明的 SPI 数据加密存储。即 CPU 对 SPI 存储设备的读写数据是明文，但存储在 SPI 设备上的数据是密文，数据的加密和解密操作在数据传输过程中完成。

SPI ENC 用于 SPI 存储设备，即 SPI NOR/ SPI NAND，需要配合 SPI 控制器使用。

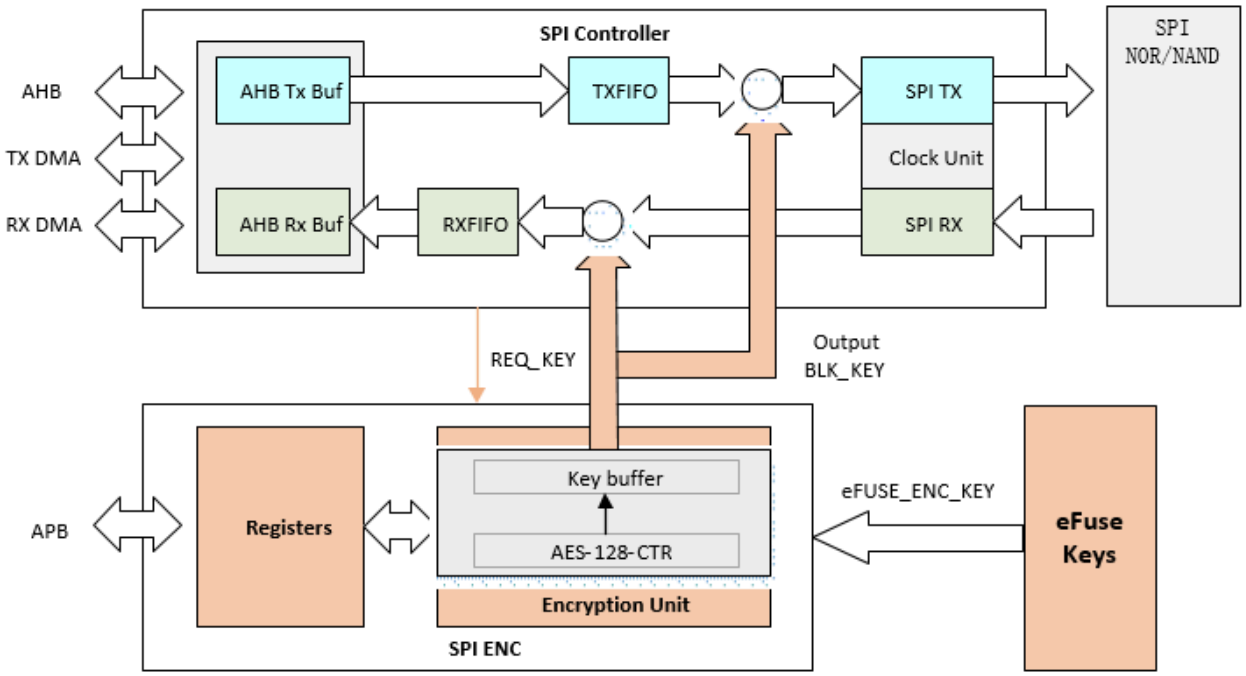
6.1.1. 特性说明

- 支持 AES-128-CTR 加密和解密。
- 通过配置连接到不同的 SPI 控制器。
- 通过 eFuse 配置密钥。
- 支持明文和密文混合传输。
- 不影响 SPI 总线传输带宽。
- 不支持 SPI 全双工模式。
- 内置 64 字节分组密钥 Buffer。
- 支持全 0xFF 空页检测。

ArtInChip

Copyright © 2023-2025 ArtInChip Technology Co., Ltd.
All rights reserved

6.1.2. 原理框图



6.1.3. 功能描述

SPI 数据加密使用 AES–128–CTR 算法，加解密总体计算过程如下：

1. 以 16 字节为单位对数据进行分组，并且为每个数据分组产生对应的 Counter 值。

   关于详细的计算详细流程，可查看 计算分组密钥。

2. 使用一个 128–bit AES 密钥，对 Counter 值进行加密运算，产生数据分组的对应分组密钥。

   产生数据分组对应的 Counter 值，需要使用一个 Nonce，在 SPI ENC 使用过程中，Nonce 作为密钥数据的一部分提供。

3. 使用分组密钥与数据进行异或运算，得到密文。

   关于如何输出分组密钥的流程，可查看 输出分组密钥。

   关于异或运算的详细描述，可查看 数据加解密。

密钥数据的来源有两个：

- SPI 加密模块从 eFuse 中读取 AES KEY 和 Nonce。eFuse 中保存的密钥数据如下：

名称	长度	说明
SPI_ENC_KEY	128 bits	加密密钥，仅 SPI_ENC 模块可见。
SPI_ENC_NONCE	64 bits	生成 Counter，仅 SPI_ENC 模块可见。

- 用户通过相应的寄存器配置 AES KEY 和 Nonce，主要用于内部调试。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

6.1.3.1. 计算分组密钥

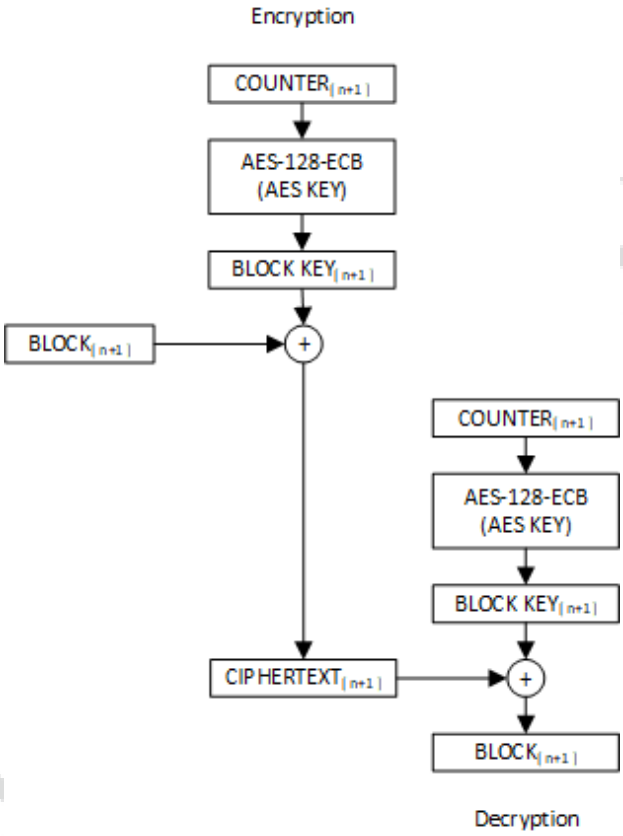


图 6–1 AES–128–CTR 算法原理

图 6–1 : AES–128–CTR 算法原理中，使用 AES–128–CTR 进行加密或者解密的过程分为三个步骤：

1. 生成数据分组的 Counter 值。

   Counter 值的产生方式如下：

   ```
   ID = (SPIE_ADDR & 0xFFFFFFF0) / 16
   Counter(ID) = (Nonce << 64) +(SPIE_TWEAK << 32) +  ID
   ```

2. 生成分组密钥。

   a. 使用数据地址寄存器 SPIE_ADDR 中的地址值，计算分组 ID。

   b. 分组 ID 与 Nonce 和 Tweak 值组成 Counter 值。

   c. 使用 AES–128–ECB 加密算法，用 AES KEY 对 Counter 值进行加密，得到分组密钥。

   SPI ENC 一次可产生 16 字节的分组密钥，输出到内部 Key buffer。

   模块开始工作后，只要 Key buffer 有空间，计算分组密钥就会继续，并且总是按地址顺序自动计算下一组密钥，直到控制寄存器 SPIE_CTL[0] 被设置为 0。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

注：

如图 6–1 : AES–128–CTR 算法原理所示，数据的加密和解密的分组密钥计算流程完全一致。

3. 使用分组密钥对数据进行异或运算，产生加密数据。详细流程可查看 输出分组密钥。

6.1.3.2. 输出分组密钥

SPI ENC 模块仅计算 AES–128–CTR 的分组密钥，最后一步使用分组密钥对数据进行异或运算由 SPI 控制器完成。因此在启用 SPI ENC 的情况下，SPI 控制器需要为发送到总线或者从总线接收到的每一字节数据读取分组密钥。

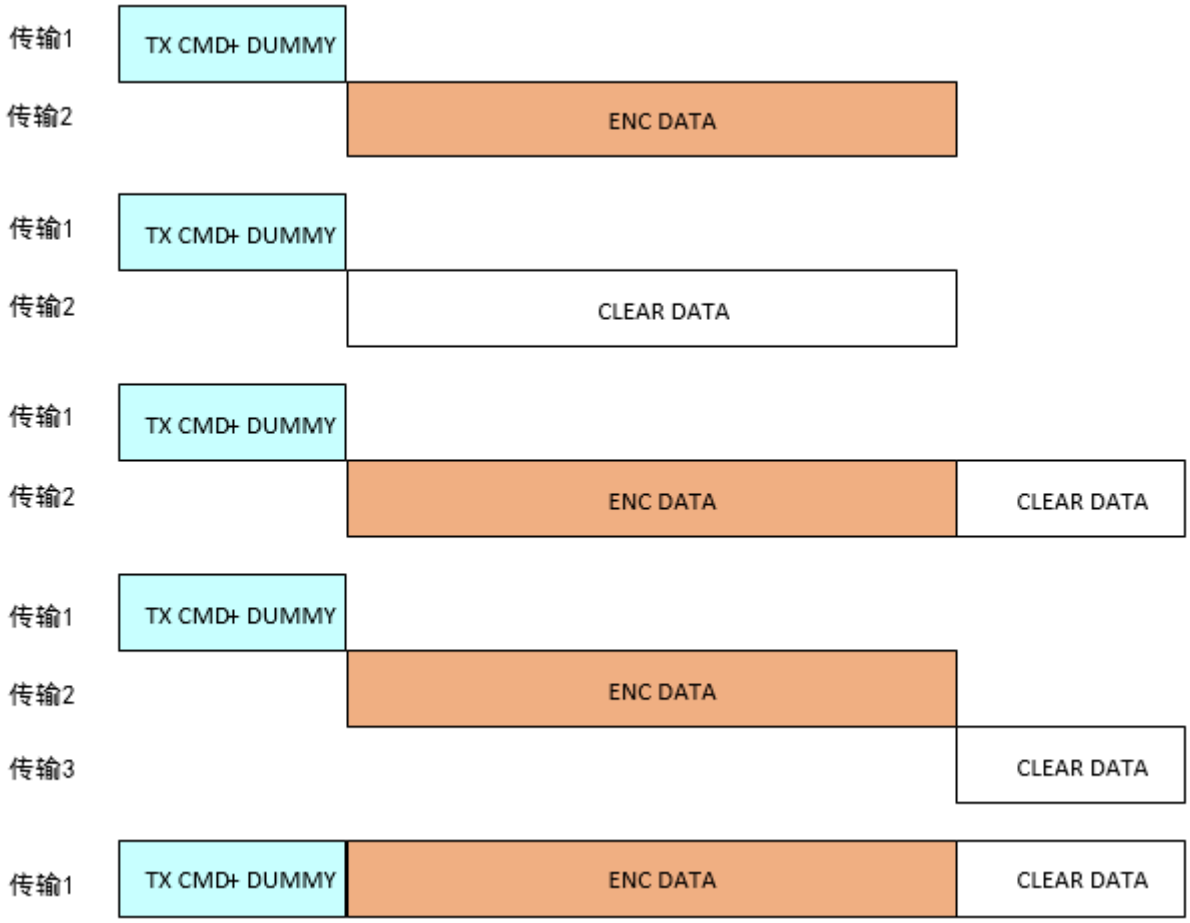


图 6–2 SPI 数据传输模式

如 图 6–2 : SPI 数据传输模式所示，在与 SPI NOR/ NAND 通信过程中，传输的数据可能出现明文和密文混合传输的情形。

SPI ENC 输出分组密钥的规则如下：

- SPI 控制器每收发一个字节数据，都要读取一个字节的分组密钥
- 明文数据位置对应的分组密钥，总是读取到 0；
- 密文数据位置对应的分组密钥，总是计算得到的分组密钥；

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

分组密钥的计算总是以 16 字节为单位，但用户访问的地址可能并不是 16 字节对齐，如从 0x0000_0005 地址开始读取数据。 这种情况下，SPI ENC 计算的第一个分组密钥仍然为 16 字节，只是对应的数据地址为 0x0000_0000 ~ 0x0000_000F；SPI ENC 输出数据分组密钥时，应输出 0x05~0x0F 数据对应的分组密钥值。

注：

SPI ENC 模块要准确的计算和输出分组密钥，依赖于用户开始传输前配置的数据地址、长度、加密数据的起始地址和长度等信息。

6.1.3.3. 数据加解密

使用分组密钥与对应的数据分组进行异或运算，即可完成数据的加密或者解密。

注：

由于 SPI ENC 输出的密钥，只能应用于接收的数据，或者发送的数据，不可以同时应用于两组数据，因此使能 SPI ENC 时，不支持 SPI 全双工模式，

- 对于发送数据，数据会先被写入 TXFIFO，SPI 控制器在将数据发送到 SPI 总线上之前执行：
  1. 读取一个字节的分组密钥数据。
  2. 将分组密钥数据与要发送的一个字节数据进行异或运算。
  3. 将结果发送到 SPI 总线。
- 对于接收数据：
  1. 从 SPI 总线读取到一个字节数据的同时，读取一个字节的分组密钥。
  2. 将分组密钥数据与接收的数据进行异或运算。
  3. 将解密后的数据，写入 RXFIFO。

6.1.4. 编程指南

开始配置之前，确保 SPI ENC 模块已经初始化。初始化步骤如下：

1. 根据系统配置，将 SPI ENC 模块连接到正确的 SPI 控制器。

   通过设置 SPIE_CTL[13:12] 选择连接的 SPI 控制器。
2. 设置密钥。默认使用 eFuse 中的密钥数据。
3. 通过设置 SPIE_ICR 寄存器使能相应中断。

SPI_ENC 模块的配置步骤如下：

1. 读写 SPI 存储，传输时确保 SPIE_CTL[0] = 0 (KEY_START = 0)。
   读写 SPI 存储时，不需要对命令和 SPI 器件的 OTP 读写操作进行加解密。
2. 设置以下参数进行 SPI 存储的加密数据传输。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- SPIE_ADDR：当前访问的数据地址
- SPIE_TWEAK：用于调整 Counter 值
- SPIE_CPOS, SPIE_CLEN：当前传输的密文数据位置

3. 设置 SPIE_CTL[0] = 1 (KEY_START = 1) 值，开始计算分组密钥。
   产生密钥 (SPIE_ISR[0])。
   确保密钥已经产生。如密钥产生，则 SPI 控制器开始传输数据。
4. 通过 SPIE_ISR 检查加解密是否完成，并查看读取的数据是否为空数据（全为 0xFF）。

6.1.5. 寄存器列表

表 6–1 SPIENC 寄存器列表

偏移地址	缩写	寄存器名称	章节
0x000	SPIE_CTL	控制寄存器	0x000 SPIE_CTL
0x004	SPIE_ICR	中断控制寄存器	0x004 SPIE_ICR
0x008	SPIE_ISR	中断状态寄存器	0x008 SPIE_ISR
0x00C	SPIE_KCNT	密钥计数寄存器	0x00C SPIE_KCNT
0x010	SPIE_OCNT	输出计数寄存器	0x010 SPIE_OCNT
0x014	SPIE_ADDR	数据地址寄存器	0x014 SPIE_ADDR
0x018	SPIE_TWEAK	计数调整寄存器	0x018 SPIE_TWEAK
0x01C	SPIE_CPOS	密文位置寄存器	0x01C SPIE_CPOS
0x020	SPIE_CLEN	密文长度寄存器	0x020 SPIE_CLEN
0xFFC	SPIE_VER	SPIENC 版本	0xFFC SPIE_VER

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

6.1.6. 寄存器描述

6.1.6.1. 0x000 SPIE_CTL

默认值：0x00000000			控制寄存器（Control Register）
位域	类型	默认值	描述
31:17	–	–	–
16	R/W	0x0	XIP_EN SPI XIP 的总线数据加密使能 使能后，硬件自动连接 SPI 控制器。
15:14	–	–	–
13:12	R/W	0x0	SPI_SEL SPI 控制器选择 选择连接的总线，加密器对所选总线上的数据进行加解密。 • 0x0：BYPASS，不连接任何总线 • 0x1：连接 SPI0 • 0x2：连接 SPI1 XIP_EN 时，本字段设置无效。
11:1	–	–	–
0	R/W	0x0	KEY_START 开始计算分组密钥 数据加解密开始之前或者同时，必须先计算分组密钥，否则加解密单元会一直等待密钥生成，导致传输失败。传输完成，需要清零以停止密钥计算。 • 0：停止计算分组密钥 • 1：开始计算分组密钥 XIP_EN 时，本字段设置无效。

6.1.6.2. 0x004 SPIE_ICR

默认值：0x00000000			中断控制寄存器（Interrupt Control Register）
位域	类型	默认值	描述
31:6	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			中断控制寄存器（Interrupt Control Register）
位域	类型	默认值	描述
5	R/W	0x0	KEY_OVF 分组密钥 FIFO 上溢中断使能
4	R/W	0x0	KEY_UDF 分组密钥 FIFO 下溢中断使能
3	R/W	0x0	READ_HALF_EMPTY 读过半空数据中断使能
2	R/W	0x0	READ_EMPTY_DATA 读空数据中断使能
1	R/W	0x0	ENC_DEC_FINISHED 加解密完成中断使能
0	R/W	0x0	KEY_GEN 第一组分组密钥生成中断使能

6.1.6.3. 0x008 SPIE_ISR

默认值：0x00000000			中断状态寄存器（Interrupt Status Register）
位域	类型	默认值	描述
31:6	–	–	–
5	W1C	0x0	KEY_OVF 分组密钥 FIFO Overflow 中断状态。 • 0：分组密钥 FIFO 未发生上溢 • 1：分组密钥 FIFO 发生上溢
4	W1C	0x0	KEY_UDF 分组密钥 FIFO Underflow 中断状态。 • 0：分组密钥 FIFO 未发生上溢 • 1：分组密钥 FIFO 发生上溢
3	W1C	0x0	READ_HALF_EMPTY 读过半空数据中断，即读取的密文位置的数据，过半都是 0xFF。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			中断状态寄存器（Interrupt Status Register）
位域	类型	默认值	描述
			• 0：中断未发生 • 1：当前读取的密文数据，超过一半都是 0xFF
2	W1C	0x0	READ_EMPTY_DATA 读空数据中断，即读取的密文位置的数据全部为 0xFF，即 Flash 擦除后的状态。 • 0：中断未发生 • 1：当前读取的密文数据，全部为 0xFF 软件可以根据该中断判断当前读取的 Page，是否为擦除后的数据。
1	W1C	0x0	ENC_DEC_FINISHED 加解密完成中断。 • 0：中断未发生 • 1：对密文位置的数据加密或者解密完成 若当前传输的密文长度为 0 时，该中断不会产生。
0	W1C	0x0	KEY_GEN 第一组分组密钥生成的中断状态。 • 0：中断未发生 • 1：第一组分组密钥已经产生 SPI 控制器可以开始传输数据。 在数据传输前，应确保该中断状态位为 1。 由于 ENC 模块产生密钥的速度比 SPI 数据传输的速度快，因此可以保证传输过程中，密钥总是能跟上数据传输的速度。

6.1.6.4. 0x00C SPIE_KCNT

默认值：0x00000000			密钥计数寄存器（Key Counter Register）
位域	类型	默认值	描述
31:0	RO	0x0	KEY_CNTSPI ENC 输出对应密文区域的分组密钥个数，单位为字节。

6.1.6.5. 0x010 SPIE_OCNT

默认值：0x00000000			输出计数寄存器（Output Counter Register）
位域	类型	默认值	描述
31:0	RO	0x0	OUTPUT_CNT

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			输出计数寄存器（Output Counter Register）
位域	类型	默认值	描述
			SPI ENC 输出的分组密钥个数，以字节为单位计数。 SPI ENC 为每一个传输的 SPI 数据输出分组密钥，但是如果对应输出的位置是明文，SPI ENC 输出的分组密钥为 0。所以此处计数是输出的 0 的个数和密文对应的分组密钥个数之和。

6.1.6.6. 0x014 SPIE_ADDR

默认值：0x00000000			数据地址寄存器（Data Address Register）
位域	类型	默认值	描述
31:0	R/W	0x0	DATA_ADDR 本次访问的 Flash 数据地址。从 0 开始，是相对 Flash 开始位置的访问偏移值，单位为字节。 该地址信息有两个用途： 1. 计算数据分组对应的 Counter 值 2. 输出分组密钥数据 XIP_EN 时，本字段设置无效。

6.1.6.7. 0x018 SPIE_TWEAK

默认值：0x00000000			计数调整寄存器（Tweak Register）
位域	类型	默认值	描述
31:0	R/W	0x0	CTR_TWEAK 用于产生数据分组的 Counter 值。数据分组的 Counter 值由 eFuse 中的 SPI_ENC_NONCE、SPIE_ADDR 以及 TWEAK 值共同产生，其中 NONCE 与数据的分组地址是固定的，TWEAK 值软件可调。 通过配置 TWEAK 值，软件可以根据实际情况调整 Counter 值，以免每次产生的分组密钥相同。

6.1.6.8. 0x01C SPIE_CPOS

默认值：0x00000000			密文位置寄存器（Ciphertext Position Register）
位域	类型	默认值	描述
31:0	R/W	0x0	CIPHERTEXT_POS

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			密文位置寄存器（Ciphertext Position Register）
位域	类型	默认值	描述
			本次传输的密文数据开始位置 用于指示本次传输的数据中密文数据所在的偏移位置。如果当前传输包含了发送和接收，则位置总是从发送开始计算。 值 M：表示本次传输的数据，从第 M 个字节开始是密文 XIP_EN 时，本字段设置无效。

6.1.6.9. 0x020 SPIE_CLEN

默认值：0x00000000			密文长度寄存器（Ciphertext Length Register）
位域	类型	默认值	描述
31:0	R/W	0x0	CIPHERTEXT_LEN 本次传输的密文数据长度。 值 N：表示本次传输的密文数据为 N 字节 XIP_EN 时，本字段设置无效。

6.1.6.10. 0xFFC SPIE_VER

默认值：0x00000101			SPIENC 版本（Version Register）
位域	类型	默认值	描述
31:0	RO	0x00000101	Version（模块版本号） 采用 BCD 码显示，V1.1

6.2. Secure ID (SID)

Secure ID (SID) 模块主要用于读写 eFuse，包括每颗芯片独立 ID、每颗芯片量产校准以及安全管理用到的 KEY 和 HASH 码等信息。

6.2.1. 特性说明

- 1Kbit eFuse 双备份实现 512bit 空间，软件读写按照1Kbit 空间，以 32 bits 为单位
- 512bit eFuse Buffer，上电复位放开后读取 eFuse Cell 到 eFuse Buffer
- 512bit 空间，以 32 bits 为单位，可烧写禁止读操作或写操作
- 提供 SPI_ENC 烧录的 KEY

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 支持烧写关闭 JTAG 功能，并可通过安全认证再打开 JTAG
- 提供 128-bit Chip ID，每颗芯片唯一
- 提供量产测试电阻电压校准值
- 每次烧写之间需要延迟至少 3 us 以上

6.2.2. 原理框图

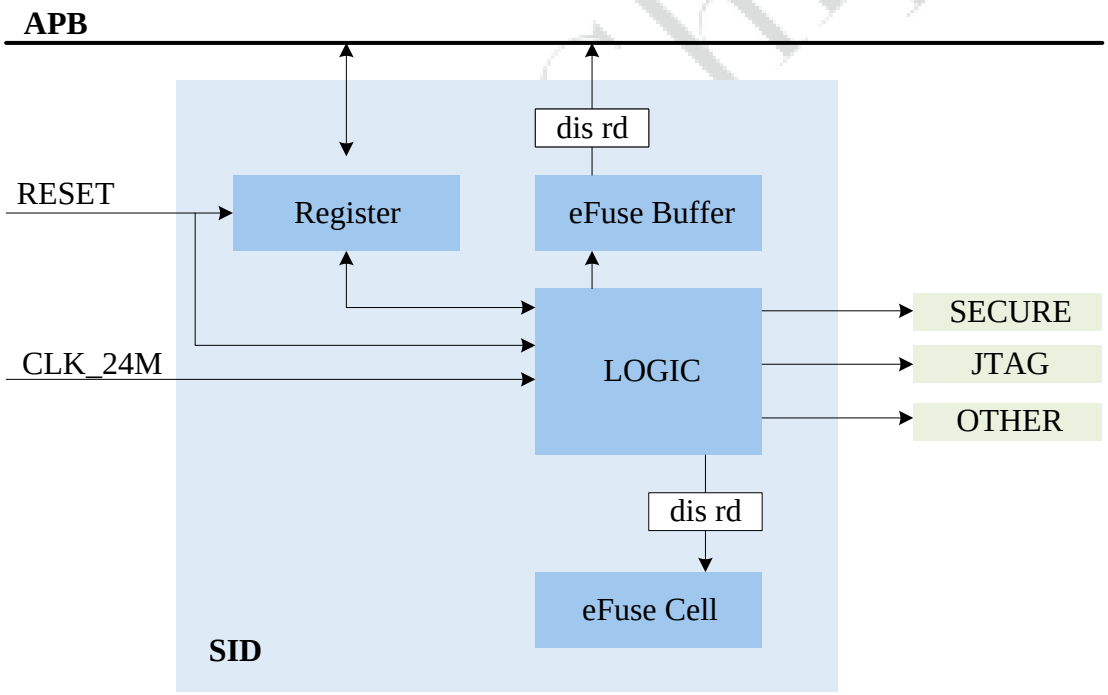


图 6-3 SID 原理框图

6.2.3. 功能描述

6.2.3.1. eFuse 和 SRAM 空间映射

软件读写 eFuse 空间，在 1Kbit 范围内以 32 bits 为单位读写。

为提升可靠性，采用双备份方案，在对 eFuse 进行烧写时，建议先后往 eFuse word[n] 和 word[64+n] 写入相同的值。

开机时，硬件自动读取 eFuse Cell 到 eFuse Buffer 空间，针对两个 word 空间进行按位或操作， 任意一位为 1 则 eFuse Buffer 对应位结果为 1，起到双备份的目的， 如 图 6-4 : eFuse Cell 和 eFuse Buffer 空间映射所示。

Copyright © 2023-2025 ArtInChip Technology Co., Ltd.
All rights reserved

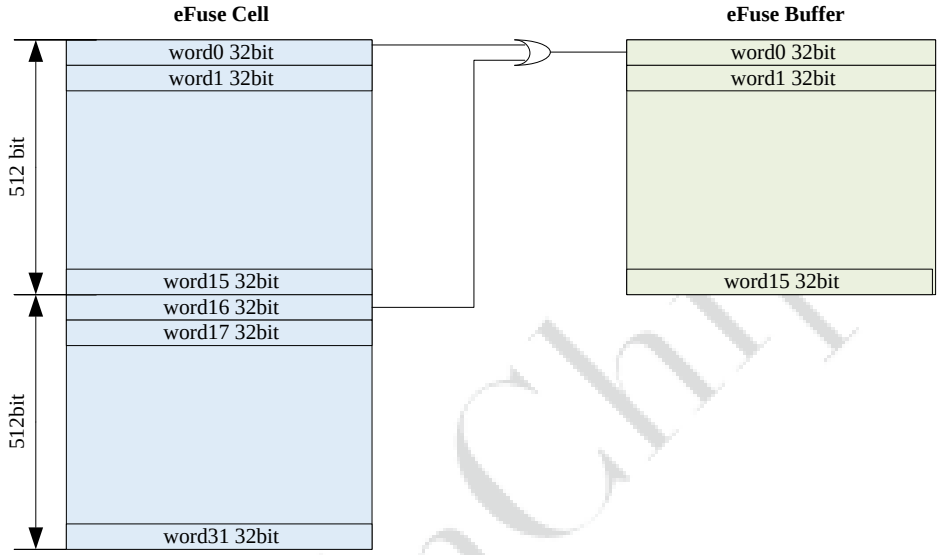


图 6–4 eFuse Cell 和 eFuse Buffer 空间映射

6.2.3.2. 512bit 空间映射定义

地址	用途	位数	禁止位	禁写	禁读	归属	备注
0~3	DIS	32	0	–	–	CSTM	15:00 eFuse 读禁止配置 31:16 eFuse 写禁止配置每位对应 1 个 32bits 空间
4~7	BROM_SECURE	32	1	V	–	CSTM	BROM 参数配置区域
8~F	CALI	64	2~3	V	–	AIC	模拟校准使用
10~1F	CHIP ID	128	4~7	V	–	AIC	芯片编号（子编号用到 Reserved 区域）
20~2F	SPI_ENC KEY	128	8~11	V	V	CSTM	安全，连接到 SPI_ENC，对称密钥
–	SPI_ENC NONCE	64	–	–	–	CSTM	此版本不可烧录，固定为某个值
30~3F	Reserved	128	12~15	–	–	CSTM	OEM 可自定义使用

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

6.2.3.3. CALI 区域定义

表 6–2 偏移地址: 0x08

比特位	名称	描述	备注
31:24	OSC24_OUT_TR	24MHz 振荡器校准值	配置到0x0A4 PLL_INOSC_OUT_TR24M
23:16	–	–	–
15:8	AVCC_BG_CTRL	AVCC BG 校准值	配置到0x020 LDO25_CFGBG_CTRL
7:4	–	–	–
3	THS_VAL_VALID	THS 烧写值有效	–
2	OSC24_OUT_VDLID	24MHz 振荡器校准值有效	–
0	AVCC_BG_VALID	AVCC BG 校准值有效	–

表 6–3 偏移地址: 0x0C

比特位	名称	描述	备注
31:25	–	–	–
24	CP_VALID	CP eFuse 已烧写	–
23:12	THS1_ADC_VAL	THS ADC1 采样值	–
11:0	THS0_ADC_VAL	THS ADC0 采样值	–

6.2.3.4. BROM_SECURE 区域定义

比特位	名称	描述
31:29	–	–
28	SPI_ENC_EN	BROM 读取使用，使能 SPI 总线数据加密功能
27:25	–	–
24	JTAG_LOCK	逻辑组合后连接到 CPU 屏蔽 TDO，关闭 JTAG 调试功能，在安全方案中烧录为 1。
23:21	–	–
20	SDMC_EXT_MUX_DIV	设置 SDMC 外部时钟选择

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

比特位	名称	描述
19:18	SDMC_SMP_PHASE	设置 SDMC 采样时钟的相位
17:16	SDMC_DRV_PHASE	设置 SDMC 驱动时钟的相位
15	CPU_HS_EN	1：BROM 使能 CPU 高速模式，CPU 216 MHz，AXI/AHB 200 MHz
14	CHECKSUM_DIS	1：启动过程中 BROM 不检查固件的 Checksum
13	PLL_FRA0_DIS	1：BROM 不启用 PLL_FRA0
12	PLL_INT1_DIS	1：BROM 不启动 PLL_INT1，USB 不能使用
11	SPI_DMA_DIS	1：BROM 访问 SPI 存储时，不使用 DMA
10	–	–
9	SD/eMMC_DMA_DIS	1：BROM 访问 SD/eMMC 时，不使用 DMA
8	SKIP_SD_PHASE	1：BROM 在访问 PRIMARY、SECONDARY 存储前不访问 SD 卡。设置为 1 影响卡量产。
7:4	SECONDARY	指定 BROM 的次选启动设备
3:0	PRIMARY	指定 BROM 的首选启动设备

6.2.3.5. JTAG 安全保护

JTAG 使能受 SID 控制，用于对芯片做安全保护。图 6–5 : JTAG 安全保护硬件逻辑及使用场景如下：

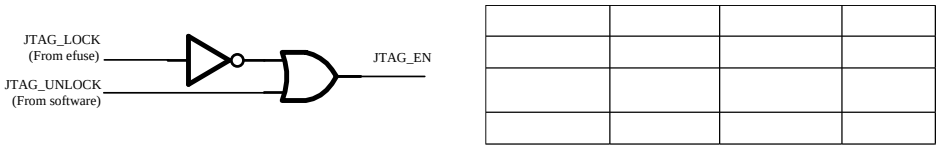


图 6–5 JTAG 安全保护硬件逻辑及使用场景

对于关闭 JTAG 的安全芯片方案，用户可以使用安全密钥进行认证， 认证成功后由 BROM 开启 JTAG_UNLOCK 位来打开 JTAG。

注：

JTAG_UNLOCK 位只在 BROM 中进行读写。BROM 退出时，会配置 BROM_PRIVILEGE_LOCK 为 1，此时对 JTAG_UNLOCK 进行写操作无效。。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

6.2.4. 编程指南

6.2.4.1. 初始化流程

每次系统上电复位后，SID 自动读取所有 eFuse Cell 值到 eFuse Buffer，软件只需读取 eFuse Buffer 对应地址值即可。

> **注：**
>
> 第一次烧写后，在不断电的情况下，eFuse Buffer 值不会自动更新，需通过软件读操作将对应地址的值读到寄存器。

6.2.4.2. eFuse 烧写流程

1. 检查 eFuse 控制寄存器 0x000 EFUSE_CTL 状态位，若为非空闲则等待，若都为空闲则继续第 2 步。状态位包括：
   - EFUSE_WRITE_START
   - EFUSE_READ_START
   - EFUSE_AUTO_STATUS
2. 配置要烧写的 eFuse 地址到 eFuse 地址寄存器 0x004 EFUSE_ADDR。
3. 配置要烧写的 eFuse 数据到 eFuse 写数据寄存器 0x008 EFUSE_WDATA。
4. 在 eFuse 控制寄存器 0x000 EFUSE_CTL 中，配置 EFUSE_WRITE_START 为 1 且在 EFUSE_OP_CODE 位写入 0xA1C，启动烧写流程。
5. 等待 EFUSE_WRITE_START 自动清零，硬件自动关闭 LDO。

为达到双备份目的，要求针对备份地址，即偏移 512bit 地址，重复步骤 1~步骤 5 完成烧写。

6.2.4.3. eFuse 读取流程

1. 检查 eFuse 控制寄存器 0x000 EFUSE_CTL 状态位，若为非空闲则等待，若都为空闲则继续第 2 步。状态位包括：
   - EFUSE_WRITE_START
   - EFUSE_READ_START
   - EFUSE_AUTO_STATUS
2. 配置要读取的 eFuse_cell 地址到 eFuse 地址寄存器 0x004 EFUSE_ADDR 。
3. 在 eFuse 控制寄存器 0x000 EFUSE_CTL 中，配置 EFUSE_READ_START 为 1 且在 EFUSE_OP_CODE 位写入 0xA1C，启动读取流程。
4. 等待 EFUSE_READ_START 自动清零，即可读取 eFuse 读数据寄存器 (0x00C EFUSE_RDATA) 并获得 eFuse 的值。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

6.2.5. 寄存器列表

表 6–4 寄存器列表

偏移地址	缩写	寄存器描述	章节
0x000	EFUSE_CTL	eFuse 控制 （eFuse Control）	0x000 EFUSE_CTL
0x004	EFUSE_ADDR	eFuse 地址 （eFuse Address）	0x004 EFUSE_ADDR
0x008	EFUSE_WDATA	eFuse 写数据 （eFuse Write Data）	0x008 EFUSE_WDATA
0x00C	EFUSE_RDATA	eFuse 读数据 （eFuse Read Data）	0x00C EFUSE_RDATA
0x010	EFUSE_TIMING_LOW	eFuse 时序低位 （eFuse Timing）	0x010 EFUSE_TIMING_LOW
0x080	BROM_PRIV	BROM 特权 （BROM Privilege）	0x080 BROM_PRIV
0x0FC	SID_VER	SID 版本 （SID Version）	0x0FC SID_VER
0x200~0x23F	EFUSE_BUFFER	eFuse 数据 （EFUSE DATA）	0x200~0x23F EFUSE_BUFFER

6.2.6. 寄存器描述

6.2.6.1. 0x000 EFUSE_CTL

默认值：0x00000200			eFuse 控制 （eFuse Control）
位域	类型	默认值	描述
31	–	–	–
29	W1C	0x0	SID_ERROR SID 错误 • 0：正常 • 1：出现过 Busy 时启动读写
28	R/W	0x0	BROM_PRIV_LOCK

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000200			eFuse 控制 （eFuse Control）
位域	**类型**	**默认值**	**描述**
			BROM 特权锁 • 0：BROM 特权可用，软件可以使用相关特权功能。 • 1：BROM 特权关闭，软件无法使用相关特权功能。写 1 后在下次 POR 之前无法清零。 **注：** BROM 特权锁特权包括： • JTAG 解锁功能可使能。 • eFuse SSK 的 CE 运算结果可输出到 SRAM。 BROM 中会配置该位为 1，确保相关特权功能仅在 BROM 中可用，其它软件无法更改；
27:16	WO	0x0	EFUSE_OP_CODE eFuse 操作码 为防止误操作，设置为写入 0xA1C 后允许 eFuse 进行读写操作。
15:13	–	–	–
12:8	RO	0x2	EFUSE_STATUS eFuse 状态机状态 • 1：开机自动读取中 • 2：空闲中 • 4：烧录中 • 8：读取中 • 空闲状态建议同步查询 EFUSE_DATA_READY。 • 写结束状态建议查询 EFUSE_WRITE_START。 • 读结束状态建议查询 EFUSE_READ_START。
7:5	–	–	–
4	R/WAC	0x0	EFUSE_READ_START eFuse 读启动寄存器 • 0：无操作 • 1：启动 eFuse 读操作 写该位时与 EFUSE_OP_CODE = 0xA1C 同时写入。写 1 启动 eFuse 读取，读取完该位自动清零。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000200			eFuse 控制 （eFuse Control）
位域	**类型**	**默认值**	**描述**
3:1	–	–	–
0	R/WAC	0x0	EFUSE_WRITE_START eFuse 写启动寄存器： • 0：无操作 • 1：启动 eFuse 写操作 写该位时与 EFUSE_OP_CODE = 0xA1C 同时写入。 写 1 启动 eFuse 烧写，烧写完该位自动清零。期间会自动开关 LDO，无需配置开关 LDO。

6.2.6.2. 0x004 EFUSE_ADDR

默认值：0x00000000			eFuse 地址 （eFuse Address）
位域	**类型**	**默认值**	**描述**
31:9	–	–	–
8:0	R/W	0x0	EFUSE_ADDR eFuse 读写地址寄存器 全空间读写，以 byte 为单位，由于读写操作按照 32 位进行，低两位需配置为 0。

6.2.6.3. 0x008 EFUSE_WDATA

默认值：0x00000000			eFuse 写数据 （eFuse Write Data）
位域	**类型**	**默认值**	**描述**
31:0	R/W	0x0	EFUSE_WDATA eFuse 写数据寄存器 **注：** eFuse 为单次烧写，默认未烧写时为 0，烧写为 1，后无法再变回为 0。该字段对应的 bit 为 1，则对应进行烧写，对应的 bit 为 0 即不进行烧写， 32 bits 可以分多次针对不同 bits 进行烧写。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

6.2.6.4. 0x00C EFUSE_RDATA

默认值：0x00000000			eFuse 读数据（eFuse Read Data）
位域	类型	默认值	描述
31:0	RO	0x0	EFUSE_RDATA eFuse 读数据寄存器

6.2.6.5. 0x010 EFUSE_TIMING_LOW

默认值：0x04026536			eFuse 时序低位（eFuse Timing）
位域	类型	默认值	描述
31:24	R/W	0x04	TAEN_RD 单次读数据处理时间所占的时钟周期数
23:16	R/W	0x02	TRD 单次读数据时高电平所占的时钟周期数
15:8	R/W	0x65	TAEN_PGM 单次写数据处理时间所占的时钟周期数
7:0	R/W	0x36	TPGM 单次写数据时高电平所占的时钟周期数

6.2.6.6. 0x080 BROM_PRIV

默认值：0x00000000			BROM 特权（BROM Privilege）
位域	类型	默认值	描述
31:1	–	–	–
0	R/W	0	JTAG_UNLOCK JTAG 解锁 • 0：未启动 JTAG 解锁 • 1：启动 JTAG 解锁 仅在 JTAG_LOCK = 1 时有效。 优先级高于 JTAG_LOCK（来自 eFuse），解锁功能仅在 BROM_PRIVILEGE_LOCK 为 0 时才有效。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			BROM 特权（BROM Privilege）
位域	类型	默认值	描述
			若 BROM_PRIVILEGE_LOCK 为 1，即使 JTAG_UNLOCK 设置为 1，也不会启动 JTAG 解锁。

6.2.6.7. 0x0FC SID_VER

默认值：0x00000102			SID 版本（SID Version）
位域	类型	默认值	描述
31:0	RO	0x00000102	VERSION 模块版本。 V1.2

6.2.6.8. 0x200~0x23F EFUSE_BUFFER

默认值：0x00000000			eFuse 数据（EFUSE DATA）
位域	类型	默认值	描述
31:0	RO	0x0	EFUSE_BUFFER eFuse 数据 芯片上电复位，会将 eFuse Cell 值同步到该区域。 配置寄存器发起读操作，也会将 eFuse 值同步到该区域。

6.3. Data Check Engine (DCE)

Data Check Engine (DCE) 模块是一种用于数据校验的硬件加速器，主要支持 CRC 校验和 SUM 校验功能。

6.3.1. 特性说明

- 可单独作为 AHB master 取数，支持最大 256M 数据进行计算
- 使用 CRC-32 多项式：0x4C11DB7

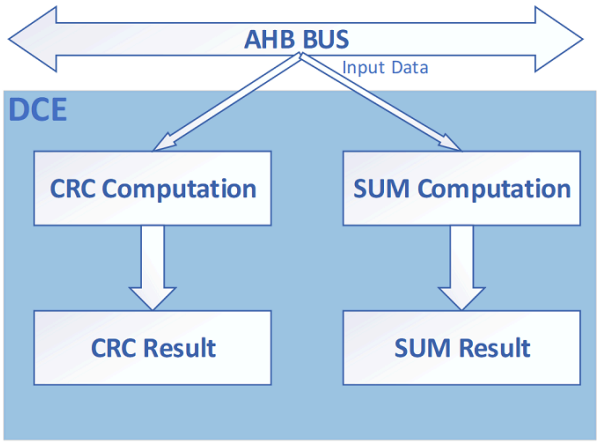
$$X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$$

- 支持可配置的 CRC 初始值
- 支持可配置的 CRC 结果异或值
- 支持 CRC 输入输出数据反转可配置

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- SUM 校验输出为累加结果保留低 32 位
- 起始地址需 4 byte 对齐，数据长度以 byte 为单位

6.3.2. 原理框图



6.3.3. 功能描述

6.3.3.1. CRC 校验算法配置

CRC 校验算法的配置流程如下：

1. 通过 CRC _INIT 寄存器配置 CRC 初始值。
2. 通过 CRC _XOROUT 寄存器配置 CRC 结果异或值，输出结果前会与该值进行异或。
3. 选择硬件取数的方式处理数据。

   硬件取数以 Word 为单位进行取数，当尾端数据长度不足一个 Word 时，硬件会自动补零。
4. 通过寄存器配置是否反转输入数据，反转方式选择如下：

   - Byte In Word 每个 Word 里的 Byte 进行逆序处理。由于硬件以 Word 为单位取数，高位在前低位在后。所以使用需要注意，如需以 Byte 从低到高顺序运算，需要配置该反转。
   - Bit In Word 每个 Word 里的 Bit 进行逆序处理。
   - Bit In Byte 每个 Byte 里的 Bit 进行逆序处理，

   三种反转方式的优先级为 Byte In Word > Bit In Word > Bit In Byte。
5. 通过寄存器配置是否反转输出数据，反转方式选择如下：

   - Bit In Word 每个 Word 里的 Bit 进行逆序处理。

注：

关于输入输出反转运算的详细说明，可查看 CRC 算法的反转运算说明。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

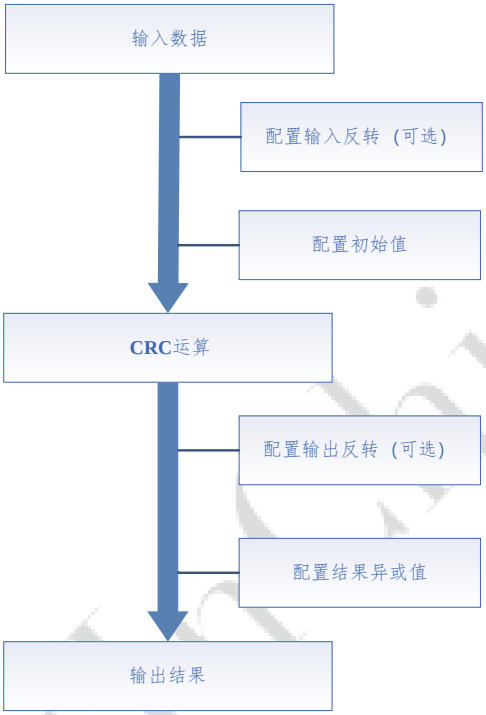


图 6–6 CRC 使用流程

6.3.3.2. CRC 算法的反转运算说明

CRC 算法涉及的反转运算的详细说明如下图所示：

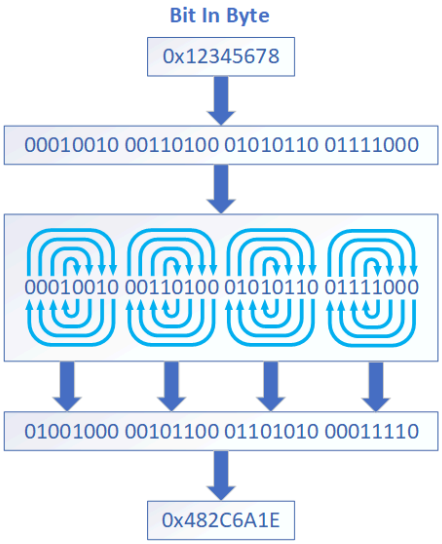


图 6–7 Bit In Byte 反转

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

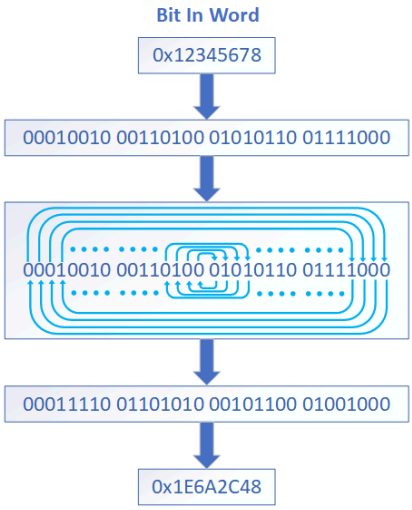


图 6–8 Bit In Word 反转

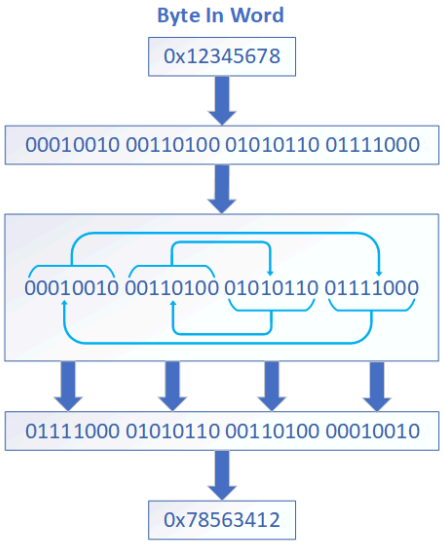


图 6–9 Byte In Word 反转

6.3.3.3. CRC 硬件运算流程

CRC 硬件运算流程如下所示：

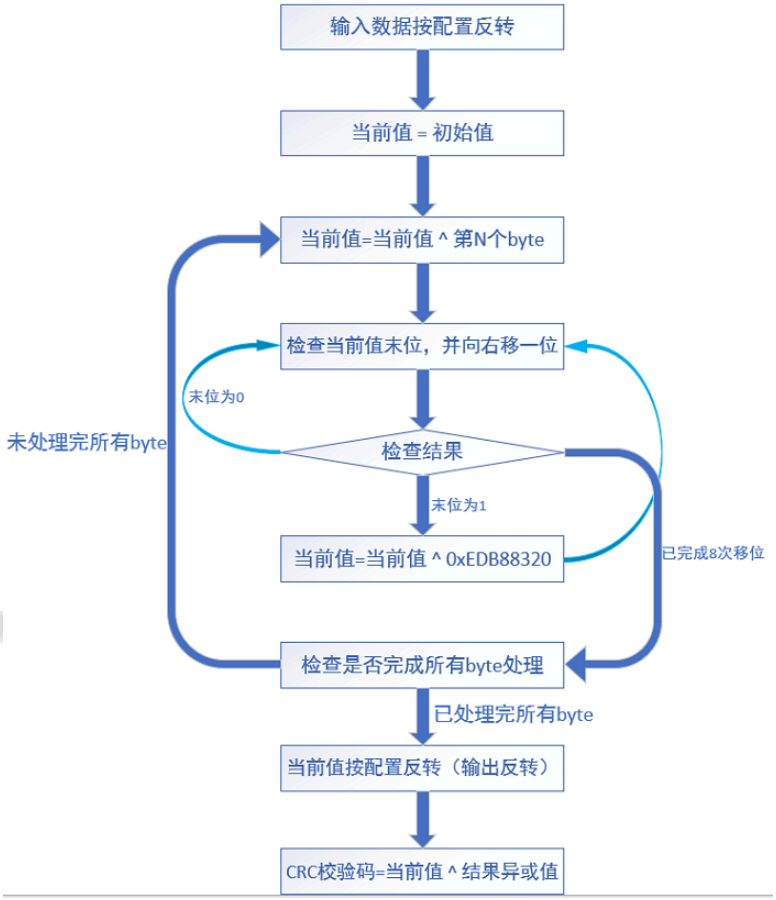
1. 按照配置，选择对输入数据进行输入反转或不反转。
2. 设置 CRC 初始值。

   CRC 初始值为当前值，当前值长度固定为 32 bit。
3. 将输入数据的第 n 个 Byte 与当前值进行异或，从第一个 Byte 开始。
4. 检查当前值最低 bit 是否为 1，并将该值向右移一位，末位丢弃，高位补零。
5. 如果最低 bit 为 0，重复 4 步；否则，将当前值与 0xEDB88320 进行异或。
6. 重复 4 到 5 步，直到总共完成了 8 次移位。至此，输入数据的第一个 Byte 处理完毕。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

7. 重复 4 – 6 步，对输入数据的下一个 Byte 进行相同的处理，直至所有数据全部处理完毕。
8. 所有数据处理完毕后，当前值按照配置进行输出反转或不反转。
9. 将经过上面所有步骤处理后的当前值与结果异或值进行异或运算得出的即为最终的 CRC 校验码。

   图示如下：



6.3.3.4. SUM 校验

SUM 校验指的是把所需校验的数据段以 32 bits 为单位进行累加运算，并将累加结果保留低 32 位输出。尾端数据如不足 32 bits 则进行高位补零。

SUM 算法描述：

1. 以 32 bits 为单位按顺序依次对数据进行累加和。
2. 如第 32 bits 遇到需要进位的情况时，该次累加结束后丢弃进位得来的第 33 bits，下次加法运算依旧以 32 bits 进行。
3. 最终运算结果即为保留低 32 位的累加结果。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

6.3.4. 寄存器列表

表 6–5 寄存器列表

偏移地址	缩写	寄存器描述	章节
0x000	DCE_CTL	DCE 控制 （DCE_CTL）	0x000 DCE_CTL
0x004	DCE_CFG	DCE 配置 （DCE_CFG）	0x004 DCE_CFG
0x008	IRQ_EN	中断控制寄存器（IRQ_EN）	0x008 IRQ_EN
0x00C	IRQ_STA	中断状态寄存器（IRQ_STA）	0x00C IRQ_STA
0x0010	DATA_ADDR	地址配置寄存器（DATA_ADDR）	0x0010 DATA_ADDR
0x0014	DATA_LEN	数据长度配置寄存器（DATA_LEN）	0x0014 DATA_LEN
0x0040	DCE_RESULT	结果存储寄存器（DCE_RESULT）	0x0040 DCE_RESULT
0x080	CRC_CFG	CRC 配置寄存器（CRC_CFG）	0x080 CRC_CFG
0x0084	CRC_INIT	初始值配置寄存器（CRC_INIT）	0x0084 CRC_INIT
0x0088	CRC_XOROUT	结果异或值配置寄存器（CRC_XOROUT）	0x0088 CRC_XOROUT
0x00C0	CRC_RESULT	结果存储寄存器（CRC_RESULT）	0x00C0 CRC_RESULT
0x0140	SUM_RESULT	结果存储寄存器（SUM_RESULT）	0x0140 SUM_RESULT
0x0FFC	VERSION	DCE 版本号 （VERSION）	0x0FFC VERSION

6.3.5. 寄存器描述

6.3.5.1. 0x000 DCE_CTL

默认值：0x00000000			DCE 控制 （DCE_CTL）
位域	类型	默认值	描述
31:1	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			DCE 控制（DCE_CTL）
位域	类型	默认值	描述
0	R/WAC	0x0	CAL_ENABLE 计算使能 • 0：关闭 • 1：计算使能 当计算完成并将结果存到 Result 寄存器后，该位自动清零。可以通过读取该位表示 DCE 模块当前状态。

6.3.5.2. 0x004 DCE_CFG

默认值：0x00000000			DCE 配置（DCE_CFG）
位域	类型	默认值	描述
31:4	–	–	–
3:0	R/W	0x0	MODE_SEL 算法选择 • 0x01：CRC • 0x02：SUM

6.3.5.3. 0x008 IRQ_EN

默认值：0x0			中断控制寄存器（IRQ_EN）
位域	类型	默认值	描述
31:27	–	–	–
26	R/W	0x0	DATA_LEN_ERR_EN 数据长度错误中断使能 • 0: 关闭中断 • 1: 打开中断
25	R/W	0x0	ADDR_ALIG_ERR_EN 地址非对齐错误中断使能 • 0: 关闭中断 • 1: 打开中断

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x0			中断控制寄存器（IRQ_EN）
位域	类型	默认值	描述
24	R/W	0x0	ADDR_REQ_ERR_EN 请求地址错误中断使能 • 0: 关闭中断 • 1: 打开中断
23:2	–	–	–
1	R/W	0x0	SUM_FINISH_EN SUM 运算完成中断使能 • 0: 关闭中断 • 1: 打开中断
0	R/W	0x0	CRC_FINISH_ENCRC 运算完成中断使能 • 0: 关闭中断 • 1: 打开中断

6.3.5.4. 0x00C IRQ_STA

默认值：0x0			中断状态寄存器（IRQ_STA）
位域	类型	默认值	描述
31:27	–	–	–
26	R/W	0x0	DATA_LEN_ERR_STA 数据长度错误中断状态 • 0: 无中断 • 1: 有中断，表示数据长度过长或者跨界到非法地址 **注：** 若使能输入数据源为 DATA_INPUT 寄存器，则此位无效。
25	R/W	0x0	ADDR_ALIG_ERR_STA 地址非对齐错误中断状态 • 0: 无中断 • 1: 有中断，表示请求的地址为非 4 byte 对齐

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x0			中断状态寄存器（IRQ_STA）
位域	**类型**	**默认值**	**描述**
24	R/W	0x0	ADDR_REQ_ERR_STA 请求地址错误中断状态 • 0: 无中断 • 1: 有中断，表示请求的地址为非法地址
23:2	–	–	–
1	R/W1C	0x0	SUM_FINISH_STA SUM 运算完成中断状态 • 0: 无中断 • 1: 有中断，表示运算完成
0	R/W1C	0x0	CRC_FINISH_STA CRC 运算完成中断状态 • 0: 无中断 • 1: 有中断，表示运算完成

6.3.5.5. 0x0010 DATA_ADDR

默认值：0x0			地址配置寄存器（DATA_ADDR）
位域	**类型**	**默认值**	**描述**
31:0	R/W	0x0	从存储空间获取数据的起始地址，需要 4 byte 对齐，即最低两位需要为 0

6.3.5.6. 0x0014 DATA_LEN

默认值：0x0			数据长度配置寄存器（DATA_LEN）
位域	**类型**	**默认值**	**描述**
31:29	–	–	–
28:0	R/W	0x0	从存储空间获取数据的长度，最长支持 256 MByte

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

6.3.5.7. 0x0040 DCE_RESULT

默认值：0x0			结果存储寄存器（DCE_RESULT）
位域	类型	默认值	描述
31:0	RO	0x0	运算结果（如果两种算法同时使能，此处存储 CRC 算法的结果）

6.3.5.8. 0x080 CRC_CFG

默认值：0x0000000b			CRC 配置寄存器（CRC_CFG）
位域	类型	默认值	描述
31:4	–	–	–
3	R/W	0x1	INPUT_BYTE_IN_WORD_REV 输入数据反转，指输入数据每个 Word 里的 Byte 进行逆序处理。由于硬件以 Word 为单位取数，高位在前低位在后。如需按数据的 Byte 从低到高顺序运算，需要配置该反转 • 0：禁用 • 1：使能 输入数据三种反转的优先级为 Byte In Word > Bit In Word > Bit In Byte
2	R/W	0x0	INPUT_BIT_IN_WORD_REV 输入数据反转，指输入数据每个 Word 里的 Bit 进行逆序处理 • 0：禁用 • 1：使能 输入数据三种反转的优先级为 Byte In Word > Bit In Word > Bit In Byte
1	R/W	0x1	INPUT_BIT_IN_BYTE_REV 输入数据反转，指输入数据每个 Byte 里的 Bit 进行逆序处理 • 0：禁用 • 1：使能 输入数据三种反转的优先级为 Byte In Word > Bit In Word > Bit In Byte
0	R/W	0x1	OUTPUT_BIT_IN_WORD_REV 输出数据反转，指输出数据每个 Word 里的 Bit 进行逆序处理 • 0：禁用 • 1：使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

6.3.5.9. 0x0084 CRC_INIT

默认值：0xFFFFFFFF			初始值配置寄存器（CRC_INIT）
位域	类型	默认值	描述
31:0	R/W	0xFFFFFFFF	CRC 初始值

6.3.5.10. 0x0088 CRC_XOROUT

默认值：0xFFFFFFFF			结果异或值配置寄存器（CRC_XOROUT）
位域	类型	默认值	描述
31:0	R/W	0xFFFFFFFF	CRC 结果异或值

6.3.5.11. 0x00C0 CRC_RESULT

默认值：0x0			结果存储寄存器（CRC_RESULT）
位域	类型	默认值	描述
31:0	RO	0x0	CRC 运算结果

6.3.5.12. 0x0140 SUM_RESULT

默认值：0x0			结果存储寄存器（SUM_RESULT）
位域	类型	默认值	描述
31:0	RO	0x0	SUM 运算结果

6.3.5.13. 0x0FFC VERSION

默认值：0x00000100			DCE 版本号（VERSION）
位域	类型	默认值	描述
31:0	RO	0x00000100	Version 模块版本号，采用 BCD 码显示，V1.0

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

7. 启动

7.1. 模块概述

BROM 程序的位置从地址 0x3000_0000 处开始，大小 64 KB。芯片在上电复位 (POR) 时即从地址 0x3000_0000 处按照选择的启动类型和启动介质开始执行启动指令。

BROM 程序支持下列启动相关功能：

启动介质选择	• eMMC • SD Card • SD Card FAT32/exFAT • SPI NAND • SPI NOR
eFuse 中的 BROM 配置区域	可通过 eFuse 选择启动介质，在没有设置相关选择 eFuse 的情况下，则按照固定顺序逐个介质尝试启动
烧录和升级模式	• 通过 UART 烧录 • 通过 SD 卡升级

7.2. 选择启动类型

BROM 支持以下两种启动类型：

- 正常启动模式，即 BROM 从启动介质中读取并且执行镜像文件进行启动。

  正常启动支持多种启动介质，用户可以通 过 eFuse 烧录选择固定的启动介质。如果用户未设置固定启动介质，BROM 则按照 图 7–1 : BROM 默认的启动介质顺序 执行启动。关于如何设置固定启动介质，请查看使用 eFuse 配置启动介质。

- 启动进入升级模式，即 BROM 在所有启动介质中均未读取到合法的启动镜像则进入升级模式。

  满足下列任意条件，可直接启动进入升级模式：

  - 从所有的启动介质，都没有读取到合法的启动镜像

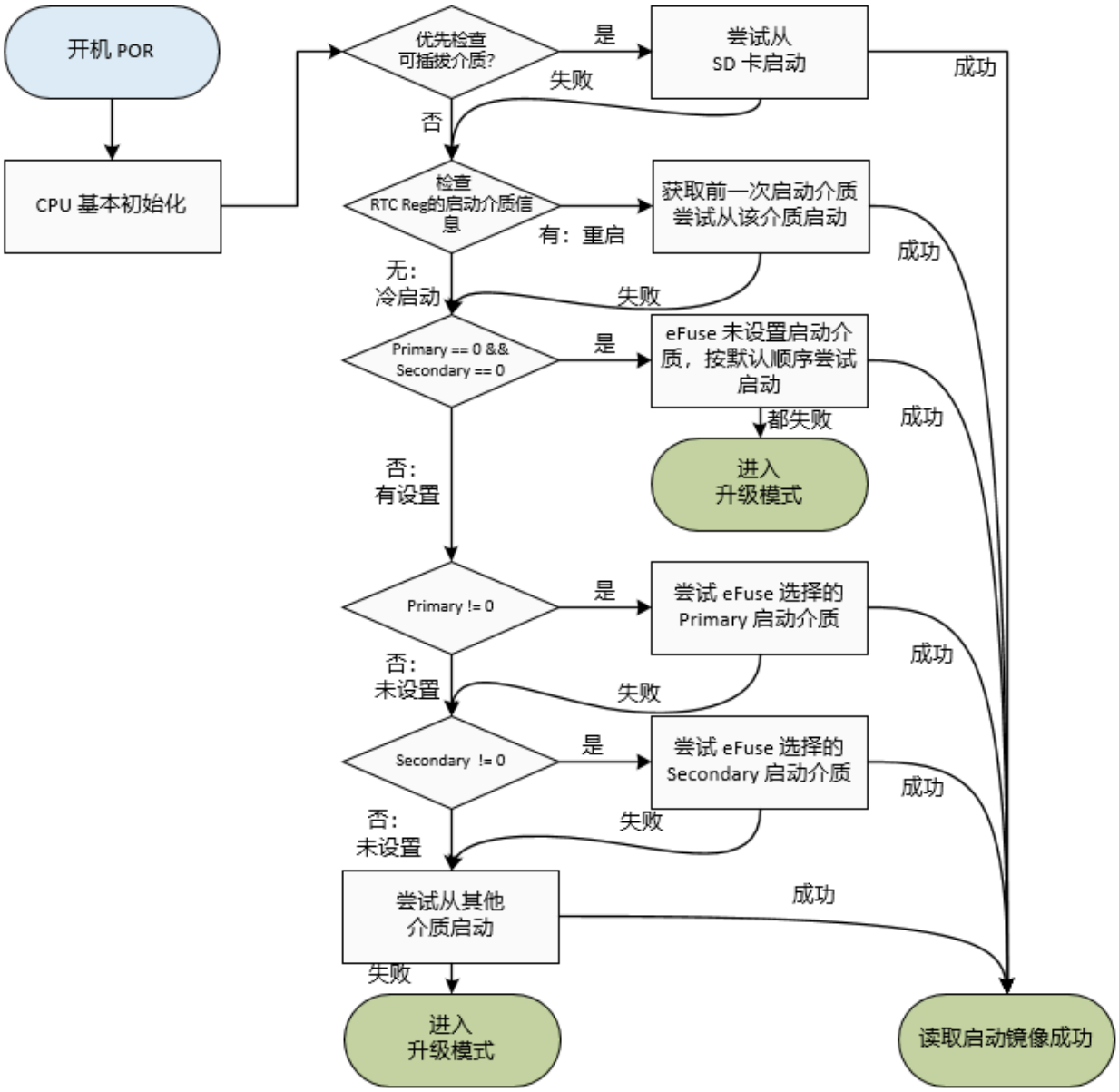
  在升级模式下，用户可通过以下方式烧录升级镜像：

  - UART 升级
  - SiP Flash 烧录：

    直接读写 SiP SPI Flash，仅对合封了 SPI Flash 的芯片有效。在此种模式下，芯片将合封在内部的 SPI Flash 引脚映射到外部，使用特制的板卡，可以使用烧录器对内部的 SPI Flash 进行直接读写。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

7.2.1. 启动流程



7.3. 正常启动模式

7.3.1. 启动介质选择

BROM 程序支持多种启动介质。在上电复位之后，BROM 需要确定从哪种启动介质开始启动：

1. 通过 eFuse 配置固定的启动介质

   用户可以 正常启动模式选择固定的启动介质。启动过程中，BROM 会读取对应的配置， 尝试从选择的启动介质进行启动。

2. 默认的启动介质顺序

   如用户未通过 eFuse 配置固定的启动介质，BROM 则按照以下默认的启动介质顺序逐一尝试启动：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

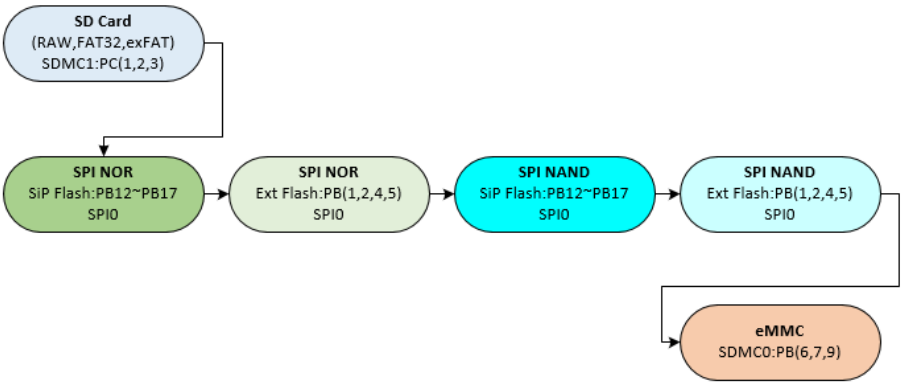


图 7–1 BROM 默认的启动介质顺序

注：

如果 BROM 通过上述启动介质顺序执行启动时均未成功，则需要进入烧录和升级模式。

7.3.2. 使用 eFuse 配置启动介质

用户可以通过配置 eFuse 中的以下三个字段选择固定的启动介质：

1. **Primary boot device**：通过 eFuse 烧录首选启动介质。如果该字段值有效，BROM 直接尝试从该字段指定的介质读取固件。
2. **Secondary boot device**：通过 eFuse 烧录次选启动介质。

   如果用户没有烧录有效的 **Primary boot device** 值，或者 BROM 从 **Primary boot device** 读取固件失败且 **Secondary boot device** 值有效，则尝试从 **Secondary boot device** 指定的介质读取固件。

关于 eFuse 中的启动配置，可查看 正常启动模式。

BROM 在执行启动的过程中，对相同的启动介质只会尝试一次启动。

如果 **Primary boot device** 和 **Secondary boot device** 都设为 SD 卡启动且未设置跳过前置的 SD 卡启动阶段，按照流程，BROM 会尝试从 SD 卡启动三次，但是 BROM 会记录是否已经尝试过通过该介质启动，如果尝试过则不再尝试。因此实际上 BROM 只会尝试一次，后续两次直接被跳过。启动介质的有效值如下表所示：

表 7–1 启动介质索引值

eFuse 启动介质索引值	描述
0x0	默认值，没有配置启动介质，按照固定顺序尝试
0x1	SPI NAND（仅支持 SPI0 启动）
0x2	SPI NOR（仅支持 SPI0 启动）
0x3	SDMC0 eMMC

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 7–1 启动介质索引值 (续)

eFuse 启动介质索引值	描述
0x4	SDMC1 SD Card
其它	无效值，等同 0x0

7.3.3. eFuse 中的 BROM 配置区域

下表列出了 eFuse 中的 64 位 BROM 配置位域及其详情：

表 7–2 BROM 配置位域

位域	字段	默认值	功能描述
31:30	–	–	–
29:	JTAG disable	0	• 0: JTAG 使能 • 1: JTAG 关闭
28:27	SPI_ENC_EN	0	• 0: SPI ENC 不使能 • 1: SPI ENC 使能
26:22	PLL CINT CFG	0	PLL_CINT’s bit[6:2]
21	PLL CINT enable	0	• 0: 不通过 eFuse 配置 CINT 的值 • 1: 通过 eFuse 配置 CINT 的值
20	SDMC ext_mux_div	0	• 0: Ext DIV 为 1 • 1: Ext DIV 为 2
19:18	SDMC sample phase	0	• 0: 0 度 • 1：90 度 • 2：180 度 • 3：270 度
17:16	SDMC drive phase	0	• 0: 180 度 • 1：270 度 • 2：0 度 • 3：90 度

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 7–2 BROM 配置位域 (续)

位域	字段	默认值	功能描述
15	CPU High speed enable	0	• 0: BROM CPU 24 MHz，AXI/AHB 60 MHz • 1: BROM 设置 CPU 216 MHz，AXI/AHB 200 MHz
14	Checksum disable	0	• 0: BROM 校验所读取镜像的 32 位累加校验和 • 1: BROM 不校验所读取镜像的 32 位累加校验和
13	PLL_FRA0 disable	0	• 0: PLL_FRA0 使能，默认输出频率 1008 MHz • 1: PLL_FRA0 关闭
12	PLL_INT1 disable	0	• 0: PLL_INT1 使能，默认输出频率 1.2 GHz • 1: PLL_INT1 关闭
11	SPI DMA disable	0	• 0: SPI NOR/SPI NAND 读取数据时，使用 DMA • 1: SPI NOR/SPI NAND 读取数据时，不使用 DMA
10	dce_dis	0	• 0: BROM 启动时，使用 DCE 来校验启动镜像 • 1: BROM 启动时，不使用 DCE 来校验启动镜像
9	SD/eMMC DMA disable	0	• 0: SD/eMMC 读取数据时，使用 DMA • 1: SD/eMMC 读取数据时，不使用 DMA
8	Skip SD Phase	0	• 0: 启动时，总是先尝试从 SD card 启动。用于量产或开发调试 • 1: 启动时，直接按照选定的启动介质进行启动
7:4	Secondary boot device	0	次选启动介质，具体值参考启动介质索引值
3:0	Primary boot device	0	首选启动介质，具体值参考启动介质索引值

7.3.4. SD/eMMC 启动

芯片包含下列 SDMC 控制器：

- SDMC0：设计对接 eMMC、SDIO 设备
- SDMC1：设计对接 SD 卡

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

介质	描述
eMMC	• 支持 eMMC1.2 ~eMMC 5.1 • 默认使用 26 MHz 和 1 线模式读取数据，如果读取数据过程失败，则尝试 6 MHz 和 1 线模式 • 从 UDA 分区启动，不支持从 eMMC 的两个 Boot partition 分区启动
SD Card	• 支持 SD 1.0 以及以后的版本 • 工作电压 3.3 V • 默认使用 25 MHz 和 1 线模式启动，如果读取数据过程失败，则尝试 6 MHz 和 1 线模式 • 主要用于开发者调试、工厂量产和升级

注：

此处标记的接口频率（25 MHz、26 MHz）为协议中指明的 Legacy 模式的频率，实际工作频率以除频结果为准，通常在 24 MHz 附近。

7.3.4.1. 分区和启动流程

固件在 SD/eMMC 中统一使用 GPT (GUID Partition Table) 的方式进行分区存储，默认的存储分区排列如下表。

分区	用途
SPL0	bootloader.bin
SPL1	bootloader.bin
…	…

本节只讨论 BROM 涉及的前两个分区，其中保存了第一级引导程序及其备份程序，大小固定为每个分区 128 KB。后续分区及内容可以根据不同的应用进行规划 。

区域	大小/位置	备注
GPT Header	17 KB (LBA0 ~ LBA33)	–
Partition0	128 KB (LBA34 ~ LBA289)	BROM 尝试从 SD/eMMC 启动时，首先从 LBA34 的位置读取数据尝试启动，如果失败，则再尝试从 LBA290 的位置读取数据。 Logic Block 大小 512 Byte。
Partition1	128 KB (LBA290 ~ LBA545)	
…	…	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

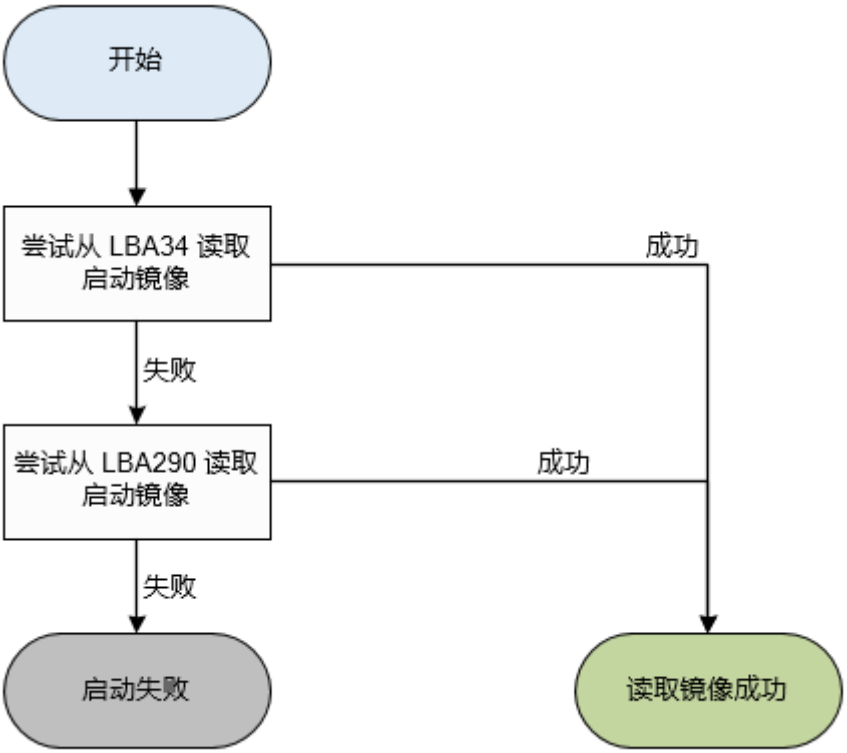


图 7–2 SD/eMMC 读取启动固件

7.3.4.2. SDMC 控制器初始化

在 SD/eMMC 启动的过程中，首先要对控制器进行初始化，主要几个步骤：

1. Pinmux 配置
2. SDMC 时钟频率初始化
3. SDMC 内部 DMA 初始化
4. Reset SDMC（FIFO/DMA）
5. 清除 SDMC 内部中断，配置 DMA trigger threshold

7.3.4.3. SD/eMMC 类型识别

在进行数据读取前，首先要识别当前插入的卡的类型。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

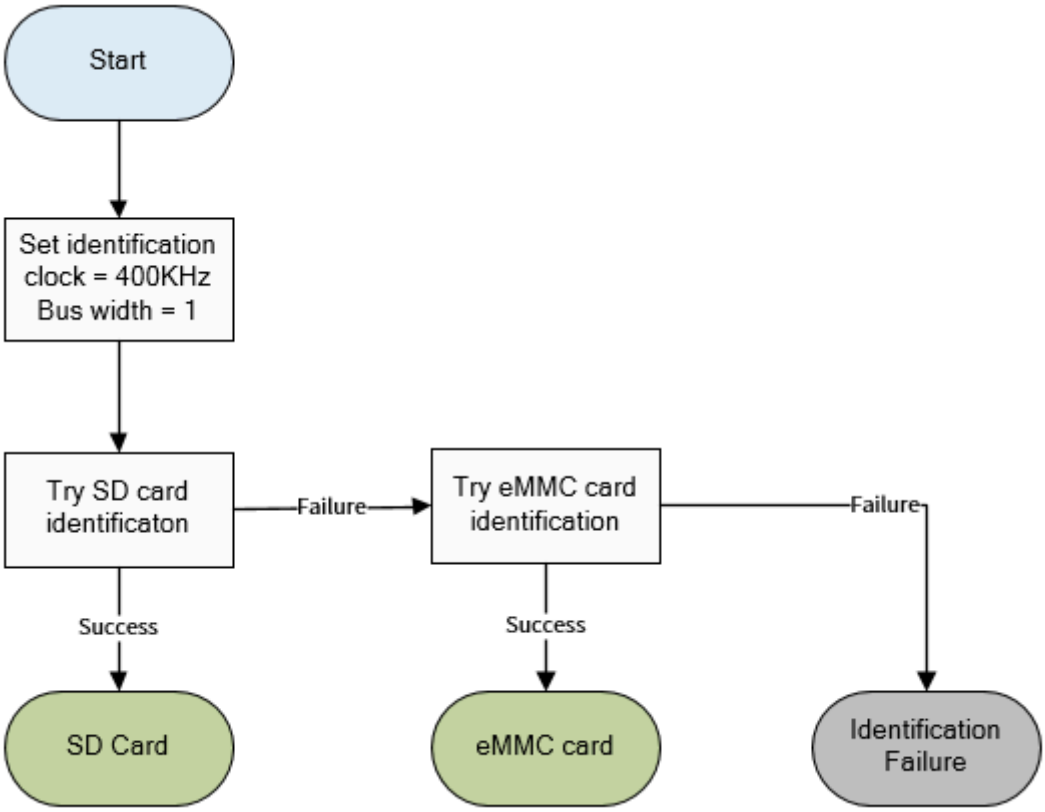


图 7–3 SD/eMMC 的类型识别

在卡识别阶段，设置频率为 400KHz，1–bit 模式。

区分 SD 卡和 eMMC 时，可以在 IDLE 状态下，通过发送 CMD8 的方式进行测试。从 SD v2.0 开始， SD 卡都支持 CMD8（SD_CMD_SEND_IF_COND），而在 IDLE 状态下 eMMC 是不响应该命令的，因此通过测试则表明该卡为 SD。

对于 eMMC，在上电后进入 IDLE 状态（由于没有启用 Boot partition，直接进入 IDLE 状态）， 在该状态下卡响应 CMD1（MMC_CMD_SEND_OP_COND），而 SD 卡是不响应该命令的。

因此通过在 IDLE 状态发送不同命令，可以区分具体是什么类型的卡。

7.3.4.4. SD 卡版本的识别

SD 卡识别需要获取的信息包括：

- 卡的版本类型
- 卡的容量类型 High Capacity or Standard Capacity
- 支持的工作电压

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

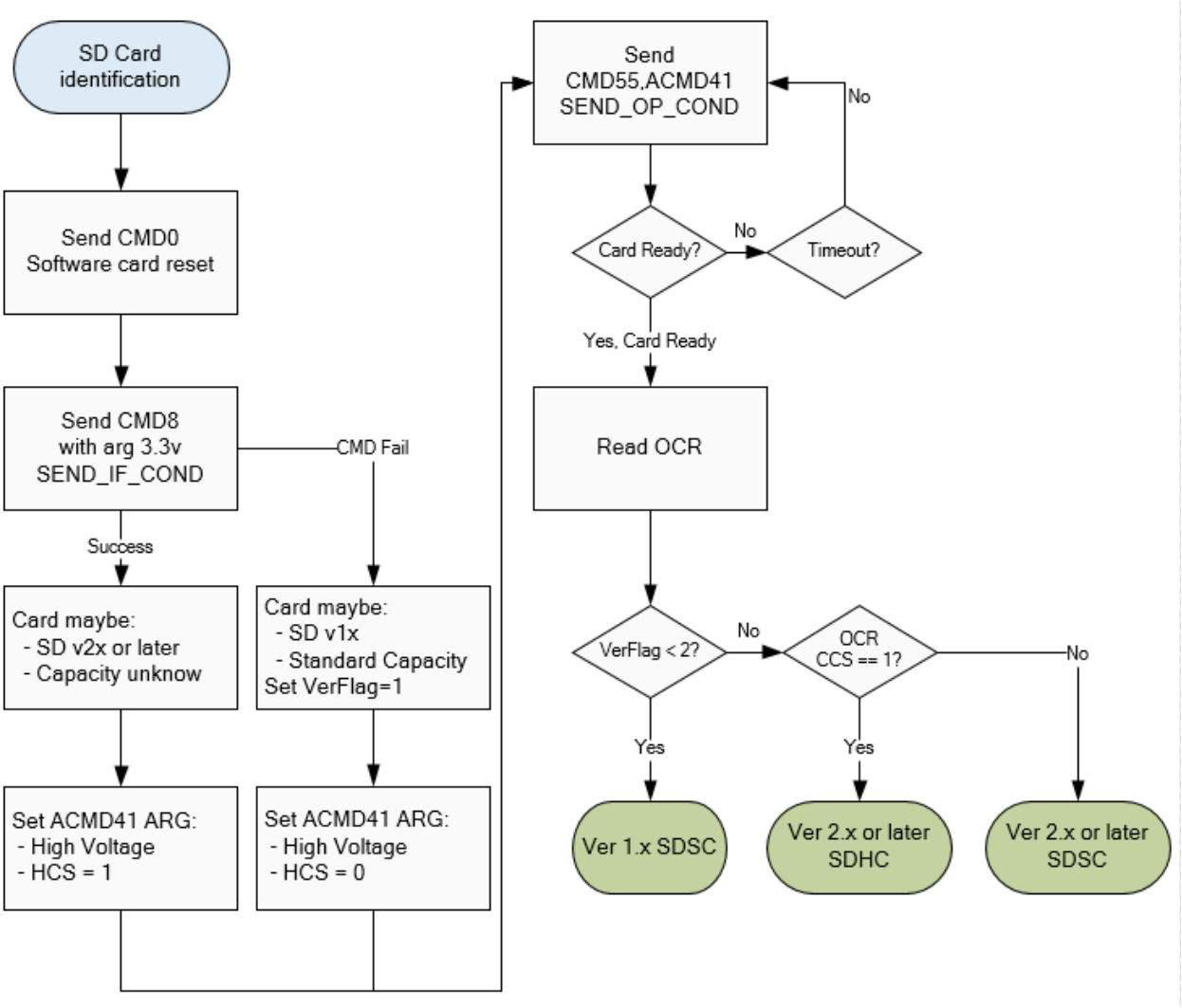


图 7–4 SD 卡能力的识别

ACMD41 (APP_SEND_OP_COND) 是 SD 卡开始初始化的命令，通过该命令让 SD 开始初始化， 并且读取关于 SD 卡的 OCR(Operation Conditions Register)信息。

发送 ACMD41 命令时，需要将参数中的 HCS(Host Capacity Support) 比特设置为 1， 表示 Host 支持 SDHC 或 SDXC。

ACMD41 命令需要重复发送，轮询 SD 卡是否初始化完成。通过读取 Response 中的 OCR 寄存器 Busy 比特位判断是否完成初始化，值为 1 表示已经完成初始化。

最后通过 OCR 寄存器的 CCS(Card Capacity Status) 比特位来判断该卡是否为 SDHC 或 SDXC.

由于 BROM 不会使用 UHS–1 的模式进行工作，因此不需要关心其它信息。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

7.3.4.5. SD 卡的配置

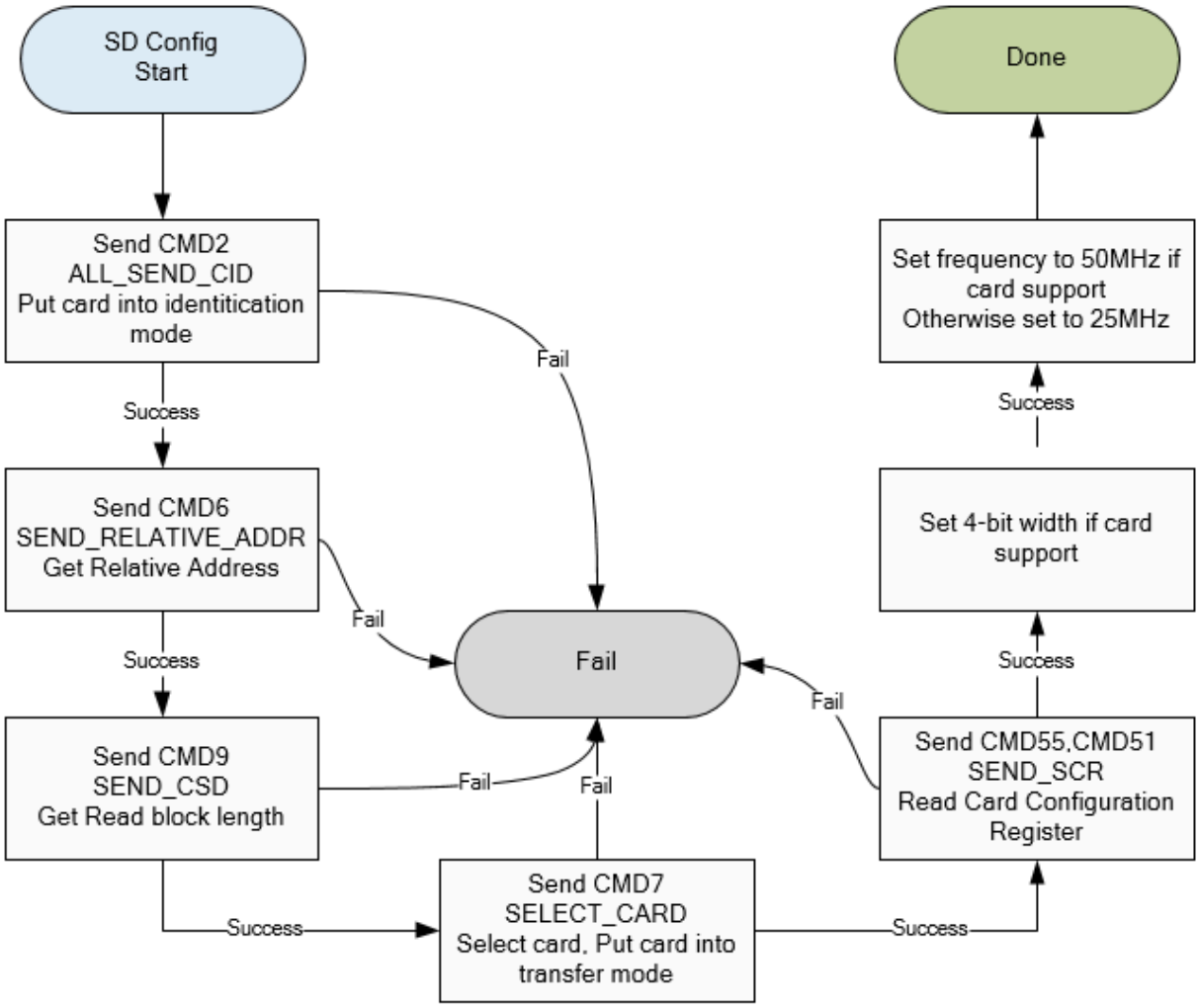


图 7–5 SD 卡配置流程

在进行卡的配置时，主机首先需要发送 CMD2(ALL_SEND_CID)，读取卡的唯一 ID， 并且让卡进入 Identification 状态。

然后主机发送 CMD3(SEND_RELATIVE_ADDR) 命令，询问卡的地址 RCA(Relative Card Address)， 该地址在后续的传输模式中需要使用，用来寻址要进行操作的卡。

发送 CMD9 (SEND_CSD)，让卡发送 CSD(Card Specific Data)。CSD 提供了如何访问 SD 卡的信息， 可以读取到关于 Read block length 相关的信息。

发送 CMD7(SELECT_CARD) 命令，通过指定 RCA 来选中具体的 SD 卡，并且让 SD 卡进入传输模式。

发送 CMD51(APP_SEND_SCR) 命令，读取卡的 SCR(SD Configuration Register) 。 SCR 包含卡实现的 Spec 版本，数据总线位宽等信息。

通过上述配置和命令读取的信息，可以确定当前 SD 卡的确切版本，以及数据位宽信息。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

7.3.4.6. eMMC 的版本识别和配置

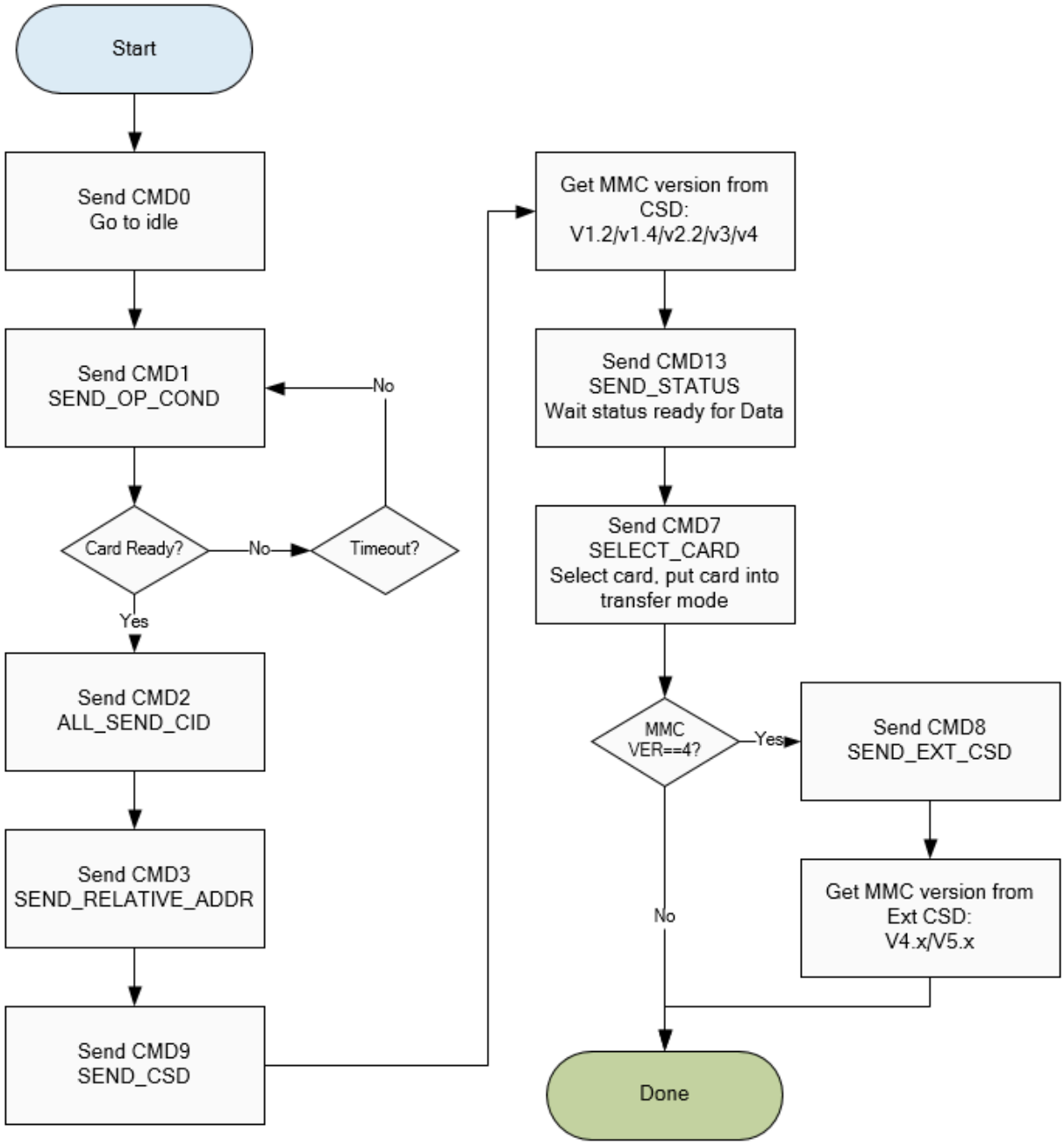


图 7–6 eMMC 的版本识别和配置

上述流程中，获取 eMMC 相关信息的地方有两个，分别是发送 CMD9 读取 CSD 和 CMD8 读取 ext CSD。 通过读取的 CSD 数据，可以获取关于 eMMC 的详细版本信息，以及其它参数。

经过上述流程，eMMC 进行了初始化，并且进入传输模式，可以进行数据传输。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

7.3.4.7. SD/eMMC 数据的读取

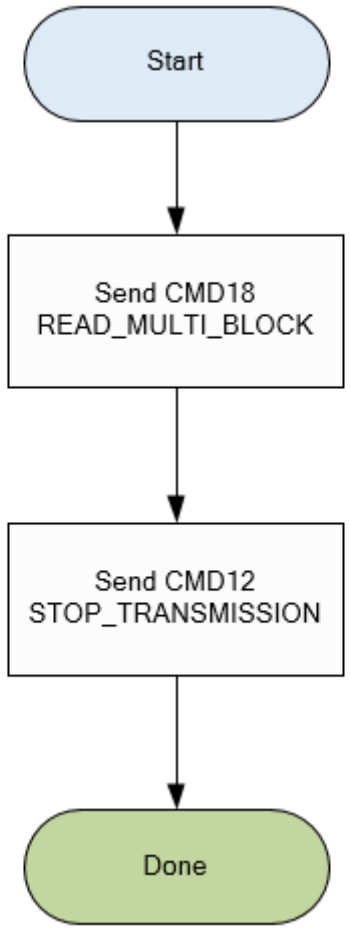


图 7–7 SD/eMMC 数据读取流程

BROM 在读取 SD/eMMC 数据时，以 Block 为单位，统一使用 READ_MULTI_BLOCK 命令进行读取。 读取完毕后发送 STOP_TRANSMISSION 结束读取动作。

在读取数据的策略上，采取兼容性较好的方式进行。

对于 SD 卡，首先通过 400 KHz 和 1 线模式进行识别和配置。数据传输时首先使用 SD Legacy 的方式， 25MHz 和 4 线模式尝试读取。如果读取失败，则转为 6 MHz 和 1 线模式进行读取。如果都失败则放弃。

对于 eMMC，首先通过 400 KHz 和 1 线模式进行识别和配置。数据传输时首先使用 MMC HS 的方式， 26 MHz 和 4 线模式尝试读取。如果读取失败，则转为 6 MHz 和 1 线模式进行读取。如果都失败则放弃。

为了实现的简单和兼容性上的考虑，其它更快的模式并没有使用。

7.3.4.8. eMMC 启动的烧录配置

eMMC 烧录时需要确认下列 CSD 配置。 PARTITION_CONFIG(Ext CSD[179) 应该保持默认配置：

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Reserved	BOOT_ACK	BOOT_PARTITION_ENABLE			PARTITION_ACCESS		
	R/W/E	R/W/E			R/W/E_P		

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

其中：

- BOOT_ACK: 0x0, No boot acknowledge send(Default)
- BOOT_PARTITION_ENABLE: 0x0, Device not boot enabled(Default)
- PARTITION_ACCESS: 0x0, No access to boot partition(Default)

7.3.5. SPI NAND 启动

在常见的方案中，通常划出前面 4 个 Block 作为保存第一级引导程序的保存分区，并且在每个 Block 中保存一份启动镜像。本芯片的实现方案有一些不同。

7.3.5.1. 候选启动块

本方案中同样划出前面 4 个 Block 作为第一级引导程序的保存区域，这个区域称为显式的第一级引导程序保存区域，进行 MTD 分区时，应该将其预留为启动分区。

除此之外，本方案还另外定义了 14 个索引号为偶数的 Block 作为保存第一级引导程序的候选块，这 14 个块从 NAND 的前 1024 个块中随机选取。对于块大小为 128 K 的 NAND，候选块分布在前 128 MB，对于块大小为 256 KB 的 NAND，候选块分布在前 256 MB。

如下表，总共有 18 个 Block 作为候选启动块。这些启动块的使用规则是：

1. 按照优先顺序，逐个检查是否为好块
2. 按顺序选出 4 个好块作为启动块，分别写入启动镜像
3. 对于后 14 个候选块，如果被选中作为启动块，则在烧录启动镜像时将其标记为坏块，以及启动块的特殊标记。标记为坏块的目的是防止其它应用再次使用该块而导致启动镜像被破坏
4. 如果表中所列候选块都是坏块，该 SPI NAND 无法作为启动器件使用

表 7–3 候选启动块列表

优先顺序	Block ID	说明
1	0	启动分区内，优先使用
2	1	启动分区内，优先使用
3	2	启动分区内，优先使用
4	3	启动分区内，优先使用
5	202	如果是好块，则作为启动块。写入启动镜像后标记为坏块
6	32	如果是好块，则作为启动块。写入启动镜像后标记为坏块
7	312	如果是好块，则作为启动块。写入启动镜像后标记为坏块
8	296	如果是好块，则作为启动块。写入启动镜像后标记为坏块

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 7–3 候选启动块列表 (续)

优先顺序	Block ID	说明
9	142	如果是好块，则作为启动块。写入启动镜像后标记为坏块
10	136	如果是好块，则作为启动块。写入启动镜像后标记为坏块
11	392	如果是好块，则作为启动块。写入启动镜像后标记为坏块
12	526	如果是好块，则作为启动块。写入启动镜像后标记为坏块
13	452	如果是好块，则作为启动块。写入启动镜像后标记为坏块
14	708	如果是好块，则作为启动块。写入启动镜像后标记为坏块
15	810	如果是好块，则作为启动块。写入启动镜像后标记为坏块
16	552	如果是好块，则作为启动块。写入启动镜像后标记为坏块
17	906	如果是好块，则作为启动块。写入启动镜像后标记为坏块
18	674	如果是好块，则作为启动块。写入启动镜像后标记为坏块

准备这些候选块的目的是为了增强 BROM 对各种品质 SPI NAND 的适应性。 虽然每个厂商都宣称 NAND 的前面几个块质量较好的，但是在实际应用中，不可避免会遇到前面 4 个候选块为坏块的场景。 并且 NAND 出现坏块时，通常会出现连续坏块。因此这里设计 16 个候选块时， 后 14 个块是随机选取的。这样既兼容传统用法，方便理解和使用，又在坏块较多的情况下，提高可启动的概率。

注：

在正常情况下，烧录的 SPI NAND 前面四个块为好块，则仅有前面四个块被用作启动块， 后面的 14 个候选块不需要保留，可以作为其它用途。

7.3.5.2. Page table

烧录在启动块上的数据并不仅仅是启动镜像，还包括一个页索引表 (Page Table)。Page Table 的目的是构建一个全局的索引表，通过该索引表，可以找到保存 4 个备份镜像的所在 Page 的地址 (PA)。 这样 BROM 一旦读取到正确的 Page Table，就可以直接读取到 4 个备份镜像，减少尝试的时间。

Page Table 总是保存在启动块的 Page0 中，启动镜像则从 Page1 开始保存。启动块中数据的分布如下。

表 7–4 启动块中的 Page 数据

页号	内容
Page 0	保存 Page Table(2 KB) 的数据，仅使用前 2KB

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 7–4 启动块中的 Page 数据 (续)

页号	内容
Page 1	启动镜像的第 1 个 2 KB/4 KB 数据
Page 2	启动镜像的第 2 个 2 KB/4 KB 数据
Page 3	启动镜像的第 3 个 2 KB/4 KB 数据
Page 4	启动镜像的第 4 个 2 KB/4 KB 数据
……	……

⚠ 警告

- 本方案只支持 Page 大小为 2KB 和 4KB 的 SPI NAND。
-

Page Table 中包括一个头信息，以及 4 个备份镜像所在的页地址。其格式如下表所示。

表 7–5 Page table 的格式

区域	名字	字节	备注
Head	Magic	4	“AICP” 标记
	Entry Count	4	启动镜像占用的 Page 数量
	Page Size	2	当前 SPI NAND 的 Page Size
	Page per block	1	当前 SPI NAND 中，一个块的页数量
	Blocks	1	当前 SPI NAND 的块数量
	Planes	1	当前 SPI NAND 的 Plane 数量
	Padding	7	–
Data Page1	Block0 Page 1 Address	4	启动镜像第 1 个 Page 数据保存在 Block0 中的 Page 地址
	Block1 Page 1 Address	4	启动镜像第 1 个 Page 数据保存在 Block1 中的 Page 地址
	Block2 Page 1 Address	4	启动镜像第 1 个 Page 数据保存在 Block2 中的 Page 地址
	Block3 Page 1 Address	4	启动镜像第 1 个 Page 数据保存在 Block3 中的 Page 地址

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 7–5 Page table 的格式 (续)

区域	名字	字节	备注
	Page 1 Data Checksum	4	启动镜像第 1 个 Page 数据的 32 位校验和
Data Page2	Block0 Page 2 Address	4	启动镜像第 2 个 Page 数据保存在 Block0 中的 Page 地址
	Block1 Page 2 Address	4	启动镜像第 2 个 Page 数据保存在 Block1 中的 Page 地址
	Block2 Page 2 Address	4	启动镜像第 2 个 Page 数据保存在 Block2 中的 Page 地址
	Block3 Page 2 Address	4	启动镜像第 2 个 Page 数据保存在 Block3 中的 Page 地址
	Page 2 Data Checksum	4	启动镜像第 2 个 Page 数据的 32 位校验和
…	…	…	…

注：

此处中的 Block0、Block1、Block2、Block3 是指选定的四个候选块。

使用 Page Table，除了可以快速定位 4 个启动备份镜像在 SPI NAND 中的位置，还可以提高启动成功的概率。 在极端情况下，可能出现四个启动块都在使用过程中变成坏块的情况，BROM 无法从单个块中读取完整启动镜像。 借助 Page Table 的信息，只要不出现同一块镜像数据在 4 个启动块中都出现损坏的情况， 此时 BROM 还可以通过组织分布在不同启动块中没有损坏的 Page 数据，重建出一个完整的启动镜像。

7.3.5.3. SPI NAND 的启动流程

SPI NAND 启动时，首先按照上述的候选块顺序，逐个尝试从候选启动块读取 Page Table。一旦找到可以用的 Page Table，就可以找到 4 个备份镜像保存的页地址 (PA)。 接下来尝试从单个启动块中读取完整的备份镜像。共有 4 个备份镜像，逐个尝试。 如果 4 个启动块中的备份镜像都出现读取错误，这时可以尝试将 4 个启动块中没有出错的 Page 数据进行组合的方式，重建完整的启动镜像。 如果都失败，则从 SPI NAND 启动失败。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

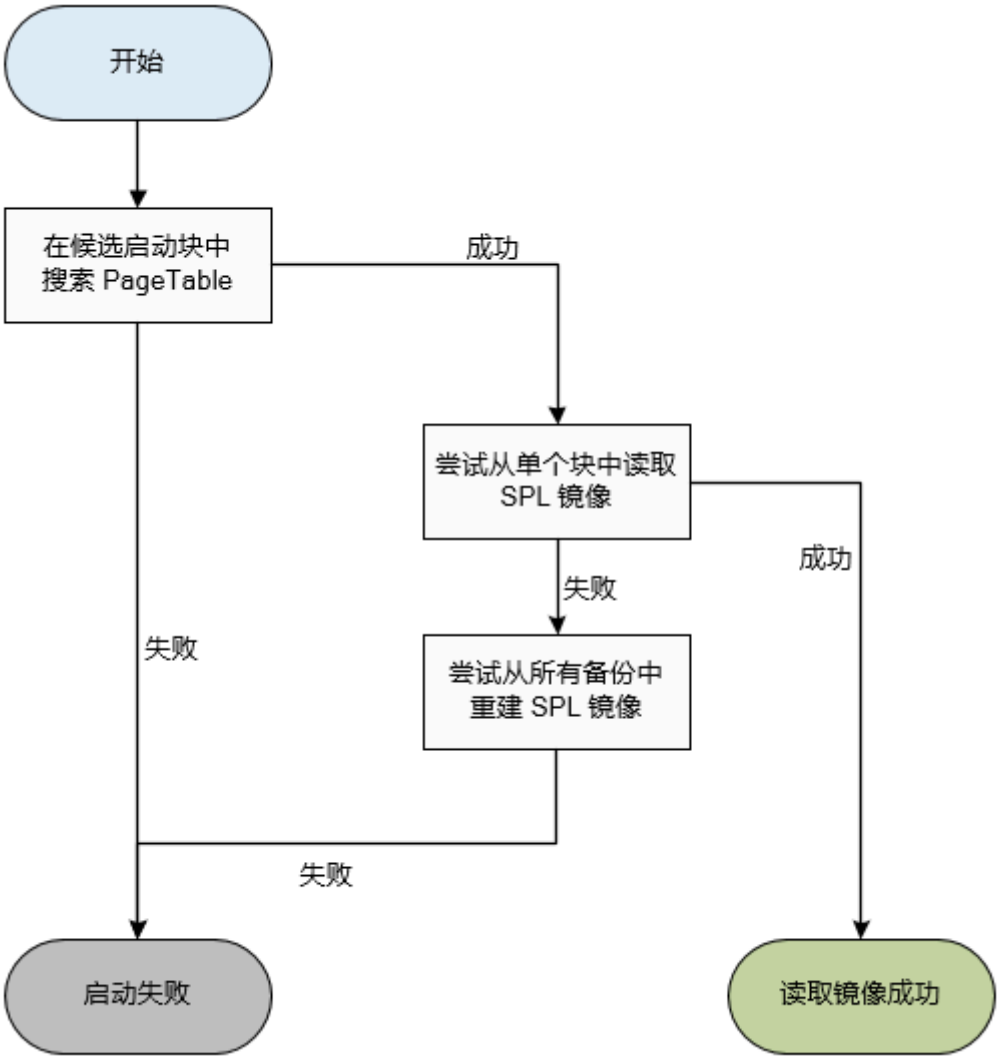


图 7–8 SPI NAND 启动流程

7.3.6. SPI NOR 启动

对于 SPI NOR 设备，预留两个的启动分区，用于保存第一级引导程序，每个分区大小 128 KB。 启动时， BROM 首先尝试从第一个分区读取启动镜像，如果检验失败则尝试从第二个分区读取启动镜像。如果都失败，则从该设备启动失败。

区域	大小位置
Partition0	128 KB (0~128 KB)
Partition1	128 KB (128 KB~256 KB)
……	……

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

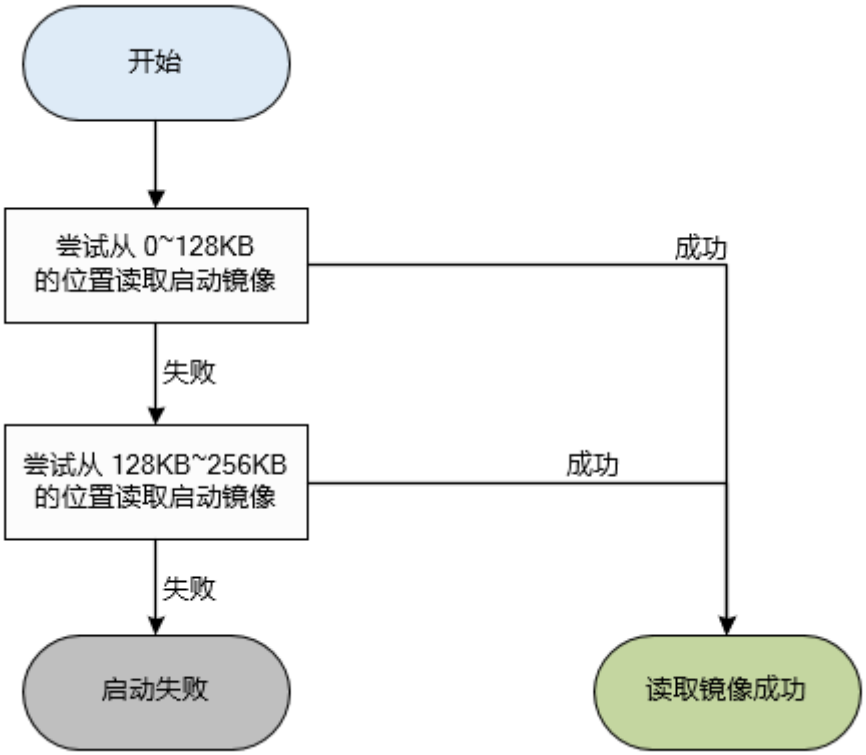


图 7–9 SPI NOR 启动流程

注：

Partition1 并不是必须的，方案可根据实际情况安排该区域的用途。

7.3.6.1. 使用限制

32 MB 或者以上的 SPI NOR 支持通过配置非易失性寄存器的方式设置上电/Reset 默认进入 4 字节地址模式。 由于各家的配置方式会有不同，BROM 无法兼容该配置，因此 BROM 不支持默认 4 字节地址模式。

使用限制：

1. 确保 SPI NOR 上电默认为 3 字节地址模式
2. 如果 BootLoader 或者 Kernel 需要使用 4 字节地址模式，应该在对应的驱动中，通过发送命令的方式切换模式

7.4. 烧录和升级模式

BROM 阶段使用升级模式的具体步骤如下：

1. 选择升级模式启动。使用以下两种方式均可进入升级模式启动：
    - BROM 尝试所有启动介质均未成功启动，则自动进入升级模式。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

7.4.1. SiP Flash 烧录

对于量产场景，本芯片可以通过设置 SiP 在芯片内部的 SPI Flash 引脚映射，使得外部可访问，用于量产烧录。在量产场景下，由于没有启动镜像，BROM 会按照默认顺序尝试从所有的启动介质。当所有启动介质都尝试启动失败时，可通过设置引脚映射，使用此功能。以下为详细步骤：

1. 设置以下引脚映射：

   量产场景下，由于没有启动镜像，BROM 会按照默认顺序尝试从所有的启动介质。在所有的启动介质都尝试启动失败时，可以设置引脚映射。

   注：

   Hold 和 WP 引脚未做映射，内部为上拉状态。

引脚	功能
PB6	CS
PB7	MISO
PB8	MOSI
PB9	CLK

2. 引脚设置完成之后，BROM 即进入升级模式。

   如果在升级模式下，成功进行 USB 或者 UART 的连接， 此时引脚映射会被取消。

3. 如果启动失败，则打开所有可能的烧录方式，包括引脚映射到外部。

   引脚映射只是 SiP Flash 烧录的一种方式，其它烧录方式仍然适用于 SiP Flash。

4. 固件烧录。使用引脚映射的方式进行烧录量产时，需要将芯片插到特制的板卡，然后使用烧录器进行烧录。

7.4.2. 在线升级模式

7.4.2.1. 数据通信协议

升级通信协议以 USB 升级为基础，扩展支持 UART。通信协议分为三层结构：

- 应用层
- 传输层
- 物理层

其中应用层为物理传输无关的命令通信，传输层将具体的命令和数据进行封装，物理层处理具体物理连接相关的通信操作。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

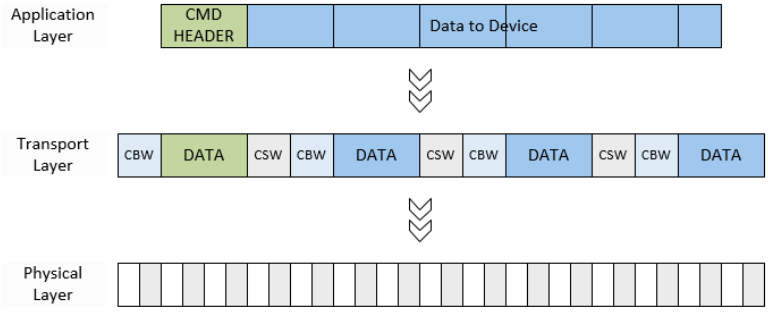


图 7–10 发送数据

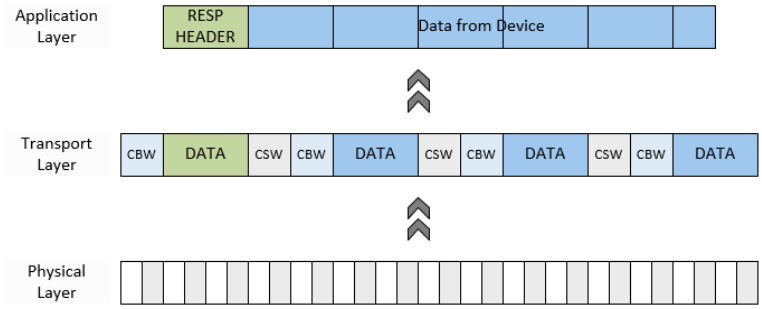


图 7–11 接收数据

7.4.2.2. USB 升级

完整的固件升级流程分为两个阶段：

1. BROM 阶段：完成系统运行环境的初始化， 包括初始化 DDR，下载并运行下一级升级程序。
2. U–Boot 阶段：实现固件下载和烧录等功能。

USB 升级功能通过实现基于 USB Bulk 传输的自定义设备来完成的，设备端与主机端通过自定义的 USB 升级协议进行通信。

7.4.2.2.1. 自定义传输设备

端点 (End Point) 是设备与主机之间进行数据传输的逻辑接口。自定义的 USB Bulk 传输设备包含下面几个端点：

1. EP0

   EP0 属于控制端点，数据传输为双向传输，用于对设备进行枚举和基本的控制。在设备完成配置之前， 只有 EP0 端点可用，主机通过 EP0 来发送命令，获取设备的信息，并通过它来配置设备。 EP0 端点没有专门的描述符，只需要在设备描述符中定义它的最大包长度。

2. EP Bulk–in

   EP Bulk–in 端点，用来从设备端传输数据和状态给主机端，对设备端而言，是一个数据发送端点。该端点采用 Bulk 传输模式。

3. EP Bulk–out

   EP Bulk–out 端点，用来从主机端传输命令和数据给设备端，对设备端而言，是一个数据接收端点。该端点使用 Bulk 传输模式。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

初始化过程中，传输设备需要提供下列描述符。

- 设备描述符 (Device Descriptor)
  - 必须提供 idVendor 和 idProduct 信息。
  - Vendor ID 需要向 USB-IF 购买，Product ID 由厂商自定义。

    AIC 购买的 Vendor ID 为 0x33C3，USB 升级所用的 Product ID 为 0x6677 (”fw” 的 ASCII 码值，表示用于 firmware upgrade) 。

Offset	Field	Size	Value	Description
0	*bLength*	Byte	12h	Size of this descriptor in bytes.
1	*bDescriptorType*	Byte	01h	DEVICE descriptor type.
2	*bcdUSB*	Word	????h	*USB Specification* Release Number in Binary-Coded Decimal (i.e. 2.10 = 210h). This field identifies the release of the *USB Specification* with which the device and is descriptors are compliant.
4	*bDeviceClass*	Byte	00h	**Class is specified in the interface descriptor.**
5	*bDeviceSubClass*	Byte	00h	**Subclass is specified in the interface descriptor.**
6	*bDeviceProtocol*	Byte	00h	**Protocol is specified in the interface descriptor.**
7	*bMaxPacketSize0*	Byte	??h	Maximum packet size for endpoint zero. (only 8, 16, 32, or 64 are valid **(08h, 10h, 20h, 40h)**).
8	*idVendor*	Word	????h	Vendor ID (assigned by the USB-**IF**).
10	*idProduct*	Word	????h	Product ID (assigned by the manufacturer).
12	*bcdDevice*	Word	????h	Device release number in binary-coded decimal.
14	*iManufacturer*	Byte	??h	Index of string descriptor describing the manufacturer.
15	*iProduct*	Byte	??h	Index of string descriptor describing this product.
16	*iSerialNumber*	Byte	??h	Index of string descriptor describing the device's serial number. **(Details in 4.1.1 below)**
17	*bNumConfigurations*	Byte	??h	Number of possible configurations.

- 配置描述符 (Configuration Descriptor)

  bNumInterfaces 设置为 1，仅支持一个 Bulk-Only 接口。bmAttributes 设置为 0x80。

Offset	Field	Size	Value	Description
0	*bLength*	Byte	09h	Size of this descriptor in bytes.
1	*bDescriptorType*	Byte	02h	CONFIGURATION Descriptor Type.
2	*wTotalLength*	Word	????h	Total length of data returned for this configuration. Includes the combined length of all descriptors (configuration, interface, endpoint, and class- or vendor-specific) returned for this configuration.
4	*bNumInterfaces*	Byte	??h	Number of interfaces supported by this configuration. **The device shall support at least the Bulk-Only Data Interface.**
5	*bConfigurationValue*	Byte	??h	Value to use as an argument to the *SetConfiguration()* request to select this configuration.
6	*iConfiguration*	Byte	??h	Index of string descriptor describing this configuration.
7	*bmAttributes*	Byte	?0h	Configuration characteristics: **Bit** **Description** 7 Reserved (set to one) 6 Self-powered 5 Remote Wakeup 4..0 Reserved (reset to zero) **Bit** 7 is reserved and must be set to one for historical reasons. **For a full description of this *bmAttributes* bitmap, see the *USB 1.1 Specification*.**
8	*MaxPower*	Byte	??h	Maximum power consumption of the USB device from the bus in this specific configuration when the device is fully operational. Expressed in 2mA units (i.e. 50 = 100mA)

- 接口描述符 (Interface Descriptor)

  bNumEndpoints 设置为 0，包含 Bulk-in, Bulk-out 两个端点。 bInterfaceClass，bInterfaceSubClass 和 bInterfaceProtocol 设置为 0xff ，表示自定义类型。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

Offset	Field	Size	Value	Description
0	*bLength*	Byte	09h	Size of this descriptor in bytes.
1	*bDescriptorType*	Byte	04h	INTERFACE Descriptor Type.
2	*bInterfaceNumber*	Byte	0?h	Number of interface. Zero-based value identifying the index in the array of concurrent interfaces supported by this configuration.
3	*bAlternateSetting*	Byte	??h	Value used to select alternate setting for the interface identified in the prior field.
4	*bNumEndpoints*	Byte	??h	Number of endpoints used by this interface (excluding endpoint zero). **This value shall be at least 2.**
5	*bInterfaceClass*	Byte	08h	**MASS STORAGE Class.**
6	*bInterfaceSubClass*	Byte	0?h	Subclass code (assigned by the USB-**IF**). **Indicates which industry standard command block definition to use. Does not specify a type of storage device such as a floppy disk or CD-ROM drive.** (See *USB Mass Storage Overview Specification*)
7	*bInterfaceProtocol*	Byte	50h	**BULK-ONLY TRANSPORT.** (See *USB Mass Storage Overview Specification*)
8	*iInterface*	Byte	??h	Index to string descriptor describing this interface.

- Bulk–in 端点描述符 (Bulk–in Endpoint Descriptor)

Offset	Field	Size	Value	Description
0	*bLength*	Byte	07h	Size of this descriptor in bytes.
1	*bDescriptorType*	Byte	05h	ENDPOINT Descriptor Type.
2	*bEndpointAddress*	Byte	8?h	The address of this endpoint on the USB device. The address is encoded as follows. Bit – Description 3..0 – The endpoint number 6..4 – Reserved, set to 0 7 – 1 = In
3	*bmAttributes*	Byte	02h	This is a Bulk endpoint.
4	*wMaxPacketSize*	Word	00??h	Maximum packet size. Shall be 8, 16, 32 or 64 bytes (08h, 10h, 20h, 40h).
6	*bInterval*	Byte	00h	Does not apply to Bulk endpoints.

- Bulk–out 端点描述符 (Bulk–out Endpoint Descriptor)

Offset	Field	Size	Value	Description
0	*bLength*	Byte	07h	Size of this descriptor in bytes.
1	*bDescriptorType*	Byte	05h	ENDPOINT descriptor type.
2	*bEndpointAddress*	Byte	0?h	The address of this endpoint on the USB device. This address is encoded as follows: Bit – Description 3..0 – Endpoint number 6..4 – Reserved, set to 0 7 – 0 = Out
3	*bmAttributes*	Byte	02h	This is a Bulk endpoint.
4	*wMaxPacketSize*	Word	00??h	Maximum packet size. Shall be 8, 16, 32 or 64 bytes (08h, 10h, 20h, or 40h).
6	*bInterval*	Byte	00h	Does not apply to Bulk endpoints.

具体的细节可以参考《Universal Serial Bus Mass Storage Class Bulk–Only Transfer》。

7.4.2.3. UART 升级

表 7–6 UART 传输涉及的关键字符说明

名称	数值	说明
SOH	0x01	Start of Heading
STX	0x02	Start of Text
ACK	0x06	Acknowledge
DC1	0x11	Device Control，定义为切换到主机发送状态
DC2	0x12	Device Control，定义为切换到主机接收状态

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 7–6 UART 传输涉及的关键字符说明 (续)

名称	数值	说明
NAK	0x15	Negative Acknowledge
CAN	0x18	Cancel
A	0x41	ASCII 字符 ‘A’，从机程序发出的连接信号
C	0x43	ASCII 字符 ‘C’，主机程序发出的连接信号

升级流程如下：

1. 设置传输波特率

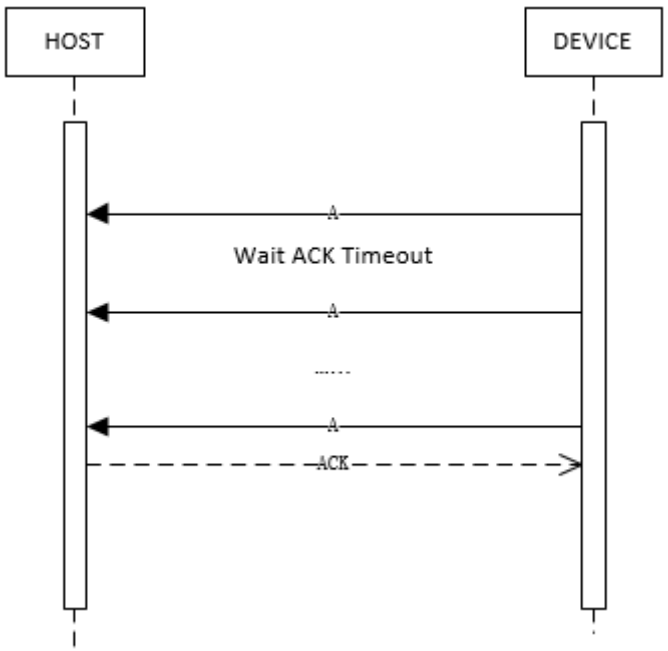
表 7–7 BROM 阶段使用的串口参数说明

波特率	数据位	校验位	停止位	流控制
115200	8	N	1	N

2. 建立连接

在连接检测阶段，由从机程序主动发出就绪信号，主机程序检测并回应的方式建立连接。

从机程序运行起来首先发送字符 A，表示就绪，并等待主机程序回应 ACK。如果等待超时， 则重复发送字符 A，进入等待。该过程循环执行，直到收到主机程序的 ACK 回复。 主机程序就绪时接收到字符 A，则应发送 ACK 建立连接。



从机程序的连接处理流程：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

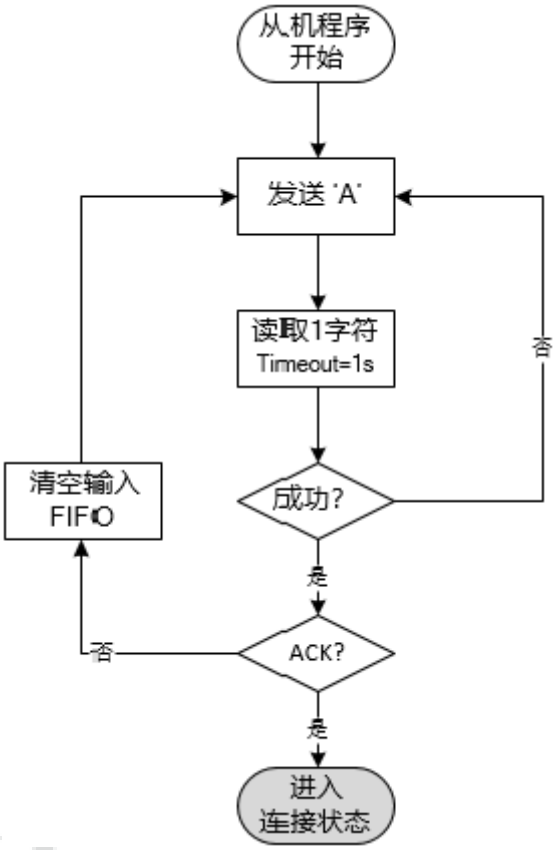


图 7–12 从机端建立连接的流程

主机程序的连接处理流程：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

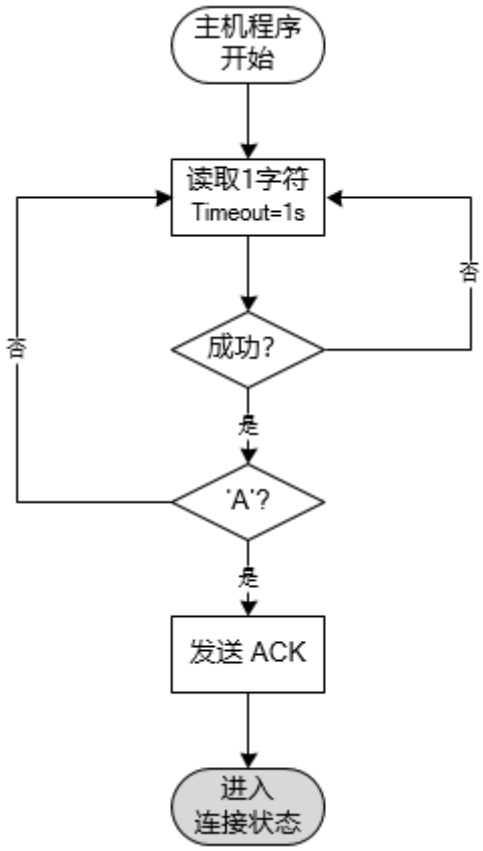


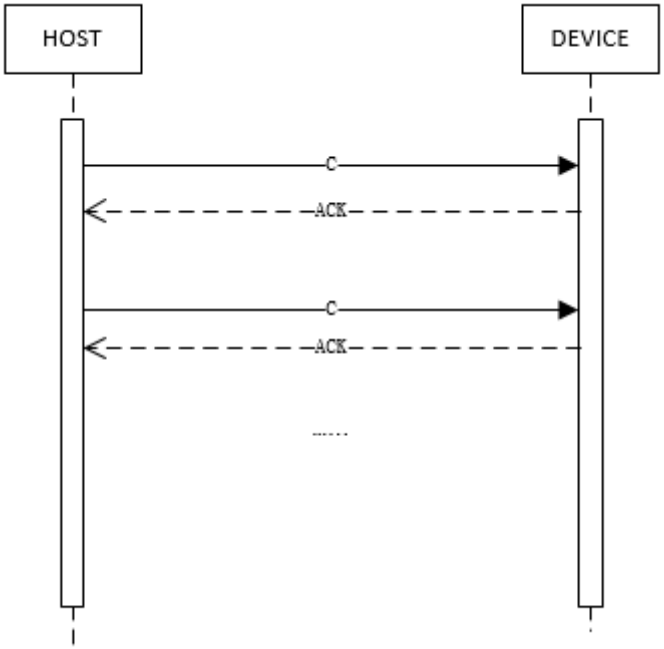
图 7–13 主机端建立连接的流程

3. 掉线检测。

   在没有传输任务的情况下，主机程序应该每隔 0.5 秒发送一次 C 进行确认连接状态， 如果收到 ACK，表明连接有效，超时则表明连接已经断开。

   从机程序通过命令通信的间隔时间进行掉线检测，即最后一次接收到有效的命令的时间到当前时间之间， 如果间隔超过 3 秒，则认为掉线，重新进入建立连接的状态，主动向外发出字符 A。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



4. 数据传输。

   主机程序与从机程序完成连接阶段即进入数据传输阶段。

   本协议的数据传输过程的一个特点是，数据传输是一个主机驱动的过程，从机总是被动响应。 即主机程序控制整个数据传输过程，从机程序是一个被动响应命令的程序。这也是为了保持与 USB 升级协议的行为一致。 这个特点与 YMODEM 协议相反。YMODEM 协议是一个从机驱动的协议，从机程序驱动数据的传输。

   关键点：

   a. 无论是主机程序还是从机程序，只有处于数据传输阶段才可以传输数据

   b. 数据传输阶段，主机程序和从机程序都要区分两个状态：数据发送状态和数据接收状态

   c. 从机程序不会主动传输数据，传输数据的动作，总是被动触发

   状态切换

   为了描述方便，在数据传输阶段的状态统一分为两个状态进行描述：

   - 主机发送状态
   - 主机接收状态

   默认情况下，系统处于主机发送状态（从机程序的角度为数据接收状态）。 在该状态下主机程序仅向外发送数据帧，并且等待 ACK。从机程序等待读取数据帧， 读取校验正确之后，发送 ACK。

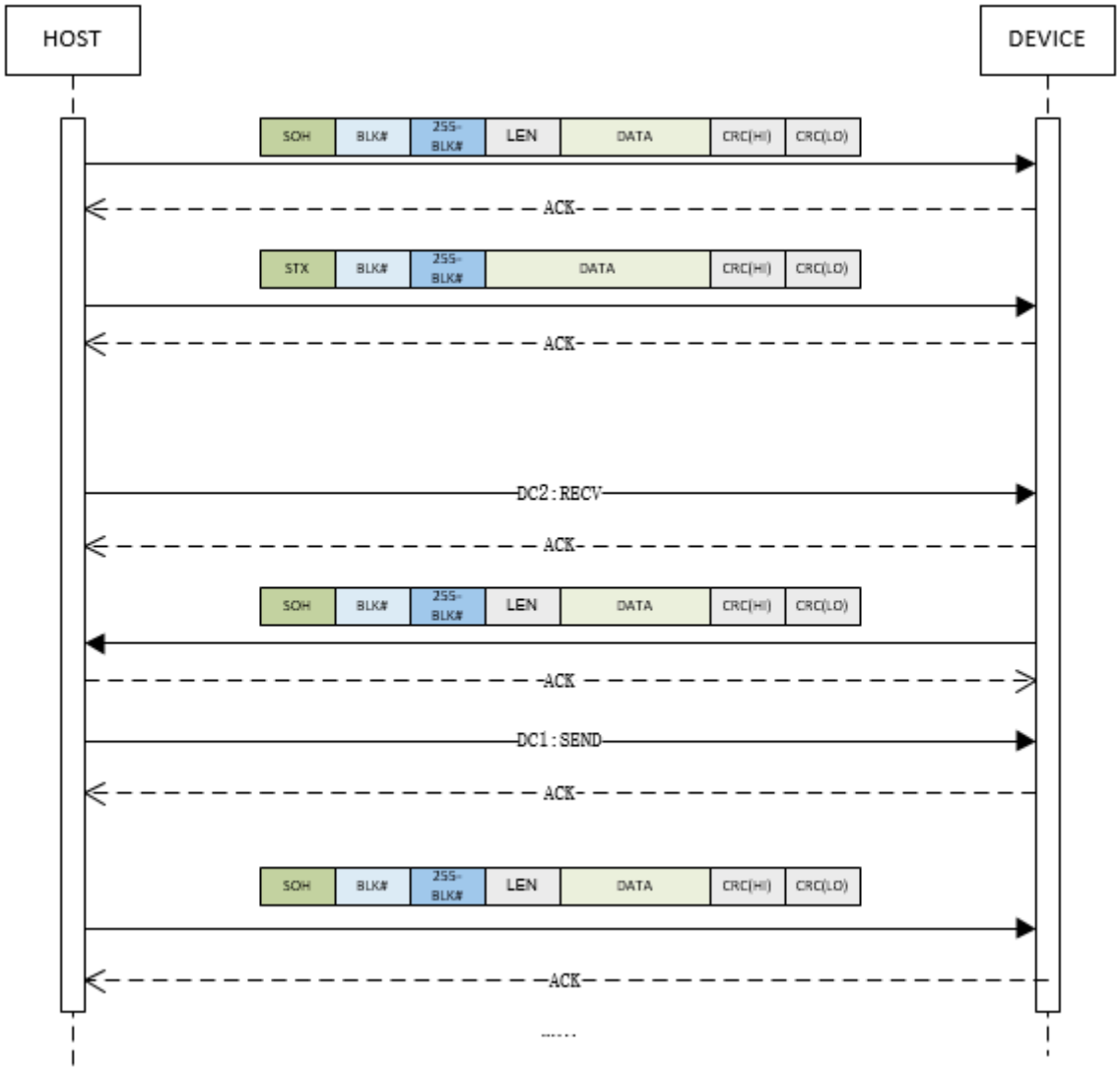
   状态切换由主机程序执行。当主机程序需要接收数据时，通过发送 DC2(RECV) 给从机程序， 从机程序回复 ACK 之后，通信状态切换为主机接收状态，此时从机程序可以发送数据帧， 主机程序等待接收数据帧，并且在校验正确后发送 ACK。

   当主机程序接收数据完毕，需要发送 DC1(SEND) 给从机程序，将状态切换到主机发送状态。

5. 传输流程

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

一个典型的数据传输流程如下图。



7.4.2.4. 升级流程

在进行固件升级的流程中，BROM 阶段需要完成下列任务：

1. 下载 Bootloader 到指定位置
2. 运行 Bootloader 进入 Bootloader 的升级模式

升级过程中，每一个参与升级的部分都视为升级固件的组件。制作升级固件时，每一个组件都有一个对应的组件描述信息。通过抽象化组件描述信息，可以形成一个统一的升级任务处理流程。具体的组件处理流程如下：

1. SET_FWC_META

   工具将对应的组件描述信息发送给设备端。设备端的升级程序依据描述信息，准备接收数据。

2. GET_BLOCK_SIZE

   工具获取本次组件传输的数据块大小，以数据块大小为单位发送组件数据。

3. SEND_FWC_DATA

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

工具发送组件数据给设备端。设备端在接收数据的同时，对接收到的数据进行 CRC 校验。 接收完成之后，设备端依据接收的组件描述信息，对数据执行相应操作。例如，在 BROM 阶段， 如果组件是一个可执行程序，则在接收完成之后，运行组件。

4. GET_FWC_CRC

   工具获取设备端计算的 CRC 校验值，判断组件的传输是否正确。

5. GET_FWC_BURN_RESULT

   工具获取设备端对组件数据的烧录是否正确。此步是一个统一的流程，但 BROM 不支持烧录组件，因此直接回复成功。

6. GET_FWC_RUN_RESULT

   工具获取组件运行的返回值。对于可执行组件，则将组件运行返回值传给工具端。 对于不可执行组件，则统一返回 0.

固件升级时，主机从固件包中顺序读取上述组件，按照流程逐个组件发送给设备端，即可完成 BROM 阶段的任务。

7.4.2.5. 通信和调试命令

BROM 阶段的升级模式支持以下命令：

1. 应用命令：

命令	值	功能
GET_HWINFO	0x00	获取硬件相关信息
GET_TRACEINFO	0x01	获取 BROM 的一些调试跟踪信息，AIC 内部命令
WRITE	0x02	写数据到 CPU 可访问的地址空间
READ	0x03	从 CPU 可访问的地址空间读取数据
EXEC	0x04	指定位置开始运行程序
JTAG_UNLOCK	0x06	解锁 JTAG
BOOT_CONTINUE	0x07	跳出升级模式的循环，重新从存储设备启动
SET_FWC_META	0x10	发送固件组件 (Firmware Component)的描述信息
GET_BLOCK_SIZE	0x11	获取传输的数据块大小，发送的有效数据须以该块大小为单位
SEND_FWC_DATA	0x12	发送固件组件数据
GET_FWC_CRC	0x13	获取设备端对所接收数据计算的 CRC 值，以确认传输是否出错
GET_FWC_BURN_RESULT	0x14	获取组件烧录后，设备端的校验结果
GET_FWC_RUN_RESULT	0x15	获取组件在运行之后的返回结果

2. 调试命令：

   BROM 阶段的升级协议支持下列调试命令，用于芯片验证等调试场景，可以完成诸如读写内存，读写寄存器，以及下载一段程序并执行等操作。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- WRITE
- READ
- EXEC

7.5. 时钟配置

BROM 阶段使用的总线和模块，以及各模块的时钟配置如下表所示。

表 7–8 BROM 阶段模块时钟配置

名字	量产工作频率 (MHz)	量产时钟来源	时钟源频率 (MHz)	时钟除频
APB0	24	OSC24M	24	-
APB1	24	OSC24M	24	-
CPU	24	OSC24M	24	-
SRAM	60	PLL_INT1	1200	20
SID	24	OSC24M	24	-
UART	24	OSC24M (APB1)	24	-
TIMER	24	OSC24M (APB1)	24	-
DMA	60	PLL_INT1 (AHB)	1200	20
SDMC	24	PLL_FRA0	1008	21
SPI	24	PLL_FRA0	1008	21
SPIENC	60	PLL_INT1 (AHB)	1200	20
CE	200	PLL_INT1	1200	6
USB	60	PLL_INT1 (AHB)	1200	20

说明：

- SDMC/ SPI 模块输入的频率为 48 MHz，内部二分频，因此最大工作频率为 24 MHz。
- USB/ DMA/ SPIENC/ SRAM/ ROM 直接使用 AHB 的时钟频率。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

8. 时钟和电源

8.1. Clock Manage Unit (CMU)

Clock Management Unit (CMU) 是一个关键的系统组件，用于配置系统时钟，包括 PLL 频率和展频、AXI/ AHB/ APB 总线时钟、各模块输入时钟、以及 IO 输出时钟。通过对系统时钟的精确控制和管理，包括时钟信号的频率与分配，CMU 确保了系统运行的效率与稳定性。

8.1.1. 特性说明

- 两种锁相环共四个 PLL
- 两个 PLL_INT，无小数分频和展频功能，可旁路直接输出 CLK_24M 时钟
- 两个 PLL_FRA，带小数分频和展频功能，可旁路直接输出 CLK_24M 时钟
- CPU 时钟可选 CLK_24M 或 PLL_INT0 进行 1~16 分频
- AXI/ AHB/ APB0 时钟可选 CLK_24M 或 PLL_INT1 进行 1~16 除频
- APB1 固定使用 CLK_24M 时钟
- 每个模块固定时钟如图所示，模块时钟可选 1~16 除频
- 每个模块总线时钟、模块时钟和复位三个开关可独立配置

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

8.1.2. 原理框图

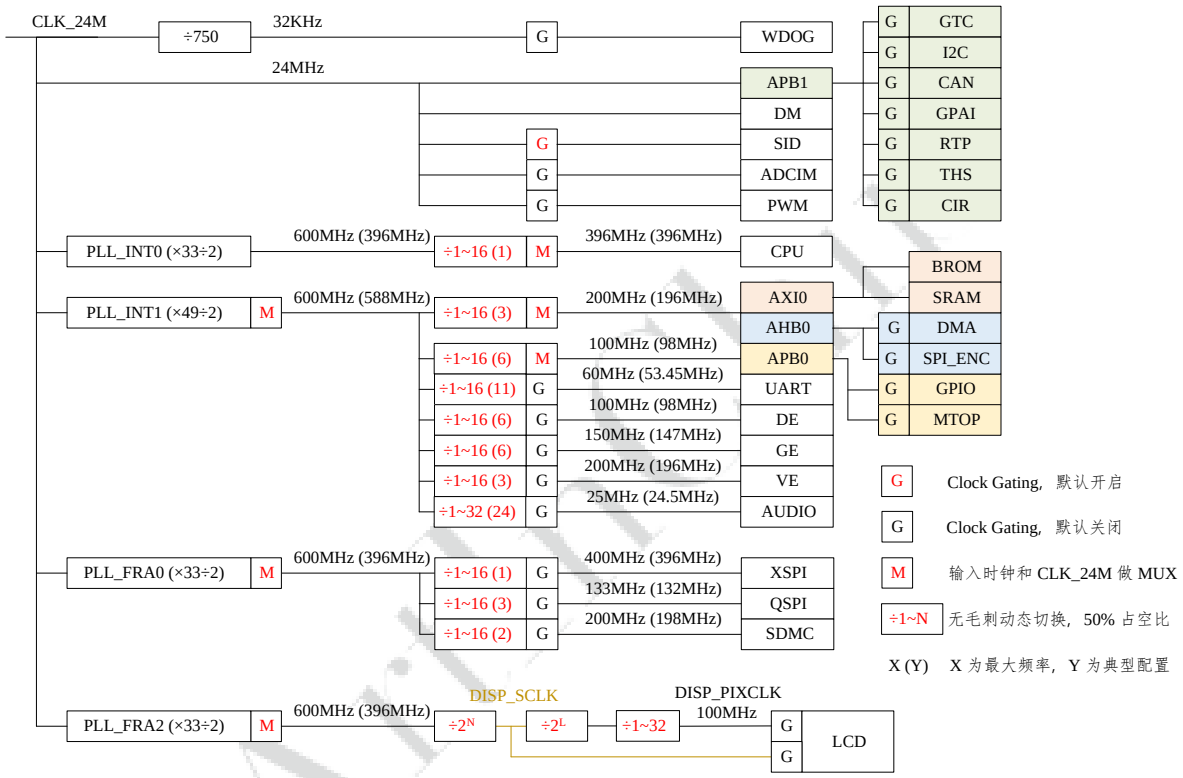


图 8–1 CMU 原理框图

8.1.3. 功能描述

8.1.3.1. PLL

模拟电路 PLL 用于产生时钟供给整个芯片。

下表列出了各个 PLL 的用途及特性。

> **注：**
>
> 为保证 PLL 性能要求，PLL_VCO = 24 MHz ÷ (P + 1) × (N + 1) 频率建议在 720 MHz~1584 MHz 之间。

表 8–1 CMU PLL 用途及特性

名称	用途	典型频率	展频
PLL_INT0	CPU	396 MHz	不支持
PLL_INT1	AXI/ AHB/ APB/DE/GE/ VE/UART/PWM/I2S/AUDIO	588 MHz	不支持

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 8–1 CMU PLL 用途及特性 (续)

名称	用途	典型频率	展频
PLL_FRA0	XSPI/ SPI/ SDMC	384 MHz	展频
PLL_FRA2	LCD	–	展频

PLL 内部结构如下图所示， PLL 频率计算公式为：PLL_OUT = 24 MHz × (N + 1 + F ÷ 0x1FFFF)/(P+1)/(M+1)。其中，F 为 FRA_IN 小数分频系数，当不使用小数分频时，F=0，频率计算公式简化为：PLL_OUT = 24 MHz × (N + 1)/(P + 1) ÷ (M+1)

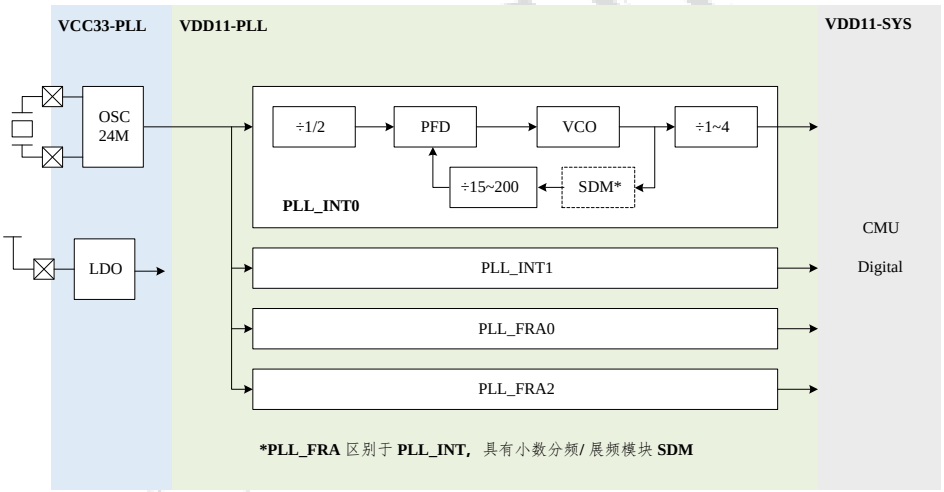


图 8–2 CMU PLL 内部框图

PLL 配置流程

为保证 PLL 正常开启，需要遵循以下的配置流程：

- PLL 输出配置旁路为 24 MHz， PLL_FRAn_GEN.PLL_OUT_MUX = 0，选择为 24M 时钟
- PLL 电路关闭， PLL_FRAn_GEN.PLL_EN = 0
- PLL 频率系数配置，
  PLL_FRAn_GEN.FACTOR_P，PLL_FRAn_GEN.FACTOR_N，PLL_FRAn_GEN.FACTOR_M，PLL_FRAn_GEN.FACTOR_M_EN
- PLL 展频/小数分频配置， PLL_FRAn_CFG PLL_FRAn_SDM
- PLL 电路打开， PLL_FRAn_GEN.PLL_EN = 1，延迟 200 us
- PLL 输出使能， PLL_FRAn_GEN.PLL_OUT_EN = 1
- PLL 输出配置切换为高频时钟， PLL_FRAn_GEN.PLL_OUT_MUX = 1

展频配置

展频相关参数如下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 展频幅度：寄存器为 PLL_FRAn_SDM.SDM_BOT，对应幅度为 24 MHz / (P + 1) × (0x1FFFF − SDM_BOT) ÷ 0x1FFFF
- 展频频率：寄存器为 PLL_FRAn_SDM.SDM_FREQ，建议设置为 3 (33 KHz)
- 展频模式：寄存器为 PLL_FRAn_SDM.SDM_MODE，建议设置为 2 (三角波)

展频配置公式如下：

- 根据展频幅度需求，求得 SDM_BOT = 0x1FFFF − SDM_AMP*0x1FFFF × (P+1) / 24 MHz
- 根据展频频率需求，求得 STEP_NUM = 24 MHz ÷ (P + 1) ÷ (SDM_FREQ ÷ 2) = 24000000 ÷ (P+1) ÷ (33000/2)
- 根据展频幅度和展频步数，求得 SDM_STEP = (0x1FFFF− SDM_BOT) ÷ STEP_NUM

注：

- PLL_FRAn_SDM.SDM_STEP 设置值需要和展频幅度和展频频率匹配，以产生正常的展频波形
- 当 P=0 时，最大展频幅度为 24 MHz，如果 N=32，则 PLL 频率为 792 MHz，展频深度为 24/792 ≈ 3.0%
- 当 P=1 时，最大展频幅度为 12 MHz，如果 N=65，则 PLL 频率为 792 MHz，展频深度为 12/792 ≈ 1.5%

8.1.3.2. 模块时钟

模块名称	总线时钟	模块时钟源	模块时钟极限频率	备注
E907 CORE	–	PLL_INT0	396 MHz	–
E907 CLIC	–	PLL_INT0 ÷ 2	200 MHz	–
E907 CLINT	–	PLL_INT0 ÷ 2	200 MHz	–
E907 DM	–	CLK_24M	24 MHz	–
AXI	AXI	PLL_INT1	200 MHz	–
AHB	AHB	PLL_INT1	200 MHz	–
APB0	APB0	PLL_INT1	100 MHz	–
APB1	APB1	–	24 MHz	–
BROM	AXI	–	–	–
SRAM	AXI	–	–	–
AHB Matrix	AHB	–	–	–
DMA	AHB	–	–	–
DCE	AHB	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

模块名称	总线时钟	模块时钟源	模块时钟极限频率	备注
XSPI	AHB	PLL_FRA0	400 MHz	–
SPI0	AHB	PLL_FRA0	100 MHz	–
SPI1	AHB	PLL_FRA0	100 MHz	–
SDMC0	AHB	PLL_FRA0	200 MHz	–
SDMC1	AHB	PLL_FRA0	200 MHz	–
SYSCFG	APB0	CLK_24M	24 MHz	–
CMU	APB0	–	–	–
SPI_ENC	APB0	HCLK	–	–
AXICFG	APB0	–	–	–
MTOP	APB0	–	–	–
AUDIO	APB0	PLL_INT1	26 MHz	–
GPIO	APB0	–	–	–
UART0	APB0	PLL_INT1	60 MHz	–
UART1	APB0	PLL_INT1	60 MHz	–
UART2	APB0	PLL_INT1	60 MHz	–
UART3	APB0	PLL_INT1	60 MHz	–
LCD	APB0	PLL_FRA2	200 MHz	DISP_PIXCLK: 100 MHz
DE	APB0	PLL_INT1	100 MHz	DISP_PIXCLK: 100 MHz
GE	APB0	PLL_INT1	150 MHz	–
VE	APB0	PLL_INT1	200 MHz	–
WDOG	APB1	CLK_32K	32 KHz	–
WRI	APB1	CLK_24M	24 MHz	–
SID	APB1	CLK_24M	24 MHz	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

模块名称	总线时钟	模块时钟源	模块时钟极限频率	备注
GTC	APB1	–	–	–
I2C0	APB1	–	–	–
I2C1	APB1	–	–	–
CAN0	APB1	–	–	–
CAN1	APB1	–	–	–
PWM	APB1	CLK_24M	24 MHz	–
ADCIM	APB1	CLK_24M	24 MHz	–
GPAI	APB1	–	–	–
RTP	APB1	–	–	–
THS	APB1	–	–	–
CIR	APB1	–	–	–

UART 在 PLL_INT1=588 MHz 下波特率精度

设定波特率	实际波特率	波特率偏差	CMU 除频	Over sampling	Clock source	UART 除频
300	300	0%	11	16	53454545	11136
600	600	0%	11	16	53454545	5568
1200	1200	0%	11	16	53454545	2784
2400	2400	0%	11	16	53454545	1392
4800	4800	0%	11	16	53454545	696
9600	9600	0%	11	16	53454545	348
14400	14400	0%	11	16	53454545	232
19200	19200	0%	11	16	53454545	174
38400	38401	0%	11	16	53454545	87
57600	57601	0%	11	16	53454545	58

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

设定波特率	实际波特率	波特率偏差	CMU 除频	Over sampling	Clock source	UART 除频
115200	115203	0%	11	16	53454545	29
230400	229687	–0.31%	10	16	58800000	16
380400	382812	0.63%	12	16	49000000	8
460800	459375	–0.31%	10	16	58800000	8
921600	918750	–0.31%	10	16	58800000	4
1000000	1020833	2.08%	12	16	49000000	3
1152000	1148437	–0.31%	16	16	36750000	2
1500000	1531250	2.08%	12	16	49000000	2
1750000	1750000	0%	7	16	84000000	3
2000000	2041666	2.08%	9	16	65333333	2
2450000	2450000	0%	15	16	39200000	1
2500000	2450000	–2%	15	16	39200000	1
3000000	3062500	2.08%	12	16	49000000	1

AUDIO 在 PLL_INT1 = 588 MHz 下精度

设定频率	CMU 除频	实际频率	频率偏差
24.5760 MHz	16	24.5000 MHz	0.3%
22.5792 MHz	17	22.6153 MHz	0.1%

8.1.3.3. 模块开关控制和时序

模块一般有三个控制位：

- MOD_RSTN（模块复位控制）：同时复位模块和模块总线。
- MOD_BUS_EN（模块总线时钟开关）：控制模块的所有总线时钟，包括 AX、AHB 和 APB，如果模块使用到多个总线，由于只有一个控制位，该控制位同时控制多个总线时钟。
- MOD_CLK_EN（模块时钟开关）： 控制除了总线时钟以外的模块时钟，如果模块用到多个时钟，由于只有一个控制位，该控制位同时控制多个模块时钟。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

模块开启和关闭具有一定的时序要求，其中 SP_USBD 时序要求较为特殊，其它模块统一开关时序，具体描述如下：

表 8–2 模块开关时序描述

模块	打开时序	关闭时序
USB_DEV	• 打开控制器时钟 • 打开模拟 PHY 时钟 • 延迟 100 微秒 • 放开模拟 PHY 复位 • 放开控制器复位	• 使能控制器复位 • 使能模拟 PHY 复位 • 关闭模拟 PHY 时钟 • 关闭控制器时钟
其它模块	• 打开模块时钟 • 打开总线时钟 • 放开复位控制	• 使能复位控制 • 关闭总线时钟 • 关闭模块时钟 • 关闭控制器时钟

8.1.4. 寄存器列表

表 8–3 寄存器列表

偏移地址	缩写	寄存器描述	章节
0x000	PLL_INT0_GEN	PLL_INT0 通用	0x000 PLL_INT0_GEN
0x004	PLL_INT1_GEN	PLL_INT1 通用	0x004 PLL_INT1_GEN
0x020	PLL_FRA0_GEN	PLL_FRA0 通用	0x020 PLL_FRA0_GEN
0x028	PLL_FRA2_GEN	PLL_FRA2 通用	0x028 PLL_FRA2_GEN
0x040	PLL_INT0_CFG	PLL_INT0 配置	0x040 PLL_INT0_CFG
0x044	PLL_INT1_CFG	PLL_INT1 配置	0x044 PLL_INT1_CFG
0x060	PLL_FRA0_CFG	PLL_FRA0 配置	0x060 PLL_FRA0_CFG
0x068	PLL_FRA2_CFG	PLL_FRA2 配置	0x068 PLL_FRA2_CFG
0x080	PLL_FRA0_SDM	PLL_FRA0 展频	0x080 PLL_FRA0_SDM
0x088	PLL_FRA2_SDM	PLL_FRA2 展频	0x088 PLL_FRA2_SDM
0x0A0	PLL_COM	PLL 公共	0x0A0 PLL_COM
0x0A4	PLL_IN	PLL 输入	0x0A4 PLL_IN
0x100	CLK_AXI_AHB	AXI AHB 时钟	0x100 CLK_AXI_AHB
0x120	CLK_APB0	APB0 时钟	0x120 CLK_APB0
0x200	CLK_CPU	CPU 时钟	0x200 CLK_CPU

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 8–3 寄存器列表 (续)

偏移地址	缩写	寄存器描述	章节
0x20C	CLK_WDOG	WDOG 时钟	0x20C CLK_WDOG
0x220	CLK_DISP	DISP 时钟	0x220 CLK_DISP
0x230	CLK_AUD_SCLK	AUDIO 串行时钟 AUDIO 串行时钟	0x230 CLK_AUD_SCLK
0x410	CLK_DMA	DMA 时钟	0x410 CLK_DMA
0x414	CLK_DCE	DCE 时钟	0x414 CLK_DCE
0x45C	CLK_XSPI	XSPI 时钟	0x45C CLK_XSPI
0x460	CLK_QSPI0	QSPI0 时钟	0x460 CLK_QSPI0
0x464	CLK_QSPI1	QSPI1 时钟	0x464 CLK_QSPI1
0x470	CLK_SDMC0	SDMC0 时钟	0x470 CLK_SDMC0
0x474	CLK_SDMC1	SDMC1 时钟	0x474 CLK_SDMC1
0x800	CLK_SYSCFG	SYSCFG 时钟	0x800 CLK_SYSCFG
0x810	CLK_SPI_ENC	SPI_ENC 时钟	0x810 CLK_SPI_ENC
0x81C	CLK_MTOP	MTOP 时钟	0x81C CLK_MTOP
0x83C	CLK_GPIO	GPIO 时钟	0x83C CLK_GPIO
0x840	CLK_UART0	UART0 时钟	0x840 CLK_UART0
0x844	CLK_UART1	UART1 时钟	0x844 CLK_UART1
0x848	CLK_UART2	UART2 时钟	0x848 CLK_UART2
0x84C	CLK_UART3	UART3 时钟	0x84C CLK_UART3
0x880	CLK_LCD	LCD 时钟	0x880 CLK_LCD
0x8C0	CLK_DE	DE 时钟	0x8C0 CLK_DE
0x8C4	CLK_GE	GE 时钟	0x8C4 CLK_GE
0x8C8	CLK_VE	VE 时钟	0x8C8 CLK_VE
0x904	CLK_SID	SID 时钟	0x904 CLK_SID
0x90C	CLK_GTC	GTC 时钟	0x90C CLK_GTC
0x960	CLK_I2C0	I2C0 时钟	0x960 CLK_I2C0
0x964	CLK_I2C1	I2C1 时钟	0x964 CLK_I2C1
0x980	CLK_CAN0	CAN0 时钟	0x980 CLK_CAN0
0x984	CLK_CAN1	CAN1 时钟	0x984 CLK_CAN1
0x990	CLK_PWM	PWM 时钟 XPWM0 时钟	0x990 CLK_PWM

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 8–3 寄存器列表 (续)

偏移地址	缩写	寄存器描述	章节
0x9A0	CLK_ADCIM	ADCIM 时钟	0x9A0 CLK_ADCIM
0x9A4	CLK_GPAI	GPAI 时钟	0x9A4 CLK_GPAI
0x9A8	CLK_RTP	RTP 时钟	0x9A8 CLK_RTP
0x9AC	CLK_THS	THS 时钟	0x9AC CLK_THS
0x9B0	CLK_CIR	CIR 时钟	0x9B0 CLK_CIR
0xFFC	CMU_MOD_VER	CMU 模块版本	0xFFC CMU_MOD_VER

8.1.5. 寄存器描述

8.1.5.1. 0x000 PLL_INT0_GEN

默认值：0x08002010			PLL_INT0 通用 (PLL_INT0 General)
位域	类型	默认值	描述
31:29	–	–	–
28:24	R/W	0x8	PLL_ICP 锁相环环路带宽调节 数值越小带宽越低
23:21	–	–	–
20	R/W	0x0	PLL_OUT_MUX 锁相环输出选择 • 0： OSC_24M 输出 • 1： PLL 时钟输出 该选择不受其它寄存器影响，PLL 损坏不工作时，时钟可旁路为 OSC_24M 时钟使用。在启动和休眠场景也可使用 OSC_24M 时钟。
19	R/W	0x0	FACTOR_M_EN 锁相环输出除频系数使能 • 0：关闭 • 1：打开
18	R/W	0x0	PLL_OUT_SYS 锁相环输出到系统使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x08002010			PLL_INT0 通用 (PLL_INT0 General)
位域	**类型**	**默认值**	**描述**
			• 0：关闭 • 1：打开
17	R	0x0	PLL_LOCK 锁相环锁定状态 • 0： 未锁定 • 1： 锁定，锁相环稳定工作 **注：** 在锁相环使能后，查询该寄存器等待稳定再进行其它操作，保证 PLL 正常使用。
16	R/W	0x0	PLL_EN 锁相环电路使能 • 0：关闭锁相环电路 • 1：打开锁相环电路 锁相环输出频率 PLL_O = 24 ÷ (P + 1) × (N + 1) ÷ (M + 1) MHz，P、N、M 分别为 FACTOR_P、FACTOR_N、FACTOR_M。
15:8	R/W	0x20	FACTOR_N 锁相环倍频系数在应用中，N 配置范围为 14 到 199。
7:6	–	–	–
5:4	R/W	0x1	FACTOR_M 锁相环输出除频系数 FACTOR_M ≠0 时，需要配置 FACTOR_M_EN = 1 锁相环才有输出。
3:1	–	–	–
0	R/W	0	FACTOR_P 锁相环输入除频系数

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

8.1.5.2. 0x004 PLL_INT1_GEN

默认值：0x08003010			PLL_INT1 通用 (PLL_INT1 General)
位域	类型	默认值	描述
31:29	–	–	–
28:24	R/W	0x8	PLL_ICP 锁相环环路带宽调节 数值越小带宽越低
23:21	–	–	–
20	R/W	0x0	PLL_OUT_MUX 锁相环输出选择 • 0： OSC_24M 输出 • 1： PLL 时钟输出 该选择不受其它寄存器影响，PLL 损坏不工作时，时钟可旁路为 OSC_24M 时钟使用。在启动和休眠场景也可使用 OSC_24M 时钟。
19	R/W	0x0	FACTOR_M_EN 锁相环输出除频系数使能 • 0：关闭 • 1：打开
18	R/W	0x0	PLL_OUT_SYS 锁相环输出到系统使能 • 0：关闭 • 1：打开
17	R	0x0	PLL_LOCK 锁相环锁定状态 • 0： 未锁定 • 1： 锁定，锁相环稳定工作 **注：** 在锁相环使能后，查询该寄存器等待稳定再进行其它操作，保证 PLL 正常使用。
16	R/W	0x0	PLL_EN 锁相环电路使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x08003010			PLL_INT1 通用 (PLL_INT1 General)
位域	类型	默认值	描述
			• 0：关闭锁相环电路 • 1：打开锁相环电路 锁相环输出频率 PLL_O = 24 ÷ (P + 1) × (N + 1) ÷ (M + 1) MHz，P、N、M 分别为 FACTOR_P、FACTOR_N、FACTOR_M。
15:8	R/W	0x30	FACTOR_N 锁相环倍频系数在应用中，N 配置范围为 14 到 199。
7:6	–	–	–
5:4	R/W	0x1	FACTOR_M 锁相环输出除频系数 FACTOR_M ≠ 0 时，需要配置 FACTOR_M_EN = 1 锁相环才有输出。
3:1	–	–	–
0	R/W	0	FACTOR_P 锁相环输入除频系数

8.1.5.3. 0x020 PLL_FRA0_GEN

默认值：0x08002010			PLL_FRA0 通用 (PLL_FRA0 General)
位域	类型	默认值	描述
31:29	–	–	–
28:24	R/W	0x8	PLL_ICP 锁相环环路带宽调节 数值越小带宽越低
23:21	–	–	–
20	R/W	0x0	PLL_OUT_MUX 锁相环输出选择 • 0： OSC_24M 输出 • 1： PLL 时钟输出 该选择不受其它寄存器影响，PLL 损坏不工作时，时钟可旁路为 OSC_24M 时钟使用。在启动和休眠场景也可使用 OSC_24M 时钟

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x08002010			PLL_FRA0 通用 (PLL_FRA0 General)
位域	**类型**	**默认值**	**描述**
19	R/W	0x0	FACTOR_M_EN 锁相环输出除频系数使能 • 0：关闭 • 1：打开
18	R/W	0x0	PLL_OUT_SYS 锁相环输出到系统使能 • 0：关闭 • 1：打开
17	R	0x0	PLL_LOCK 锁相环锁定状态 • 0： 未锁定 • 1： 锁定，锁相环稳定工作 **注：** 在锁相环使能后，查询该寄存器等待稳定再进行其它操作，保证 PLL 正常使用。
16	R/W	0x0	PLL_EN 锁相环电路使能 • 0：关闭锁相环电路 • 1：打开锁相环电路 锁相环输出频率 PLL_O = 24 ÷ (P+1) × (N+1) ÷ (M+1) MHz，P、N、M 分别为 FACTOR_P、FACTOR_N、FACTOR_M。
15:8	R/W	0x20	FACTOR_N 锁相环倍频系数在应用中，N 配置范围为 14 到 199。
7:6	–	–	–
5:4	R/W	0x1	FACTOR_M 锁相环输出除频系数 FACTOR_M ≠ 0 时，需要配置 FACTOR_M_EN = 1 锁相环才有输出。
3:1	–	–	–
0	R/W	0	FACTOR_P

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x08002010			PLL_FRA0 通用 (PLL_FRA0 General)
位域	类型	默认值	描述
			锁相环输入除频系数

8.1.5.4. 0x028 PLL_FRA2_GEN

默认值：0x08002010			PLL_FRA2 通用 (PLL_FRA2 General)
位域	类型	默认值	描述
31:29	–	–	–
28:24	R/W	0x8	PLL_ICP 锁相环环路带宽调节数值越小带宽越低
23:21	–	–	–
20	R/W	0x0	PLL_OUT_MUX 锁相环输出选择 • 0： OSC_24M 输出 • 1： PLL 时钟输出 该选择不受其它寄存器影响，PLL 损坏不工作时，时钟可旁路为 OSC_24M 时钟使用。在启动和休眠场景也可使用 OSC_24M 时钟。
19	R/W	0x0	FACTOR_M_EN 锁相环输出除频系数使能 • 0：关闭 • 1： 使能
18	R/W	0x0	PLL_OUT_SYS 锁相环输出到系统使能 • 0：关闭 • 1： 使能
17	R	0x0	PLL_LOCK 锁相环锁定状态 • 0： 未锁定 • 1： 锁定，锁相环稳定工作

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x08002010			PLL_FRA2 通用 (PLL_FRA2 General)
位域	类型	默认值	描述
			注： 在锁相环使能后，查询该寄存器等待稳定再进行其它操作，保证 PLL 正常使用。
16	R/W	0x0	PLL_EN 锁相环电路使能 • 0：关闭锁相环电路 • 1：使能锁相环电路 锁相环输出频率 PLL_O = 24 ÷ (P + 1) × (N + 1) ÷ (M + 1) MHz，P、N、M 分别为 FACTOR_P、FACTOR_N、FACTOR_M。
15:8	R/W	0x20	FACTOR_N 锁相环倍频系数在应用中，N 配置范围为 14 到 199。
7:6	–	–	–
5:4	R/W	0x1	FACTOR_M 锁相环输出除频系数 **FACTOR_M≠0 时，需要配置 FACTOR_M_EN = 1 锁相环才有输出。**
3:1	–	–	–
0	R/W	0x0	FACTOR_P 锁相环输入除频系数

8.1.5.5. 0x040 PLL_INT0_CFG

默认值：0x240C4010			PLL_INT0 配置 (PLL_INT0 Configuration)
位域	类型	默认值	描述
31	–	–	–
30:28	R/W	0x2	PLL_LOCK_TIME
27	–	–	–
26:24	R/W	0x4	PLL_IVCO
23:22	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x240C4010			PLL_INT0 配置 (PLL_INT0 Configuration)
位域	类型	默认值	描述
21:20	R/W	0x0	PLL_VCO_SEL
19	R/W	0x1	PLL_VCO_RST
18:16	R/W	0x4	PLL_VCO_GAIN
15	–	–	–
14:8	R/W	0x40	PLL_BINT bit9:LDO 偏置电流 • 0：4uA • 1：2uA bit8:偏置电流倍数 • 0：x4 • 1：x1
7	–	–	–
6:0	R/W	0x10	PLL_CINT PLL 初始振荡频率

8.1.5.6. 0x044 PLL_INT1_CFG

默认值：0x240C4010			PLL_INT1 配置 (PLL_INT1 Configuration)
位域	类型	默认值	描述
31	–	–	–
30:28	R/W	0x2	PLL_LOCK_TIME
27	–	–	–
26:24	R/W	0x4	PLL_IVCO
23:22	–	–	–
21:20	R/W	0x0	PLL_VCO_SEL
19	R/W	0x1	PLL_VCO_RST

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x240C4010			PLL_INT1 配置 (PLL_INT1 Configuration)
位域	类型	默认值	描述
18:16	R/W	0x4	PLL_VCO_GAIN
15	–	–	–
14:8	R/W	0x40	PLL_BINT bit9:LDO 偏置电流 • 0：4uA • 1：2uA bit8:偏置电流倍数 • 0：x4 • 1：x1
7	–	–	–
6:0	R/W	0x10	PLL_CINT PLL 初始振荡频率

8.1.5.7. 0x060 PLL_FRA0_CFG

默认值：0x00000000			PLL_FRA0 配置 (PLL_FRA0 Configuration)
位域	类型	默认值	描述
31:25	–	–	–
24	R/W	0x0	DITHER_EN 小数分频使能 • 0：关闭 • 1：打开
23:21	–	–	–
20	R/W	0x0	FRA_EN 小数分频使能 • 0：关闭 • 1：打开
19:17	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			PLL_FRA0 配置 (PLL_FRA0 Configuration)
位域	类型	默认值	描述
16:0	R/W	0x0	FRA_IN 小数分频使能

8.1.5.8. 0x068 PLL_FRA2_CFG

默认值：0x00000000			PLL_FRA2 配置 (PLL_FRA2 Configuration)
位域	类型	默认值	描述
31:25	–	–	–
24	R/W	0x0	DITHER_EN 抖动使能 • 0：关闭 • 1： 使能 **该字段为内部调试使用**
23:21	–	–	–
20	R/W	0x0	FRA_EN 小数分频使能 • 0：关闭 • 1：使能
19:17	–	–	–
16:0	R/W	0x0	FRA_IN 小数分频系数

8.1.5.9. 0x080 PLL_FRA0_SDM

默认值：0x00000000			PLL_FRA0 展频 (PLL_FRA0 Spread Spectrum)
位域	类型	默认值	描述
31	R/W	0x0	SDM_EN 展频使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			PLL_FRA0 展频 (PLL_FRA0 Spread Spectrum)
位域	类型	默认值	描述
			• 0：关闭 • 1：打开
30:29	R/W	0x0	SDM_MODE 展频模式 • 0： DC=0 • 1： DC=1 • 2：三角波 • 3：预留
28:20	R/W	0x0	SDM_STEP 展频步进
19	–	–	–
18:17	R/W	0x0	SDM_FREQ 展频频率 • 0： 31.5 KHz • 1： 32.0 KHz • 2: 32.5 KHz • 3: 33.0 KHz
16:0	R/W	0x0	SDM_BOT 展频底线 该位域越小表示幅度越大

8.1.5.10. 0x088 PLL_FRA2_SDM

默认值：0x00000000			PLL_FRA2 展频 (PLL_FRA2 Spread Spectrum)
位域	类型	默认值	描述
31	R/W	0x0	SDM_EN 展频使能 • 0:关闭 • 1:使能
30:29	R/W	0x0	SDM_MODE

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			PLL_FRA2 展频 (PLL_FRA2 Spread Spectrum)
位域	类型	默认值	描述
			展频模式 • 0：DC=0 • 1：DC=1 • 2：三角波 • 3：预留
28:20	R/W	0x0	SDM_STEP 展频步进
19	–	–	–
18:17	R/W	0x0	SDM_FREQ 展频频率 • 0: 31.5 KHz • 1: 32.0 KHz • 2: 32.5 KHz • 3: 33.0 KHz
16:0	R/W	0x0	SDM_BOT 展频底线 该位域越小表示幅度越大

8.1.5.11. 0x0A0 PLL_COM

默认值：0x80000009			PLL 公共 (PLL Common)
位域	类型	默认值	描述
31	R/W	0x1	COMMON_GATE 整个电路使能 (包括 PLL/ LDO/ BIAS) • 0：关闭 • 1：打开
30:4	–	–	–
3:1	R/W	0x4	LDO_VSET LDO 电压设置

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x80000009			PLL 公共 (PLL Common)
位域	类型	默认值	描述
			• 0: 0.90 V • 1: 0.95 V • 2: 1.00 V • 3: 1.05 V • 4: 1.10 V • 5: 1.15 V • 6: 1.20 V • 7: 1.25 V
0	R/W	0x1	LDO_EN LDO 使能 • 0：关闭 • 1：打开

8.1.5.12. 0x0A4 PLL_IN

默认值：0xE0004032			PLL 输入 (PLL Input)
位域	类型	默认值	描述
31:30	R/W	0x3	XTAL_GM24M 晶体驱动调节值 越大表示驱动越强
29	R/W	0x1	XTAL_START24M 晶体启动 • 0：关闭 • 1：打开 该寄存器在晶体起振后可以关闭，节省功耗
28	R/W	0x1	XTAL_ENXTAL 振荡使能 • 0：关闭 • 1：打开
14:8	R/W	0x40	OSC_OUT_TR24M 振荡器频率配置

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0xE0004032			PLL 输入 (PLL Input)
位域	类型	默认值	描述
			• 0：12 MHz • · · • 64: 24 MHz • · · • 127: 36 MHz **每一档对应 187.5 KHz。**
7	–	–	–
6:4	R/W	0x3	OSC_OUT_SEL(OSC_OUT_TR=1000000)24M 振荡器频率选择输出 • 0： 18 MHz • 1： 20 MHz • 2：22 MHz • 3：24 MHz • 4：26 MHz • 5：28 MHz • 6：30 MHz • 7：32 MHz
3:2	–	–	–
1	R/W	0x1	OSC24M_ENOSC24M 振荡使能 • 0：关闭 • 1：打开
0	–	–	PLL_IN_SEL(CLK_24) 时钟选择 • 0：OSC24M • 1：XTAL **注：** 使用外挂晶振时，需先配置 XTAL _EN=1，延迟 1 ms，再配置 XTAL _SEL=1

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

8.1.5.13. 0x100 CLK_AXI_AHB

默认值：0x00000000			AXI AHB 时钟 (AXI & AHB Clock)
位域	类型	默认值	描述
31:9	–	–	–
8	R/W	0x0	BUS_CLK_SEL 总线时钟选择 • 0：CLK_24M • 1：PLL_INT1/(DIV+1) 除频系数为 DIV + 1，50% 占空比，支持动态调频。
7:4	–	–	–
3:0	R/W	0x0	BUS_CLK_DIV

8.1.5.14. 0x120 CLK_APB0

默认值：0x00000000			APB0 时钟 (APB0 Clock)
位域	类型	默认值	描述
31:9	–	–	–
8	R/W	0x0	BUS_CLK_SEL 总线时钟选择 • 0：CLK_24M • 1：PLL_INT1/(DIV+1) 除频系数为 DIV + 1，50% 占空比，支持动态调频。
7:4	–	–	–
3:0	R/W	0x0	BUS_CLK_DIV

8.1.5.15. 0x200 CLK_CPU

默认值：0x00000000			CPU 时钟 (CPU Clock)
位域	类型	默认值	描述
31:9	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			CPU 时钟 (CPU Clock)
位域	**类型**	**默认值**	**描述**
8	R/W	0x0	CPU_CLK_SEL CPU 时钟源选择 • 0：CLK_24M • 1：PLL_INT0/(DIV+1)，除频系数为 DIV+1 **50% 占空比，支持动态调频。**
7:4	–	–	–
3:0	R/W	0x0	CLK_DIV 输入时钟除频系数

8.1.5.16. 0x20C CLK_WDOG

默认值：0x00000000			WDOG 时钟 (WDOG Clock)
位域	**类型**	**默认值**	**描述**
31:29	–	–	–
28	R/W	0x0	CLK_WDOG_WR_DIS CLK_WDOG 寄存器写失效 • 0： 控制位可写 • 1： 控制位写失效 **注：** 该位一旦写 1 后该寄存器无法写入，即通过软件无法清零，只有在芯片复位时候才能实现清零。。
27:14	–	–	–
13	R/W	0x0	WDOG_RSTN WDOG 复位控制 • 0：复位有效 • 1：复位无效
12	R/W	0x0	MOD_BUS_EN 总线时钟使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			WDOG 时钟 (WDOG Clock)
位域	类型	默认值	描述
			• 0：关闭 • 1：打开
11:9	–	–	–
8	R/W	0x0	MOD_CLK_EN 模块时钟使能 • 0：关闭 • 1：打开
7:0	–	–	–

8.1.5.17. 0x220 CLK_DISP

默认值：0x00000000			DISP 时钟 (DISP Clock)
位域	类型	默认值	描述
31:13	–	–	–
12	R/W	0x0	PIXCLK_DIV_SEL 像素时钟除频设置，像素时钟来源于串行时钟 • 0：整数除频，50% 占空比除频系数为 2L × (M + 1) • 1：固定 3.5 除频，非 50% 占空比
11:10	R/W	0x0	PIXCLK_DIV_L 像素时钟除频系数 L 除频系数为 2L (1/2/4/8)，50% 占空比。
9	–	–	–
8:4	R/W	0x0	PIXCLK_DIV_M 像素时钟除频系数 M 除频系数为 M+1(1~32)，50% 占空比。
3	–	–	–
2:0	R/W	0x0	SCLK_DIV_N 串行时钟除频系数，串行时钟源于 PLL_FRA2。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			DISP 时钟 (DISP Clock)
位域	类型	默认值	描述
			除频系数为 2N(1/2/4/8/16/32/64/128)，50% 占空比。

8.1.5.18. 0x230 CLK_AUD_SCLK

默认值：0x00000000			AUDIO 串行时钟 (Audio Serial Clock)
位域	类型	默认值	描述
31:9	–	–	–
8	R/W	0x0	AUD_CLK_SEL 时钟源选择 • 0: PLL_INT1 AUD_CLK_SEL=0，输出时钟频率为 PLL_INT1 / (SCLK_DIV+1) • 1: PLL_FRA0 AUD_CLK_SEL=1，输出时钟频率为 PLL_FRA0 / (SCLK_DIV+1)
7:5	–	–	–
4:0	R/W	0x0	SCLK_DIV SCLK 除频系数，SCLK 来源于 PLL_INT1。 除频系数为 DIV+1(1~32)，50%占空比。当 PLL 配置为 588 MHz: • **DIV=0x17，AUD_SCLK=24.5 MHz，约为 24.576 MHz（48KHz 应用）** • **DIV=0x19，AUD_SCLK≈22.6415 MHz，约为 22.6154 MHz（44.1 KHz 应用），精度偏差在 5‰ 以内**

8.1.5.19. 0x410 CLK_DMA

默认值：0x00000000			DMA 时钟 (DMA Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			DMA 时钟 (DMA Clock)
位域	类型	默认值	描述
			注： 此信号同时复位模块和寄存器总线。
12	R/W	0x0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:0	–	–	–

8.1.5.20. 0x414 CLK_DCE

默认值：0x00000000			DCE 时钟 (DCE Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0x0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:0	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

8.1.5.21. 0x45C CLK_XSPI

默认值：0x00000000			XSPI 时钟 (XSPI Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0x0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:9	–	–	–
8	R/W	0x0	MOD_CLK_EN 模块时钟使能 • 0：关闭 • 1：打开
7:4	–	–	–
3:0	R/W	0x0	MOD_CLK_DIV 模块时钟除频系数，除频系数为 DIV + 1，50% 占空比。

8.1.5.22. 0x460 CLK_QSPI0

默认值：0x00000000			QSPI0 时钟 (QSPI0 Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			QSPI0 时钟 (QSPI0 Clock)
位域	类型	默认值	描述
			• 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0x0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:9	–	–	–
8	R/W	0x0	MOD_CLK_EN 模块时钟使能 • 0：关闭 • 1：打开
6:4	–	–	–
3:0	R/W	0x0	MOD_CLK_DIV 模块时钟除频系数，除频系数为 DIV +1,50%占空比

8.1.5.23. 0x464 CLK_QSPI1

默认值：0x00000000			QSPI1 时钟 (QSPI1 Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			QSPI1 时钟 (QSPI1 Clock)
位域	类型	默认值	描述
12	R/W	0x0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:9	–	–	–
8	R/W	0x0	MOD_CLK_EN 模块时钟使能 • 0：关闭 • 1：打开
6:4	–	–	–
3:0	R/W	0x0	MOD_CLK_DIV 模块时钟除频系数，除频系数为 DIV +1,50%占空比

8.1.5.24. 0x470 CLK_SDMC0

默认值：0x00000000			SDMC0 时钟 (SDMC0 Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0x0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			SDMC0 时钟 (SDMC0 Clock)
位域	类型	默认值	描述
11:9	–	–	–
8	R/W	0x0	MOD_CLK_EN 模块时钟使能 • 0：关闭 • 1：打开
7:4	–	–	–
3:0	R/W	0x0	MOD_CLK_DIV 模块时钟除频系数，除频系数为 DIV + 1，50% 占空比。

8.1.5.25. 0x474 CLK_SDMC1

默认值：0x00000000			SDMC1 时钟 (SDMC1 Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0x0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:9	–	–	–
8	R/W	0x0	MOD_CLK_EN 模块时钟使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			SDMC1 时钟 (SDMC1 Clock)
位域	类型	默认值	描述
			• 0：关闭 • 1：打开
7:4	–	–	–
3:0	R/W	0x0	MOD_CLK_DIV 模块时钟除频系数，除频系数为 DIV + 1，50% 占空比。

8.1.5.26. 0x800 CLK_SYSCFG

默认值：0x00000000			SYSCFG 时钟 (SYSCFG Clock)
位域	类型	默认值	描述
31:13	–	–	–
12	R/W	0x0	MOD_BUS_EN MOD_CLK_EN 总线时钟使能模块时钟使能 • 0：关闭 • 1：打开
11:0	–	–	–

8.1.5.27. 0x810 CLK_SPI_ENC

默认值：0x00000000			SPI_ENC 时钟 (SPI_ENC Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			SPI_ENC 时钟 (SPI_ENC Clock)
位域	类型	默认值	描述
12	R/W	0x0	MOD_BUS_EN MOD_CLK_EN 总线时钟使能模块时钟使能 • 0：关闭 • 1：打开
11:9	–	–	–
8	R/W	0x0	MOD_CLK_EN 模块时钟使能 • 0：关闭 • 1：打开
7:0	–	–	–

8.1.5.28. 0x81C CLK_MTOP

默认值：0x00000000			MTOP 时钟 (MTOP Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0x0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:0	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			AUDIO 时钟 (AUDIO Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0x0	MOD_BUS_EN 总线时钟使能 • 0:总线时钟关闭 • 1:总线时钟打开
11:9	–	–	–
8	R/W	0x0	MOD_CLK_EN 模块时钟使能 • 0：关闭 • 1：打开
7:0	–	–	–

8.1.5.29. 0x83C CLK_GPIO

默认值：0x00000000			GPIO 时钟 (GPIO Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			GPIO 时钟 (GPIO Clock)
位域	类型	默认值	描述
			注： 此信号同时复位模块和寄存器总线。
12	R/W	0x0	MOD_BUS_EN 总线时钟使能模块时钟使能 • 0：关闭 • 1：打开
11:0	–	–	–

8.1.5.30. 0x840 CLK_UART0

默认值：0x00000000			UART0 时钟 (UART0 Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 注： 此信号同时复位模块和寄存器总线。
12	R/W	0x0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:9	–	–	–
8	R/W	0x0	MOD_CLK_EN 模块时钟使能 • 0：关闭 • 1：打开

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			UART0 时钟 (UART0 Clock)
位域	类型	默认值	描述
7:4	–	–	–
3:0	R/W	0x0	MOD_CLK_DIV 模块时钟除频系数，除频系数为 DIV + 1，50% 占空比。

8.1.5.31. 0x844 CLK_UART1

默认值：0x00000000			UART1 时钟 (UART1 Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0x0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:9	–	–	–
8	R/W	0x0	MOD_CLK_EN 模块时钟使能 • 0：关闭 • 1：打开
7:4	–	–	–
3:0	R/W	0x0	MOD_CLK_DIV 模块时钟除频系数，除频系数为 DIV + 1，50% 占空比。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

8.1.5.32. 0x848 CLK_UART2

默认值：0x00000000			UART2 时钟 (UART2 Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0x0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:9	–	–	–
8	R/W	0x0	MOD_CLK_EN 模块时钟使能 • 0：关闭 • 1：打开
7:4	–	–	–
3:0	R/W	0x0	MOD_CLK_DIV 模块时钟除频系数，除频系数为 DIV + 1，50% 占空比。

8.1.5.33. 0x84C CLK_UART3

默认值：0x00000000			UART3 时钟 (UART3 Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			UART3 时钟 (UART3 Clock)
位域	**类型**	**默认值**	**描述**
			• 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0x0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:9	–	–	–
8	R/W	0x0	MOD_CLK_EN 模块时钟使能 • 0：关闭 • 1：打开
7:4	–	–	–
3:0	R/W	0x0	MOD_CLK_DIV 模块时钟除频系数，除频系数为 DIV + 1，50% 占空比。

8.1.5.34. 0x880 CLK_LCD

默认值：0x00000000			LCD 时钟 (LCD Clock)
位域	**类型**	**默认值**	**描述**
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **此信号同时复位模块和寄存器总线。**
12	R/W	0x0	MOD_BUS_EN 总线时钟使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			LCD 时钟 (LCD Clock)
位域	类型	默认值	描述
			• 0：关闭 • 1：打开
11:9	–	–	–
8	R/W	0x0	MOD_CLK_EN 模块时钟使能 • 0：关闭 • 1：打开
7:0	–	–	–

8.1.5.35. 0x8C0 CLK_DE

默认值：0x00000000			DE 时钟 (DE Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0x0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:9	–	–	–
8	R/W	0x0	MOD_CLK_EN 模块时钟使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			DE 时钟 (DE Clock)
位域	类型	默认值	描述
			• 0：关闭 • 1：打开
7:4	–	–	–
3:0	R/W	0x0	MOD_CLK_DIV 模块时钟除频系数，除频系数为 DIV + 1，50% 占空比。

8.1.5.36. 0x8C4 CLK_GE

默认值：0x00000000			GE 时钟 (GE Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0x0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:9	–	–	–
8	R/W	0x0	MOD_CLK_EN 模块时钟使能 • 0：关闭 • 1：打开
7:4	–	–	–
3:0	R/W	0x0	MOD_CLK_DIV

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			GE 时钟 (GE Clock)
位域	类型	默认值	描述
			模块时钟除频系数，除频系数为 DIV + 1，50% 占空比。

8.1.5.37. 0x8C8 CLK_VE

默认值：0x00000000			VE 时钟 (VE Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0x0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:9	–	–	–
8	R/W	0x0	MOD_CLK_EN 模块时钟使能 • 0：关闭 • 1：打开
7:4	–	–	–
3:0	R/W	0x0	MOD_CLK_DIV 模块时钟除频系数，除频系数为 DIV + 1，50% 占空比。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

8.1.5.38. 0x904 CLK_SID

默认值：0x00000100			SID 时钟 (SID Clock)
位域	**类型**	**默认值**	**描述**
31:13	–	–	–
12	R/W	0x0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:9	–	–	–
8	R/W	0x1	MOD_CLK_EN 模块时钟使能 • 0：关闭 • 1：打开
7:0	–	–	–

8.1.5.39. 0x90C CLK_GTC

默认值：0x00000000			GTC 时钟 (GTC Clock)
位域	**类型**	**默认值**	**描述**
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			GTC 时钟 (GTC Clock)
位域	类型	默认值	描述
11:0	–	–	–

8.1.5.40. 0x960 CLK_I2C0

默认值：0x00000000			I2C0 时钟 (I2C0 Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:0	–	–	–

8.1.5.41. 0x964 CLK_I2C1

默认值：0x00000000			I2C1 时钟 (I2C1 Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			I2C1 时钟 (I2C1 Clock)
位域	类型	默认值	描述
			注： 此信号同时复位模块和寄存器总线。
12	R/W	0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:0	–	–	–

8.1.5.42. 0x980 CLK_CAN0

默认值：0x00000000			CAN0 时钟 (CAN0 Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 注： 此信号同时复位模块和寄存器总线。
12	R/W	0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:0	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

8.1.5.43. 0x984 CLK_CAN1

默认值：0x00000000			CAN1 时钟 (CAN1 Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:0	–	–	–

8.1.5.44. 0x990 CLK_PWM

默认值：0x00000000			PWM 时钟 (PWM Clock) XPWM0 时钟 (XPWM0 Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0	MOD_BUS_EN 总线时钟使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			PWM 时钟 (PWM Clock) XPWM0 时钟 (XPWM0 Clock)
位域	类型	默认值	描述
			• 0：关闭 • 1：打开
11:9	–	–	–
8	R/W	0	MOD_CLK_EN 模块时钟使能 • 0：关闭 • 1：打开
7:0	–	–	–

8.1.5.45. 0x9A0 CLK_ADCIM

默认值：0x00000000			ADCIM 时钟 (ADCIM Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0	MOD_BUS_EN MOD_CLK_EN 总线时钟使能 • 0：关闭 • 1：打开
11:9	–	–	–
8	R/W	0	MOD_CLK_EN 模块时钟使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			ADCIM 时钟 (ADCIM Clock)
位域	类型	默认值	描述
			• 0：关闭 • 1：打开
7:0	–	–	–

8.1.5.46. 0x9A4 CLK_GPAI

默认值：0x00000000			GPAI 时钟 (GPAI Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:0	–	–	–

8.1.5.47. 0x9A8 CLK_RTP

默认值：0x00000000			RTP 时钟 (RTP Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			RTP 时钟 (RTP Clock)
位域	**类型**	**默认值**	**描述**
			• 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:0	–	–	–

8.1.5.48. 0x9AC CLK_THS

默认值：0x00000000			THS 时钟 (THS Clock)
位域	**类型**	**默认值**	**描述**
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:0	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

8.1.5.49. 0x9B0 CLK_CIR

默认值：0x00000000			CIR 时钟 (CIR Clock)
位域	类型	默认值	描述
31:14	–	–	–
13	R/W	0x0	MOD_RSTN 模块复位控制 • 0：复位有效 • 1：复位放开 **注：** 此信号同时复位模块和寄存器总线。
12	R/W	0	MOD_BUS_EN 总线时钟使能 • 0：关闭 • 1：打开
11:0	–	–	–

8.1.5.50. 0xFFC CMU_MOD_VER

默认值：0x00000102			CMU 模块版本 (CMU Module Version)
位域	类型	默认值	描述 GPT
31:0	RO	0x00000102	VERSION 版本 V1.2

8.2. System Configuration (SYSCFG)

System Config (SYSCFG) 模块为系统域以及模块顶层提供丰富的基础配置选项，包括但不限于配置 LDO 的工作模式或电压范围、SRAM 的大小或速度、PRS 的频率或周期、以及 USB 工作模式选择等。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

8.2.1. 特性说明

- 内置 LDO25，用于系统模拟及 GPADC/ eFuse 供电
- 内置 LDO18，用于 XSPI IO 和合封 PSRAM 颗粒供电
- 内置 LDO1x，用于 CORE 供电或外挂设备供电
- 提供 PRS 用于检测当前芯片的速度
- 提供参数配置到 SRAM，包含 CPU/ DE/ GE/ VE 等模块
- 调试信息输出配置

8.2.2. 原理框图

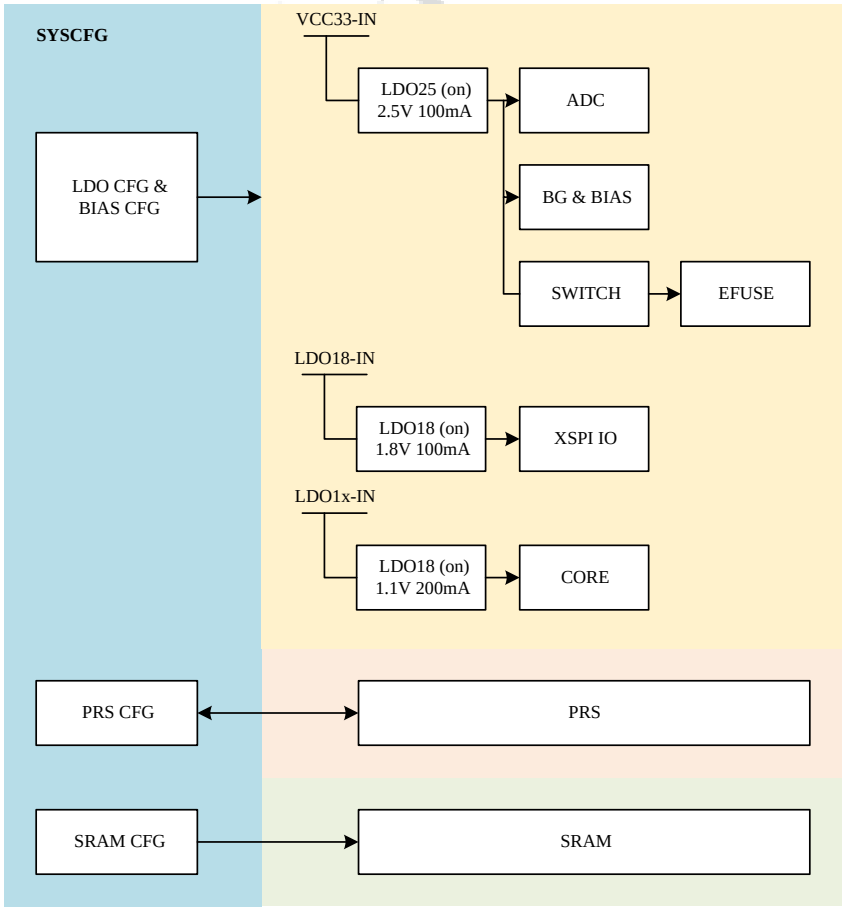


图 8–3 SYSCFG 原理框图

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

8.2.3. 功能描述

8.2.3.1. PSEN

Process Sensor (PSEN)，即工艺传感器，特定温度及供电电压情况下，其振荡频率可一定程度上表征芯片的制造工艺偏差。

图 8–4 : PSEN 工作流程：

1. 在软件中配置测试参数 (RO_SEL, CNT_TIME) 后，打开测试 (PSEN_EN=1)。
2. 打开测试后，PSEN 触发数字电路打开 RO_EN，模拟电路 RO 开始振荡工作，并产生一个经过 16 分频的低频时钟信号供数字电路计数使用。
3. 数字电路打开 RO_EN 后延迟打开 CNT_EN。此时 CNT 开始计数，直到预设的 CNT_TIME 时间结束，系统会自动关闭 CNT_EN 并延时关闭 RO_EN，最后将 PSEN_EN 清零表示测试完成。
4. 软件检测到 PSEN_EN 清零后，读取 CNT_VAL 值，作为当前 PSEN 的速度。
5. 重复此测试可以获取更多 RO 的数据，剔除异常数据取平均，可进一步提高数据的可信度。

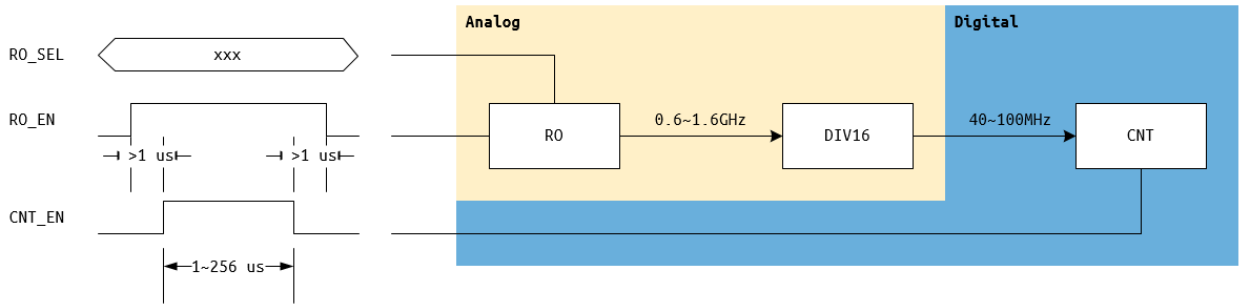


图 8–4 PSEN 工作流程

8.2.4. 寄存器列表

表 8–4 寄存器列表

偏移地址	缩写	寄存器描述	章节
0x020	LDO25_CFG	LDO25 配置	0x020 LDO25_CFG
0x024	LDO18_CFG	LDO18 配置	0x024 LDO18_CFG
0x028	LDO1x_CFG	LDO1x 配置	0x028 LDO1x_CFG
0x070	ATB_CFG	ATB 配置	0x070 ATB_CFG
0x0C0	PSEN_CFG	PSEN 配置	0x0C0 PSEN_CFG

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 8–4 寄存器列表 (续)

偏移地址	缩写	寄存器描述	章节
0x0C4	PSEN_CNT_VAL	PSEN 计数值	0x0C4 PSEN_CNT_VAL
0x100	SYS_SRAM_PAR	SYS SRAM 参数	0x100 SYS_SRAM_PAR
0x10C	VE_SRAM_PAR	VE SRAM 参数	0x10C VE_SRAM_PAR
0x110	GE_SRAM_PAR	GE SRAM 参数	0x110 GE_SRAM_PAR
0x114	DE_SRAM_PAR	DE SRAM 参数	0x114 DE_SRAM_PAR
0x140	SRAM_CLK_CFG	SRAM 时钟配置	0x140 SRAM_CLK_CFG
0x164	VE_SRAM_MAP	VE_SRAM 映射配置	0x164 VE_SRAM_MAP
0x1F0	FLASH_CFG	FLASH 配置	0x1F0 FLASH_CFG
0xFFC	SYSCFG_VER	SYSCFG 版本	0xFFC SYSCFG_VER

8.2.5. 寄存器描述

8.2.5.1. 0x020 LDO25_CFG

默认值：0x00040011			LDO25 配置 (LDO25 Configuration)
位域	类型	默认值	描述
	–	–	–
18	R/W	0x1	LVDS0_IBIAS_EN LVDS0 参考电流使能 • 0: 关闭 • 1: 使能
17	R/W	0x0	XSPI_DLLC1_IBIAS_EN XSPI DLLC1 参考电流使用 • 0: 关闭 • 1: 使能
16	R/W	0x0	XSPI_DLLC0_IBIAS_EN XSPI DLLC0 参考电流使用

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00040011			LDO25 配置 (LDO25 Configuration)
位域	类型	默认值	描述
			• 0: 关闭 • 1: 使能
15:8	R/W	0x0	BG_CTRL BG 电压控制
7:5	–	–	–
4	R/W	0x1	LDO25_EN LDO25 使能 • 0: LDO 关闭 • 1: LDO 打开
3	–	–	–
2:0	R/W	0x1	LDO25_VAL LDO25 电压设置，为了避免影响 EFUSE 的烧写，此处配置为 2.50V。 • 0: 2.40 V • 1: 2.50 V • 2: 2.60 V • 3: 2.70 V • 4: 2.80 V • 5: 2.90 V • 6: 3.00 V • 7: 3.10 V

8.2.5.2. 0x024 LDO18_CFG

默认值：0x00030003			LDO18 配置 (dLDO18 Configuration)
位域	类型	默认值	描述
31:28	–	–	–
27	R/W	0x0	ATB2_ANA_EN 模拟测试使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00030003			LDO18 配置 (dLDO18 Configuration)
位域	类型	默认值	描述
			• 0: 关闭 • 1: 使能
26	–	–	–
25:24	R/W	0x0	ATB2_ANA_SEL 模拟测试模块选择 • 0: ATB2_IN0 输出 • 1: ATB2_IN1 输出 • 2: ATB2_IN2 输出 • 3: ATB2_IN3 输出
23:6	–	–	–
5	R/W	0x0	LDO18_PD_FAST LDO18 快速掉电 • 0: 正常掉电 • 1: 快速掉电
4	R/W	0x0	LDO18_EN LDO18 使能 • 0: LDO 关闭 • 1: LDO 打开
3	–	–	–
2:0	R/W	0x3	LDO18_VAL LDO18 电压设置 • 0: 1.71 V • 1: 1.74 V • 2: 1.77 V • 3: 1.80 V • 4: 1.83 V • 5: 1.86 V • 6: 1.89 V • 7: 1.92 V

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

8.2.5.3. 0x028 LDO1x_CFG

默认值：0x00040011			LDO1 配置 (LDO1 Configuration)
位域	类型	默认值	描述
31:7	–	–	–
6	R/W	0x0	LDO1x_SOFT_EN LDO1x 软件模式使能 • 0：硬件模式，根据封装打线确定 LDO 打开还是关闭 硬件模式下，默认为 LDO 打开。 • 1：软件模式，根据寄存器 BIT4 确定 LDO 打开还是关闭
5	R/W	0x1	LDO1_PD_FAST LDO1 快速掉电 • 0: 正常掉电 • 1: 快速掉电
4	R/W	0x0	LDO1x_EN LDO1x 使能 **注：** 仅在 LDO1x_SOFT_EN 配置为1 时此位才有效。 • 0: LDO 关闭 • 1: LDO 打开
3:0	R/W	0x6	LDO1x_VAL LDO1x 电压设置 • 0: 1.0 V • 1: 1.025 V • 2: 1.05 V • 3: 1.075 V • 4: 1.1 V • 5: 1.125 V • 6: 1.15 V • 7: 1.175 V • 8: 1.2 V • 9: 1.225 V • 10: 1.25 V

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00040011			LDO1 配置 (LDO1 Configuration)
位域	类型	默认值	描述
			• 11: 1.275 V • 12: 1.3 V • 13: 1.325 V • 14: 1.35 V • 15: 1.375 V

8.2.5.4. 0x070 ATB_CFG

默认值：0x00000000			ATB 配置 (ATB Configuration)
位域	类型	默认值	描述
31:27	–	–	–
26:25	R/W	0x0	ATB3_SEL • 0: ana_atb_in0 • 1: ana_atb_in1 • 2: ana_atb_in2 • 3: ana_atb_in3
24	R/W	0x0	ATB3_OUT_EN • 0: 关闭 • 1: 开启
23:19	–	–	–
18:17	R/W	0x0	ATB2_SEL • 0: ana_atb_in0 • 1: ana_atb_in1 • 2: ana_atb_in2 • 3: ana_atb_in3
16	R/W	0x0	ATB2_OUT_EN • 0: 关闭 • 1: 开启

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			ATB 配置 (ATB Configuration)
位域	类型	默认值	描述
15:11	–	–	–
10:9	R/W	0x0	ATB1_SEL • 0: ana_atb_in0 • 1: ana_atb_in1 • 2: ana_atb_in2 • 3: ana_atb_in3
8	R/W	0x0	ATB1_OUT_EN • 0: 关闭 • 1: 开启
7:4	–	–	–
3:1	R/W	0x0	ATB0_SEL • 0: avcc_atb_out • 1: rtp_atb_out • 2: osc24m_atb • 3: fra0_atb_vout • 4: fra2_atb_vout • 5: int0_atb_vout • 6: int1_atb_vout • 7: cmu_bias_atb_vout
0	R/W	0x0	ATB0_OUT_EN • 0: 关闭 • 1: 开启

8.2.5.5. 0x0C0 PSEN_CFG

默认值：0x00000000			PSEN 配置 (Process Sensor Configuration)
位域	类型	默认值	描述
31:16	R/W	0x0	CNT_TIME 计数时间

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			PSEN 配置 (Process Sensor Configuration)
位域	类型	默认值	描述
			单位为 APB 总线时钟周期
15:4	–	–	–
3:1	R/W	0x0	RO_SEL PSEN 选择 • 0: 无效 • 1: RVT_40_12T • 2: LVT_40_12T • 3: RVT_40_8T • 4: 无效 • 5: RVT_50_12T • 6: LVT_50_12T • 7: RVT_50_8T
0	R/W	0x0	PSEN_START PSEN 开启 • 0: 关闭 • 1: 开启 **注：** PSEN 测试结束后，该位自动清零表示测试结束。

8.2.5.6. 0x0C4 PSEN_CNT_VAL

默认值：0x00000000			PSEN 计数值 (Process Sensor Counter Value)
位域	类型	默认值	描述
31:16	–	–	–
15:0	RO	0x0	CNT_VAL 计数值

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

8.2.5.7. 0x100 SYS_SRAM_PAR

默认值：0x22022202			SYS SRAM 参数 (System SRAM Parameter)
位域	类型	默认值	描述
31:0	R/W	0x22022202	SRAM_PAR SRAM 参数

8.2.5.8. 0x10C VE_SRAM_PAR

默认值：0x22022202			VE SRAM 参数 (VE SRAM Parameter)
位域	类型	默认值	描述
31:0	R/W	0x22022202	SRAM_PARS RAM 参数

8.2.5.9. 0x110 GE_SRAM_PAR

默认值：0x2202			GE SRAM 参数 (GE SRAM Parameter)
位域	类型	默认值	描述
31:16	–	–	–
15:0	R/W	0x2202	SRAM_PAR SRAM 参数

8.2.5.10. 0x114 DE_SRAM_PAR

默认值：0x22022202			DE SRAM 参数 (DE SRAM Parameter)
位域	类型	默认值	描述
31:0	R/W	0x22022202	SRAM_PAR SRAM 参数

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

8.2.5.11. 0x140 SRAM_CLK_CFG

默认值：0x00000000			SRAM 时钟配置 (SRAM Clock Configuration)
位域	类型	默认值	描述
31:19	–	–	–
18:0	R/W	0x0	SRAM_CLK_UNGATE SRAM 时钟不自动关闭 • 0: 自动关闭 • 1: 不自动关闭 每位对应一个模块，定义如下– • BIT15: AUDIO • BIT14: I2S • BIT12: SPI • BIT11: USB • BIT10: UART • BIT9: SD • BIT7: DE • BIT6: GE • BIT5: VE • BIT4: DVP • BIT3: GMAC • BIT2: DMA • BIT0: SYS

8.2.5.12. 0x164 VE_SRAM_MAP

默认值：0x00000000			VE_SRAM 映射配置 (VE SRAM Mapping Configuration)
位域	类型	默认值	描述
31:1	–	–	–
0	R/W	0x0	SRAM MAP • 0: 提供给系统使用 • 1：提供给 VE 使用

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

8.2.5.13. 0x1F0 FLASH_CFG

默认值：0x00000000			FLASH 配置 (FLASH Configuration)
位域	类型	默认值	描述
31:15	–	–	–
14:12	R/W	0x0	FLASH_IOMAP_012 FLASH_IO[0:2] 映射 • 0: WP – SO – CS • 1: WP – CS – SO • 2: SO – WP – CS • 3: SO – CS – WP • 4: CS – WP – SO • 5: CS – SO – WP • 6~7: 预留 **注：** BROM 启动后需要从 EFUSE 获取 FLASH_IOMAP 信息配置到该位域。
11	–	–	–
10:8	R/W	0x0	FLASH_IOMAP_345 FLASH_IO[3:5] 映射 • 0: HOLD – SCLK – SI • 1: HOLD – SI – SCLK • 2: SCLK – HOLD – SI • 3: SCLK – SI – HOLD • 4: SI – HOLD – SCLK • 5: SI – SCLK – HOLD • 6~7: 预留 **注：** BROM 启动后需要从 eFuse 获取 FLASH_IOMAP 信息配置到该位域。
7:2	–	–	–
1:0	R/W	0x0	FLASH_SRCSEL Flash 访问接口选择

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			FLASH 配置 (FLASH Configuration)
位域	类型	默认值	描述
			• 0: NONE，SiP Flash 接口关闭，无法访问 • 1: PIN，SiP Flash 接口通过 PIN 映射的方式进行访问，可被外部设备访问 • 2: SPI0，SiP Flash 通过 SPI0 控制器访问 • 3: SPI1，SiP Flash 通过 SPI1 控制器访问

8.2.5.14. 0xFFC SYSCFG_VER

默认值：0x00000102			SYSCFG 版本 (SYSCFG Version)
位域	类型	默认值	描述
31:0	RO	0x00000102	VERSION 版本 V1.2

8.3. Warm Reset Information (WRI)

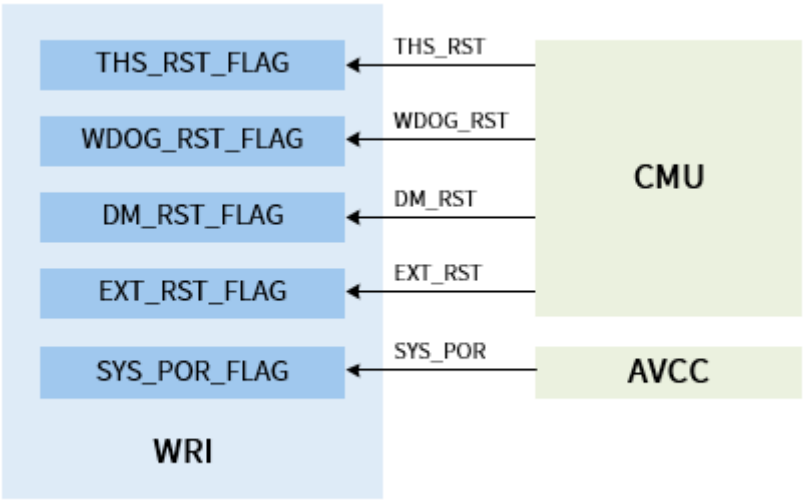
Warm Reset Info (WRI) 可以记录暖复位发生的状态信息， 提供引起死机重启的原因分析。暖复位包括有记录触发硬件复位的复位源等。

8.3.1. 特性说明

记录触发硬件复位的复位源。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

8.3.2. 原理框图



8.3.3. 功能描述

暖复位，也称热复位 (Hot Reset)，相对于冷复位 (Cold Reset)，是指系统不做掉电直接复位。

WRI 模块可用于记录暖复位信息，在暖复位期间稳定工作，使用到的系统资源不受复位影响：

- WRI 模块复位源于 SYS_POR，不受其它复位源影响。
- 时钟来源于 CLK_24M，不受 CMU 时钟选择影响。

WRI 提供复位源记录功能，即每个复位源信号经过去抖后，使用其低电平置位对应标志位。对应标志位的值可进行写 1 清 0。每次启动后，建议检查复位源信息并进行清除，方便记录下次复位源信息。

8.3.4. 寄存器列表

表 8–5 寄存器列表

偏移地址	缩写	寄存器描述	章节
0x000	RST_FLAG	复位标志	0x000 RST_FLAG
0x100	BOOT_INFO	启动信息	0x100 BOOT_INFO
0x104	SYS_BAK	系统备份	0x104 SYS_BAK
0x200	GPIO_ERR_INFO	GPIO 错误信息	0x200 GPIO_ERR_INFO
0xFFC	WRI_VER	WRI 版本	0xFFC WRI_VER

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

8.3.5. 寄存器描述

8.3.5.1. 0x000 RST_FLAG

默认值：0x00000001			复位标志 (Reset Flag)
位域	类型	默认值	描述
31:12	–	–	–
11	R/W1C	0x0	THS_RST_FLAG 温度过高复位标志 • 0: 没发生过复位 • 1: 发生过复位
10	R/W1C	0x0	WDOG_RST_FLAG 看门狗复位标志 • 0: 没发生过复位 • 1: 发生过复位
9	R/W1C	0x0	CPU_DM_RST_FLAG 处理器调试模块复位标志 • 0: 没发生过复位 • 1: 发生过复位
8	R/W1C	0x0	PIN_RST_FLAG 引脚输入复位标志 • 0: 没发生过复位 • 1: 发生过复位
7:2	–	–	–
1	R/W1C	0x0	RTC_POR_FLAG RTC 上电复位标志 • 0: 没发生过复位 • 1: 发生过复位
0	R/W1C	0x1	SYS_POR_FLAG 系统上电复位标志 • 0: 没发生过复位 • 1: 发生过复位

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

8.3.5.2. 0x100 BOOT_INFO

默认值：0x00000000			启动信息 (Boot Info)
位域	类型	默认值	描述
31:8	R/W	0x0	保留位
7:4	R/W	0x0	REBOOT_REASON 启动/ 重启的原因记录寄存器
3:0	R/W	0x0	BOOT_DEV 启动介质记录寄存器，该信息由 BROM 设置。

8.3.5.3. 0x104 SYS_BAK

默认值：0x00000000			系统备份 (System Backup)
位域	类型	默认值	描述
31:0	R/W	0x0	SYS_BAK

8.3.5.4. 0x200 GPIO_ERR_INFO

默认值：0x00000000			GPIO 错误信息 (GPIO ERROR INFO)
位域	类型	默认值	描述
31:2	–	–	–
1	R/W1C	0x0	XTAL_GPAI3_ERR 记录 XTAL 和 GPAI3 同时使能产生冲突时的状态位 • 0：未产生冲突 • 1：产生过冲突 若写 1 无法清零代表冲突仍然存在。该寄存器受所有复位源影响，会随之复位。
0	R/W1C	0x0	XTAL_GPAI2_ERR 记录 XTAL 和 GPAI2 同时使能产生冲突时的状态位 • 0：未产生冲突 • 1：产生过冲突 若写 1 无法清零代表冲突仍然存在。该寄存器受所有复位源影响，会随之复位。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

8.3.5.5. 0xFFC WRI_VER

默认值：0x00000102			WRI 版本 (WRI Version)
位域	类型	默认值	描述
31:0	RO	0x00000102	VERSION 版本，V1.2

ArtInChip

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9. 存储

9.1. Secure Digital Host Controller (SDMC)

SoC 包含多套 SD/ MMC 控制器，简称 SDMC。以下控制器均支持访问 SD 卡/ eMMC/ SDIO 设备，每套控制器常用的访问介质如下：

- SDMC0 用于访问 SD 卡和 eMMC 设备。
- SDMC1 用于访问 SD 卡或 SDIO 设备。

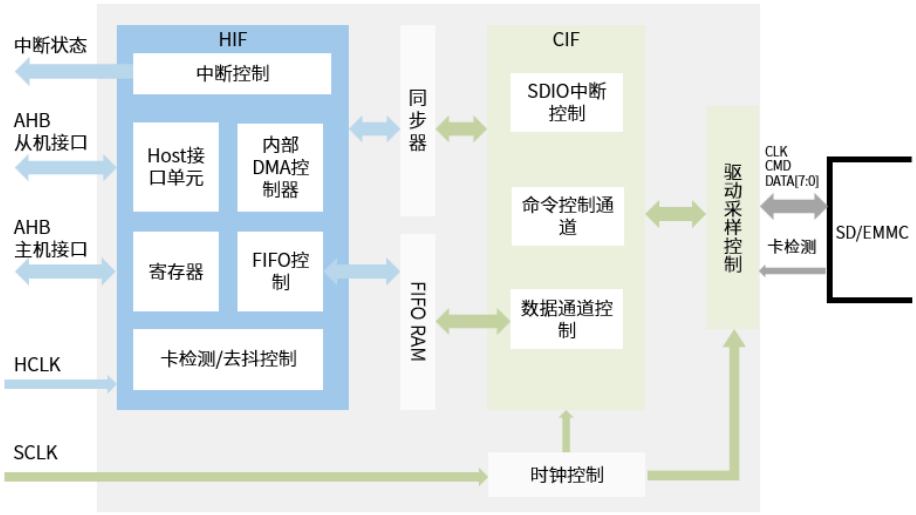
SDMC 支持的最大接口时钟频率为 50 MHz ，支持 1/ 4 线数据总线模式，支持 eMMC 标准协议和 SD/ SDIO 接口协议。

9.1.1. 功能特性

- 支持 eMMC 4.41，SD 2.0 和 SDIO 2.0，向下兼容。
- 支持 eMMC HS-SDR/ HS-DDR模式，接口最大传输速率100 Mbps 。
- 支持 3.3 V 工作电压。
- 支持 DDR 4 线模式。
- 支持内部 DMA 模式，且具备以下特性：
    - 支持单通道传输
    - 支持双缓存和描述符链表模式，描述符链表模式下每个描述符最大传输 4 KB 数据，双缓存模式下每个描述符最大传输 8 KB 数据
    - 支持可编程的突发传输数据长度和中断
- 支持的 FIFO 大小为 128 * 32 bits。
- 支持 CRC 生成和错误检测。
- 支持可编程的时钟分频系数，内部包含时钟分频器。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.1.2. 原理框图



9.1.3. 功能描述

SDMC 控制器由主机接口模块 (HIF)、卡接口模块 (CIF)、时钟控制器和驱动采样控制器组成，主要的工作流程如下：

- Host 主机通过 AHB 总线访问 HIF 模块和 HIF 寄存器，与 FIFO 进行收发数据。
- HIF 根据寄存器的配置信息，通过 FIFO 控制单元来控制数据传输，并将传输过程中产生的中断信号发送给 CPU 进行中断处理。
- HIF 通过卡检测单元，通知 Host 有卡插入或拔出。
- CIF 模块从 FIFO 中接收或写入数据，通过命令通道发送命令给外部设备，通过数据通道发送 FIFO 数据给外部设备，或从外部设备接收数据写入 FIFO。
- 时钟控制器根据外设的需求，提供适当的工作时钟频率。
- 驱动采样控制器，用于调整驱动时钟或采样时钟与数据之间的相位关系，来解决传输路径的延时问题。

9.1.3.1. 主机接口模块 (HIF)

9.1.3.1.1. 寄存器单元

通过 AHB 接口访问 SDMC 控制器，控制器内部使用 AHB 从机接口连接 AHB 主机。以下为主机接口模块相关单元的具体描述：

寄存器单元提供了寄存器的读写访问功能，寄存器位宽为 32 位。所有的寄存器都属于 HIF 时钟域。

1. 当一个命令发送到 Card 时，设置 0x00C SDMC_CMD 寄存器的 START_CMD 位为 1。

   所有需要 CIF 操作的相关寄存器均发送到 CIF 模块。

2. 一旦 START_CMD 置 1，在没有硬件清零之前，从 HIF 到 CIF 的下列寄存器不能被写入。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

注：

寄存器单元具有硬件锁功能，防止非法写入寄存器。

- CMD
- CMDARG
- CLKCTRL
- TTMC
- HCTRL2

3. 等待硬件清零 START_CMD 位，即可写入上述寄存器。

9.1.3.1.2. 中断控制单元

中断控制单元用于产生中断信号。中断信号的产生必须满足以下条件：

- 中断使能寄存器 0x03C SDMC_INTEN 相应的中断位置 1。
- 使能全局中断使能位。

一旦中断控制单元检测到某个中断条件，初始中断状态寄存器 0x040 SDMC_INTST 对应的中断状态位会置 1。初始中断状态位保持，直到通过软件写 1 清零。

9.1.3.1.3. FIFO 控制单元

FIFO 控制单元连接内部 FIFO 到 Host 接口以及卡接口模块。FIFO 深度为 128，FIFO 位宽为 32 位。由于读和写传输不会同时发生， 使用共享 FIFO 用于读写传输，以节省芯片面积。

9.1.3.1.4. 卡检测单元

卡检测单元用于检测 SD 卡插入和拔出时信号电平变化。上电后，控制器读取卡检测端口的状态值并将其存储到内存里，每次检测的状态值都与之前的状态值进行异或比较并更新到 0x054 SDMC_CDET 中的 CDET_STAT 位。

注：

仅 SDMC1 控制器支持硬件卡检测单元，其它控制器可以通过 GPIO 进行卡检测。

卡检测信号连接到卡座上，需要外部上拉：

- 当没有卡插入时，检测到高电平，CDET_STAT 位为 1。
- 当检测到卡插入时，卡检测引脚通过卡座连接到地，检测到低电平，CDET_STAT 位为 0。

通过一个去抖计时器来延迟检测时间，避免由于机械插拔造成的信号抖动，并产生一个中断信号发送给 CPU。用户可以通过编程 DEBC_CNT 位域设置去抖时间。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.1.3.2. 内部 DMA 控制器 (IDMAC)

SDMC 控制器内部包含一套 DMA 控制器（简称 IDMAC），IDMAC 由控制寄存器、状态寄存器和单向的发送/ 接收引擎组成。IDMAC 使用一个有效的描述符将数据从源地址搬移到目标地址，用于发送和接收数据，使 CPU 干预最小化。IDMAC 包含以下特性：

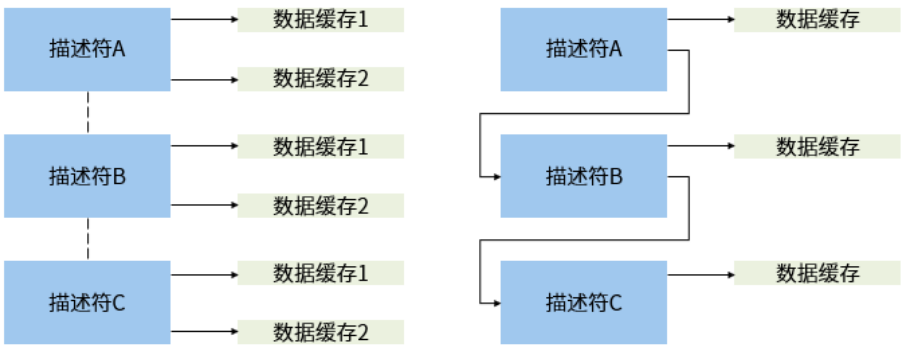
- IDMAC 从 Card 接收数据存储到内存中，并从内存中读取数据发送到 FIFO。
- IDMAC 的描述符存储在内存中，控制器通过内存的指针来获取描述符内容。
- Host 的数据缓存存储在内存中，由完整的数据或部分数据组成，描述符通过指针指向数据缓存：
  - 数据缓存中仅包含数据。
  - 数据缓存的状态包含在描述符中。

9.1.3.2.1. 描述符

IDMAC 使用两种描述符结构：

- 双缓存结构：每个描述符指向两个数据缓存，两个描述符的间隔，通过 PBUSCFG 寄存器的 DSL 位域进行编程。
- 链式结构：每个描述符指向唯一的缓存和下一个描述符地址。多个描述符可以组成描述符链表，包含以下特性：
  - 链表的基址写入描述符链表基址寄存器 IDMAINTEN。
  - 描述符链表向前连接，最后一个描述符可以指回到第一个描述符，形成一个环状结构。
  - 描述符链表存储在内存中。
  - 每个描述符可以指向两个数据缓存中容量最大的一个。

两种描述符结构见下图：



描述符地址必须按照 32 位对齐，每个描述符包含 16 字节的控制和状态信息，以及数据缓存的指针和容量。每个描述符由四个 DES 结构体组成，每个 DES 都存储 32 位数据。

- DES0 用于配置控制信息
- DES1 用于配置每个描述符的数据长度
- DES2 用于配置第一个数据缓存的起始地址
- DES3 用于配置第二个数据缓存的起始地址，或第二个描述符的起始地址

关于四个描述符结构体的详细配置说明，可查看 DES 结构体。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.1.3.2.2. IDMA 控制器初始化

1. 配置 HCTRL1 寄存器的 bit25，使用 IDMAC 访问 FIFO。
2. 写 IDMAC 总线模式寄存器 (PBUSCFG) 。
   a. 配置突发传输的数据长度。
   b. 使能 IDMAC。
   c. 复位 IDMAC 内部寄存器。
   d. 此步骤可选。如果使用双缓存结构的描述符，配置描述符之间跳过的数据长度。
3. 写 IDMAC 中断使能寄存器 (IDMAINTEN)，屏蔽掉不需要的中断。
4. 配置 IDMAC 描述符链表基址寄存器 (IDMASADDR)。

   驱动软件在内存中创建发送或接收的描述符链表，并写入 IDMAC 描述符链表基址寄存器 (IDMASADDR)，为 IDMAC 提供描述符的起始地址。
5. IDMAC 模块从描述符链表获取描述符。

9.1.3.2.3. DES 结构体

描述符地址必须按照 32 位对齐，每个描述符包含 16 字节的控制和状态信息，以及数据缓存的指针和容量。每个描述符由四个 DES 结构体组成，每个 DES 都存储 32 位数据。

DES0 用于配置控制信息，说明如下：

位域	位域名称	说明
31	OWN	• 1，表示描述符由 IDMAC 控制。 • 0，表示描述符由 Host 控制。当数据传输完成时，IDMAC 控制器对该位域清零。
30	CES (Card Error Summary)	错误位 • 表示发送到 Card 或来自 Card 的错误状态。 • 也存在于 OINTST 寄存器，与以下位域为逻辑或的关系： ◦ BE：结束位错误 ◦ TO：响应超时 ◦ CRC：响应数据 CRC 错误 ◦ RTO：数据读超时 ◦ CRC：数据接收 CRC 错误 ◦ RE：响应错误
29:6	RESET	–
5	ER (End of Ring)	置 1，表示描述符链表到达最后一个描述符。IDMAC 返回链表的基址，创建一个描述符环。仅在双缓存描述符结构时有效。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

位域	位域名称	说明
4	CH (Second Address Chained)	置 1，表示描述符的第二个地址是下一个描述符的地址，不是第二个缓存的地址。该位域置 1 时，DES1[25:13] 应该清零。
3	FS (Fist Descriptor)	置 1，表示描述符包含第一个缓存的数据。如果第一个缓存的容量为 0，下一个描述符包含起始数据。
2	LD (Last Descriptor)	与一次 IDMAC 传输的最后一个 block 有关。 • 置 1，表示缓存指向该描述符的最后一个缓存数据。 • 描述符结束后，剩余的字节数为 0。即描述符完成后且该位域置 1 时，剩余的字节数为 0。
1	DIC (Disable Interrupt on Completion)	置 1，防止 IDMAC 状态寄存器的 bit0/ 1 置位。
0	RES	–

DES1 用于配置每个描述符的数据长度，说明如下：

位域	位域名称	说明
31:26	RES	–
25:13	BS2 (Buffer 2 Size)	缓存 2 容量，表示第二个数据缓存的字节容量。 • 缓存容量必须按 32 位对齐。 • 置 0，DMA 忽略该缓存，跳到下一个双缓存结构的缓存。 • 对于链式结构，该位域无效。此时 DES0[4] 置 1。 • 如果在任意的描述符里该位被置 0，剩下的描述符不能为非零值，直到最后一个描述符。
12:0	BS1 (Buffer 1 Size)	缓存 1 容量，表示第一个数据缓存的字节容量。 • 缓存容量必须按 32 位对齐。 • 该位域不能为 0。 **注：** 如果仅有一个缓存被编程，必须使用缓存 1，而不是缓存 2。

DES2 用于配置第一个描述符的起始地址，说明如下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

位域	位域名称	说明
31:0	缓存地址指针 1	• 使用双缓存结构时，表示第一个数据缓存的物理地址。 • 对于链式结构，表示数据缓存的物理地址。

DES3 用于配置下一个描述符的起始地址，说明如下：

位域	位域名称	说明
31:0	缓存地址指针 2	使用双缓存结构时，表示第二个缓存的物理地址。如果 DES0[4] 置 1，该地址指向下一个描述符的地址。如果不是最后一个描述符，下一个描述符的地址必须按 32 位对齐，即 bit[1:0] 为 0，内部最低有效位被忽略。

9.1.3.3. 卡接口模块 (CIF)

卡接口模块连接 HIF 和 外部 SD/ MMC Card/ SDIO 设备。卡接口模块参与的工作流程如下：

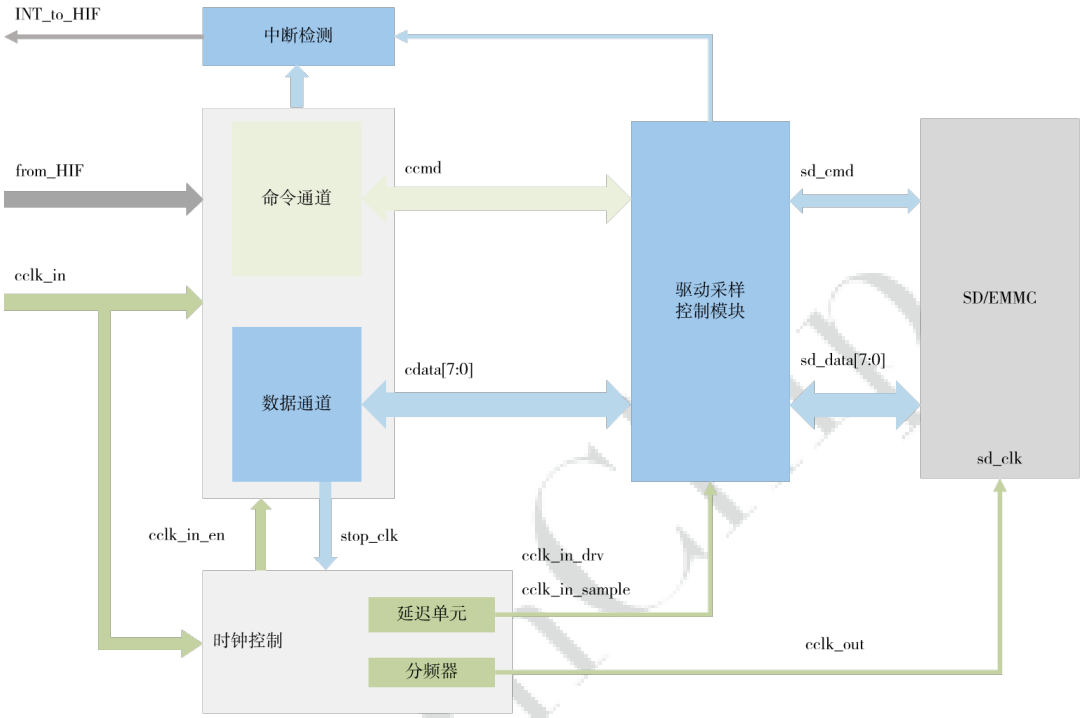
- CPU 向 HIF 的控制寄存器写入命令参数。
- 控制器将命令参数发送至 CIF。
- CIF 根据控制寄存器的值，生成命令和数据。命令和数据会根据不同的设备协议，在对应的通道进行传输。

为了使 CIF 正确的工作，软件必须满足如下要求：

- 同一时间仅能发送一个命令。
- 对于开放式的写数据，如果因为 FIFO 为空造成 Card 时钟停止了，软件必须向 FIFO 填满数据，才能再次启动时钟。
- 对于读数据，如果 Card 时钟因为 FIFO 满而停止，软件应该继续读至少两个 FIFO 地址，再启动 Card 时钟。
- 当 Card 正在处理数据传输时，发送卡复位命令（CMD0、CMD15 或 CMD52），软件必须设置命令寄存器的 STOP_ABORT_CMD 位，确保发送复位命令之前先停止数据传输。
- 对于 SDIO 传输，如果 SDIO 设备挂起，软件想要恢复挂起的传输，在发送新的数据传输命令时，必须复位 FIFO 并发送恢复命令。
- 当 OINTST 寄存器的 EBE 置位，控制器不能及时处理 SDIO 中断。软件应该忽略 SDIO 中断，发送停止或丢弃命令给 SDIO 设备，确保 SDIO 设备停止发送读数据。

CIF 模块主要包括命令通路、数据通路、中断检测单元、时钟控制单元和驱动采样控制单元，框图如下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



9.1.3.3.1. 命令通道

命令通道主要功能如下：

- 加载时钟参数
- 加载 Card 命令参数
- 发送命令到 Card 总线
- 从 Card 总线接收设备响应
- 发送响应到 HIF

命令通道的工作流程如下：

1. 通过对 HIF 寄存器编程发送一个新的命令，将命令寄存器的 STAT_CMD 位置 1。
2. HIF 的 STAT_CMD 位有效，表示有新的命令被发送。
3. 命令通路加载新的命令，并发送一个应答信号给 HIF。
4. 命令加载后，命令通路的状态机发送命令给 SD/ MMC 总线，包括内部产生的 CRC7 校验码，并接收 Device 返回的响应数据。
5. 状态机发送接收的响应数据和命令完成信号给 HIF，并等待八个时钟周期，再加载新的命令。

9.1.3.3.2. 数据通道

数据通路模块加载新的数据参数，包括是否传输数据、读或写数据传输方向、流传输或块传输方式、数据块容量、字节数量、Card 类型和超时时间。

数据通路在发送和接收数据时的行为如下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 在发送数据时，从 FIFO 中取出数据通过 cdata 发送数据。
- 在接收数据时，通过 cdata 将接收到的数据送入 FIFO。

如果命令寄存器的 DATA_EXP 位置 1，新的传输命令会根据如下情况在数据通道执行：

- 如果 TRANS_DIR 位为 1，发送数据。
- 如果 TRANS_DIR 位为 0，接收数据。

9.1.3.3.3. SDIO 中断检测

所有来自卡的中断请求报告给 HIF，中断信号的有效时间需要维持两个时钟周期。对于 SDIO 设备，在中断周期内通过保持 Data[1] 引脚为低电平来产生中断信号。

通过中断控制状态机决定被选中的设备工作在中断周期内。

- 对于未选中的设备，或被选中的设备工作在 1 线模式下，中断周期总是有效的。
- 对于被选中的设备，且工作在 4 线模式下，需要满足以下条件之一，中断周期才是有效的：
  - 设备空闲时产生中断。
  - 没有正在处理的数据传输命令时产生中断。
  - 在两个数据块之间的结束位之后的第三个时钟产生中断。
  - 上一笔数据结束位之后的两个时钟到下一笔数据命令的结束位期间产生中断。

SDIO 中断为电平触发，需要在一个中断周期内进行采样，控制器才不会丢失该中断。当 SDIO 设备工作在 4 线模式时，在以下任意情况下，控制器不会采样被选中卡的 SDIO 中断：

- 读或写恢复：CIF 将恢复命令当作正常的数据传输命令处理。SDIO 中断在恢复命令处理时，与其它数据命令类似。根据 SDIO 协议：
  - 对于正常的数据命令，在命令结束位之后中断周期结束。
  - 对于恢复命令，在响应结束位之后中断周期结束。在恢复命令的结束位之后，停止中断采样。
- 读数据期间挂起：如果读数据传输被控制器挂起，CPU 将 ABORT_RDATA 位域置 1 并复位数据状态机。
  - 在 CIF 中，从恢复命令响应到 ABORT_RDATA 位域置 1 期间，SDIO 中断不会被采样。
  - 在 Host 将 ABORT_RDATA 位域置 1 后，SDIO 中断被处理，启动中断采样。

9.1.3.3.4. 时钟控制单元

时钟控制单元提供了不同的时钟频率来满足 SD/ MMC 不同速度模式下的频率要求。cclk_in 信号是源时钟（频率大于等于 Card 最大工作时钟频率），由 PLL 分频而来，再通过 SDMC 控制器内部的时钟分频器分频后作为输出时钟 cclk_out 发送给 Card。

时钟控制单元包括以下功能：

- 打开和关闭时钟。
- 控制器进入低功耗模式，在传输空闲时，自动关闭时钟。
- 内部包含延时单元和移相器，提供驱动时钟和采样时钟，参考时钟和延时控制的详细描述。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.1.3.3.5. 错误检测

SDMC 控制器可检测以下错误类型：

表 9–1 错误检测

错误分类	错误详情
响应错误	
响应超时	计时器到达超时寄存器配置的超时时间后，未接收到响应的起始位。
响应 CRC 错误	收到期望的响应并检查响应数据的 CRC 校验码，响应数据中的 CRC7 校验码与内部生成的校验码不匹配。
响应错误	响应的传输位不为 0，命令序列与发送的命令序列不匹配，或响应的结束位不为 1。
数据发送错误	
无 CRC 状态信号	在一次写数据期间，如果在数据块发出之后的两个时钟周期后，CRC 状态信号的起始位没有接收到，产生无 CRC 状态的错误。
无效的 CRC	如果在写数据块发送之后产生无效的 CRC 状态（非 010），数据 CRC 错误信号会发送到 HIF，数据传输继续进行。
由于 FIFO 为空造成的 Data Starvation 错误	如果在写数据时 FIFO 为空，或 Card 时钟停止且数据超时后 FIFO 剩余的数据为空，data-starvation 错误信号会发送到 HIF 单元，数据通道继续等待数据写入 FIFO。
数据接收错误	
数据超时	在读数据时，如果计时器到达超时时间后，没有接收到数据的起始位，会发生数据超时错误。
数据起始位错误	在 4 位或 8 位模式读数据时，如果发生起始位错误，应用程序必须发送 CMD12 或 CMD52 结束错误状态。在命令接收完成后，应用程序必须复位 IDMAC 和 CIF，并清除错误状态。发送任何数据传输命令之前都必须复位 FIFO。
数据 CRC 错误	在读数据块传输期间，如果接受到的 CRC16 校验码与内部产生的 CRC16 校验码不匹配，数据通道发送数据 CRC 错误的信号给 HIF，继续传输数据。
数据结束位错误	在读数据期间，如果接收到的数据结束位不为 1，数据通道发送结束位错误信号给 HIF，并终止数据传输，该信号发送 HIF 则数据传输完成。
由于 FIFO 满造成的 Data Starvation 错误	在读数据传输期间，当 FIFO 为满，Card 时钟停止。如果 FIFO 满并且数据超时，data-starvation 错误信号发送到 HIF（此时 OINTST 寄存器的 bit10 为 1），数据通道继续等待 FIFO 为空。
Host 控制器不能正确的采样到数据	如果在读操作时 Card 时钟因为 FIFO 满而停止，Host 控制器不能正确的采样到数据。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

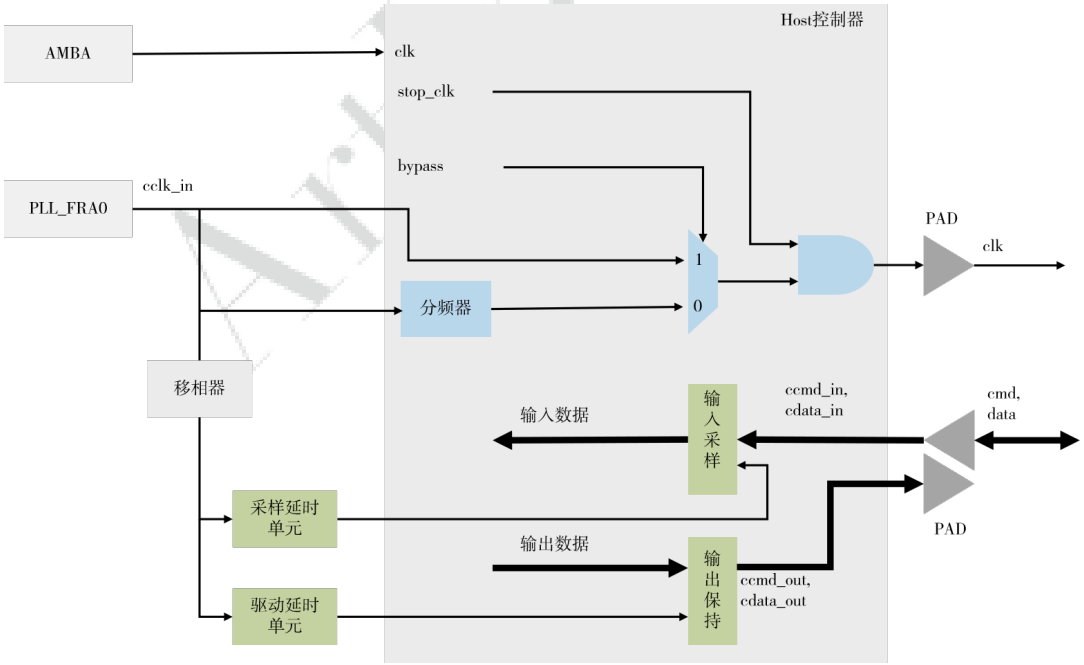
9.1.3.4. 时钟和延时控制

9.1.3.4.1. 时钟域

控制器内部有多种不同的时钟域，各个时钟域之间的关系描述如下：

- clk：HIF 模块工作时钟和 AHB 总线时钟，所有的配置寄存器使用该时钟。
- cclk_in：输入时钟，CIF 模块大部分模块和寄存器工作在该时钟。
- cclk_in_drv：驱动时钟，输入时钟经过移相器和延迟单元输出的驱动时钟，用于数据和命令的输出驱动。该时钟用来满足不同速度模式下保持时间的要求。
- cclk_in_sample：采样时钟，输入时钟经过移相器和延迟单元输出的采样时钟，用于数据和命令接收采样。
- cclk_out：输出时钟，由 cclk_in 分频后输出给 Device 的工作时钟。

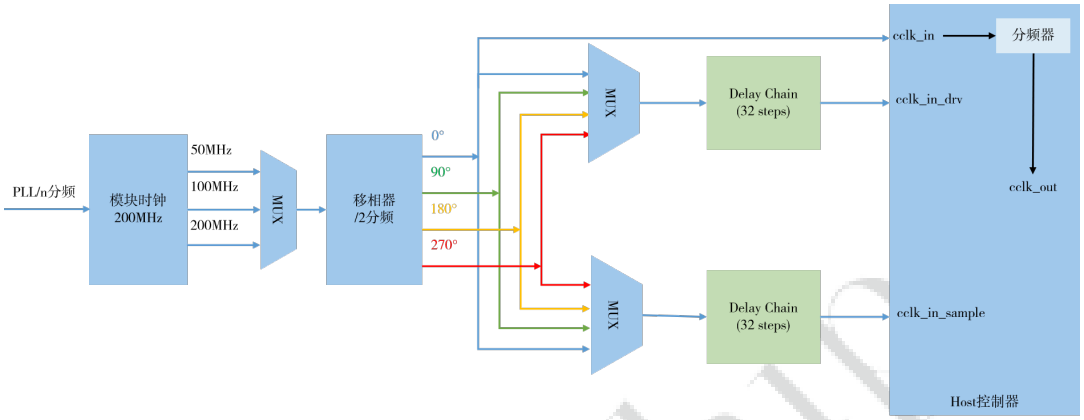
控制器时钟域框图如下：



9.1.3.4.2. 延时控制

由于控制器内部由多种不同的时钟组成，输出驱动时钟和采样时钟需要经过一个相移器和延迟单元进行延迟，以满足不同速度等级的 SD/ MMC 设备稳定的工作时序要求。 因此，需要在 SDMC 控制器外部实现相移和延迟电路。结构框图如下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



延时控制单元的工作流程如下：

1. CMU 模块对 PLL 进行分频，输入到 SDMC 控制器的模块时钟频率最大为200 MHz 。
2. 配置时钟选择复用器 EXT_CLK_MUX 位域，选择控制器输入时钟。

   为满足不同的速度等级要求，共有以下几种模式选择：

   - 2b'00：25 MHz，对应速度模式：DS，HS-SDR，HS-DDR
   - 2b'01：50 MHz，对应速度模式：HS，HS-SDR，HS-DDR
   - 2b'10：100 MHz，对应速度模式：HS-DDR

   速度模式等级的说明如下：

   - DS：SD 卡默认速度模式，工作电平 3.3 V，工作频率 0~25 MHz，最大传输速率 25 Mbps。
   - HS：SD 卡高速模式，工作电平 3.3 V，工作频率 0~50 MHz，最大传输速率 50 Mbps。
   - HS-SDR：eMMC 高速 SDR 模式，工作电平 3.3 V，工作频率 0~50 MHz，最大传输速率 50 Mbps。
   - HS-DDR：eMMC 高速 DDR 模式，工作电平 3.3 V，工作频 率 0~50 MHz，最大传输速率 100 Mbps。
3. 经过时钟复用器选择后，时钟进入移相器，通过 2 分频产生四个不同相位的时钟。通过 SDMC_DLYCTRL 寄存器对时钟相位编程选择，共有四种相位可编程：

   - 0° ，相位与输入时钟相同
   - 90° ，延时四分之一个时钟周期
   - 180° ，延时二分之一个时钟周期
   - 270° ，延时四分之三个时钟周期
4. 每个相位之后再经过 Delay Chain 进行 32 级的延迟。

   Delay Chain 内部由若干个延时单元组成，每个延时单元的延时时间相同。通过 SDMC_DLYCTRL 寄存器对 Delay Chain 进行编程选择。

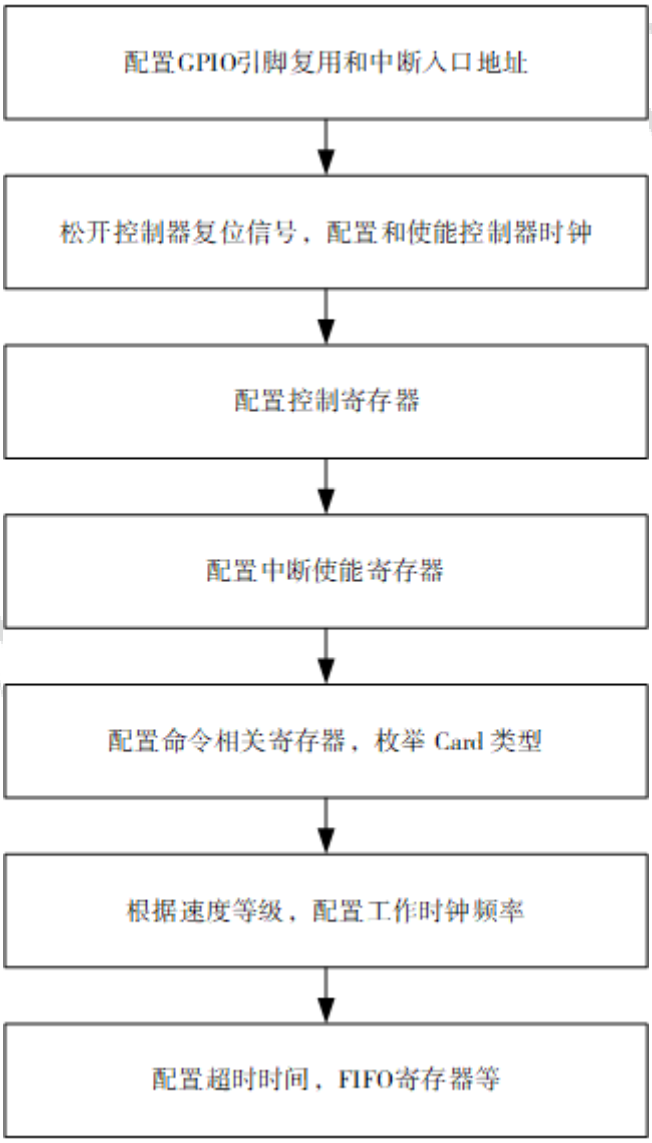
   - 未经过相位延迟和 Delay Chain 的时钟作为控制器的输入时钟 cclk_in。
   - 经过控制器内部的分频器分频输出 cclk_out，作为外设的接口时钟。
5. 经过内部时钟分频器进行分频（通过配置 CLK_DIV 位域设置分配系数）后输出符合需求的接口工作时钟。

Copyright © 2023-2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.1.4. 编程指南

9.1.4.1. 初始化流程

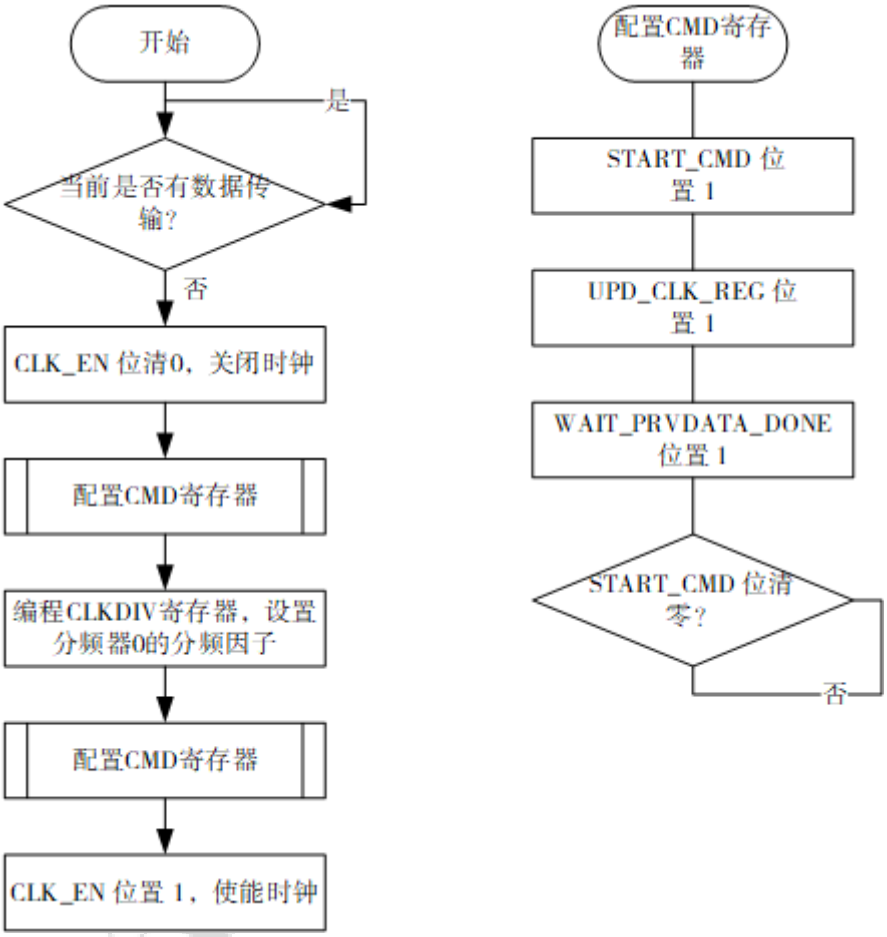
在进行数据传输之前，需要初始化控制器，初始化流程图如下：



9.1.4.2. 时钟配置流程

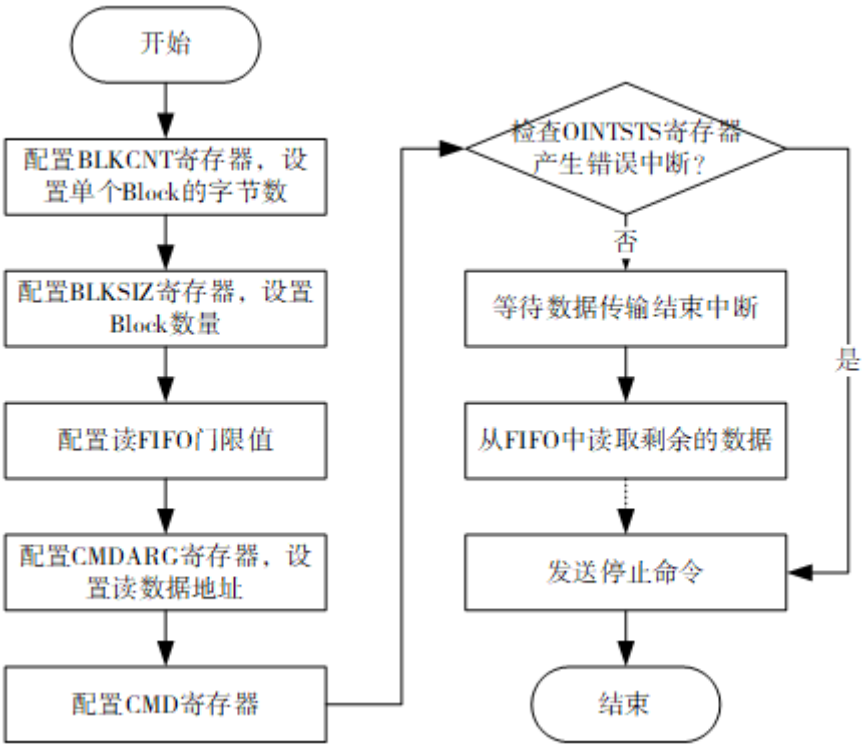
建议按照以下流程进行时钟配置：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



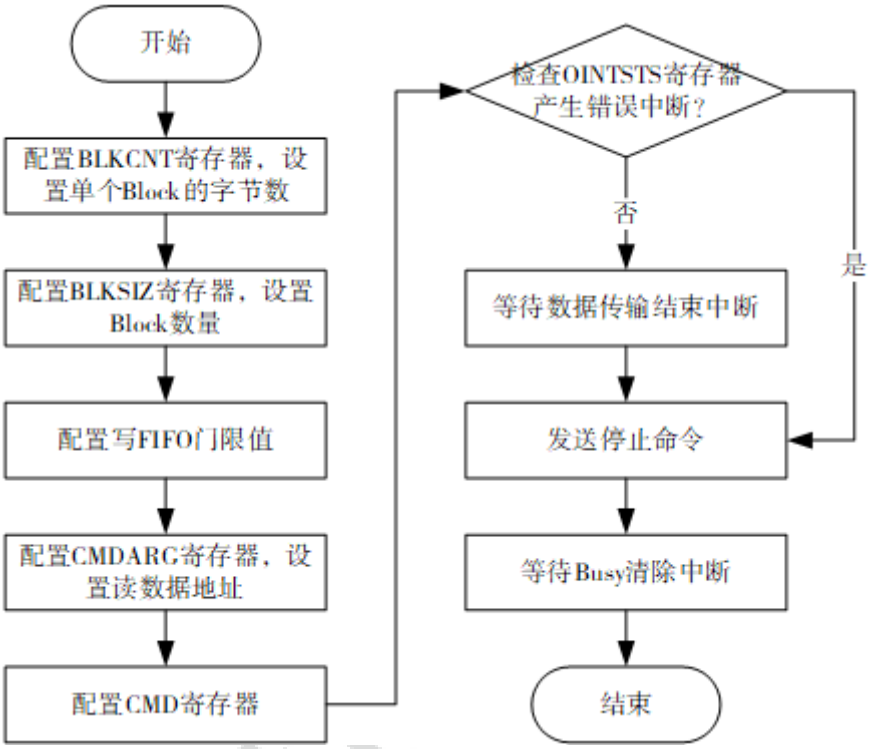
9.1.4.3. 数据传输

读数据流程图



写数据流程图

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



CMD 寄存器配置表如下：

表 9–2 CMD 寄存器配置

位域	值	说明
START_CMD	1	发送命令完成后自动清零
USE_HOLD_REG	1 或 0	基于速度模式选择是否使能保持寄存器
UPD_CLK_REG	–	非时钟参数刷新命令
SEND_INIT	–	非初始化序列
STOP_ABORT_CMD	–	非停止或丢弃命令
AUTO_STOP	0 或 1	根据需要选择是否自动发送停止命令
TRANS_MODE	–	Block 传输
TRANS_DIR	0 或 1	• 0：从 Card 读数据 • 1：写数据到 Card
DATA_EXP	1	数据命令
RESP_LEN	–	响应长度
RESP_EXP	1	命令有响应返回则置 1

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 9-2 CMD 寄存器配置 (续)

位域	值	说明
CMD_INDEX	命令序列	读或写命令序列
WAIT_PRVDATA_DONE	1	• 0：立即发送命令 • 1：等待之前的数据传输结束后再发送命令
CHECK_RESP_CRC	1	• 0：不检查响应的 CRC 校验值 • 1：检查响应的 CRC 校验值

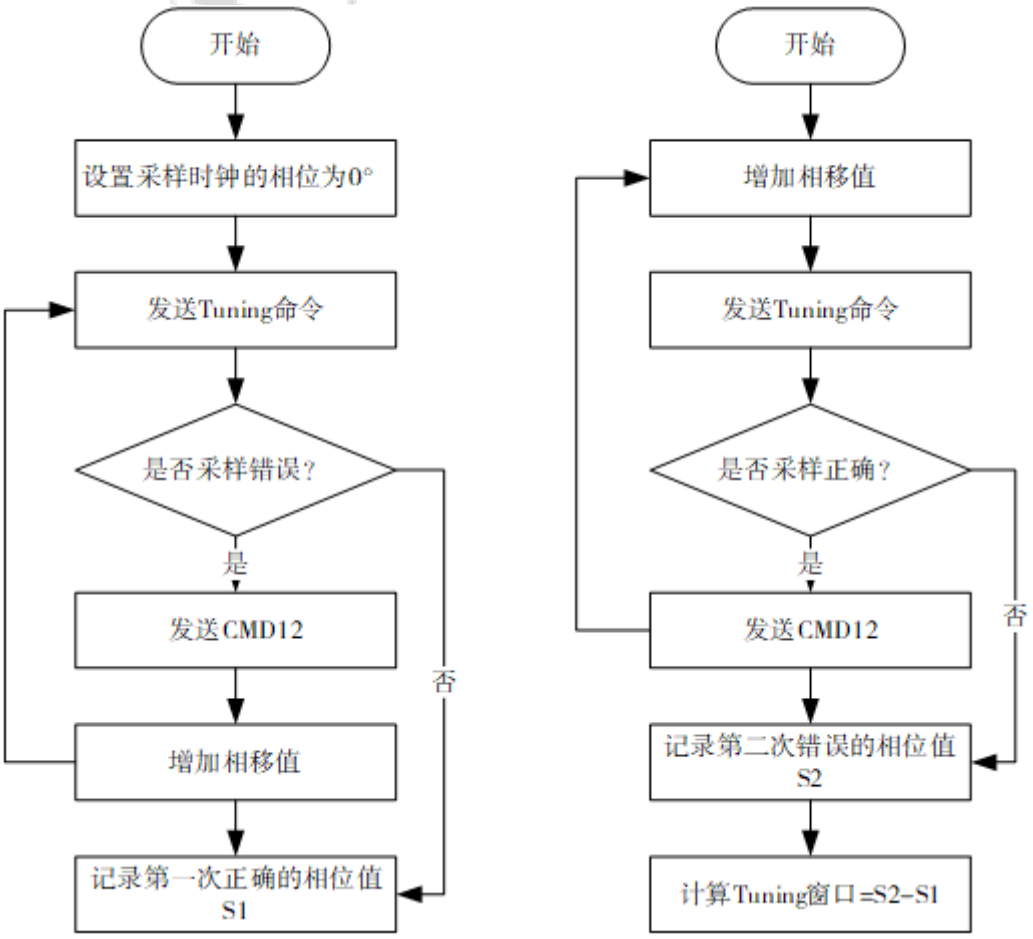
9.1.4.4. 软件 Tuning 采样点

Tuning 用于 Host 纠正读数据时的采样点，利用移相器或 Delay Chain 来寻找最佳的采样点，补偿时序偏差。

时序偏差主要由工艺偏差、PCB 走线、电压和温度变化因素引起。

在 DDR50 模式下，即使此时来自 Card 的输出延迟小于一个时钟周期，但是在 Host 控制器内部输入 PAD 的延迟足够大，使整体的读延时大于一个时钟周期，所以需要进行软件 tuning。

Tuning 最佳采样点，计算 tuning 窗口的流程如下：



Copyright © 2023-2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.1.4.5. DDR 模式编程

对于 eMMC 设备，控制器支持 DDR 模式，支持 4 线 模式。

使用 DDR 模式时需要将时钟控制寄存器的 CLK_DIV 位配置为 0，通过调整采样相位和 Delay Chain 找到最佳的采样点。此时，时钟频率配置有两种方式：

- 通过 CMU 模块分频配置外部 PLL 输入到 SDMC 模块的时钟，最大支持 32 分频。
- 通过外部时钟输入选择控制寄存器，配置输入到 SDMC 模块的时钟，仅支持 1/ 2/ 4 分频。

DDR 模式 tuning 最佳采样点工作流程如下：

1. 初始化 SDMC 控制器和 eMMC，在 SDR 模式下正确读写多个 block 数据。
2. 配置 CLK_DIV 位分频系数为 0，配置工作时钟为 50 MHz。
3. 配置 eMMC 工作模式为 HS-DDR 模式，控制器总线配置为 DDR 模式。
4. 先 tuning 读数据，遍历配置采样相位和 Delay Chain，采样并比较数据，选择窗口最大的一组连续正确的采样点，并计算中间值作为最佳采样点。
5. 配置最佳采样点的相位值和 Delay Chain 值，tuning 写操作，选择窗口最大的一组连续正确的驱动值，并计算中间值作为最佳的驱动值。
6. 配置最佳的驱动值，再次进行读写测试。

9.1.5. 寄存器列表

表 9-3 SDMC 寄存器列表

偏移	缩写	寄存器名称	章节
0x000	SDMC_BLKCNT	传输块数量	0x000 SDMC_BLKCNT
0x004	SDMC_BLKSIZE	传输块大小	0x004 SDMC_BLKSIZ
0x008	SDMC_CMDARG	命令参数	0x008 SDMC_CMDARG
0x00C	SDMC_CMD	命令配置	0x00C SDMC_CMD
0x010	SDMC_RESP0	响应数据 0	0x010 SDMC_RESP0
0x014	SDMC_RESP1	响应数据 1	0x014 SDMC_RESP1
0x018	SDMC_RESP2	响应数据 2	0x018 SDMC_RESP2
0x01C	SDMC_RESP3	响应数据 3	0x01C SDMC_RESP3
0x020	SDMC_TTMC	传输超时时间控制	0x020 SDMC_TTMC
0x024	SDMC_TCBC	发送到 Card 的字节数	0x024 SDMC_TCBC

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 9–3 SDMC 寄存器列表 (续)

偏移	缩写	寄存器名称	章节
0x028	SDMC_TFBC	发送到 FIFO 的字节数	0x028 SDMC_TFBC
0x02C	SDMC_CTRST	控制器状态	0x02C SDMC_CTRST
0x030	SDMC_HCTRL1	主机控制配置 1	0x030 SDMC_HCTRL1
0x034	SDMC_CLKCTRL	时钟控制配置	0x034 SDMC_CLKCTRL
0x038	SDMC_HCTRL2	主机控制配置 2	0x038 SDMC_HCTRL2
0x03C	SDMC_INTEN	中断使能	0x03C SDMC_INTEN
0x040	SDMC_INTST	已使能中断状态	0x040 SDMC_INTST
0x044	SDMC_OINTST	初始中断状态	0x044 SDMC_OINTST
0x048	SDMC_FIFOCFG	FIFO 配置	0x048 SDMC_FIFOCFG
0x054	SDMC_CDET	卡检测配置	0x054 SDMC_CDET
0x080	SDMC_PBUSCFG	外设总线配置	0x080 SDMC_PBUSCFG
0x084	SDMC_IDMARCAP	IDMAC 恢复获取描述符	0x084 SDMC_IDMARCAP
0x088	SDMC_IDMASADDR	IDMAC 描述符起始地址	0x088 SDMC_IDMASADDR
0x08C	SDMC_IDMAST	IDMAC 状态	0x08C SDMC_IDMAST
0x090	SDMC_IDMAINTEN	IDMAC 中断使能	0x090 SDMC_IDMAINTEN
0x094	SDMC_IDMACDA	IDMAC 描述符地址指针	0x094 SDMC_IDMACDA
0x098	SDMC_IDMACBA	IDMAC 缓存地址指针	0x098 SDMC_IDMACBA
0x100	SDMC_CTC	卡门限值配置	0x100 SDMC_CTC
0x104	SDMC_DLYCTRL	延时链和相位控制	0x104 SDMC_DLYCTRL
0x108	SDMC_EMCR	EMMC 配置	0x108 SDMC_EMCR
0x118	SDMC_VERID	控制器版本号	0x118 SDMC_VERID
0x200	SDMC_FIFO	数据 FIFO 缓存地址	0x200 SDMC_FIFO

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.1.6. 寄存器描述

9.1.6.1. 0x000 SDMC_BLKCNT

默认值：0x00000001			传输块数量 (Block Count)
位域	类型	默认值	描述
31:16	–	–	–
15:0	R/W	0x1	BLK_CNT 传输块数量 (Block Count) 对于开放式的传输模式，控制器会连续的传输数据，此时该位应该设置为 0。当该位设置为 0 时，需要软件配置发送停止命令，终止当前数据传输。

9.1.6.2. 0x004 SDMC_BLKSIZ

默认值：0x00000200			传输块大小 (Block Size)
位域	类型	默认值	描述
31:16	–	–	–
15:0	R/W	0x200	BLK_SIZE 传输块大小 (Block Size) 配置单个传输块的字节数量，默认值为 512 个字节。

9.1.6.3. 0x008 SDMC_CMDARG

默认值：0x00000000			命令参数 (Command Argument)
位域	类型	默认值	描述
31:0	R/W	0x0	CMD_ARG 发送给 Card 的命令参数 (Command Argument)

9.1.6.4. 0x00C SDMC_CMD

默认值：0x20000000			命令配置 (Command Configure)
位域	类型	默认值	描述
31	R/W	0x0	START_CMD

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x20000000			命令配置 (Command Configure)
位域	类型	默认值	描述
			开始发送命令 (Start Send Command) • 一旦命令被 CIF 获取，该位自动清零。 • 当该位置 1，Host 不能再写任何命令到寄存器。如果此时试图写新的命令，硬件锁定错误中断被置 1。 • 一旦命令发出并接收到响应，命令完成中断被置 1。
30	–	–	–
29	R/W	0x1	USE_HOLD_REG 使用保持寄存器 (Use Hold Register) • 0：命令和数据直接发送到卡，保持时间没有附加延时。 • 1：命令和数据通过保持寄存器发送到卡，保持时间增加 0.8 ns 延时。
28	R/W	0x0	VOLT_SWITCH SD 卡电压切换 (Voltage Switch) • 0：无电压切换 • 1：电压切换使能，对于卡电压切换命令 (CMD11) 必须置位。
27	R/W	0x0	BOOT_MODE 启动引导模式选择 (Boot Mode Select) • 0：执行默认的引导操作 • 1：执行替代的引导操作
26	R/W	0x0	BOOT_DIS 禁止启动引导操作 (Boot Mode Disable) • 不能与 BOOT_EN 位同时置 1。 • 与 START_CMD 同时置 1 时，CIF 会终止引导操作。
25	R/W	0x0	BOOT_ACK_EXP 接收引导操作应答信号 (Boot Ack Expect) • 与 BOOT_EN 位同时置 1 时，CIF 接收来自卡的引导操作应答信号 0–1–0。
24	R/W	0x0	BOOT_EN 使能启动引导操作 (Boot Mode Enable)

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x20000000			命令配置 (Command Configure)
位域	类型	默认值	描述
			• BOOT_MODE = 0，即执行默认引导模式时，必须置 1。 • 不能与 BOOT_DIS 位同时置 1。 • 与 START_CMD 位同时置 1 时，CIF 开始发送引导序列，对应卡的命令线为低电平。
23:22	–	–	–
21	R/W	0x0	UPD_CLK_REG 仅更新时钟寄存器 (Update Clock Registers Only) • 0：正常命令序列 • 1：不发送命令，仅更新时钟寄存器的值到相关的时钟域
20:16	–	–	–
15	R/W	0x0	SEND_INIT 发送初始化序列 (Send Initialization Sequence) • 0：发送命令之前不发送初始化序列 • 1：发送命令之前发送初始化序列 上电后，发送第一个命令之前，必须发送 80 个时钟，等待 Card 内部初始化完成。在 Boot 模式下，该位不用置 1。
14	R/W	0x0	STOP_ABORT_CMD 发送停止或丢弃命令 (Send Stop or Abort Command) • 0：在数据传输时，不使用停止或丢弃命令终止当前传输。 • 1：在数据传输时，使用停止或丢弃命令终止当前传输。 当工作在开放或者预定义数据传输模式时，Host 需要发送停止或丢弃命令来终止数据传输。该位应该置 1，确保 CIF 的命令、数据状态机正确的返回到空闲状态。
13	R/W	0x0	WAIT_PRVDATA_DONE 等待之前的数据传输完成 (Wait Previous Data Done) • 0：即使之前的数据传输未完成，也可以再次发送命令 • 1：必须等待之前的数据传输完成，再发送命令 通常，在数据传输期间或者终止当前传输时，如果想要查询外部设备当前的状态，可以将该位清零。
12	R/W	0x0	AUTO_STOP 自动发送停止命令 (Auto Send Stop Command)

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x20000000			命令配置 (Command Configure)
位域	类型	默认值	描述
			• 0：数据传输结束时，控制器不自动发送停止命令 • 1：数据传输结束时，控制器自动发送停止命令
11	R/W	0x0	TRANS_MODE 传输模式 (Transfer Mode) • 0：基于块的数据传输命令 • 1：基于字节流的数据传输命令
10	R/W	0x0	TRANS_DIR 传输方向 (Transfer Direction) • 0：从卡读数据 • 1：写数据到卡
9	R/W	0x0	DATA_EXP 有期望数据的传输 (Data Expected) • 0：无数据的命令传输 (read/write) • 1：有期望数据的命令传输 (read/write)
8	R/W	0x0	CHECK_RESP_CRC 检查响应信号的 CRC (Check Response CRC) • 0：不检查响应的 CRC 对于不返回有效 CRC 数据的命令，应禁止 CRC 检查。 • 1：检查响应的 CRC
7	R/W	0x0	RESP_LEN 响应信号长度 (Response Length) • 0：从卡获取短响应数据 (48–bit) • 1：从卡获取长响应数据 (136–bit)
6	R/W	0x0	RESP_EXP 获取响应信号 (Response Expect) • 0：不从卡获取响应信号 • 1：从卡获取响应信号
5:0	R/W	0x0	CMD_INDEX 命令序号 (Command Index)

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.1.6.5. 0x010 SDMC_RESP0

默认值：0x00000000			响应数据 0 (Response 0)
位域	类型	默认值	描述
31：0	R	0x0	RESP0 响应数据的 bit[31:0]

9.1.6.6. 0x014 SDMC_RESP1

默认值：0x00000000			响应数据 1 (Response 1)
位域	类型	默认值	描述
31:0	R	0x0	RESP1 响应数据的 bit[63:32]

9.1.6.7. 0x018 SDMC_RESP2

默认值：0x00000000			响应数据 2 (Response 2)
位域	类型	默认值	描述
31:0	R	0x0	RESP2 响应数据的 bit[95:64]

9.1.6.8. 0x01C SDMC_RESP3

默认值：0x00000000			响应数据 3 (Response 3)
位域	类型	默认值	描述
31:0	R	0x0	RESP3 响应数据的 bit[127:96]

9.1.6.9. 0x020 SDMC_TTMC

默认值：0xffffff40			传输超时时间控制 (Transfer Timeout Control)
位域	类型	默认值	描述
31:8	R/W	0xffffff	DATA_TMO 读数据传输超时时间 (Data Timeout)

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0xffffff40			传输超时时间控制 (Transfer Timeout Control)
位域	类型	默认值	描述
			超时计数器仅在 Card 时钟停止后启动。 单位：输出时钟的时钟周期 **注：** 如果超时值接近 100 ms，需要使用软件配置 GTC 计时器用于超时检测。在这种情况下，应该屏蔽读数据超时中断（SDMC_INTEN 寄存器的 bit9 清 0）。
7:0	R/W	0x40	RESP_TMO 响应超时时间 (Response Timeout) 单位：输出时钟的时钟周期

9.1.6.10. 0x024 SDMC_TCBC

默认值：0x00000000			发送到卡的字节数 (Transferred Card Byte Count)
位域	类型	默认值	描述
31:0	R	0x0	CBC 传输到卡的字节数 (Transferred Card Byte Count) 表示经过 CIF 模块已传输到 Card 的字节数。

9.1.6.11. 0x028 SDMC_TFBC

默认值：0x00000000			发送到缓存的字节数 (Transferred FIFO Byte Count)
位域	类型	默认值	描述
31:0	R	0x0	FBC 传输到 FIFO 的字节数 (Transferred FIFO Byte Count) 表示 Host 或 DMA 在内存和 HIF FIFO 之间已传输的字节数

9.1.6.12. 0x02C SDMC_CTRST

默认值：0x00000206			控制器状态 (Controller Status)
位域	类型	默认值	描述
31	R	0x0	DMA_REQ DMA 请求信号状态 (DMA Request Status)

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000206			控制器状态 (Controller Status)
位域	**类型**	**默认值**	**描述**
30	R	0x0	DMA_ACK DMA 应答信号状态 (DMA Acknowledge Status)
29:17	R	0x0	FIFO_CNT FIFO 数量 (FIFO Count) 当前 FIFO 中的字节数
16:11	R	0x0	RESP_INDEX 前一个响应的索引值 (Previous Response Index)
10	R	0x0	DATA_MC_BUSY 数据发送或接收状态机忙碌 (Data State Machine Busy)
9	R	0x1 or 0x0	DATA0_BUSY 根据 data[0] 的电平状态判断是否忙碌 • 0：卡数据传输处于非忙碌状态 • 1：卡数据传输处于忙碌状态 **注：** 默认值取决于 data[0] 的初始电平。
8	R	0x1 or 0x0	DATA3_STATUS 根据 data[3] 的电平状态检查是否有卡插入 • 0：没有 SD 卡插入 • 1：有 SD 卡插入 **注：** 卡检测仅支持 SDC1，默认值取决于 data[3] 的初始电平。
7:4	R	0x0	CMD_STAT 命令状态机状态 (Command FSM States) • 0：空闲 • 1：发送初始化序列 • 2：Tx 命令开始位 • 3：Tx 命令 tx 位 • 4：Tx 命令索引值 + 参数 • 5：Tx 命令 CRC7 校验

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000206			控制器状态 (Controller Status)
位域	类型	默认值	描述
			• 6：Tx 命令结束位 • 7：Rx 响应起始位 • 8：Rx 响应中断请求响应 • 9：Rx 响应 tx 位 • 10：Rx 响应命令索引值 • 11：Rx 响应数据 • 12：Rx 响应 CRC7 校验 • 13：Rx 响应结束位 • 14：命令通路等待 NCC • 15：等待状态
3	R	0x0	FIFO_FULL FIFO 为满
2	R	0x1	FIFO_EMPTY FIFO 为空
1	R	0x1	FIFO_TX_WMARK FIFO 发送数据的水位 (FIFO TX Watermark) FIFO 水位达到发送数据门限值，不允许数据传输
0	R	0x0	FIFO_RX_WMARK FIFO 接收数据的水位 (FIFO RX Watermark) FIFO 水位达到接收数据门限值，不允许数据传输

9.1.6.13. 0x030 SDMC_HCTRL1

默认值：0x00000008			主机控制配置 1 (Host Control 1)
位域	类型	默认值	描述
31:26	–	–	–
25	R/W	0x0	USE_IDMAC 使用内部 DMA 控制器 (Use Inside DMA Controller) • 0：Host 通过 AHB 从机接口进行数据传输 • 1：仅使用内部 DMA 传输数据

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000008			主机控制配置 1 (Host Control 1)
位域	**类型**	**默认值**	**描述**
24	R/W	0x0	OD_PUP_EN 外部开漏上拉使能 (Enable Open Drain Pull-up) • 0：禁止 • 1：使能。置 1，命令信号的输出为开漏模式，命令信号驱动为低电平或者高阻态，不能驱动为高电平。
23:11	–	–	–
10:9	–	–	–
8	R/W	0x0	ABORT_RDATA 丢弃读数据 (Abort Read Data)，用于 SDIO 的暂停操作。 • 0：不改变 • 1：在读数据时，发送暂停命令之后，软件循环查询 SDIO 设备是否进入暂停状态。当暂停操作发生时，软件将该位置 1 用于复位数据状态机，等待下一笔数据传输。一旦数据状态机复位到空闲状态，该位会自动清零。
7	R/W	0x0	SEND_IRQ_RESP 发送中断请求响应 (Send IRQ Response) • 0：不改变 • 1：自动发送中断请求并接收响应 一旦响应被发送该位自动清零。 为了等待 MMC 卡中断，Host 发送 CMD40 命令，控制器等待来自 MMC 卡的中断响应。 如果用户想要控制器退出中断状态等待，可以将该位置 1，此时控制器命令状态机自动发送 CMD40，等待总线获取到响应后返回到空闲状态。
6	R/W	0x0	READ_WAIT 读等待 (Read Wait) • 0：清除读等待 • 1：使能读等待 用于 SDIO 卡设备读等待功能。 该位置 1，控制器将 Data[2] 引脚拉低暂时中止读数据，允许发送 CMD52 等其它 I/O 命令。
5	–	–	–
4	R/W	0x0	INT_EN 全局中断使能 (Global Interrupt Enable)

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000008			主机控制配置 1 (Host Control 1)
位域	**类型**	**默认值**	**描述**
			• 0：禁止中断 • 1：使能中断 仅当该位置 1，且对应的中断位置 1 时，对应的中断端口才为 1。
3	R/W	0x1	HW_RSTN 硬件 Reset 控制 (Hardware Reset Control) • 0：复位信号有效。Device 进入 pre-idle 状态，需要重新初始化 device。 • 1：复位信号无效
2	R/W	0x0	DMA_RESET DMA 复位 • 0：DMA 状态不改变 • 1：复位 IDMA 接口控制逻辑 经过两个 AHB 时钟周期后该位自动清零。
1	R/W	0x0	FIFO_RESET FIFO 复位 • 0：FIFO 状态不改变 • 1：复位数据 FIFO 和 FIFO 指针，复位完成后自动清零。 **注：** 该位清零后，FIFO 指针在 2 个周期的系统时钟 + 2 个周期的 Card 时钟后被复位。
0	R/W	0x0	CTRL_RESET 控制器复位 (Controller Reset) • 0：控制器状态不改变 • 1：控制器复位 经过两个 AHB 时钟周期后该位自动清零。 复位的模块包括但不限于： • HIF/ CIF 接口 • CIF 和状态机 • 控制寄存器的 ABORT_RDATA/ SEND_IRQ_RESP/ READ_WAIT 位域 • 命令寄存器的 START_CMD 位域 该复位不影响其它寄存器、DMA 接口、FIFO 和 Host 中断。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.1.6.14. 0x034 SDMC_CLKCTRL

默认值：0x00000000			时钟控制配置 (Clock Control)
位域	类型	默认值	描述
31:17	–	–	–
16	R/W	0x0	CLK_LP_EN 时钟低功耗模式使能 (Clock Low Power Enable) • 0：无低功耗模式 • 1：低功耗模式 ◦ 当 SD/ MMC 处于空闲状态时，关闭时钟。 ◦ 使用 MMC 或者 SD 卡时，置 1。 ◦ 对于 SDIO 卡，如果使用中断，时钟不能被停止。
15:8	R/W	0x0	CLK_DIV 输出时钟分频值 (Clock Divider Value) 时钟分频为 2*n （ n 为设定值 ）。
7:1	–	–	–
0	R/W	0x0	CLK_EN 输出时钟使能 (Clock Enable) • 0：关闭时钟 • 1：使能时钟

9.1.6.15. 0x038 SDMC_HCTRL2

默认值：0x00000000			主机控制配置 2 (Host Control 2)
位域	类型	默认值	描述
31:30	–	–	–
29:28	R/W	0x0	BUS_WIDTH 总线位宽配置 (Bus Width Configuration) • 0：1 线模式 • 1：4 线模式 • 其它值：预留
27:26	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			主机控制配置 2 (Host Control 2)
位域	类型	默认值	描述
25	R/W	0x0	DTO_UNIT 数据超时时间单位 (Data Timeout Unit) • 0：1 x cclk_out • 1：256 x cclk_out
24	R/W	0x0	RTO_UNIT 响应超时时间单位 (Response Timeout Unit) • 0：1 x cclk_out • 1：256 x cclk_out
23:17	–	–	–
16	R/W	0x0	DDR_MODE DDR 传输模式使能 (DDR Mode Enable) • 0：数据传输工作在 SDR 模式 • 1：数据传输工作在 DDR 模式
15:0	–	–	–

9.1.6.16. 0x03C SDMC_INTEN

默认值：0x000000000			中断使能 (Interrupt Enable)
位域	类型	默认值	描述
31:17	–	–	–
16	R/W	0x0	SDIO_INTEN SDIO 中断使能 (SDIO Interrupt Enable) 当中断被屏蔽后，SDIO 中断检测无效。 • 0：屏蔽中断 • 1：使能中断
15:0	R/W	0x0	INT_EN 中断使能 对于每个位域的值定义如下： • 0：屏蔽中断 • 1：使能中断

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x000000000			中断使能 (Interrupt Enable)
位域	类型	默认值	描述
			每个位的中断定义如下： • bit 15–EBE（读操作结束位错误或者写操作无 CRC） • bit 14–ACD（自动发送命令完成） • bit 13–SBE（起始位错误） BCI（忙清除中断） • bit 12–HLE（硬件锁定写操作错误） • bit 11–FRUN（FIFO 向下溢出/向上溢出） • bit 10–HTO（Host 获取数据超时）或者电压切换中断 • bit 9–DRTO（读数据超时） • bit 8–RTO（响应超时） • bit 7–DCRC（数据 CRC 错误） • bit 6–RCRC（响应 CRC 错误） • bit 5–RXDR（接收 FIFO 数据请求） • bit 4–TXDR（发送 FIFO 数据请求） • bit 3–DTO（数据传输完成） • bit 2–CD（命令发送完成） • bit 1–RE（响应错误） • bit 0–CD（卡检测中断）

9.1.6.17. 0x040 SDMC_INTST

默认值：0x00000000			已使能中断状态 (Enabled Interrupt Status)
位域	类型	默认值	描述
31:17	–	–	–
16	R	0x0	SDIO_INT_STAT SDIO 中断状态 (SDIO Interrupt Status) • 0：无来自 SDIO 的中断信号 • 1：有来自 SDIO 的中断信号
15:0	R	0x0	INT_STAT 中断状态 (Interrupt Status) 每个位的中断状态定义如下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			已使能中断状态 (Enabled Interrupt Status)
位域	类型	默认值	描述
			• bit 15–EBE（读操作结束位错误或者写操作无 CRC) • bit 14–ACD（自动发送命令完成） • bit 13–SBE（起始位错误） BCI（忙清除中断） • bit 12–HLE（硬件锁定写操作错误） • bit 11–FRUN（FIFO 向下溢出/向上溢出） • bit 10–HTO（Host 获取数据超时） 或者电压切换中断 • bit 9–DRTO（读数据超时） • bit 8–RTO（响应超时） • bit 7–DCRC（数据 CRC 错误） • bit 6–RCRC（响应 CRC 错误） • bit 5–RXDR（接收 FIFO 数据请求） • bit 4–TXDR（发送 FIFO 数据请求） • bit 3–DTO（数据传输完成） • bit 2–CD（命令发送完成） • bit 1–RE（响应错误） • bit 0–CD（卡检测中断）

9.1.6.18. 0x044 SDMC_OINTST

默认值：0x00000000			初始中断状态 (Original Interrupt Status)
位域	类型	默认值	描述
31:17	–	–	–
16	R/W1C	0x0	SDIO_INT_STAT 初始 SDIO 中断状态 (Original SDIO Interrupt Status) • 0：无来自 SDIO 的中断信号 • 1：有来自 SDIO 的中断信号
15:0	R/W1C	0x0	INT_STAT 初始中断状态 (Original Interrupt Status) 每个位的中断状态定义如下： • bit 15–EBE（读操作结束位错误或者写操作无 CRC） • bit 14–ACD（自动发送命令完成）

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			初始中断状态 (Original Interrupt Status)
位域	类型	默认值	描述
			• bit 13–SBE（起始位错误） BCI（忙清除中断） • bit 12–HLE（硬件锁定写操作错误） • bit 11–FRUN（FIFO 向下溢出/向上溢出） • bit 10–HTO（Host 获取数据超时） 或者电压切换中断 • bit 9–DRTO（读数据超时） • bit 8–RTO（响应超时） • bit 7–DCRC（数据 CRC 错误） • bit 6–RCRC（响应 CRC 错误） • bit 5–RXDR（接收 FIFO 数据请求） • bit 4–TXDR（发送 FIFO 数据请求） • bit 3–DTO（数据传输完成） • bit 2–CD（命令发送完成） • bit 1–RE（响应错误） • bit 0–CDET（卡检测中断）

9.1.6.19. 0x048 SDMC_FIFOCFG

默认值：0x007F0000			FIFO 配置 (FIFO Configuration)
位域	类型	默认值	描述
31	–	–	–
30:28	R/W	0x0	DMA_BL DMA 突发传输长度 (DMA Burst Length) • 0：1 • 1：4 • 2：8 • 3：16 • 4：32 • 5：64 • 6：128 • 7：256 突发传输单位为：word = 32 bits。 突发传输长度应设置为 (RX_WMARK + 1) 和 (FIFO_DEPTH – TX_WMARK) 的公约数。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x007F0000			FIFO 配置 (FIFO Configuration)
位域	类型	默认值	描述
			以 FIFIO_DEPTH = 16 为例： • 允许的 DMA_BL 和 TX_WMARK 的组合如下： ◦ DMA_BL = 1，TX_WMARK = 1 – 15 ◦ DMA_BL = 4，TX_WMARK = 8 ◦ DMA_BL = 4，TX_WMARK= 4 ◦ DMA_BL = 4，TX_WMARK = 12 ◦ DMA_BL = 8，TX_WMARK = 8 ◦ DMA_BL = 8，TX_WMARK = 4 • 允许的 DMA_BL 和 RX_WMARK 的组合如下： ◦ DMA_BL = 1，RX_WMARK = 0 – 14 ◦ DMA_BL = 4，RX_WMARK = 3 ◦ DMA_BL = 4，RX_WMARK = 7 ◦ DMA_BL = 4，RX_WMARK = 11 ◦ DMA_BL = 8，RX_WMARK = 7 推荐值：DMA_BL = 8，TX_WMARK = 8，RX_WMARK = 7
26:16	R/W	0x7F	RX_WMARK FIFO 接收门限值 (FIFO RX Watermark) 当 FIFO 的接收的数据大于 FIFO 接收门限值，DMA/FIFO 请求信号拉高。如果中断信号使能，会产生一个中断。 • 在非 DMA 模式下，当 RXDR 中断使能时，会产生一个中断信号替代 DMA 请求。数据包传输结束后，如果门限值大于剩余的字节数大小，则不会产生中断。当产生数据传输完成中断时， Host 应该读完剩余的字节。 • 在 DMA 模式下，在接收到最后一个数据包时，即使剩下的数据小于门限值，DMA 也会发送一个请求读出 FIFO 里面剩下的数据，然后产生一个数据传输完成的中断。 限制条件：RX_WMARK ≤ FIFO_DEPTH – 2 推荐值：(FIFO_DEPTH)/2 – 1
15:12	–	–	–
11:0	R/W	0x0	TX_WMARK FIFO 发送门限值 (FIFO TX Watermark) 当 FIFO 内数据量小于或等于 FIFO 发送门限值，DMA/FIFO 请求信号拉高。如果中断信号使能，会产生一个中断。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x007F0000			FIFO 配置 (FIFO Configuration)
位域	类型	默认值	描述
			• 在非 DMA 模式下，当 TXDR 中断使能时，会产生一个中断信号替代 DMA 请求信号。发送完最后一个数据包时，Host 应该用剩下的数据填充 FIFO。 • 在 DMA 模式下，最后一个数据包完成后，如果最后一笔传输小于突发数据长度，DMA 控制器会再执行一次传输直到所有数据传输完成。 限制条件：TX_WMARK ≥ 1 推荐值：(FIFO_DEPTH)/2

9.1.6.20. 0x054 SDMC_CDET

默认值：0x01FFFFFF			卡检测配置 (Card Detect Configuration)
位域	类型	默认值	描述
31:25	–	–	–
24	R	0x1	CDET_STAT 表示卡检测信号的电平状态，默认为 1。 • 0：表示有卡插入，电平状态值为 0 • 1：表示卡拔出，电平状态值为 1
23:0	R/W	0xFFFFFF	DEBC_CNT 去除抖动维持的时钟周期数 (Debounce Count) 由于卡插拔是机械动作，会产生电平抖动，导致采样的电平值可能不稳定。去抖逻辑根据去抖维持的时钟周期数来设置计数器的初值。当计数器值变为 0 时，卡检测信号稳定，此时可以采样到正确的电平值。 典型的去抖时间为 5 – 25 ms。

9.1.6.21. 0x080 SDMC_PBUSCFG

默认值：0x00000000			外设总线配置 (Peripheral Bus Configuration)
位域	类型	默认值	描述
31:11	–	–	–
10:8	R	0x0	PBL 已编程的突发传输长度 (Programed Burst Length) 表示 IDMAC 执行一次突发传输的最大长度。已编程的突发传输长度值为 FIFOCFG 寄存器中 DMA_BL 位域的镜像值，如需更改，需要写 FIFOCFG 寄存器。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			外设总线配置 (Peripheral Bus Configuration)
位域	类型	默认值	描述
			• 0：1 • 1：4 • 2：8 • 3：16 • 4：32 • 5：64 • 6：128 • 7：256 突发传输的单位：word = 32 bits。
7	R/W	0x0	DE 内部 DMA 使能 (Inside DMA Enable) • 0：内部 DMA 禁用 • 1：内部 DMA 使能
6:2	R/W	0x0	DSL 描述符跳过长度 (Description Skip Length) 定义了两个非链式描述之间跳过的数据长度，单位是 32 位。仅用于双缓冲结构的描述应用中。
1	R/W	0x0	FB 固定突发长度 (Fixed Burst Length) 控制 AHB 主机接口是否执行固定长度的突发传输。 • 0：AHB 使用 SINGLE 和 INCR 方式执行突发传输。 • 1：AHB 固定使用 SINGLE、INCR4、INCR8 或 INC16 其中一种方式执行突发传输。
0	R/WAC	0x0	SWR 软件复位 (Software Reset) 置 1，DMA 控制器复位内部寄存器，经过一个时钟周期后自动清零。

9.1.6.22. 0x084 SDMC_IDMARCAP

默认值：0x00000000			恢复获取描述符 (Inside DMA Resume Capture Descriptor)
位域	类型	默认值	描述
31:0	W	0x0	RCAP 恢复获取描述符 (Resume Capture Descriptor)

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			恢复获取描述符 (Inside DMA Resume Capture Descriptor)
位域	类型	默认值	描述
			该寄存器为只写。 如果描述的 OWN 位没有置 1，IDMAC 的状态机会进入暂停状态。Host 可以对 RCAP 写任意值，使 IDMAC 的状态机恢复到正常的描述符获取操作。

9.1.6.23. 0x088 SDMC_IDMASADDR

默认值：0x00000000			描述符起始地址 (Descriptor List Start Address)
位域	类型	默认值	描述
31:0	R/W	0x0	DLSA 描述符链表的起始地址 (Descriptor List Start Address) IDMAC 会忽略低有效位 bit[1:0]（只读，对应 32 位总线位宽）的数据，并将其作为全零处理。

9.1.6.24. 0x08C SDMC_IDMAST

默认值：0x00000000			内部 DMA 状态 (Inside DMAC Status)
位域	类型	默认值	描述
31:17	–	–	–
16:13	R	0x0	IDMA_FSM DMA 控制器状态机当前的状态 (IDMA FSM Status) • 0：DMA_IDLE • 1：DMA_SUSPEND • 2：DMAC_RD • 3：DMAC_CHK • 4：DMA_RD_REQ_WAIT • 5：DMA_WR_REQ_WATI • 6：DMA_RD • 7：DMA_WR • 8：DESC_CLOSE
12:10	R	0x0	FBE_CODE 致命的总线错误 (Fatal Bus Error Code) 表示造成总线出错的错误类型，仅在 IDMAST[2] 置 1 时有效。该位域不产生中断。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			内部 DMA 状态 (Inside DMAC Status)
位域	类型	默认值	描述
			• 1：在发送数据时 Host 中止 • 2：在接收数据时 Host 中止
9	R/W1C	0x0	ABNORM_INT 异常中断 (Abnormal Interrupt) 以下任意一个中断产生时，触发异常中断： • IDMAST[2]：致命的总线中断 • IDMAST[4]：描述符不可用中断 只有未屏蔽的位域才影响异常中断位的值。该位写 1 清零。
8	R/W1C	0x0	NORM_INT 正常中断 (Normal Interrupt) 以下任意一个中断产生时，触发正常中断： • IDMAST[0]：发送中断 • IDMAST[1]：接收中断 只有未屏蔽的位域才影响正常中断位的值。该位写 1 清零。
7:6	–	–	–
5	R/W1C	0x0	CARD_ERR_INT 卡错误中断 (Card Error Interrupt) 表示卡操作的状态信息，也存在于 OINTSTS 寄存器，以下任意中断产生，触发卡错误中断： • EBE–结束位错误 • RTO–响应超时/ Boot 应答超时 • RCRC–响应 CRC • SBE–起始位错误 • DRTO–读数据超时 • DCRC–接收的数据 CRC 错误 • RE–响应错误 该位写 1 清零。
4	R/W1C	0x0	DES_UNU_INT 描述符不可用中断 (Descriptor Unused Interrupt) 该位置 1 表示当前描述符不可用（因为 OWN 位为 0）。 该位写 1 清零。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			内部 DMA 状态 (Inside DMAC Status)
位域	类型	默认值	描述
3	–	–	–
2	R/W1C	0x0	FBE_INT 致命的总线错误中断 (Fatal Bus Error Interrupt) 表示有一个总线错误发生，当该位置 1，DMA 禁止所有的总线访问。 该位写 1 清零。
1	R/W1C	0x0	RX_DONE_INT 接收完成中断 (RX Done Interrupt) 表示一个描述符的数据接收完成。 该位写 1 清零。
0	R/W1C	0x0	TX_DONE_INT 发送完成中断 (TX Done Interrupt) 表示一个描述符的数据发送完成。 该位写 1 清零。

9.1.6.25. 0x090 SDMC_IDMAINTEN

默认值：0x00000000			内部 DMA 中断使能 (Inside DMAC Interrupt Enable)
位域	类型	默认值	描述
31:10	–	–	–
9	R/W	0x0	ABNORM_INTEN 异常中断使能 (Abnormal Interrupt Enable) 置 1 时，使能异常中断，包括以下中断： • IDMAINTEN[2]–致命的总线中断 • IDMAINTEN[4]–描述符不可用中断
8	R/W	0x0	NORM_INTEN 正常中断使能 (Normal Interrupt Enable) 置 1 时，使能正常中断，包括以下中断： • IDMAINTEN[0]–发送完成中断 • IDMAINTEN[1]–接收完成中断
7:6	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			内部 DMA 中断使能 (Inside DMAC Interrupt Enable)
位域	类型	默认值	描述
5	R/W	0x0	CARD_ERR_INTEN 卡错误中断使能 (Card Error Interrupt Enable) 置 1 时，使能卡错误中断。
4	R/W	0x0	DES_UNU_INTEN 描述符不可用中断 (Descriptor Unused Interrupt Enable) 当 ABNORM_INTEN 位域使能时，该位也一起使能。
3	–	–	–
2	R/W	0x0	FBE_INTEN 致命的总线错误中断使能 (Fatal Bus Error Interrupt Enable)
1	R/W	0x0	RX_DONE_INTEN 接收完成中断使能 (RX Done Interrupt Enable)
0	R/W	0x0	TX_DONE_INTEN 发送完成中断使能 (TX Done Interrupt Enable)

9.1.6.26. 0x094 SDMC_IDMACDA

默认值：0x00000000			当前描述符地址指针 (Current Descriptor Address)
位域	类型	默认值	描述
31:0	R	0x0	CDA 当前描述符地址指针 (Current Descriptor Address) 该寄存器指向当前描述符的起始地址，复位时清零。IDMAC 操作时指针会自动刷新。

9.1.6.27. 0x098 SDMC_IDMACBA

默认值：0x00000000			当前缓存地址指针 (Current Buffer Address)
位域	类型	默认值	描述
31:0	R	0x0	CBA 当前缓存地址指针 (Current Buffer Address) 该寄存器指向当前数据缓存的地址，复位时清零。IDMAC 操作时指针会自动刷新。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.1.6.28. 0x100 SDMC_CTC

默认值：0x00000000			卡门限值配置 (Card Threshold Control)
位域	类型	默认值	描述
31:26	–	–	–
25:16	R/W	0x0	CTH_SIZE 卡传输门限值大小 (Card Threshold Size) **注：** 仅当 CDWTH_EN 置 1 或 CDRTH_EN 置 1 时，该寄存器有效。
15:3	–	–	–
2	R/W	0x0	CDWTH_EN 卡写操作门限值使能 (Card Write Threshold Enable) • 0：卡写操作门限禁止 • 1：卡写操作门限值使能 仅当发送 FIFO 达到门限值的数据量时，SDMC 控制器初始化写传输。
1	R/W	0x0	BC_INTEN 忙状态清除中断使能 (Busy Clear Interrupt Enable) • 0：忙状态清除中断禁止 • 1：忙状态清除中断使能
0	R/W	0x0	CDRTH_EN 卡读操作门限值使能 (Card Read Threshold Enable) • 0：卡读操作门限值禁止 • 1：卡读操作门限值使能 仅当接收 FIFO 达到门限值的数据量时读数据。

9.1.6.29. 0x104 SDMC_DLYCTRL

默认值：0x00000000			延时链和相位控制 (Delay Chain and Phase Control)
位域	类型	默认值	描述
31:30	R/W	0x0	EXT_CLK_MUX 外部输入时钟选择 (External Clock Multiplex)

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			延时链和相位控制 (Delay Chain and Phase Control)
位域	**类型**	**默认值**	**描述**
			• 0：输入时钟 4 分频，典型值为 50 MHz（DS，HS–SDR，HS–DDR 模式） • 1：输入时钟 2 分频，典型值为 100 MHz（HS，HS–SDR，HS–DDR 模式） • 2：输入时钟 1 分频，典型值为 200 MHz（HS–DDR 模式）
29:28	R/W	0x0	CLK_DRV_PHASE 驱动时钟相位控制 (Driver Clock Phase) 配置输出驱动时钟 cclk_in_drv 使用的相位值 • 0：0° • 1：90° • 2：180° • 3：270°
27:23	R/W	0x0	CLK_DRV_DELAY 驱动时钟延迟控制 (Driver Clock Delay) 配置输出驱动时钟 cclk_in_drv 使用的延时值
22:21	R/W	0x0	CLK_SMPL_PHASE 采样时钟相位控制 (Sample Clock Phase) 配置采样时钟 cclk_in_sample 使用的相位值 • 0：0° • 1：90° • 2：180° • 3：270°
20:16	R/W	0x0	CLK_SMPL_DELAY 采样时钟延迟控制 (Sample Clock Delay) 配置采样时钟 cclk_in_sample 使用的延时值。
15:14	–	–	–
13:9	–	–	–
8	R/W	0x0	SAMP_SWAP 采样边沿交换 (DDR Sample Edge Swap) 默认先采样奇数边沿数据。
7	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			延时链和相位控制 (Delay Chain and Phase Control)
位域	类型	默认值	描述
6	R/W	0x0	SAMP_SEQ_EN DDR 模式采样顺序交换使能 (DDR Sample Edge Swap Enable)
5:4	R/W	0x0	EVEN_SHIFT 奇数据时钟移相选择 (Even Data Clock Phase) • 0：选择默认相位 • 1：相位移动 90° • 2：相位移动 180° • 3：相位移动 270°
3:2	–	–	–
1:0	R/W	0x1	SHIFT_EN 时钟相移使能 (Clock Phase Shift Enable) • 0：选择默认相位 • 1：相位移动 90° • 2：相位移动 180° • 3：相位移动 270°

9.1.6.30. 0x108 SDMC_EMCR

默认值：0x00000000			EMMC 配置 (EMMC Configuration)
位域	类型	默认值	描述
31:1	–	–	–
0	R/W	0x0	HALF_START_BIT 起始位保持 对于 eMMC 4.5 版本，起始位可以保持： • 0：完整的时钟周期 • 1：小于一个完整的时钟周期 对于 eMMC 4.5 及以上版本，该位置 1。对于 SD 卡应用，该位清零。 **注：** DDR 模式时该位域会被忽略。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.1.6.31. 0x118 SDMC_VERID

默认值：0x00000100			控制器版本号 (Version ID)
位域	类型	默认值	描述
31:0	R	0x00000100	VERID 控制器版本号 (Version ID) 采用 BCD 码显示，V1.0

9.1.6.32. 0x200 SDMC_FIFO

默认值：0x00000000			数据 FIFO 缓存寄存器 (Data FIFO)
位域	类型	默认值	描述
31:0	R/W	0x0000_0000	DATA_FIFO 读写数据缓存 FIFO 寄存器，该寄存器映射到 FIFO 的入口地址。 • 对于写操作，CPU 或 IDMA 通过 AHB 总线写数据到该地址。 • 对于读操作，CPU 或 IDMA 通过 AHB 总线从该地址读数据。 **注：** 读写数据共享相同的 FIFO，不支持同时对该地址进行读或写访问。

9.2. eXpanded Serial Peripheral Interface (xSPI)

xSPI 模块是一个高度灵活和可配置的接口，能够支持多种不同的串行和并行总线协议，包括外部并行总线 Xccela、HyperBus、OPI、QPI、和1/ 2/ 4/ 8 线 SPI。xSPI 模块可与外部 PSRAM、Flash 存储设备等元器件实现连接，无需额外的硬件接口或适配器。

9.2.1. 功能特性

- 支持 Xccela 协议
- 支持 HyperBus 协议
- 支持 OPI 协议
- 支持 QPI 协议
- 支持 1/ 2/ 4/ 8 线 SPI
- 支持 200 MHz DDR/ SDR 采样
- 支持 XIP (eXecute In Place)

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 支持中断以及 DMA
- 支持两片并行读写数据
- 支持单片 16 bits 位宽读写数据
- 具有 16x32 bits TXFIFO 以及 16x32 bits RXFIFO
- 具有 512 MB 内存映射空间

9.2.2. 原理框图

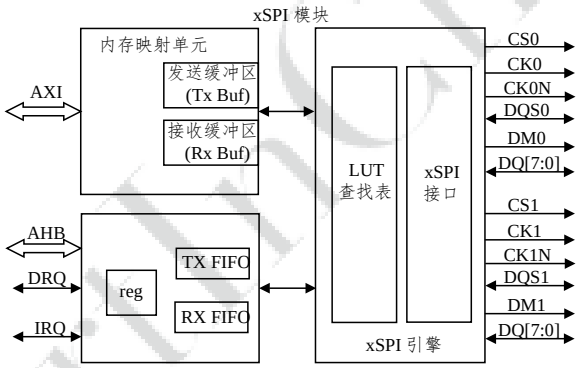


图 9–1 xspi_block

9.2.3. 功能描述

9.2.3.1. xSPI 支持的协议模式

xSPI 模块约定了以下缩写描述：

- S：SDR 模式
- D：DDR 模式
- 数字 1 / 2 / 4/ 8：输出 IO 的数量，即分别输出 1 / 2 / 4 / 8 个 IO。

举例说明如下：

xSPI 设备协议均可以看成三个部分组成：CMD + ADD + DATA。CMD 是命令部分，ADD 是地址部分，DATA 是数据部分。

1. 8D–8D–8D：CMD–ADD–DATA 都以 8 个 IO 输入输出且为 DDR 模式。
2. 1S–1S–8D：CMD–ADD 都以 1 个 IO 输出且为 SDR 模式，DATA 则 8 个 IO 输入输出且为 DDR 模式。

为了灵活配置各协议，在 CMD–ADD–DATA 单个完整传输中，各部分结构都可为 SDR/ DDR 以及 IO 独立配置，协议配置数据以查找表的方式存储，详细见查找表 (LUT) 功能。

xSPI 支持的典型协议行为包括：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- Xccela 协议
- Hyperbus 协议
- OPI 协议
- QPI 协议
- Octal SPI 协议
- Quad SPI 协议
- 双线 SPI 协议
- 单线 SPI 协议

9.2.3.1.1. Xccela 协议

图 9–2 : Xccela 读操作以 8D–8D–8D 为例进行说明，Xccela 协议的工作模式如下所示。

1. Host 发起 CMD–ADD。

> **注：**
>
> 下列 CMD 占两个采样沿，ADD 占一个采样沿，之后接着可配的 Dummy Cycles。

2. Dummy 期间 Memory Device 获得 DQS 线权，在约定的 Dummy Cycles 后，接着的 CK 触发 Memory Device 读数据输出。此时 Host 以 DQS 为采样时钟。

> **注：**
>
> CK 与 DQS 存在一定的延时，DQS 与 DATA 边沿对齐。

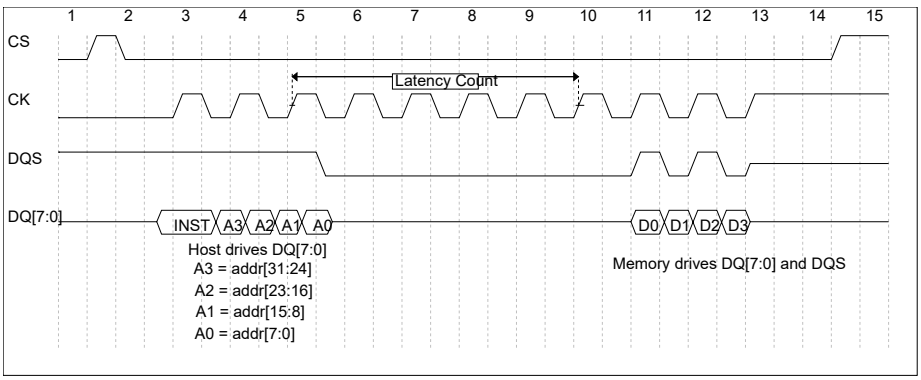


图 9–2 Xccela 读操作

在 Memory Device 进行跨行边界读取数据时，DQS 会在行尾数据保持状态，如Xccela 协议所示。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

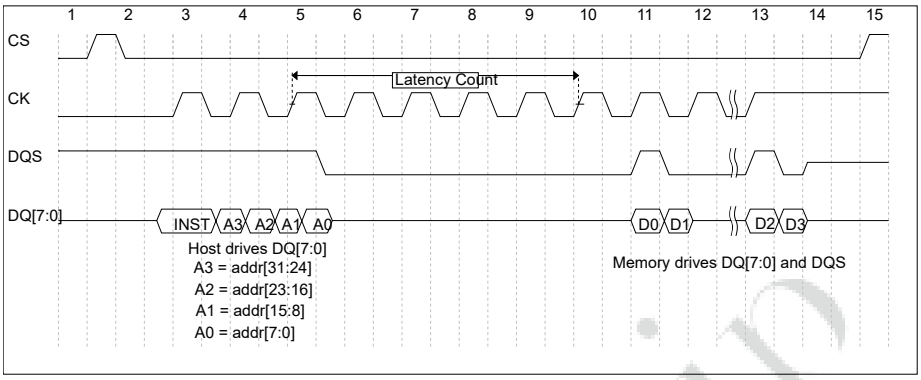


图 9–3 Xccela 跨行边界读操作

图 9–4 : Xccela 写操作以 8D–8D–8D 为例。CMD–ADD 以及 Dummy Cycles 与读操作一致，DQS 由 Host 占据线权，作为 Data Mask。

注：

CK 与 DATA 中心对齐，DQS 与 DATA 边沿对齐。其中 D0、D3 以及 D4 被屏蔽，不作为有效数据写进 Memory Device。

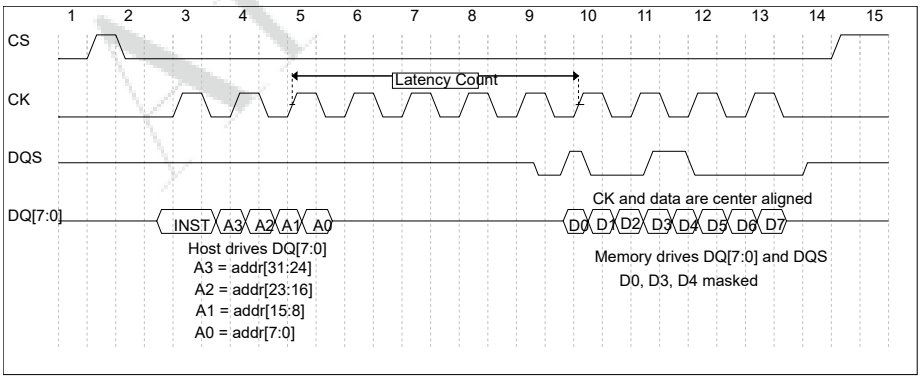


图 9–4 Xccela 写操作

9.2.3.1.2. Hyperbus 协议

图 9–5 : Hyperbus 读操作以 8D–8D–8D 为例进行说明。

1. Hyperbus 中 CA 存在两个 bytes 的 CMD，第二个 byte 的 CMD 定义为 Command–Extension (CMD–Ex)，用于填充高位地址，具体实现见 Hyperbus 行为配置。
2. 在 Host 发起 CA 期间，Host 检测 RWDS 的电平状态。
   a. 若 RWDS 电平为低，则等待配置好的 Dummy Cycles。
   b. 若 RWDS 电平为高，则等待 2 倍的已配置 Dummy Cycle 数，接下来的 CK 触发 Memory Device 读数据输出，此时 Host 以 RWDS 为采样时钟。

注：

CK 与 RWDS 存在一定的延时，RWDS 与 DATA 边沿对齐，也会存在跨行边界读取数据 RWDS 等待过程。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

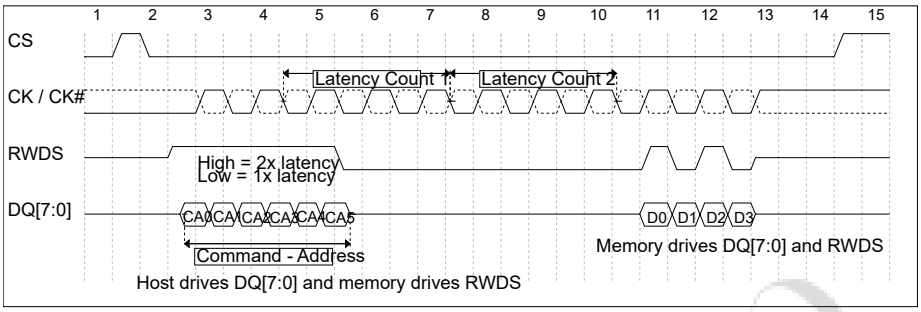


图 9–5 Hyperbus 读操作

Hyperbus 写操作以 8D–8D–8D 为例进行说明。

1. CA 与读操作一致，在 Dummy Cycles 期间占据 RWDS 线权。

> **注：**
>
> 写操作中 CK 与 DATA 中心对齐，RWDS 与 DATA 边沿对齐。

2. 其中 RWDS 也作为 Data Mask，图示 D0，D3 被屏蔽，不作为有效数据写进 Memory Device。
3. Host 发起 CA（Command–Address），其中 CA 各 Bit 的定义如Command–Address所示。

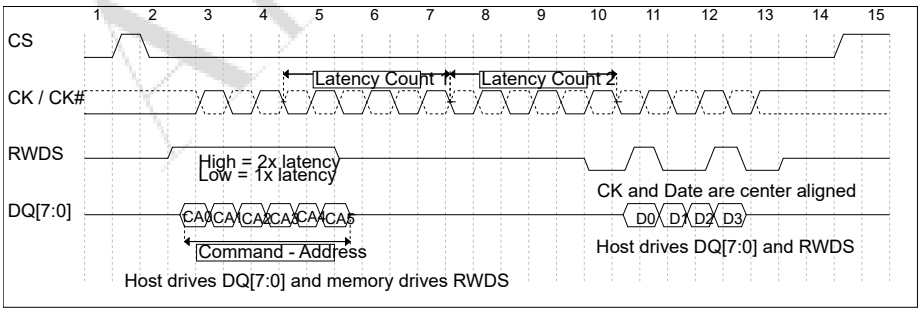


图 9–6 Hyperbus 写操作

表 9–4 Command–Address

CA bit	Defined	Function
47	R/W	指示该命令为读或者写操作 • 0：指示为写操作 • 1：指示为读操作
46	Address space	指示该命令为内存或者寄存器访问 • 0：指示为内存访问 • 1：指示为寄存器访问
45	Burst type	指示该命令的突发类型

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 9–4 Command–Address (续)

CA bit	Defined	Function
		• 0：wrapped 突发 • 1：linear 突发
44–16	Row and upper column address	32 位地址系统中的 A31–A3
15–3	reserved	预留未来行地址扩展，补 0
2–0	Lower column address	32 位地址系统中的 A2–A0

9.2.3.1.3. OPI 协议

图 9–7 : OPI 读操作以 8D–8D–8D 为例进行说明。

1. Host 发起 CMD–ADD，之后接着可配的 Dummy Cycles，Dummy 期间 Memory Device 获得 DQS 线权。
2. 在约定的 Dummy Cycles 后，接着的 CK 触发 Memory Device 读数据输出，此时 Host 以 DQS 为采样时钟。

> **注：**
>
> CK 与 DQS 存在一定的延时，DQS 与 DATA 边沿对齐。

另外，在 Memory Device 进行跨行边界读取数据时，也会存在跨行边界读取数据 DQS 等待过程。

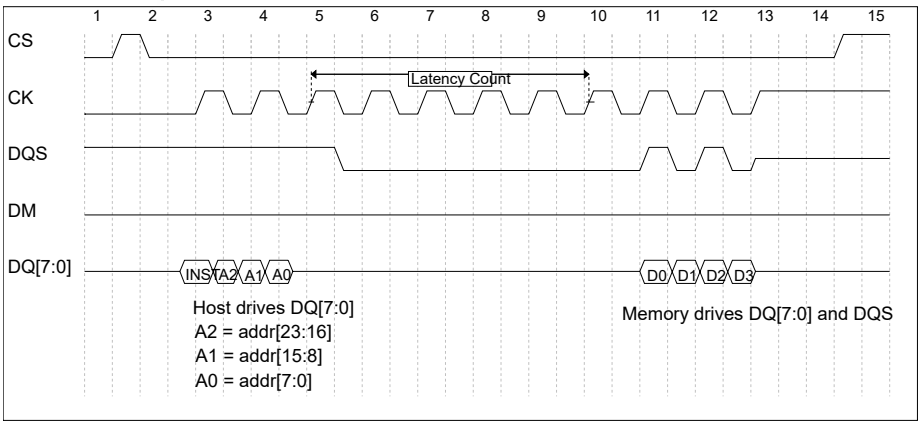


图 9–7 OPI 读操作

图 9–8 : OPI 写操作 图中以 8D–8D–8D 为例进行说明。

1. CMD–ADD 以及 Dummy Cycles 与读操作一致，DQS 由 Host 占据线权，作为采样时钟。DM 作为 Data Mask。

> **注：**
>
> DQS 与 DATA 中心对齐，DM 与 DATA 边沿对齐。

2. 写操作中，D3 以及 D5 被屏蔽，不作为有效数据写进 Memory Device。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

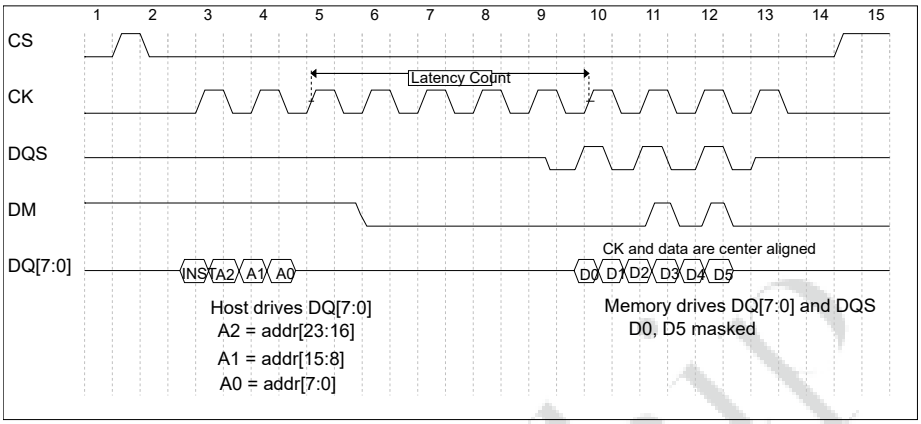


图 9–8 OPI 写操作

9.2.3.1.4. QPI 协议

图 9–9 : QPI 读操作 图中以 8D–8D–8D 为例进行说明。

1. Host 发起 CMD–ADD，之后接着可配的 Dummy Cycles，Dummy 期间 Memory Device 获得 DQS 线权。
2. 在约定的 Dummy Cycles 后，接着的 CK 触发 Memory Device 读数据输出，此时 Host 以 DQS 为 采样时钟。

> **注：**
>
> CK 与 DQS 存在一定的延时，DQS 与 DATA 边沿对齐。

另外，在 Memory Device 进行跨行边界读取数据时，也会存在跨行边界读取数据 DQS 等待过程。

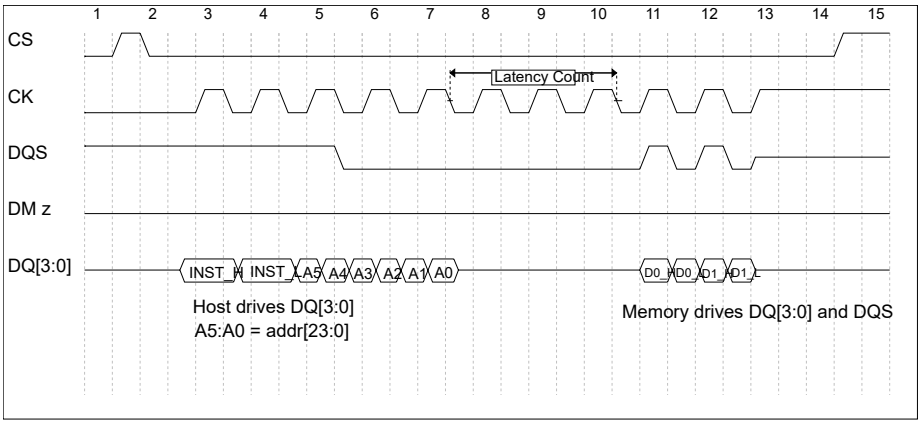


图 9–9 QPI 读操作

图 9–10 : QPI 写操作图中以 8D–8D–8D 为例进行说明。

1. CMD–ADD 以及 Dummy Cycles 与读操作一致，DQS 由 Host 占据线权，作为采样时钟。DM 作为 Data Mask。

> **注：**
>
> DQS 与 DATA 中心对齐，DM 与 DATA 边沿对齐

2. 写操作中，D0 被屏蔽，不作为有效数据写进 Memory Device。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

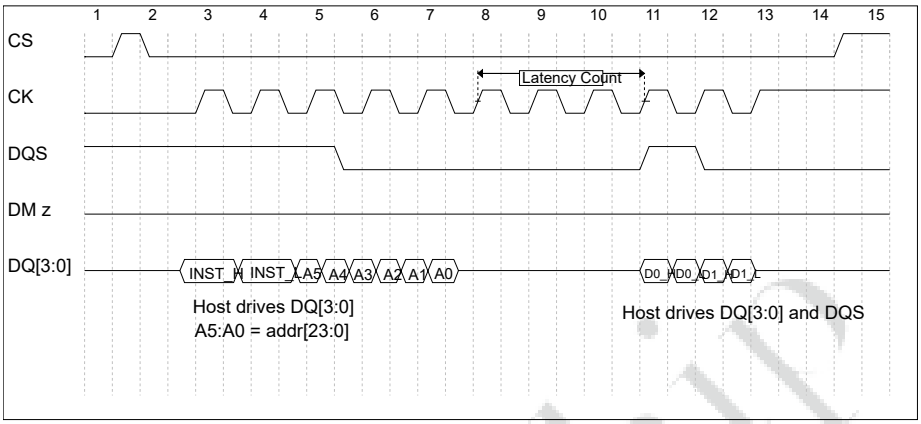


图 9–10 QPI 写操作

9.2.3.1.5. Octal SPI 协议

图 9–11 : Octal SPI 读写操作图中以 1S–1S–8S 为例进行说明。

DQ[0] 输出 CMD–ADDR，等待约定的 Dummy Cycles 后，DQ[7:0] 进行数据输入输出传输。Octal SPI 协议也存在 1S–8S–8S，1D–1D–8D 等格式，此部分不列举图示。

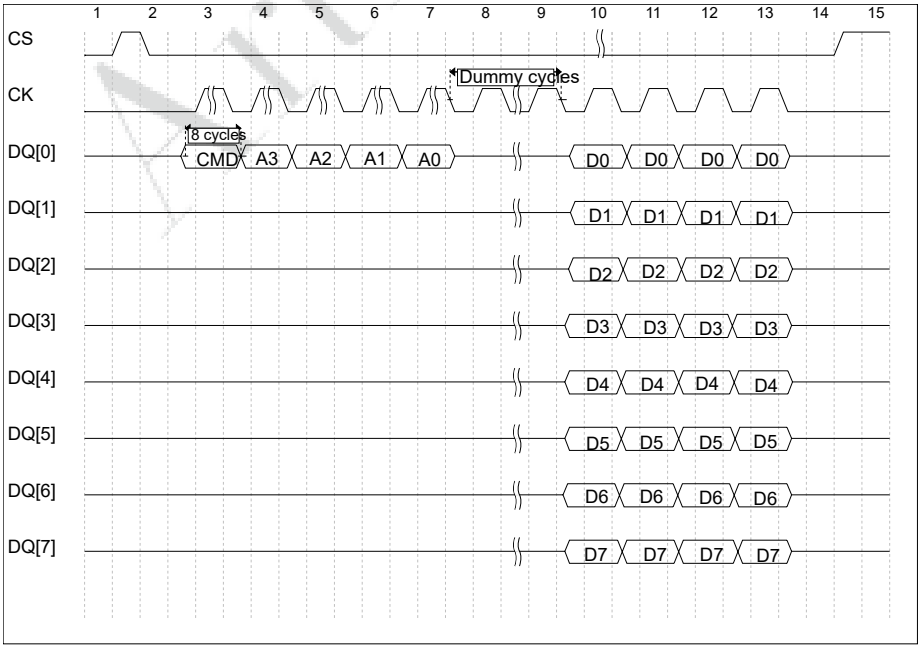


图 9–11 Octal SPI 读写操作

9.2.3.1.6. Quad SPI 协议

图 9–12 : Quad SPI 读写操作以 1S–1S–4S 为例进行说明。

DQ[0] 输出 CMD–ADDR，等待约定的 Dummy Cycles 后，DQ[3:0] 进行数据输入输出传输。Quad SPI 协议也存在 1S–4S–4S，1D–1D–4D 等格式，此部分不列举图示。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

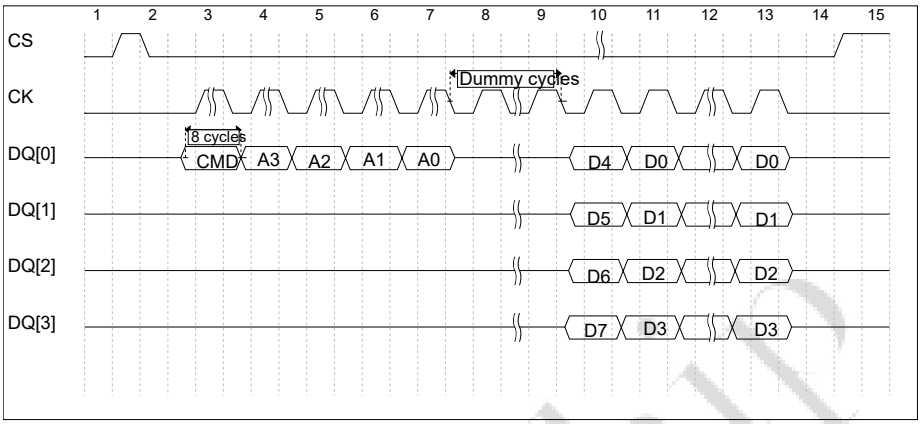


图 9–12 Quad SPI 读写操作

9.2.3.1.7. 双线 SPI 协议

图 9–13 : Dual SPI 读写操作以 1S–1S–2S 为例进行说明。

DQ[0] 输出 CMD–ADDR，等待约定的 Dummy Cycles 后，DQ[1:0] 进行数据输入输出传输。双线 SPI 协议也存在 1S–2S–2S，1D–2D–2D 等格式，此部分不列举图示。

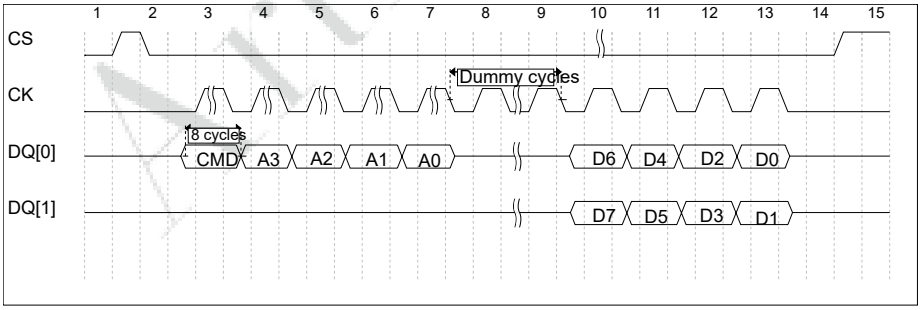


图 9–13 Dual SPI 读写操作

9.2.3.1.8. 单线 SPI 协议

图 9–14 : Single SPI 读写操作图中以 1S–1S–2S 为例进行说明。

DQ[0] 输出 CMD–ADDR，等待约定的 Dummy Cycles 后，DQ[1] 进行数据输入传输。

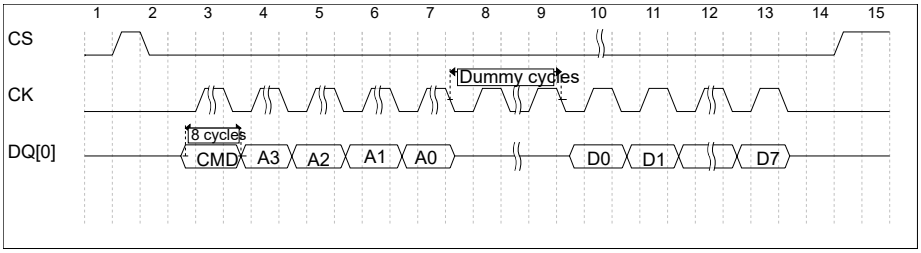


图 9–14 Single SPI 读写操作

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.2.3.2. 查找表 (LUT) 功能

xSPI 支持查找表功能，每个 command 有 6 bits 控制执行的操作类型，2 bits 定义输出 IO 为 1/ 2/ 4/ 8 线，8 bits 定义操作数。

表 9–5 指令

查找表指令 (6 bits)	定义
CMD (0x1)	指示为 SDR Command
CMD_EX (0x2)	指示为 SDR Command Extension
ADDR (0x3)	指示为 SDR Address
WRITE (0x4)	指示为 SDR 写操作
READ (0x5)	指示为 SDR 读操作
CMD_DDR (0x11)	指示为 DDR Command
CMD_EX_DDR (0x12)	指示为 DDR Command Extension
ADDR_DDR (0x13)	指示为 DDR Address
WRITE_DDR (0x14)	指示为 DDR 写操作
READ_DDR (0x15)	指示为 DDR 读操作
DUMMY (0x10)	指示为 Dummy Cycles
JUMP_ID (0x20)	跳转 ID 号命令（AXI 通道使用）
JUMP_INS (0x30)	跳转指令命令（同个 ID 内跳转指令，AXI 通道使用）
STOP (0x0)	结束命令

以下为 ID 生成和触发规则：

注：

为了供访问外部不同 device 使用，比如 Flash 与 PSRAM，在 AXI 访问中：

- 若为写访问，从 0 号 ID 开始执行
- 若为读访问，从 4 号 ID 开始执行

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 寄存器 LUTn (n=0~31)，每个寄存器配置两个 commands，可存放 64 个 commands。一个完整的 xSPI 读写操作通常需要使用多个 commands 设置组成，因此四个 LUT 寄存器划为一组并赋予一个 ID 号，即 LUT[0~3] 的 ID 号为 0，LUT[4~7] 的 ID 号为 1，以此类推共有八个 ID 号，如下表所示。

表 9–6 寄存器 LUTn(n=0~31) 分组及其 ID 映射表

LUTn(n=0~31)	ID
LUT[0~3]	0
LUT[4~7]	1
LUT[8~11]	2
LUT[12~15]	3
LUT[16~19]	4
LUT[20~23]	5
LUT[24~27]	6
LUT[28~31]	7

- AHB 通道读写操作由写入寄存器 xSPI_START 的 ID 触发。
- AXI 通道读写 Memory Mapping 即可触发，

关于各类协议的 LUT 配置可查看下列对应的章节：

- Xccela LUT 配置
- Hyperbus LUT 配置
- OPI LUT 配置
- SPI LUT 配置

9.2.3.2.1. Xccela LUT 配置

举例 Xccela 行为配置如下 （ xSPI_CTL[5:4] 配置为 Xccela 模式 ）：

表 9–7 Xccela 行为配置

指令 （ 6 bits ）	IO 数 (2 bits)	操作数 （ 8 bits ）	说明
CMD_DDR	0x3	0x2	八个 IO 输出 8 bits CMD，并且 CMD 为 0x2，DDR 模式。 **注：** 由于配置了 Xccela 模式，CMD 会占两个采样沿。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 9–7 Xccela 行为配置 (续)

指令（6 bits）	IO 数 (2 bits)	操作数（8 bits）	说明
			• 若为 AXI 通道访问，读写信息 CMD bit[7]、Burst 类型信息 CMD bit[5] 以及地址信息由 AXI 决定，此处只保留 CMD bit[6] 的访问空间（内存/ 寄存器）信息。 • 若为 AHB 通道访问，则此处 0x2 即为 CMD。
ADDR_DRR	0x3	0x18	8 个 IO 输出 ADDR，并且 ADDR 为 24 bits，DDR 模式。 • 若是 AXI 通道访问，则此地址由 AXI 访问 Memory Mapping 映射生成。 • 若是 AHB 通道访问，则由用户配置 xSPI_ADDR 生成。
DUMMY	0x0	0x4	等待四个 xSPI_CLK Cycles
WRITE_DDR	0x3	0x9	8 个 IO 输出 10–byte 数据，DDR 模式
STOP	0x0	0x0	结束此写命令。

注：

其中 ADDR/ ADDR_DDR 的操作数只支持配置为 0x18/ 0x20，即 24 位/ 32 位地址访问，其它配置会出现 Error 状态。READ/ READ_DDR/ WRITE/ WRITE_DDR 的操作数配置为 N，则实际数据长度为 N+1。

1. 图 9–15 : Xccela 执行查找表，CMD 为 0x2，DDR 模式八个 IO 输出。ADDR 为 24 bits 寻址，DDR 模式八个 IO 输出。
   - 如果为 AHB 通道访问，则 ADDR[23:0] 由 XSPI_ADDR 寄存器决定。
   - 如果为 AXI 通道访问，则 ADDR[23:0] 由访问的 Memory Mapping 决定。
2. 接着等待四个 DUMMY Cycles。
   - 如果为 AHB 访问，则写 10 bytes DATA。
   - 如果为 AXI 访问，则由 AXI 决定写数据长度。
3. DDR 模式八个 IO 输出。
4. 执行 STOP 命令结束传输。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

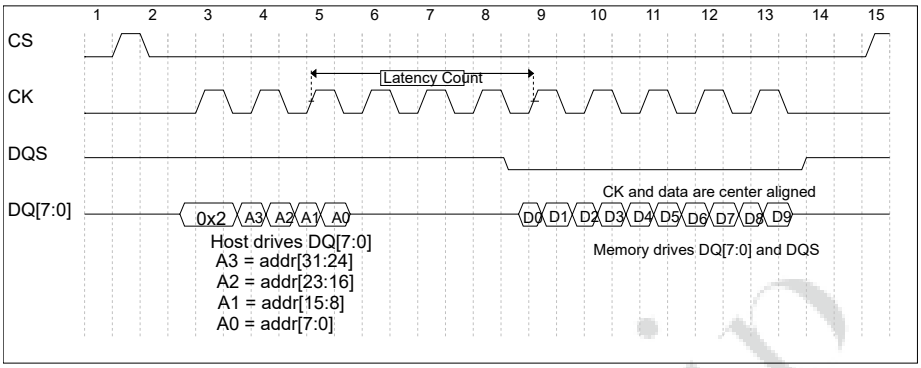


图 9–15 Xccela 执行查找表

9.2.3.2.2. Hyperbus LUT 配置

举例 Hyperbus 行为配置如下（xSPI_CTL[5:4] 配置为 Hyperbus 模式）

表 9–8 Hyperbus 行为配置

指令 (6 bits)	IO 数 (2 bits)	操作数 (8 bits)	说明
CMD_DDR	0x3	0x40	8 个 IO 输出 8–bits CMD 以及 CMD_EX，DDR 模式。 **注：** 由于配置了 Hyperbus 模式： • 若为 AXI 通道访问，读写信息、Burst 类型信息以及地址信息由 AXI 决定，此处只保留 CA bit[46] 的访问空间（内存/寄存器）信息。 • 若为 AHB 通道访问，则此处 0x4003 即为 CA bit[47:32]
CMD_EX_DDR	0x3	0x03	见上描述
ADDR_DDR	0x3	0x20	8 个 IO 输出 ADDR，并且 ADDR 为 32 bits，DDR 模式，如果配置为 24 bits 模式，则高 8 位自动补 0。
DUMMY	0x0	0x3	等待 3 或者 6 个 xSPI_CLK Cycles，注意由于配置了 Hyperbus 模式，在 CA 期间检测的 RWDS 电平决定此处为 3 或者 6 个 cycles。
READ_DDR	0x3	0xff	8 个 IO 输入 256 bytes 数据，DDR 模式
STOP	0x0	0x0	结束此写命令。

注：

其中 ADDR/ ADDR_DDR 的操作数只支持配置为 0x18/ 0x20，即 24 位/ 32 位地址访问，其它配置会出现 Error 状态。READ/ READ_DDR/ WRITE/ WRITE_DDR 的操作数配置为 N，则实际数据长度为 N+1。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

1. 执行 图 9–16 : Hyperbus 执行查找表，CMD 为配置为 0x40，DDR 模式 8 个 IO 输出，为图示 CA0。
   - 若为 AHB 访问，则 CA0 的值即为 0x40。
   - 若为 AXI 访问，则 CA0 的值只保留 bit[6]，此位代表寄存器或者内存访问信息，由用户决定。
2. CA0 的 bit[7] 与 bit[5] 代表的读写信息，Burst Type 信息则由 AXI 决定，即此时 0x40 有可能会被 AXI 改写。
3. CMD_EX 为配置为 0x03，DDR 模式 8 个 IO 输出，为图示 CA1
   - 若为 AHB 访问，则 CA1 的值即为 0x03。
   - 若为 AXI 访问，则由 AXI 访问的映射地址改写。
4. ADDR 为 32 bits 寻址，DDR 模式 8 个 IO 输出。
   - 如果为 AHB 通道访问，CA2/ CA3/ CA4/ CA5 由 xSPI_ADDR 寄存器决定。
   - 如果为 AXI 通道访问，CA2/ CA3/ CA4/ CA5 由 AXI 访问的 Memory Mapping 决定。
5. 接着等待 3 个或 6 个 DUMMY Cycles，查找表配置为 3 个，图示 CA 期间 RWDS 电平检测为高，因此等待 2 x 3 个 DUMMY Cycles。
   - 如果为 AHB 通道访问，则接着读 256 bytes DATA。
   - 如果为 AXI 通道访问，则由 AXI 决定读数据长度，DDR 模式 8 个 IO 输入。
6. 最后执行 STOP 命令结束传输。

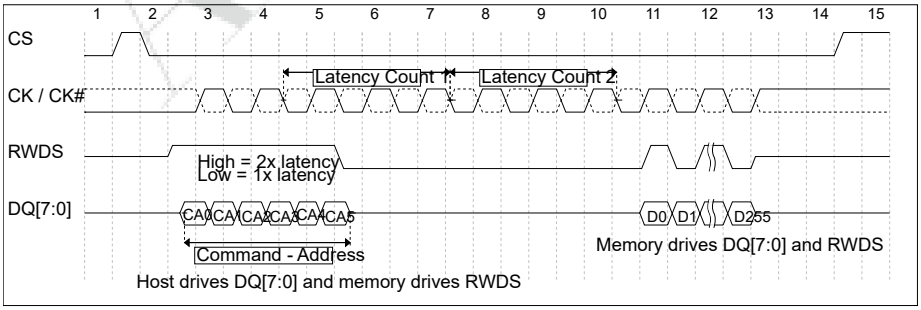


图 9–16 Hyperbus 执行查找表

注：

Hyperbus CA 的特殊性，若为 AXI 通道读写行为，则读写以及 Burst 类型信息由 AXI 决定，地址由 Memory Mapping 映射决定，此时 LUT 中配置的 CMD 以及 CMD_EX 只需要配置内存或者寄存器访问信息（bit46）。若为 AHB 通道读写，在 Hyperbus CA 的 bit[47:32] 由 CMD 以及 CMD_EX 决定，bit[31:0] 由寄存器 xSPI_ADDR 决定。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.2.3.2.3. OPI LUT 配置

举例 OPI 行为配置如下（xSPI_CTL[5:4] 配置为 OPI 模式）

表 9–9 OPI 行为配置

指令 (6 bits)	IO 数 (2 bits)	操作数（8 bits）	说明
CMD_DDR	0x3	0x2	8 个 IO 输出 8 bits CMD，并且 CMD 为 0x2，DDR 模式。注意此时由于配置了 OPI 模式，若为 AXI 通道访问，读写信息 CMD bit[7]、Burst 类型信息 CMD bit[5] 以及地址信息由 AXI 决定，此处只保留 CMD bit[6] 的访问空间（memory or register access）信息。若为 AHB 通道访问，则此处 0x2 即为 CMD
ADDR_DDR	0x3	0x18	8 个 IO 输出 ADDR，并且 ADDR 为 24 bits，DDR 模式。若是 AXI 通道访问，则此地址由 AXI 访问 Memory Mapping 映射生成。若是 AHB 通道访问，则由用户配置 xSPI_ADDR 生成
DUMMY	0x0	0x4	等待 4 个 xSPI_CLK Cycles
WRITE_DDR	0x3	0x9	8 个 IO 输出 10 bytes 数据，DDR 模式
STOP	0x0	0x0	结束此写命令

注：

其中 ADDR/ ADDR_DDR 的操作数只支持配置为 0x18/ 0x20，即 24 位/ 32 位地址访问，其它配置会出现 Error 状态。READ/ READ_DDR/ WRITE/ WRITE_DDR 的操作数配置为 N，则实际数据长度为 N+1。

1. 执行 图 9–17 : OPI 执行查找表图示，CMD 为 0x2，DDR 模式 8 个 IO 输出。
2. ADDR 为 24 bits 寻址，DDR 模式 8 个 IO 输出。
   - 如果为 AHB 通道访问，ADDR[23:0] 由 xSPI_ADDR 寄存器决定。
   - 如果为 AXI 通道访问，ADDR[23:0] 由访问的 Memory Mapping 决定。
3. 接着 4 个 DUMMY Cycles.
   - 如果为 AHB 访问则写 10 bytes DATA。
   - 如果为 AXI 访问则由 AXI 决定写数据长度。
4. DDR 模式 8 个 IO 输出。
5. 最后 STOP 命令结束传输。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

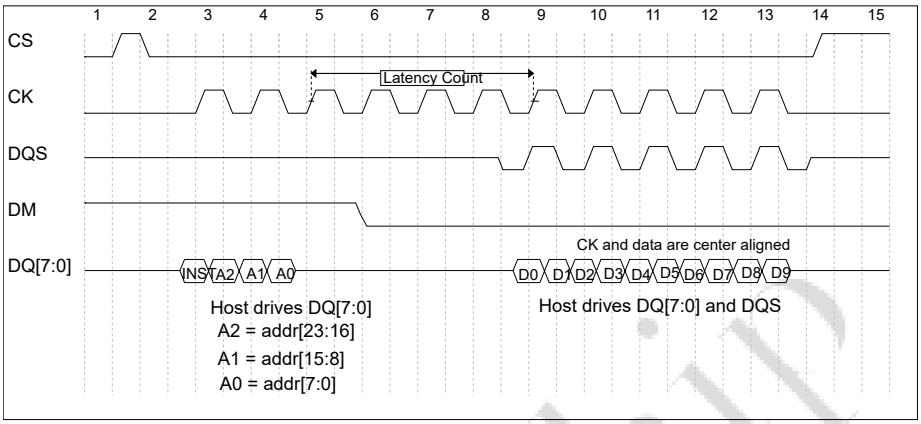


图 9–17 OPI 执行查找表图示

9.2.3.2.4. SPI LUT 配置

举例 SPI 配置（AHB 模式）如下（xSPI_CTL[5:4]配置为 SPI 模式）

表 9–10 SPI AHB 行为配置

指令（6 bits）	IO 数 (2 bits)	操作数（8 bits）	说明
CMD	0x0	0xa0	1 个 IO 输出 8 bits CMD，SDR 模式。CMD 为 0xa0
ADDR	0x0	0x20	1 个 IO 输出 ADDR，并且 ADDR 为 32 bits，SDR 模式
DUMMY	0x0	0x5	等待 5 个 xSPI_CLK Cycles
READ	0x2	0xff	4 个 IO 输入 256 bytes 数据，SDR 模式
STOP	0x0	0x0	结束命令

注：

其中 ADDR/ ADDR_DDR 的操作数只支持配置为 0x18/ 0x20，即 24 位/ 32 位地址访问，其它配置会出现 Error 状态。READ/ READ_DDR/ WRITE/ WRITE_DDR 的操作数配置为 N，则实际数据长度为 N+1。READ/ READ_DDR/ WRITE/ WRITE_DDR 的 IO 数配置为 0，则为单线 SPI 模式，此时输入为 D[1]。

1. 执行 图 9–18 : AHB 通道 SPI 执行查找表，CMD 为 0xa0，SDR 模式 1 个 IO 输出。
2. ADDR 为 32 bits 寻址，SDR 模式 1 个 IO 输出，ADDR[31:0] 由 xSPI_ADDR 寄存器决定。
3. 接着 5 个 DUMMY Cycles。
4. 接着读 256 bytes DATA， SDR 模式 8 个 IO 输入。
5. 最后 STOP 命令结束传输。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

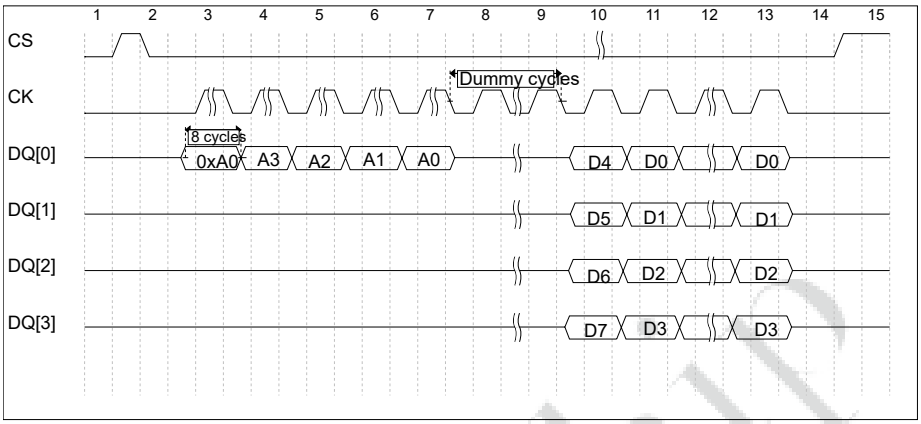


图 9–18 AHB 通道 SPI 执行查找表

举例 SPI 配置（AXI 模式）如下（xSPI_CTL[5:4] 配置为 SPI 模式）

表 9–11 SPI AXI 行为配置

指令（6 bits）	IO 数 (2 bits)	操作数（8 bits）	说明
CMD	0x0	0x77	一个 IO 输出 8 bits CMD，SDR 模式。CMD 为 0x77
DUMMY	0x0	0x6	等待六个 xSPI_CLK Cycles
WRITE	0x2	0x1	写一个 byte，AXI Burst Type 自动触发，xSPI Burst Type 预先配置值，默认 Linear 为 0x10，Wrapped 为 0x40
JUMP_ID	0x0	0x4	跳转 4 号 ID 执行 Burst 读操作
CMD	0x0	0xa0	一个 IO 输出 8 bits CMD，SDR 模式。CMD 为 0xa0
ADDR	0x0	0x20	一个 IO 输出 ADDR，并且 ADDR 为 32 bits，SDR 模式
DUMMY	0x0	0x5	等待五个 xSPI_CLK Cycles
READ	0x2	0xff	四个 IO 输入 256 bytes 数据，SDR 模式
STOP	0x0	0x0	结束命令

注：

其中 ADDR/ ADDR_DDR 的操作数只支持配置为 0x18/ 0x20，即 24 位/ 32 位地址访问，其它配置会出现 Error 状态。READ/ READ_DDR/ WRITE/ WRITE_DDR 的操作数配置为 N，则实际数据长度为 N+1。READ/ READ_DDR/ WRITE/ WRITE_DDR 的 IO 数配置为 0，则为单线 SPI 模式，此时输入为 D[1]。

1. 执行 图 9–19 : AXI 通道 SPI 执行查找表，如果为 AXI 访问，则执行一个 CMD 0x77，SDR 模式 1 个 IO 输出。
2. 接着 6 个 Cycles，3 个 bytes 时间。
3. 接着写 1 个 byte，SDR 模式 4 个 IO 输出。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 如果 Burst Type 为 Linear，则这个 byte 为 0x10（可在 xSPI_RTR 寄存器配置）。
- 如果 Burst Type 为 Wrapped，则这个 byte 为 0x40（可在 xSPI_RTR 寄存器配置配置）。

4. 接着跳转到 4 号 ID 开始执行 CMD 0xa0，SDR 模式 1 个 IO 输出。
5. ADDR 为 32 bits 寻址，SDR 模式 1 个 IO 输出，ADDR[31:0] 由访问的 Memory Mapping 决定。
6. 接着 5 个 DUMMY Cycles。
7. 接着由 AXI 决定读数据长度， SDR 模式 8 个 IO 输入。
8. 最后 STOP 命令结束传输。

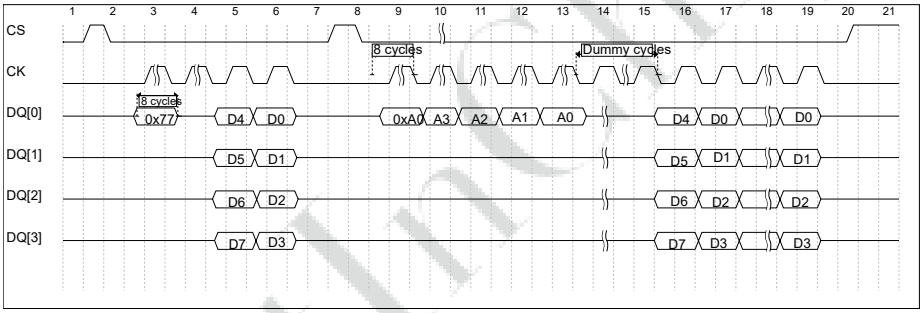


图 9–19 AXI 通道 SPI 执行查找表

9.2.3.3. AHB 通道以及 AXI 通道访问

9.2.3.3.1. AHB 通道访问

AHB 访问通道需将访问外设的地址写入 xSPI_ADDR 寄存器，CMD 以及读写信息在 LUTn 寄存器配置，xSPI_START 寄存器写入对应的 LUTn ID 号触发读写行为。

1. 若是写操作，需提前将数据写进 xSPI_TX DATA 寄存器，此寄存器为 32 位宽，硬件将写数据搬至 TX FIFO。

   FIFO 位宽为 8 bits，bit[7:0] 为第一个 byte，bit[15:8] 为第二个 byte，bit[23:16] 为第三个 byte，bit[31:24] 为第四个 byte，最多可写入 64 个 bytes，可查看 xSPI_FSR 查看 FIFO 中的数据个数。

2. 若是读操作，则读数据存进 RX FIFO，FIFO 位宽为 8 bits，最多可写入 64 个 bytes，可在 xSPI_FSR 查看 FIFO 中的数据个数。

   可在 xSPI_RX DATA 寄存器将数据取走，此寄存器为 32 位宽，bit[7:0] 为第一个 byte，bit[15:8] 为第二个 byte，bit[23:16] 为第三个 byte，bit[31:24] 为第四个 byte。

9.2.3.3.2. AXI 通道访问

AXI 访问通道需提前配置 LUTn 寄存器，为接下来的 AXI 做预准备。

AXI 对 Memeory Mapping 进行读写访问直接触发 xSPI 的读写操作，被访问的 Memory Mapping 地址映射为 Memory Device 的地址。

- 若是 AXI 写访问，则数据存进 AXI TX Buffer，触发 xSPI 从 LUT0 开始执行传输。
- 若是 AXI 读访问，则 AXI 发送读请求后，触发 xSPI 从 LUT16 开始执行数据传输，Memory Device 输入的数据存进 AXI RX Buffer，完成传输后 AXI 取走数据。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.2.3.4. xSPI 单通道与双通道模式

9.2.3.4.1. XSPI 单通道模式

单通道支持 8/ 16 bits 模式。在单通道模式下可进行

AHB 通道访问 以及 AXI 通道访问。xSPI 提供两套控制器：

- 8 bits 模式时：两个控制器分别控制通道 0 与通道 1。如需选择此模式，可通过配置 xSPI_CTL bit[6] 为 1，xSPI_TCR bit[4] 配置为 0 或 1 选择通道 0 或通道 1。
- 16 bits 模式时：如需选择此模式，可配置 xSPI_CTL bit[6]、xSPI_CTL bit[16] 为 1。

9.2.3.4.2. XSPI 双通道模式

xSPI 双通道模式仅支持 AHB 通道访问。可通过 xSPI_CTL bit[6] 配置为 1 选择双通道。

双通道并行模式：支持两片并行读写操作，此时相当于 1 个 Cycle 读写 2 个 bytes，数据以通道 1 为高 8 位，通道 0 为低 8 位的方式存储。

此时 Memory Mapping 不再是偏移量一一对应，而是遵循如下计算规则：

- Memory Device A 地址 =（ Memory Mapping 访问地址 － Memory Mapping 基地址 ）÷ 2
- Memory Device B 地址 =（ Memory Mapping 访问地址 － 1 － Memory Mapping 基地址 ）÷ 2

举例：假设 Memory Mapping 基地址为 0x30000000

- Memory Mapping 访问地址为 0x30000000，则 Memory Device A 地址为 0x0。
- Memory Mapping 访问地址为 0x30000001，则 Memory Device B 地址为 0x0。
- Memory Mapping 访问地址为 0x30000002，则 Memory Device A 地址为 0x1。
- Memory Mapping 访问地址为 0x30000003，则 Memory Device B 地址为 0x1。

表 9–12 数据存储方式

Memory Mapping Address	Memory Device Address	Device B(CS1)	Device A(CS0)
0x3000_0000	0x0 0x1 0x2 0x3	DQ[15:8] DQ[31:24] DQ[47:40] DQ[63:56]	DQ[7:0] DQ[23:16] DQ[39:32] DQ[55:48]
0x3000_0008	0x4 0x5 0x6 0x7	DQ[15:8] DQ[31:24] DQ[47:40] DQ[63:56]	DQ[7:0] DQ[23:16] DQ[39:32] DQ[55:48]
…	…	…	…

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.2.3.5. 传输模式配置

xSPI SDR 支持四种不同的传输模式，由外设决定使用某种传输模式，可以通过 SPI_TCR.bit1 配置时钟极性，SPI_TCR.bit0 配置时钟相位。

xSPI 控制器根据 xSPI_CLK 信号从移位寄存器中读取或写入数据，在任意一种传输格式下被锁存。

- 在时钟相位为 0，极性为 0 或者相位 1，极性为 1 的模式下，输出数据在时钟的上升沿采样，下降沿移位。
- 在时钟相位为 0，极性为 1 或者相位为 1，极性为 0 模式下，输出数据在时钟的下降沿采样，上升沿移位。

xSPI_CLK 的决定因素如下所示：

- POL 定义了 xSPI_CLK 为空闲状态时的信号极性：当 POL 为 1，xSPI_CLK 为高电平。当 POL 为 0，xSPI_CLK 为低电平。
- PHA 的状态决定了 xSPI_CLK 在第一个时钟边沿采样数据还是保持数据：当 PHA 为 1，在第一个时钟边沿保持数据。当 PHA 为 0，在第一个时钟边沿采样数据。

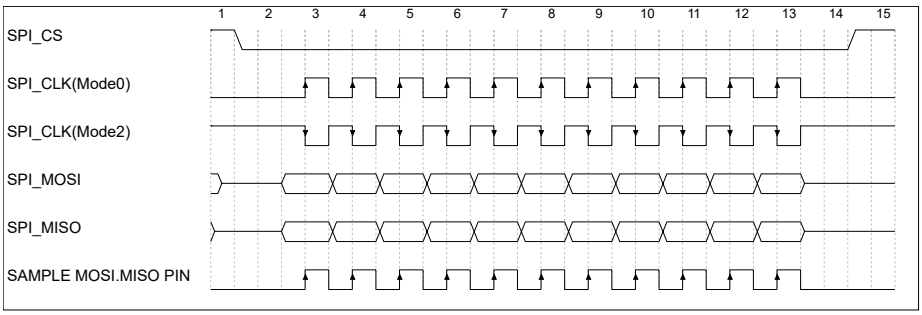
下表总结了四种不同的传输模式。

注：

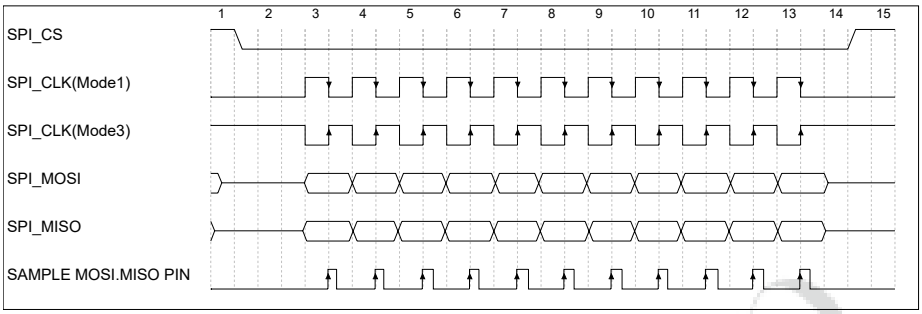
xSPI DDR 无论为那种模式，均以第一个边沿开始采样。

表 9–13 传输模式

xSPI 传输模式	POL（极性）	PHA（相位）	第一个边沿	第二个边沿
0	0	0	上升沿采样	下降样保持
1	0	1	上升沿保持	下降沿采样
2	1	0	下降沿采样	上升沿保持
3	1	1	下降沿保持	上升沿采样



Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



9.2.3.6. xSPI 错误类型

对于 xSPI 控制器，任何一种错误发生，硬件都会将 xSPI 中断状态寄存器中对应的位域置 1 并停止传输，主要的错误类型描述如下：

表 9–14 xSPI 错误类型

xSPI 错误类型	原因	说明
TX_FIFO 下溢出	从空的 FIFO 中读取数据。	此种情况下： 1. xSPI 控制器停止传输并将对应的中断状态位 TF_UDF 置 1。 2. xSPI 控制器产生一个硬件中断信号。 3. 处理完该异常后需要软件将该状态位清零。 4. 如果开始新的数据传输，需要在 xSPI 全局控制寄存器中将 RESET 位置 1，将 FIFO 进行复位。
TX_FIFO 上溢出	向满的 FIFO 中发送数据。	此种情况下： 1. xSPI 控制器会停止传输并将对应的中断状态位 OVF 置 1 2. xSPI 控制器产生一个硬件中断信号。 3. 处理完该异常后需要软件将该状态位清零。 4. 如果开始新的数据传输，需要在 xSPI 全局控制寄存器中将 RESET 位置 1，将 FIFO 进行复位。
RX_FIFO 下溢出	从空的 FIFO 中读取数据。	1. xSPI 控制器停止传输并将对应的中断状态位 RF_OVF 置 1 2. xSPI 控制器产生一个硬件中断信号。 3. 处理完该异常后需要软件将该状态位清零。 4. 如果开始新的数据传输，需要在 xSPI 全局控制寄存器中将 RESET 位置 1，将 FIFO 进行复位。
RX_FIFO 上溢出	向满的 FIFO 中发送数据。	1. xSPI 控制器停止传输并将对应的中断状态位 RF_OVF 置 1 2. xSPI 控制器产生一个硬件中断信号。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 9–14 xSPI 错误类型 (续)

xSPI 错误类型	原因	说明
		3. 处理完该异常后需要软件将该状态位清零。 4. 如果开始新的数据传输，需要在 xSPI 全局控制寄存器中将 RESET 位置 1，将 FIFO 进行复位。
CS0_TO_ERROR	通道 0 传输超时。	–
CS1_TO_ERROR	通道 1 传输超时。	–
LUT_ADDR_ERROR	LUT 配置地址位宽错误。	LUT 地址位宽只支持 24 位以及 32 位宽。
LUT_INSTRUCTION_ERROR	LUT 配置指令错误。	此错误发生在 LUT 配置未定的指令，已定义的见查找表 (LUT) 功能。
AHB_TRAN_ERROR	AHB 通道传输错误。	–
AXI_TRAN_ERROR	AXI 通道传输错误。	–
XCCELA_TRAN_ERROR	XCCELA 协议行为 LUT 配置错误。	–
HYPERBUS_TRAN_ERROR	HYPERBUS 协议行为 LUT 配置错误。	–
OPI_TRAN_ERROR	OPI 协议行为 LUT 配置错误。	–
AXI_ERROR	AXI 总线发生错误。	–
XIP_ERROR	XSPI_CTL[2] 未置 1 使能 XIP 功能，但发生 AXI 访问，此时 XSPI 模块发生 XIP 错误并且 Bypass AXI 访问。	–

9.2.4. 寄存器列表

表 9–15 xSPI 寄存器列表

偏移	缩写	寄存器名称	章节
0x0000	XSPI_CTL	xSPI 控制	0x0000 XSPI_CTL
0x0004	XSPI_CLK	xSPI 时钟	0x0004 XSPI_CLK

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 9–15 xSPI 寄存器列表 (续)

偏移	缩写	寄存器名称	章节
0x0008	XSPI_TCR	xSPI 传输控制寄存器	0x0008 XSPI_TCR
0x000C	XSPI_STAS	xSPI 状态	0x000C XSPI_STAS
0x0010	XSPI_CS0_CTL	xSPI 通道 0 控制	0x0010 XSPI_CS0_CTL
0x0014	XSPI_CS0_DCTL	xSPI 通道 0 DLL 控制	0x0014 XSPI_CS0_DCTL
0x0018	XSPI_CS1_CTL	xSPI 通道 1 控制	0x0018 XSPI_CS1_CTL
0x001C	XSPI_CS1_DCTL	xSPI 通道 1 DLL 控制	0x001C XSPI_CS1_DCTL
0x0020	XSPI_IER	xSPI 中断使能寄存器	0x0020 XSPI_IER
0x0024	XSPI_ISR	xSPI 中断状态寄存器	0x0024 XSPI_ISR
0x0028	XSPI_FCR	xSPI FIFO 控制寄存器	0x0028 XSPI_FCR
0x002C	XSPI_FSR	xSPI FIFO 状态寄存器	0x002C XSPI_FSR
0x0030	XSPI_START	xSPI 启动寄存器	0x0030 XSPI_START
0x0034	XSPI_ADDR	xSPI 地址配置寄存器	0x0034 XSPI_ADDR
0x0038	XSPI_FMR	xSPI 格式配置寄存器	0x0038 XSPI_FMR
0x0040	XSPI_BTR	xSPI 突发类型寄存器	0x0040 XSPI_BTR
0x0044	XSPI_RCC	xSPI 读命令控制	0x0044 XSPI_RCC
0x0048	XSPI_NDMA_MODE_CTL	xSPI DMA 模式控制	0x0048 XSPI_NDMA_MODE_CTL
0x0050	XSPI_TO	xSPI 超时配置	0x0050 XSPI_TO
0x0054	XSPI_LCKCR	xSPI 锁定配置	0x0054 XSPI_LCKCR
0x0058	XSPI_LUT_UP	xSPI 查找表更新	0x0058 XSPI_LUT_UP
0x0060	0x0060 XSPI_CS0_SEQUENCE	xSPI 通道 0 线序	0x0060 XSPI_CS0_SEQUENCE
0x0064	XSPI_CS1_SEQUENCE	xSPI 通道 1 线序	0x0064 XSPI_CS1_SEQUENCE
0x0068	XSPI_IO_CTL	xSPI IO 控制	0x0068 XSPI_IO_CTL

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 9–15 xSPI 寄存器列表 (续)

偏移	缩写	寄存器名称	章节
0x0070	XSPI_CS0_IOCFG1	xSPI 通道 0 IO 配置 1	0x0070 XSPI_CS0_IOCFG1
0x0074	XSPI_CS0_IOCFG2	xSPI 通道 0 IO 配置 2	0x0074 XSPI_CS0_IOCFG2
0x0078	XSPI_CS0_IOCFG3	xSPI 通道 0 IO 配置 3	0x0078 XSPI_CS0_IOCFG3
0x007C	XSPI_CS0_IOCFG4	xSPI 通道 0 IO 配置 4	0x007C XSPI_CS0_IOCFG4
0x0080	XSPI_CS1_IOCFG1	xSPI 通道 1 IO 配置 1	0x0080 XSPI_CS1_IOCFG1
0x0084	XSPI_CS1_IOCFG2	xSPI 通道 1 IO 配置 2	0x0084 XSPI_CS1_IOCFG2
0x0088	XSPI_CS1_IOCFG3	xSPI 通道 1 IO 配置 3	0x0088 XSPI_CS1_IOCFG3
0x008C	XSPI_CS1_IOCFG4	xSPI 通道 1 IO 配置 4	0x008C XSPI_CS1_IOCFG4
0x0090	XSPI_TRAINING_CFG	xSPI TRAINING 配置	0x0090 XSPI_TRAINING_CFG
0x0094	XSPI_TRAINING_PATTERN	xSPI TRAINING PATTERN	0x0094 XSPI_TRAINING_PATTERN
0x0100	XSPI_LUTn	xSPI 查找表 n	0x0100 XSPI_LUTn
0x0200	XSPI_TDR	xSPI 发送数据寄存器	0x0200 XSPI_TDR
0x0300	XSPI_RDR	xSPI 接收数据寄存器	0x0300 XSPI_RDR
0x0400	XSPI_DEBUG	xSPI 调试寄存器	0x0400 XSPI_DEBUG
0x0404	XSPI_DEBUG_SEL	xSPI 调试选择寄存器	0x0404 XSPI_DEBUG_SEL
0xFFC	VERSION	xSPI 版本寄存器	0xFFC VERSION

9.2.5. 寄存器描述

9.2.5.1. 0x0000 XSPI_CTL

默认值：0x00000100			xSPI 控制 （xSPI Control）
位域	类型	默认值	描述
31:24	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000100			xSPI 控制（xSPI Control）
位域	**类型**	**默认值**	**描述**
19	R/W	0x0	Xccela_QPI_CTL QPI 模式控制，XSPI_MODE_SEL 配置为 Xccela 模式时，此位有效。
18	R/W	0x0	AXI_ABITER_EN 响应 AXI 读请求，延缓写请求。
17	R/W	0x0	Column_Address_Control • 0：byte_address(A0—A10) • 1：word_address(A0—A9) 可选地址映射为字节（1 个字节）地址或者字（2 个字节）地址。 应用场景：PARAM 从 x8 切换到 x16 时，存在字节地址切换到字地址，此时 Column Address 位宽丢弃最高比特，默认 Column Addrrss 长度为 2 K。Hyperbus 模式此功能无效。
16	R/W	0x0	PIN_CTL • 0：x8 • 1：x16 此位配置 1 时，XSPI_CTL[6] 需配置为 PARALLEL_MODE 必须配置为 1，此时 CS0 的 DQ[7:0] 为 x16 DQ[7:0]，CS1 的 DQ[7:0] 为 x16 DQ[15:8]
15	–	–	–
14:13	R/W	0x0	Boundary_Control • 0：2K • 1：1K • 3/4：Reserved Boundary_En 需要配置为 1
12	R/W	0x0	Boundary_En 边界控制使能
11:10	–	–	–
9	R/W	0x0	Reset_en DM 引脚使能为 RESET 引脚，bit[8] 控制电平。
8	R/W	0x1	Reset_level • 0：0 电平 • 1：1 电平

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000100			xSPI 控制（xSPI Control）
位域	类型	默认值	描述
7	R/W	0x0	TIMEOUT_EN
6	R/W	0x0	PARALLEL_MODE • 0：单通道传输，CS_SEL 需选中通道 0 或通道 1 • 1：双通道传输
5:4	R/W	0x0	XSPI_MODE_SEL xSPI 协议模式选择 • 0：Xccela 模式 • 1：Hyperbus 模式 • 2：OPI 模式 • 3：SPI 模式 硬件根据查找表 (LUT) 功能已配置的 Command 输出符合对应协议行为，其中 SPI 模式包括 1/ 2/ 4/ 8 线模式，由查找表配置实现。
3	R/W	0x0	AXI_WRAP_BURST_CTL • 0：wrap burst • 1：wrap burst 拆分为 linear burst 输出 wrap burst 只支持 wrap4。
2	R/W	0x0	XIP_EN XIP 模式使能控制，用于控制 AXI 访问通道 • 0：禁止，AXI 通道不工作 • 1：使能，AXI 访问正常工作 若有 AXI 访问，此时不能挂死 AXI 总线。
1	R/W	0x0	IDL_LOW_POWER_EN 空闲状态低功耗功能使能
0	R/W	0x0	XSPI_EN XSPI 模块使能控制 • 0：禁止 • 1：使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.2.5.2. 0x0004 XSPI_CLK

默认值：0x00000002			xSPI 时钟 （xSPI Clock）
位域	类型	默认值	描述
31:13	–	–	–
12	R/W	0x0	CLK_SEL 时钟分频器选择 • 0：时钟分频器 1 • 1：时钟分频器 2
11:8	R/W	0x0	CDR1_M 时钟分频器 1 SPI_SCLK 时钟根据如下公式计算： SPI_CLK = 源时钟/(2^CDR1_M)
7:0	R/W	0x2	CDR2_N 时钟分频器 2 SPI_CLK=源时钟/2*（CDR2_N +1）

9.2.5.3. 0x0008 XSPI_TCR

默认值：0x00000001			xSPI 传输控制寄存器 （xSPI Transmit Control Reg）
位域	类型	默认值	描述
31:28	R/W	0x0	OPI_HOLD_EX
27:24	R/W	0x0	DQS_CLK_GATING_CTL • 0：normal • 1：before • 2：delay • 3：bypass
23:20	R/W	0x0	CS_RD_HOLD_CTL CS 读保持时间 指示两个命令操作之间 CS 至少保持无效的时间，n 代表 n 个 Cycles。 XSPI_BUSY 为 1 时不可配置。
19:16	R/W	0x0	CS_WR_HOLD_CTL CS 写保持时间

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000001			xSPI 传输控制寄存器（xSPI Transmit Control Reg）
位域	类型	默认值	描述
			指示两个命令操作之间 CS 至少保持无效的时间，n 代表 n 个 Cycles。 XSPI_BUSY 为 1 时不可配置。
15:12	R/W	0x0	CS SETUP 控制 片选有效后，数据与 CLK 建立时间，指示何时输出有效数据，0 与 1 代表 1 个 Cycles，n(n>1) 代表 n 个 XSPI_CLK Cycles，以此类推。 XSPI_BUSY 为 1 时不可配置。
11	R/W	0x0	JUMP_INS_EN JUMP_INS 使能控制 XIP 模式使能后，例如 NORFLSH 连续读数据的应用场景，可以使用 LUT 定义的 JUMP_INS 指令
10:9	–	–	–
8	R/W	0x0	DUMMY_TYPE Dummy Cycles 数据填充类型 • 0：填充 0 • 1：填充 1 XSPI_BUSY 为 1 时不可配置。
7	R/W	0x0	CS_LEVEL 片选信号电平配置 • 0：低电平 • 1：高电平 **注：** 单通道模式此处配置的是 CS0 还是 CS1 由 CS_SEL 决定，双通道模式 CS0 与 CS1 同时生效。 XSPI_BUSY 为 1 时不可配置。
6	R/W	0x0	CS_OWNER 片选信号控制选择 • 0：XSPI 控制器 • 1：软件控制 通常由控制器自动发送 CS 信号，当该位置 1，必须手动配置 CS_LEVEL 的电平状态。 XSPI_BUSY 为 1 时不可配置

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000001			xSPI 传输控制寄存器（xSPI Transmit Control Reg）
位域	类型	默认值	描述
			注： 单通道模式此处配置的是 CS0 还是 CS1 由 CS_SEL 决定，双通道模式 CS0 与 CS1 同时生效。。
5	–	–	–
4	R/W	0x0	CS_SEL 片选信号选择，决定选中通道 0 或者通道 1 • 0：选择 CS0 • 1：选择 CS1 XSPI_BUSY 为 1 时不可配置。
3	–	–	–
2	R/W	0x0	CS_POL 片选信号极性控制 • 0：低电平有效（空闲状态为高） • 1：高电平有效（空闲状态为低） XSPI_BUSY 为 1 时不可配置。 **注：** 单通道模式此处配置的是 CS0 还是 CS1 由 CS_SEL 决定，双通道模式 CS0 与 CS1 同时生效。
1	R/W	0x0	CPOL 时钟极性控制 • 0：Active High Polarity（空闲时为低电平） • 1：Active Low Polarity（空闲时为高电平） **注：** 此处配置为 CK 状态，CK# 需保持与 CK 差分关系。 XSPI_BUSY 为 1 时不可配置。 **注：** 单通道模式此处配置的是 CS0 还是 CS1 由 CS_SEL 决定，双通道模式 CS0 与 CS1 同时生效。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000001			xSPI 传输控制寄存器（xSPI Transmit Control Reg）
位域	类型	默认值	描述
0	R/W	0x1	CPHA SPI 时钟/ 数据相位控制 • 0：Phase 0（第一个边沿采样数据） • 1：Phase 1（第二个边沿采样数据） **注：** 此处配置为 CK 状态，CK# 需保持与 CK 差分关系。 XSPI_BUSY 为 1 时不可配置。 **注：** 单通道模式此处配置的是 CS0 还是 CS1 由 CS_SEL 决定，双通道模式 CS0 与 CS1 同时生效。。

9.2.5.4. 0x000C XSPI_STAS

默认值：0x00000000			xSPI 状态（xSPI Status）
位域	类型	默认值	描述
31:3	–	–	–
2	R	0x0	AHB_TRANS 此位置 1 指示为 AHB 通道传输。
1	R	0x0	AXI_TRANS 此位置 1 指示为 AXI 通道传输。
0	R	0x0	XSPI_BUSY 此位置 1 指示正在传输中。

9.2.5.5. 0x0010 XSPI_CS0_CTL

默认值：0x00200000			xSPI 通道 0 控制（xSPI Chip Select 0 Control）
位域	类型	默认值	描述
31:29	–	–	–
28:24	R/W	0x0	WR_DELAY_CHAIN_SEL

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00200000			xSPI 通道 0 控制（xSPI Chip Select 0 Control）
位域	**类型**	**默认值**	**描述**
			输出时钟 Delay Chain 选择
23:22	–	–	–
21	R/W	0x1	WR_PHASE_SEL • 当 SPI CPHA=1 时 ◦ 0：0° ◦ 1：90° • 当 SPI CPHA=0 时 ◦ 0：180° ◦ 1：270° **注：** 使用 SPI 四模式时，此位需配为 0。
20	R/W	0x0	WR_DELAY_CHAIN_EN 此位配 1，则使能 Delay Chain。
19:18	–	–	–
17:16	R/W	0x0	RD_PHASE 内部采样时钟相位选择 • 0：0° • 1：90° • 2：180° • 3：360°
15	–	–	–
14:12	R/W	0x0	RD_DELAY_CYCLE 内部采样延迟时钟数
11:9	–	–	–
8:4	R/W	0x0	RD_DELAY_CHAIN_SEL Delay Chain 选择
3	R/W	0x0	RD_VALID_CONTROL

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00200000			xSPI 通道 0 控制 （xSPI Chip Select 0 Control ）
位域	类型	默认值	描述
			• 0：当前 Cycle 有效 • 1：延迟一个 Cycle 有效 读操作时，内部控制逻辑访问异步 FIFO 时间控制
2	R/W	0x0	RD_DELAY_CHAIN_EN 此位配 1，则使能 Delay Chain。
1:0	R/W	0x0	RD_SAMPLE_CTL 读数据采样控制 • 0：DQS 输入时钟采样 DLL 控制 • 1：内部时钟采样 Delay Chain 控制 • 2/3：DQS 输入时钟采样 Delay Chain 控制

9.2.5.6. 0x0014 XSPI_CS0_DCTL

默认值：0x00000000			xSPI 通道 0 DLL 控制 （XSPI Chip Select 0 DLL Control ）
位域	类型	默认值	描述
31:30	–	–	–
29	R/W	0x0	FORCE_LOCK
28	R/W	0x0	EN_ATB
27	–	–	–
26:24	R/W	0x0	REG_ATBSEL
23:22	–	–	–
21:20	R/W	0x0	REG_BYPASS • 0：772p–935p–1.191n • 1：2.2n–2.92n–4.1n • 2：3.63n–4.92n–6.87n • 3：5.04n–6.91n–9.66n
19:18	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			xSPI 通道 0 DLL 控制 （XSPI Chip Select 0 DLL Control）
位域	**类型**	**默认值**	**描述**
17:16	R/W	0x0	REG_DLY
15:14	–	–	–
13:12	R/W	0x0	REG_ICP 参考时钟范围 • 0：50~100M • 1：100~150M • 2：150~200M • 3：200~266M
11:8	R/W	0x0	PHASE_SEL 输入时钟相位选择 • 0：22.5° • 1：45° • 2：67.5° • 3：90° • 4：112.5° • 5：135° • 6：157.5° • 7：180° • 8：202.5° • 9：225° • 10：247.5° • 11：270° • 12：292.5° • 13：315° • 14：337.5° • 15：BYPASS 此位配置 0~14 时，需配置下列值： • EN_DLL 为 1 • EN_VCDL 为 1 • EN_CP 为 1 • EN_BYPASS 为 0

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			xSPI 通道 0 DLL 控制（XSPI Chip Select 0 DLL Control）
位域	类型	默认值	描述
			此位配置 15 时，需配置下列值： • EN_DLL 为 0 • EN_VCDL 为 1 • EN_CP 为 0 • EN_BYPASS 为 1 若配置 EN_DLL 为 0，EN_VCDL 为 0，EN_CP 为 0，EN_BYPASS 为 0，则输入时钟为 0 相位。
7:6	–	–	–
5	R/W	0x0	En_lvs
4	R/W	0x0	En_ldo
3	R/W	0x0	EN_BYPASS
2	R/W	0x0	EN_CP
1	R/W	0x0	EN_VCDL
0	R/W	0x0	EN_DLL 使能上升沿检测参考时钟周期 重新配置时钟频率后，重新配置此位使能 DLL 功能（在 SYS_CFG 模块中配置）。

9.2.5.7. 0x0018 XSPI_CS1_CTL

默认值：0x00200000			xSPI 通道 1 控制（xSPI Chip Select 1 Control）
位域	类型	默认值	描述
31:29	–	–	–
28:24	R/W	0x0	WR_DELAY_CHAIN_SEL 输出时钟 Delay Chain 选择
23:22	–	–	–
21	R/W	0x1	WR_PHASE_SEL

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00200000			xSPI 通道 1 控制 （xSPI Chip Select 1 Control）
位域	类型	默认值	描述
			• 当 SPI CPHA=1 时 ◦ 0：0° ◦ 1：90° • 当 SPI CPHA=0 时 ◦ 0：180° ◦ 1：270° **注：** 使用 SPI 四模式时，此位需配为 0。
20	R/W	0x0	WR_DELAY_CHAIN_EN 此位配 1，则使能 Celay Chain。
19:18	–	–	–
17:16	R/W	0x0	RD_PHASE 内部采样时钟相位选择 • 0：0° • 1：90° • 2：180° • 3：360°
15	–	–	–
14:12	R/W	0x0	RD_DELAY_CYCLE 内部采样延迟时钟数
11:9	–	–	–
8:4	R/W	0x0	RD_DELAY_CHAIN_SEL Delay Chain 选择
3	R/W	0x0	RD_VALID_CONTROL • 0：当前 Cycle 有效 • 1：延迟 1 Cycle 有效 读操作时，内部控制逻辑访问异步 FIFO 时间控制。
2	R/W	0x0	RD_DELAY_CHAIN_EN

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00200000			xSPI 通道 1 控制 （xSPI Chip Select 1 Control）
位域	类型	默认值	描述
			此位配 1，则使能 Delay Chain。
1:0	R/W	0x0	RD_SAMPLE_CTL 读数据采样控制 • 0：DQS 输入时钟采样 DLL 控制 • 1：内部时钟采样 Delay Chain 控制 • 2/3：DQS 输入时钟采样 Delay Chain 控制

9.2.5.8. 0x001C XSPI_CS1_DCTL

默认值：0x00000000			xSPI 通道 1 DLL 控制 （XSPI Chip Select 1 DLL Control）
位域	类型	默认值	描述
31:30	–	–	–
29	R/W	0x0	FORCE_LOCK
28	R/W	0x0	EN_ATB
27	–	–	–
26:24	R/W	0x0	REG_ATBSEL
23:22	–	–	–
21:20	R/W	0x0	REG_BYPASS • 0：772p–935p–1.191n • 1：2.2n–2.92n–4.1n • 2：3.63n–4.92n–6.87n • 3：5.04n–6.91n–9.66n
19:18	–	–	–
17:16	R/W	0x0	REG_DLY
15:14	–	–	–
13:12	R/W	0x0	REG_ICP

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			xSPI 通道 1 DLL 控制 （XSPI Chip Select 1 DLL Control ）
位域	类型	默认值	描述
			• 0：50~100M • 1：100~150M • 2：150~200M • 3：200~266M
11:8	R/W	0x0	PHASE_SEL • 0：22.5° • 1：45° • 2：67.5° • 3：90° • 4：112.5° • 5：135° • 6：157.5° • 7：180° • 8：202.5° • 9：225° • 10：247.5° • 11：270° • 12：292.5° • 13：315° • 14：337.5° • 15：BYPASS 此位配置 0~14 时，需配置下列值： • EN_DLL 为 1 • EN_VCDL 为 1 • EN_CP 为 1 • EN_BYPASS 为 0 此位配置 15 时，需配置下列值： • EN_DLL 为 0 • EN_VCDL 为 1 • EN_CP 为 0 • EN_BYPASS 为 1

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			xSPI 通道 1 DLL 控制（XSPI Chip Select 1 DLL Control）
位域	类型	默认值	描述
			若配置 EN_DLL 为 0，EN_VCDL 为 0，EN_CP 为 0，EN_BYPASS 为 0，则输入时钟为 0 相位。
7:6	–	–	–
5	R/W	0x0	En_lvs
4	R/W	0x0	En_ldo
3	R/W	0x0	EN_BYPASS
2	R/W	0x0	EN_CP
1	R/W	0x0	EN_VCDL
0	R/W	0x0	EN_DLL 使能上升沿检测参考时钟周期 重新配置时钟频率后，重新配置此位使能 DLL 功能。

9.2.5.9. 0x0020 XSPI_IER

默认值：0x00000000			xSPI 中断使能寄存器（xSPI Interrupt Enable Register）
位域	类型	默认值	描述
31:25	–	–	–
24	R/W	0x0	XIP ERROR
23	R/W	0x0	AXI ERROR
22	R/W	0x0	OPI_ERROR OPI 协议行为配置错误使能 • 0：禁止 • 1：使能 XSPI 配置为 OPI 模式，并且 LUT 配置不符合 OPI 协议。
21	R/W	0x0	HYPERBUS_ERROR HYPERBUS 协议行为配置错误使能 • 0：禁止 • 1：使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			xSPI 中断使能寄存器（xSPI Interrupt Enable Register）
位域	类型	默认值	描述
			xSPI 配置为 HYPERBUS 模式，并且 LUT 配置不符合 HYPERBUS 协议。
20	R/W	0x0	XCCELA_ERROR XCCELA 协议行为配置错误 • 0：禁止 • 1：使能 XSPI 配置为 XCCELA 模式，并且 LUT 配置不符合 XCCELA 协议。
19	R/W	0x0	AXI_TRAN_ERROR_ENAXI 传输错误中断使能 • 0：禁止 • 1：使能
18	R/W	0x0	AHB_TRAN_ERROR_ENAHB 传输错误中断使能 • 0：禁止 • 1：使能
17	R/W	0x0	LUT_INSTRUCTION_ERROR_ENLUT 指令错误中断使能 • 0：禁止 • 1：使能
16	R/W	0x0	LUT_ADDR_OPREND_ERROR_ENLUT 地址位宽错误中断使能 • 0：禁止 • 1：使能
15	R/W	0x0	CS1_TO_ENCS1 传输超时中断使能 • 0：禁止 • 1：使能
14	R/W	0x0	CS0_TO_ENCS0 传输超时中断使能 • 0：禁止 • 1：使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			xSPI 中断使能寄存器（xSPI Interrupt Enable Register）
位域	类型	默认值	描述
13	R/W	0x0	CS1_DONE_ENCS1 传输完成中断使能 • 0：禁止 • 1：使能
12	R/W	0x0	CS0_DONE_ENCS0 传输完成中断使能 • 0：禁止 • 1：使能
11	R/W	0x0	TF_UDR_INT_ENTX FIFO 下溢出中断使能 • 0：禁止 • 1：使能 当硬件读取空的 TX FIFO 会产生此中断。
10	R/W	0x0	TF_OVF_INT_ENTX FIFO 上溢出中断使能 • 0：禁止 • 1：使能 当 CPU/ DMA 写入 Full 的 TX FIFO 会产生此中断。
9	R/W	0x0	RF_UDR_INT_ENRX FIFO 下溢出中断使能 • 0：禁止 • 1：使能 当 CPU/ DMA 读取空的 RX FIFO 会产生此中断
8	R/W	0x0	RF_OVF_INT_ENRX FIFO 上溢出中断使能 • 0：禁止 • 1：使能 当硬件写入 Full 的 RX FIFO 会产生此中断
7	–	–	–
6	R/W	0x0	TX_FULL_INT_ENTX

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			xSPI 中断使能寄存器（xSPI Interrupt Enable Register）
位域	类型	默认值	描述
			FIFO 满中断使能 • 0：禁止 • 1：使能
5	R/W	0x0	TX_EMP_INT_ENTX FIFO 为空中断使能 • 0：禁止 • 1：使能
4	R/W	0x0	TX_ERQ_INT_ENTX FIFO 为空请求中断使能 • 0：禁止 • 1：使能
2	R/W	0x0	RX_FULL_INT_ENRX FIFO 满中断使能 • 0：禁止 • 1：使能
1	R/W	0x0	RX_EMP_INT_ENRX FIFO 为空中断使能 • 0：禁止 • 1：使能
0	R/W	0x0	RX_ERQ_INT_ENRX FIFO 为空请求中断使能 • 0：禁止 • 1：使能

9.2.5.10. 0x0024 XSPI_ISR

默认值：0x00000033			xSPI 中断状态寄存器（xSPI Interrupt Status Register）
位域	类型	默认值	描述
31:25	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000033			xSPI 中断状态寄存器（xSPI Interrupt Status Register）
位域	类型	默认值	描述
24	R/W1C	0x0	XIP ERROR
23	R/W1C	0x0	AXI ERROR
22	R/W1C	0x0	OPI_ERROR OPI 协议行为配置错误 置 1，表示 XSPI 配置为 OPI 模式，并且 LUT 配置不符合 OPI 协议，写 1 清除。
21	R/W1C	0x0	HYPERBUS_ERROR HYPERBUS 协议行为配置错误 置 1，表示 XSPI 配置 为 HYPERBUS 模式，并且 LUT 配置不符合 HYPERBUS 协议，写 1 清除。
20	R/W1C	0x0	XCCELA_ERROR XCCELA 协议行为配置错误 置 1，表示 XSPI 配置为 XCCELA 模式，并且 LUT 配置不符合 XCCELA 协议，写 1 清除。
19	R/W1C	0x0	AXI_TRAN_ERROR AXI 传输错误 置 1，表示 AHB 正在传输时，发起 AXI 传输请求，写 1 清除。 此时需立刻结束 AHB 传输，启动 AXI 传输。
18	R/W1C	0x0	AHB_TRAN_ERROR AHB 传输错误 置 1，表示 AXI 正在传输时，发起 AHB 传输请求，写 1 清除。
17	R/W1C	0x0	LUT_INSTRUCTION_ERROR LUT 指令错误 置 1，表示 LUT 配置无定义指令，写 1 清除。
16	R/W1C	0x0	LUT_ADDR_OPREND_ERROR LUT 地址位宽错误 置 1，表示 LUT 地址位宽配置错误，写 1 清除。
15	R/W1C	0x0	CS1_TOCS1 传输超时中断，写 1 清除。
14	R/W1C	0x0	CS0_TOCS0 传输超时中断，写 1 清除。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000033			xSPI 中断状态寄存器（xSPI Interrupt Status Register）
位域	类型	默认值	描述
13	R/W1C	0x0	CS1_DONE 通道 1 传输完成中断 • 0：busy（忙等待，设备正在处理） • 1：传输完成 写 1 清除。
12	R/W1C	0x0	CS0_DONE 通道 0 传输完成中断 • 0：busy（忙等待，设备正在处理） • 1：传输完成 写 1 清除。
11	R/W1C	0x0	TF_UDFTX FIFO 下溢出中断 • 0：TXFIFO 没有下溢出 • 1：TXFIFO 下溢出 写 1 清除。
10	R/W1C	0x0	TF_OVFTX FIFO 上溢出中断 • 0：TXFIFO 没有上溢出 • 1：TXFIFO 上溢出 写 1 清除。
9	R/W1C	0x0	RF_UDFRX FIFO 下溢出中断 • 0：RXFIFO 没有下溢出 • 1：RXFIFO 下溢出 写 1 清除。
8	R/W1C	0x0	RF_OVFRX FIFO 上溢出中断 • 0：RXFIFO 没有上溢出 • 1：RXFIFO 上溢出 写 1 清除。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000033			xSPI 中断状态寄存器（xSPI Interrupt Status Register）
位域	类型	默认值	描述
7	–	–	–
6	R/W1C	0x0	TX_FULLTX FIFO 满中断 • 0：TXFIFO 未满 • 1：TXFIFO 已满
5	R/W1C	0x1	TX_EMPTX FIFO 为空中断 • 0：TXFIFO 不为空 • 1：TXFIFO 为空
4	R/W1C	0x1	TX_READYTX FIFO 完成中断 • 0：TX_WL >= TX_TRIG_LEVEL • 1：TX_WL < TX_TRIG_LEVEL
2	R/W1C	0x0	RX_FULL RX FIFO 满中断 • 0：RXFIFO 未满 • 1：RXFIFO 已满
1	R/W1C	0x1	RX_EMPRX FIFO 为空中断 • 0：RXFIFO 不为空 • 1：RXFIFO 为空
0	R/W1C	0x1	RX_READYRX FIFO 完成中断 • 0：RX_WL >= RX_TRIG_LEVEL • 1：RX_WL < RX_TRIG_LEVEL

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.2.5.11. 0x0028 XSPI_FCR

默认值：0x00400001			xSPI FIFO 控制寄存器（xSPI FIFO Control Register）
位域	类型	默认值	描述
31	R/WAC	0x0	TX_FIFO_RST TX FIFO 复位，完成后自动清零
30:25	–	–	–
24	R/W	0x0	TF_DRQ_EN TX FIFO DMA 请求使能 • 0：禁止 • 1：使能 TX FIFO 数据深度小于触发等级时请求 DMA 传输。
23	–	–	–
22:16	R/W	0x40	TX_TRIG_LEVEL TX FIFO 为空请求信号的触发等级
15	R/WAC	0x0	RF_RSTRX FIFO Reset，完成后自动清零
14:9	–	–	–
8	R/W	0x0	RF_DRQ_EN RX FIFO DMA 请求使能 • 0：禁止 • 1：使能 RX FIFO 数据深度大于触发等级时请求 DMA 传输。
7	–	–	–
6:0	R/W	0x1	RX_TRIG_LEVEL RX FIFO 为空请求信号的触发等级。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.2.5.12. 0x002C XSPI_FSR

默认值：0x00000000			xSPI FIFO 状态寄存器（xSPI FIFO Status Register）
位域	类型	默认值	描述
31	R	0x0	TB_WR TX FIFO 写缓存使能
30:28	R	0x0	TB_CNT TX FIFO Write Buffer Counter（写缓存计数器） 表示 TX FIFO 写缓存中的字（1 个字 = 4 个字节）的数量。
27:24	–	–	–
23:16	R	0x0	TF_CNT TX FIFO 计数器 • 0：在 TX FIFO 中有 0 字节数据 • 1：在 TX FIFO 中有 1 字节数据 • … • 64：在 TX FIFO 中有 64 字节数据
15	R	0x0	RB_WR RX FIFO 读缓存写使能
14:12	R	0x0	RB_CNT RX FIFO Read Buffer Counter（读缓存计数器） 表示 RX FIFO 读缓存中的字（1 个字 = 4 个字节）的数量。
11:8	–	–	–
7:0	R	0x0	RF_CNT RX FIFO 计数器 • 0：在 RX FIFO 中有 0 字节数据 • 1：在 RX FIFO 中有 1 字节数据 • … • 64：在 RX FIFO 中有 64 字节数据

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.2.5.13. 0x0030 XSPI_START

默认值：0x00000000			xSPI 启动寄存器（xSPI Start）
位域	类型	默认值	描述
31:4	–	–	–
3:0	R/WAC	0x0	START_CTL AHB 通道启动控制 • 0：从 LUT0 开始执行 • 1：从 LUT4 开始执行 • … • 7：从 LUT28 开始执行 LUT 寄存器分八组，每四个为一组。 启动完成后自动清 0。

9.2.5.14. 0x0034 XSPI_ADDR

默认值：0x00000000			xSPI 地址配置寄存器（XSPI Address）
位域	类型	默认值	描述
31:0	R/W	0x0	ADDR 此字段为 AHB 通道配置外设的访问地址。

9.2.5.15. 0x0038 XSPI_FMR

默认值：0x00000000			xSPI 格式配置寄存器（xSPI Format Register）
位域	类型	默认值	描述
31:24	R/W	0x0	FORMAT_CMD
23:16	R/W	0x0	FORMAT_CMD_EX
15:3	–	–	–
2:0	R/WAC	0x0	FORMAT_SEL • 0：LUT 配置，AHB/ AXI 读写访问 • 1：COMMAND，1S • 2：COMMAND + 3-byte Address，1S-1S

Copyright © 2023-2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			xSPI 格式配置寄存器（xSPI Format Register）
位域	类型	默认值	描述
			• 3：COMMAND + 4-byte Address，1S-1S • 4：COMMAND，8D • 5：COMMAND + COMMAND_EX，8D • 6：COMMAND + COMMAND_EX + 4-byte Address，8D-8D 此位配 1-6，则直接触发相应描述行为，无需配置 LUT，此时 CMD 以及 CMD_EX 在 bit[31:16] 配置，Address 在 XSPI_ADDR 寄存器配置。 主要应用简答典型操作，如外部设备 reset 命令，擦除命令等。传输结束后自动清零。

9.2.5.16. 0x0040 XSPI_BTR

默认值：0x00004010			xSPI 突发类型寄存器（XSPI Burst Type Register）
位域	类型	默认值	描述
31:16	–	–	–
15:8	R/W	0x40	SPI_BURST_WRAPPED SPI 模式 XIP 使能时，若 AXI Burst Type 是 Linear 切换 Wrapped，触发此命令发送。
7:0	R/W	0x10	SPI_BURST_LINEAR SPI 模式 XIP 使能时，若 AXI Burst Type 是 Wrapped 切换 Linear，触发此命令发送。

9.2.5.17. 0x0044 XSPI_RCC

默认值：0x00002000			xSPI 读命令控制（xSPI Read Command Control）
位域	类型	默认值	描述
31:18	–	–	–
17	R/W	0x0	READ_MODE_BYTE_EN SPI 读命令模式位使能，当该位使能，地址后面增加 1 byte 模式位，模式位的线宽与地址位一致，模式位的设定值由 RDCMD_BYPASS_EN 决定。 **注：** SPI 工作在 Dual I/O，Quad I/O，QPI，DTR Quad I/O 模式下时，用户必须配置该位使能。
16	R/W	0x0	RDCMD_BYPASS_EN

Copyright © 2023-2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00002000			xSPI 读命令控制 （xSPI Read Command Control）
位域	类型	默认值	描述
			SPI 读命令 Bypass 模式使能，当该位使能，模式位的设定值为 RDCMD_BYPPASS_CODE，否则模式位的设定值为 RDCMD_NORMAL_CODE。
15:8	R/W	0x20	RDCMD_BYPASS_CODE 该位域定义读命令 Bypass 模式下的设定值，之后的读操作不需要再次发送读命令。
7:0	R/W	0x00	RDCMD_NORMAL_CODE 该位域定义正常模式下的设定值，每次读操作都需要发送读命令。

9.2.5.18. 0x0048 XSPI_NDMA_MODE_CTL

默认值：0x000000E5			xSPI DMA 模式控制 （xSPI DMA Mode Control）
位域	类型	默认值	描述
31:8	–	–	–
7:6	R/W	0x3	• 00：dma_active 信号为低 • 01：dma_active 信号为高 • 10：dma_active 信号由 DMA 请求控制 • 11：dma_active 信号由控制器控制
5	R/W	0x1	• 0：active fall do not care ack • 1：active fall must after detect ack is high
4:0	R/W	0x05	延迟时钟数，从 DMA 最后一笔信号为高到 dma_active 信号为高保持的时钟数。

9.2.5.19. 0x0050 XSPI_TO

默认值：0xffffffff			xSPI 超时配置 （XSPI Timeout）
位域	类型	默认值	描述
31:0	R/W	0xffff_ffff	TIMEOUT 超时计数器单位为 XSPI_CLK

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.2.5.20. 0x0054 XSPI_LCKCR

默认值：0x00000002			xSPI 锁定配置（xSPI LOCK Configuration）
位域	类型	默认值	描述
31:2	–	–	–
1:0	R/W	0x2	LOCK_CFG LUT 锁定配置 • 0x1：锁定 • 0x2：解锁 锁定时不可配置查找表。

9.2.5.21. 0x0058 XSPI_LUT_UP

默认值：0x00000000			xSPI 查找表更新（xSPI Look Up Table Update）
位域	类型	默认值	描述
31:1	–	–	–
0	R/WAC	0x0	LUT_UP 写 1 将当前 LUT 读写控制信息更新至 AXI 通道传输使用，节省硬件查找 LUT 时间，完成后自动清 0。 **注：** 需在 XIP_EN 配 1 之前完成 UPDATE。

9.2.5.22. 0x0060 XSPI_CS0_SEQUENCE

默认值：0x76543210			xSPI 通道 0 线序（XSPI Chip Select 0 Sequence）
位域	类型	默认值	描述
31	–	–	–
30:28	R/W	0x7	D7_SELD 7 位数据选择
27	–	–	–
26:24	R/W	0x6	D6_SELD

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x76543210			xSPI 通道 0 线序（XSPI Chip Select 0 Sequence）
位域	**类型**	**默认值**	**描述**
			6 位数据选择
23	–	–	–
22:20	R/W	0x5	D5_SELD 5 位数据选择
19	–	–	–
18:16	R/W	0x4	D4_SELD 4 位数据选择
15	–	–	–
14:12	R/W	0x3	D3_SELD 3 位数据选择
11	–	–	–
10:8	R/W	0x2	D2_SELD 2 位数据选择
7	–	–	–
6:4	R/W	0x1	D1_SELD 1 位数据选择
3	–	–	–
2:0	R/W	0x0	D0_SELD 0 位数据选择 • 0：D0 • 1：D1 • 2：D2 • 3：D3 • 4：D4 • 5：D5 • 6：D6 • 7：D7

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.2.5.23. 0x0064 XSPI_CS1_SEQUENCE

默认值：0x76543210			xSPI 通道 1 线序（xSPI Chip Select 1 Sequence）
位域	类型	默认值	描述
31	–	–	–
30:28	R/W	0x7	D7_SELD 7 位数据选择
27	–	–	–
26:24	R/W	0x6	D6_SELD 6 位数据选择
23	–	–	–
22:20	R/W	0x5	D5_SELD 5 位数据选择
19	–	–	–
18:16	R/W	0x4	D4_SELD 4 位数据选择
15	–	–	–
14:12	R/W	0x3	D3_SELD 3 位数据选择
11	–	–	–
10:8	R/W	0x2	D2_SELD 2 位数据选择
7	–	–	–
6:4	R/W	0x1	D1_SELD 1 位数据选择
3	–	–	–
2:0	R/W	0x0	D0_SELD • 0：D0 • 1：D1

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x76543210			xSPI 通道 1 线序（xSPI Chip Select 1 Sequence）
位域	类型	默认值	描述
			• 2：D2 • 3：D3 • 4：D4 • 5：D5 • 6：D6 • 7：D7

9.2.5.24. 0x0068 XSPI_IO_CTL

默认值：0x00000000			xSPI IO 控制（xSPI IO Control）
位域	类型	默认值	描述
312	–	–	–
1	R/W	0x0	CS1_IO_CFG • 0：打开 • 1：关闭
0	R/W	0x0	CS0_IO_CFG • 0：打开 • 1：关闭

9.2.5.25. 0x0070 XSPI_CS0_IOCFG1

默认值：0x00000000			xSPI 通道 0 IO 配置 1（xSPI Chip Select 0 IO Configuration1）
位域	类型	默认值	描述
31:30	–	–	–
29:28	R/W	0x0	D7_PIN_PULL 引脚上下拉设置
27	–	–	–
26:24	R/W	0x0	D7_PIN_DRV 引脚输出驱动能力

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			xSPI 通道 0 IO 配置 1（xSPI Chip Select 0 IO Configuration1）
位域	类型	默认值	描述
23:22	–	–	–
21:20	R/W	0x0	D6_PIN_PULL 引脚上下拉设置
19	–	–	–
18:16	R/W	0x0	D6_PIN_DRV 引脚输出驱动能力
15:14	–	–	–
13:12	R/W	0x0	D5_PIN_PULL 引脚上下拉设置
11	–	–	–
10:8	R/W	0x0	D5_PIN_DRV 引脚输出驱动能力
7:6	–	–	–
5:4	R/W	0x0	D4_PIN_PULL 引脚上下拉设置 • 0：关闭 • 1：预留 • 2：下拉 33KΩ 到 GND • 3：上拉 33KΩ 到 VCC-IO
3	–	–	–
2:0	R/W	0x0	D4_PIN_DRV 引脚输出驱动能力

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.2.5.26. 0x0074 XSPI_CS0_IOCFG2

默认值：0x00000000			xSPI 通道 0 IO 配置 2（xSPI Chip Select 0 IO Configuration2）
位域	类型	默认值	描述
31:30	–	–	–
29:28	R/W	0x0	D3_PIN_PULL 引脚上下拉设置
27	–	–	–
26:24	R/W	0x0	D3_PIN_DRV 引脚输出驱动能力
23:22	–	–	–
21:20	R/W	0x0	D2_PIN_PULL 引脚上下拉设置
19	–	–	–
18:16	R/W	0x0	D2_PIN_DRV 引脚输出驱动能力
15:14	–	–	–
13:12	R/W	0x0	D1_PIN_PULL 引脚上下拉设置
11	–	–	–
10:8	R/W	0x0	D1_PIN_DRV 引脚输出驱动能力
7:6	–	–	–
5:4	R/W	0x0	D0_PIN_PULL 引脚上下拉设置 • 0：关闭 • 1：预留 • 2：下拉 33KΩ 到 GND • 3：上拉 33KΩ 到 VCC-IO

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			xSPI 通道 0 IO 配置 2（xSPI Chip Select 0 IO Configuration2）
位域	类型	默认值	描述
3	–	–	–
2:0	R/W	0x0	D0_PIN_DRV 引脚输出驱动能力

9.2.5.27. 0x0078 XSPI_CS0_IOCFG3

默认值：0x30000000			xSPI 通道 0 IO 配置 3（xSPI Chip Select 0 IO Configuration3）
位域	类型	默认值	描述
31:30	–	–	–
29:28	R/W	0x3	CS_PIN_PULL 引脚上下拉设置
27	–	–	–
26:24	R/W	0x0	CS_PIN_DRV 引脚输出驱动能力
23:22	–	–	–
21:20	R/W	0x0	DQS_PIN_PULL 引脚上下拉设置
19	–	–	–
18:16	R/W	0x0	DQS_PIN_DRV 引脚输出驱动能力
15:14	–	–	–
13:12	R/W	0x0	CK_PIN_PULL 引脚上下拉设置
11	–	–	–
10:8	R/W	0x0	CK_PIN_DRV 引脚输出驱动能力
7:6	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x30000000			xSPI 通道 0 IO 配置 3（xSPI Chip Select 0 IO Configuration3）
位域	类型	默认值	描述
5:4	R/W	0x0	CKN_PIN_PULL 引脚上下拉设置 • 0：关闭 • 1：预留 • 2：下拉 33KΩ 到 GND • 3：上拉 33KΩ 到 VCC–IO
3	–	–	–
2:0	R/W	0x0	CKN_PIN_DRV 引脚输出驱动能力

9.2.5.28. 0x007C XSPI_CS0_IOCFG4

默认值：0x00000000			XSPI 通道 0 IO 配置 4（XSPI Chip Select 0 IO Configuration4）
位域	类型	默认值	描述
31:6	–	–	–
5:4	R/W	0x0	DM_PIN_PULL 引脚上下拉设置 • 0：关闭 • 1：预留 • 2：下拉 33KΩ 到 GND • 3：上拉 33KΩ 到 VCC–IO
3	–	–	–
2:0	R/W	0x0	DM_PIN_DRV 引脚输出驱动能力

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.2.5.29. 0x0080 XSPI_CS1_IOCFG1

默认值：0x00000000			XSPI 通道 1 IO 配置 1（XSPI Chip Select 1 IO Configuration1）
位域	类型	默认值	描述
31:30	–	–	–
29:28	R/W	0x0	D7_PIN_PULL 引脚上下拉设置
27	–	–	–
26:24	R/W	0x0	D7_PIN_DRV 引脚输出驱动能力
23:22	–	–	–
21:20	R/W	0x0	D6_PIN_PULL 引脚上下拉设置
19	–	–	–
18:16	R/W	0x0	D6_PIN_DRV 引脚输出驱动能力
15:14	–	–	–
13:12	R/W	0x0	D5_PIN_PULL 引脚上下拉设置
11	–	–	–
10:8	R/W	0x0	D5_PIN_DRV 引脚输出驱动能力
7:6	–	–	–
5:4	R/W	0x0	D4_PIN_PULL 引脚上下拉设置 • 0：关闭 • 1：预留 • 2：下拉 33KΩ 到 GND • 3：上拉 33KΩ 到 VCC-IO

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			XSPI 通道 1 IO 配置 1（XSPI Chip Select 1 IO Configuration1）
位域	类型	默认值	描述
3	–	–	–
2:0	R/W	0x0	D4_PIN_DRV 引脚输出驱动能力

9.2.5.30. 0x0084 XSPI_CS1_IOCFG2

默认值：0x00000000			XSPI 通道 1 IO 配置 2（XSPI Chip Select 1 IO Configuration2）
位域	类型	默认值	描述
31:30	–	–	–
29:28	R/W	0x0	D3_PIN_PULL 引脚上下拉设置
27	–	–	–
26:24	R/W	0x0	D3_PIN_DRV 引脚输出驱动能力
23:22	–	–	–
21:20	R/W	0x0	D2_PIN_PULL 引脚上下拉设置
19	–	–	–
18:16	R/W	0x0	D2_PIN_DRV 引脚输出驱动能力
15:14	–	–	–
13:12	R/W	0x0	D1_PIN_PULL 引脚上下拉设置
11	–	–	–
10:8	R/W	0x0	D1_PIN_DRV 引脚输出驱动能力
7:6	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			XSPI 通道 1 IO 配置 2（XSPI Chip Select 1 IO Configuration2）
位域	类型	默认值	描述
5:4	R/W	0x0	D0_PIN_PULL 引脚上下拉设置 • 0：关闭 • 1：预留 • 2：下拉 33KΩ 到 GND • 3：上拉 33KΩ 到 VCC-IO
3	–	–	–
2:0	R/W	0x0	D0_PIN_DRV 引脚输出驱动能力

9.2.5.31. 0x0088 XSPI_CS1_IOCFG3

默认值：0x30000000			XSPI 通道 1 IO 配置 3（XSPI Chip Select 1 IO Configuration3）
位域	类型	默认值	描述
31:30	–	–	–
29:28	R/W	0x3	CS_PIN_PULL 引脚上下拉设置
27	–	–	–
26:24	R/W	0x0	CS_PIN_DRV 引脚输出驱动能力
23:22	–	–	–
21:20	R/W	0x0	DQS_PIN_PULL 引脚上下拉设置
19	–	–	–
18:16	R/W	0x0	DQS_PIN_DRV 引脚输出驱动能力
15:14	–	–	–
13:12	R/W	0x0	CK_PIN_PULL

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x30000000			XSPI 通道 1 IO 配置 3（XSPI Chip Select 1 IO Configuration3）
位域	类型	默认值	描述
			引脚上下拉设置
11	–	–	–
10:8	R/W	0x0	CK_PIN_DRV 引脚输出驱动能力
7:6	–	–	–
5:4	R/W	0x0	CKN_PIN_PULL 引脚上下拉设置 • 0：关闭 • 1：预留 • 2：下拉 33KΩ 到 GND • 3：上拉 33KΩ 到 VCC-IO
3	–	–	–
2:0	R/W	0x0	CKN_PIN_DRV 引脚输出驱动能力

9.2.5.32. 0x008C XSPI_CS1_IOCFG4

默认值：0x00000000			XSPI 通道 1 IO 配置 4（XSPI Chip Select 1 IO Configuration4）
位域	类型	默认值	描述
31:6	–	–	–
5:4	R/W	0x0	DM_PIN_PULL 引脚上下拉设置 • 0：关闭 • 1：预留 • 2：下拉 33KΩ 到 GND • 3：上拉 33KΩ 到 VCC-IO
3	–	–	–
2:0	R/W	0x0	DM_PIN_DRV

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			XSPI 通道 1 IO 配置 4（XSPI Chip Select 1 IO Configuration4）
位域	类型	默认值	描述
			引脚输出驱动能力

9.2.5.33. 0x0090 XSPI_TRAINING_CFG

默认值：0x00000000			XSPI TRAINIING 配置（XSPI Training Configuration）
位域	类型	默认值	描述
31:25	–	–	–
24	R/W	0x0	TRAINING_PHASE_CAL training 结果计算 • 0：(temp1 + temp2)/2, 向下取整 • 1：((temp1 + temp2) – 1) / 2 + 1, 向上取整
23：20	–	–	–
19:16	R/W	0x0	TRAINING_PATTERN_SEL • 0：stuck address • 1：随机数 • 2：高低电平反转 • 3：自定义
15:0	R/W	0x0	DATA_LEN training 数据长度，单位 bytes

9.2.5.34. 0x0094 XSPI_TRAINING_PATTERN

默认值：0x00000000			XSPI TRAINIING PATTERN（XSPI Training Pattern）
位域	类型	默认值	描述
31:0	R/W	0x0	TRAINING_PATTERN TRAINING_PATTERN_SEL 选择 3 时，此字段可设置 training 数据内容

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.2.5.35. 0x0100 XSPI_LUTn

默认值：0x00000000			xSPI 查找表 n（XSPI Look Up Table n,n=0~31）
位域	**类型**	**默认值**	**描述**
31:26	R/W	0x0	Instruction1 指令
25:24	R/W	0x0	IO CFG Instruction1 的 IO 配置信息 • 0x0：1 个 IO • 0x1：2 个 IO • 0x2：4 个 IO • 0x3：8 个 IO
23:16	R/W	0x0	OPERAND Instruction1 的操作数配置
15:10	R/W	0x0	Instruction0 指令
9:8	R/W	0x0	IO CFG Instruction0 的 IO 配置信息 • 0x0：1 个 IO • 0x1：2 个 IO • 0x2：4 个 IO • 0x3：8 个 IO
7:0	R/W	0x0	OPERAND Instruction0 的操作数配置

9.2.5.36. 0x0200 XSPI_TDR

默认值：0x00000000			xSPI 发送数据寄存器（xSPI Transmit Data Register）
位域	**类型**	**默认值**	**描述**
31:0	W	0x0	TDR 发送数据 该寄存器可以通过 AHB 总线以字节，半字或字为单位进行访问。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			xSPI 发送数据寄存器（xSPI Transmit Data Register）
位域	类型	默认值	描述
			• 以字节访问时，每次写一笔数据到 TXFIFO，FIFO 深度加 1。 • 以半字访问时，每次写两笔数据到 TXFIFO，FIFO 深度加 2。 • 以字为单位访问时，每次写四笔数据到 TXFIFO，FIFO 深度加 4。

9.2.5.37. 0x0300 XSPI_RDR

默认值：0x00000000			xSPI 接收数据寄存器（xSPI Recieve Data Register）
位域	类型	默认值	描述
31:0	R	0x0	RDR 接收数据 该寄存器可以通过 AHB 总线以字节，半字或字为单位进行访问。 • 以字节访问时，每次从 RXFIFO 读一笔数据到寄存器，FIFO 深度减 1 • 以半字访问时，每次从 RXFIFO 读两笔数据，FIFO 深度减 2 • 以字为单位访问时，每次从 RXFIFO 读四笔数据，FIFO 深度减 4

9.2.5.38. 0x0400 XSPI_DEBUG

默认值：0x00000000			xSPI 调试寄存器（XSPI Debug）
位域	类型	默认值	描述
31:0	R	0x0	DEBUG

9.2.5.39. 0x0404 XSPI_DEBUG_SEL

默认值：0x00000000			xSPI 调试选择寄存器（xSPI Debug Select）
位域	类型	默认值	描述
31:4	–	–	–
3:0	R/W	0x0	DEBUG_SEL

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

9.2.5.40. 0xFFC VERSION

默认值：0x00000101			xSPI 版本寄存器（xSPI Version）
位域	类型	默认值	描述
31:0	RO	0x00000101	Version 模块版本号，采用 BCD 码显示，V1.1。

ArtInChip

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10. 多媒体

10.1. Display Engine (DE)

显示引擎 (Display Engine) 主要用来提供显示图层叠加、颜色空间转换、图像缩放、图像后处理等功能。 可以实现多个图层输入和多个显示接口输出，多个图层资源可以灵活的分配给不同的显示接口。

10.1.1. 特性说明

- 最高性能 1024*768@60fps
- 支持一个 UI 图层，UI 图层内支持四个矩形窗口
- 支持图像格式：ARGB8888/ XRGB8888/ RGB888/ ARGB4444/ ARGB1555/ RGB565
- 支持色彩矫正矩阵 (CCM)
- 支持 Gamma 矫正
- 支持误差扩散 Dither

10.1.2. 原理框图

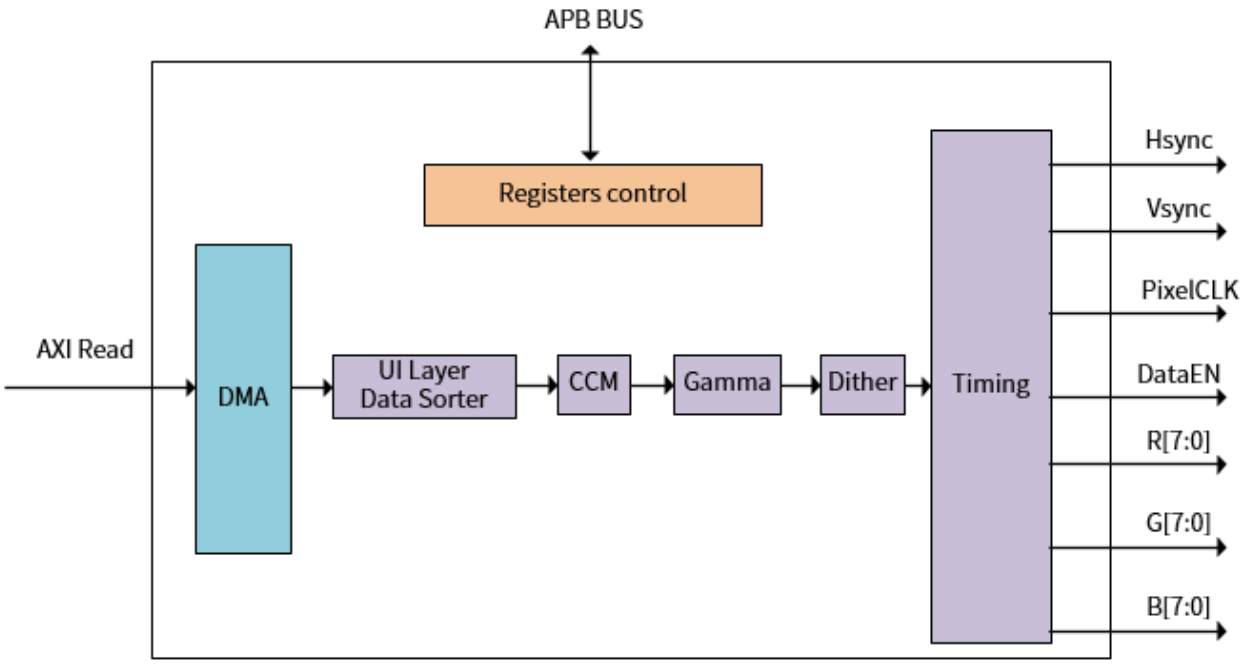


图 10–1 整体框图

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.1.3. 功能描述

10.1.3.1. 影子寄存器 (Shadow Register)

为了实现一组配置同时生效，DE 的大部分寄存器都是对应两个寄存器。

- Preload register：软件可以操作的寄存器，用于存储配置参数。
- Shadow register：软件不可见，由硬件操作，用于控制硬件行为。
- CONFIG_UPDATE：设置该标志位为 1，可以将 Preload 寄存器的值同时更新到 Shadow 寄存器。

除了 CRC 校验、CCM 和 Gamma 矫正、QoS、Timing 相关寄存器和 DE 配置更新寄存器之外， 其它的寄存器都有 shadow register。

10.1.3.2. 图层控制

10.1.3.2.1. UI 图层

UI 图层功能如下：

- **多窗口支持**：有四个矩形窗口，可单独控制开关。
- **位置控制**：各窗口以图层左上角为原点设置坐标，控制显示位置。
- **颜色格式一致性**：所有窗口的颜色格式必须一致。
- **图像裁剪**：窗口可显示原始图像的裁剪区域，软件需计算矩形区域的起始地址。
- **非重叠约束**：四个矩形窗口不可重叠，需进行有效性检查。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

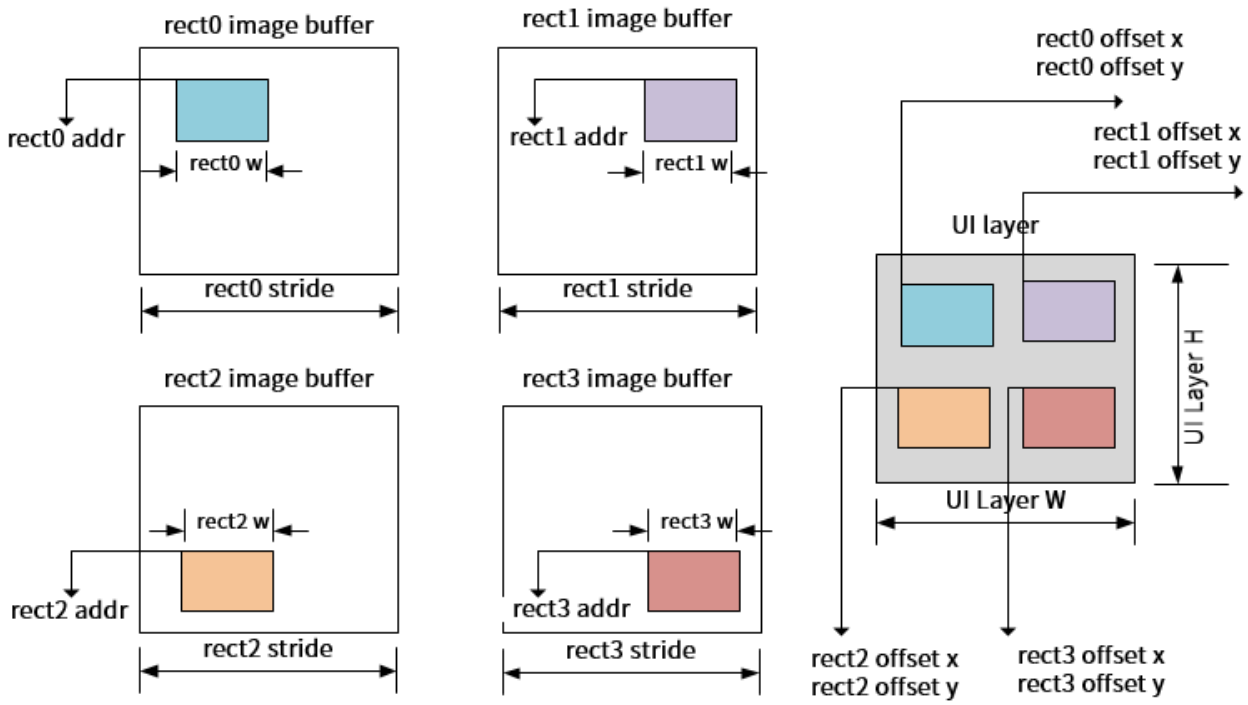


图 10–2 UI 图层

10.1.3.3. 图层混合

10.1.3.4. 色彩矩阵 (CCM)

色彩矩阵 (Color Correction Matrix，CCM) 用于对图像的色彩进行校正。

色彩矩阵（CCM）组成如下：

- **增益矩阵**：一个 3x3 的增益矩阵，用于调整图像的色彩增益。
- **偏移矩阵**：一个 3x1 的偏移矩阵，用于对图像的色彩进行偏移调整。

CCM 矩阵如下公式所示。

$$\begin{bmatrix} R_{out} \\ G_{out} \\ B_{out} \end{bmatrix} = \left(\left(\begin{bmatrix} C_{00} & C_{01} & C_{02} \\ C_{10} & C_{11} & C_{12} \\ C_{20} & C_{21} & C_{22} \end{bmatrix} \times \begin{bmatrix} R_{in} \\ G_{in} \\ B_{in} \end{bmatrix} + 128 \right) \gg 8 \right) + \begin{bmatrix} Offset_0 \\ Offset_1 \\ Offset_2 \end{bmatrix}$$

图 10–3 CCM 公式

$$\begin{bmatrix} R_{out} \\ G_{out} \\ B_{out} \end{bmatrix} = \mathrm{Clip}\left(\begin{bmatrix} R_{out} \\ G_{out} \\ B_{out} \end{bmatrix}, 0, (1 \ll Nbit) - 1 \right)$$

图 10–4 CCM 输出范围限制

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- Rin, Gin, Bin：输入的红色、绿色、蓝色分量。
- Rout, Gout, Bout：输出的红色、绿色、蓝色分量。
- Nbit: R，G，B 输入输出，位宽为 8
- Cij：增益矩阵的系数，为 12 位有符号数，其中包含 8 位小数。
- Offset：偏移矩阵的值，为 9 位有符号整数。

10.1.3.5. Gamma 矫正

Gamma 矫正用于调整图像的对比度和亮度，使图像在不同设备上显示一致。在视频编码和解码中进行 Gamma 校正，优化视觉效果。

Gamma 矫正通过查表对 R、G、B 三个通道分别独立进行矫正，输入数据位宽为 8 bits，LUT 长度为 64。硬件通过线性插值计算两个节点之间的 Gamma 矫正结果。

配置 Gamma 矫正：

- **设置 LUT**：通过寄存器配置每个通道的 LUT 值。
- **使能 Gamma 矫正**：GAMMA_EN 位写 1 启用 Gamma 矫正功能。

10.1.3.6. 限制条件

- 支持屏幕最小 size 为 64x64
- 模块时钟最高可到 100 MHz
- Pixel clk 要小于 DE 模块时钟
- 当开启 Dither 功能时，屏幕宽度最大到 1024，单位为 pixel
- UI 图层矩形窗支持的最小 size：2x2，单位为 pixel
- 所有格式的 stride 都要求 8 bytes 对齐

10.1.4. 寄存器列表

表 10–1 DE 寄存器列表

偏移	缩写	寄存器名称	章节
0x000	DE_CTRL	DE 控制	0x000 DE_CTRL
0x004	DE_MODE_SELECT	DE 模式选择	0x004 DE_MODE_SELECT
0x008	DE_CONFIG_UPDATE	DE 配置更新	0x008 DE_CONFIG_UPDATE
0x00C	OUTPUT_COLOR_DEPTH	输出颜色深度	0x00C OUTPUT_COLOR_DEPTH
0x010	DE_VER_ID	DE 版本	0x010 DE_VER_ID

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 10–1 DE 寄存器列表 (续)

偏移	缩写	寄存器名称	章节
0x014	DITHER_RAND_SEED	Dither 随机种子	0x014 DITHER_RAND_SEED
0x018	DITHER_RAND_MASK_BITS	Dither 随机掩码	0x018 DITHER_RAND_MASK_BITS
0x0A0	UI_LAYER_CTRL	UI 图层控制	0x0A0 UI_LAYER_CTRL
0x0A4	UI_LAYER_INPUT_SIZE	UI 图层宽高	0x0A4 UI_LAYER_INPUT_SIZE
0x0A8	UI_LAYER_BG_COLOR	UI 图层背景色	0x0A8 UI_LAYER_BG_COLOR
0x0B0	UI_LAYER_RECT_CTRL	UI 矩形区控制	0x0B0 UI_LAYER_RECT_CTRL
0x0B4	UI_LAYER_PALETTE_ADDR	UI 图层调色板地址	0x0B4 UI_LAYER_PALETTE_ADDR
0x0C0	UI_RECT_INPUT_SIZE	UI 矩形区输入宽高	0x0C0+N*0x10(N=0:3) UI_RECT_INPUT_SIZE
0x0C4	UI_RECT_OFFSET	UI 矩形区偏移	0x0C4+N*0x10(N=0:3) UI_RECT_OFFSET
0x0C8	UI_RECT_STRIDE	UI 矩形区行对齐	0x0C8+N*0x10(N=0:3) UI_RECT_STRIDE
0x0CC	UI_RECT_ADDR	UI 矩形区地址	0x0CC+N*0x10(N=0:3) UI_RECT_ADDR
0x170	CRC_CTRL	回写控制	0x170 CRC_CTRL
0x194	CRC_CHECK	CRC 校验	0x194 CRC_CHECK
0x198	INPUT_ID	输入 ID	0x198 INPUT_ID
0x19C	OUTPUT_ID	输出 ID	0x19C OUTPUT_ID
0x1D0	TIMING_CTRL	Timing 控制	0x1D0 TIMING_CTRL
0x1D4	TIMING_INT	Timing 中断	0x1D4 TIMING_INT
0x1D8	TIMING_INT_STATUS	Timing 中断状态	0x1D8 TIMING_INT_STATUS
0x1DC	TIMING_LINE_SET	Timing 行数设置	0x1DC TIMING_LINE_SET
0x1E0	TIMING_ACTIVE_SIZE	Timing 有效宽高	0x1E0 TIMING_ACTIVE_SIZE
0x1E4	TIMING_H_PORCH	Timing Horizontal Porch	0x1E4 TIMING_H_PORCH
0x1E8	TIMING_V_PORCH	Timing Vertical Porch	0x1E8 TIMING_V_PORCH

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 10–1 DE 寄存器列表 (续)

偏移	缩写	寄存器名称	章节
0x1EC	TIMING_SYNC_PLUSE	Timing 同步脉冲	0x1EC TIMING_SYNC_PULSE
0x1F0	TIMING_POL_SET	Timing 极性配置	0x1F0 TIMING_POL_SET
0x1F4	TIMING_DEBUG	Timing 调试	0x1F4 TIMING_DEBUG
0x88C	QOS_UI	QOS UI	0x88C QOS_UI
0x890	QOS_URGENT	QOS Urgent	0x890 QOS_URGENT
0x900	CCM_C00	CCM 系数 C00	0x900 CCM_C00
0x904	CCM_C01	CCM 系数 C01	0x904 CCM_C01
0x908	CCM_C02	CCM 系数 C02	0x908 CCM_C02
0x90C	CCM_OFFSET0	CCM 偏移 OFFSET0	0x90C CCM_OFFSET0
0x910	CCM_C10	CCM 系数 C10	0x910 CCM_C10
0x914	CCM_C11	CCM 系数 C11	0x914 CCM_C11
0x918	CCM_C12	CCM 系数 C12	0x918 CCM_C12
0x91C	CCM_OFFSET1	CCM 偏移 OFFSET1	0x91C CCM_OFFSET1
0x920	CCM_C20	CCM 系数 C20	0x920 CCM_C20
0x924	CCM_C21	CCM 系数 C21	0x924 CCM_C21
0x928	CCM_C22	CCM 系数 C22	0x928 CCM_C22
0x92C	CCM_OFFSET2	CCM 偏移 OFFSET2	0x92C CCM_OFFSET2
0x940	GAMMA_RED_LUT	Gamma 红色分量查找表	0x940+4*N(N=0:15) GAMMA_RED_LUT
0x980	GAMMA_GREEN_LUT	Gamma 绿色分量查找表	0x980+4*N(N=0:15) GAMMA_GREEN_LUT
0x9C0	GAMMA_BLUE_LUT	Gamma 绿色分量查找表	0x9C0+4*N(N=0:15) GAMMA_BLUE_LUT

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.1.5. 寄存器描述

10.1.5.1. 0x000 DE_CTRL

默认值：0x00000000			DE 控制 （DE Control）
位域	类型	默认值	描述
31:1	–	–	–
2	R/W	0x0	CCM_CTRLCCM 开关控制 • 0：关闭 • 1：开启
1	R/W	0x0	GAMMA_CTRLGamma 开关控制 • 0：关闭 • 1：开启
0	R/W	0x0	DITHER_ENDither 开关控制 • 0：关闭 Dither • 1：打开 Dither 当颜色输出深度寄存器设置的输出小于 8 bits 的时候打开 Dither 功能，Dither 功能有效。

10.1.5.2. 0x004 DE_MODE_SELECT

默认值：0x00000000			DE 模式选择 （DE Mode Select）
位域	类型	默认值	描述
31:11	–	–	–
10:8	R/W	0x0	COLOR_BAR_MODE • 0：color bar 第一个竖条颜色为 0x000000 • 1：color bar 第一个竖条颜色为 0x0000FF • 2：color bar 第一个竖条颜色为 0x00FF00 • 3：color bar 第一个竖条颜色为 0x00FFFF • 4：color bar 第一个竖条颜色为 0xFF0000 • 5：color bar 第一个竖条颜色为 0xFF00FF • 6：color bar 第一个竖条颜色为 0xFFFF00 • 7：color bar 第一个竖条颜色为 0xFFFFFF 当打开 color bar 模式时，可以选择 color bar 第一个竖条的颜色值。
7:1	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			DE 模式选择 （DE Mode Select）
位域	类型	默认值	描述
0	R/W	0x0	MODE_SELECT 模式选择 • 0：nomal mode 正常模式 • 1：color bar 彩色条纹模式 Nomal mode 模式是 DE 的默认处理模式，从 memory 读取数据，完成图层叠加后，然后通过 timing 接口输出。Color bar 模式不需要读取外部 memory 的数据，由 DE 内部生成一幅 RGB 的彩色条纹图像，通过 timing 接口或者回写接口输出，主要用来 debug。

10.1.5.3. 0x008 DE_CONFIG_UPDATE

默认值：0x00000000			DE 配置更新 （DE Config Update）
位域	类型	默认值	描述
31:1	–	–	–
0	R/W	0x0	CONFIG_UPDATE 配置更新控制 • 0：配置不更新 • 1：配置准备好更新 软件写 1 后，表示 preload register 配置信息已经准备好，允许硬件更新 preload register 配置信息到 shadow register。当硬件完成更新后，自动清零。当软件写 1 后，软件也可以在任意时刻写 0，由硬件来保证对此控制阈的操作是原子操作

10.1.5.4. 0x00C OUTPUT_COLOR_DEPTH

默认值：0x00000000			输出颜色深度 （Output Color Depth）
位域	类型	默认值	描述
31:18	–	–	–
17:16	R/W	0x0	R_DEPTHR 分量颜色深度 • 0：8 bits • 1：6 bits • 2：5 bits • 3：保留 当输出为 6 bits 或者 5 bits 时，可以打开 Dither 功能，使图像更平滑，否则直接丢弃低位 bit(s)。
15:10	–	–	–
9:8	R/W	0x0	G_DEPTHG 分量颜色深度 • 0：8 bit • 1：6 bit

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			输出颜色深度（Output Color Depth）
位域	类型	默认值	描述
			• 2：5 bit • 3：保留 当输出为 6 bits 或者 5 bits 时，可以打开 Dither 功能，使图像更平滑，否则直接丢弃低位 bit(s)。
7:2	–	–	–
1:0	R/W	0x0	B_DEPTHB 分量颜色深度 • 0：8 bit • 1：6 bit • 2：5 bit • 3：保留 当输出为 6 bits 或者 5 bits 时，可以打开 Dither 功能，使图像更平滑，否则直接丢弃低位 bit(s)。

10.1.5.5. 0x010 DE_VER_ID

默认值：0x00000102			DE 版本（DE Version ID）
位域	类型	默认值	描述
31:16	R0	–	DE_PATCH_ID 修订号
15:0	RO	0x00000102	DE_VER_ID[15:8]为主版本号，[7:0]为子版本号，V1.2

10.1.5.6. 0x014 DITHER_RAND_SEED

默认值：0x00000100			Dither 随机种子（Dither Rand Seed）
位域	类型	默认值	描述
31	R/W	0x0	RAND_DITHER_EN • 0：随机 Dither 关闭，RAND_DITHER_EN 无效 • 1：随机 Dither 打开，即打开随机 Dither 功能，误差扩散 Dither 的模板随机选择。 当 DITHER_EN 为 1 的时候，可以选择打开随机 Dither 功能，当选择随机 Dither 的时候，误差扩散 Dither 的模板随机选择。当 EN 为 0 的时候，RAND_DITHER_EN 无效。
30:28	–	–	–
27	R/W	0x0	RAND_TEMPORAL_DITHER_EN • 0：时域随机 Dither 关闭 • 1：时域随机 Dither 打开 当 RAND_DITHER_EN 关闭时，RAND_TEMPORAL_DITHER_EN 无效。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000100			Dither 随机种子（Dither Rand Seed）
位域	类型	默认值	描述
26:24	–	–	–
23:0	R/W	0x100	RAND_DITHER_SEED 当选择随机 Dither 时，需要设置位宽为 24 bits 的随机种子。

10.1.5.7. 0x018 DITHER_RAND_MASK_BITS

默认值：0x00E10000			Dither 随机掩码（Dither Rand Mask Bits）
位域	类型	默认值	描述
31:24	–	–	–
23:0	R/W	0xE10000	DITHER_MASK_BITS 当选择随机 Dither 时，需要设置位宽为 24 bits 的 Bits。

10.1.5.8. 0x0A0 UI_LAYER_CTRL

默认值：0x00000000			UI 图层控制（UI Layer Control）
位域	类型	默认值	描述
31:15	–	–	–
14:8	R/W	0x0	INPUT_FORMAT 像素格式 • 0x00: ARGB 8–8–8–8 • 0x01: ABGR 8–8–8–8 • 0x02: RGBA 8–8–8–8 • 0x03: BGRA 8–8–8–8 • 0x04: XRGB 8–8–8–8 • 0x05: XBGR 8–8–8–8 • 0x06: RGBX 8–8–8–8 • 0x07: BGRX 8–8–8–8 • 0x08: RGB 8–8–8 • 0x09: BGR 8–8–8 • 0x0A: ARGB 1–5–5–5 • 0x0B: ABGR 1–5–5–5 • 0x0C: RGBA 5–5–5–1 • 0x0D: BGRA 5–5–5–1 • 0x0E: RGB 5–6–5 • 0x0F: BGR 5–6–5 • 0x10: ARGB 4–4–4–4 • 0x11: ABGR 4–4–4–4

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			UI 图层控制（UI Layer Control）
位域	类型	默认值	描述
			• 0x12: RGBA 4–4–4–4 • 0x13: BGRA 4–4–4–4 • 0x14: PALETTE_8BITS
7:1	–	–	–
0	R/W	0x0	EN 图层使能控制 • 0：关闭 • 1：开启

10.1.5.9. 0x0A4 UI_LAYER_INPUT_SIZE

默认值：0x00000000			UI 图层宽高（UI Layer Size）
位域	类型	默认值	描述
31:29	–	–	–
28:16	R/W	0x0	LAYER_HEIGHT 图层输入高度
15:13	–	–	–
12:0	R/W	0x0	LAYER _WIDTH 图层输入宽度

10.1.5.10. 0x0A8 UI_LAYER_BG_COLOR

默认值：0x00000000			UI 图层背景色（UI Layer Backgroud Color）
位域	类型	默认值	描述
31:24	R/W	0x0	ALPHA_VALUE 透明度分量
23:16	R/W	0x0	R_VALUE 红色分量
15:8	R/W	0x0	G_VALUE 绿色分量
7:0	R/W	0x0	B_VALUE 蓝色分量

10.1.5.11. 0x0B0 UI_LAYER_RECT_CTRL

默认值：0x00000000			UI 矩形区控制（UI Layer Rectangle Control）
位域	类型	默认值	描述
31:24	–	–	–
3	R/W	0x0	RECT_EN_3 矩形区域 3 开关控制

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			UI 矩形区控制 （UI Layer Rectangle Control ）
位域	类型	默认值	描述
			• 0：关闭 • 1：开启
2	R/W	0x0	RECT_EN_2 矩形区域 2 开关控制 • 0：关闭 • 1：开启
1	R/W	0x0	RECT_EN_1 矩形区域 1 开关控制 • 0：关闭 • 1：开启
0	R/W	0x0	RECT_EN_0 矩形区域 0 开关控制 • 0：关闭 • 1：开启

10.1.5.12. 0x0B4 UI_LAYER_PALETTE_ADDR

默认值：0x00000000			UI 图层调色板地址 （UI Layer Palette Address ）
位域	类型	默认值	描述
31:7	R/W	0x0	PALETTE_ADDR[31:7]UI 图层调色板地址调色板空间的大小为 1K bytes，按照索引顺序排列，包含 A、R、G 和 B 四个通道，总共可以索引 256 色。
6:0	RO	0x00	调色板地址低 7 bits 固定为 0。

10.1.5.13. 0x0C0+N*0x10(N=0:3) UI_RECT_INPUT_SIZE

默认值：0x00000000			UI 矩形区输入宽高 （UI Rectangle Input Size ）
位域	类型	默认值	描述
31:29	–	–	–
28:16	R/W	0x0	INPUT_HEIGHT 图层输入高度
15:13	–	–	–
12:0	R/W	0x0	INPUT _WIDTH 图层输入宽度

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.1.5.14. 0x0C4+N*0x10(N=0:3) UI_RECT_OFFSET

默认值：0x00000000			UI 矩形区偏移（UI Rectangle Offset）
位域	类型	默认值	描述
31:29	–	–	–
28:16	R/W	0x0	Y_OFFSET 垂直偏移
15:13	–	–	–
12:0	R/W	0x0	X_OFFSET 水平偏移

10.1.5.15. 0x0C8+N*0x10(N=0:3) UI_RECT_STRIDE

默认值：0x00000000			UI 矩形区行对齐（UI Rectangle Stride）
位域	类型	默认值	描述
31:16	–	–	–
15:0	R/W	0x0	STRIDE 矩形区 memory 行对齐宽度，需要 8 bytes 对齐。

10.1.5.16. 0x0CC+N*0x10(N=0:3) UI_RECT_ADDR

默认值：0x00000000			UI 矩形区地址（UI Rectangle Address）
位域	类型	默认值	描述
31:0	R/W	0x0	ADDR 矩形区地址

10.1.5.17. 0x170 CRC_CTRL

默认值：0x00000000			回写控制（CRC Control）
位域	类型	默认值	描述
31	R/W	0x0	CRC_EN • 0：关闭 CRC 校验功能 • 1：打开 CRC 校验功能
30:17	–	–	–
16	R/W	0x0	SOFT_RESET_EN • 0：关闭软复位 • 1：开启软复位
15:0	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.1.5.18. 0x194 CRC_CHECK

默认值：0x00000000			CRC 校验（CRC Check）
位域	类型	默认值	描述
31:0	R/W	0x0	CRC_CHECK 打开 CRC 校验功能后，在帧结束后可以查看到 CRC 校验值。

10.1.5.19. 0x198 INPUT_ID

默认值：0x00000000			输入 ID（Input ID）
位域	类型	默认值	描述
31:0	R/W	0x0	INPUT_ID 输入 ID 寄存器

10.1.5.20. 0x19C OUTPUT_ID

默认值：0x00000000			输出 ID（Output ID）
位域	类型	默认值	描述
31:0	R/W	0x0	OUTPUT_ID 输出 ID 寄存器

10.1.5.21. 0x1D0 TIMING_CTRL

默认值：0x0000FF00			Timing 控制（Timing Control）
位域	类型	默认值	描述
31:24	–	–	–
23:8	R/W	0xFF	TE 脉冲宽度
7:6	–	–	–
5:4	R/W	0x0	DE 模式 • 0：连续模式 • 1：单帧模式 • 2：自动单帧模式。自动单帧模式需要配置，TE 脉冲宽度 • 3：reserved
3:1	–	–	–
0	R/W	0x0	TIMING_EN • 0：关闭时序输出 • 1：开启时序输出

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.1.5.22. 0x1D4 TIMING_INT

默认值：0x00000000			Timing 中断（Timing Interrupt）
位域	类型	默认值	描述
31:9	–	–	–
8	R/W	0x0	SF_END_INTTE 中断 • 0：关闭 • 1：开启
7:3	–	–	–
2	R/W	0x0	UNDERFLOW_INT 读取数据下溢中断 • 0：关闭 • 1：开启
1	R/W	0x0	LINE_INT 行中断 • 0：关闭 • 1：开启
0	R/W	0x0	V_BLANK_INT 垂直消隐中断 • 0：关闭 • 1：开启

10.1.5.23. 0x1D8 TIMING_INT_STATUS

默认值：0x00000000			Timing 中断状态（Timing Interrupt Status）
位域	类型	默认值	描述
31:3	–	–	–
2	R/W1C	0x0	UNDERFLOW_FLAG 读取数据下溢标记 • 0：未溢出 • 1：溢出
1	R/W1C	0x0	LINE_FLAG 行中断状态标记 • 0：未到所设置行 • 1：到达所设置行
0	R/W1C	0x0	V_BLANK_FLAG 垂直消隐中断状态 • 0：未到达垂直消隐区 • 1：已到达垂直消隐区

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.1.5.24. 0x1DC TIMING_LINE_SET

默认值：0x00020000			Timing 行数设置（Timing Line Set）
位域	类型	默认值	描述
31:29	–	–	–
28:16	R/W	0x2	PREFETCH_LINE 预取行数
15:13	–	–	–
12:0	R/W	0x0	LINE_NUM 行中断行数

10.1.5.25. 0x1E0 TIMING_ACTIVE_SIZE

默认值：0x00000000			Timing 有效宽高（Timing Active Size）
位域	类型	默认值	描述
31:29	–	–	–
28:16	R/W	0x0	ACTIVE_H 有效高度
15:13	–	–	–
12:0	R/W	0x0	ACTIVE_W 有效宽度

10.1.5.26. 0x1E4 TIMING_H_PORCH

默认值：0x00000000			Timing Horizontal Porch（Timing Horizontal Porch）
位域	类型	默认值	描述
31:29	–	–	–
28:16	R/W	0x0	H_B_PORCH 水平后肩
15:13	–	–	–
12:0	R/W	0x0	H_F_PORCH 水平前肩

10.1.5.27. 0x1E8 TIMING_V_PORCH

默认值：0x00000000			Timing Vertical Porch（Timing Vertical Porch）
位域	类型	默认值	描述
31:29	–	–	–
28:16	R/W	0x0	V_B_PORCH 垂直后肩
15:13	–	–	–
12:0	R/W	0x0	V_F_PORCH 垂直前肩

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.1.5.28. 0x1EC TIMING_SYNC_PULSE

默认值：0x00000000			Timing 同步脉冲（Timing Sync Pulse）
位域	类型	默认值	描述
31:29	–	–	–
28:16	R/W	0x0	V_SYNC_WIDTH 垂直同步脉冲宽度
15:13	–	–	–
12:0	R/W	0x0	H_SYNC_WIDTH 水平同步脉冲宽度

10.1.5.29. 0x1F0 TIMING_POL_SET

默认值：0x00000000			Timing 极性配置（Timing Polarization Set）
位域	类型	默认值	描述
31:2	–	–	–
1	R/W	0x0	V_SYNC_POL 垂直同步脉冲极性 • 0：负极 • 1：正极
0	R/W	0x0	H_SYNC_POL 水平同步脉冲极性 • 0：负极 • 1：正极

10.1.5.30. 0x1F4 TIMING_DEBUG

默认值：0x00000000			Timing 调试（Timing Debug）
位域	类型	默认值	描述
31:13	–	–	–
12:0	RO	0x0	CUR_LINE 调试时候可以读取当前正在输出行数

10.1.5.31. 0x88C QOS_UI

默认值：0x00000000			QOS UI（QOS_UI）
位域	类型	默认值	描述
31:28	R/W	0x0	DMAR_QOS_GREEN
27:23	–	–	–
22:16	R/W	0x0	DMAR_QOS_HIGHRGB: 0 – 127RGB: 0 – 511

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			QOS UI（QOS_UI）
位域	类型	默认值	描述
15:12	R/W	–	DMAR_QOS_RED
11:7	–	–	–
6:0	R/W	0x0	DMAR_QOS_LOWRGB: 0 – 127RGB: 0 – 511

10.1.5.32. 0x890 QOS_URGENT

默认值：0x00000000			QoS Urgent（QOS_URGENT）
位域	类型	默认值	描述
31:23	–	–	–
22:16	R/W	0x0	DMAR_URGENT_HIGH_TH0 – 127 在 urgent 标记拉起后，当 FIFO 水位大于 TH，才会清除 urgent 标记。
15	R/W	0x0	DMAR_URGENT_EN • 0：disable • 1：enable
14	R/W	0x0	ONLY_ACTIVE_REGION_EN • 0：disable • 1：enable
13:7	–	–	–
6:0	R/W	0x0	DMAR_URGENT_LOW_TH0 – 127 当 FIFO 水位小于 DMAR_URGENT_LOW_TH 时，DE 会拉起 urgent 标记。

10.1.5.33. 0x900 CCM_C00

默认值：0x00000100			CCM 系数 C00（CCM_C00）
位域	类型	默认值	描述
31:12	–	–	–
11:0	R/W	0x100	CCM_C0012 bits 有符号数，8 bits 小数

10.1.5.34. 0x904 CCM_C01

默认值：0x00000000			CCM 系数 C01（CCM_C01）
位域	类型	默认值	描述
31:12	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			CCM 系数 C01（CCM_C01）
位域	类型	默认值	描述
11:0	R/W	0x000	CCM_C0112 bits 有符号数，8 bits 小数

10.1.5.35. 0x908 CCM_C02

默认值：0x00000000			CCM 系数 C02（CCM_C02）
位域	类型	默认值	描述
31:12	–	–	–
11:0	R/W	0x000	CCM_C0212 bits 有符号数，8 bits 小数

10.1.5.36. 0x90C CCM_OFFSET0

默认值：0x00000000			CCM 偏移 OFFSET0（CCM_OFFSET0）
位域	类型	默认值	描述
31:9	–	–	–
8:0	R/W	0x000	CCM_OFFSET09 bits 有符号整数

10.1.5.37. 0x910 CCM_C10

默认值：0x00000000			CCM 系数 C10（CCM_C10）
位域	类型	默认值	描述
31:12	–	–	–
11:0	R/W	0x000	CCM_C1012 bits 有符号数，8 bits 小数

10.1.5.38. 0x914 CCM_C11

默认值：0x00000100			CCM 系数 C11（CCM_C11）
位域	类型	默认值	描述
31:12	–	–	–
11:0	R/W	0x100	CCM_C1112 bits 有符号数，8 bits 小数

10.1.5.39. 0x918 CCM_C12

默认值：0x00000000			CCM 系数 C12（CCM_C12）
位域	类型	默认值	描述
31:12	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			CCM 系数 C12（CCM_C12）
位域	类型	默认值	描述
11:0	R/W	0x000	CCM_C1212 bits 有符号数，8 bits 小数

10.1.5.40. 0x91C CCM_OFFSET1

默认值：0x00000000			CCM 偏移 OFFSET1（CCM_OFFSET1）
位域	类型	默认值	描述
31:9	–	–	–
8:0	R/W	0x00	CCM_OFFSET09 bits 有符号整数

10.1.5.41. 0x920 CCM_C20

默认值：0x00000000			CCM 系数 C20（CCM_C20）
位域	类型	默认值	描述
31:12	–	–	–
11:0	R/W	0x000	CCM_C2012 bits 有符号数，8 bits 小数

10.1.5.42. 0x924 CCM_C21

默认值：0x00000000			CCM 系数 C21（CCM_C21）
位域	类型	默认值	描述
31:12	–	–	–
11:0	R/W	0x000	CCM_C2112 bits 有符号数，8 bits 小数

10.1.5.43. 0x928 CCM_C22

默认值：0x00000100			CCM 系数 C22（CCM_C22）
位域	类型	默认值	描述
31:12	–	–	–
11:0	R/W	0x100	CCM_C2212 bits 有符号数，8 bits 小数

10.1.5.44. 0x92C CCM_OFFSET2

默认值：0x00000000			CCM 偏移 OFFSET2（CCM_OFFSET2）
位域	类型	默认值	描述
31:9	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			CCM 偏移 OFFSET2（CCM_OFFSET2）
位域	类型	默认值	描述
8:0	R/W	0x00	CCM_OFFSET29 bits 有符号整数

10.1.5.45. 0x940+4*N(N=0:15) GAMMA_RED_LUT

默认值：0x00000000			Gamma 红色分量查找表（GAMMA_RED_LUT）
位域	类型	默认值	描述
31:24	R/W	0x0	RED_LUT[4*N + 3]，取值范围为 0 ~ 255
23:16	R/W	0x0	RED_LUT[4*N + 2]，取值范围为 0 ~ 255
15:8	R/W	0x0	RED_LUT[4*N + 1]，取值范围为 0 ~ 255
7:0	R/W	0x0	RED_LUT[4*N]，取值范围为 0 ~ 255

10.1.5.46. 0x980+4*N(N=0:15) GAMMA_GREEN_LUT

默认值：0x00000000			Gamma 绿色分量查找表（GAMMA_GREEN_LUT）
位域	类型	默认值	描述
31:24	R/W	0x0	GREEN_LUT[4*N + 3]，取值范围为 0 ~ 255
23:16	R/W	0x0	GREEN_LUT[4*N + 2]，取值范围为 0 ~ 255
15:8	R/W	0x0	GREEN_LUT[4*N + 1]，取值范围为 0 ~ 255
7:0	R/W	0x0	GREEN_LUT[4*N]，取值范围为 0 ~ 255

10.1.5.47. 0x9C0+4*N(N=0:15) GAMMA_BLUE_LUT

默认值：0x00000000			Gamma 绿色分量查找表（GAMMA_BLUE_LUT）
位域	类型	默认值	描述
31:24	R/W	0x0	BLUE_LUT[4*N + 3]，取值范围为 0 ~ 255
23:16	R/W	0x0	BLUE_LUT[4*N + 2]，取值范围为 0 ~ 255
15:8	R/W	0x0	BLUE_LUT[4*N + 1]，取值范围为 0 ~ 255
7:0	R/W	0x0	BLUE_LUT[4*N]，取值范围为 0 ~ 255

10.2. Graphics Engine (GE)

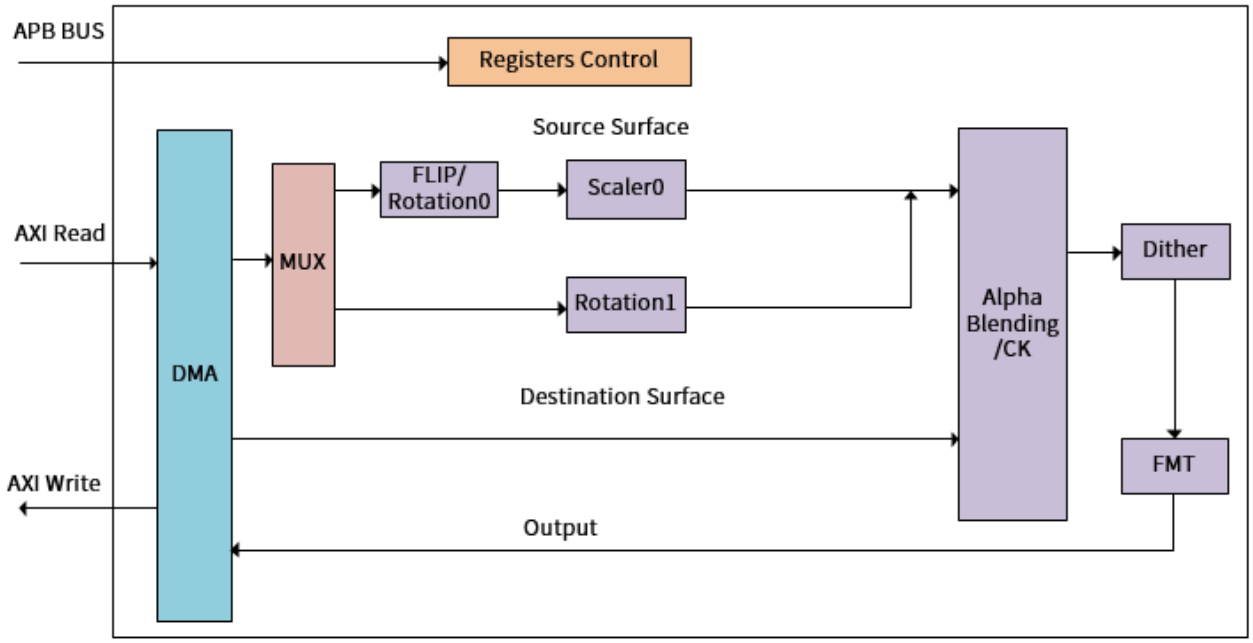
Graphics Engine (GE) 是用来进行 2D 图形加速的硬件模块。主要包括格式转换、旋转、镜像、 缩放、Alpha 混合、Color Key、位块搬移、Dither 等功能。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.2.1. 特性说明

- 最大输入/输出图像大小 4096x4096
- 支持命令队列
- 支持输入输出格式：ARGB8888/ XRGB8888/ RGB888/ ARGB4444/ ARGB1555/ RGB565
- 支持水平和垂直 Flip
- 支持 90/ 180/ 270 度旋转
- 支持任意角度旋转
- 支持 scan order
- 支持 1/16x ~ 16x 缩放，采用 bilinear 滤波算法
- 支持 Porter–Duff 规则的 Alpha 混合
- 支持 Color Key
- 支持矩形填充
- 位块搬移 (bit block transfer)
- 支持误差扩散 Dither

10.2.2. 原理框图



10.2.3. 功能描述

GE 模块包含以下子模块，各模块功能描述如下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- Flip/ Rotation0 子模块：可以进行水平镜像、垂直镜像、0/ 90/ 180/ 270 度固定角度旋转。

  水平镜像、垂直镜像和固定角度旋转三个功能可分别单独开启，也可同时开启。如果同时开启，水平镜像和垂直镜像旋转功能会在固定角度旋转功能之前生效。

- Rotation1 子模块：可以对图像进行任意角度旋转。任意角度旋转只支持 RGB 格式，不支持 YUV 格式。

  关于详细描述，可查看任意角度旋转。

- Scaler0 子模块：对特定格式的图像进行缩放。

  关于图像缩放的详细说明，可查看图像缩放。

- Porter–Duff Alpha Blending：支持选择不同的叠加系数对图像进行叠加混合。

- Dither 子模块：支持输入 ARGB8888 或者 RGB888 格式，输出 ARGB4444、RGB565、ARGB1555。打开 Dither 功能，可以使图像渐变区域更平滑。

- 格式化输出 (FMT) 子模块：对接收到的像素按照像素格式的排列要求写入 memory。

除了上述子模块支持的功能外，GE 还具有以下功能：

- 基本 2D 加速操作
- 命令队列 (Command Queue)

GE 使用过程中存在一定限制约束，详情可查看限制条件。

10.2.3.1. 任意角度旋转

Rotation1 子模块可以对图像进行任意角度旋转，以图像左上角为原点，原图的旋转中心在（m, n）目标图的旋转中心在（c, d），如下图所示：

> **注：**
>
> 任意角度旋转只支持 RGB 格式，不支持 YUV 格式。

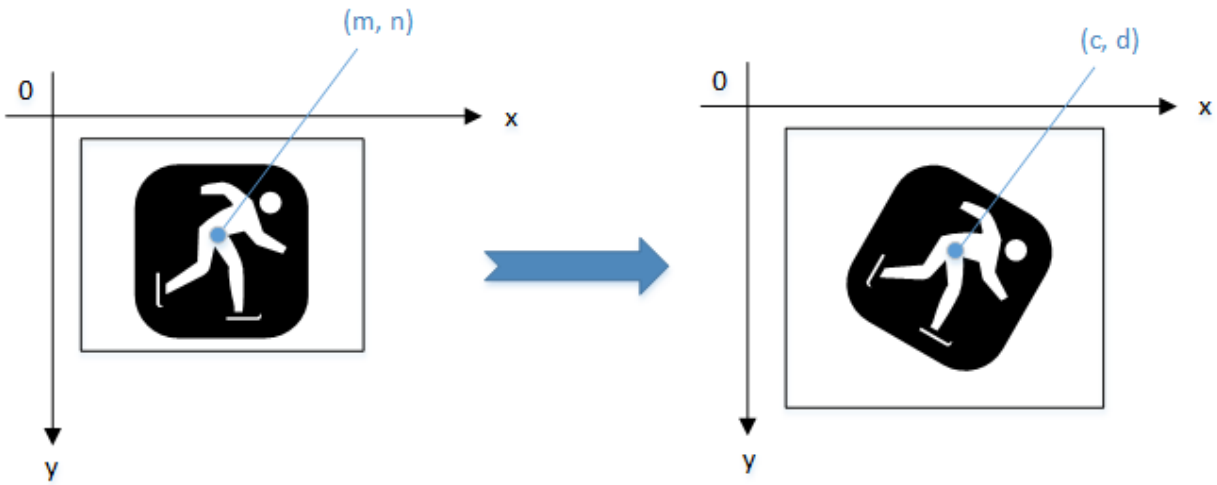


图 10–5 旋转示意图

由原图到目标图的旋转变换矩阵为：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

$$\begin{bmatrix} x1 \\ y1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & c \\ 0 & -1 & d \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} cosa & sina & 0 \\ -sina & cosa & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -m \\ 0 & -1 & n \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x0 \\ y0 \\ 1 \end{bmatrix}$$

图 10–6 旋转公式

10.2.3.2. 图像缩放

Scaler0 子模块支持对 RGB 格式图像进行缩放，支持缩放倍数范围为 1/16 ~ 16。

10.2.3.3. Porter/Duff Alpha Blending

```
pixel = (source * fs + destination * fd),
sa = source alpha,
da = destination alpha
```

序号	Blending 模式	fs	fd
0	NONE(默认)	sa	1–sa
1	CLEAR	0	0
2	SRC	1	0
3	DST	0	1
4	SRC_OVER	1	1–sa
5	DST_OVER	1–da	1
6	SRC_IN	da	0
7	DST_IN	0	sa
8	SRC_OUT	1–da	0
9	DST_OUT	0	1–sa
10	SRC_ATOP	da	1–sa
11	DST_ATOP	1–da	sa
12	XOR	1–da	1–sa
13	ADD	1	1

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.2.3.4. 基本 2D 加速操作

矩形填充和位块搬移是最基本的 2D 加速操作，其操作方法以及性能如下所示：

1. 矩形填充（Fill Rectangle）：指定内存中的一块矩形区域，并向矩形区域填充固定颜色，如下图所示：

   • Output Image Addr 为要矩形填充图像的基地址

   • 坐标 (x_offset, y_offset) 为填充区域相对于基地址的坐标偏移。

   由软件计算出坐标（x_offset，y_offset）地址 addr 配置给硬件，计算方法如下：

   ```
   bpp：每个像素占用的字节 （Bytes Per Pixel）
   addr = output_image_addr + output_image_stride * y_offset  +  x_offset * bpp
   ```

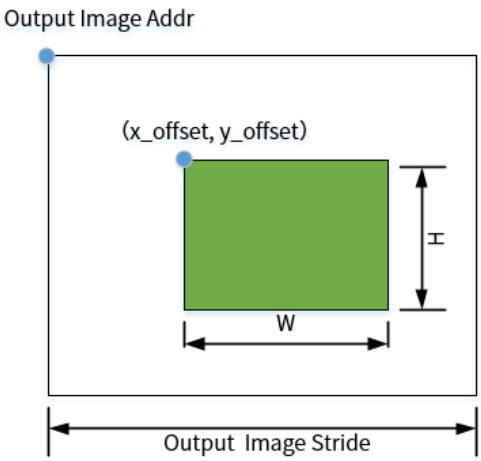


图 10–7 矩形填充

2. 位块搬移 (Bit BLT/ Stretch BLT)：把内存中源图像的一块矩形区域，搬移到目标图像中的一块矩形区域中。在位块搬移过程中，源图和目标图可以同时进行 alpha blending/CK 操作。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

a. 原图的矩形区域搬移到目标图的矩形区域中时不进行缩放（Bit BLT），如下图所示：

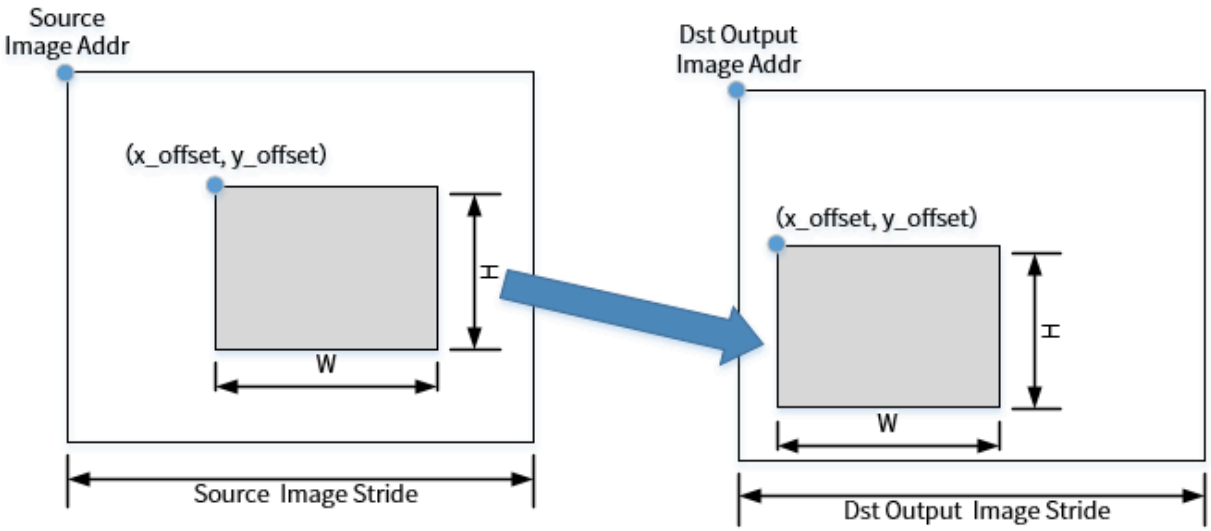


图 10–8 位图搬移

b. 原图的矩形区域搬移到目标图的矩形区域中同时进行放大或者缩小（Stretch BLT），如下图所示：

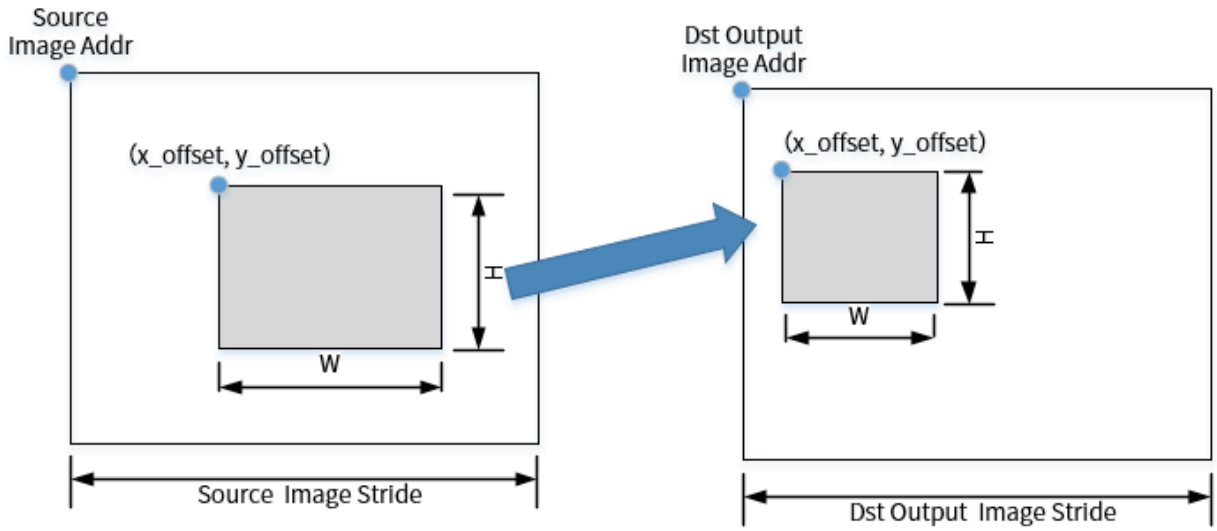


图 10–9 位图搬移同时缩放

3. 性能：GE 的最高工作频率为 150 MHz，开启不同功能的性能如下表所示。

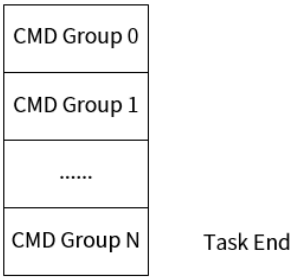
序号	2D 加速操作	性能（关闭 Rotation1）	性能（开启 Rotation1）
0	Fill Rectanle	1 pixel/cycle	N/A
1	Blit	0.95 pixel/cycle	0.45 pixel/cycle
2	Stretch BLT	0.7 pixel/cycle	N/A

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 设 mclk 为 GE 的工作时钟，GE 的性能为 P，则每秒钟可以处理的像素个数为：pixel_num = mclk * P。
  - mclk 单位：Hz
  - p 单位： pixel/cycle
- 每一帧处理的像素个数 frame_pixel = width * height，当使用 Stretch BLT 功能时，计算输入或者输出中最大图片的像素个数。
- GE 可以处理的最大帧率 frame per second (fps) = pixel_num ÷ frame_pixel。
- 当受到系统实际带宽限制时，可以用分配给 GE 的实际带宽量来计算实际帧率。

10.2.3.5. 命令队列 (Command Queue)

命令队列 (Command Queue) 涉及以下几个关键要素：

要素	说明
Task	GE 可以执行的最小任务单元，比如一次 blit 操作、一次矩形填充。 Task 由多个 CMD Group 组成： 
Batch	软件控制 GE 的最小单位，是一系列命令的集合，可以包含一个或者多个 Task。
Command Queue	对 Batch 进行管理的命令队列，硬件通过 Command Queue 来读取控制命令。

Command Queue 功能有如下特点：

- 采用 Command Queue 时，GE 可以完成多个 Task 后， 才触发一次中断，减少系统调度开销。

  不采用 Command Queue 时，GE 每个 task 结束都要触发一次中断。
- 采用 ring buffer 方式实现。

Ring buffer 示意图如下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

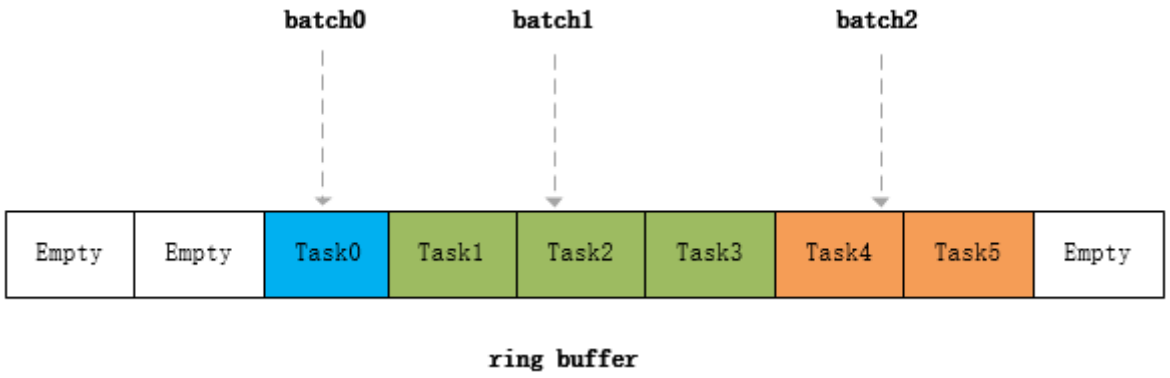


图 10–10 Ring Buffer

Ring Buffer 的工作流程如下：

1. 软件以 Batch 为单位向 CMD Ring Buffer 中写入控制命令。
2. 当控制命令写到 CMD Ring Buffer 尾部没有空间时，写地址跳到 CMD Ring Buffer 起始地址。完成此过程需要配给下列硬件信息：

   • CMD Buf Start Address：CMD Ring Buffer 的起始地址

   • CMD Buf End Address： CMD Ring Buffer 的结束地址

   • CMD Buf Address Offset：硬件读取 CMD 的地址相对于 CMD Buf Start Address 的偏移

   • CMD Buf Valid Length： 硬件读取命令的有效长度。

   假设有效长度为 length，当硬件读取长度为 length0 (length0 < length) 的命令，且 CMD 的读地址大于 CMD Buffer End Address 时，硬件会切换 CMD 读地址到 CMD Buffer Start Address，继续读取长度为 length1 的命令。此时有效长度 length 由 length0 和 length1 两部分组成，length = length0 + length1。

   a. Case 0：CMD Ring Buffer 没有回绕

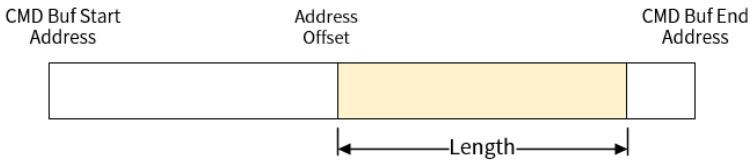


图 10–11 Ring Buf fer 无回绕

   b. Case 1：CMD Ring Buffer 有回绕

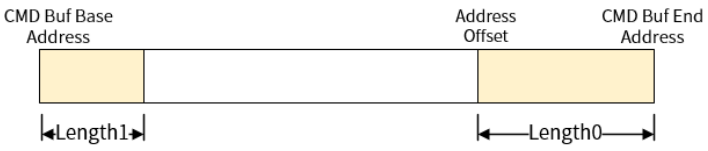


图 10–12 Ring Buf fer 有回绕

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.2.3.6. CMD Group 格式

每个 CMD Group 都由 group header 和多个 CMD 共同组成。通过 CMD Group 的方式将设置的命令和 GE 的寄存器建立起映射关系，如下表所示：

group header	4 bytes (little-endian)
CMD 0	4 bytes
……	……
CMD N	4 bytes

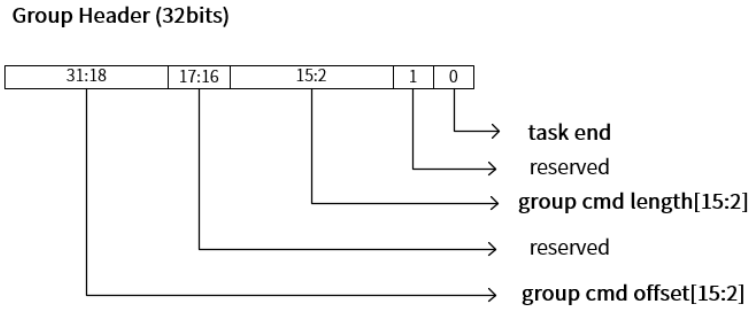


图 10–13 Group 格式

如图所示，Group Header 结构如下：

- task end：标记为 1 时，表示当前 Group 为当前 Task 的最后一个 Group。
- group cmd length：当前 Group 命令的总长度，单位为 byte。由于长度为 4 的倍数，低位 2 bits 固定为 0。
- group cmd offset：当前 Group 设置的命令，相对于 GE 寄存器的基地址的偏移，单位为 byte。由于 offset 为 4 的倍数，低位 2 bits 固定为 0。
- CMD 0 ~ N: 由 length 和 group cmd offset 指定的 GE 寄存器的配置。

表 10–2 GE 部分功能寄存器列表

寄存器名	偏移地址	寄存器功能描述
GE_CTRL	0x000	GE 控制寄存器
GE_STATUS	0x004	GE 状态寄存器
GE_START	0x008	GE 启动寄存器
GE_RESET	0x00C	GE 复位寄存器
SRC_SURFACE_CTRL	0x010	SRC 控制寄存器

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 10–2 GE 部分功能寄存器列表 (续)

寄存器名	偏移地址	寄存器功能描述
SRC_SURFACE_INPUT_SIZE	0x014	SRC 输入宽高寄存器
SRC_SURFACE_STRIDE	0x018	SRC 行对齐寄存器
SRC_SURFACE_FILL_COLOER	0x01C	SRC 颜色填充寄存器
SRC_SURFACE_ADDR0	0x020	SRC 地址 0 寄存器
–	0x024	–
–	0x028	–
–	0x02C	–
SRC_GRADIENT_A_STEP	0x030	SRC 渐变填充 Alpha 步进寄存器
SRC_GRADIENT_R_STEP	0x034	SRC 渐变填充 Red 步进寄存器
SRC_GRADIENT_G_STEP	0x038	SRC 渐变填充 Green 步进寄存器
SRC_GRADIENT_B_STEP	0x03C	SRC 渐变填充 Blue 步进寄存器

以下展示了几种不同 CMD Group 中命令和 GE 寄存器之间的映射关系示例：

1. 如果 Group 对应 GE 的 0x010、0x014、0x018 三 个寄存器，则：

   • group cmd length：0x0C

   • group cmd offset：0x10

   • task end: 0

group header	4 bytes (little–endian)	0x0010000C
CMD 0	4 bytes	映射为寄存器 0x010
CMD 1	4 bytes	映射为寄存器 0x014
CMD 2	4 bytes	映射为寄存器 0x018

2. 如果 CMD Group 对应 GE 的 0x030、0x034、0x038、0x03C 四个寄存器：

   • group cmd length：0x10

   • group cmd offset：0x30

   • task end: 0

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

group header	4 bytes (little-endian)	0x00300010
CMD 0	4 bytes	映射为寄存器 0x030
CMD 1	4 bytes	映射为寄存器 0x034
CMD 2	4 bytes	映射为寄存器 0x038
CMD 3	4 bytes	映射为寄存器 0x03C

3. 如果 Group 对应 GE 的 0x030、0x034、0x038、0x03C 四个寄存器，并且当前 Group 为当前 Task 最后一个 Group：

   • group cmd length：0x10

   • group cmd offset：0x30

   • task end: 1

group header	4 bytes (little-endian)	0x00300011
CMD 0	4 bytes	映射为寄存器 0x030
CMD 1	4 bytes	映射为寄存器 0x034
CMD 2	4 bytes	映射为寄存器 0x038
CMD 3	4 bytes	映射为寄存器 0x03C

10.2.3.7. 限制条件

• 任意角度旋转不支持 Dither，不支持 YUV 格式，并且 DST 和 OUT 地址需要一致。

• Scan order 只在 scaler、rot0、rot1 关闭的情况下开启，并且不可以开启 Dither。

• 不开启任意角度旋转支持最小 size：1x1（单位为 pixel）。

• 开启任意角度旋转支持最小 size：4x4（单位为 pixel）。

• 支持最大 size：4096x4096（单位为 pixel）。

• 颜色填充模式/渐变填充模式下，不可以开启 scaler、rot0、rot1。

• 配置 queue 的 Ring Buffer 的起始地址和总长度需要 128 bytes 对齐。

• 任意角度旋转只支持 blending，不支持 color key。

• 不支持 YUV 格式。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.2.4. 寄存器列表

表 10–3 GE 寄存器列表

偏移	缩写	寄存器名称	章节
0x000	GE_INT_CTRL	GE 中断控制	0x000 GE_INT_CTRL
0x004	GE_STATUS	GE 状态	0x004 GE_STATUS
0x008	GE_START	GE 启动	0x008 GE_START
0x00C	GE_VER_ID	GE 版本	0x00C GE_VER_ID
0x010	SRC_SURFACE_CTRL	SRC 控制	0x010 SRC_SURFACE_CTRL
0x014	SRC_SURFACE_INPUT_SIZE	SRC 输入宽高	0x014 SRC_SURFACE_INPUT_SIZE
0x018	SRC_SURFACE_STRIDE	SRC 行对齐	0x018 SRC_SURFACE_STRIDE
0x01C	SRC_SURFACE_FILL_COLOR	SRC 颜色填充	0x01C SRC_SURFACE_FILL_COLOR
0x020	SRC_SURFACE_ADDR0	SRC 地址 0	0x020 SRC_SURFACE_ADDR0
0x030	SRC_GRADIENT_A_STEP	SRC 渐变填充 Alpha 步进	0x30 SRC_GRADIENT_A_STEP
0x034	SRC_GRADIENT_R_STEP	SRC 渐变填充 Red 步进	0x34 SRC_GRADIENT_R_STEP
0x038	SRC_GRADIENT_G_STEP	SRC 渐变填充 Green 步进	0x38 SRC_GRADIENT_G_STEP
0x03C	SRC_GRADIENT_B_STEP	SRC 渐变填充 Blue 步进	0x3C SRC_GRADIENT_B_STEP
0x050	DST_SURFACE_CTRL	DST 控制	0x050 DST_SURFACE_CTRL
0x054	DST_SURFACE_INPUT_SIZE	DST 输入宽高	0x054 DST_SURFACE_INPUT_SIZE
0x058	DST_SURFACE_STRIDE	DST 行对齐	0x058 DST_SURFACE_STRIDE
0x060	DST_SURFACE_ADDR0	DST 地址 0	0x060 DST_SURFACE_ADDR0
0x070	SRC_ROT1_CENTER	SRC 旋转中心	0x070 SRC_ROT1_CENTER
0x074	SRC_ROT1_DEGREE	SRC 旋转角度	0x074 SRC_ROT1_DEGREE
0x078	DST_ROT1_CENTER	DST 旋转中心	0x078 DST_ROT1_CENTER
0x080	HW_TIMEOUT_CYCLE	硬件超时 Cycle	0x080 HW_TIMEOUT_CYCLE

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 10–3 GE 寄存器列表 (续)

偏移	缩写	寄存器名称	章节
0x088	HW_COUNTER	硬件执行时间计数	0x088 HW_COUNTER
0x08C	SOFT_RESET_CYCLE	软复位 Cycle	0x08C SOFT_RESET_CYCLE
0x090	BLENDING_CTRL	Blending 控制	0x90 BLENDING_CTRL
0x094	COLORKEY_MATCH_COLOR	Colorkey 匹配颜色	0x94 COLORKEY_MATCH_COLOR
0x100	OUTPUT_CTRL	DST 输出控制	0x100 OUTPUT_CTRL
0x104	OUTPUT_SIZE	DST 输出宽高	0x104 OUTPUT_SIZE
0x108	OUTPUT_STRIDE	DST 输出行对齐	0x108 OUTPUT_STRIDE
0x110	OUTPUT_ADDR0	DST 输出地址 0	0x110 OUTPUT_ADDR0
0x120	DITHER_LINE_BUF_ADDR	Dither 行缓冲地址	0x120 DITHER_LINE_BUF_ADDR
0x130	CMD_BUF_START_ADDR	CMD BUF 起始地址	0x130 CMD_BUF_START_ADDR
0x134	CMD_BUF_END_ADDR	CMD BUF 结束地址	0x134 CMD_BUF_END_ADDR
0x138	CMD_BUF_OFFSET	CMD BUF 地址偏移寄存器	0x0138 CMD_BUF_OFFSET
0x13C	CMD_BUF_VALID_LENGTH	CMD BUF 有效长度寄存器	0x013C CMD_BUF_VALID_LENGTH
0x200	SCALER0_CTRL	Scaler0 控制	0x200 SCALER0_CTRL
0x210	SCALER_CH0_INPUT_SIZE	Scaler0 Chan0 输入宽高	0x210 SCALER_CH0_INPUT_SIZE
0x214	SCALER_CH0_OUTPUT_SIZE	Chan0 输出宽高	0x214 SCALER_CH0_OUTPUT_SIZE
0x218	SCALER0_CH0_H_INIT_PHASE	Scaler0 Chan0 水平初始相位	0x218 SCALER0_CH0_H_INIT_PHASE
0x21C	SCALER0_CH0_H_RATIO	Scaler0 Chan0 水平缩放比例	0x21C SCALER0_CH0_H_RATIO
0x220	SCALER0_CH0_V_INIT_PHASE	Scaler0 Chan0 垂直初始相位	0x220 SCALER0_CH0_V_INIT_PHASE
0x224	SCALER0_CH0_V_RATIO	Scaler0 Chan0 垂直缩放比例	0x224 SCALER0_CH0_V_RATIO

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.2.5. 寄存器描述

10.2.5.1. 0x000 GE_INT_CTRL

默认值：0x00000000			GE 中断控制 （GE Interrupt Control）
位域	类型	默认值	描述
31:9	–	–	–
8	R/W	0x0	HW_CMDQ_LENGTH_ERR_IRQ_EN 命令队列长度错误中断控制 • 0:不使能 • 1: 使能 用于控制命令队列长度出错后，是否发送中断请求。
7:4	–	–	–
3	R/W	0x0	HW_TIMEOUT_IRQ_EN 超时中断控制 • 0: 不使能 • 1: 使能 用于控制硬件执行时间超时后，是否发送中断请求，如果要使能超时中断，需要通过 HW_TIMEOUT_CYCLE 来设置超时时间。
2	–	–	–
1	R/W	0x0	HW_ERR_IRQ_EN 硬件错误中断控制 • 0:不使能 • 1: 使能 HW_ERR_IRQ_EN 用于控制 GE 硬件出错后，是否发送中断请求，当 EN 为 1 的时候，发送中断请求，否则不发送。是否产生中断请求，不影响相应的状态标志位的产生。
0	R/W	0x0	FINISH_IRQ_EN 任务结束中断控制。用于控制任务处理结束后，是否发送中断请求： • 0:关闭，不发送断请求 • 1:开启，发送中断请求 是否产生中断请求，不影响相应的状态标志位的产生。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.2.5.2. 0x004 GE_STATUS

默认值：0x00000000			GE 状态（GE Status）
位域	类型	默认值	描述
31:16	R/W1C	0	TASK_NUM 启动一次硬件完成任务个数统计，当开启 Command Queue 模式的时候，启动一次硬件可以连续完成多个任务，没有开启 Queue，一次只能完成一个任务。
15:9	–	–	–
8	R/W1C	0x0	HW_CMDQ_LENGTH_ERR_IRQ_STATUS 命令队列长度错误状态 当硬件检测到命令队列长度出错，停止当前任务，由硬件置 1，软件向该位写 1 清除中断状态。
7:6	–	–	–
5	R/W1C	0x0	HW_BUSY_STATUS 硬件 busy 状态，如果为 1，表示硬件处于 busy 状态。
4	–	–	–
3	R/W1C	0x0	HW_TIMEOUT_STATUS 硬件超时状态，如果为 1，表示硬件执行时间超时。
2	–	–	–
1	R/W1C	0	HW_ERR_IRQ_STATUS 硬件错误标记状态 当出现硬件错误后，停止当前任务，由硬件置 1，软件向该位写 1 清除中断状态。
0	R/W1C	0	FINISH_IRQ_STATUS 当前任务结束后，由硬件置‘1’，软件向该位写 1 清除中断状态。

10.2.5.3. 0x008 GE_START

默认值：0x00000000			GE 启动（GE Start）
位域	类型	默认值	描述
31:3	–	–	–
2	R/W	0x0	SW_RESET

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			GE 启动（GE Start）
位域	类型	默认值	描述
			• 0: 关闭每帧 reset • 1: 开启每帧 reset
1	R/W	0x0	GE_CMD_QUEUE_CTRL • 0: 关闭 Command Queue • 1: 开启 Command Queue
0	R/W	0x0	GE_START GE 启动控制由软件写 1 启动 GE，启动后由硬件自动清零。

10.2.5.4. 0x00C GE_VER_ID

默认值：0x00000101			GE 版本（GE Version ID）
位域	类型	默认值	描述
31:16	R0	–	GE_PATCH_ID
15:0	RO	0x00000101	GE_VER_ID 采用 BCD 码显示，V1.1

10.2.5.5. 0x010 SRC_SURFACE_CTRL

默认值：0x00000000			SRC 控制 （Source Surface Control）
位域	类型	默认值	描述
31:24	R/W	0x0	G_ALPHA 图层全局 alpha 值 0 ~ 255
23:22	R/W	0x0	SOURCE_ALPHA_MODE Source Alpha 模式 • 0:Pixels alpha 模式 Source Alpha 为 Pixels alpha，每个 pixel 都有独立的 alpha，当 Pixels alpha 不存在时，默认 Pixels alpha 值为 255 • 1:全局 alpha 模式 Source Alpha 的值为 G_ALPHA • 2:混合 alpha 模式

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			SRC 控制 （Source Surface Control）
位域	类型	默认值	描述
			Source Alpha 值为 Pixels alpha 与 G_ALPHA 的乘积 Source Alpha = Pixels alpha * G_ALPHA / 255
21	R/W	0x0	P_MUL Source Pre-multiplied alpha 标记 • 0: 不进行预乘操作 • 1: 进行预乘操作 此标记只应用于 RGB 格式。
20	–	–	–
19:18	–	–	SCAN_ORDER • 0x00: top to bottom, left to right • 0x01: top to bottom, right to left • 0x10: bottom to top, left to right • 0x11: bottom to top, right to left
17	–	–	–
16	R/W	0x0	FUNC_SELECT 功能选择 • 0: 基本功能 • 1: 任意角度旋转 (Rotation1 ） 当 FUNC_ SELECT 为 0 的时候，可以进行固定角度旋转，镜像和 scaler。当 FUNC_SELECT 为 1 的时候选择任意角度旋转功能。将不可以使用 scaler 与 Flip 功能。
15	–	–	–
14:8	R/W	0x0	FORMAT 像素格式 • 0x00: ARGB 8-8-8-8 • 0x01: ABGR 8-8-8-8 • 0x02: RGBA 8-8-8-8 • 0x03: BGRA 8-8-8-8 • 0x04: XRGB 8-8-8-8 • 0x05: XBGR 8-8-8-8 • 0x06: RGBX 8-8-8-8

Copyright © 2023-2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			SRC 控制 （Source Surface Control ）
位域	类型	默认值	描述
			• 0x07: BGRX 8–8–8–8 • 0x08: RGB 8–8–8 • 0x09: BGR 8–8–8 • 0x0A: ARGB 1–5–5–5 • 0x0B: ABGR 1–5–5–5 • 0x0C: RGBA 5–5–5–1 • 0x0D: BGRA 5–5–5–1 • 0x0E: RGB 5–6–5 • 0x0F: BGR 5–6–5 • 0x10: ARGB 4–4–4–4 • 0x11: ABGR 4–4–4–4 • 0x12: RGBA 4–4–4–4 • 0x13: BGRA 4–4–4–4 • 0x14–0x29: reserved • 0x2a: YUV400 • 0x2b: reserved
7	R/W	0x0	V_FLIP 垂直镜像控制 • 0: 关闭 • 1: 开启 当 FUNC_SELECT 为 0 时，此功能可以使用，在 Rotation0 之前生效。
6	R/W	0x0	H_FLIP 水平镜像控制 • 0: 关闭 • 1: 开启 当 FUNC_ SELECT 为 0 时，此功能可以使用，在 Rotation0 之前生效。
5:4	R/W	0x0	ROT0_CTRL Rotation0 旋转控制 • 0: 不旋转 • 1: 90 度顺时针旋转 • 2: 180 度顺时针旋转 • 3: 270 度顺时针旋转

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			SRC 控制 （Source Surface Control）
位域	类型	默认值	描述
			当 FUNC_SELECT 为 0 的时候，此功能可以使用，在 Rotation0 之前生效。
3:2	R/W	0x0	SOURCE_MODE Source Surface 模式选择 • 0: 读取 buffer 模式 • 1: 固定色填充 • 2: 水平渐变填充 • 3: 垂直渐变填充 当 MODE 为 0 的时候，为从 buffer 中读取数据，当 MODE 为 1 的时候为固定颜色填充模式，填充颜色为 COLOR，当 MODE 为 2 或 3 时，SRC_SURFACE_FILL_COLOR 为初始颜色值。
1	–	–	–
0	R/W	0x0	EN Source Surface 使能控制 • 0: 关闭 • 1: 开启

10.2.5.6. 0x014 SRC_SURFACE_INPUT_SIZE

默认值：0x00000000			SRC 输入宽高 （Source Surface Input Size）
位域	类型	默认值	描述
31:29	–	–	–
28:16	R/W	0x0	INPUT_HEIGHT 输入高度
15:13	–	–	–
12:0	R/W	0x0	INPUT_WIDTH 输入宽度

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.2.5.7. 0x018 SRC_SURFACE_STRIDE

默认值：0x00000000			SRC 行对齐（Source Surface Stride）
位域	类型	默认值	描述
31:16	–	–	–
15:0	R/W	0x0	PLANE0_STRIDE Plane0 行对齐宽度，需要 8 bytes 对齐。

10.2.5.8. 0x01C SRC_SURFACE_FILL_COLOR

默认值：0x00000000			SRC 颜色填充（Source Surface Fill color）
位域	类型	默认值	描述
31:25	R/W	0x0	PIXELS_ALPHA 颜色填充 Alpha 分量
24:16	R/W	0x0	PIXELS_R 颜色填充红色分量
15:8	R/W	0x0	PIXELS_G 颜色填充绿色分量
7:0	R/W	0x0	PIXELS_B 颜色填充蓝色分量

10.2.5.9. 0x020 SRC_SURFACE_ADDR0

默认值：0x00000000			SRC 地址 0（Source Surface Address0）
位域	类型	默认值	描述
31:0	R/W	0x0	ADDR0 Source Surface plane0 地址

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.2.5.10. 0x30 SRC_GRADIENT_A_STEP

默认值：0x00000000			SRC 渐变填充 Alpha 步进（Source Gradient Alpha Step）
位域	类型	默认值	描述
31:25	–	–	–
24:16	R/W	0x0	A_STEP_INIT 渐变步进 alpha 整数部分，为有符号数。
15:0	R/W	0x0	A_STEP_FRAC 渐变步进 alpha 小数部分。

10.2.5.11. 0x34 SRC_GRADIENT_R_STEP

默认值：0x00000000			SRC 渐变填充 Red 步进（Source Gradient Red Step）
位域	类型	默认值	描述
31:25	–	–	–
24:16	R/W	0x0	R_STEP_INIT 渐变步进 Red 整数部分，为有符号数。
15:0	R/W	0x0	R_STEP_FRAC 渐变步进 Red 小数部分。

10.2.5.12. 0x38 SRC_GRADIENT_G_STEP

默认值：0x00000000			SRC 渐变填充 Green 步进（Source Gradient Green Step）
位域	类型	默认值	描述
31:25	–	–	–
24:16	R/W	0x0	G_STEP_INIT 渐变步进 Green 整数部分，为有符号数。
15:0	R/W	0x0	G_STEP_FRAC 渐变步进 Green 小数部分。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.2.5.13. 0x3C SRC_GRADIENT_B_STEP

默认值：0x00000000			SRC 渐变填充 Blue 步进 （Source Gradient Blue Step ）
位域	类型	默认值	描述
31:25	–	–	–
24:16	R/W	0x0	B_STEP_INIT 渐变步进 Blue 整数部分，为有符号数。
15:0	R/W	0x0	B_STEP_FRAC 渐变步进 blue 小数部分。

10.2.5.14. 0x050 DST_SURFACE_CTRL

默认值：0x00000000			DST 控制 （Dst Surface Control ）
位域	类型	默认值	描述
31:24	R/W	0x0	G_ALPHA 图层全局 alpha 值 0 ~ 255
23:22	R/W	0x0	DST_ALPHA_MODE Dst Alpha 模式 • 0：Pixels alpha 模式 Dst Alpha 为 Pixels alpha，每个 pixel 都有独立的 alpha，当 Pixels alpha 不存在时，默认 Pixels alpha 值为 255 • 1：全局 alpha 模式 Dst Alpha 的值为 G_ALPHA • 2：混合 alpha 模式 Dst Alpha 值为 Pixels alpha 与 G_ALPHA 的乘积 Dst Alpha = Pixels alpha * G_ALPHA/255
21:15	–	–	–
14:8	R/W	0x0	FORMAT 像素格式 • 0x00: ARGB 8–8–8–8 • 0x01: ABGR 8–8–8–8 • 0x02: RGBA 8–8–8–8 • 0x03: BGRA 8–8–8–8 • 0x04: XRGB 8–8–8–8 • 0x05: XBGR 8–8–8–8

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			DST 控制 （Dst Surface Control）
位域	类型	默认值	描述
			• 0x06: RGBX 8–8–8–8 • 0x07: BGRX 8–8–8–8 • 0x08: RGB 8–8–8 • 0x09: BGR 8–8–8 • 0x0A: ARGB 1–5–5–5 • 0x0B: ABGR 1–5–5–5 • 0x0C: RGBA 5–5–5–1 • 0x0D: BGRA 5–5–5–1 • 0x0E: RGB 5–6–5 • 0x0F: BGR 5–6–5 • 0x10: ARGB 4–4–4–4 • 0x11: ABGR 4–4–4–4 • 0x12: RGBA 4–4–4–4 • 0x13: BGRA 4–4–4–4
7: 1	–	–	–
0	R/W	0x0	EN Destination Surface 使能控制 • 0:关闭 • 1:开启

10.2.5.15. 0x054 DST_SURFACE_INPUT_SIZE

默认值：0x00000000			DST 输入宽高 （Dst Surface Input Size）
位域	类型	默认值	描述
31:29	–	–	–
28:16	R/W	0x0	INPUT_HEIGHT 图层高度
15:13	–	–	–
12:0	R/W	0x0	INPUT_WIDTH 图层宽度

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.2.5.16. 0x058 DST_SURFACE_STRIDE

默认值：0x00000000			DST 行对齐（Dst Surface Stride）
位域	类型	默认值	描述
31:16	–	–	–
15:0	R/W	0x0	PLANE0_STRIDE Plane0 行对齐宽度，需要 8 bytes 对齐。

10.2.5.17. 0x060 DST_SURFACE_ADDR0

默认值：0x00000000			DST 地址 0（Dst Surface Address0）
位域	类型	默认值	描述
31:0	R/W	0x0	ADDR0 Destination Surface plane0 地址。

10.2.5.18. 0x070 SRC_ROT1_CENTER

默认值：0x00000000			SRC 旋转中心（Source Rotate1 Center）
位域	类型	默认值	描述
31:30	–	–	–
29:16	R/W	0x0	Y_POTISION 当 SRC 开启任意角度旋转的时候，旋转中心 Y 坐标，取值范围 0 ~ 4095。
15:14	–	–	–
13:0	R/W	0x0	X_POTISION 当 SRC 开启任意角度旋转的时候，旋转中心 X 坐标，取值范围 0 ~ 4095。

10.2.5.19. 0x074 SRC_ROT1_DEGREE

默认值：0x00000000			SRC 旋转角度（Source Rotate1 Degree）
位域	类型	默认值	描述
31:30	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			SRC 旋转角度（Source Rotate1 Degree）
位域	类型	默认值	描述
29:16	R/W	0x0	COS_VALUE 当 SRC 开启任意角度旋转的时候，旋转角度的 cos 值，为有符号数，其中 bits[27:16] 为小数部分，bits[29:28] 为整数部分。
15:14	–	–	–
13:0	R/W	0x0	SIN_VALUE 当 SRC 开启任意角度旋转的时候，旋转角度的 sin 值，为有符号数，其中 bits[11:0] 为小数部分，bits[13:12] 为整数部分。

10.2.5.20. 0x078 DST_ROT1_CENTER

默认值：0x00000000			DST 旋转中心（Dst Rotate1 Center）
位域	类型	默认值	描述
31:30	–	–	–
29:16	R/W	0x0	Y_POTISION 当 SRC 开启任意角度旋转的时候，SRC 旋转中心对应的 DST 旋转中心的位置 Y 坐标，取值范围 0 ~ 4095。
15:14	–	–	–
13:0	R/W	0x0	X_POTISION 当 SRC 开启任意角度旋转的时候，SRC 旋转中心对应的 DST 旋转中心的位置 X 坐标，取值范围 0 ~ 4095。

10.2.5.21. 0x080 HW_TIMEOUT_CYCLE

默认值：0x00000000			硬件超时 Cycle（HW Timeout Cycle）
位域	类型	默认值	描述
31:0	R/W	0x0	HW_TIMEOUT_CYCLE 硬件超时计数（APB Clock 域）以 Cycle 个数进行设置，当设置为 0 时，不开启超时功能。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.2.5.22. 0x088 HW_COUNTER

默认值：0x00000000			硬件执行时间计数（HW Counter）
位域	类型	默认值	描述
31:0	RO	0x0	HW_COUNTER 硬件执行时间计数（APB Clock 域）以 Cycle 个数进行统计。

10.2.5.23. 0x08C SOFT_RESET_CYCLE

默认值：0x00100010			软复位 Cycle（Soft Reset Cycle）
位域	类型	默认值	描述
31:16	R/W	0x0010	SOFT_RESET_DELAY 软复位后延时时间（AXI Clock 域）以 Cycle 个数进行设置。
15:0	R/W	0x0010	SOFT_RESET_CYCLE 软复位时间（AXI Clock 域）以 Cycle 个数进行设置。

10.2.5.24. 0x90 BLENDING_CTRL

默认值：0x00000000			Blending 控制（Blending Control）
位域	类型	默认值	描述
31:18	–	–	–
17	R/W	0x0	SRC_DE_P_MUL SRC 去预乘标记，在进行 Blending 之前生效，此标记只应用于 RGB 格式。 • 0: SRC 不进行去预乘操作 • 1:SRC 进行去除预乘操作
16	R/W	0x0	DST_DE_P_MUL DST 去预乘标记，在进行 Blending 之前生效，此标记只应用于 RGB 格式。 • 0: DST 不进行去预乘操作 • 1: DST 进行去除预乘操作
15	R/W	0x0	OUTPUT_ALPHA_CTRL Alpha 输出控制

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			Blending 控制 （Blending Control）
位域	类型	默认值	描述
			• 0：Source Surface 和 Surface 的 alpha 进行混合后输出 • 1：固定输出 0xFF
14	–	–	–
13:11	R/W	0x02	SRC_ALPHA_COEF Source Surface Alpha 混合系数 • 0x000: 0.0 • 0x001: 1.0 • 0x010: Source Alpha • 0x011: 1.0 – Source Alpha • 0x100: Dst alpha • 0x101: 1.0 – Dst Alpha
10:8	R/W	0x03	DST_ALPHA_COEF Dst Surface Alpha 混合系数 • 0x000: 0.0 • 0x001: 1.0 • 0x010: Source Alpha • 0x011: 1.0 – Source Alpha • 0x100: Dst alpha • 0x101: 1.0 – Dst Alpha
7:2	–	–	–
1	R/W	0x0	CK_EN Color Key 开关控制 • 0：关闭 Color Key • 1：Color Key 匹配 Source Surface 当 BLEND_EN 和 CK_EN 同时开启的时候，Color Key (CK) 匹配上的区域做 Color Key，没有匹配的区域做 Alpha Blending，当 Rotation1 功能打开时不支持 CK。
0	R/W	0x0	ALPHA_BLEND_EN Alpha 混合开关控制 • 0：关闭 Alpha Blending • 1：开启 Alpha Blending

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.2.5.25. 0x94 COLORKEY_MATCH_COLOR

默认值：0x00000000			Color Key 匹配颜色（Color Key Match Color）
位域	类型	默认值	描述
31:24	–	–	–
23:16	R/W	0x0	CK_R Color Key 红色分量匹配值
15:8	R/W	0x0	CK_G Color Key 绿色分量匹配值
7:0	R/W	0x0	CK_B Color Key 蓝色分量匹配值

10.2.5.26. 0x100 OUTPUT_CTRL

默认值：0x00000000			DST 输出控制（Dst Output Control）
位域	类型	默认值	描述
31:17	–	–	–
16	R/W	0x0	OUT_P_MUL_ALPHA 输出像素预乘 Alpha • 0x0：输出像素预乘 Alpha • 0x1：输出像素不预乘 Alpha
15	–	–	–
14:8	R/W	0x0	FORMAT 像素格式 • 0x00: ARGB 8–8–8–8 • 0x01: ABGR 8–8–8–8 • 0x02: RGBA 8–8–8–8 • 0x03: BGRA 8–8–8–8 • 0x04: XRGB 8–8–8–8 • 0x05: XBGR 8–8–8–8 • 0x06: RGBX 8–8–8–8 • 0x07: BGRX 8–8–8–8

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			DST 输出控制（Dst Output Control）
位域	类型	默认值	描述
			• 0x08: RGB 8–8–8 • 0x09: BGR 8–8–8 • 0x0A: ARGB 1–5–5–5 • 0x0B: ABGR 1–5–5–5 • 0x0C: RGBA 5–5–5–1 • 0x0D: BGRA 5–5–5–1 • 0x0E: RGB 5–6–5 • 0x0F: BGR 5–6–5 • 0x10: ARGB 4–4–4–4 • 0x11: ABGR 4–4–4–4 • 0x12: RGBA 4–4–4–4 • 0x13: BGRA 4–4–4–4
7:5	–	–	–
4	R/W	0x0	DITHER_EN Dither 控制 • 0:关闭 dither • 1:打开 dither 当输出为 16–bit RGB 格式时，开启 Dither 功能可以优化图像渐变区域的显示效果，Dither 不对 alpha 分量进行处理。
3:1	–	–	–
0	–	–	–

10.2.5.27. 0x104 OUTPUT_SIZE

默认值：0x00000000			DST 输出宽高（Dst Output Size）
位域	类型	默认值	描述
31:29	–	–	–
28:16	R/W	0x0	OUTPUT_HEIGHT 输出高度
15:13	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			DST 输出宽高（Dst Output Size）
位域	类型	默认值	描述
12:0	R/W	0x0	OUTPUT_WIDTH 输出宽度

10.2.5.28. 0x108 OUTPUT_STRIDE

默认值：0x00000000			DST 输出行对齐（Dst Output Stride）
位域	类型	默认值	描述
31:16	–	–	–
15:0	R/W	0x0	PLANE0_STRIDE Plane0 行对齐宽度，需要 8 bytes 对齐。

10.2.5.29. 0x110 OUTPUT_ADDR0

默认值：0x00000000			DST 输出地址 0（Dst Output Address0）
位域	类型	默认值	描述
31:0	R/W	0x0	ADDR0 输出 plane0 地址

10.2.5.30. 0x120 DITHER_LINE_BUF_ADDR

默认值：0x00000000			Dither 行缓冲地址（Dither Line Buffer Address）
位域	类型	默认值	描述
31:0	R/W	0x0	DITHER_LINE_BUF_ADDR Dither 行缓冲地址

10.2.5.31. 0x130 CMD_BUF_START_ADDR

默认值：0x00000000			CMD BUF 起始地址（CMD Buffer Start Address）
位域	类型	默认值	描述
31:7	R/W	0x0	CMD_BUF_START_ADDR[31:7]

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			CMD BUF 起始地址 （CMD Buffer Start Address ）
位域	类型	默认值	描述
			命令队列缓冲区的起始地址
6:0	RO	0x0	固定为 0

10.2.5.32. 0x134 CMD_BUF_END_ADDR

默认值：0x00000000			CMD BUF 结束地址 （CMD Buffer End Address ）
位域	类型	默认值	描述
31:7	R/W	0x0	CMD_BUF_END_ADDR[31:7] 命令队列缓冲区的结束地址
6:0	RO	0x7F	每个 bit 固定全为 1

10.2.5.33. 0x0138 CMD_BUF_OFFSET

默认值：0x00000000			CMD BUF 地址偏移寄存器 (CMD Buffer Address Offset ）
位域	类型	默认值	描述
31:24	–	–	–
23:0	R/W	0x0000_0000	当前有效配置命令开始的位置，该位置是相对于 ADDR 计算的，以 bytes 为单位。

10.2.5.34. 0x013C CMD_BUF_VALID_LENGTH

默认值：0x00000000			CMD BUF 有效长度寄存器 (CMD Buffer Valid Length ）
位域	类型	默认值	描述
31:24	–	–	–
23:0	R/W	0x0000_0000	当前 Buffer 的有效长度，以 bytes 为单位，最大 16 MB，即命令缓冲区最大为 16 MB。 CMD Buffer 是一个 Ring Buffer，有效长度为 length。 当硬件读取长度为 length0（length0 < length ）的命令时，CMD 的读地址大于 CMD Buffer End Address，硬件会切换 CMD 读取地址 到 Address，继续读取长度为 length1 的命令。此时有效长度 length 由 length0 和 length1 两部分组成，length = length0 + length1。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.2.5.35. 0x200 SCALER0_CTRL

默认值：0x00000000			Scaler0 控制 （Scaler0 Control）
位域	类型	默认值	描述
31:1	–	–	–
0	R/W	0x0	EN Scaler0 使能控制 （采用 bilinear 滤波算法） • 0:关闭 • 1:开启

10.2.5.36. 0x210 SCALER_CH0_INPUT_SIZE

默认值：0x00000000			Scaler0 Chan0 输入宽高 （Scaler0 Ch0 Input Size）
位域	类型	默认值	描述
31:29	–	–	–
28:16	R/W	0x0	INPUT_HEIGHT Channel 0 输入高度
15:13	–	–	–
12:0	R/W	0x0	INPUT_WIDTH Channel 0 输入宽度

10.2.5.37. 0x214 SCALER_CH0_OUTPUT_SIZE

默认值：0x00000000			Chan0 输出宽高 （Scaler0 Ch0 Output Size）
位域	类型	默认值	描述
31:29	–	–	–
28:16	R/W	0x0	OUTPUT_HEIGHT Channel 0 输出高度
15:13	–	–	–
12:0	R/W	0x0	OUTPUT_WIDTH Channel 0 输出宽度

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.2.5.38. 0x218 SCALER0_CH0_H_INIT_PHASE

默认值：0x00000000			Scaler0 Chan0 水平初始相位（Scaler0 Ch0 H Init Phase）
位域	类型	默认值	描述
31:20	–	–	–
19:16	R/W	0x0	H_INIT_PHASE_INT 水平初始相位整数部分
15:0	R/W	0x0	H_INIT_PHASE_FRAC 水平初始相位小数部分

10.2.5.39. 0x21C SCALER0_CH0_H_RATIO

默认值：0x00000000			Scaler0 Chan0 水平缩放比例（Scaler0 Ch0 H Ratio）
位域	类型	默认值	描述
31:21	–	–	–
20:16	R/W	0x0	H_RATIO_INIT 水平缩放比例整数部分
15:0	R/W	0x0	H_RATIO_FRAC 水平缩放比例小数部分

10.2.5.40. 0x220 SCALER0_CH0_V_INIT_PHASE

默认值：0x00000000			Scaler0 Chan0 垂直初始相位（Scaler0 Ch0 V Init Phase）
位域	类型	默认值	描述
31:20	–	–	–
19:16	R/W	0x0	V_INIT_PHASE_INIT 垂直初始相位整数部分
15:0	R/W	0x0	V_INIT_PHASE_FRAC 垂直初始相位小数部分

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.2.5.41. 0x224 SCALER0_CH0_V_RATIO

默认值：0x00000000			Scaler0 Chan0 垂直缩放比例（Scaler0 Ch0 V Ratio）
位域	类型	默认值	描述
31:21	–	–	–
20:16	R/W	0x0	V_RATIO_INIT 垂直缩放比例整数部分
15:0	R/W	0x0	V_RATIO_FRAC 垂直缩放比例小数部分

10.3. Video Engine (VE)

Video Engine (VE) 是匠芯创科技自研的视频/图片编解码加速引擎，可支持多种视频和图片格式解码。

10.3.1. 特性说明

VE 引擎支持的视频解码格式如下：

表 10–4 规格说明

标准	Profile	Level	最大分辨率	最小分辨率	最大码率
JPEG decoder	Baseline	–	8192x8192	16x16	120 Mbps (YUV444)
PNG decoder	–	–	4096x4096	–	–

表 10–5 VE 各模块性能说明

特性	性能
JPEG 模块 • 兼容 ISO/IEC 10918–1 JPEG 基线 • 支持一个或者三个颜色分量 • 支持 8 bit 位深 • 支持 4:2:0、4:2:2、2:2:4、4:4:4 和 4:0:0 颜色格式，每个 MCU 包括最多六个 8x8 块 • 支持 1/2、1/4、1/8 缩放 • 支持 0、90、180、270 度旋转以及水平、垂直镜像 • 支持 RGB 颜色格式输出，包括 RGBA8888、BGRA8888、ARGB8888、ABGR8888、RGB888、BGR888、RGB565 和 BGR565 • 支持任意起始位置、任意矩形框大小 crop 输出	1080P@60fps

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

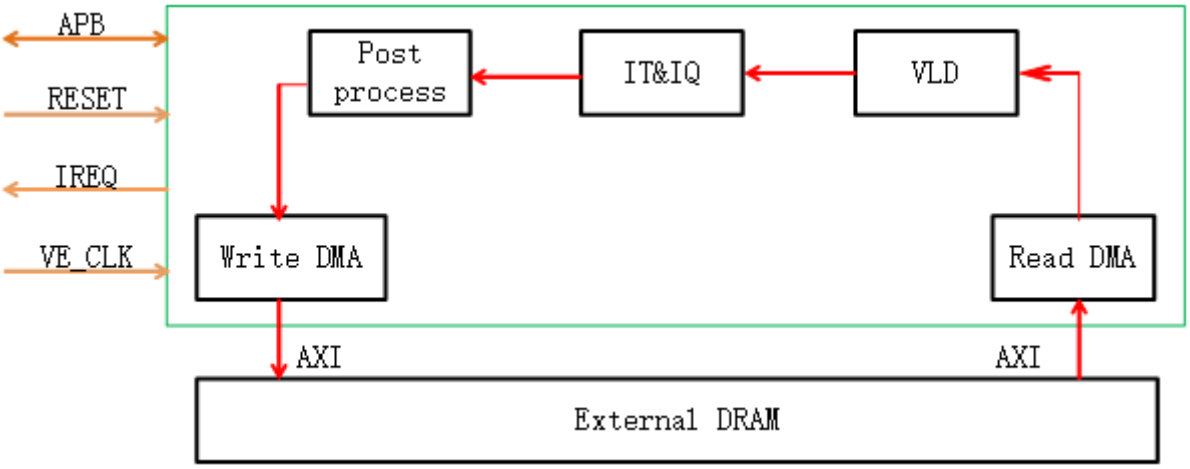
表 10–5 VE 各模块性能说明 (续)

特性	性能
注： 旋转、镜像不能与缩放同时开启。	
PNG 模块 • 支持 png8，存储方式为索引色存储，索引色位深只支持 8 bit，索引最多 256 色，支持通过数据块 tRNS 来设置索引透明度 • 支持 png24，每个像素包含 R、G 和 B 三个通道，每个通道 8 bits • 支持 png32，每个像素包含 R、G、B 和 alpha 四个通道，每个通道 8 bits • 支持 png 标准五种 filters，包括 none、sub、up、average 和 paeth • 支持标准的 zlib 和 gzip 解压缩，LZ77 最大窗口为 32K • png 图片支持任意起始位置、任意矩形框大小 crop 输出 • 对于 png8 图片，支持以索引值输出（最终由 DE 查索引表得到颜色值） **注：** 不支持 interleace 模式。	• 1080 P 纯色纹理：11 ms • 1080 P 普通 UI：42 ms • 1080 P 复杂纹理：90 ms

10.3.2. JPEG 模块

图 10–14 : JPEG 解码引擎框架示意图如下：

图 10–14 JPEG 解码引擎框架示意图



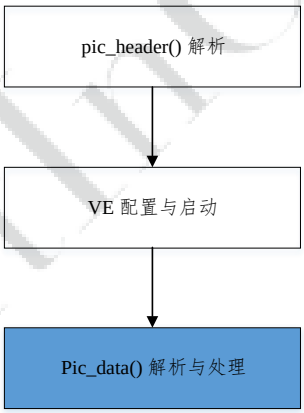
JPEG 解码流程如下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

1. VLD 解码模块从外部存储器读取原始码流，解码出 MCU 块系数信息，并将块系数信息传递给变换/量化模块，即 IT&IQ。
2. IT&IQ 根据接收到的块系数进行反量化和反变换。
3. 后处理模块读取反变换后的块数据，镜像旋转、镜像或下采样。
4. 最终解码得到的 MCU 数据通过 DMA 写入外部存储器。
5. 时序控制对各个子模块进行调度，确保解码过程的正确性与高效率，为了提高解码效率，各子模块之前采取并行的方式进行。

10.3.2.1. JPEG 文件解码流程

JPEG 图片解码启动一次硬件就可以完成一张图片的解码。具体流程如下：



10.3.2.2. JPEG 解码后处理流程

解码后处理支持缩放以及旋转镜像，但两个功能不能同时开启；同时也支持 RGB 颜色格式转换以及 Crop 输出。

各个功能详细描述如下：

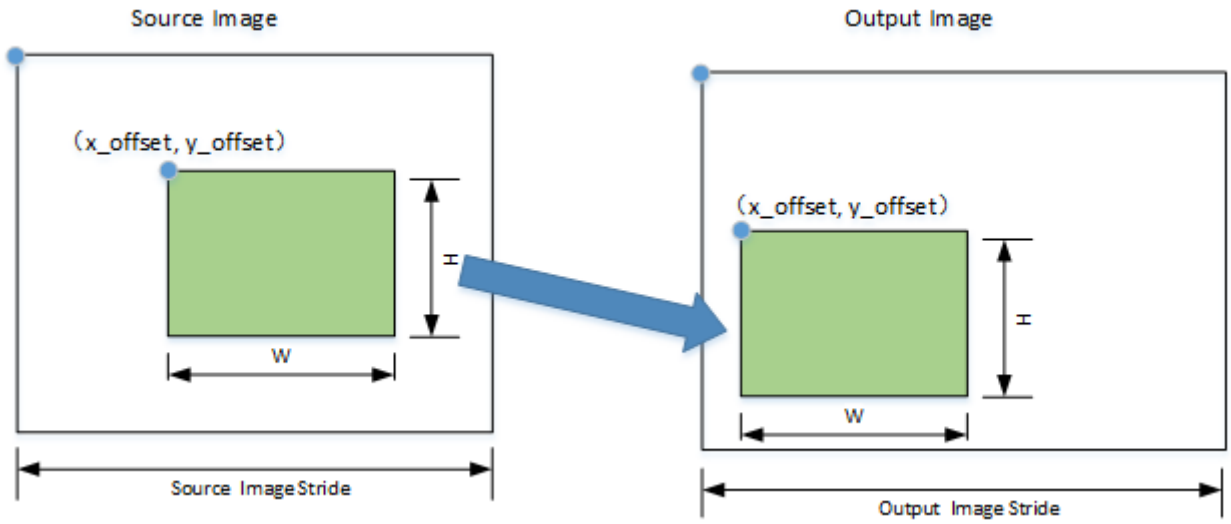
- 缩放：支持对解码后数据进行下采样，水平、垂直像素宽度可以分别缩小为原始的 1、1/2、1/4、1/8。
- 旋转和镜像：
  - 支持四种旋转模式，分别对应逆时针旋转 90 度 * n (n=0, 1, 2, 3)。
  - 支持两种镜像模式：水平镜像和垂直镜像。

  旋转与镜像可组合使用，关于具体说明，请查看旋转和镜像组合使用说明。
- 颜色格式转换模块：支持 YUV 格式转 RGB/ ARGB
- 误差扩散 Dither：把相邻像素的量化误差扩散到当前像素，从而使图像更平滑。

  当像素格式从高位宽转为低位宽时，输出颜色格式为 RGB565 或 BGR565，如果直接对数值做截断处理，图像的渐变区域会有明显的阶梯效应。 使用 Dither 算法可以使图像渐变区域更平滑。
- crop 功能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

解码后图像的矩形区域数据搬移到目标图像的指定矩形区域中。 只有 RGB/ARGB 颜色格式输出才支持 crop 功能。



10.3.2.3. 旋转和镜像组合使用说明

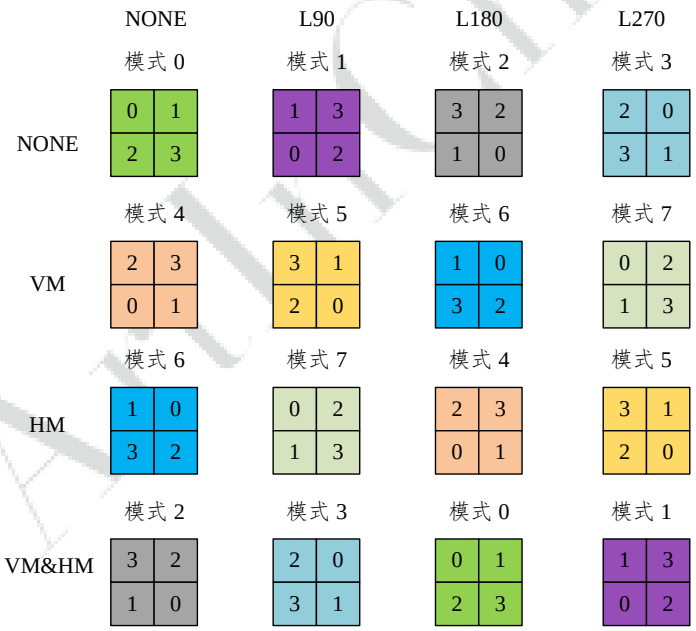
旋转与镜像是解码后处理流程支持的两个功能。旋转与镜像可组合使用，如下表所示。

当旋转和镜像组合使用时，先做镜像再做旋转。当旋转镜像模块被使能后，寄存器 JPEG_ROTMIR 有四个比特位用于工作模式的选择：两位代表选择的角度，另外两位分别代表水平与垂直镜像功能的使能，实际有效只有八种表现。

垂直镜像使能	水平镜像使能	旋转模式	寄存器值	模式
否	否	0	4'b0000	0
否	否	90	4'b0001	1
否	否	180	4'b0010	2
否	否	270	4'b0011	3
是	否	0	4'b0100	4
是	否	90	4'b0101	5
是	否	180	4'b0110	6
是	否	270	4'b0111	7
否	是	0	4'b1000	8
否	是	90	4'b1001	9
否	是	180	4'b1010	10
否	是	270	4'b1011	11

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

垂直镜像使能	水平镜像使能	旋转模式	寄存器值	模式
是	是	0	4'b1100	12
是	是	90	4'b1101	13
是	是	180	4'b1110	14
是	是	270	4'b1111	15



10.3.3. PNG 模块

PNG 解码模块可以细分为以下几个更小的功能单元：

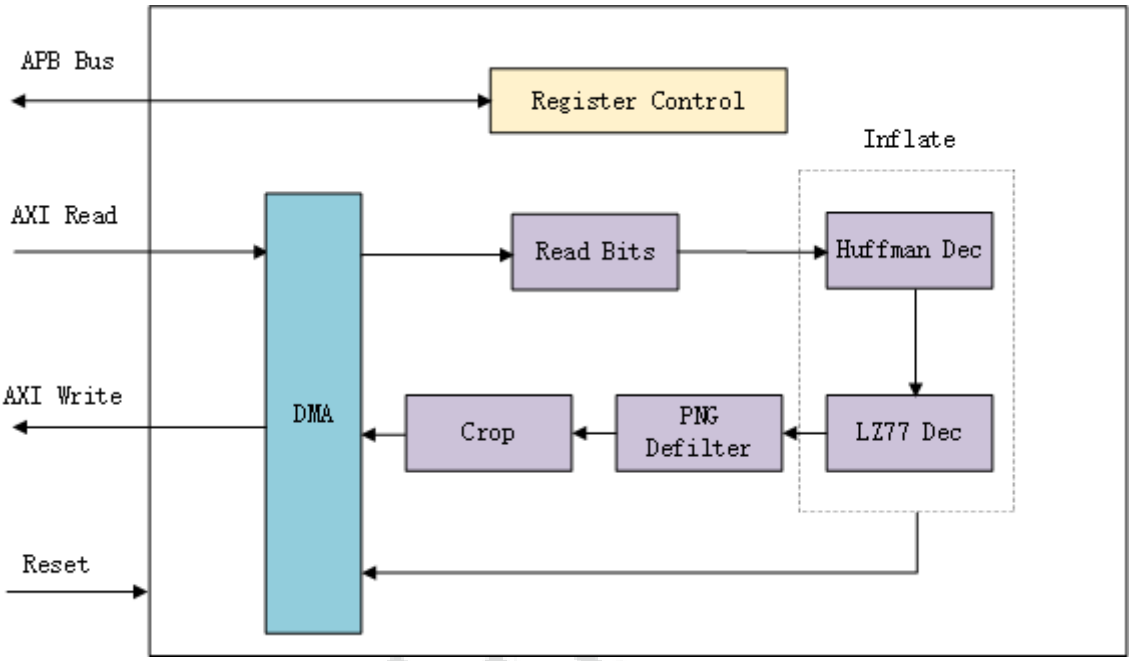
- Bits 读取模块
- Huffman 解码
- LZ77 Dec 模块
- PNG DeFilter 模块
- Crop 模块

注：

Zlib/ gzip 解压缩复用 PNG 解码模块。具体可查看 Zlib 解码和 Gzip 解码流程。

PNG 解码模块中各个功能单元的工作流程如图所示：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



10.3.3.1. PNG 文件结构说明

PNG 文件头	PNG 数据块...PNG 数据块

图 10–15 PNG 文件结构

如图所示，PNG 文件包括以下结构：

- PNG 文件头
- PNG 数据块：
  - 标准数据块
  - 辅助数据块

表 10–6 PNG 数据块说明

数据块符号	数据块名称	多数据块	是否可选	可获取的参数
IHDR	文件头数据块	否	否	PNG 图片的宽、高、位深、颜色类型、Interlace Method 等。关于 IHDR 的详细说明，可查看 IHDR 对 PNG 标注的支持说明。
PLTE	调色板数据块	否	否	–
tRNS	图像透明数据块	否	否	–
IDAT	图像数据块	多数据块	是	Zlib 格式的压缩码流数据。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 10–6 PNG 数据块说明 (续)

数据块符号	数据块名称	多数据块	是否可选	可获取的参数
				注： Zlib 压缩之前，可以采用 Filter 算法对原始数据进行处理，有助于提升压缩率。关于具体流程说明，可查看 Filter 算法说明。
IEND	图像结束数据块	否	否	–

10.3.3.1.1. IHDR 对 PNG 标注的支持说明

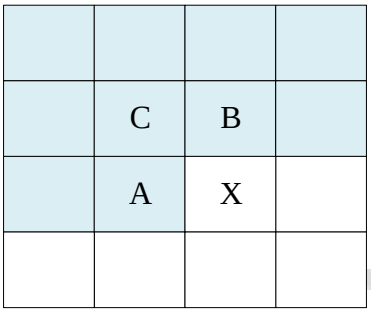
IHDR 中可以获取到 PNG 图片的宽、高、位深、颜色类型和 Interlace Method 等参数。本模块对 PNG 标准的支持情况如下：

域名	标准定义	本模块是否支持
位深	索引彩色图像：1，2，4，8	是
	灰度图像：1，2，4，8 或 16	否
	真彩色图像：8 或者 16	支持 8 bits，不支持 16 bits
颜色格式	0：灰度图像，1，2，4，8 或 16	否
	2：真彩色图像	是
	3：索引彩色图像，1，2，4 或 8	是
	4：带 α 通道数据的灰度图像，8 或 16	否
	6：带 α 通道数据的真彩色图像，8 或 16	支持 8 bits，不支持 16 bits
扫描方式	0：非隔行扫描	是
	1：Adam7	否

10.3.3.1.2. Filter 算法说明

PNG 图片的码流数据都在数据块 IDAT 中，IDAT 中是以 zlib 格式的压缩码流，在 zlib 压缩之前，可以采用 Filter 算法对原始数据进行处理，有助于提升压缩率。PNG 图像如下如所示，X 表示当前要进行 Filter 的像素，A 为当前像素左边像素，B 为当前像素上边像素，C 为当前像素左上边像素，则对 Fiter 的说明如下所示：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

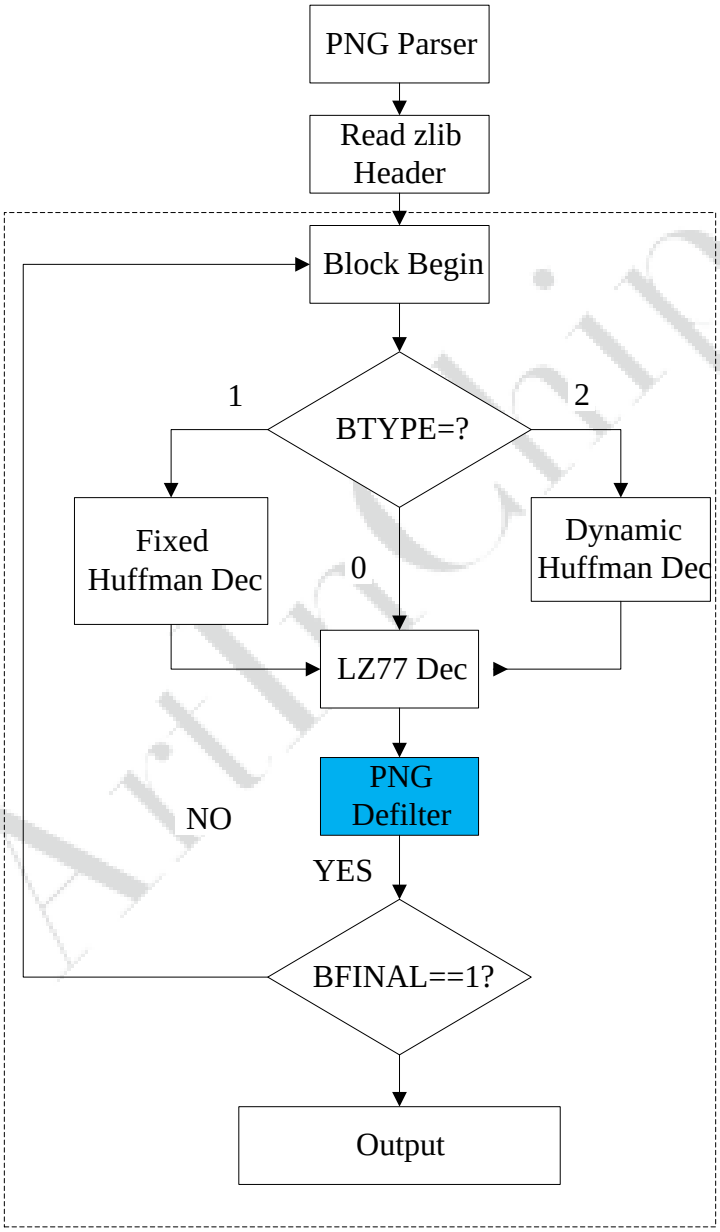


Filter 类型	函数	Filter 方式
0	无	保留原始数据
1	减	减去 A
2	上	减去 B
3	平均	根据 A 和 B 取平均，并向下取整
4	Paeth	使用最接近与 A + B – C 的 A、B 或者 C

10.3.3.2. PNG 文件解码流程

PNG 图像的解码流程分为以下几个部分，如图所示：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



1. 硬件实现：即上图中的虚线框部分。
2. 软件实现：即图中虚线框外的 PNG parser 和 Read zlib Header。

   图中蓝色部分是 PNG 解码和 zlib 解码不一样的地方：

   - 对于 zlib 解码：解码后，每一行都需要进行 Defilter 处理，恢复原始的像素。关于 zlib 解码的详细说明，可查看 Zlib 解码和 Gzip 解码流程。
   - 对于 PNG 解码：PNG 图片的 deflate 数据可能存放在多个 IDAT 数据块中，硬件不支持解析 IDAT，所以需与软件配合才能完成解码。
     a. 软件解析所有 IDAT 数据，并把 IDAT 中的 deflate 数据拷贝到一块物理连续 buffer 中。
     b. 硬件读取该 buffer 中的 deflate 数据并完成解码。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

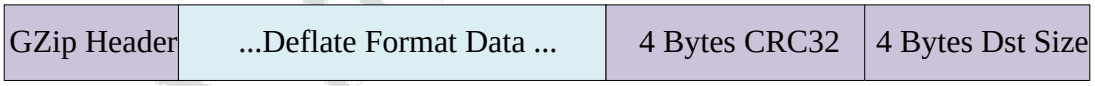
10.3.3.2.1. Zlib 解码和 Gzip 解码流程

Deflate 压缩格式有以下两种封装方式：

- Zlib 封装，结构如下：

  - CMF 是压缩方式的描述， FLG 中的 FDICT 标记了是否使用预先设置的字典，目前不支持 FDICT 为 1 的情况。ADLER32 为类似于 CRC32 的校验码。zlib 封装详细说明请参考 RFC1950。
  - Zlib 封装部分由软件解析，Deflate 码流部分，由硬件 Inflate 模块解析，硬件解码完成后生成对应的 ADLER32 校验码，然后由软件比较硬件生成的校验码和 zlib 格式封装中 ADLER32 是否一致。如果一致则说明 zlib 解码正确。

- Gzip 封装，结构如下：

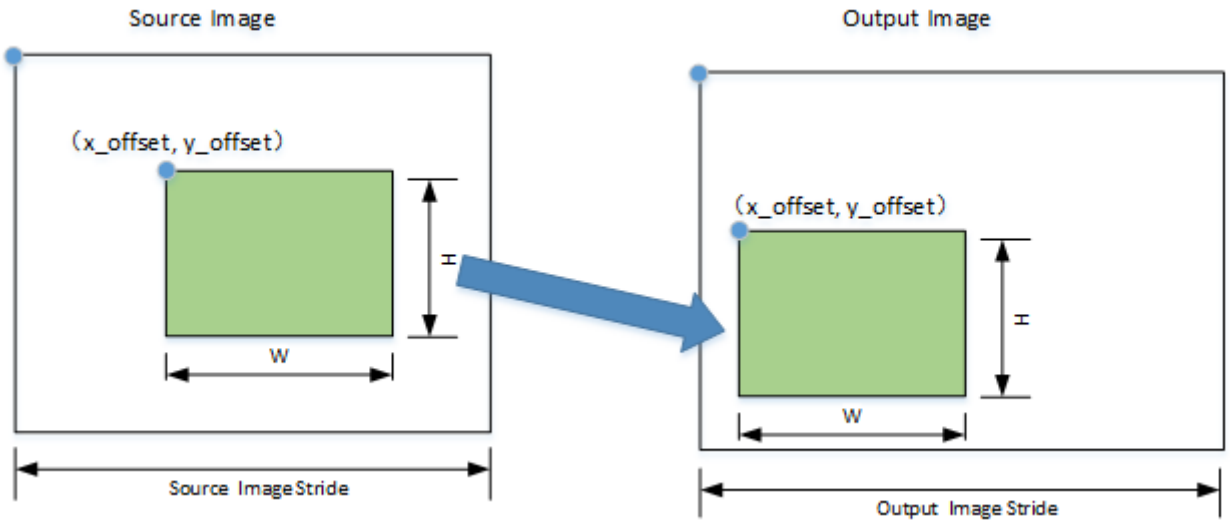


  - 对 Deflate 格式加入了头信息和尾部信息。gzip 对原始码流用 CRC32 校验，和 zlib 中的校验方式不同。并且尾部加入了 4 bytes 的原始码流的 size 信息。硬件解码时候，可以根据 size 的大小，来申请输出 Buffer。Gzip 封装的详细说明请参考 RFC1952。
  - Gzip 封装部分由软件解析，Deflate 码流部分，由硬件 Inflate 模块解析，硬件解码完成后生成对应的 CRC32 校验码，然后由软件比较硬件生成的校验码和 gzip 格式封装中 CRC32 是否一致。如果一致则说明 gzip 解码正确。

10.3.3.3. PNG 解码后处理流程

解码后图像的矩形区域数据搬移到目标图像的指定矩形区域中。 只有 RGB/ARGB 颜色格式输出才支持 crop 功能。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



10.3.4. 寄存器列表

VE 寄存器按功能模块划分不同地址空间范围：

模块名	地址范围
VE_TOP	0x18C0_0000–0x18C0_00FF
PNG	0x18C0_0100–0x18C0_01FF
JPEG	0x18C0_0200–0x18C0_03FF
PIC_INFO	0x18C0_0400–0x18C0_07FF

10.3.4.1. VE_TOP 寄存器列表

偏移	缩写	寄存器名称	章节
0x0000	VE_CLK	VE 时钟使能	0x0000 VE_CLK
0x0004	VE_RST	VE 模块复位	0x0004 VE_RST
0x0008	VE_INIT	VE 初始化	0x0008 VE_INIT
0x000C	VE_IRQ	VE 中断使能	0x000C VE_IRQ
0x0014	JPEG_EN	JPEG 模块使能	0x0014 JPEG_EN
0x0018	PNG_EN	PNG 模块使能	0x0018 PNG_EN
0x00FC	VE_VERSION	VE 版本信息	0x00FC VE_VERSION

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.3.4.2. JPEG 寄存器列表

偏移	缩写	寄存器名称	章节
0x200	JPEG_START	JPEG 启动解码	0x200 JPEG_START
0x204	JPEG_STATUS	JPEG 解码状态	0x204 JPEG_STATUS
0x208	JPEG_IRQ_EN	JPEG 中断使能 (JPEG Interrupt Enable)	0x208 JPEG_IRQ_EN
0x210	JPEG_START_POS	JPEG 解码起始位置	0x210 JPEG_START_POS
0x214	JPEG_SIZE	JPEG 图像大小	0x214 JPEG_SIZE
0x218	JPEG_CTRL	JPEG 模式控制	0x218 JPEG_CTRL
0x21C	JPEG_MCU_INFO	JPEG MCU 信息	0x21C JPEG_MCU_INFO
0x220	JPEG_HANDLE_NUM	JPEG 解码同时处理 MCU 个数	0x220 JPEG_HANDLE_NUM
0x224	JPEG_QMAT_INFO	JPEG 量化表信息	0x224 JPEG_QMAT_INFO
0x228	JPEG_QMAT_ADDR	JPEG 量化表地址	0x228 JPEG_QMAT_ADDR
0x22C	JPEG_QMAT_DATA	JPEG 量化表数据	0x22C JPEG_QMAT_DATA
0x230	JPEG_HUFF_INFO	JPEG Huffman 表信息配置	0x230 JPEG_HUFF_INFO
0x234	JPEG_HUFF_ADDR	JPEG Huffman 表地址	0x234 JPEG_HUFF_ADDR
0x238	JPEG_HUFF_DATA	JPEG Huffman 表数据	0x238 JPEG_HUFF_DATA
0x23C	JPEG_CLIP_EN	JPEG Clip 输出使能配置	0x23C JPEG_CLIP_EN
0x240	JPEG_CLIP_BASE	JPEG Clip 输出起始位置	0x240 JPEG_CLIP_BASE
0x244	JPEG_CLIP_SIZE	JPEG Clip 输出矩形框大小	0x244 JPEG_CLIP_SIZE
0x248	JPEG_CLIP_OUT_BASE	JPEG Clip 输出在缓存中的起始位置	0x248 JPEG_CLIP_OUT_BASE
0x24C	JPEG_ROTMIR	JPEG 旋转镜像配置	0x24C JPEG_ROTMIR
0x250	JPEG_SCALE	JPEG 缩放配置	0x250 JPEG_SCALE
0x254	JPEG_RGB_EN	JPEG RGB 颜色格式输出	0x254 JPEG_RGB_EN

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

偏移	缩写	寄存器名称	章节
0x25C	JPEG_RST_INTVAL	JPEG DC 系数重置间隔	0x25C JPEG_RST_INTVAL
0x268	JPEG_UV_INT	JPEG 解码输出 UV 分量交错配置	0x268 JPEG_UV_INT
0x26C	JPEG_CYCLES	JPEG 解码时钟数	0x26C JPEG_CYCLES
0x28C	JPEG_DITHER_PARAM	JPEG dither 参数配置	0x28C JPEG_DITHER_PARAM
0x300	JPEG_SUB_CTRL	JPEG 内部子模块控制	0x300 JPEG_SUB_CTRL
0x304	JPEG_MEM_IA	JPEG 内部子模块缓存中断地址	0x304 JPEG_MEM_IA
0x308	JPEG_MEM_HA	JPEG 内部子模块缓存暂停地址	0x308 JPEG_MEM_HA
0x31C	JPEG_RBIT_OFFSET	JPEG 读数据偏移	0x31C JPEG_RBIT_OFFSET
0x324	JPEG_MEM_SA	JPEG 内部子模块缓存起始地址	0x324 JPEG_MEM_SA
0x328	JPEG_MEM_EA	JPEG 内部子模块缓存结束地址	0x328 JPEG_MEM_EA
0x384	JPEG_STREAM_NUM	JPEG stream 个数	0x384 JPEG_STREAM_NUM
0x38C	JPEG_DATA_COUNT	JPEG 一次读写数据大小	0x38C JPEG_DATA_COUNT
0x390	JPEG_BUSY	JPEG 读数据状态	0x390 JPEG_BUSY
0x398	JPEG_STREAM_END_ADDR	JPEG 码流结束地址	0x398 JPEG_STREAM_END_ADDR
0x394	JPEG_REQ	JPEG 请求数据	0x394 JPEG_REQ
0x39C	JPEG_STREAM_START_ADDR	JPEG 码流起始地址	0x39C JPEG_STREAM_START_ADDR
0x3B0	JPEG_CUR_POS	JPEG 当前读数据位置	0x3B0 JPEG_CUR_POS

10.3.4.3. PNG 寄存器列表

偏移	缩写	寄存器名称	章节
0x0100	INFLATE_INT	Inflate 中断使能	0x0100 INFLATE_INT
0x0104	INFLATE_STATUS	Inflate 解码状态	0x0104 INFLATE_STATUS
0x0108	INFLATE_START	Inflate 解码开始	0x0108 INFLATE_START

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

偏移	缩写	寄存器名称	章节
0x010C	PNG_ERROR_STATE	PNG 解码错误状态	0x010C PNG_ERROR_STATE
0x0110	INFLATE_CTRL	Inflate 解码控制	0x0110 INFLATE_CTRL
0x0114	PNG_SIZE	PNG 图像大小	0x0114 PNG_SIZE
0x0118	PNG_OUTPUT_STRIDE	PNG 输出线宽	0x0118 PNG_OUTPUT_STRIDE
0x011C	PNG_OUTPUT_FORMAT	PNG 解码输出颜色格式	0x011C PNG_OUTPUT_FORMAT
0x0120	INPUT_BS_START_ADDR	输入码流起始地址	0x0120 INPUT_BS_START_ADDR
0x0124	INPUT_BS_END_ADDR	输入码流结束地址	0x0124 INPUT_BS_END_ADDR
0x0128	INPUT_BS_OFFSET	输入码流偏移	0x0128 INPUT_BS_OFFSET
0x012C	INPUT_BS_LENGTH	输入码流长度	0x012C INPUT_BS_LENGTH
0x0130	OUTPUT_BUF_ADDR	解码输出地址	0x0130 OUTPUT_BUF_ADDR
0x0134	OUTPUT_BUF_END_ADDR	解码输出缓存结束地址	0x0134 OUTPUT_BUF_END_ADDR
0x0138	OUTPUT_COUNT	解码输出数据长度	0x0138 OUTPUT_COUNT
0x013C	INFLATE_CHECK	解码输出数据校验码	0x013C INFLATE_CHECK
0x0140	INFLATE_WINDOW_BUF_ADDR	LZ77 窗口缓存地址	0x0140 INFLATE_WINDOW_BUF_ADDR
0x0144	PNG_PAL_BUF_ADDR	PNG 索引色缓存地址	0x0144 PNG_PAL_BUF_ADDR
0x0148	DATA_VALID	输入码流数据有效标志	0x0148 DATA_VALID
0x014C	PNG_PALETTE_ADDR	PNG 索引表地址	0x014C PNG_PALETTE_ADDR
0x0160	PNG_CROP_BASE	PNG CROP 的起始坐标	0x0160 PNG_CROP_BASE
0x0164	PNG_CROP_SIZE	PNG CROP 的窗口大小	0x0164 PNG_CROP_SIZE

10.3.4.4. 图像信息寄存器列表

偏移	缩写	寄存器名称	章节
0x458	RD_PRIO_L	0–7 号读总线优先级	0x458 RD_PRIO_L

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

偏移	缩写	寄存器名称	章节
0x45C	RD_PRIO_H	8–15 号读总线优先级	0x45C RD_PRIO_H
0x460	WR_PRIO_L	0–7 号写总线优先级	0x460 WR_PRIO_L
0x464	WR_PRIO_H	8–15 号写总线优先级	0x464 WR_PRIO_H
0x470	RD_QOS_L	0–7 号读总线 QOS	0x470 RD_QOS_L
0x474	RD_QOS_H	8–15 号读总线 QOS	0x474 RD_QOS_H
0x478	WR_QOS_L	0–7 号写总线 QOS	0x478 WR_QOS_L
0x47C	WR_QOS_H	8–15 号写总线 QOS	0x47C WR_QOS_H
0x600	PICTURE_FORMAT	图像格式信息	0x600 PICTURE_FORMAT
0x604	PICTURE_SIZE	图像大小	0x604 PICTURE_SIZE
0x608	PICTURE_LUMA_ADDR	图像亮度数据地址	0x608 PICTURE_LUMA_ADDR
0x60C	PICTURE_CB_ADDR	图像 Cb 分量数据地址	0x60C PICTURE_CB_ADDR
0x610	PICTURE_CR_ADDR	图像 Cr 分量数据地址	0x610 PICTURE_CR_ADDR

10.3.5. 寄存器描述

VE 寄存器按功能模块划分不同地址空间范围：

表 10–7 寄存器地址空间映射

模块名	地址范围
VE_TOP	0x18C0_0000–0x18C0_00FF
PNG	0x18C0_0100–0x18C0_01FF
JPEG	0x18C0_0200–0x18C0_03FF
PIC_INFO	0x18C0_0400–0x18C0_07FF

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.3.5.1. VE_TOP 寄存器描述

10.3.5.1.1. 0x0000 VE_CLK

默认值：0x00000000			VE 时钟使能 (VE CLOCK ENABLE)
位域	类型	默认值	描述
31:1	–	–	–
0	R/W	0x0	CLK_ENABLE 1：clk 使能

10.3.5.1.2. 0x0004 VE_RST

默认值：0x0000003f			VE 模块复位 (VE RESET)
位域	类型	默认值	描述
31:16	R	0x0	STATUS 重置的完成状态，0 表示重置 完成
15:8	–	–	–
7:6	R/W	0x3	RST_PIC 0：pic 模块寄存器重置，内部信号复位
5:4	R/W	0x3	RST_PNG 0：PNG 模块寄存器重置，内部信号复位
3:2	R/W	0x3	RST_JPEG 0：JPEG 模块寄存器重置，内部信号复位
1:0	–	–	–

10.3.5.1.3. 0x0008 VE_INIT

默认值：0x00000000			VE 初始化 (VE INITTIALIZE)
位域	类型	默认值	描述
31:1	–	–	–
0	R/W	0x0	INIT 1：VE 初始化

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.3.5.1.4. 0x000C VE_IRQ

默认值：0x00000000			VE 中断使能 (VE INTERRUPT ENABLE)
位域	类型	默认值	描述
31:1	–	–	–
0	R/W	0x0	IRQ_ENABLE 1：中断信号使能

10.3.5.1.5. 0x0014 JPEG_EN

默认值：0x00000000			JPEG 模块使能 (JPEG MODULE ENABLE)
位域	类型	默认值	描述
31:1	–	–	–
0	R/W	0x0	JPEG_EN 1：JPEG 模块使能

10.3.5.1.6. 0x0018 PNG_EN

默认值：0x00000000			PNG 模块使能 (PNG MODULE ENABLE)
位域	类型	默认值	描述
31:1	–	–	–
0	R/W	0x0	PNG_EN 1：PNG 模块使能

10.3.5.1.7. 0x00FC VE_VERSION

默认值：0x00000400			VE 版本信息 (VE Version)
位域	类型	默认值	描述
31:0	RO	0x00000400	VERSION VE 版本信息，V4.0

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.3.5.2. JPEG 寄存器描述

10.3.5.2.1. 0x200 JPEG_START

默认值：0x00000000			JPEG 启动解码 (JPEG Start)
位域	类型	默认值	描述
31:1	–	–	–
0	R/W	0x0	1：启动 JPEG 解码

10.3.5.2.2. 0x204 JPEG_STATUS

默认值：0x00000000			JPEG 解码状态 (JPEG Status)
位域	类型	默认值	描述
31:3	–	–	–
2	R	0x0	BIT_REQ 读完最后一笔数据后解码还未结束，则认为数据不够，VE 停止解码并向主控发送中断信号，该位置 1。
1	R	0x0	JPEG_ERROR 当执行当前任务过程中发现码流错误，则停止当前任务并向主控发送中断信号后，该位置 1。
0	R	0x0	JPEG_FINISH 解码结束时向主控发送中断信号后，该位置 1。

10.3.5.2.3. 0x208 JPEG_IRQ_EN

默认值：0x00000007			JPEG 中断使能 (JPEG Interrupt Enable)
位域	类型	默认值	描述
31:3	–	–	–
2	W1C	0x1	BIT_REQ_IRQ_EN 解码数据不够中断使能 写 1 清中断
1	W1C	0x1	ERROR_IRQ_EN 解码错误中断使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000007			JPEG 中断使能 (JPEG Interrupt Enable)
位域	类型	默认值	描述
			写 1 清中断
0	W1C	0x1	FINISH_IRQ_EN 解码正常结束中断使能 写 1 清中断

10.3.5.2.4. 0x210 JPEG_START_POS

默认值：0x00000000			JPEG 解码起始位置 (JPEG Start Position)
位域	类型	默认值	描述
31:28	–	–	–
27:16	R/W	0x0	STRAT_POS_Y 解码开始 Y 坐标
15:12	–	–	–
11:0	R/W	0x0	STRAT_POS_X 解码开始 X 坐标

10.3.5.2.5. 0x214 JPEG_SIZE

默认值：0x00000000			JPEG 图像大小 (JPEG Size)
位域	类型	默认值	描述
31:16	R/W	0x0	JPEG_HSIZE 图像宽度，需要按 MCU 对齐
15:0	R/W	0x0	JPEG_VSIZE 图像高度，需要按 MCU 对齐

10.3.5.2.6. 0x218 JPEG_CTRL

默认值：0x00000000			JPEG 模式控制 (JPEG Control)
位域	类型	默认值	描述
31:15	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			JPEG 模式控制 (JPEG Control)
位域	类型	默认值	描述
14:12	R/W	0x0	AC_HUFFMAN_IDX AC 系数 Huffman 表索引值
11	–	–	–
10:8	R/W	0x0	DC_HUFFMAN_IDX DC 系数 Huffman 表索引值
7:6	–	–	–
5	R/W	0x0	USR_HUFF_EN • 1：使用 DHT 中定义的 Huffman 表 • 0：使用默认 Huffman 表（VE 内部生成，不需要写 Huffman 表数据）
4:0	–	–	–

10.3.5.2.7. 0x21C JPEG_MCU_INFO

默认值：0x00000000			JPEG MCU 信息 (JPEG MCU Info)
位域	类型	默认值	描述
31:20	–	–	–
19:16	R/W	0x0	BLK_NUM 一个 MCU 中 8x8 blk 的数量
15	–	–	–
14:12	R/W	0x0	COMP_NUM 通道数量
11:10	R/W	0x0	Y_H_COUNT 一个 MCU 中 Y 分量水平方向上 8x8 块个数，对应 SOF 中解析的 h_count 信息。
9:8	R/W	0x0	Y_V_COUNT 一个 MCU 中 Y 分量垂直方向上 8x8 块个数，对应 SOF 中解析的 v_count 信息
7:6	R/W	0x0	CB_H_COUNT 一个 MCU 中 Cb 分量水平方向上 8x8 块个数，对应 SOF 中解析的 h_count 信息
5:4	R/W	0x0	CB_V_COUNT

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			JPEG MCU 信息 (JPEG MCU Info)
位域	类型	默认值	描述
			一个 MCU 中 Cb 分量垂直方向上 8x8 块个数，对应 SOF 中解析的 v_count 信息
3:2	R/W	0x0	CR_H_COUNT 一个 MCU 中 Cr 分量水平方向上 8x8 块个数，对应 SOF 中解析的 h_count 信息
1:0	R/W	0x0	CR_V_COUNT 一个 MCU 中 Cr 分量垂直方向上 8x8 块个数，对应 SOF 中解析的 v_count 信息

10.3.5.2.8. 0x220 JPEG_HANDLE_NUM

默认值：0x00000000			JPEG 解码同时处理 MCU 个数 (JPEG Handle Number)
位域	类型	默认值	描述
31:2	–	–	–
1:0	R/W	0x0	HANDLE_NUM_MINUS1 同时处理的 MCU 个数减 1，(12/ total_du_num_in_mcu)– 1，最大值为 4。

10.3.5.2.9. 0x224 JPEG_QMAT_INFO

默认值：0x00000000			JPEG 量化表信息 (JPEG Quant Table Info)
位域	类型	默认值	描述
31:8	–	–	–
7:6	R/W	0x0	QMAT_IDX • 0: Y 分量量化表 • 1: Cb 分量量化表 2:Cr 分量量化表
5:2	–	–	–
1	R/W	0x0	QMAT_AUTO 1: 使能地址自动加 1 模式
0	R/W	0x0	QMAT_EN 1: 使能该量化表

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.3.5.2.10. 0x228 JPEG_QMAT_ADDR

默认值：0x00000000			JPEG 量化表地址 (JPEG Quant Table Address)
位域	类型	默认值	描述
31:8	–	–	–
7:6	R/W	0x0	QMAT_IDX • 0: Y 分量量化表 • 1: Cb 分量量化表 • 2: Cr 分量量化表
5:0	R/W	0x0	QMAT_ADDR 量化表地址，取值范围 0–63 量化表地址按照 zigzag 扫描后的顺序，也就是 DQT 解析的顺序.

10.3.5.2.11. 0x22C JPEG_QMAT_DATA

默认值：0x00000000			JPEG 量化表数据 (JPEG Quant Table Data)
位域	类型	默认值	描述
31:0	R/W	0x0	QMAT_DATA 量化参数 **注：** 如果是编码，该值为 (1<<19)/q_data

10.3.5.2.12. 0x230 JPEG_HUFF_INFO

默认值：0x00000000			JPEG Huffman 表信息配置 (JPEG Huffman Table Info)
位域	类型	默认值	描述
31:12	–	–	–
11:10	R/W	0x0	HUFF_IDX Huffman 表索引值 • 0: Huffman 表每个分量每种码字长度的最小码字 • 1: Huffman 表每个分量每种码字长度的最大码字 • 2: Huffman 表每个分量每种码字长度对应数值在 buffer 的起始地址 • 3: Huffman 值

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			JPEG Huffman 表信息配置 (JPEG Huffman Table Info)
位域	类型	默认值	描述
9:2	–	–	–
1	R/W	0x0	HUFF_AUTO • 1: 使能地址自动加 1 模式 • 0: 不使能地址自动加 1 模式，每次通过 0x2088 寄存器写 Huffman 表数据前都需要通过 0x2084 寄存器配置地址。
0	R/W	0x0	HUFF_EN 1: 使能该 Huffman 表

10.3.5.2.13. 0x234 JPEG_HUFF_ADDR

默认值：0x00000000			JPEG Huffman 表地址 (JPEG Huffman Table Address)
位域	类型	默认值	描述
31:8	–	–	–
11:10	R/W	0x0	HUFF_IDX Huffman 表索引值 • 0: Huffman 表每个分量每种码字长度的最小码字 • 1: Huffman 表每个分量每种码字长度的最大码字 • 2: Huffman 表每个分量每种码字长度对应数值在 buffer 的起始地址 • 3: Huffman 值
9:0	R/W	0x0	HUFF_ADDR 如果 huff_idx 为 0/1/2，起始地址分别为 • DC_LUMA: 0 • DC_CHROMA: 16 • AC_LUMA: 32 • AC_CHROMA: 48 huff_idx 为 3 时： • DC_LUMA: 0; • DC_CHROMA: 12 •

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			JPEG Huffman 表地址 (JPEG Huffman Table Address)
位域	类型	默认值	描述
			• AC_LUMA: 24 • AC_CHROMA: 186

10.3.5.2.14. 0x238 JPEG_HUFF_DATA

默认值：0x00000000			JPEG Huffman 表数据 (JPEG Huffman Table Data)
位域	类型	默认值	描述
31:0	R/W	0x0	HUFF_DATA Huffman 表数据

10.3.5.2.15. 0x23C JPEG_CLIP_EN

默认值：0x00000000			JPEG Clip 输出使能配置 (JPEG Clip Enable)
位域	类型	默认值	描述
31:2	–	–	–
1	R/W	0x0	DEC_CLIP_EN 解码输出数据 clip 使能
0	R/W	0x0	ENC_CLIP_EN 编码输入数据 clip 使能

10.3.5.2.16. 0x240 JPEG_CLIP_BASE

默认值：0x00000000			JPEG Clip 输出起始位置 (JPEG Clip Base Address)
位域	类型	默认值	描述
31:16	R/W	0x0	CLIP_XBASE clip 窗口的起始 X 坐标，以像素为单位
15:0	R/W	0x0	CLIP_YBASE clip 窗口的起始 Y 坐标，以像素为单位

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.3.5.2.17. 0x244 JPEG_CLIP_SIZE

默认值：0x00000000			JPEG Clip 输出矩形框大小 (JPEG Clip SIZE)
位域	类型	默认值	描述
31:16	R/W	0x0	CLIP_XSIZE clip 矩形框的宽度，以像素为单位
15:0	R/W	0x0	CLIP_YSIZE clip 矩形框的高度，以像素为单位

10.3.5.2.18. 0x248 JPEG_CLIP_OUT_BASE

默认值：0x00000000			JPEG Clip 输出在缓存中的起始位置 (JPEG Clip OUTPUT BASE ADDRESS)
位域	类型	默认值	描述
31:16	R/W	0x0	CLIP_OUT_XBASE clip 起始位置在输出缓存中的 X 坐标，以字节为单位
15:0	R/W	0x0	CLIP_OUT_YBASE clip 起始位置在输出缓存中的 Y 坐标，以字节为单位

10.3.5.2.19. 0x24C JPEG_ROTMIR

默认值：0x00000000			JPEG 旋转镜像配置 (JPEG Rotate/Mirror Config)
位域	类型	默认值	描述
31:5	–	–	–
4	R/W	0x0	ROTMIR_EN • 0：不使能旋转镜像 • 1：使能旋转镜像
3	R/W	0x0	Horizontal MIRROR • 0：不使能水平镜像 • 1：使能水平镜像
2	R/W	0x0	Vertical MIRROR • 0：不使能垂直镜像 • 1：使能垂直镜像

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			JPEG 旋转镜像配置 (JPEG Rotate/Mirror Config)
位域	类型	默认值	描述
1:0	R/W	0x0	ROTATE • 0：不旋转 • 1：逆时针旋转 90 度 • 2：逆时针旋转 180 度 • 3：逆时针旋转 270 度

10.3.5.2.20. 0x250 JPEG_SCALE

默认值：0x00000000			JPEG 缩放配置 (JPEG Scaledown Config)
位域	类型	默认值	描述
31:5	–	–	–
4	R/W	0x0	SCALE_EN • 0：不使能缩放功能 • 1：使能缩放功能
3:2	R/W	0x0	H_SCALE 水平方向 scale 系数 • 0：不缩放 • 1：1/2 缩放 • 2：1/4 缩放 • 3：1/8 缩放
1:0	R/W	0x0	V_SCALE 垂直方向 scale 系数 • 0：不缩放 • 1：1/2 缩放 • 2：1/4 缩放 • 3：1/8 缩放

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.3.5.2.21. 0x254 JPEG_RGB_EN

默认值：0x00000000			JPEG RGB 颜色格式输出 (JPEG RGB FORMAT)
位域	类型	默认值	描述
31:16	–	–	–
15:8	R/W	0x0	ALPHA_VAL RGBA 颜色格式的 alpha 值
7:4	–	–	–
3:1	R/W	0x0	RGB_FORMAT • 0：ARGB8888 • 1：ABGR8888 • 2：RGBA8888 • 3：BGRA8888 • 4：RGB888 • 5：BGR888 • 6：RGB565 • 7：BGR565
0	R/W	0x0	RGB_OUT_ENABLE 1：RGB 颜色格式输出使能

10.3.5.2.22. 0x25C JPEG_RST_INTVAL

默认值：0x00000000			JPEG DC 系数重置间隔 (JPEG Restart Intval)
位域	类型	默认值	描述
31:0	R/W	0x0	RST_INTVAL 重置 DC 预测值的 MCU 间隔个数

10.3.5.2.23. 0x268 JPEG_UV_INT

注：

显示只支持 RGB 格式输出，如果解码和显示共用 buffer，应该配置 JPEG_RGB_EN 寄存器，寄存器 JPEG_UV_INT 无效。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			JPEG 解码输出 UV 分量交错配置 (JPEG UV Interleave)
位域	类型	默认值	描述
31:1	–	–	–
0	R/W	0x0	UV_INTERLEAVE • 0: UV 分量不交错存放 • 1: UV 分量交错存放

10.3.5.2.24. 0x26C JPEG_CYCLES

默认值：0x00000000			JPEG 解码时钟数 (JPEG Cycles)
位域	类型	默认值	描述
31:0	R/W	0x0	CYCLES Jpeg 解码用的时钟数

10.3.5.2.25. 0x28C JPEG_DITHER_PARAM

默认值：0x000000AE			JPEG dither 参数配置 (JPEG Dither Parameter Configuration)
位域	类型	默认值	描述
31:9	–	–	–
8:6	R/W	0x2	LEFTUP_WEIGHT_MINUS1 左上相邻像素对当前像素的误差扩散权重减 1
5:3	R/W	0x5	UP_WEIGHT_MINUS1 上相邻像素对当前像素的误差扩散权重减 1
2:0	R/W	0x6	LEFT_WEIGHT_MINUS1 左相邻像素对当前像素的误差扩散权重减 1

10.3.5.2.26. 0x300 JPEG_SUB_CTRL

注：

在启动 JPEG 解码前，即配置 JPEG Start 寄存器之前，需要先初始化内部子模块 1、2， 但两个模块不能同时初始化，必须分两个时刻配置，先配置 INIT1 再配置 INIT2 。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			JPEG 内部子模块控制 (JPEG SubModual Control)
位域	类型	默认值	描述
31:2	–	–	–
1	R/W	0x0	INIT1 1: 初始化内部子模块 1
0	R/W	0x0	INIT2 1: 初始化内部子模块 2

10.3.5.2.27. 0x304 JPEG_MEM_IA

默认值：0x00000000			JPEG 内部子模块缓存中断地址 (JPEG MEM Intr Address)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	INTR_ADDR 解码配置为 0。编码配置为 0x40

10.3.5.2.28. 0x308 JPEG_MEM_HA

默认值：0x00000000			JPEG 内部子模块缓存暂停地址 (JPEG MEM Halt Address)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	HALT_ADDR 解码配置为 0。编码配置为 0x40

10.3.5.2.29. 0x31C JPEG_RBIT_OFFSET

默认值：0x00000000			JPEG 读数据偏移 (JPEG Read Bit Offset)
位域	类型	默认值	描述
31:0	R/W	0x0	解码数据起始比特偏移

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.3.5.2.30. 0x324 JPEG_MEM_SA

默认值：0x00000000			JPEG 内部子模块缓存起始地址 (JPEG MEM Start Address)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	Start_ADDR 配置为 0

10.3.5.2.31. 0x328 JPEG_MEM_EA

默认值：0x00000000			JPEG 内部子模块缓存结束地址 (JPEG MEM End Address)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	END_ADDR 配置为 0x7f

10.3.5.2.32. 0x384 JPEG_STREAM_NUM

默认值：0x00000000			JPEG stream 个数 (JPEG Stream Number)
位域	类型	默认值	描述
31:0	R/W	0x0	STREAM_NUM 数据流个数 **注：** 以 256 byte 为单位。

10.3.5.2.33. 0x38C JPEG_DATA_COUNT

默认值：0x00000000			JPEG 一次读写数据大小 (JPEG Stream Data Count)
位域	类型	默认值	描述
31:0	R/W	0x0	DATA_CNT 固定配置为 64，64 个 32bit 数据，表示一次读写 256 字节数据。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.3.5.2.34. 0x390 JPEG_BUSY

默认值：0x00000000			JPEG 读数据状态 (JPEG Busy Status)
位域	类型	默认值	描述
31:1	–	–	–
0	R	0x0	BUSY 是否处于读数据状态

10.3.5.2.35. 0x398 JPEG_STREAM_END_ADDR

默认值：0x00000000			JPEG 码流结束地址 (JPEG Stream End Address)
位域	类型	默认值	描述
31:0	R/W	0x0	END_ADDR 码流结束地址，256 字节对齐

10.3.5.2.36. 0x394 JPEG_REQ

默认值：0x00000000			JPEG 请求数据 (JPEG Request Data)
位域	类型	默认值	描述
31:1	–	–	–
0	R/W	0x0	REQ_DATA 1: 请求一次数据

10.3.5.2.37. 0x39C JPEG_STREAM_START_ADDR

默认值：0x00000000			JPEG 码流起始地址 (JPEG Stream Start Address)
位域	类型	默认值	描述
31:0	R/W	0x0	Start_ADDR 码流起始地址，8 字节对齐

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.3.5.2.38. 0x3B0 JPEG_CUR_POS

默认值：0x00000000			JPEG 当前读数据位置 (JPEG Current Postion)
位域	类型	默认值	描述
31:0	R/W	0x0	CUR_POS 当前读码流数据位置 **注：** JPEG 读数据以 256 byte 为单位，当前读码流数据位置也以 256 byte 为单位。比如，需要读的数据从 100 byte 开始，则 CUR_POS 为 0。

10.3.5.3. PNG 寄存器描述

10.3.5.3.1. 0x0100 INFLATE_INT

默认值：0x00000000			Inflate 中断使能 (Inflate Interrupt Enable)
位域	类型	默认值	描述
31:3	–	–	–
2	R/W	0x0	BIT_REQ_INT_EN 请求 bitsteam 中断控制 • 0：不使能 • 1：使能 DEC_REQ_INT_EN 用于控制解码所需要的原始码流不够需要请求下笔数据时是否向 CPU 发出中断请求。 当 DEC_REQ_INT_EN 为 1 时，产生中断请求，否则不产生中断请求。是否产生中断请求，不影响相应状态标志位的产生。
1	R/W	0x0	DEC_ERR_INT_EN 解码 error 中断控制 • 0：不使能 • 1：使能 DEC_ERR_INT_EN 用于控制解码发生错误结束后是否向 CPU 发出中断请求。 当 DEC_ERR_INT_EN 为 1 时，产生中断请求，否则不产生中断请求。是否产生中断请求，不影响相应状态标志位的产生。
0	R/W	0x0	DEC_FINISH_INIT 解码结束中断控制

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			Inflate 中断使能 (Inflate Interrupt Enable)
位域	类型	默认值	描述
			• 0：不使能 • 1：使能 DEC_FINISH_INT_EN 用于控制正常解码结束后是否向 CPU 发出中断请求。当 DEC_FINISH_INT_EN 为 1 时，产生中断请求，否则不产生中断请求。是否产生中断请求，不影响相应状态标志位的产生。

10.3.5.3.2. 0x0104 INFLATE_STATUS

默认值：0x00000000			Inflate 解码状态 (Inflate Status)
位域	类型	默认值	描述
31:5	–	–	–
4	R	0x0	ERROR • 0：解码正确 • 1：解码出错，具体错误原因参考 0x010C PNG_ERROR_STATE 寄存器
3	–	–	–
2	R/W1C	0x0	BIT_REQ_INT_FLAG 请求 bitstream 标记状态 当执行当前任务过程中发现码流不够，则暂停当前任务并向 CPU 发送中断信号后，该位置 1。软件向该位写 1 则清除相应中断。
1	R/W1C	0x0	ERR_INIT_FLAG 解码 error 标记状态 当执行当前任务过程中发现码流错误，则停止当前任务并向 CPU 发送中断信号后，该位置 1。软件向该位写 1 则清除相应中断
0	R/W1C	0x0	FINISH_INIT_FLAG 解码结束标记状态 当完成当前任务并向 CPU 发送中断信号后，该位置 1。主控向该位写 1 则清除相应中断。

默认值：0x00000000			Inflate 解码状态 (Inflate Status)
位域	类型	默认值	描述
31:24	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			Inflate 解码状态 (Inflate Status)
位域	类型	默认值	描述
23:22	R	0x0	DEFILTER_ERR • [23]: ◦ 0 表示 defilter 解码数据足够 ◦ 1 表示 defilter 解码数据不足 • [22]: ◦ 0 表示 filter 类型正确 ◦ 1 表示 filter 类型错误
21:19	R	0x0	WDMA_ERR 写 DMA 出错
18:15	R	0x0	RDMA_ERR 读 DMA 出错
14:11	R	0x0	LZ77_ERR LZ77 解码 error • [14]: ◦ 0：LZ77 解码数据足够 ◦ 1：LZ77 解码数据不足 • [13]: ◦ 0：LZ77 huffman 码值正确 ◦ 1：LZ77 huffman 码值错误 • [12]: ◦ 0：LZ77 huffman 码表 1 解码正确 ◦ 1：LZ77 huffman 码表 1 解码错误; • [11]: ◦ 0：LZ77 huffman 码表 2 解码正确 ◦ 1：LZ77 huffman 码表 2 解码错误
10:8	R	0x0	HUFF_ERR huffman 解码 error • [10]: ◦ 0：cl 个数正确 ◦ 1： cl 个数不匹配 • [9]:

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			Inflate 解码状态 (Inflate Status)
位域	类型	默认值	描述
			◦ 0：cl huffman 解码正确 ◦ 1：cl huffman 解码错误 • [8]： ◦ 0：ccl 解码数据足够 ◦ 1：ccl 解码数据不足
7:5	R	0x0	PC_ERR PNG 文件头解码错误 • [7]： ◦ 0：deflate 文件头解码数据足够 ◦ 1：deflate 文件头解码数据不足 • [6]： ◦ 0：deflate 文件头参数正确 ◦ 1：deflate 文件头参数错误 • [5]： ◦ 0：deflate 文件头解码正确 ◦ 1：deflate 文件头解码错误
4	R	0x0	OUT_BUF_OVERFLOW • 0：无 • 1：当前 error 的原因是输出 buffer 已满
3	–	–	–
2	R/W1C	0x0	BIT_REQ_INT_FLAG 请求 bitstream 标记状态 当执行当前任务过程中发现码流不够，则暂停当前任务并向 CPU 发送中断信号后，该位置 1。软件向该位写 1 则清除相应中断。
1	R/W1C	0x0	ERR_INIT_FLAG 解码 error 标记状态当执行当前任务过程中发现码流错误，则停止当前任务并向 CPU 发送中断信号后，该位置 1。软件向该位写 1 则清除相应中断
0	R/W1C	0x0	FINISH_INIT_FLAG 解码结束标记状态当完成当前任务并向 CPU 发送中断信号后，该位置 1。主控向该位写 1 则清除相应中断。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.3.5.3.3. 0x0108 INFLATE_START

默认值：0x00000006			Inflate 解码开始 (Inflate START)
位域	类型	默认值	描述
31:3	–	–	–
2	R/W	0x1	STOP_DMA_EN • 0：PNG 解码出错后，直接向 CPU 发送 error 中断，不等待 DMA 数据请求完成 • 1：PNG 解码出错后，拦截 DMA 数据请求后再向 CPU 发送 error 中断
1	R/W	0x1	WAIT_DMA_EN • 0：PNG 解码出错后，直接向 CPU 发送 error 中断 • 1：PNG 解码出错后，等待 256 个 cycles 再向 CPU 发送 error 中断
0	R/W	0x0	START • 0：未开始解码 • 1：开始解码 写 1 会自动清零

10.3.5.3.4. 0x010C PNG_ERROR_STATE

默认值：0x00000000			PNG 解码错误状态 (PNG Error State)
位域	类型	默认值	描述
31:3	–	–	–
26	RO	0x0	DEFILTER_ERR • 0：defilter 处理正确 • 1：defilter 处理错误
25	RO	0x0	INFLATE_ERR • 0：inflate 解码正确 • 1：inflate 解码错误
24	RO	0x0	BS_LAST_ERR • 0：码流数据可以正确解码 • 1：码流数据不够导致出错
23:16	RO	0x0	DEFILTER_FSM_STATE

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			PNG 解码错误状态 (PNG Error State)
位域	类型	默认值	描述
			Defilter 模块当前所处状态
15:8	RO	0x0	INFLATE_BLOCK_NUM 当前正在解码的 Inflate 块序号
7:4	RO	0x0	INFLATE_ERROR_STATE Inflate 解码出错状态
3:0	RO	0x0	INFLATE_FSM_STATE Inflate 模块当前所处状态

10.3.5.3.5. 0x0110 INFLATE_CTRL

默认值：0x00000600			Inflate 解码控制 (Inflate Control)
位域	类型	默认值	描述
31:13	–	–	–
12:11	R/W	0x0	BIT_DEPTH 位宽 0: 8 bits 当 COLOR_TYPE 为 3 时的索引位宽， 只支持 8 bits。
10:8	R/W	0x6	COLOR_TYPE 格式类型 • 0：reserved • 1：reserved • 2: PNG24 • 3: 索引彩色图像 • 4: reserved • 5: reserved • 6: PNG32
7:4	–	–	–
3:2	R/W	0x0	CHECK_FUNCTION

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000600			Inflate 解码控制 (Inflate Control)
位域	类型	默认值	描述
			• 0：无 • 1：ADLER32 • 2: CRC32 当 CHECK_FUNCTION 为 1 时，Inflate 解码同时输出 ADLER32 校验值，当 CHECK_FUNCTION 为 2 时，Inflate 解码同时输出 CRC32 校验值
1:0	R/W	0x0	DEC_TYPE_FLAG 解码类型标记 • 0：Inflate (gzip) • 1：Inflate and png defilter 当解码类型为 Inflate and pngdefilter，需要设置 PNG 相关参数。

10.3.5.3.6. 0x0114 PNG_SIZE

默认值：0x00000000			PNG 图像大小 (PNG Picture Size)
位域	类型	默认值	描述
31:29	–	–	–
28:16	R/W	0x0	HEIGHT PNG 图片高度
15:13	–	–	–
12:0	R/W	0x0	WIDTH PNG 图片宽度

10.3.5.3.7. 0x0118 PNG_OUTPUT_STRIDE

默认值：0x00000000			PNG 输出线宽 (PNG Line Stride)
位域	类型	默认值	描述
31:15	–	–	–
14:0	R /W	0x0	STRIDE PNG 输出对齐

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.3.5.3.8. 0x011C PNG_OUTPUT_FORMAT

默认值：0x00000000			PNG 解码输出颜色格式 (PNG Output Format)
位域	类型	默认值	描述
31:6	–	–	–
5	R/W	0x0	LINEBUF_EN • 0：不使能内部存放 linebuffer 的 SRAM • 1：使能内部存放 linebuffer 的 SRAM
4	R/W	0x0	DEFILTER_TYPE_CHECK_EN • 0：检测到 defilter type 超出数值范围后不报错，继续解码 • 1：检测到 defilter type 超出数值范围后报错
3	R/W	0x0	PNG8_PALETTE_BYPASS • 0：PNG8 解码得到索引值后,通过 palette 表索引得到最终的像素值 • 1：PNG8 解码得到索引值后直接输出，跳过 palette 表映射
2:0	R/W	0x0	OUTPUT_FORMAT • 0: ARGB8888 • 1: ABGR8888 • 2: RGBA8888 • 3: BGRA8888 • 4: RGB888 • 5: BGR888 PNG8、PNG24 和 PNG32 都可以设置以上任意像素格式： • 当 PNG 原图中不包含 alpha 通道时，选择有 alpha 通道的格式，alpha 值固定为 0xFF。 • 当 PNG 原图中包含 alpha 通道时，输出格式中不含 alpha，则 alpha 信息丢失。 • 当 PNG8_PALETTE_BYPASS 为 1 时，输出格式 OUTPUT_FORMAT 设置无效。

10.3.5.3.9. 0x0120 INPUT_BS_START_ADDR

默认值：0x00000000			输入码流起始地址 (Input Bitstream Start Address)
位域	类型	默认值	描述
31:4	R/W	0x000_0000	INPUT_BS_START_ADDR[31:4] 原始码流缓冲区起始位置，低 4 位固定为 0。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			输入码流起始地址 (Input Bitstream Start Address)
位域	类型	默认值	描述
3:0	RO	0x00	固定为 0

10.3.5.3.10. 0x0124 INPUT_BS_END_ADDR

默认值：0x0000000F			输入码流结束地址 (Input Bitstream End Address)
位域	类型	默认值	描述
31:4	R/W	0x000_0000	INPUT_BS_END_ADDR[31:4] 原始码流缓冲区结束位置
3:0	RO	0xF	固定为全 1

10.3.5.3.11. 0x0128 INPUT_BS_OFFSET

默认值：0x00000000			输入码流偏移 (Input Bitstream Offset)
位域	类型	默认值	描述
31:30	–	–	–
29:0	R/W	0x0000_0000	BIT_OFFSET 当前有效配置数据开始的位置，该位置是相对于 ADDR 计算的，以 bit 为单位。最大 128 MB，即码流缓冲区最大 128 MB。

10.3.5.3.12. 0x012C INPUT_BS_LENGTH

默认值：0x00000000			输入码流长度 (Input Bitstream Length)
位域	类型	默认值	描述
31:30	–	–	–
29:0	R/W	0x0000_0000	BIT_LENGTH 当前配置的有效数据长度，以 bit 为单位。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.3.5.3.13. 0x0130 OUTPUT_BUF_ADDR

默认值：0x00000000			解码输出地址 (Output Buffer Address)
位域	类型	默认值	描述
31:0	R/W	0x000_0000	解码输出 buffer 地址

10.3.5.3.14. 0x0134 OUTPUT_BUF_END_ADDR

默认值：0x00000000			解码输出缓存结束地址 (Output Buffer End Address)
位域	类型	默认值	描述
31:0	R/W	0x000_0000	解码输出 buffer 结束地址

10.3.5.3.15. 0x0138 OUTPUT_COUNT

默认值：0x00000000			解码输出数据长度 (Output Data Count)
位域	类型	默认值	描述
31:0	RO	0x000_0000	解码输出到 OUTPUT_BUF_ADDR 的数据，以 byte 为单位的计数，启动解码后将从零开始计数。

10.3.5.3.16. 0x013C INFLATE_CHECK

默认值：0x00000000			解码输出数据校验码 (Inflate CheckSum)
位域	类型	默认值	描述
31:0	RO	0x000_0000	当打开 Inflate 解码校验功能时候，解码结束后，可以从此寄存器读取 ADLER32 或者 CRC32 校验值。

10.3.5.3.17. 0x0140 INFLATE_WINDOW_BUF_ADDR

默认值：0x00000000			LZ77 窗口缓存地址 (Inflate Window Buffer Address)
位域	类型	默认值	描述
31:3	R/W	0x000_0000	LZ77 窗口 buffer 地址的高 29 bits LZ77 的 DICT Buffer 大小为 32K bytes。
2:0	RO	0x00	LZ77 窗口 buffer 地址低 3 bits 固定为 0。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.3.5.3.18. 0x0144 PNG_PAL_BUF_ADDR

默认值：0x00000000			PNG 索引色缓存地址 (PNG Pallete Buffer Address)
位域	类型	默认值	描述
31:3	R/W	0x000_0000	PNG 索引色缓存 buffer 地址的高 29 bits。
2:0	RO	0x00	缓存 buffer 地址低 3 bits 固定为 0。

注：

索引色缓存地址在以下两种情况下使用：

- 索引色图片解码时，需要使用该 buffer 缓存上一行索引数据 (index)，buffer 大小为图像宽。
- 图像 crop 输出时，需要使用该 buffer 缓存上一行的像素数据 (RGB/ RGBA)，buffer 大小为图像宽 * 4 (RGBA) 或图像宽 * 3 (RGB)。

为了减少 VE 访问 DRAM 的带宽，VE 内部增加一个 4KB 大小的 SRAM，用于替换该缓存。

- 当图像宽小于或等于 1024 时，可以开启 0x11c 寄存器的 linebuf_en 位，使能内部 SRAM 用于 存放缓存数据，此时不会使用 PNG_PAL_BUF 配置的地址。
- 当图像大于 1024 时，内部 SRAM 不够用， 此时必须配置 PNG_PAL_BUF 地址，将数据缓存于外部 DRAM。

10.3.5.3.19. 0x0148 DATA_VALID

默认值：0x00000000			输入码流数据有效标志 (Data Valid Flag)
位域	类型	默认值	描述
31	R/W	0x0	DATA_VALID • 置 1：当前配置的码流数据已经有效写入原始码流缓冲区，VE 可以预取码流数据到内部的缓冲区，等待启动解码。 • 置 0：码流数据还没有正确写入原始码流缓冲区。
30:2	–	–	–
1	R/W	0x0	DATA_LAST • 置 1：当前配置的数据是一个完整解码的最后一笔数据。 • 置 0：当前配置的数据不是一个完整解码的最后一笔数据，当前解码任务完成后需要硬件保持状态，待下笔数据配置后继续完成一个完整解码。
0	R/W	0x0	DATA_FIRST

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			输入码流数据有效标志 (Data Valid Flag)
位域	类型	默认值	描述
			• 置 1：当前配置的数据是一个完整解码的第一笔数据。 • 置 0：当前配置的数据不是一个完整解码的第一笔数据，当前解码任务需要承接前一次的解码继续完成一个完整解码。

10.3.5.3.20. 0x014C PNG_PALETTE_ADDR

默认值：0x00000000			PNG 索引表地址 (PNG Palette Buffet Address)
位域	类型	默认值	描述
31:3	R/W	0x000_0000	PNG 调色板地址的高29bits PNG 调色板空间的大小为 1K 调色板空间的大小为 1K bytes，按照索引顺序排列，包含 A、R、G 和 B 四个通道，总共可以索引 256 色。
2:0	RO	0x00	PNG 调色板地址低 3 bits 固定为 0。

10.3.5.3.21. 0x0160 PNG_CROP_BASE

默认值：0x00000000			PNG CROP 的起始坐标 (Png Crop Base Position)
位域	类型	默认值	描述
31:29	–	–	–
28:16	R/W	0x0	CROP_BASE_X 图片 CROP 输出的起始 X 坐标（以像素为单位）
15:13	–	–	–
12:0	R/W	0x0	CROP_BASE_Y 图片 CROP 输出的起始 Y 坐标（以像素为单位）

10.3.5.3.22. 0x0164 PNG_CROP_SIZE

默认值：0x00000000			PNG CROP 的窗口大小 (Png Crop Size)
位域	类型	默认值	描述
31	R/W	0x0	CROP_ENCROP 功能使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			PNG CROP 的窗口大小 (Png Crop Size)
位域	类型	默认值	描述
30:29	–	–	–
28:16	R/W	0x0	CROP_SIZE_X 图片 CROP 窗口的宽（以像素为单位）
15:13	–	–	–
12:0	R/W	0x0	CROP_SIZE_Y 图片 CROP 窗口的高（以像素为单位）

10.3.5.4. 图像信息寄存器描述

PICTURE_INFO 寄存器用于配置 JPEG 解码输出的图像信息以及总线优先级和 QOS 信息。

读写总线优先级寄存器包括：

- 0x458 RD_PRIO_L
- 0x45C RD_PRIO_H
- 0x460 WR_PRIO_L
- 0x464 WR_PRIO_H

总线 QOS 相关寄存器如下：

- 0x470 RD_QOS_L
- 0x474 RD_QOS_H
- 0x478 WR_QOS_L
- 0x47C WR_QOS_H

10.3.5.4.1. 0x458 RD_PRIO_L

默认值：0x00000000			0–7 号读总线优先级 (Read Bus Priority)
位域	类型	默认值	描述
31:0	R/W	0x0	RD_PRIO_L • [3:0]: 0 号读总线优先级 • [31:29]: 7 号读总线优先级

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.3.5.4.2. 0x45C RD_PRIO_H

默认值：0x00000000			8–15 号读总线优先级 (Read Bus Priority)
位域	类型	默认值	描述
31:0	R/W	0x0	RD_PRIO_H • [3:0]: 8 号读总线优先级 • [31:29]: 15 号读总线优先级

10.3.5.4.3. 0x460 WR_PRIO_L

默认值：0x00000000			0–7 号写总线优先级 (Write Bus Priority)
位域	类型	默认值	描述
31:0	R/W	0x0	WR_PRIO_L • [3:0]: 0 号写总线优先级 • [31:29]: 7 号写总线优先级

10.3.5.4.4. 0x464 WR_PRIO_H

默认值：0x00000000			8–15 号写总线优先级 (Write Bus Priority)
位域	类型	默认值	描述
31:0	R/W	0x0	WR_PRIO_H • [3:0]: 8 号写总线优先级 • [31:29]: 15 号写总线优先级

10.3.5.4.5. 0x470 RD_QOS_L

默认值：0x00000000			0–7 号读总线 QOS (Read Bus Qos)
位域	类型	默认值	描述
31:0	R/W	0x0	RD_QOS_L • [3:0]: 0 号读总线 QOS • [31:29]: 7 号读总线 QOS

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.3.5.4.6. 0x474 RD_QOS_H

默认值：0x00000000			8–15 号读总线 QOS (Read Bus Qos)
位域	类型	默认值	描述
31:24	–	–	–
23:20	R/W	0x0	PNG_NB_QOS PNG 读相邻数据总线 QOS
19:16	R/W	0x0	PNG_PLTE_QOS PNG 读索引表总线 QOS
15:12	R/W	0x0	PNG_DBUF_QOS PNG 读字典总线 QOS
11:8	R/W	0x0	PNG_BT_QOS PNG 读码流总线 QOS
7:4	–	–	–
3:0	R/W	0x0	JEPG_QOS JPEG 读码流总线 QOS

10.3.5.4.7. 0x478 WR_QOS_L

默认值：0x00000000			0–7 号写总线 QOS (Write Bus Qos)
位域	类型	默认值	描述
31:28	R/W	0x0	JPEG_MB_QOSJPEG 输出解码数据总线 QOS
27:0	–	–	–

10.3.5.4.8. 0x47C WR_QOS_H

默认值：0x00000000			8–15 号写总线 QOS (Write Bus Qos)
位域	类型	默认值	描述
31:12	–	–	–
11:8	R/W	0x0	PNG_IDX_QOS PNG 写索引值总线 QOS

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			8–15 号写总线 QOS (Write Bus Qos)
位域	类型	默认值	描述
7:4	R/W	0x0	PNG_DOUT_QOS PNG 读码流总线 QOS
3:0	R/W	0x0	PNG_DBUF_QOSPNG 写字典总线 QOS

图像信息的五个寄存器如下所示：

- 0x600 PICTURE_FORMAT
- 0x604 PICTURE_SIZE
- 0x608 PICTURE_LUMA_ADDR
- 0x60C PICTURE_CB_ADDR
- 0x610 PICTURE_CR_ADDR

10.3.5.4.9. 0x600 PICTURE_FORMAT

默认值：0x00000000			图像格式信息 (Picture Format)
位域	类型	默认值	描述
31:20	–	–	–
19:17	R/W	0x0	COLOR_MODE • 0: YUV420 • 1: YUV422 • 2: YUV224 • 3: YUV444 • 4: YUV400 **注：** • JPEG 支持 YUV420、YUV422、YUV224、YUV444、YUV400，不支持 YUV411。 • 如果输出颜色格式为 RGB，COLOR_MODE 需配置为 4 (YUV400)。
16	R/W	0x0	INTERLEAVED • 0: CbCr 分量独立存放 • 1: CbCr 分量交错存放

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			图像格式信息 (Picture Format)
位域	类型	默认值	描述
15:0	R/W	0x0	STRIDE 图像的线宽

10.3.5.4.10. 0x604 PICTURE_SIZE

默认值：0x00000000			图像大小 (Picture Size)
位域	类型	默认值	描述
31:16	R/W	0x0	PIC_X_SIZE 图像宽
15:0	R/W	0x0	PIC_Y_SIZE 图像高 **注：** 当前解码图像为场格式，需要按照场高度 x2 配置。

10.3.5.4.11. 0x608 PICTURE_LUMA_ADDR

默认值：0x00000000			图像亮度数据地址 (Picture Luma Address)
位域	类型	默认值	描述
31:0	R/W	0x0	LUMA_ADDR 图像亮度分量 DRAM 地址

10.3.5.4.12. 0x60C PICTURE_CB_ADDR

默认值：0x00000000			图像 Cb 分量数据地址 (Picture Cb Address)
位域	类型	默认值	描述
31:0	R/W	0x0	CB_ADDR 图像 Cb 分量 DRAM 地址

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.3.5.4.13. 0x610 PICTURE_CR_ADDR

默认值：0x00000000			图像 Cr 分量数据地址 (Picture Cr Address)
位域	类型	默认值	描述
31:0	R/W	0x0	CR_ADDR 图像 Cr 分量 DRAM 地址

10.4. LCD

LCD 模块可将 DE 送过来的 DPI 接口转换为各种屏支持格式。

LCD 支持五种接口模式输出：

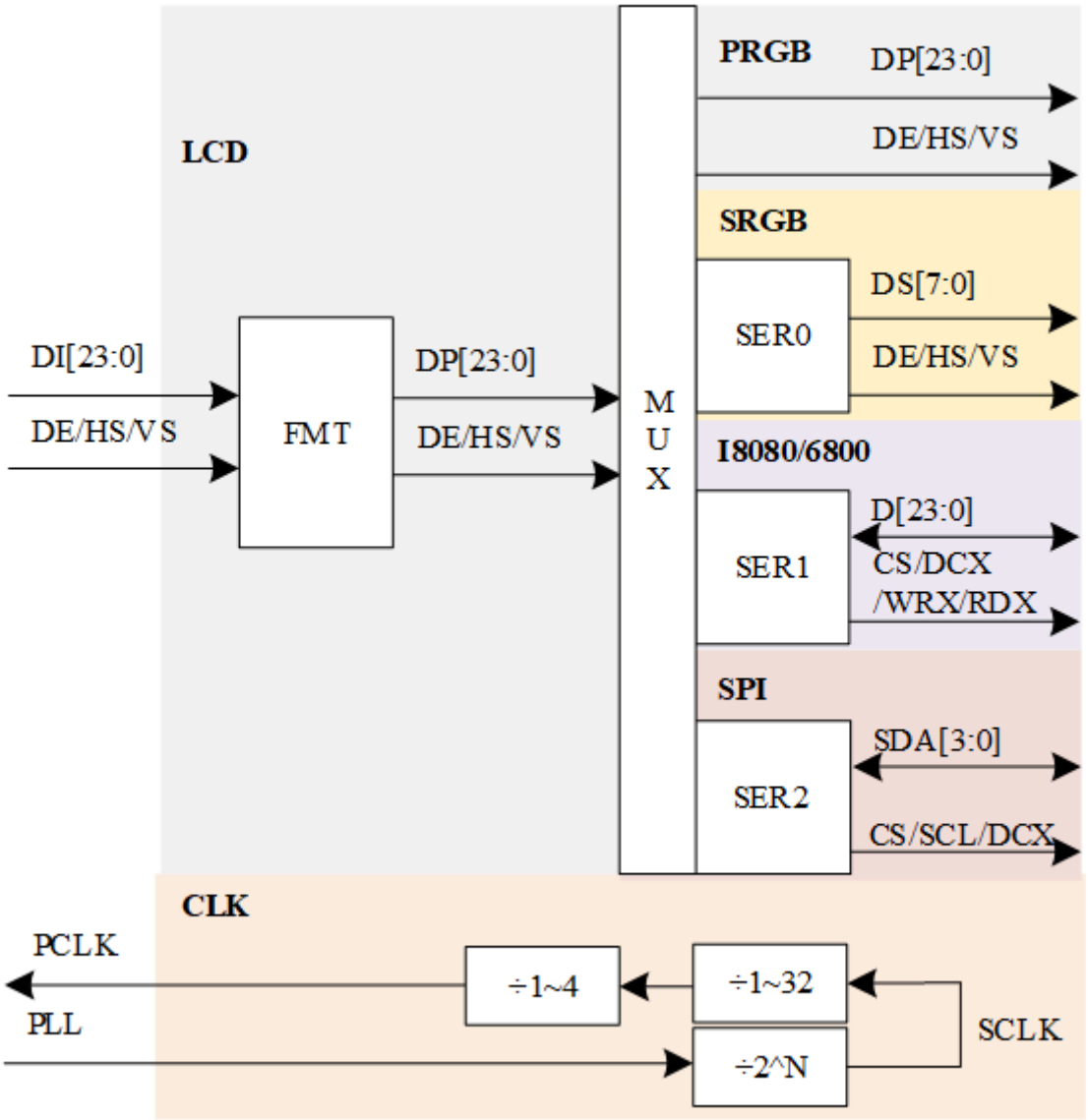
- PRGB 接口时序
- SRGB 接口时序
- I8080 接口时序
- SPI 接口时序
- 6800 接口时序

10.4.1. 特性说明

- 支持 PRGB/ SRGB/ I8080/ 6800/ SPI 接口
- 支持八位色深，R/G/B 三组信号可任意交叉，空白区域数据可配置
- 支持 R/G/B 组内数据输出顺序切换
- 支持 PRGB 24/ 18/ 16 bit 模式，最高 80 MHz PIXELCLK
- 支持 SRGB 8/ 6 bit 模式，最高 480x320@60fps
- 支持 I8080/ 6800 24/ 18/ 16/ 9/ 8 bit 模式，最高 960x540@60fps
- 支持 SPI 3/4–wire 以及 4–sda 模式，最高 480x320@60fps
- 支持 SPI 8/ 24/ 32 read 模式
- PRGB/ SRGB 时钟提供四个相位选择
- I8080/ 6800 输出支持三种采样方式
- SPI 支持四种采样方式
- I8080/ SPI 支持 64x8bit FIFO
- 支持像素时钟频率最高 80 MHz
- 分辨率及其它规格参考 DE 规格书

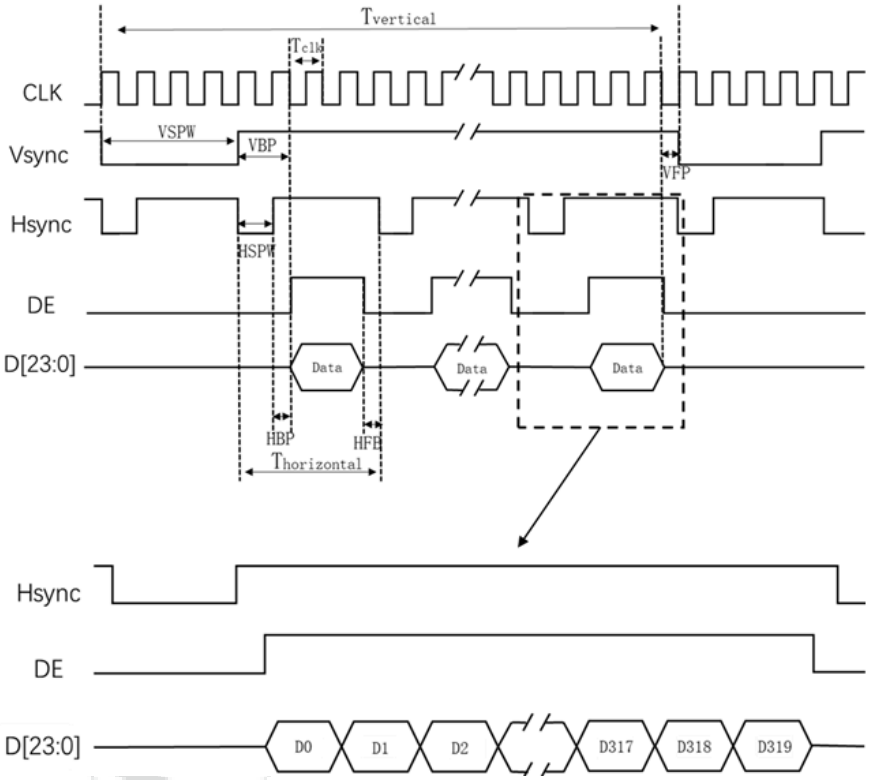
Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.4.2. 原理框图



Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.4.3. PRGB 接口时序



Vsync 为帧同步信号，表示帧开始。Hsync 为行同步信号，表示行开始。DE 为数据有效使能，表示数据有效，每 1 个 CLK 周期，采样一个像素数据 D[23:0]，以此完成一帧图像的输出显示。

表 10–8 时序描述

Defined	Function
CLK	像素时钟
Vsync	帧同步信号
Hsync	行同步信号
DE	数据有效使能
D[23:0]	像素数据信号
Tclk	时钟周期
Tvertical	列扫描周期
Thorizontal	行扫描周期
VBP	列扫描后肩

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 10–8 时序描述 (续)

Defined	Function
VFP	列扫描前肩
VSPW	列同步信号脉宽
HBP	行扫描后肩
HFP	行扫描前肩
HSPW	行同步信号脉宽

10.4.3.1. 数据块选择以及并转串功能

由 DE 模块输入的 DI[23:0] 信号进到 FMT 进行格式转换，输出并行 RGB 数据 DP[23:0] 信号，如下图所示。

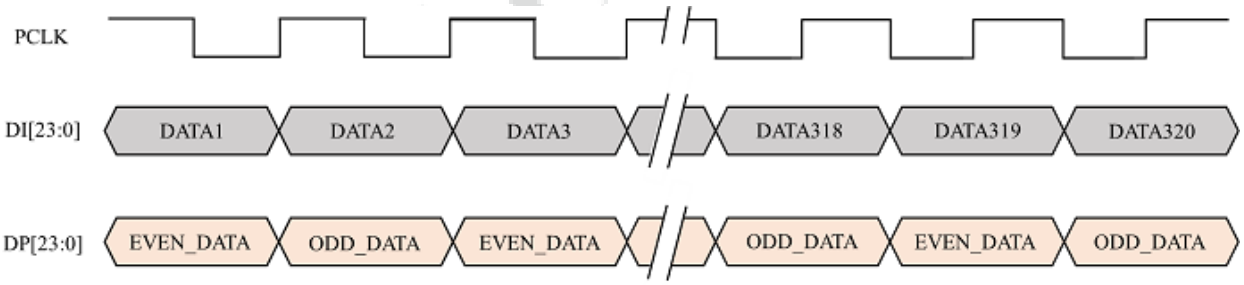


图 10–16 lcd_prgb

其中 DP[23:0] 每 8 位数据可任意互换输出 R/G/B 信号或者输出用户设置的 data 数据，用于功能调试以及方便 layout 走线， DP[23:0] 再进入到 SER0 转换成串行 RGB 数据 DS[7:0] 信号，如 lcd_srgb 所示。

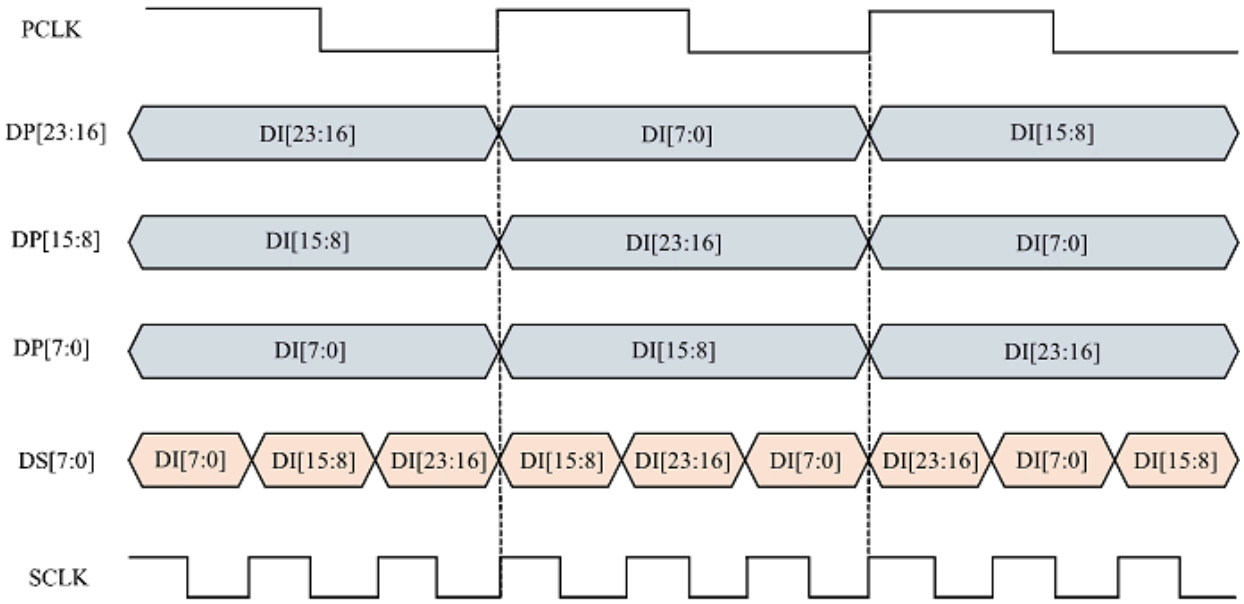


图 10–17 lcd_srgb

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

并行信号转串行信号控制方式：

1. DE 模块输入的完整图片 RGB 信号 DI[23:0] 分为以下三组，进入 FMT 转换 DP 输出，用户可自由配置。

   - DI[23:16]
   - DI[15:8]
   - DI[7:0]

   例如 DP[7:0]，可通过配置输出 DI[23:16]、 DI[15:8] 或者 DI[7:0] 的数据。

2. 之后三组 DP 信号进入 SER0 进行并行转串行。

   一个 PCLK 内：

   a. DP[7:0] 的数据在第一个 SCLK cycle 给到 DS[7:0]。

   b. DP[15:8] 的数据在第二个 SCLK cycle 给到 DS[7:0]

   c. DP[23:16] 的数据在第三个 SCLK cycle 给到 DS[7:0]，以此循环。

   注：

   无论三组 DP 信号里存着任何的 DI 数据，在并行信号转串行信号过程中均为 DP[7:0]/ DP[15:8]/ DP[23:16] 依次给到 DS[7:0]。

3. 配置下列相关寄存器：

   - LCD_CTL
   - DATA_SEL
   - ODD_DATA
   - EVEN_DATA

10.4.3.2. R/ G/ B 组内数据输出顺序切换功能

组内数据顺序切换：

以 8 bits 为单位进行切换，D23–D16/ D15–D8/ D7–D0 三组。

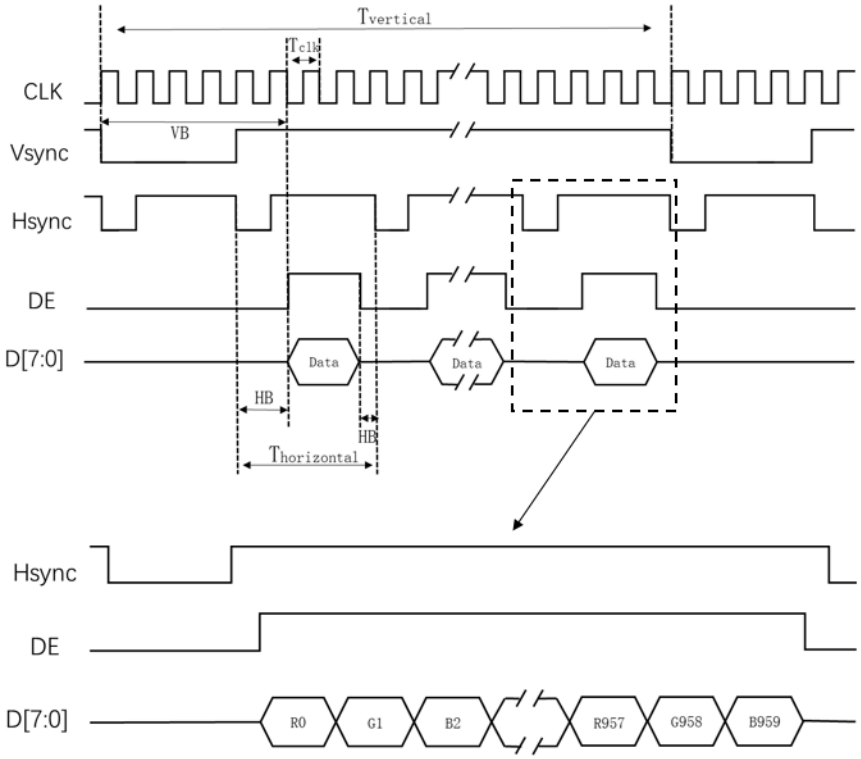
示例 1：D7–D0 组默认输出 D7–D6–D5–D4–D3–D2–D1–D0，可切换成顺序 D0–D1–D2–D3–D4–D5–D6–D7。

示例 2：D7–D0 组默认输出 D0，可切换成顺序 D0。 用于 PCB layout 走线可选。

关于详细说明，可查看相关配置寄存器DATA_QUT_SEL。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.4.4. SRGB 接口时序



Vsync 为帧同步信号，表示帧开始。Hsync 为行同步信号，表示行开始。DE 为数据有效使能，表示数据有效，每 1 个 CLK 周期，采样一个像素数据 D[7:0]，每 3 个 CLK 为一个像素数据，以此完成一帧图像的输出显示。

图 10–18 SRGB 接口时序

表 10–9 时序描述

Defined	Function
CLK	像素时钟
Vsync	帧同步信号
Hsync	行同步信号
DE	数据有效使能
D[7:0]	像素数据信号
Tclk	时钟周期
Tvertical	列扫描周期
Thorizontal	行扫描周期

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 10–9 时序描述 (续)

Defined	Function
VBP	列扫描后肩
VFP	列扫描前肩
VSPW	列同步信号脉宽
HBP	行扫描后肩
HFP	行扫描前肩
HSPW	行同步信号脉宽

10.4.5. I8080 接口时序

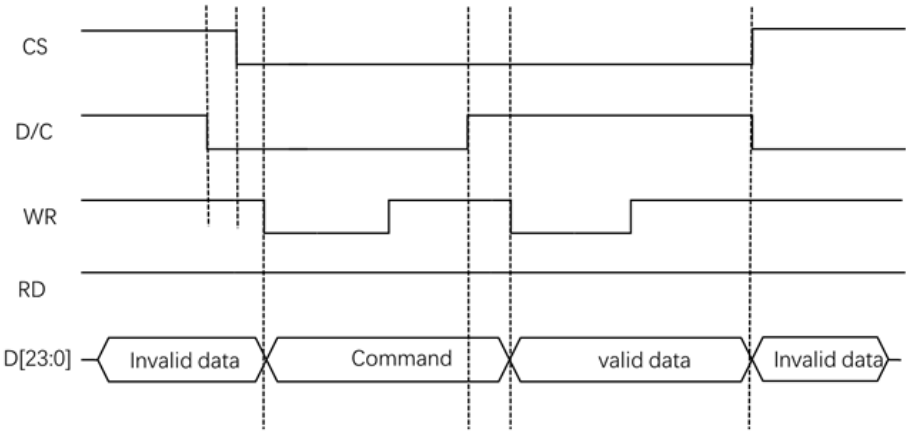


图 10–19 I8080 写时序

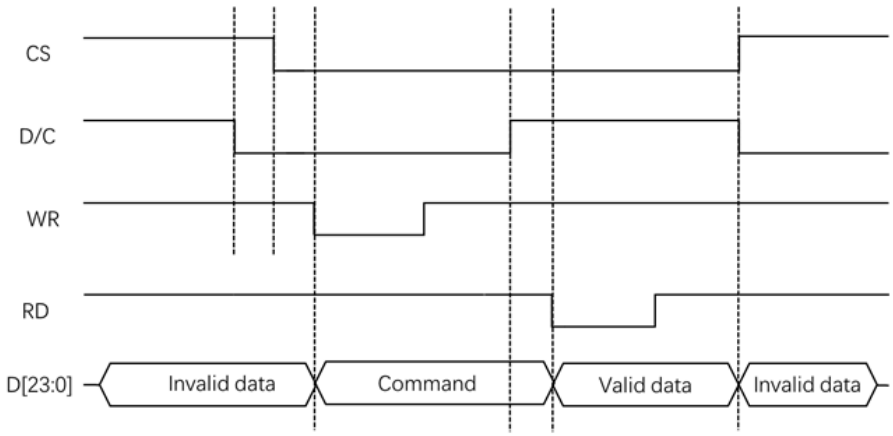


图 10–20 读时序

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

I8080 写时序中：

- CS 为片选信号，电平低有效，处于有效期间才可进行读写操作。
- D/C 为 Data 或者 Command 标志信号，D/C 为 0，则表示为 Command。D/C 为 1，则表示 parameter 或者 data 传输。D/C 在一些文档中又称为 RS、D/CX。
- WR 为写控制信号，RD 为读控制信号，WR 与 RD 均为低有效。进行写操作时，RD 需置 1。WR 由 1 变 0 则表示即将进行写操作。WR、RD 在一些文档中又称为 WRX、RDX。
- 进行读操作时，需先发送一个读命令，随后再进行读取数据。读取数据期间 WR 需置 1，RD 由 1 变 0 则表示即将进行读操作。

10.4.6. 6800 接口时序

10.4.6.1. Clocked E 模式

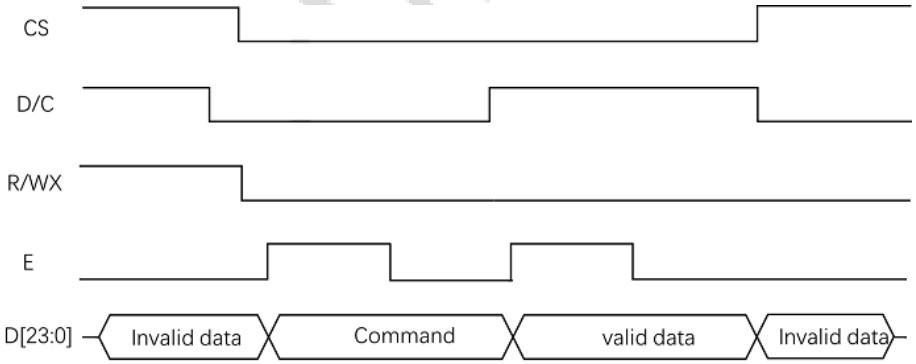


图 10–21 6800 Clocked E 模式写时序

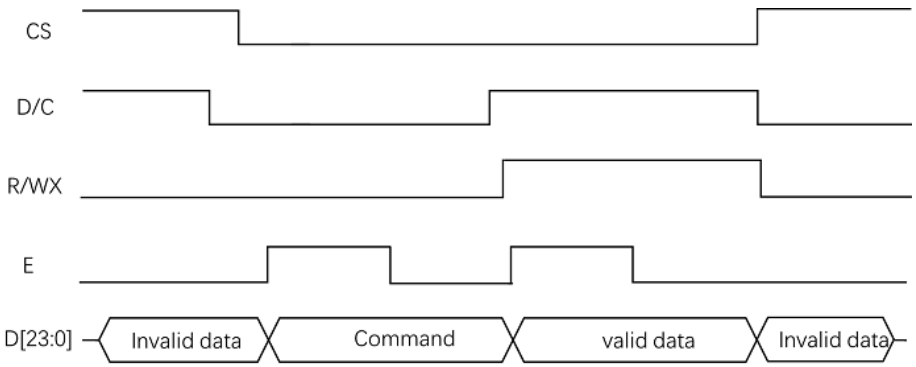


图 10–22 6800 Clocked E 模式读时序

Clocked E 模式中：

- CS 为片选信号，电平低有效，处于有效期间才可进行读写操作。
- D/C 为 Data 或者 Command 标志信号，D/C 为 0，则表示为 Command。D/C 为 1，则表示 parameter 或者 data 传输。D/C 在一些文档中又称为 RS、D/CX。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- R/WX 为读写控制信号，R/WX 为 0，进行写操作。R/WX 为 1，进行读操作。
- E 作为时钟信号，上升沿主控发出数据，下降沿屏端采样数据。

10.4.6.2. Fixed E 模式

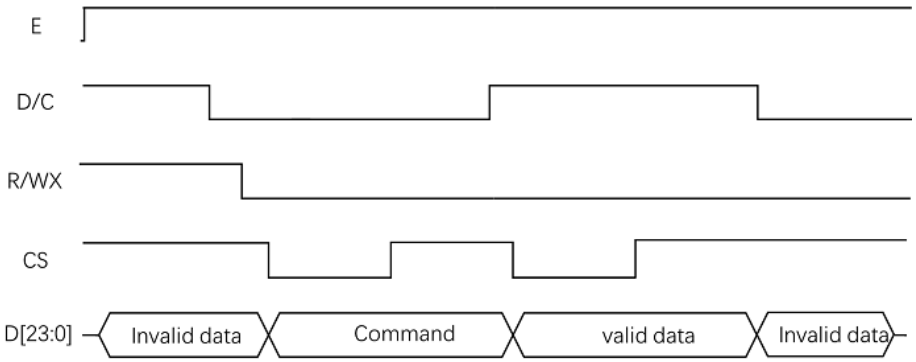


图 10–23 6800 Fixed E 模式写时序

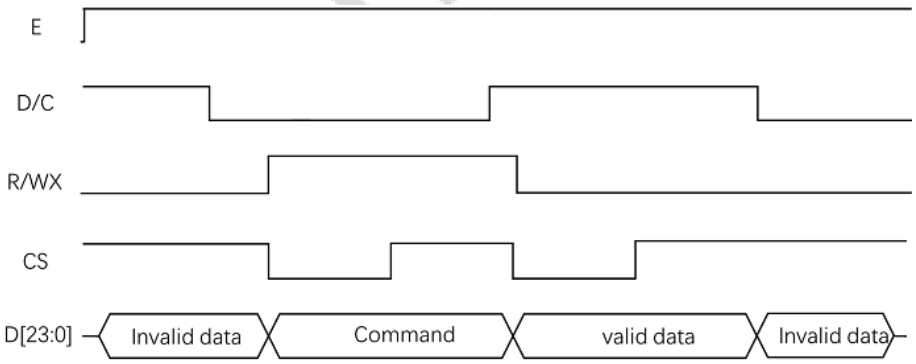


图 10–24 6800 Fixed E 模式读时序

Fixed E 模式中：

- CS 作为时钟信号，下降沿主控发出数据，上升沿屏端采样数据。
- D/C 为 Data 或者 Command 标志信号，D/C 为 0，则表示为 Command。D/C 为 1，则表示 parameter 或者 data 传输。D/C 在一些文档中又称为 RS、D/CX。
- R/WX 为读写控制信号，R/WX 为 0，进行写操作。R/WX 为 1，进行读操作。
- E 始终拉高为 1。

10.4.7. SPI 接口时序

SPI-8BIT 4 线模式时序图如下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

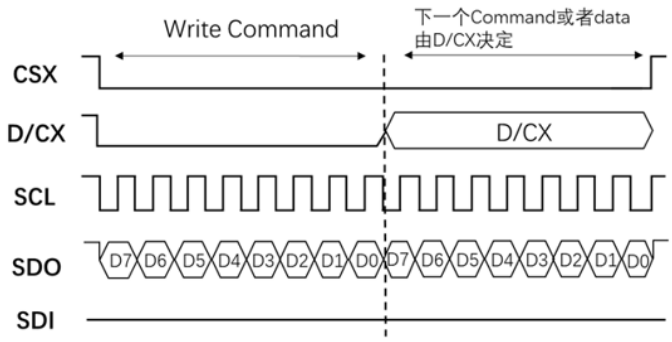


图 10–25 SPI 4 线写时序

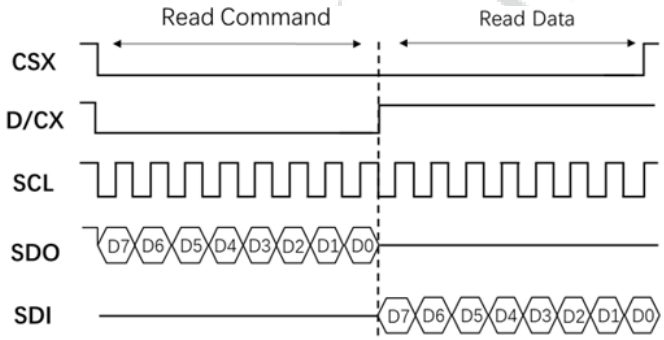


图 10–26 SPI 4 线 8bit 读时序

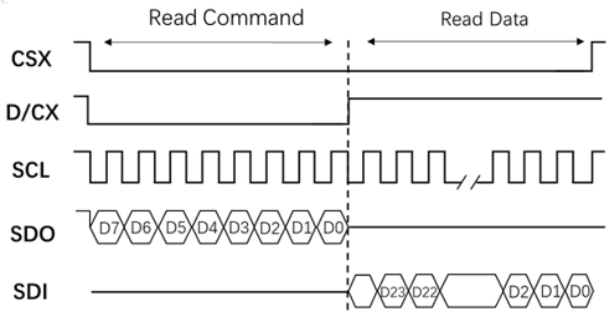


图 10–27 SPI 4 线 24bit 读时序

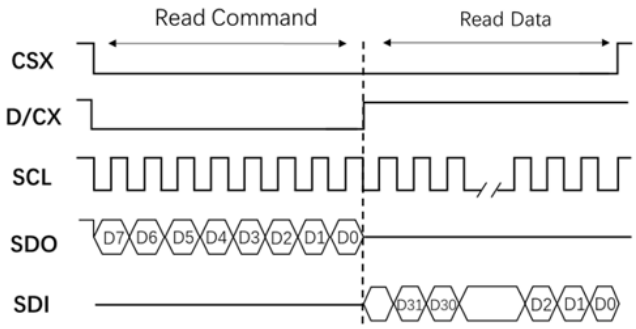


图 10–28 SPI 4 线 32bit 读时序

SPI-9BIT 3 线模式时序图如下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

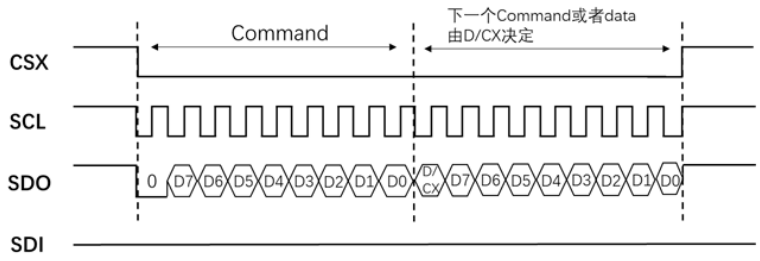


图 10–29 SPI 3 线写时序

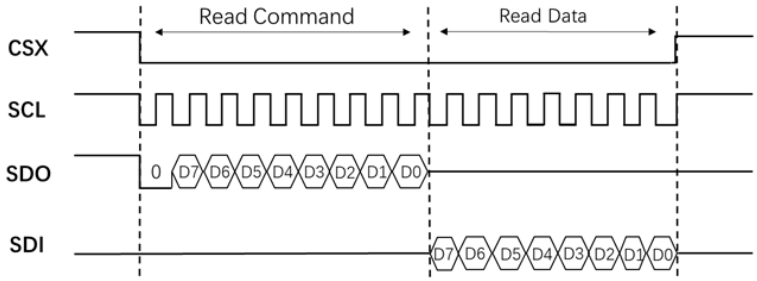


图 10–30 SPI 3 线 8bit 读时序

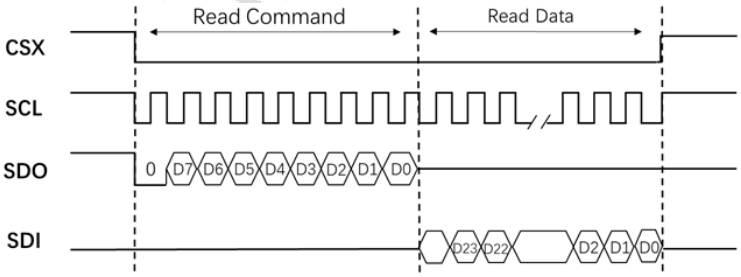


图 10–31 SPI 3 线 24bit 读时序

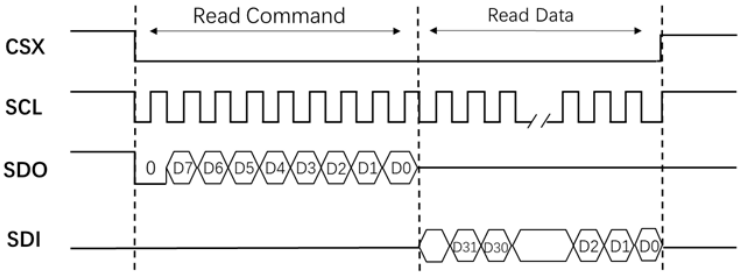


图 10–32 SPI 3 线 32bit 读时序

SPI 4-SDA 模式时序图如下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

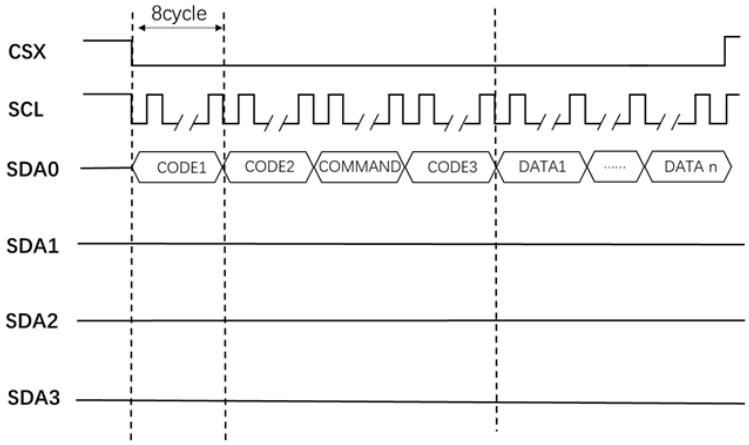


图 10–33 SPI 4–SDA 读写时序

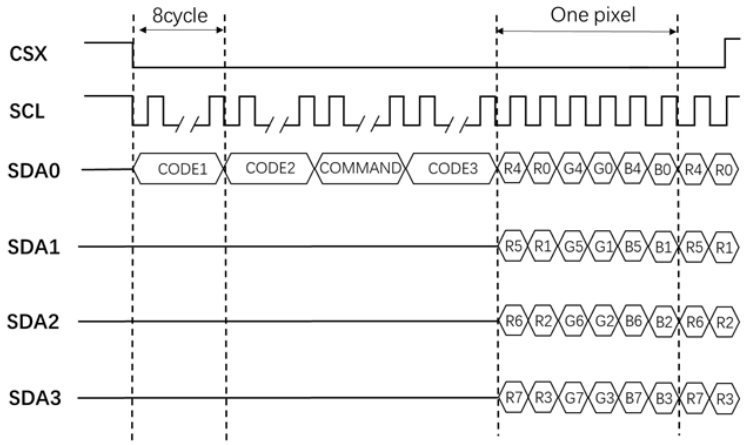


图 10–34 SPI 4–SDA 刷屏时序

图 10–34 : SPI 4–SDA 刷屏时序中：

- CODE1/ CODE2/ CODE3/ COMMAND 由 0x204 QSPI_CODE 寄存器配置。
- CODE1/ CODE2/ CODE3 需在开始刷屏前写入。
- 刷屏时 COMMAND 默认为 0x2c/ 0x3c，可通过寄存器修改。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

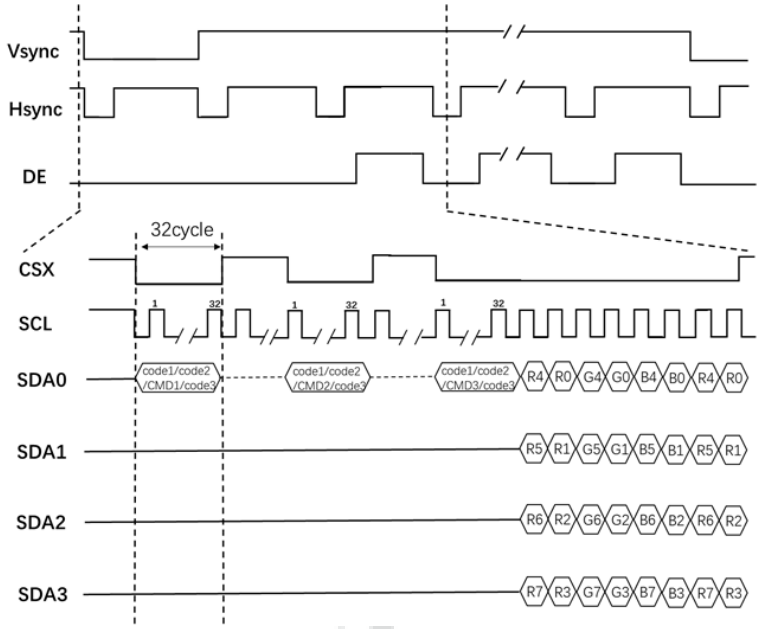


图 10–35 SPI 4–SDA 带时序信息的刷屏时序

图 10–35 : SPI 4–SDA 带时序信息的刷屏时序中：

- CMD1 即 spi–interface–timingCMD 控制寄存器中的 0x2c，表示帧开始。
- CMD1 即 spi–interface–timingCMD 控制寄存器中的 0x3c，表示行开始。
- 空白行 CS 只需持续 32 个 SCL 周期，输出空白行信息。

> **注：**
>
> 空白行以及有效行都需要输出引导码 CODE1/ CODE2/ COMMAND/ CODE3。

3 线 SPI、4 线 SPI 以及 4–SDA SPI 的数据传输格式支持 RGB888、RGB666 和 RGB565，如下所示：

> **注：**
>
> RGB888 转 RGB666 或 RGB565 格式，数据舍弃低位。下列示意图表示不同格式的信号出 pin 排序，比如 RGB666 中的 R5–R0 数据实际上对应 RGB888 中的 R7–R2 数据，依次类推。

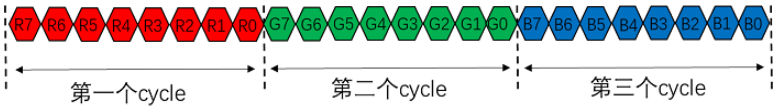


图 10–36 RGB888 格式

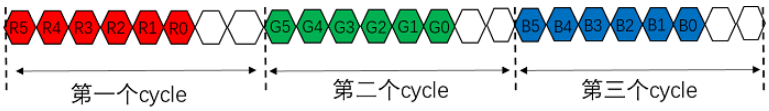


图 10–37 RGB666 格式

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

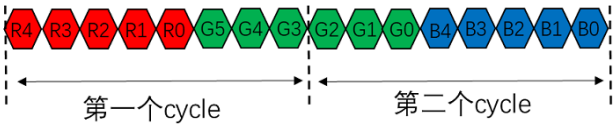


图 10–38 RGB565 格式

10.4.8. LCD 模式切换功能

PRGB 模式兼容五种输出模式，默认为 24 bits，可配置为 18 bits/ 16 bits。

SRGB 模式兼容两种输出模式，默认为 8 bits，可配置为 6 bits。

I8080 / 6800 模式兼容八种输出模式。

SPI 模式兼容三种输出模式。

相关配置寄存器为 LCD_CTL。

注：

PRGB 模式支持 RGB 组间信号以及组内信号倒序，其余模式不支持。

10.4.9. 像素时钟 (pixelclk) 与串行时钟 (sclk) 比例关系

表 10–10 pixelclk 与 sclk 的比例关系

模式	pixelclk : sclk	说明
PRGB	1:2	2 倍频作为时钟相位调节，pixel cycle 为 1:1
SRGB	1:6	2 倍频作为时钟相位调节，pixel cycle 为 1:3（24bit 并行转换成 8bit 串行）
I8080_24BIT	1:10	10 倍频作为时钟相位调节，pixel cycle 为 1:1 (6800 与 8080 一致)
I8080_18BIT	1:10	10 倍频作为时钟相位调节，pixel cycle 为 1:1(6800 与 8080 一致)
I8080_16BIT_1	1:15	10 倍频作为时钟相位调节，pixel cycle 为 2:3 (6800 与 8080 一致)
I8080_16BIT_2	1:20	10 倍频作为时钟相位调节，pixel cycle 为 1:2 (6800 与 8080 一致)
I8080_16BIT_3	1:10	10 倍频作为时钟相位调节，pixel cycle 为 1:1 (6800 与 8080 一致)
I8080_9BIT	1:20	10 倍频作为时钟相位调节，pixel cycle 为 1:2 (6800 与 8080 一致)
I8080_8BIT_1	1:30	10 倍频作为时钟相位调节，pixel cycle 为 1:3 (6800 与 8080 一致)

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 10–10 pixelclk 与 sclk 的比例关系 (续)

模式	pixelclk : sclk	说明
I8080_8BIT_2	1:20	10 倍频作为时钟相位调节，pixel cycle 为 1:2 (6800 与 8080 一致)
4 线 SPI rgb888	1:96	4 倍频作为时钟相位调节，pixel cycle 为 1:24（24bit 并行转换成 1bit 串行）
4 线 SPI rgb565	1:64	4 倍频作为时钟相位调节，pixel cycle 为 1:16（16bit 并行转换成 1bit 串行）
3 线 SPI rgb888	1:108	4 倍频作为时钟相位调节，pixel cycle 为 1:27（24bit 并行转换成 1bit 串行，D/C 传输需要消耗 3cycle）
3 线 SPI rgb565	1:72	4 倍频作为时钟相位调节，pixel cycle 为 1:18（16bit 并行转换成 1bit 串行，D/C 传输需要消耗 2cycle）
4SDA SPI rgb888	1:24	4 倍频作为时钟相位调节，pixel cycle 为 1:6（24bit 并行转换成 4bit 串行）
4SDA SPI rgb565	1:16	4 倍频作为时钟相位调节，pixel cycle 为 1:4（16bit 并行转换成 4bit 串行）

10.4.10. 寄存器列表

表 10–11 LCD 寄存器列表

偏移	缩写	寄存器名称	章节
0x000	LCD_CTL	LCD 控制	0x000 LCD_CTL
0x010	RGB_CLK_CTL	RGB 模式时钟控制	0x010 RGB_CLK_CTL
0x020	RGB_DATA_SEL	RGB 模式数据选择	0x020 RGB_DATA_SEL
0x024	RGB_ODD_DATA	RGB 模式奇行数据	0x024 RGB_ODD_DATA
0x028	RGB_EVEN_DATA	RGB 模式偶行数据	0x028 RGB_EVEN_DATA
0x030	RGB_DATA_OUT_SEL	RGB 模式数据输出选择	0x030 RGB_DATA_OUT_SEL
0x034	POL_CFG	RGB 模式信号极性配置	0x034 POL_CFG
0x100	I8080_COMMAND_CTL	I8080 模式命令控制	0x100 I8080_COMMAND_CTL
0x104	I8080_WR_CMD	I8080 模式写命令	0x104 I8080_WR_CMD
0x108	I8080_WR_DATA	I8080 模式写数据	0x108 I8080_WR_DATA
0x10C	I8080_WR_CTL	I8080 模式写控制	0x10C I8080_WR_CTL
0x110	I8080_RD_CTL	I8080 模式读控制	0x110 I8080_RD_CTL

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 10–11 LCD 寄存器列表 (续)

偏移	缩写	寄存器名称	章节
0x114	I8080_RD_DATA	I8080 模式读数据	0x114 I8080_RD_DATA
0x118	I8080_FIFO_DEPTH	I8080 模式 FIFO 深度	0x118 I8080_FIFO_DEPTH
0x11C	I8080_INT_ENABLE	I8080 模式中断使能	0x11C I8080_INT_ENABLE
0x120	I8080_INT_CLR	I8080 模式中断清除	0x120 I8080_INT_CLR
0x124	I8080_STATUS	I8080 模式状态	0x124 I8080_STATUS
0x128	I8080_SAMPLE_CTL	I8080 模式采样控制	0x128 I8080_SAMPLE_CTL
0x12C	I8080_CMD_DATA_OUT	I8080 模式 CMD 输出配置	0x12C I8080_CMD_DATA_OUT
0x200	SPI_SCL_CFG	SPI 模式串行时钟配置	0x200 SPI_SCL_CFG
0x204	QSPI_CODE	QSPI 模式标志码	0x204 QSPI_CODE
0x208	SPI_COMMAND_CTL	SPI 模式命令控制	0x208 SPI_COMMAND_CTL
0x20C	SPI_WR_CMD	SPI 模式写命令	0x20C SPI_WR_CMD
0x210	SPI_WR_DATA	SPI 模式写数据	0x210 SPI_WR_DATA
0x214	SPI 模式写控制	IDMAC 中断使能	0x214 SPI_WR_CTL
0x218	SPI_RD_CTL	SPI 模式读控制	0x218 SPI_RD_CTL
0x21C	SPI_RD_DATA	SPI 模式读数据	0x21C SPI_RD_DATA
0x220	SPI_FIFO_DEPTH	SPI 模式 FIFO 深度	0x220 SPI_FIFO_DEPTH
0x224	SPI_INT_ENABLE	SPI 模式中断使能	0x224 SPI_INT_ENABLE
0x228	SPI_INT_CLR	SPI 模式中断清除	0x228 SPI_INT_CLR
0x22C	SPI_STATUS	SPI 模式状态	0x22C SPI_STATUS
0x230	SPI_RD_MODE	SPI 模式读模式	0x230 SPI_RD_MODE
0x234	QSPI_MODE	QSPI 模式	0x234 QSPI_MODE
0xFF0	DEBUG_SEL	调试信号选择	0xFF0 DEBUG_SEL
0xFF4	DEBUG	调试	0xFF4 DEBUG
0xFFC	VERSION	版本号	0xFFC VERSION

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.4.11. 寄存器描述

10.4.11.1. 0x000 LCD_CTL

默认值：0x00000000			LCD 控制 (LCD Control)
位域	类型	默认值	描述
31:30	–	–	–
29:28	R/W	0x0	SPI_READ_FORMAT SPI 读数据格式 • 0: 8bit–read • 1: 24bit–read • 2: 32bit–read
27:24	R/W	0x0	SPI_FORMAT SPI 数据格式 • 0: RGB565 • 1: RGB666 • 2: RGB888
23:22	–	–	–
21:20	R/W	0x0	SPI_MODE SPI 模式选择 • 0: 3 线模式 • 1: 4 线模式 • 2: 4–sda 模式
19:16	R/W	0x0	I8080_MODE I8080 模式选择 • 0: 8bit–RGB565 • 1: 8bit–RGB666 • 2: 9bit–RGB666 • 3: 16bit–RGB666_mode1 • 4: 16bit–RGB666_mode2 • 5: 16bit–RGB565 • 6: 18bit–RGB666 • 7: 24bit–RGB888
15:13	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			LCD 控制 (LCD Control)
位域	类型	默认值	描述
12	R/W	0x0	SRGB_MODE SRGB 模式选择 • 0: 8–bit 模式 • 1: 6–bit 模式
11	–	–	–
10:8	R/W	0x0	PRGB_MODE PRGB 模式选择 • 0: 24bit 模式 • 1: 18bit 模式 1 • 2: 18bit 模式 2 • 3: 16bit 模式 1 • 4: 16bit 模式 2
7:6	R/W	0x0	I8080/ 6800_CTL • 0: I8080 模式 • 1: I6800 Clocked E 模式（E 当时钟） • 2: I6800 Fixed E 模式（CS 当时钟） 仅在 bit[5:4] 配置为 I8080 模式时有效
5:4	R/W	0x0	LCD_MODE LCD 接口模式选择 • 0: PRGB • 1: SRGB • 2: I8080 • 3: SPI
3:1	–	–	–
0	R/W	0x0	LCD_EN 模块使能控制 • 0: 模块关闭 • 1: 模块打开 先配置完其他参数，此位最后打开。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.4.11.2. 0x010 RGB_CLK_CTL

默认值：0x00000000			RGB 模式时钟控制 (RGB Clock Control)
位域	类型	默认值	描述
31:2	–	–	–
1:0	R/W	0x0	CKO_PHASE_SEL pixel 时钟输出相位选择 • 0: 时钟上升沿延后数据 0° 相位 • 1: 时钟上升沿延后数据 90° 相位 • 2: 时钟上升沿延后数据 180° 相位 • 3: 时钟上升沿延后数据 270° 相位

10.4.11.3. 0x020 RGB_DATA_SEL

默认值：0x02100210			RGB 模式数据选择 (RGB Data Select)
位域	类型	默认值	描述
31:26	–	–	–
25:24	R/W	0x2	EVEN _DP2316_SEL 偶数行 DP[23:16] 输出选择 • 0: DI[7:0] Display Engine 模块输入的 DI[7:0] 将被选择为偶数行 DP[23:16] 的输出。 • 1: DI[15:8] Display Engine 模块输入的 DI[15:8] 将被选择为偶数行 DP[23:16] 的输出。 • 2: DI[23:16]，默认值 Display Engine 模块输入的 DI[23:16] 将被选择为偶数行 DP[23:16] 的输出。 • 3: EVEN_DATA[23:16] 偶数行数据寄存器 [23:16] 的内容将被选择为偶数行 DP[23:16] 的输出。 用户可强制输出数据进行调试。此位默认为 2，保持输入输出位一致。
23:22	–	–	–
21:20	R/W	0x1	EVEN _DP1508_SEL 偶数行 DP[15:8] 输出选择 • 0: DI[7:0] Display Engine 模块输入的 DI[7:0] 将被选择为偶数行 DP[15:8] 的输出。 • 1: DI[15:8]，默认值 Display Engine 模块输入的 DI[15:8] 将被选择为偶数行 DP[15:8] 的输出。 • 2: DI[23:16]

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x02100210			RGB 模式数据选择 (RGB Data Select)
位域	类型	默认值	描述
			Display Engine 模块输入的 DI[23:16] 将被选择为偶数行 DP[15:8] 的输出。 • 3: EVEN_DATA [15:8] 偶数行数据寄存器 [15:8] 的内容将被选择为偶数行 DP[15:8] 的输出。 用户可强制输出数据进行调试。此位默认为 1，保持输入输出位一致。
19:18	–	–	–
17:16	R/W	0x0	EVEN _DP0700_SEL 偶数行 DP[7:0] 输出选择 • 0: DI[7:0]，默认值 Display Engine 模块输入的 DI[7:0] 将被选择为偶数行 DP[7:0] 的输出。 • 1: DI[15:8] Display Engine 模块输入的 DI[15:8] 将被选择为偶数行 DP[7:0] 的输出。 • 2: DI[23:16] Display Engine 模块输入的 DI[23:16] 将被选择为偶数行 DP[7:0] 的输出。 • 3: EVEN_DATA [7:0] 偶数行数据寄存器 [7:0] 的内容将被选择为偶数行 DP[7:0] 的输出。 用户可强制输出数据进行调试。此位默认为 0，保持输入输出位一致。
15:10	–	–	–
9:8	R/W	0x2	ODD_DP2316_SEL 奇数行 DP[23:16] 输出选择 • 0: DI[7:0] Display Engine 模块输入的 DI[7:0] 将被选择为奇数行 DP[23:16] 的输出。 • 1: DI[15:8] Display Engine 模块输入的 DI[15:8] 将被选择为奇数行 [23:16] 的输出。 • 2: DI[23:16]，默认值 Display Engine 模块输入的 DI[23:16] 将被选择为奇数行 [23:16] 的输出。 • 3: ODD_DATA[23:16] 奇数行数据寄存器 [23:16] 的内容将被选择为奇数行 [23:16] 的输出。 用户可强制输出数据进行调试。此位默认为 2，保持输入输出位一致。
7:6	–	–	–
5:4	R/W	0x1	ODD_DP1508_SEL 奇数行 DP[15:8] 输出选择 • 0: DI[7:0] Display Engine 模块输入的 DI[7:0] 将被选择为奇数行 DP[15:8] 的输出。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x02100210			RGB 模式数据选择 (RGB Data Select)
位域	类型	默认值	描述
			• 1: DI[15:8]，默认值 Display Engine 模块输入的 DI[15:8] 将被选择为奇数行 [15:8] 的输出。 • 2: DI[23:16] Display Engine 模块输入的 DI[23:16] 将被选择为奇数行 [15:8] 的输出。 • 3: ODD_DATA [15:8] 奇数行数据寄存器 [15:8] 的内容将被选择为奇数行 [15:8] 的输出。 用户可强制输出数据进行调试。此位默认为 1，保持输入输出位一致。
3:2	–	–	–
1:0	R/W	0x0	ODD_DP0700_SEL 奇数行 DP[7:0] 输出选择 • 0: DI[7:0]，默认值 Display Engine 模块输入的 DI[7:0] 将被选择为奇数行 DP[7:0]的输出。 • 1: DI[15:8] Display Engine 模块输入的 DI[15:8] 将被选择为奇数行[7:0]的输出 • 2: DI[23:16] Display Engine 模块输入的 DI[23:16] 将被选择为奇数行[7:0]的输出 • 3: ODD_DATA [7:0] 奇数行数据寄存器[7:0]的内容将被选择为奇数行[7:0]的输出 用户可强制输出数据进行调试。此位默认为 0，保持输入输出位一致。

10.4.11.4. 0x024 RGB_ODD_DATA

默认值：0x00000000			RGB 模式奇行数据 (RGB Odd Data Select)
位域	类型	默认值	描述
31:24	–	–	–
23:16	R/W	0x0	ODD_DATA[23:16] 若配置 OOD_DP2316_SEL 为 0x3，则使奇数行 DP[23:16] 强制输出。 用户配置此位的数据。
15:8	R/W	0x0	ODD_DATA[15:8] 若配置 OOD_DP1508_SEL 为 0x3，则使奇数行 DP[15:8] 强制输出。 用户配置此位的数据。
7:0	R/W	0x0	ODD_DATA[7:0] 若配置 OOD_DP0700_SEL 为 0x3，则使奇数行 DP[7:0] 强制输出。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			RGB 模式奇行数据 (RGB Odd Data Select)
位域	类型	默认值	描述
			用户配置此位的数据。

10.4.11.5. 0x028 RGB_EVEN_DATA

默认值：0x00000000			RGB 模式偶行数据 (RGB Even Data Select)
位域	类型	默认值	描述
31:24	–	–	–
23:16	R/W	0x0	EVEN_DATA[23:16] 若配置 EVEN _DP2316_SEL 为 0x3，则使偶数行 DP[23:16] 强制输出。 用户配置此位的数据。
15:8	R/W	0x0	EVEN _DATA[15:8] 若配置 EVEN _DP1508_SEL 为 0x3，则使偶数行 DP[15:8] 强制输出。 用户配置此位的数据。
7:0	R/W	0x0	EVEN _DATA[7:0] 若配置 EVEN _DP0700_SEL 为 0x3，则使偶数行 DP[7:0] 强制输出。 用户配置此位的数据。

10.4.11.6. 0x030 RGB_DATA_OUT_SEL

默认值：0x00007770			RGB 模式数据输出选择 (RGB Data Output Select)
位域	类型	默认值	描述
31:15	–	–	–
14:12	R/W	0x7	DATA[23:16]_VALID • 0：1 位有效 • 1：2 位有效 • … • 7：8 位有效
11	–	–	–
10:8	R/W	0x7	DATA[15:8]_VALID • 0：1 位有效 • 1：2 位有效 • … • 7：8 位有效

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00007770			RGB 模式数据输出选择 (RGB Data Output Select)
位域	类型	默认值	描述
7	–	–	–
6:4	R/W	0x7	DATA[7:0]_VALID DATA[7:0] 倒序有效位配置 • 0：1 位有效 • 1：2 位有效 • … • 7：8 位有效
3	–	–	–
2	R/W	0x0	DATA[23:16] • 0: 输出低位到高位 D16—D23 • 1: 输出低位到高位 D23—D16
1	R/W	0x0	DATA[15:8] • 0: 输出低位到高位 D8—D15 • 1: 输出低位到高位 D15—D8
0	R/W	0x0	DATA[7:0] • 0: 输出低位到高位 D0—D7 • 1: 输出低位到高位 D7—D0 若 DATA[7:0]_VALID 配置为 5，即 6 位倒序有效，此时本 bit 配置为 1，则 DATA[7:0] 更改为 DATA[2:7, 1, 0]。

10.4.11.7. 0x034 POL_CFG

默认值：0x00000000			RGB 模式信号极性配置 (Polarity Configuration)
位域	类型	默认值	描述
31:2	–	–	–
1	R /W	0x0	V_SYNC_POL 垂直同步脉冲极性 • 0: 负极 • 1: 正极
0	R /W	0x0	H_SYNC_POL 水平同步脉冲极性

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			RGB 模式信号极性配置 (Polarity Configuration)
位域	类型	默认值	描述
			• 0: 负极 • 1: 正极

10.4.11.8. 0x100 I8080_COMMAND_CTL

默认值：0x3c002c00			I8080 模式命令控制 (I8080 Command Control)
位域	类型	默认值	描述
31:24	R/W	0x3c	OTHER_LINE_COMMAND，其他行命令
23:17	–	–	–
16	R/W	0x0	OTHER_LINE_COMMAND_CTL • 0: 其余行刷屏数据自动插入 0x3c 命令 • 1: 替换插入命令，需配置 bit[31:24] 0x3c: write memory continue
15:8	R/W	0x2c	FIRST_LINE_COMMAND 第一行命令
7:1	–	–	–
0	R/W	0x0	FIRST_LINE_COMMAND_CTL • 0: 第一行刷屏数据自动插入 0x2c 命令 • 1: 替换插入命令，需配置 bit[15:8] 0x2c: write memory start

10.4.11.9. 0x104 I8080_WR_CMD

默认值：0x00000000			I8080 模式写命令 (I8080 Write Command)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	WR_CMD 写命令 此字段配置命令，硬件从 WR_CTL 以及 RD_CTL 判断命令是读或者写。

10.4.11.10. 0x108 I8080_WR_DATA

默认值：0x00000000			I8080 模式写数据 (I8080 Write Data)
位域	类型	默认值	描述
31:8	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			I8080 模式写数据 (I8080 Write Data)
位域	类型	默认值	描述
7:0	R/W	0x0	WR_DATA 写数据 每次写操作需准备 WR_CMD 以及 WR_DATA，然后配置 WR_CTL 启动写操作。若只有命令无数据，只需要准备 WR_CMD，然后配置 CTL 启动写操作。

10.4.11.11. 0x10C I8080_WR_CTL

默认值：0x00000000			I8080 模式写控制 (I8080 Write Control)
位域	类型	默认值	描述
31:15	–	–	–
14:8	R/W	0x0	WR_COUNT data 数量 单位：byte
7:1	–	–	–
0	R/W1C	0x0	WR_START 写操作启动 写 1 启动写操作，写完自动清零。

10.4.11.12. 0x110 I8080_RD_CTL

默认值：0x00000000			I8080 模式读控制 (I8080 Read Control)
位域	类型	默认值	描述
31:15	–	–	–
14:8	R/W	0x0	RD_COUNT 读命令读取数据个数，硬件以此字段决定何时结束读操作。 单位：byte
7:1	–	–	–
0	R/W1C	0x0	RD_START 读操作启动 写 1 启动读操作，读完自动清零。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.4.11.13. 0x114 I8080_RD_DATA

默认值：0x00000000			I8080 模式读数据 (I8080 Read Data)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R	0x0	RD_DATA，读数据

10.4.11.14. 0x118 I8080_FIFO_DEPTH

默认值：0x00000000			I8080 模式 FIFO 深度 (I8080 FIFO Depth)
位域	类型	默认值	描述
31:22	–	–	–
21	R/W1C	0x0	RD_FIFO_FLUSH 写 1 则 RD_FIFO 复位，复位后自动清零。
20	R/W1C	0x0	WR_FIFO_FLUSH 写 1 则 WR_FIFO 复位，复位后自动清零。
19:17	–	–	–
16	R/W	0x0	RD_FIFO_CTL 若接收数据达到 FIFO 阈值，数据仍在接收的 FIFO 行为 • 0: 覆盖 FIFO 中的数据 • 1: 丢弃正在接收的数据 LCD_EN 之前配置。
15	–	–	–
14:8	R	0x0	RD_FIFO_DEPTH 读 FIFO 深度 当前读 FIFO 数据数量 用户在发送一个读命令时，读回来的数据存储在 FIFO 中，需要本寄存器判断 FIFO 的数据量是否符合预期，并且从 FIFO 取走数据时，此字段也更新剩余深度。
7	–	–	–
6:0	R	0x0	WR_FIFO_DEPTH 写 FIFO 深度 当前写 FIFO 数据数量 用户在写操作启动之前，需判断 FIFO 内的数据量是否等于用户即将发送的数量： • 若深度不足，用户可判断是否软件行为的问题。 • 若深度达到预期，则软件可启动读操作。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.4.11.15. 0x11C I8080_INT_ENABLE

默认值：0x00000000			I8080 模式中断使能 (I8080 Interrupt Enable)
位域	类型	默认值	描述
31:4	–	–	–
3	R/W	0x0	TX_UNDER 发送下溢出中断 • 0：禁止 • 1：使能 写操作时，需要发送 N byte 数据，但 FIFO 内的数据少于 N，则 FIFO 在数据发完后发生下溢出中断。
2	R/W	0x0	TX_OVER 发送上溢出中断 • 0：禁止 • 1：使能 写操作时，FIFO 深度 为 64 bytes，但写入 FIFO 数据大于 64 bytes，则 FIFO 发生上溢出中断。
1	R/W	0x0	RX_UNDER 接收下溢出中断 • 0：禁止 • 1：使能 读操作时，FIFO 内数据为空，但此时用户读取 FIFO 的数据，则发生下溢出中断。
0	R/W	0x0	RX_OVER 接收上溢出中断 • 0：禁止 • 1：使能 读操作时，FIFO 的深度为 64 bytes，但读回的数据大于 64 bytes，则发生上溢出中断。

10.4.11.16. 0x120 I8080_INT_CLR

默认值：0x00000000			I8080 模式中断清除 (I8080 Interrupt Clear)
位域	类型	默认值	描述
31:4	–	–	–
3	R/W1C	0x0	TX_UNDER_CLR 发送下溢出中断

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			I8080 模式中断清除 (I8080 Interrupt Clear)
位域	类型	默认值	描述
			• 0: 未触发 • 1: 触发
2	R/W1C	0x0	TX_OVER_CLR 发送上溢出中断 • 0: 未触发 • 1: 触发
1	R/W1C	0x0	RX_UNDER_CLR 接收下溢出中断 • 0: 未触发 • 1: 触发
0	R/W1C	0x0	RX_OVER_CLR 接收上溢出中断 • 0: 未触发 • 1: 触发

10.4.11.17. 0x124 I8080_STATUS

默认值：0x00000003			I8080 模式状态 (I8080 Status)
位域	类型	默认值	描述
31:5	–	–	–
4	R	0x0	I8080_IDEL I8080 空闲状态 • 0: I8080 空闲状态 • 1: I8080 非空闲状态
3	R	0x0	RX_FIFO_FULL 接收 FIFO 为满状态 • 0: 接收 FIFO 非满 • 1: 接收 FIFO 为满
2	R	0x0	RX_FIFO_NOT_EMPTY 接收 FIFO 非空状态

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000003			I8080 模式状态 (I8080 Status)
位域	类型	默认值	描述
			• 0: 接收 FIFO 为空 • 1: 接收 FIFO 非空
1	R	0x1	TX_FIFO_EMPTY 发送 FIFO 为空状态 • 0: 发送 FIFO 非空 • 1: 发送 FIFO 为空
0	R	0x1	TX_FIFO_NOT_FULL 发送 FIFO 非满状态 • 0: 发送 FIFO 为满 • 1: 发送 FIFO 非满

10.4.11.18. 0x128 I8080_SAMPLE_CTL

默认值：0x00000010			I8080 模式采样控制 (I8080 Sample Control)
位域	类型	默认值	描述
31:6	–	–	–
5:4	R/W	0x1	WR_SAMPLE_CTL WR 采样选择
3:2	–	–	–
1:0	R/W	0x0	INPUT_SAMPLE_CTL 输入采样位置选择 • 0: 正常采样 • 1: 提前采样 • 2: 延迟采样

10.4.11.19. 0x12C I8080_CMD_DATA_OUT

默认值：0x00000000			I8080 模式 CMD 输出配置 (I8080 Command Data Output)
位域	类型	默认值	描述
31:1	–	–	–
0	R/W	0x0	DATA_OUTCMD_DATA 输出控制

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			I8080 模式 CMD 输出配置 (I8080 Command Data Output)
位域	类型	默认值	描述
			• 0: 8080_WR_DATA 寄存器中的 data 每个 WR 周期从低 8 位输出 • 1: 由 I8080_MODE(LCD_CTL[19:16]) 决定： ◦ 若 I8080_MODE 为 8/9bit，data 从低 8 位输出。 ◦ 若 I8080_MODE 为 16/18bit，data 从低 16 位输出。 ◦ 若 I8080_MODE 为 24bit，data 24 位输出。

10.4.11.20. 0x200 SPI_SCL_CFG

默认值：0x00000003			SPI 模式串行时钟配置 (SPI Serial Clock Configuration)
位域	类型	默认值	描述
31:5	–	–	–
4	R/W	0x0	SCL_CTLSCL 刷屏行为控制 • 0: SCL 在行与行之间暂停 • 1: SCL 在行与行之间连续
3:2	–	–	–
1	R/W	0x1	SCL_PHASE_CFGSCL 相位配置 • 0: 第一个跳变沿采样 • 1: 第二个跳变沿采样
0	R/W	0x1	SCL_POLSCL 极性配置 • 0: 电平 0 为空闲状态 • 1: 电平 1 为空闲状态

10.4.11.21. 0x204 QSPI_CODE

默认值：0x00000000			QSPI 模式标志码 (QSPI Code)
位域	类型	默认值	描述
31:17	–	–	–
23:16	R/W	0x0	CODE1，4SDA 标志码
15:8	R/W	0x0	CODE2，4SDA 标志码

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			QSPI 模式标志码 (QSPI Code)
位域	类型	默认值	描述
7:0	R/W	0x0	CODE3，4SDA 标志码

10.4.11.22. 0x208 SPI_COMMAND_CTL

默认值：0x3c002c00			SPI 模式命令控制 (SPI Command Control)
位域	类型	默认值	描述
31:24	R/W	0x3c	OTHER_LINE_COMMAND，其他行命令
23:17	–	–	–
16	R/W	0x0	OTHER_LINE_COMMAND_CTL • 0: 其余行刷屏数据自动插入 0x3c 命令 • 1: 替换插入命令，需配置 bit[31:24] 0x3c: write memory continue
15:8	R/W	0x2c	FIRST_LINE_COMMAND，第一行命令
7:1	–	–	–
0	R/W	0x0	FIRST_LINE_COMMAND_CTL • 0: 第一行刷屏数据自动插入 0x2c 命令 • 1: 替换插入命令，需配置 bit[15:8] 0x2c: write memory start

10.4.11.23. 0x20C SPI_WR_CMD

默认值：0x00000000			SPI 模式写命令 (SPI Write Command)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	WR_CMD 写命令 此字段配置命令，硬件从 WR_CTL 以及 RD_CTL 判断命令是读或者写。

10.4.11.24. 0x210 SPI_WR_DATA

默认值：0x00000000			SPI 模式写数据 (SPI Write Data)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	WR_DATA 写数据

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			SPI 模式写数据 (SPI Write Data)
位域	类型	默认值	描述
			每次写操作需准备 WR_CMD 以及 WR_DATA，然后配置 WR_CTL 启动写操作。若只有命令无数据，只需要准备 WR_CMD，然后配置 CTL 启动写操作。

10.4.11.25. 0x214 SPI_WR_CTL

默认值：0x00000000			SPI 模式写控制 (SPI Write Control)
位域	类型	默认值	描述
31:15	–	–	–
14:8	R/W	0x0	WR_COUNT data 数量 单位：byte
7:1	–	–	–
0	R/W1C	0x0	WR_START 写操作启动写 1 启动写操作，写完自动清零。

10.4.11.26. 0x218 SPI_RD_CTL

默认值：0x00000000			SPI 模式读控制 (SPI Read Control)
位域	类型	默认值	描述
31:15	–	–	–
14:8	R/W	0x0	RD_COUNT 读命令读取数据个数，硬件以此字段决定何时结束读操作 单位：byte
7:1	–	–	–
0	R/W1C	0x0	RD_START 读操作启动 写 1 启动读操作，读完自动清零。

10.4.11.27. 0x21C SPI_RD_DATA

默认值：0x00000000			SPI 模式读数据 (SPI Read Data)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R	0x0	RD_DATA 读数据

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.4.11.28. 0x220 SPI_FIFO_DEPTH

默认值：0x00000000			SPI 模式 FIFO 深度 (SPI FIFO Depth)
位域	类型	默认值	描述
31:22	–	–	–
21	R/W1C	0x0	RD_FIFO_FLUSH 写 1 则 RD_FIFO 复位，复位后自动清零。
20	R/W1C	0x0	WR_FIFO_FLUSH 写 1 则 WR_FIFO 复位，复位后自动清零。
19:17	–	–	–
16	R/W	0x0	RD_FIFO_CTL 若接收数据达到 FIFO 阈值，数据仍在接收的 FIFO 行为 • 0: 覆盖 FIFO 中的数据 • 1: 丢弃正在接收的数据 LCD_EN 之前配置。
15	–	–	–
14:8	R	0x0	RD_FIFO_DEPTH 读 FIFO 深度 当前读 FIFO 数据数量 用户在发送一个读命令时，读回的数据在 FIFO 的深度，需要判断数据量是否符合预期，并且从 FIFO 取走数据并未全部取走时，次字段也需更新剩余深度。
7	–	–	–
6:0	R	0x0	WR_FIFO_DEPTH 写 FIFO 深度 当前写 FIFO 数据数量 用户在写操作启动之前，需判断 FIFO 内的数据量是否等于用户即将发送的数量 • 若深度不足，用户可判断是否软件行为的问题。 • 若深度达到预期，则软件可启动读操作。

10.4.11.29. 0x224 SPI_INT_ENABLE

默认值：0x00000000			SPI 模式中断使能 (SPI Interrupt Enable)
位域	类型	默认值	描述
31:4	–	–	–
3	R/W	0x0	TX_UNDER

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			SPI 模式中断使能 (SPI Interrupt Enable)
位域	类型	默认值	描述
			发送下溢出中断 • 0：禁止 • 1：使能 写操作时，需要发送 N byte 数据，但 FIFO 内的数据少于 N，则 FIFO 在数据发完后发生下溢出中断。
2	R/W	0x0	TX_OVER 发送上溢出中断 • 0：禁止 • 1：使能 写操作时，FIFO 深度 为 64 bytes，但写入 FIFO 数据大于 64 bytes，则 FIFO 发生上溢出中断。
1	R/W	0x0	RX_UNDER 接收下溢出中断 • 0：禁止 • 1：使能 读操作时，FIFO 内数据为空，但此时用户读取 FIFO 的数据，则发生下溢出中断
0	R/W	0x0	RX_OVER 接收上溢出中断 • 0：禁止 • 1：使能 读操作时，FIFO 的深度为 64 bytes，但读回的数据大于 64 bytes，则发生上溢出中断。

10.4.11.30. 0x228 SPI_INT_CLR

默认值：0x00000000			SPI 模式中断清除 (SPI Interrupt Clear)
位域	类型	默认值	描述
31:4	–	–	–
3	R/W1C	0x0	TX_UNDER_CLR 发送下溢出中断 • 0: 未触发 • 1: 触发
2	R/W1C	0x0	TX_OVER_CLR 发送上溢出中断

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			SPI 模式中断清除 (SPI Interrupt Clear)
位域	类型	默认值	描述
			• 0: 未触发 • 1: 触发
1	R/W1C	0x0	RX_UNDER_CLR 接收下溢出中断 • 0: 未触发 • 1: 触发
0	R/W1C	0x0	RX_OVER_CLR 接收上溢出中断 • 0: 未触发 • 1: 触发

10.4.11.31. 0x22C SPI_STATUS

默认值：0x00000003			SPI 模式状态 (SPI Status)
位域	类型	默认值	描述
31:5	–	–	–
4	R	0x0	SPI_IDELSPI 空闲状态 • 0: SPI 空闲状态 • 1: SPI 非空闲状态
3	R	0x0	RX_FIFO_FULL 接收 FIFO 为满状态 • 0: 接收 FIFO 非满 • 1: 接收 FIFO 为满
2	R	0x0	RX_FIFO_NOT_EMPTY 接收 FIFO 非空状态 • 0: 接收 FIFO 为空 • 1: 接收 FIFO 非空
1	R	0x1	TX_FIFO_EMPTY 发送 FIFO 为空状态

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000003			SPI 模式状态 (SPI Status)
位域	类型	默认值	描述
			• 0: 发送 FIFO 非空 • 1: 发送 FIFO 为空
0	R	0x1	TX_FIFO_NOT_FULL 发送 FIFO 非满状态 • 0: 发送 FIFO 为满 • 1: 发送 FIFO 非满

10.4.11.32. 0x230 SPI_RD_MODE

默认值：0x00000000			SPI 模式读模式 (SPI Read Mode)
位域	类型	默认值	描述
31:1	–	–	–
0	R/W	0x0	SPI_RD_MODE SPI 读模式选择，仅三线与四线 SPI 有效 • 0: SDI 与 SDO 独立完成输入输出功能 • 1: SDO 可完成输入输出，弃用 SDI

10.4.11.33. 0x234 QSPI_MODE

默认值：0x00000000			QSPI 模式 (QSPI Mode)
位域	类型	默认值	描述
31:24	–	–	–
23:16	R/W	0x0	CODE1_CFG Bit0 配 1 时，本字段为空白行的 CODE1。 配置本字段不影响有效行的 CODE1。
15:8	R/W	0x0	VBP_NUM 空白行数量 Bit0 配 1 时，硬件读取此字段，表示每帧开始后有多少个空白行
7:1	–	–	–
0	R/W	0x0	QSPI_MODE QSPI 刷屏模式选择

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			QSPI 模式 (QSPI Mode)
位域	类型	默认值	描述
			• 0: 无时序信息，应用于带 buffer 屏 • 1: 有时序信息

10.4.11.34. 0xFF0 DEBUG_SEL

默认值：0x00000000			调试信号选择 (Debug Select)
位域	类型	默认值	描述
31:3	–	–	–
2:0	R/W	0x0	DEBUG_SEL Debug 选择寄存器 • 0: debug0 • 1: debug1 • 2: debug2 • 3: debug3 • 4: debug4

10.4.11.35. 0xFF4 DEBUG

默认值：0x00000000			调试 (Debug)
位域	类型	默认值	描述
31:0	R/W	0x0	DEBUG Debug 寄存器

10.4.11.36. 0xFFC VERSION

默认值：0x00000102			版本号 (Version)
位域	类型	默认值	描述
31:0	RO	0x00000102	Version 模块版本号 采用 BCD 码显示，V1.2

10.5. Audio

Audio Codec 模块内置 DMIC 接口和 PWM 音频输出模块，经过数字信号的处理，实现音频信号的录入以及播放功能。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.5.1. 特性说明

- 支持数字 DMIC 立体声音频输入
- 无 DAC 设计，采用 PWM 输出
- 支持 2x 单端输出（立体声输出），或 1x 差分输出（单声道输出）
- 输入输出通路均支持数字音量控制
- 提供 128 x 32–bit FIFO 用于 DMIC 通路的数据
- 提供 128 x 32–bit FIFO 用于 Playback 通路的数据
- 支持 CPU 和 DMA 两种数据传输实现方式

10.5.2. 原理框图

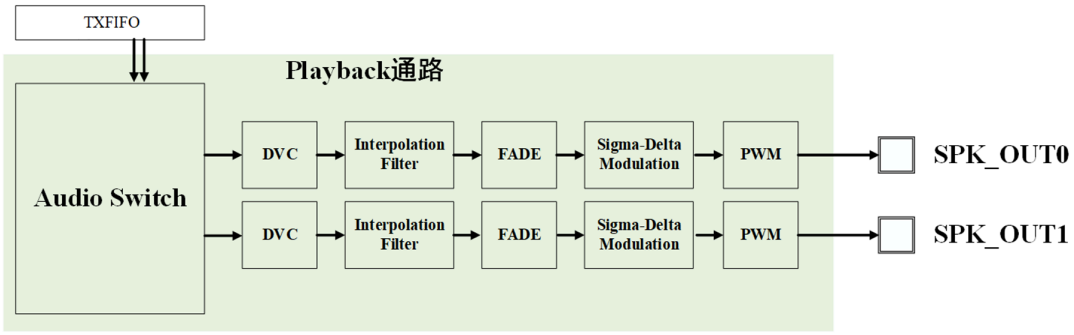


图 10–39 原理框图

图 10–39 : 原理框图中列出的组成部分包括：

- Playback 通路：用于播放。其中 Audio Switch 模块用于选择 DSPK0 和 DSPK1 的数据源。
- FIFO 数据：用于缓冲收发数据，接收即是录音，发送即是播放。

10.5.3. 功能描述

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.5.3.1. Audio Codec 时钟

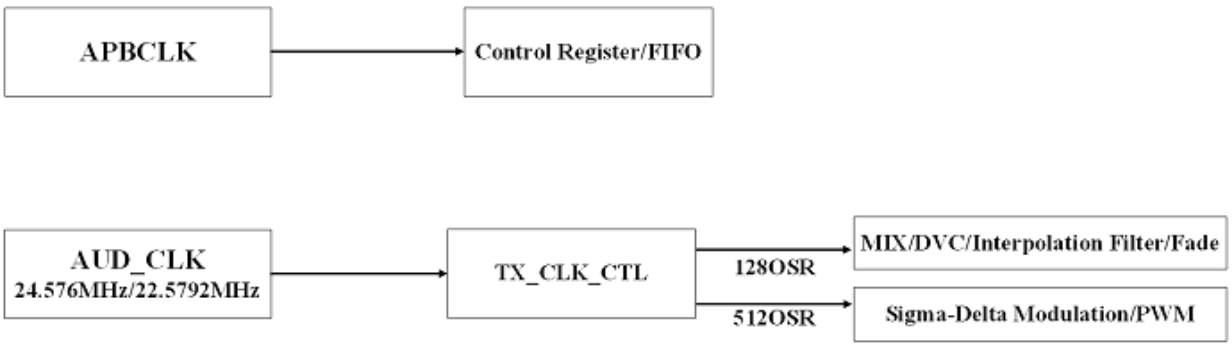


图 10–40 Audio Codec 时钟框图

如图所示：

- APBCLK：用于控制寄存器以及 FIFO。
- AUD_CLK：控制 TX（播放）部分，24.576 MHz / 22.5792 MHz 由 CMU 分频所得。

  TX 部分通过寄存器 FS_OUT 配置了 TX 的采样频率：

  - 过采样率 128 倍的时钟则提供 MIX、DVC、Interpolation Filter、Fade 模块。
  - 过采样率 512 倍的时钟则提供 Sigma–Delta Modulation、PWM 模块。

10.5.3.2. Audio Codec 信号通路

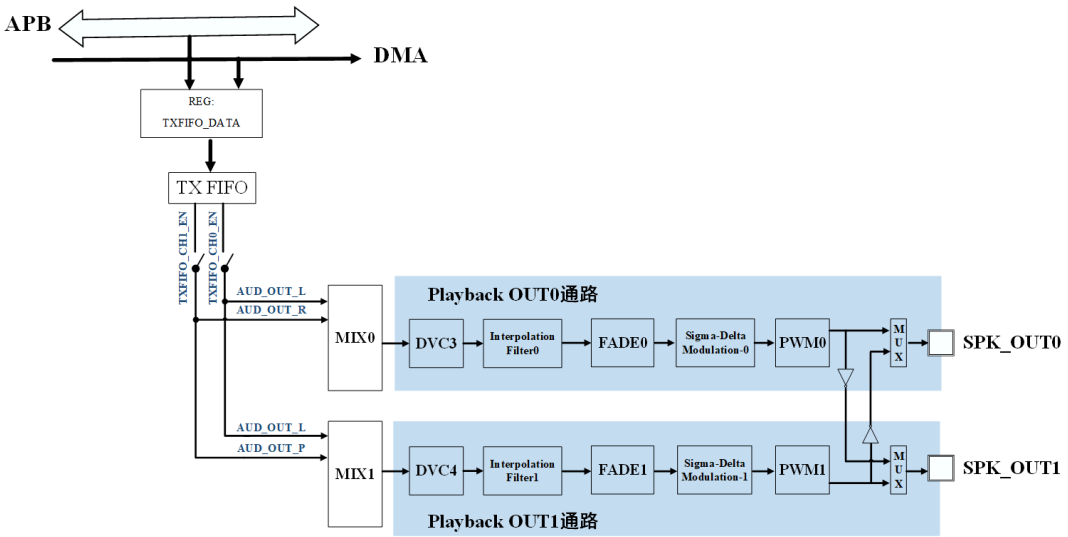


图 10–41 Audio Codec 信号通路

图 10–41 : Audio Codec 信号通路中：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

1. 通过 TXFIFO_CH0_EN/ TXFIFO_CH1_EN，分别控制 TXFIFO 中的左/右声道的数据。
2. 如需交叉 OUT0 通路和 OUT1 通路的数据，可以通过 MIX0、MIX1 的寄存器配置，实现 OUT0/1 通路的数据交换。

10.5.3.2.1. Audio Codec 信号通路中的模块功能

Audio Codec 信号通路涉及下列功能模块，其功能如下所示：

- 数字音量控制 (DVC)：
  - 每个 TX、RX 通路上均有设计独立的 DVC 模块。
  - 通过 DVC_EN 使能。如未使能，则旁通输入信号到下一个模块。
  - DVC 的可调范围为 –119.25 dB 到 71.25 dB，步进 0.75 dB。
- 插值滤波器 (Interpolation Filter)：
  - 实现 128 倍的上采样，实现抽值滤波器的相反功能。
  - 依次采用半带滤波、CIC 补偿滤波和 CIC 插值滤波。
  - 实现的上采样倍数分别为 2、2、32。经过插值滤波器，得到高采样率的数字信号。
- Sigma–Delta Modulation：
  - 经过插值滤波后，进行数字 Sigma–Delta 算法，获得 PDM 信号。
  - Sigma–Delta 调制采用两阶 1 bit 结构。
- PWM：
  - 用于对 PDM 进行调制，调制方式如下：
    - 0 代表 25% 占空比的时钟信号。
    - 1 代表 75% 占空比的时钟信号。

    时钟频率为 PDM 的采样时钟频率，即 128 × FS_OUT。
  - 支持差分模式。如 PWM0 使能差分模式：
    - SPK_OUT0 输出 PWM0。
    - SPK_OUT1 输出反相 PWM0。

    此时 PWM1 应配置为关闭状态。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.5.3.3. 中断

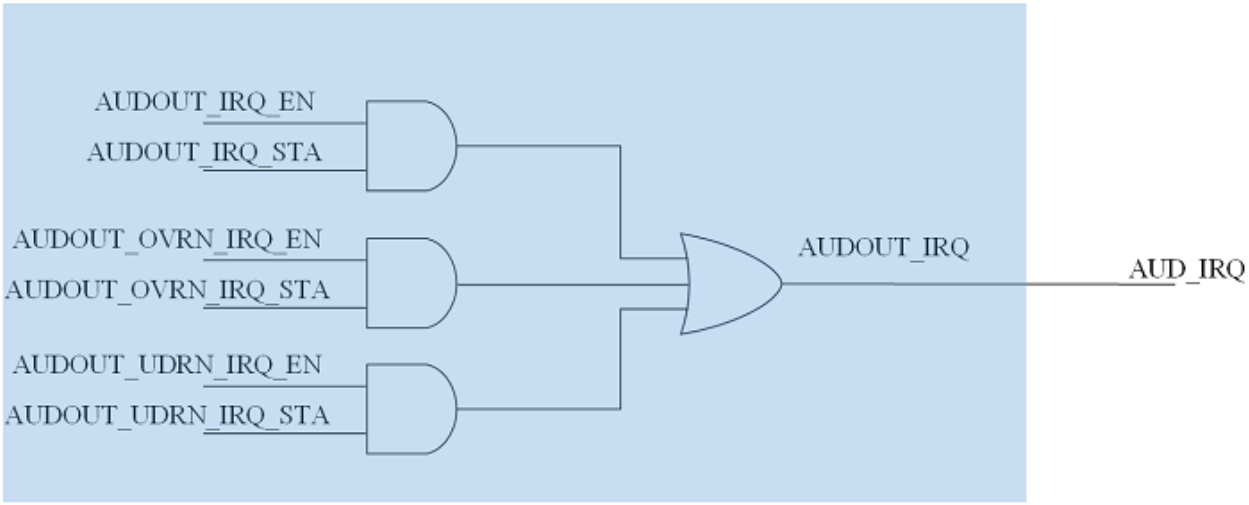


图 10–42 中断信号

Audio Codec 的图 10–42 : 中断信号具体描述如下：

- AUDOUT_IRQ_STA：TXFIFO 数据空间充足时产生的中断信号，即 TXFIFO 的剩余的数据空间多于 TXFIFO_TH 时，此时也会相应的产生 DRQ 信号。
- AUDOUT_OVRN_IRQ_STA：DMIC 的 RXFIFO 出现上溢出中断。
- AUDOUT _UDRN_IRQ_STA：DMIC 的 RXFIFO 出现下溢出中断。

10.5.4. 编程指南

常用场景参数配置如下所示：

1. 使用双通道播放，主时钟为 24.576MHz，采样频率为 48KHz ，过采样率为 128

   - TX MIXER 控制寄存器

     TX_MIXER_CTL[31:30] = 0x0，不使能混音功能，通路配置为 AUD_OUT。

     TX_MIXER_CTL[12:8] = 0x2，通路配置为 AUD_OUT_R。

     TX_MIXER_CTL[4:0] = 0x1，通路配置为 AUD_OUT_L。

   - TX DVC3&4 控制寄存器

     TX_DVC3&4_CTL[31:24] = 0xA0，DVC4 数字音量控制模块，根据需求配置。

     TX_DVC3&4_CTL[23:16] = 0xA0，DVC3 数字音量控制模块，根据需求配置。

     TX_DVC3&4_CTL[1:0] = 0x3，使能两路 DVC。

   - TX Playback 控制寄存器

     TX_PLAYBACK_CTL[14:12] = 0x0，延时输出功能，根据需求配置。

     TX_PLAYBACK_CTL[10:8] = 0x3，滤波输出 Shift 比特数配置为默认值 13 比特，推荐使用默认值。

     TX_PLAYBACK_CTL[6:4] = 0x7，插值滤波器使能。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

TX_PLAYBACK_CTL[3:0] = 0x0，主时钟配置 24.576 MHz，采样频率配置 48 KHz。

- TX SDM 控制寄存器

  TX_SDM_CTL[31:8] = 0x11011，SDM 增益默认值配置，推荐使用默认值。

  TX_SDM_CTL[1:0] = 0x3，SDM 使能。

- TX PWM 控制寄存器

  TX_PWM_CTL[6:0] = 0x11，PWM 使能并配置为输出 PWM 调制信号。

- TX FIFO 控制寄存器

  TXFIFO_CTL[2:0] = 0x3，使能 TXFIFO 通道，下溢出时发送 0。

10.5.5. 寄存器列表

表 10–12 AUDIO 寄存器列表

偏移	缩写	寄存器名称	章节
0x01C	TX_MIXER_CTL	TX Mixer 控制寄存器	0x01C TX_MIXER_CTL
0x020	TX_DVC3&4_CTL	TX 数字音量控制 3 寄存器	0x020 TX_DVC3&4_CTL
0x024	TX_PLAYBACK_CTL	播放控制寄存器	0x024 TX_PLAYBACK_CTL
0x028	TX_SDM_CTL	TX Sigma–Delta 控制寄存器	0x028 TX_SDM_CTL
0x034	TXFIFO_CTL	TXFIFO 控制寄存器	0x034 TXFIFO_CTL
0x038	FIFO_INT_EN	FIFO 中断使能寄存器	0x038 FIFO_INT_EN
0x048	TXFIFO_DATA	TXFIFO 数据寄存器	0x048 TXFIFO_DATA
0x04C	TX_CNT	TX 数据计数器寄存器	0x04C TX_CNT
0x058	FADE_CTL0	渐变控制 0 寄存器	0x058 FADE_CTL0
0x05C	FADE_CTL1	渐变控制 1 寄存器	0x05C FADE_CTL1
0x060	GLOBE_CTL	全局控制寄存器	0x060 GLOBE_CTL
0xFC	VERSION	版本号	0x0FC VERSION

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.5.6. 寄存器描述

10.5.6.1. 0x01C TX_MIXER_CTL

默认值：0x00000000			TX Mixer 控制寄存器（TX Mixer Control）
位域	类型	默认值	描述
31	R/W	0x0	MIX0_EN MIX0 使能控制信号 • 0x0：不使能 • 0x1：使能 若 MIX0_EN = 0，MIX0 则为通路选择，MIX0_SRC_SEL 只有 0x1/0x2/0x4/0x8/0x10 有效，旁路通过所选择的信号到下一个模块。
30	R/W	0x0	MIX1_EN MIX1 使能控制信号 • 0x0：不使能 • 0x1：使能 若 MIX1_EN = 0，MIX1 则为通路选择，MIX1_SRC_SEL 只有 0x1/0x2/0x4/0x8/0x10 有效，旁路通过所选择的信号到下一个模块。
29	–	–	–
28:24	R/W	0x0	MIX1_GAIN_CTL Mixer1 混频源的增益控制，相应 BIT 置 1 表示 –6dB 增益使能，相应 BIT 清 0 表示无增益使能。 • BIT24：AUD_OUT_L • BIT25：AUD_OUT_R • BIT26：Reserved • BIT27：Reserved • BIT28：Reserved
23:21	–	–	–
20:16	R/W	0x0	MIX0_GAIN_CTL Mixer0 混频源的增益控制，相应 BIT 置 1 表示 –6 dB 增益使能，相应 BIT 清 0 表示无增益使能。 • BIT16：AUD_OUT_L • BIT17：AUD_OUT_R

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			TX Mixer 控制寄存器（TX Mixer Control）
位域	类型	默认值	描述
			• BIT18：Reserved • BIT19：Reserved • BIT20：Reserved
15:13	–	–	–
12:8	R/W	0x0	MIX1_SRC_SEL Mixer1 混音源选择，相应 BIT 置 1 表示使能 • BIT8：AUD_OUT_L • BIT9：AUD_OUT_R • BIT10：Reserved • BIT11：Reserved • BIT12：Reserved 在旁通模式下，低 bit 的优先级最高，例如此域为 0xF，则旁通 L。
7:5	–	–	–
4:0	R/W	0x0	MIX0_SRC_SEL Mixer0 混音源选择，相应 BIT 置 1 表示使能 • BIT0：AUD_OUT_L • BIT1：AUD_OUT_R • BIT2：Reserved • BIT3：Reserved • BIT4：Reserved 在旁通模式下，低 bit 的优先级最高，例如此域为 0xF，则旁通 L。

10.5.6.2. 0x020 TX_DVC3&4_CTL

默认值：0xA0A00000			TX 数字音量控制 3 寄存器（TX DVC3）
位域	类型	默认值	描述
31:24	R/W	0xA0	DVC4 数字音量控制模块 DVC4 的增益 BIT[31:27] = N, BIT [26:24] = K，delta_dB = 0.75 × K+6.02 × (N–20)（–119.25 dB to 71.25 dB，0.75 dB/ Step）

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0xA0A00000			TX 数字音量控制 3 寄存器（TX DVC3）
位域	类型	默认值	描述
			• 0x00：静音 • 0x01：–119.25 dB • … • 0x9F：–0.75 dB • 0xA0：0 dB • 0xA1：0.75 dB • … • 0xFF：71. 25 dB
23:16	R/W	0xA0	DVC3 数字音量控制模块 DVC3 的增益 BIT [23:19] = N, BIT [18:16] = K，（–119.25 dB to 71.25 dB，0.75 dB/ Step） • 0x00：静音 • 0x01：–119.25 dB • … • 0x9F：–0.75 dB • 0xA0：0 dB • 0xA1：0.75 dB • … • 0xFF：71. 25 dB
15:2	–	–	–
1	R/W	0x0	DVC4_EN DVC4 使能控制位 • 0x0：关闭 DVC • 0x1：使能 DVC 若关闭 DVC，数据信号不经过处理，旁通到下一个模块。
0	R/W	0x0	DVC3_EN DVC3 使能控制位 • 0x0：关闭 DVC • 0x1：使能 DVC 若关闭 DVC，数据信号不经过处理，旁通到下一个模块。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.5.6.3. 0x024 TX_PLAYBACK_CTL

默认值：0x00000300			播放控制寄存器（TX Playback Control）
位域	类型	默认值	描述
31:15	–	–	–
14:13	R/W	0x0	TX_DLT TX 上采样滤波输出的延迟时间 • 0x0：5 ms • 0x1：10 ms • 0x2：20 ms • 0x3：30 ms
12	R/W	0x0	TX_DLT_EN TX 上采样滤波输出延迟功能使能控制 • 0x0：禁用 • 0x1：使能
11	–	–	–
10:8	R/W	0x3	FLTOUT_SHIFTN 滤波输出 Shift 比特数 • 0x0：10 比特 • 0x1：11 比特 • … • 0x7：17 比特
7	–	–	–
6	R/W	0x0	EN_TX_CH1 Playback 发送通道 1 插值滤波器控制位 • 0x0：不使能 • 0x1：使能
5	R/W	0x0	EN_TX_CH0 Playback 发送通道 0 插值滤波器控制位 • 0x0：不使能 • 0x1：使能
4	R/W	0x0	PLAYBACK_INTRE_EN Playback 插值滤波器使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000300			播放控制寄存器（TX Playback Control）
位域	类型	默认值	描述
			• 0x0：关闭 • 0x1：使能
3:1	R/W	0x0	FS_OUT 音频回放输出的采样频率 当 AUD_CLK=24.576MHz 时： • 0x0：48 KHz • 0x1：32 KHz • 0x2：24 KHz • 0x3：16 KHz • 0x4：12 KHz • 0x5：8 KHz • 0x6：Reserved • 0x7：Reserved 当 AUD_CLK=22.5792MHz 时： • 0x0：44.1 KHz • 0x2：22.05 KHz • 0x4：11.025 KHz • Others：Reserved
0	R/W	0x0	TX_CLK_FRE TX 模块所选用的工作主时钟 • 0x0：24.576 MHz • 0x1：22.5792 MHz

10.5.6.4. 0x028 TX_SDM_CTL

默认值：0x01101100			TX Sigma-Delta 控制寄存器（TX Sigma-Delta Module Control）
位域	类型	默认值	描述
31:30	R/W	0x0	TX_CH1_SDM_GAIN_C1 TX Channel1 的 SDM 增益 C1 设置 • 0x0：增益为 1 • 0x1：增益为 0.5

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x01101100			TX Sigma-Delta 控制寄存器（TX Sigma-Delta Module Control）
位域	类型	默认值	描述
			• 0x2：增益为 0.25 • 0x3：增益为 0.125
29:28	R/W	0x0	TX_CH1_SDM_GAIN_B1 TX Channel1 的 SDM 增益 B1 设置 • 0x0：增益为 1 • 0x1：增益为 0.5 • 0x2：增益为 0.25 • 0x3：增益为 0.125
27:26	R/W	0x0	TX_CH1_SDM_GAIN_A1 TX Channel1 的 SDM 增益 A1 设置 • 0x0：增益为 1 • 0x1：增益为 0.5 • 0x2：增益为 0.25 • 0x3：增益为 0.125
25:24	R/W	0x1	TX_CH1_SDM_GAIN_C0 TX Channel1 的 SDM 增益 C0 设置 • 0x0：增益为 1 • 0x1：增益为 0.5 • 0x2：增益为 0.25 • 0x3：增益为 0.125
23:22	R/W	0x0	TX_CH1_SDM_GAIN_B0 TX Channel1 的 SDM 增益 B0 设置 • 0x0：增益为 1 • 0x1：增益为 0.5 • 0x2：增益为 0.25 • 0x3：增益为 0.125
21:20	R/W	0x1	TX_CH1_SDM_GAIN_A0 TX Channel1 的 SDM 增益 A0 设置 • 0x0：增益为 1 • 0x1：增益为 0.5

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x01101100			TX Sigma-Delta 控制寄存器（TX Sigma-Delta Module Control）
位域	类型	默认值	描述
			• 0x2：增益为 0.25 • 0x3：增益为 0.125
19:18	R/W	0x0	TX_CH0_SDM_GAIN_C1 TX Channel0 的 SDM 增益 C1 设置 • 0x0：增益为 1 • 0x1：增益为 0.5 • 0x2：增益为 0.25 • 0x3：增益为 0.125
17:16	R/W	0x0	TX_CH0_SDM_GAIN_B1 TX Channel0 的 SDM 增益 B1 设置 • 0x0：增益为 1 • 0x1：增益为 0.5 • 0x2：增益为 0.25 • 0x3：增益为 0.125
15:14	R/W	0x0	TX_CH0_SDM_GAIN_A1 TX Channel0 的 SDM 增益 A1 设置 • 0x0：增益为 1 • 0x1：增益为 0.5 • 0x2：增益为 0.25 • 0x3：增益为 0.125
13:12	R/W	0x1	TX_CH0_SDM_GAIN_C0 TX Channel0 的 SDM 增益 C0 设置 • 0x0：增益为 1 • 0x1：增益为 0.5 • 0x2：增益为 0.25 • 0x3：增益为 0.125
11:10	R/W	0x0	TX_CH0_SDM_GAIN_B0 TX Channel0 的 SDM 增益 B0 设置 • 0x0：增益为 1 • 0x1：增益为 0.5

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x01101100			TX Sigma-Delta 控制寄存器（TX Sigma-Delta Module Control）
位域	类型	默认值	描述
			• 0x2：增益为 0.25 • 0x3：增益为 0.125
9:8	R/W	0x1	TX_CH0_SDM_GAIN_A0 TX Channel0 的 SDM 增益 A0 设置 • 0x0：增益为 1 • 0x1：增益为 0.5 • 0x2：增益为 0.25 • 0x3：增益为 0.125
7:2	–	–	–
1	R/W	0x0	TX_CH1_SDM_EN 发送 CH1 的 Sigma-Delta Modulate 模块使能 • 0x0：不使能 • 0x1：使能
0	R/W	0x0	TX_CH0_SDM_EN 发送 CH0 的 Sigma-Delta Modulate 模块使能 • 0x0：不使能 • 0x1：使能

10.5.6.5. 0x034 TXFIFO_CTL

默认值：0x00004000			TXFIFO 控制寄存器（TXFIFO Control）
位域	类型	默认值	描述
31	R/W1AC	0x0	TX_FIFO_FLUSH 擦除 TXFIFO 数据，写 1 擦除 TXFIFO 数据，完成自动清 0。
30:16	–	–	–
15:8	R/W	0x40	TXFIFO_TH TX FIFO Threshold，TX FIFO 的阈值，表示 TXFIFO 剩余可用空间值，单位 32bit word。 当 TXFIFO 的剩余可用空间值小于 TH 时，则认为 TXFIFO 空，产生空中断以及 DRQ 请求。
7:3	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00004000			TXFIFO 控制寄存器（TXFIFO Control）
位域	类型	默认值	描述
2	R/W	0x0	TX_UR_DATA 在 TXFIFO 出现下溢出的时候，选择发送的数据形式 • 0x0：发送上一次的音频数据 • 0x1：发送全 0
1	R/W	0x0	TX_FIFO_CH1_EN TXFIFO 通道 1 的使能控制位 • 0x0：不使能 • 0x1：使能
0	R/W	0x0	TX_FIFO_CH0_EN TXFIFO 通道 0 的使能控制位 • 0x0：不使能 • 0x1：使能

10.5.6.6. 0x038 FIFO_INT_EN

默认值：0x00000000			FIFO 中断使能寄存器（FIFO Interrupt Enable）
位域	类型	默认值	描述
31:8	–	–	–
7	R/W	0x0	AUDOUT_DRQ_EN TXFIFO 空状态请求 DMA 使能位 • 0x0: 禁止 DRQ 请求 • 0x1: 使能 DRQ 请求
6	R/W	0x0	AUDOUT_UDRN_IRQ_EN TXFIFO 下溢出中断使能位 • 0x0: 禁止中断请求 • 0x1: 使能中断请求
5	R/W	0x0	AUDOUT_OVRN_IRQ_EN TXFIFO 上溢出中断使能位 • 0x0: 禁止中断请求 • 0x1: 使能中断请求

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			FIFO 中断使能寄存器（FIFO Interrupt Enable）
位域	类型	默认值	描述
4	R/W	0x0	AUDOUT_IRQ_EN TXFIFO 空状态中断使能位 • 0x0: 禁止中断请求 • 0x1: 使能中断请求
3:0	–	–	–

10.5.6.7. 0x048 TXFIFO_DATA

默认值：0x00000000			TXFIFO 数据寄存器（TXFIFO Data）
位域	类型	默认值	描述
31:0	WO	0x0	TXFIFO_DATA TXFIFO 发送的采样点数据，往此地址写数据进行发送。低 16 位为 TXChannel0 的 16-bit，高 16 位为 Channel1 的 16-bit。

10.5.6.8. 0x04C TX_CNT

默认值：0x00000000			TX 数据计数器寄存器（TX Data Counter）
位域	类型	默认值	描述
31:0	R/W	0x0	TXFIFO_DATA_CNT TX FIFO 发送的采样点数据的数量统计，初始值可以在任意时间写入任意值。 每当有一个采样点写入 DATA，TX FIFO 发送的采样点数据计数自动加 1。

10.5.6.9. 0x058 FADE_CTL0

默认值：0x00804007			渐变控制 0 寄存器（Fade Control 0）
位域	类型	默认值	描述
31	–	–	–
30:16	R/W	0x80	FADE_STEP 音量变化步进，配合 VOL 计算渐变总时间。
15:12	R/W	0x4	FADE_SPEED

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00804007			渐变控制 0 寄存器（Fade Control 0）
位域	类型	默认值	描述
			音量变化速度，2^{FADE_SPEED}，，配合 VOL 计算渐变总时间。单位为 us。
11:3	–	–	–
2	R/W	0x1	FADE_CH1_EN FADE 使能通道 1 控制位 • 0x0：不使能 • 0x1：使能
1	R/W	0x1	FADE_CH0_EN FADE 使能通道 0 控制位 • 0x0：不使能 • 0x1：使能
0	R/W	0x1	FADE _EN FADE 功能使能控制位 • 0x0：不使能 • 0x1：使能

10.5.6.10. 0x05C FADE_CTL1

默认值：0x7FFF7FFF			渐变控制 1 寄存器（Fade Control 1）
位域	类型	默认值	描述
31	–	–	–
30:16	R/W	0x7FFF	MAX_VOL 最大音量，配置为默认值即可。
15	–	–	–
14:0	R/W	0x7FFF	TARGET_VOL 目标音量，目标音量 ≤ 最大音量。 目标音量是音量渐变的目标值： • 打开播放通路时需要实现渐变，此位配置为 0x7FFF。 • 关闭播放通路时需要实现渐变，此位配置为 0x0000。 渐变过程的总时间计算公式如下： ((TARGET_VOL – 当前值) ÷ FADE_STEP) × FADE_SPEED

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x7FFF7FFF			渐变控制 1 寄存器（Fade Control 1）
位域	类型	默认值	描述
			例如，FADE_STEP = 0x80，FADE_SPEED = 0x4，打开播放通路的渐变过程所需总时间为 ((0x7FFF–0)/0x80) × 2^4 = 4096 us。

10.5.6.11. 0x060 GLOBE_CTL

默认值：0x00000000			全局控制寄存器（Global Control）
位域	类型	默认值	描述
31:12	–	–	–
11:8	R/W	0x0	DBG_SIGNAL Debug 信号选择，选择的信号通过 PWM_OUT 输出。
7:4	R/W	0x0	DEBUG_MOD DEBUG 模式 • 0x0：DEBUG 模式不使能 • 0x1：TX 发送一个正弦信号 • 0x7：根据 signal 选择输出信号 • 0x8：TX 发送差分正弦波信号 • Others：Reserved
3	–	–	–
2	R/WAC	0x0	GLB_RST 全局复位控制信号，此位写 1 进行复位，复位完成自动清 0。
1	R/W	0x0	TX_GLBEN 发送端全局使能控制信号 • 0x0：不使能 • 0x1：使能
0	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

10.5.6.12. 0x0FC VERSION

默认值：0x00000102			Codec 版本号（VERSION）
位域	类型	默认值	描述
31:0	RO	0x00000102	Version(模块版本号) 采用 BCD 码显示，V1.2

ArtInChip

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

11. 计时器

11.1. Generic Timer Controller (GTC)

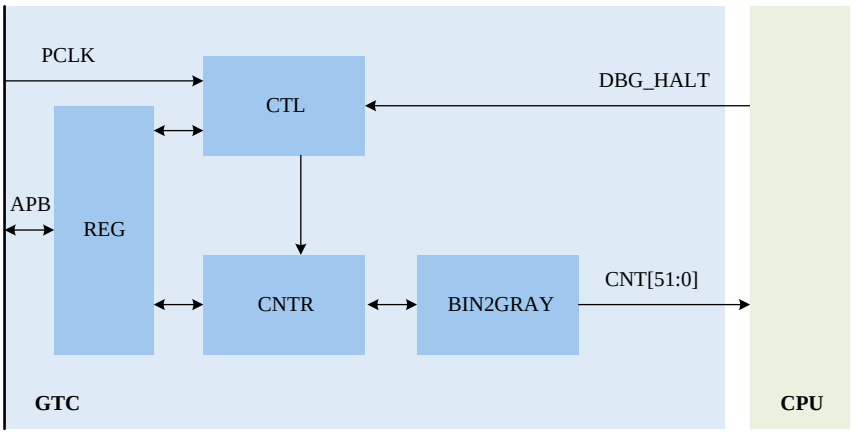
General Timer Counter (GTC) 通用计数器模块，为 CPU 提供基础时钟。GTC 模块输出计数值为 52 bits，计数的最高频率（基频）为 4 MHz，计数周期约为 35 年。

GTC 模块输出为 52 bits，但 CPU 输入为 64 bits，因此计数值高位寄存器的高 12 bits 固定为 0。

11.1.1. 特性说明

- 总线时钟 PCLK 的频率可选，但必须为 4 MHz 的整数倍频，例如 12 MHz/ 24 MHz/ 48 MHz/ 60 MHz 等
- 计数器最高计数频率（基频）为 4 MHz，计数精度为 0.25 MHz，即 250 ns
- 计数时钟频率误差不超过 10 s/24 h，约为 100 ppm
- 计数器位宽为 52 bits，计数周期约 35 年
- 计数频率可选，4 MHz/ 1 MHz/ 250 KHz
- 不同的计数频率，计数精度不变，例如 4 MHz 每周期增加 1，1 MHz 每周期增加 4，250 KHz 则每周期增加 16
- 52 bits 计数值以格雷码 (Gray Code) 编码方式输出
- 复位后模块计数从 0 开始，并且仅支持增量计数

11.1.2. 原理框图



Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

11.1.3. 寄存器列表

表 11–1 GTC 寄存器列表

偏移地址	缩写	寄存器名称	章节
0x0000	GTC_CNTCR	GTC 计数器控制	0x0000 GTC_CNTCR
0x0004	GTC_CNTSR	GTC 计数器控制	0x0004 GTC_CNTSR
0x0008	GTC_CNTVL	GTC 计数值低位	0x0008 GTC_CNTVL
0x000C	GTC_CNTVH	GTC 计数值高位	0x000C GTC_CNTVH
0x0020	GTC_CNTFID0	GTC 计数频率 ID0	0x0020 GTC_CNTFID0
0x0024	GTC_CNTFID1	GTC 计数频率 ID1	0x0024 GTC_CNTFID1
0x0028	GTC_CNTFID2	GTC 计数频率 ID2	0x0028 GTC_CNTFID2
0x002C	GTC_CNTFID3	GTC 计数频率 ID3	0x002C GTC_CNTFID3
0x00C0	GTC_CONFG	GTC 配置	0x00C0 GTC_CONFG
0x0FFC	GTC_VERSION	GTC 版本	0x0FFC GTC_VERSION

11.1.4. 寄存器描述

11.1.4.1. 0x0000 GTC_CNTCR

默认值：0x00000000			GTC 计数器控制 (GTC Counter Control)
位域	类型	默认值	描述
31:10	–	–	–
9:8	R/W	0x0	FCREQ 计数频率调整请求，本字段控制计数器计数频率选择： • 0：计数频率为基频 4 MHz • 1：计数频率为 1 MHz • 2：计数频率为 250 KHz • 3：未定义 写入新的值后，GTC 模块会切换计数频率，可通过查询状态寄存器 CNTSR 的 FCACK 确认是否切换完成。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			GTC 计数器控制 (GTC Counter Control)
位域	类型	默认值	描述
7:2	–	–	–
1	R/W	0x0	CNT_ON_DBG 调试模式下继续计时 • 0：暂停计时 • 1：继续计时
0	R/W	0x0	EN 使能控制，仅控制计数器是否正常工作，不会复位计数器的其它配置。 • 0：计数器模块停止工作 • 1：计数器模块正常工作

11.1.4.2. 0x0004 GTC_CNTSR

默认值：0x00000000			GTC 计数器控制 (GTC Counter Control)
位域	类型	默认值	描述
31:10	–	–	–
9:8	RO	0x0	FCACK 当前计数频率 显示正在使用的计数频率点,可以通过查询本字段确定频率切换是否完成。 • 0：当前计数频率为基频 4 MHz • 1：当前计数频率为 1 MHz • 2：当前计数频率为 250 KHz • 3：未定义
7:0	–	–	–

11.1.4.3. 0x0008 GTC_CNTVL

默认值：0x00000000			GTC 计数值低位 (GTC Counter Value Low)
位域	类型	默认值	描述
31:0	R/W	0x00000000	CNTVL 计数值低位

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			GTC 计数值低位 (GTC Counter Value Low)
位域	类型	默认值	描述
			显示计数器的低 32 bits。 • 计数频率为 250 KHz 时，计数值保持低 4 bits 恒定为 0。 • 计数频率为 1 MHz 时，计数值保持低 2 bits 恒定为 0。 **计数器使能后，本寄存器禁止写操作。** **模块复位后，本寄存器清零。**

11.1.4.4. 0x000C GTC_CNTVH

默认值：0x00000000			GTC 计数值高位 (GTC Counter Value High)
位域	类型	默认值	描述
31:20	–	–	–
19:0	R/W	0x000000	CNTVH 计数值高位 本寄存器显示 52–bit 计数器的高 20 bits。 **计数器使能后，本寄存器禁止写操作。** **模块复位后，本寄存器清零。**

11.1.4.5. 0x0020 GTC_CNTFID0

默认值：0x003D0900			GTC 计数频率 ID0 (GTC Counter Frequencey ID0)
位域	类型	默认值	描述
31:0	RO	0x003D0900	CNTFID0 计数频率 0，单位为 Hz 保存计数器的计数频率 0 的频率值，即 4 MHz。

11.1.4.6. 0x0024 GTC_CNTFID1

默认值：0x000F4240			GTC 计数频率 ID1 (GTC Counter Frequencey ID1)
位域	类型	默认值	描述
31:0	RO	0x000F4240	CNTFID1 计数频率 1，单位为 Hz

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x000F4240			GTC 计数频率 ID1 (GTC Counter Frequencey ID1)
位域	类型	默认值	描述
			保存计数器的计数频率 1 的频率值，即 1 MHz。

11.1.4.7. 0x0028 GTC_CNTFID2

默认值：0x0003D090			GTC 计数频率 ID2 (GTC Counter Frequencey ID2)
位域	类型	默认值	描述
31:0	RO	0x0003D090	CNTFID2 计数频率 2，单位为 Hz 保存计数器的计数频率 2 的频率值，即 250 KHz。

11.1.4.8. 0x002C GTC_CNTFID3

默认值：0x00000000			GTC 计数频率 ID3 (GTC Counter Frequencey ID3)
位域	类型	默认值	描述
31:0	RO	0x00000000	CNTFID3 计数频率 3

11.1.4.9. 0x00C0 GTC_CONFG

默认值：0x00000600			GTC 配置
位域	类型	默认值	描述
31:13	–	–	–
12:8	R/W	0x06	FDIV 总线时钟/计数基频，即 PCLK 与基频 4 MHz 的分频系数。 APB 总线时钟 PCLK 默认为 24 MHz，FDIV 可以采用默认值 6。
7:0	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

11.1.4.10. 0x0FFC GTC_VERSION

默认值：0x00000100			GTC 版本 (Version)
位域	类型	默认值	描述
31:0	R/W	0x00000100	VERSION 模块版本，V1.0

11.2. Watch Dog (WDOG)

Watch Dog (WDOG) 由一个 32 位计时器进行计数产生信号，用于监控系统。CPU 正常运行时，当计时器达到清零阈值后，可通过 CPU 操作定时清除 WDOG 计时器，保证 WDOG 模块不会超时并触发复位。当 WDOG 计时器先后达到中断和复位阈值时，WDOG 模块就会产生中断和复位。

11.2.1. 特性说明

- 固定 32K 时钟源输入，来源于 24 MHz 晶体，经过 750 分频
- 内部实现一个 32 位计时器，计时单位为一个 32K 时钟周期
- 实现两个信号输出，分别用于产生系统复位和中断信号
- 通过指令方式复位计时器和切换超时配置，支持写入窗口设置
- 调试模式下，可配置暂停计时或继续计时

11.2.2. 原理框图

11.2.3. 功能描述

11.2.3.1. WDOG 工作流程

图 11–1 : WDOG 工作流程涉及以下参数和步骤：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

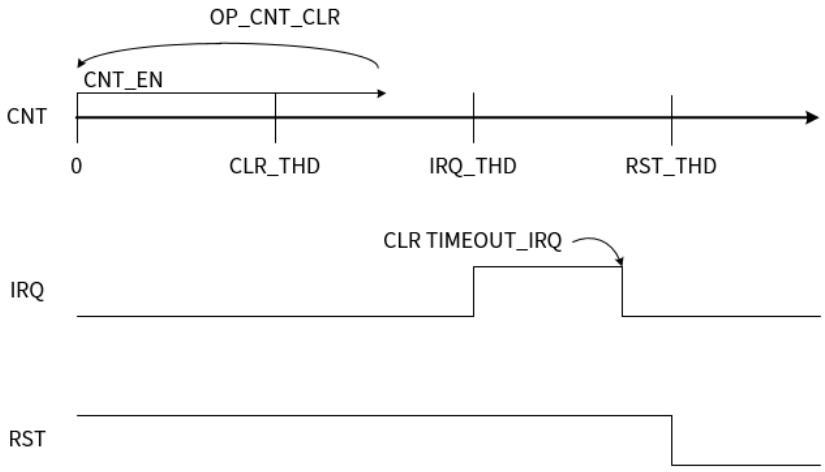


图 11-1 WDOG 工作流程

1. 计时器使能 (CNT_EN = 1) 并开始计时。计时以每个 32K 时钟周期为单位加一。

   使能计时器后，建议用户切换为写保护模式，具体配置可查看寄存器写保护功能。

2. 计时器达到清零阈值 (CNT > CLR_THD)。

   - 如系统运行正常，即可通过 CPU 执行 OP_CNT_CLR 对计时器进行清零。此时不会产生中断或者复位。

     关于 OP_CNT_CLR 的操作说明，可查看 功能描述。

   - 如系统出现异常，无法通过 CPU 对计时器进行清零，则计时器继续计时：

     注：

     为了最大化防止以下中断或复位情况中 WDOG 功能被关闭，可设置寄存器写保护功能。

     a. 计时器先达到中断阈值 (CNT > IRQ_THD)：计时器产生 CLR TIMEOUT_IRQ 中断，保存异常现场，该中断可被清除。

     b. 计时器达到复位阈值 (CNT > RST_THD)：计时器产生复位，复位整个系统。

        如需系统强制复位，设置 RST_THD 的值为较小值（例如 32），并执行 OP_CNT_CLR 进行复位。

11.2.3.2. 寄存器写保护功能

WDOG 具有寄存器写保护功能，针对异常状态下处理器的未知行为，以防止 WDOG 功能被关闭。寄存器写保护功能使能后，只有 OP 寄存器和 IRQ_STA 寄存器可被访问，只能执行 OP 指令和清中断操作。关于 OP 指令的详细说明，可查看 功能描述。

寄存器写保护功能具有多种模式，可通过 REG_WR_DIS 位域进行配置，具体模式和配置说明如下：

注：

关于 REG_WR_DIS 的说明，可查看寄存器描述。

- 写允许 (REG_WR_DIS = 0)：正常写寄存器，在 WDOG 初始化时使用。

  设置 CNT_EN = 1 后建议切换为写失效模式。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 写保护 (REG_WR_DIS = 1/2)：在需要动态配置 WDOG 的方案中使用。

  配置此寄存器前需先执行 OP_WR_EN，寄存器配置完成后再配置 REG_WR_DIS = 1/2，可以保护异常下不被误写。

- 写失效 (REG_WR_DIS = 3)：在无需动态配置 WDOG 方案中使用。

  配置完成后设置 CNT_EN = 1 && REG_WR_DIS = 3，运行期间配置不再被改写。

如有多场景需求且无需进行动态配置参数，可在初始化时完成多个场景配置后，设置 REG_WR_DIS = 3，应用过程只需要进行场景切换。 场景切换详细描述见场景切换。

11.2.3.3. 场景切换

WDOG 是基于 CLR_THD、IRQ_THD 和 RST_THD 三个参数为一组配置工作的，每个参数各有四个参数寄存器，用 n 进行区分，n = 0 ~ 3，详情可查看对应的寄存器描述。

通过设置参数寄存器，可组成不同的场景配置。WDOG 可支持保存最多四种组合场景，切换场景的 OP 指令分别为 OP_SW_CFG0、OP_SW_CFG1、OP_SW_CFG2 和 OP_SW_CFG3。

进行场景切换需要执行 OP 指令：

1. 执行 OP_SW_CFGn (n=0~3) 切换到配置好的第 n 个场景。

   若只需要一种场景，为了最大化防止异常情况下误切换了场景，可以把四个场景配置为相同参数。

2. 执行 OP_CNT_CLR 将计数器清零。

   新的一轮计数将按照新的场景参数运行。

关于 OP 指令的详细说明，可查看操作指令。

11.2.3.4. 操作指令

OP 寄存器是写失效下唯一可以进行写操作的寄存器， 通过连续两个写操作数与序列匹配，启动对应的指令。

现有支持指令如下表所示：

表 11–2 OP 指令定义

指令操作	指令名称	写操作数序列	备注
计时清零	OP_CNT_CLR	0xA1C55555 → 0xA1CAAAAA	需要满足 CNT > CLR_THD，操作才生效。
切换到配置 0	OP_SW_CFG0	0xA1C5A5A0 → 0xA1CA5A50	–
切换到配置 1	OP_SW_CFG1	0xA1C5A5A1 → 0xA1CA5A51	–
切换到配置 2	OP_SW_CFG2	0xA1C5A5A2 → 0xA1CA5A52	–
切换到配置 3	OP_SW_CFG3	0xA1C5A5A3 → 0xA1CA5A53	–
写使能	OP_WR_EN	0xA1C99999 → 0xA1C66666	在 REG_WR_DIS = 3 下无效。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

11.2.4. 寄存器列表

表 11–3 WDOG 寄存器列表

偏移地址	缩写	寄存器名称	章节
0x000	WDOG_CTL	看门狗控制	0x000 WDOG_CTL
0x004	WDOG_CNT	看门狗计时器	0x004 WDOG_CNT
0x008	IRQ_EN	中断使能	0x008 IRQ_EN
0x00C	IRQ_STA	中断状态	0x00C IRQ_STA
0x040	CLR_THDn	清除阈值	0x040+n*0x10(n=0~3) CLR_THDn
0x044	IRQ_THDn	中断阈值	0x044+n*0x10(n=0~3) IRQ_THDn
0x048	RST_THDn	复位阈值	0x048+n*0x10(n=0~3) RST_THDn
0x0E8	WDOG_OP	看门狗指令	0x0E8 WDOG_OP
0xFFC	WDOG_VER	看门狗版本	0xFFC WDOG_VER

11.2.5. 寄存器描述

11.2.5.1. 0x000 WDOG_CTL

默认值：0x00000000			看门狗控制 (Watch Dog Control)
位域	类型	默认值	描述
31:30	–	–	–
29:28	R/W	0x0	REG_WR_DIS 寄存器写失效 • 0：写允许 • 1：写保护。可通过 OP_WR_EN 回到允许写。除了 OP 寄存器，其它寄存器都不可写，可执行 OP 写使能将该寄存器清零。 • 2：写保护。可通过 OP_WR_EN 回到允许写。除了 OP 寄存器，其它寄存器都不可写，可执行 OP 写使能将该寄存器清零。 • 3：写失效，除了 OP 寄存器，其它寄存器都不可写。只能通过复位将寄存器清零，回到允许写。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			看门狗控制 (Watch Dog Control)
位域	类型	默认值	描述
27:24	RO	0x0	CFG_ID 配置场景 ID **配置当前使用的场景，复位后使用配置 0**
23:2	–	–	–
1	R/W	0x0	DBG_CNT_CONTINUE CPU 在调试模式下计时继续 • 0：计时暂停 • 1：计时继续 **调试信号来源于处理器，如连接仿真器后进入调试模式。**
0	R/W	0x0	CNT_EN 计时使能 • 0：计时暂停 • 1：计时启动

11.2.5.2. 0x004 WDOG_CNT

默认值：0x00000000			看门狗计时器 (Watch Dog Counter)
位域	类型	默认值	描述
31:0	R	0x0	WDOG_CNT 当前计时器值，单位为一个 32K 时钟周期 **注：** • CNT 每个计时时钟周期更新一次。 • 执行 OP_CLR_CNT 后需要等待一个 32K 时钟周期，才能保证该值被清零。

11.2.5.3. 0x008 IRQ_EN

默认值：0x00000000			中断使能 (Interrupt Request Enable)
位域	类型	默认值	描述
31:1	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			中断使能 (Interrupt Request Enable)
位域	类型	默认值	描述
0	R/W	0	TIMEOUT_IRQ_EN 超时中断使能

11.2.5.4. 0x00C IRQ_STA

默认值：0x00000000			中断状态 (Interrupt Request Status)
位域	类型	默认值	描述
31:1	–	–	–
0	R/W1C	0	TIMEOUT_IRQ_STA 超时中断状态

11.2.5.5. 0x040+n*0x10(n=0~3) CLR_THDn

默认值：0x00000000			清除阈值 (Clear Threshold)
位域	类型	默认值	描述
31:0	R/W	0	CLR_THDn 当计时大于该值，计时清零操作才能生效。

11.2.5.6. 0x044+n*0x10(n=0~3) IRQ_THDn

默认值：0x00000000			中断阈值 (Interrupt Request Threshold)
位域	类型	默认值	描述
31:0	R/W	0x27100	IRQ_THDn 当计时大于该值，产生中断信号。

11.2.5.7. 0x048+n*0x10(n=0~3) RST_THDn

默认值：0x00000000			复位阈值 (Reset Threshold)
位域	类型	默认值	描述
31:0	R/W	0x4e200	RST_THDn

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			复位阈值 (Reset Threshold)
位域	类型	默认值	描述
			当计时大于该值，产生系统复位。

11.2.5.8. 0x0E8 WDOG_OP

默认值：0x00000000			看门狗指令 (Watch Dog Operator)
位域	类型	默认值	描述
31:0	R/W	0x0	WDOG_OP 指令寄存器，详细说明见操作指令。

11.2.5.9. 0xFFC WDOG_VER

默认值：0x00000100			看门狗版本（Watch Dog Version）
位域	类型	默认值	描述
31:0	RO	0x00000100	VERSION 模块版本，V1.0

11.3. Pulse Width Modulation (PWM)

Pulse Width Modulation (PWM)，脉宽调制模块。在最小 CPU 负载或交互下，PWM 信号发生器可自动生成复杂的脉冲宽度信号。自由编程的脉冲宽度信号可实现屏幕背光调整、电压调整、蜂鸣器驱动等场景。

PMW 中可以例化多个信号发生器，每个信号发生器用 PWMx 表示，其中字母 x 代表 PWM 信号发生器的序号。

一个 PWMx 信号发生器可输出一对信号，用 PWMx_A 和 PWMx_B 表示，其中字母 A 和 B 分别代表 PWMx 信号发生器的一对输出信号。例如，PWM0_A 和 PWM0_B 属于 PWM0 信号发生器的一对信号， PWM2_A 和 PWM2_B 属于 PWM2 信号发生器的一对信号。PWMx 信号发生器的一对输出信号可根据具体应用配置成相关或不相关，详情可查看特性说明。

11.3.1. 特性说明

一个 PWMx 信号发生器可以控制两个 PWM 输出，分别为 PWMx_A 和 PWMx_B。

PWMx 信号发生器支持以下功能：

- 专用的 16 位可配置周期和频率的时基计数器 (Time Base Counter)
- 两个 PWM 输出信号 PWMx_A 和 PWMx_B，能进行如下配置：
  - 单独输出 PWMx_A 或 PWMx_B，频率、占空比可调。
  - 同时输出 PWMx_A 和 PWMx_B，频率相同，占空比相同。
  - 同时输出 PWMx_A 和 PWMx_B，频率相同，占空比可不相同。

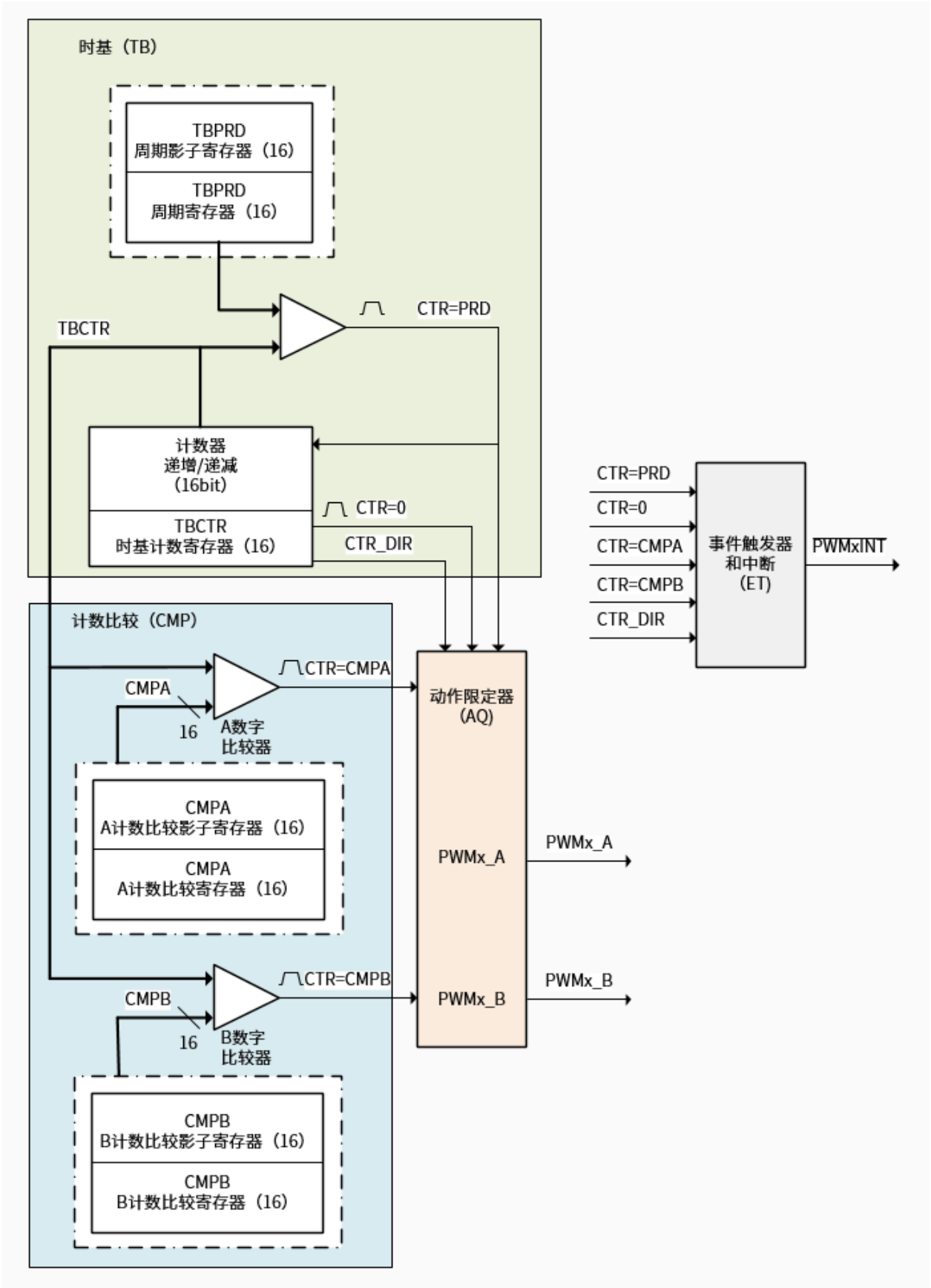
Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 软件可直接控制 PWM 输出信号
- 多种事件可产生 CPU 中断并产生相应的标志

ArtInChip

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

11.3.2. 原理框图



Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

11.3.3. 功能描述

每个 PWM 信号发生器包含以下子模块，每个子模块可以通过软件配置实现特定功能：

- 时基子模块
- 计数比较子模块
- 动作限定子模块
- 事件触发子模块

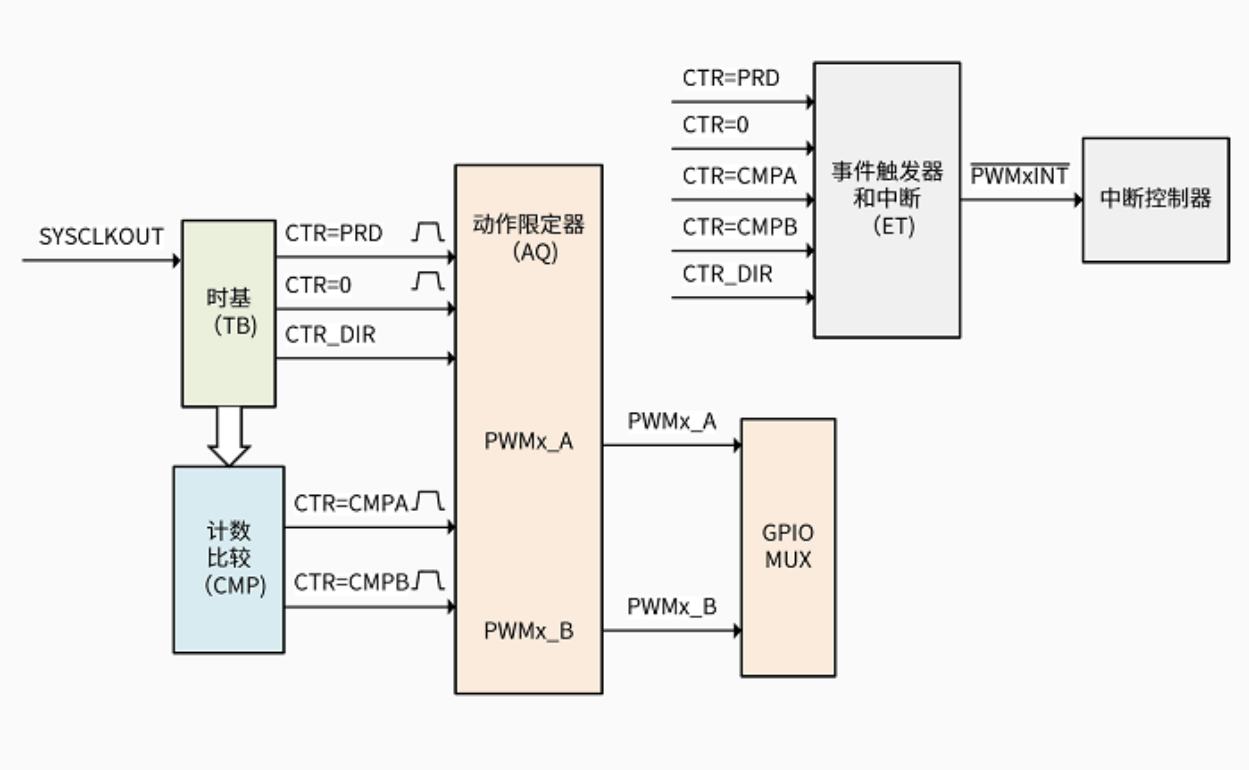


图 11–2 PWM 信号发生器框图

11.3.3.1. 时基子模块

每个 PWM 信号发生器都有一个独有的时基子模块，用以确定 PWM 信号发生器所有事件的时序。PWM 信号频率由时基周期寄存器 TBPRD 和时基计数器模式决定，图 11–3 : 时基子模块信号和寄存器关系如下所示：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

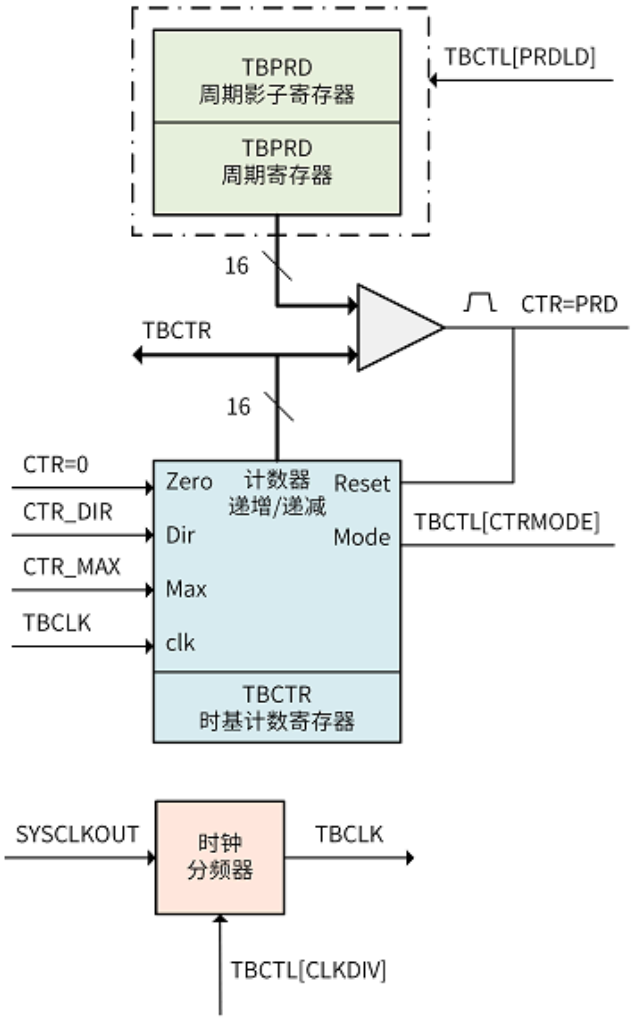


图 11–3 时基子模块信号和寄存器

图 11–4 : 不同时基计数器模式下 PWM 周期和频率的关系显示了当时基计数器周期设置为 4（即 TBPRD = 4），时基计数器在递增模式、递减模式、增减模式下 PWM 周期和频率的关系及计算方式。每一步增加时间由时基时钟 TBCLK 决定，时基时钟由系统时钟 SYSCLKOUT（SYSCLKOUT 由 PLL 配置给出，详见章节）分频得到。

时基计数器有以下三种操作模式，可通过时基控制寄存器 (TBCTL) 进行选择：

- 递增模式：时基计数器从零开始递增，直到达到时基周期寄存器 (TBPRD) 中的值时，时基计数器重置为零并开始再次递增。
- 递减模式：时基计数器从时基周期寄存器 (TBPRD) 中的值开始递减，直到达到零时，时基计数器重置为时基周期寄存器 (TBPRD) 中的值并开始再次递减。
- 增减模式：时基计数器从零开始递增，直到达到时基周期寄存器 (TBPRD) 中的值，当达到时基周期寄存器 (TBPRD) 中的值时，时基计数器开始递减，直到达到零时，时基计数器重复该模式并开始递增。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

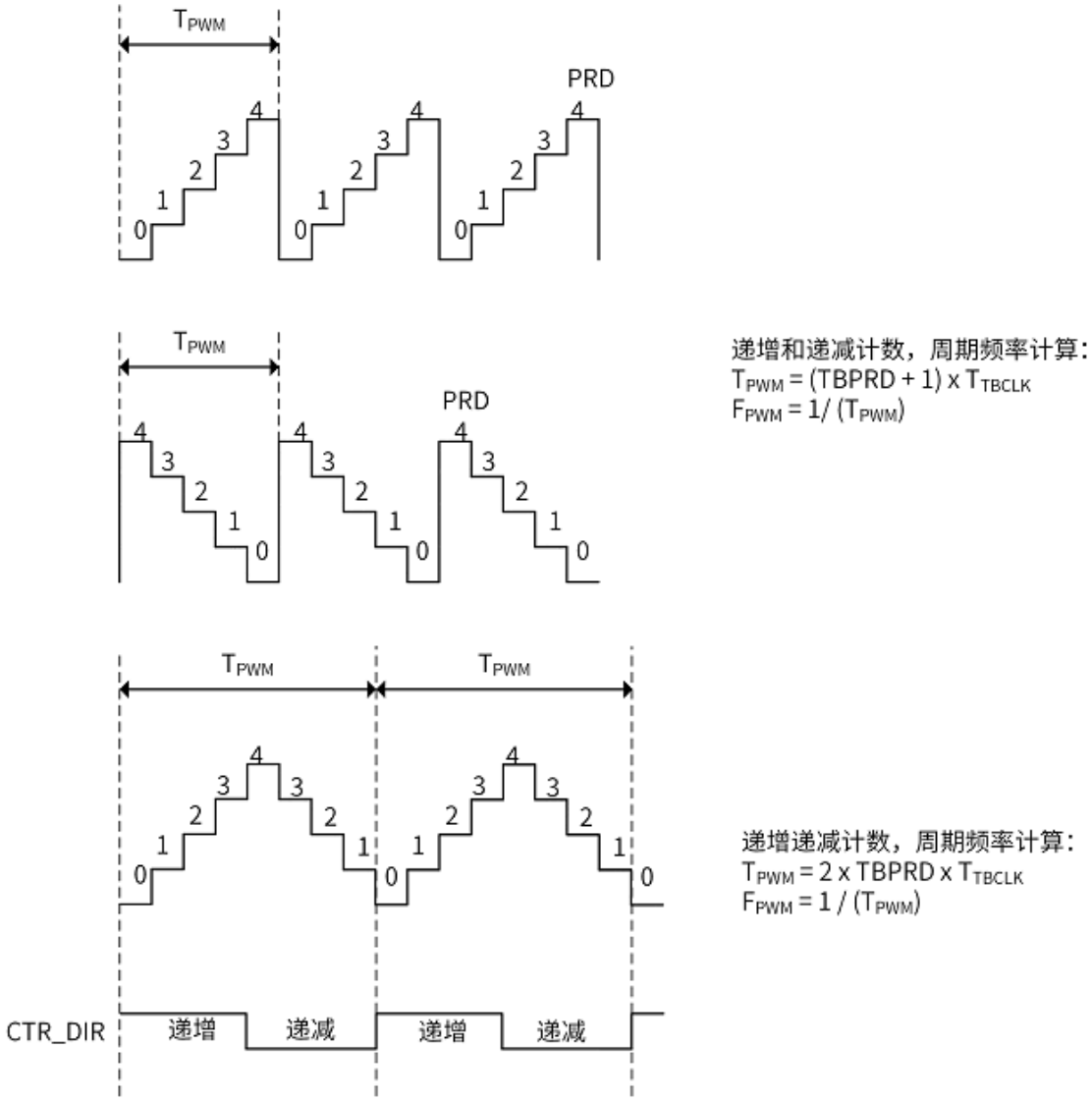


图 11–4 不同时基计数器模式下 PWM 周期和频率的关系

时基周期寄存器（TBPRD）拥有影子寄存器，影子寄存器提供了一种保持寄存器更新与硬件同步的方法。

- 影子模式：使用时基周期影子寄存器时 (PWM_TBCTL[PRDLD] =0)，只有当时基计数器 (TBCTR) 等于 0 时，时基周期寄存器的值（TBPRD）才会更新，读写时基周期寄存器实际上都是操作它的影子寄存器，这样可以防止由于软件异步修改寄存器从而导致错误操作。
- 立即模式：不使用时基周期影子寄存器时 (PWM_TBCTL[PRDLD] =1)，读写时基周期寄存器（TBPRD），就是立即对 TBPRD 进行操作，在此模式下，有可能出现在当前的周期，新写入的数值与计数器的数值错过比较时刻，造成当前周期输出的 PWM 占空比异常。

11.3.3.2. 计数比较子模块

图 11–5 : 计数比较子模块信号和寄存器显示了计数比较子模块的结构，计数比较子模块将时基计数器的值（TBCTR）作为输入，不断将时基计数器的值与两个计数比较器寄存器的值 (CMPA 和 CMPB) 作比较，生成两个独立的比较事件：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

1. CTR=CMPA：时基计数器的值等于计数比较器 A 寄存器的值（TBCTR=CMPA）。
2. CTR=CMPB：时基计数器的值等于计数比较器 B 寄存器的值（TBCTR=CMPB）。
   - 对于递增计数和递减计数模式每个事件在每个周期只发生一次
   - 对于增减计数模式如果比较值在零和 TBPRD 之间，则每个事件每个周期发生两次。如果比较值等于零或等于 TBPRD，则每个周期发生一次

这些事件被输入到动作限定子模块，根据动作限定子模块的配置转换为相应的动作。

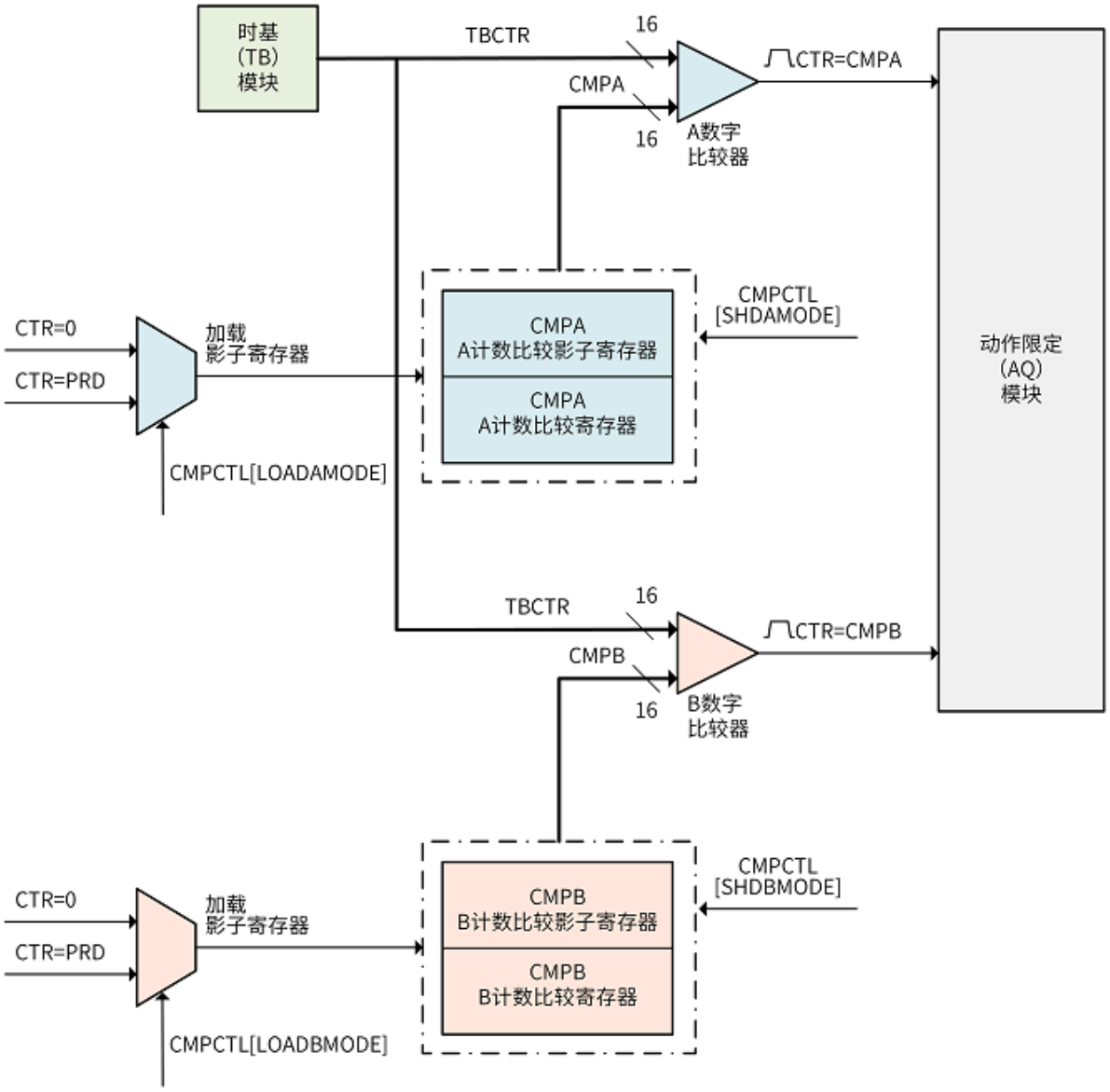


图 11–5 计数比较子模块信号和寄存器

计数比较器 A 寄存器和计数比较器 B 寄存器各自具有影子寄存器，通过 PWM_CMPCTL[SHDWAMODE] 和 PWM_CMPCTL[SHDWBMODE] 可分别配置计数比较器 A 寄存器和计数比较器 B 寄存器的操作模式为影子模式或立即模式。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

11.3.3.3. 动作限定子模块

动作限定子模块用于决定具体某个事件被用来转换成的具体动作类型，从而在 PWMx_A 和 PWMx_B 输出要求的波形。

图 11–6 : 动作限定子模块的输入和输出显示了动作限定子模块可以根据以下事件确定和生成动作：

- CTR=PRD：时基计数器的值等于时基计数器周期值（TBCTR=TBPRD）。
- CTR=0：时基计数器的值等于零（TBCTR=0）。
- CTR=CMPA：时基计数器的值等于计数比较器 A 寄存器的值（TBCTR=CMPA）。
- CTR=CMPB：时基计数器的值等于计数比较器 B 寄存器的值（TBCTR=CMPB）。

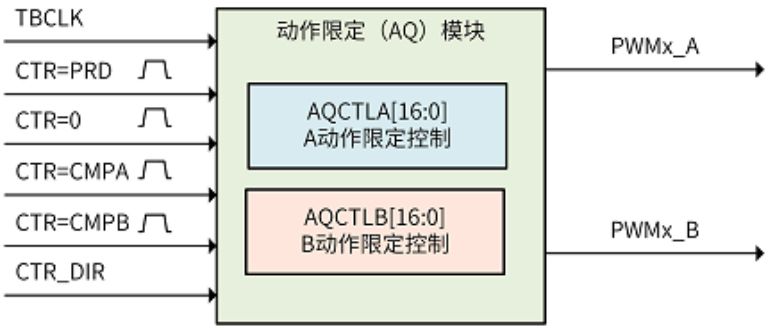


图 11–6 动作限定子模块的输入和输出

动作限定子模块根据输入的某一具体事件可以控制 PWMx_A 和 PWMx_B 生成如下动作：

- 无动作：将信号线 PWMx_A 或 PWMx_B 输出保持当前电平。
- 置低：强制信号线 PWMx_A 或 PWMx_B 输出低电平。
- 置高：强制信号线 PWMx_A 或 PWMx_B 输出高电平。
- 反转：反转 PWMx_A 或 PWMx_B 信号线电平。

图 11–7 : PWMx_A 和 PWMx_B 可能的动作限定器动作输出显示了不同输入事件和不同输出动作可能的组合，输出动作是为任一信号线（PWMx_A 或 PWMx_B）独立指定的，所以任何事件都可以分别控制 PWMx_A 和 PWMx_B 生成指定动作。例如：事件 CTR=CMPA 和 CTR=CMPB 都可以控制 PWMx_A 生成指定动作。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

时基计数器等于:				动作
0	CMPA	CMPB	PRD	
0 X	CA X	CB X	P X	无动作
Z ↓	CA ↓	CB ↓	P ↓	设置为低
Z ↑	CA ↑	CB ↑	P ↑	设置为高
Z T	CA T	CB T	P T	反转

图 11–7 PWMx_A 和 PWMx_B 可能的动作限定器动作输出

11.3.3.4. 事件触发子模块

事件触发子模块管理由时基子模块和计数比较子模块产生的事件，产生一个中断到 CPU。

图 11–8 : 事件触发子模块与中断控制器的互连如下：

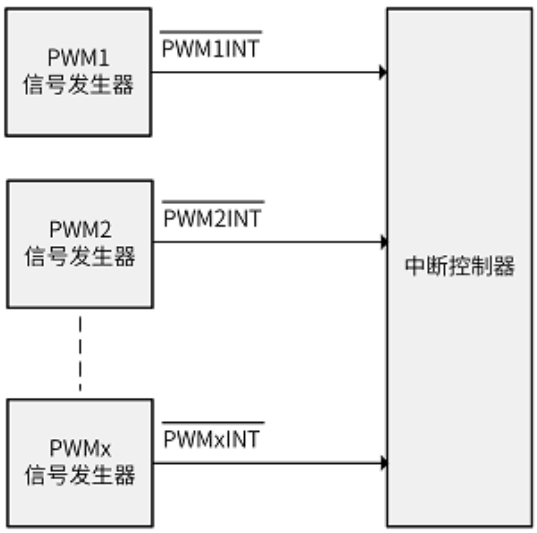


图 11–8 事件触发子模块与中断控制器的互连

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

11.3.4. 寄存器列表

表 11–4 寄存器列表

偏移地址	缩写	寄存器名称	章节
0x00	PWM_CTL	PWM 模块控制寄存器	0x000 PWM_CTL
0x04	PWM_MCTL	PWM 控制寄存器	0x004 PWM_MCTL
0x08	PWM_CKCTL	PWM 时钟控制寄存器	0x008 PWM_CKCTL
0x0C	PWM_INTCTL	PWM 中断控制寄存器	0x00C PWM_INTCTL
0x10	PWM_INTSTS	PWM 中断状态寄存器	0x010 PWM_INTSTS
0x300+0x100*n+0x00	PWM_TBCTL	PWM 时基控制寄存器	0X300+0X100*n+0x000 PWM_TBCTL
0x300+0x100*n+0x04	PWM_TBSTS	PWM 时基状态寄存器	0X300+0X100*n+0x004 PWM_TBSTS
0x300+0x100*n+0x10	PWM_TBCTR	PWM 时基计数器寄存器	0X300+0X100*n+0x010 PWM_TBCTR
0x300+0x100*n+0x14	PWM_TBPRD	PWM 时基周期寄存器	0X300+0X100*n+0x014 PWM_TBPRD
0x300+0x100*n+0x18	PWM_CMPCTL	PWM 比较器控制寄存器	0X300+0X100*n+0x018 PWM_CMPCTL
0x300+0x100*n+0x20	PWM_CMPA	PWM 比较器 A 寄存器	0X300+0X100*n+0x020 PWM_CMPA
0x300+0x100*n+0x24	PWM_CMPB	PWM 比较器 B 寄存器	0X300+0X100*n+0x024 PWM_CMPB
0x300+0x100*n+0x28	PWM_AQCTLA	PWM 动作器控制 A 寄存器	0X300+0X100*n+0x028 PWM_AQCTLA
0x300+0x100*n+0x2C	PWM_AQCTLB	PWM 动作器控制 B 寄存器	0X300+0X100*n+0x02C PWM_AQCTLB
0x300+0x100*n+0x44	PWM_ETSEL	PWM 事件触发选择寄存器	0X300+0X100*n+0x044 PWM_ETSEL
0X300+0X100*n+0x048 PWM_CNT_DIS	PWM_CNT_DIS	PWM 调试寄存器	0X300+0X100*n+0x048 PWM_CNT_DIS
0xFFC	PWM_VERSION	PWM 模块版本寄存器	0xFFC PWM_VERSION

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

11.3.5. 寄存器描述

11.3.5.1. 0x000 PWM_CTL

默认值：0x00000000			PWM 模块控制寄存器 (PWM_CTL)
位域	类型	默认值	描述
31:1	/	0x000_0000	/
0	R/W	0x0	EN PWM Module Enable，PWM 模块使能控制 • 0：PWM 模块未使能 EN=0 时，即使复位 PWM 模块，寄存器也不能访问。 • 1：PWM 模块使能 仅在 EN=1 时才能访问 PWM 模块寄存器且 PWM 模块开始正常工作。

11.3.5.2. 0x004 PWM_MCTL

默认值：0x00000000			PWM 控制寄存器 (PWM_MCTL)
位域	类型	默认值	描述
31:16	/	/	/
15:4	/	/	/
3	R/W	0x0	PWM3EN PWM3 Enable，PWM3 使能控制 • 0：PWM3 未使能，表示 PWM3EN 无效，复位 PWM3 信号发生器 ，同时复位 PWM3 相关的寄存器。 • 1：PWM3 使能，表示 PWM3EN 有效，PWM 信号发生器 PWM3 开始正常工作。 PWM 模块寄存器不受此位影响。
2	R/W	0x0	PWM2EN PWM2 Enable，PWM2 使能控制 • 0：PWM2 未使能，表示 PWM2EN 无效，复位 PWM2 信号发生器 ，同时复位 PWM2 相关的寄存器。 • 1：PWM2 使能，表示 PWM2EN 有效，PWM 信号发生器 PWM2 开始正常工作。 PWM 模块寄存器不受此位影响。
1	R/W	0x0	PWM1EN

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			PWM 控制寄存器 (PWM_MCTL)
位域	类型	默认值	描述
			PWM1 Enable，PWM1 使能控制 • 0：PWM1 未使能，表示 PWM1EN 无效，复位 PWM1 信号发生器，同时复位 PWM1 相关的寄存器。 • 1：PWM1 使能，表示 PWM1EN 有效，PWM 信号发生器 PWM1 开始正常工作。 PWM 模块寄存器不受此位影响。
0	R/W	0x0	PWM0EN PWM0 Enable，PWM0 使能控制 • 0：PWM0 未使能，表示 PWM0EN 无效，复位 PWM0 信号发生器，同时复位 PWM0 相关的寄存器。 • 1：PWM0 使能 PWM 模块寄存器不受此位影响。

11.3.5.3. 0x008 PWM_CKCTL

默认值：0x00000000			PWM 时钟控制寄存器 (PWM_CKCTL)
位域	类型	默认值	描述
31:16	/	/	/
15:4	/	/	/
3	R/W	0x0	PWM3CKON PWM3 Clock On，PWM3 工作时钟控制 • 0：PWM3 时钟关闭 • 1：PWM3 时钟打开 此位必须打开时，才能访问 PWM 信号发生器 PWM3 的寄存器。
2	R/W	0x0	PWM2CKON PWM2 Clock On，PWM2 工作时钟控制 • 0：PWM2 时钟关闭 • 1：PWM2 时钟打开 此位必须打开时，才能访问 PWM 信号发生器 PWM2 的寄存器。
1	R/W	0x0	PWM1CKON PWM1 Clock On，PWM1 工作时钟控制

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			PWM 时钟控制寄存器 (PWM_CKCTL)
位域	类型	默认值	描述
			• 0：PWM1 时钟关闭 • 1：PWM1 时钟打开 此位必须打开时，才能访问 PWM 信号发生器 PWM1 的寄存器。
0	R/W	0x0	PWM0CKON PWM0 Clock On，PWM0 工作时钟控制 • 0：PWM0 时钟关闭 • 1：PWM0 时钟打开 此位必须打开时，才能访问 PWM 信号发生器 PWM0 的寄存器。

11.3.5.4. 0x00C PWM_INTCTL

默认值：0x00000000			PWM 中断控制寄存器 (PWM_INTCTL)
位域	类型	默认值	描述
31:16	/	/	/
15:4	/	/	/
3	R/W	0x0	PWM3INTEN PWM3 Interrupt Enable，控制 PWM 信号发生器 PWM3 中断使能 • 0：PWM3 中断关闭。 当 PWM3 有中断产生时，PWM 中断状态寄存器相应位被置成 1，没有中断信号给 CPU 系统，但 CPU 系统可以通过查询状态位来处理 PWM3 事件。 • 1：PWM3 中断使能。 当 PWM3 有中断产生时，PWM 中断状态寄存器相应位被置成 1，中断信号传递给 CPU 系统，CPU 系统响应中断事件。
2	R/W	0x0	PWM2INTEN PWM2 Interrupt Enable，用于控制 PWM 信号发生器 PWM2 中断使能 • 0：PWM2 中断关闭，当 PWM2 有中断产生时，PWM 中断状态寄存器相应位被置成 1，没有中断信号给 CPU 系统，但 CPU 系统可以通过查询状态位来处理 PWM2 事件。 • 1：PWM2 中断使能，当 PWM2 有中断产生时，PWM 中断状态寄存器相应位被置成 1，中断信号传递给 CPU 系统，CPU 系统响应中断事件。
1	R/W	0x0	PWM1INTEN PWM1 Interrupt Enable，用于控制 PWM 信号发生器 PWM1 中断使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			PWM 中断控制寄存器 (PWM_INTCTL)
位域	类型	默认值	描述
			• 0：PWM1 中断关闭，当 PWM1 有中断产生时，PWM 中断状态寄存器相应位被置成 1，没有中断信号给 CPU 系统，但 CPU 系统可以通过查询状态位来处理 PWM1 事件。 • 1：PWM1 中断使能，当 PWM1 有中断产生时，PWM 中断状态寄存器相应位被置成 1，中断信号传递给 CPU 系统，CPU 系统响应中断事件。
0	R/W	0x0	PWM0INTEN PWM0 Interrupt Enable，用于控制 PWM 信号发生器 PWM0 中断使能 • 0：PWM0 中断关闭，当 PWM0 有中断产生时，PWM 中断状态寄存器相应位被置成 1，没有中断信号给 CPU 系统，但 CPU 系统可以通过查询状态位来处理 PWM0 事件。 • 1：PWM0 中断使能，当 PWM1 有中断产生时，PWM 中断状态寄存器相应位被置成 1，中断信号传递给 CPU 系统，CPU 系统响应中断事件。

11.3.5.5. 0x010 PWM_INTSTS

默认值：0x00000000			PWM 中断状态寄存器 (PWM_INTSTS)
位域	类型	默认值	描述
31:16	/	/	/
15:4	/	/	/
3	R/W1C	0x0	PWM3INTSTS PWM3 Interrupt Status，PWM3 中断状态，用于标志 PWM 信号发生器 PWM3 是否有中断事件： • 0：PWM3 无中断事件 • 1：PWM3 有中断事件，CPU 系统处理此中断事件，处理完成后写 1 清零。
2	R/W1C	0x0	PWM2INTSTS PWM2 Interrupt Status，PWM2 中断状态，用于标志 PWM 信号发生器 PWM2 是否有中断事件： • 0：PWM2 无中断事件 • 1：PWM2 有中断事件，CPU 系统处理此中断事件，处理完成后写 1 清零。
1	R/W1C	0x0	PWM1INTSTS PWM1 Interrupt Status，PWM1 中断状态，用于标志 PWM 信号发生器 PWM1 是否有中断事件：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			PWM 中断状态寄存器 (PWM_INTSTS)
位域	类型	默认值	描述
			• 0：PWM1 无中断事件 • 1：PWM1 有中断事件，CPU 系统处理此中断事件，处理完成后写 1 清零。
0	R/W1C	0x0	PWM0INTSTS PWM0 Interrupt Status，PWM0 中断状态，用于标志 PWM 信号发生器 PWM0 是否有中断事件： • 0：PWM2 无中断事件 • 1：PWM2 有中断事件，CPU 系统处理此中断事件，处理完成后写 1 清零。

11.3.5.6. 0X300+0X100*n+0x000 PWM_TBCTL

默认值：0x00000000			PWM 时基控制寄存器 (PWM_TBCTL)
位域	类型	默认值	描述
31:28	/	/	/
27:16	R/W	0x1	CLKDIV Time-Base Clock Divider，时基时钟除数 时基时钟 TBCLK = SYSCLKOUT / (CLKDIV + 1)
15:4	/	/	/
3	R/W	0x0	PRDLD Period Register Load From Shadow Register Select，时基周期寄存器 (PRD) 操作方式 • 0：影子模式 时基计数器 (TBCTR) 等于 0 时，时基周期寄存器 (TBPRD) 的数值从它的影子寄存器中装载。读写时基寄存器 (TBPRD) 实际上都是操作它的影子寄存器。 • 1：立即模式 时基周期寄存器 (TBPRD) 的数值不需要通过它的影子寄存器进行装载。读写时基周期寄存器 (TBPRD)，就是立即对 TBPRD 进行操作。
2	/	/	/
1:0	R/W	0x0	CTRMODE Counter Mode，时基计数器模式 • 0：递增模式 • 1：递减模式

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			PWM 时基控制寄存器 (PWM_TBCTL)
位域	类型	默认值	描述
			• 2：增减模式 • 3：/ 此位用于配置时基计数器的计数模式，一般情况下计数模式只能配置一次，当计数器开始工作时，不可改动。

11.3.5.7. 0X300+0X100*n+0x004 PWM_TBSTS

默认值：0x00000000			PWM 时基状态寄存器 (PWM_TBSTS)
位域	类型	默认值	描述
31:3	/	/	/
2	R/W1C	0x0	CTRMAX Time-Base Counter Max Latched Status，时基计数器达到最大值 0xFFFF 状态位 • 0：时基计数器没有达到最大值。 • 1：时基计数器达到最大值。 此位写 0 无任何效果，写 1 清除此状态。
1	R/W1C	0x0	/
0	R	0x0	CTRDIR Time-Base Counter Direction Status，时基计数器方向状态位 • 0：计数器正在向下计数 • 1：计数器正在向上计数 此位在时基计数器的计数模式被设置且开始计数时才有意义。

11.3.5.8. 0X300+0X100*n+0x010 PWM_TBCTR

默认值：0x00000000			PWM 时基计数器寄存器 (PWM_TBCTR)
位域	类型	默认值	描述
31:16	/	/	/
15:0	R/W	0x0	TBCTR Time-Base Counter Value，时基计数器当前数值

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			PWM 时基计数器寄存器 (PWM_TBCTR)
位域	类型	默认值	描述
			• 读取此位时，数值表明时基计数器的当前数值。 • 写此位时，数值直接写到时基计数器寄存器，该寄存器无影子寄存器且写时没有经过 TBCLK 同步。

11.3.5.9. 0X300+0X100*n+0x014 PWM_TBPRD

默认值：0x00000000			PWM 时基周期寄存器 (PWM_TBPRD)
位域	类型	默认值	描述
31:16	/	/	/
15:0	R/W	0x0	TBPRD The Period of Time-Base Counter，时基计数器周期 此位决定时基计数器的周期，用于设定 PWM 的频率。此寄存器有影子寄存器，受 PWM_TBCTL 寄存器中的 PRDLD 位控制，缺省时使用影子寄存器。

11.3.5.10. 0X300+0X100*n+0x018 PWM_CMPCTL

默认值：0x00000000			PWM 比较器控制寄存器 (PWM_CMPCTL)
位域	类型	默认值	描述
31:7	/	/	/
6	R/W	0x0	SHDWBMODE CMPB Operating Mode，比较寄存器 CMPB 的操作模式 • 0：影子模式 • 1：立即模式
5	/	/	/
4	R/W	0x0	SHDWAMODE CMPA Operating Mode，比较寄存器 CMPA 的操作模式 • 0：影子模式 • 1：立即模式
3:2	R/W	0x0	LOADBMODE CMPB Load From Shadow Select Mode，CMPB 寄存器从影子寄存器装载数值模式

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			PWM 比较器控制寄存器 (PWM_CMPCTL)
位域	类型	默认值	描述
			• 0：当计数器 CTR＝0 时 • 1：当计数器 CTR＝PRD 时 • 2：当计数器 CTR＝0 时或 CTR＝PRD • 3：冻结 当 SHDWBMODE 位为 1 时，即 CMPB 寄存器不使用影子寄存器时，此位无效。
1:0	R/W	0x0	LOADAMODE CMPA Load From Shadow Select Mode，CMPA 寄存器从影子寄存器装载数值模式 • 0：当计数器 CTR＝0 时 • 1：当计数器 CTR＝PRD 时 • 2：当计数器 CTR＝0 时或 CTR＝PRD • 3：冻结 当 SHDWAMODE 位为 1 时，即 CMPA 寄存器不使用影子寄存器时，此位无效。

11.3.5.11. 0X300+0X100*n+0x020 PWM_CMPA

默认值：0x00000000			PWM 比较器 A 寄存器 (PWM_CMPA)
位域	类型	默认值	描述
31:16	/	/	/
15:0	R/W	0x0	CMPA 比较寄存器 A 的数值 计数器 TBCTR 与 CMPA 进行比，相等时比较器产生一个脉冲给后续的动作器 AQ。

11.3.5.12. 0X300+0X100*n+0x024 PWM_CMPB

默认值：0x00000000			PWM 比较器 B 寄存器 (PWM_CMPB)
位域	类型	默认值	描述
31:16	/	/	/
15:0	R/W	0x0	CMPB 比较寄存器 B 的数值 计数器 TBCTR 与 CMPB 进行比较，相等时比较器产生一个脉冲给后续的动作器 AQ。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

11.3.5.13. 0X300+0X100*n+0x028 PWM_AQCTLA

默认值：0x00000000			PWM 动作器控制 A 寄存器 (PWM_AQCTLA)
位域	类型	默认值	描述
31:17	/	0x0	/
16	R/W	0x0	PWMA_INIT PWMA Initial Value，PWMA 初始电平 • 0：初始化低电平 • 1：初始化高电平
15：12	/	/	/
11:10	R/W	0x0	CBD TBCTR Equal to CMPB and Counter-Down，当时基计数器 TBCTR 等于阈值 CMPB 且计数器正递减时动作方式 • 0：无动作 • 1：置低-强制信号线 PWMxA 输出低电平 • 2：置高-强制信号线 PWMxA 输出高电平 • 3：反转-反转 PWMxA 信号线 **注：** 在增减计数模式下，当计数器等于零时，定义计数方向为 1（递增方向）。
9:8	R/W	0x0	CBU TBCTR Equal to CMPB and Counter-Up，当时基计数器 TBCTR 等于阈值 CMPB 且计数器正递增时动作方式 • 0: 无动作 • 1：置低-强制信号线 PWMxA 输出低电平 • 2：置高-强制信号线 PWMxA 输出高电平 • 3：反转-反转 PWMxA 信号线 **注：** 在增减计数模式下，当计数器等于零时，定义计数方向为 1（递增方向）。
7:6	R/W	0x0	CAD TBCTR Equal to CMPA and Counter-Down，当时基计数器 TBCTR 等于阈值 CMPA 且计数器正递减时动作方式 • 0：无动作 • 1：置低-强制信号线 PWMxA 输出低电平

Copyright © 2023-2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			PWM 动作器控制 A 寄存器 (PWM_AQCTLA)
位域	类型	默认值	描述
			• 2：置高–强制信号线 PWMxA 输出高电平 • 3：反转–反转 PWMxA 信号线 **注：** 在增减计数模式下，当计数器等于零时，定义计数方向为 1（递增方向）。
5:4	R/W	0x0	CAU TBCTR Equal to CMPA and Counter–Up，当时基计数器 TBCTR 等于阈值 CMPA 且计数器正递增时动作方式 • 0：无动作 • 1：置低–强制信号线 PWMxA 输出低电平 • 2：置高–强制信号线 PWMxA 输出高电平 • 3：反转–反转 PWMxA 信号线 **注：** 在增减计数模式下，当计数器等于零时，定义计数方向为 1（递增方向）。
3:2	R/W	0x0	PRD TBCTR Equal to PRD，当时基计数器 TBCTR 等于周期 PRD 时动作方式 • 0：无动作 • 1：置低–强制信号线 PWMxA 输出低电平 • 2：置高–强制信号线 PWMxA 输出高电平 • 3：反转–反转 PWMxA 信号线 **注：** 在增减计数模式下，当计数器等于零时，定义计数方向为 1（递增方向）。
1:0	R/W	0x0	ZRO TBCTR Equal to Zero，当时基计数器 TBCTR 等于 0 时动作方式 • 0：无动作 • 1：置低–强制信号线 PWMxA 输出低电平 • 2：置高–强制信号线 PWMxA 输出高电平 • 3：反转–反转 PWMxA 信号线 **注：** 在增减计数模式下，当计数器等于零时，定义计数方向为 1（递增方向）。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

11.3.5.14. 0X300+0X100*n+0x02C PWM_AQCTLB

默认值：0x00000000			PWM 动作器控制 B 寄存器（PWM_AQCTLB)
位域	类型	默认值	描述
31:17	/	0x0	/
16	R/W	0x0	PWMB_INIT PWMBInitial Value，PWMB 初始电平 • 0：初始化低电平 • 1：初始化高电平
15：12	/	/	/
11:10	R/W	0x0	CBD TBCTR Equal to CMPB and Counter-Down，当时基计数器 TBCTR 等于阈值 CMPB 且计数器正递减时动作方式 • 0：无动作 • 1：置低-强制信号线 PWMxB 输出低电平 • 2：置高-强制信号线 PWMxB 输出高电平 • 3：反转-反转 PWMxB 信号线 **注：** 在增减计数模式下，当计数器等于零时，定义计数方向为 1（递增方向）。
9:8	R/W	0x0	CBU TBCTR Equal to CMPB and Counter-Up，当时基计数器 TBCTR 等于阈值 CMPB 且计数器正递增时动作方式 • 0：无动作 • 1：置低-强制信号线 PWMxB 输出低电平 • 2：置高-强制信号线 PWMxB 输出高电平 • 3：反转-反转 PWMxB 信号线 **注：** 在增减计数模式下，当计数器等于零时，定义计数方向为 1（递增方向）。
7:6	R/W	0x0	CAD TBCTR Equal to CMPA and Counter-Down，当时基计数器 TBCTR 等于阈值 CMPA 且计数器正递减时动作方式 • 0：无动作 • 1：置低-强制信号线 PWMxB 输出低电平

Copyright © 2023-2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			PWM 动作器控制 B 寄存器 （PWM_AQCTLB)
位域	类型	默认值	描述
			• 2：置高–强制信号线 PWMxB 输出高电平 • 3：反转–反转 PWMxB 信号线 **注：** 在增减计数模式下，当计数器等于零时，定义计数方向为 1（递增方向）。
5:4	R/W	0x0	CAU TBCTR Equal to CMPA and Counter–Up，当时基计数器 TBCTR 等于阈值 CMPA 且计数器正递增时动作方式 • 0：无动作 • 1：置低–强制信号线 PWMxB 输出低电平 • 2：置高–强制信号线 PWMxB 输出高电平 • 3：反转–反转 PWMxB 信号线 **注：** 在增减计数模式下，当计数器等于零时，定义计数方向为 1（递增方向）。
3:2	R/W	0x0	PRD TBCTR Equal to PRD，当时基计数器 TBCTR 等于周期 PRD 时动作方式 • 0：无动作 • 1：置低–强制信号线 PWMxB 输出低电平 • 2：置高–强制信号线 PWMxB 输出高电平 • 3：反转–反转 PWMxB 信号线 **注：** 在增减计数模式下，当计数器等于零时，定义计数方向为 1（递增方向）。
1:0	R/W	0x0	ZRO TBCTR Equal to Zero，当时基计数器 TBCTR 等于 0 时动作方式 • 0：无动作 • 1：置低–强制信号线 PWMxB 输出低电平 • 2：置高–强制信号线 PWMxB 输出高电平 • 3：反转–反转 PWMxB 信号线 **注：** 在增减计数模式下，当计数器等于零时，定义计数方向为 1（递增方向）。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

11.3.5.15. 0X300+0X100*n+0x044 PWM_ETSEL

默认值：0x00000000			PWM 事件触发选择寄存器 (PWM_ETSEL)
位域	类型	默认值	描述
31:4	/	/	/
3	R/W	0x0	INTEN PWMx Interrupt Enable，PWMx 中断使能 • 0：禁止 PWMx 中断 • 1：使能 PWMx 中断 此位用来控制 PWMx 中断产生。
2:0	R/W	0x0	INTSEL PWMx Interrupt Select，PWMx 中断事件选择 • 0：/ • 1：TBCTR = 0 时触发事件 • 2：TBCTR = PRD 时触发事件 • 3：/ • 4：TBCTR = CMPA 且计数器正在递增时触发事件 • 5：TBCTR = CMPA 且计数器正在递减时触发事件 • 6：TBCTR = CMPB 且计数器正在递增时触发事件 • 7：TBCTR = CMPB 且计数器正在递减时触发事件

11.3.5.16. 0X300+0X100*n+0x048 PWM_CNT_DIS

默认值：0x00000000			PWM 调试寄存器 (PWM_CNT_DIS)
位域	类型	默认值	描述
31:1	/	/	/
0	R	0	PWM_CNT_CK_DIS 0：没有触发事件产生配置无动作时，时基计数器 disable，无比较事件发生。CMPA 与 CMPB 可配置相同值，也可不配置。默认为 0，比较动作以配置动作为较高优先级。 1：配置无动作时，时基计数器 enable，有比较事件发生。由于优先级原因，CMPA 与 CMPB 不能配置相等，或者 CMPA、CMPB 不能配置为 0。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

11.3.5.17. 0xFFC PWM_VERSION

默认值：0x00000100			PWM 模块版本寄存器 (PWM_VERSION)
位域	类型	默认值	描述
31:0	RO	0x00000100	Version（版本号），采用 BCD 码显示，V1.0

ArtInChip

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12. 接口

12.1. Quad Serial Peripheral Interface (QSPI)

Serial Peripheral Interface (SPI)，即串行外围设备接口，是一种高速、全双工、同步的串行时钟接口。SPI 可允许微控制器 (MCU) 与各种外围设备之间以串行方式和特定的传输速率进行高效的信息交换。

Quad Serial Peripheral Interface (QSPI) 作为 SPI 的扩展版本，提供了更高速的数据传输能力。标准 SPI 最大支持两线数据传输，而 QSPI 最大支持四线数据传输。与此同时，QSPI 控制器配备了一个 64 字节的接收缓存 (RX FIFO) 和一个 64 字节的发送缓存 (TX FIFO)，并通过 AHB 总线进行高效的数据访问，适用于需要快速数据处理和传输的典型场景。通过中断或 DMA 方式，QSPI 控制器可以实现快速的数据传输，无需 CPU 过多干预，来增强系统的运行效率。

SoC 集成多套 QSPI 控制器，常用的使用场景如下：

- 快速读写访问串行 Flash 设备：

  QSPI0/ 1 为 QSPI 接口，用于访问 SPI NOR/ NAND Flash 等。

12.1.1. 功能特性

- 全双工同步串行接口
- 仅支持主机模式
- 具有 64 字节的 FIFO 缓存用于发送和接收数据
- 片选信号和时钟的相位和极性可配
- 支持中断模式或 DMA 模式
- 支持以下传输模式：
  - mode0
  - mode1
  - mode2
  - mode3
- 支持可编程串行数据流长度配置：从 1 bit 到 32 bits
- 支持非标准 SPI 接口：MOSI 和 MISO 引脚复用
- 支持标准 SPI 接口，双路输出/双路输入 SPI 接口，双路 I/O SPI 接口，四路输出/四路输入 SPI 接口，QIO 和 QPI 接口
- QSPI1 支持使用控制器内部 DMA。
- QSPI0 和 QSPI1 支持引脚重映射，支持 12 种线序调整方式。
- QSPI0 支持 XiP 模式，通过 AXI 直接读取 SPI NOR Flash

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.1.2. 原理框图

图 12–1 QSPI 原理框图

12.1.3. 功能描述

主机模式

主机模式下，控制器支持的模式及其详细描述如下：

- 对于不同外设，可以选择相应的传输模式和总线模式：
  - 传输模式用于选择时钟信号的有效电平和采样相位。
  - 总线模式用于选择数据传输的有效位宽。
- 连续传输模式：通过寄存器配置突发传输的字节数，提高总线的传输效率。
- 多种不同的采样模式：满足不同传输路径引入的数据延时要求。
- 非字节对齐的位模式传输：可以 1~ 32 位任意配置，满足不同外设的传输协议要求。
- XIP 模式：对于 SPI NOR Flash，通过 AXI 总线直接访问 SPI NOR Flash 映射的物理地址，实现片上执行代码。

如果发生了传输错误，控制器的状态寄存器相应的错误状态标志会置位， 并产生中断通知给 CPU 进行错误处理，保证数据传输的正确性。

12.1.3.1. 传输模式

QSPI 控制器支持多种不同的传输模式，外设使用的具体模式取决于其功能需求和性能特点。通过正确配置 SPI_CLK 信号的时钟极性 (SPI_TCFG.CPOL) 和时钟相位 (SPI_TCFG.CPHA)，可以实现不同的传输模式，从而确保数据传输的正确性。QSPI 控制器根据 SPI_CLK 信号从移位寄存器中读取或写入数据，在 SPI_CLK 时钟的上升沿或下降沿被采样。

CPOL 定义了 SPI_CLK 为空闲状态时的信号极性：

- 当 CPOL 为 1，SPI_CLK 为高电平。
- 当 CPOL 为 0，SPI_CLK 为低电平。

CPHA 的状态决定了 SPI_CLK 在时钟的奇数边沿是采样数据还是移位数据：

- 当 CPHA 为 1，在时钟的奇数边沿移位数据。
- 当 CPHA 为 0，在时钟的奇数边沿采样数据。

下表总结了几种不同的传输模式对应的极性、相位以及采样描述：

表 12–1 传输模式

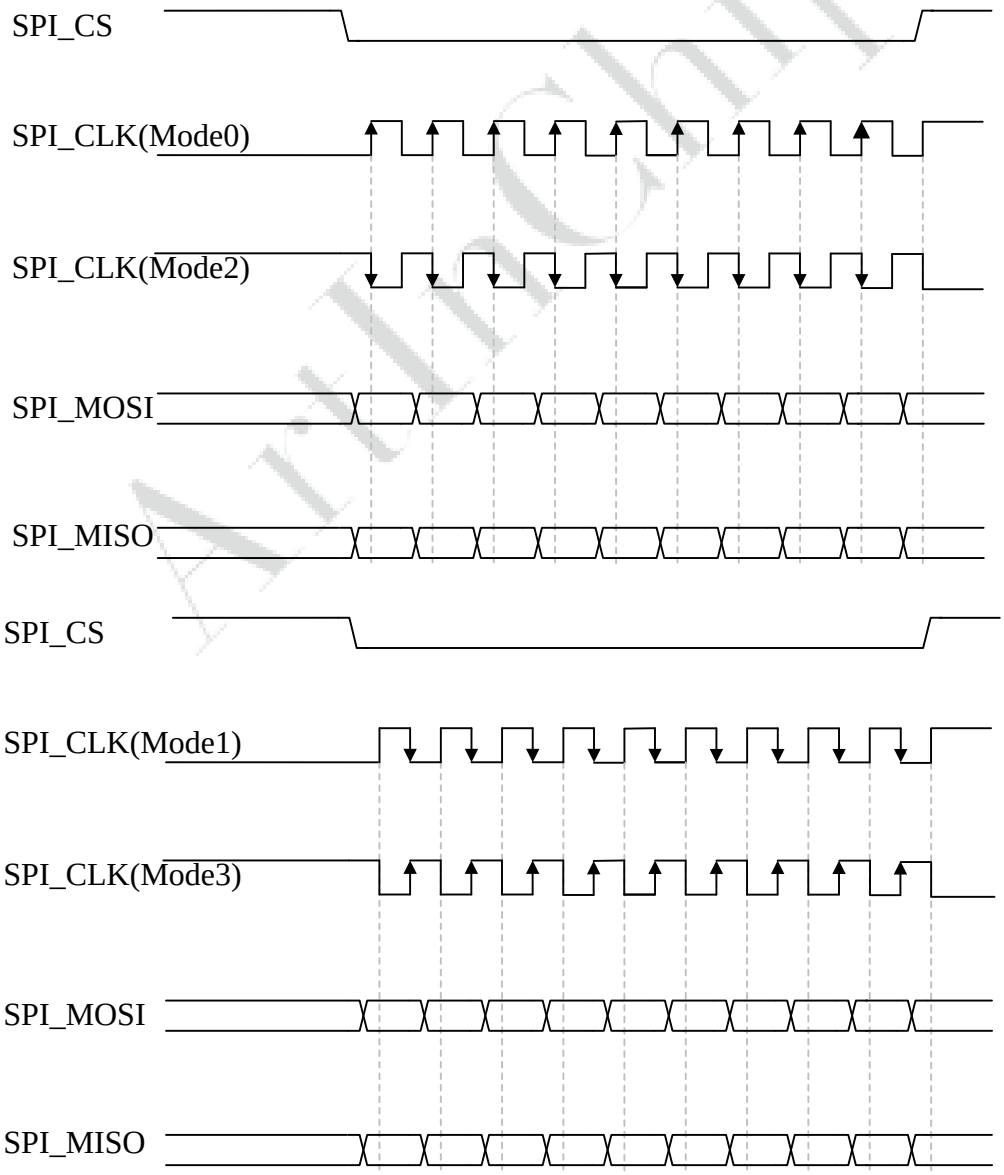
SPI 传输模式	CPOL（极性）	CPHA（相位）	时钟奇数边沿	时钟偶数边沿
0	0	0	上升沿采样	下降沿移位
1	0	1	上升沿移位	下降沿采样

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 12–1 传输模式 (续)

SPI 传输模式	CPOL（极性）	CPHA（相位）	时钟奇数边沿	时钟偶数边沿
2	1	0	下降沿采样	上升沿移位
3	1	1	下降沿移位	上升沿采样

传输模式的流程图如下：



12.1.3.2. 工作模式

QSPI 控制器默认工作在主机模式，不支持从机模式。

- 主机模式下：
  1. 控制器产生 SPI_CLK 时钟发送给外部设备
  2. 数据从 TX FIFO 传送到 MOSI 引脚，MISO 引脚接收从机发来的数据传送给 RX FIFO。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

3. 在数据传输过程中，片选信号必须为低电平。

   片选信号可以通过寄存器配置为自动控制还是软件控制：

   - 当 SPI_TCFG.CS_CTL_SEL 字段为 0，控制器自动控制片选信号。
   - 当 SPI_TCFG.CS_CTL_SEL 字段为 1，则用软件手动控制片选信号。此时配置 SPI_TCFG.CS_LEVEL 决定片选信号的状态。

12.1.3.3. 总线模式

总线模式包含以下类型：

- SPI 标准模式
- SPI 三线模式
- SPI 双路输入/ 输出模式
- SPI 双路 I/O 模式
- SPI 四路输入 /输出模式
- SPI 四路 I/O 模式
- SPI QPI 模式
- SPI 突发传输配置

注：

除了 SPI 标准模式和三线模式，在其它模式下，必须配置 SPI_TCFG.TXDLY_EN 位为 1，保证单线切换多线时命令或地址被正确采样。

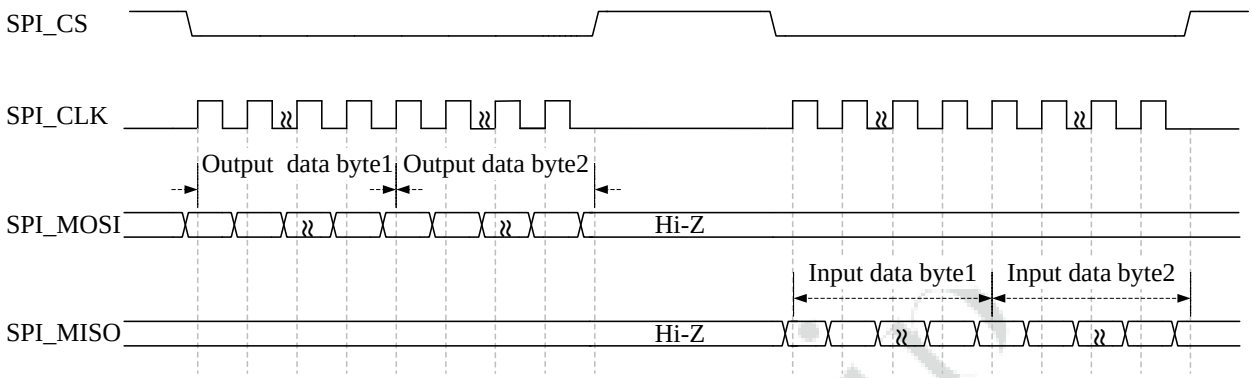
12.1.3.3.1. SPI 标准模式

SPI 标准模式，是 SPI 默认的标准总线模式，使用以下信号线实现通信：

- SPI_CS：片选控制信号，在 SPI_CS 为低电平期间进行数据传输。
- SPI_CLK：时钟线。
- SPI_MOSI：作为主机时发送数据，作为从机时接收数据。
- SPI_MISO：作为主机时接收数据，作为从机时发送数据。

标准模式时序图如下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



注：

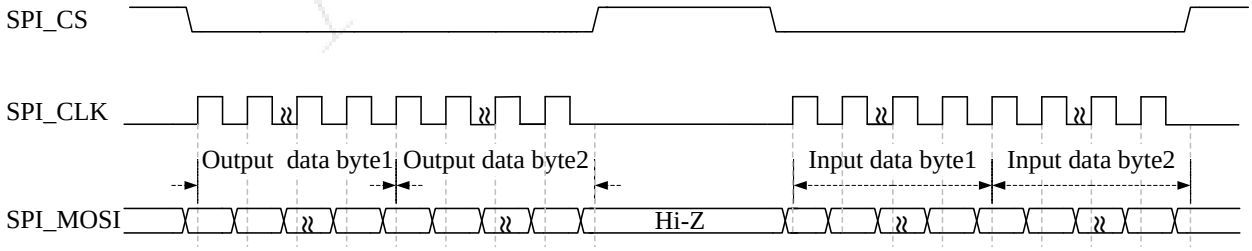
SPI 标准模式支持全双工，在发送数据的同时可以接收数据。

12.1.3.3.2. SPI 三线模式

SPI 三线模式是标准模式的变体，仅当 SPI 工作在主机模式时有效。SPI 三线模式使用以下信号线实现通信：

- SPI_CS：片选控制信号，在 SPI_CS 为低电平期间进行数据传输。
- SPI_CLK：时钟线。
- SPI_MOSI：数据的输入和输出的数据信号线 。

三线模式时序图如下：



通过配置寄存器 0x040 SPI_BMTC.BMOD_SEL 字段为 `0x2`，使用三线数据传输模式。

注：

SPI 三线模式默认是以位为单位的模式传输，通过配置 0x008 SPI_TCFG.3WIRE_EN，可以支持以字节为单位的模式传输。

12.1.3.3.3. SPI 双路输入/ 输出模式

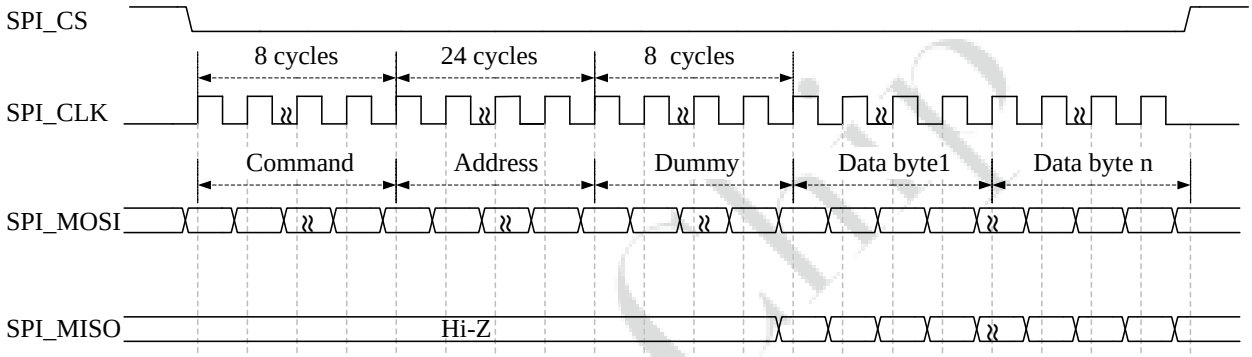
SPI 双路输入/ 双路输出模式是指通过两根信号线 SPI_MOSI 和 SPI_MISO 进行发送或接收数据， 传输速率是标准单路 SPI 模式的两倍。

- SPI_CS：片选控制信号，在 SPI_CS 为低电平期间进行数据传输。
- SPI_CLK：时钟线。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- SPI_MOSI 和 SPI_MISO：命令/ 地址/ 填充数据通过 MOSI 线传输，有效数据通过 SPI_MOSI 和 SPI_MISO 两根信号线同时传输。
- 配置寄存器 0x038 SPI_TMC.DUAL_EN 位为 1，可选择 SPI 双路模式。

双路输入/ 输出模式时序图如下：

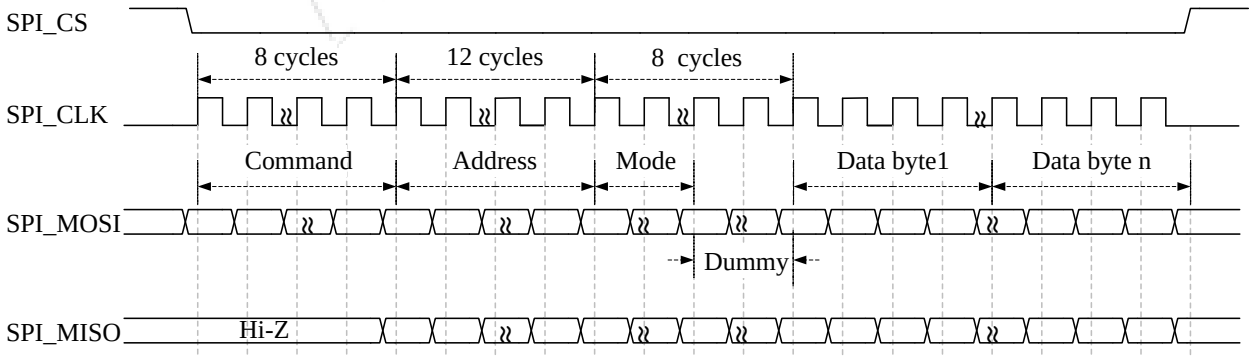


12.1.3.3.4. SPI 双路 I/O 模式

SPI 双路 I/O 模式的特点如下：

- 地址、填充数据和有效数据都通过两根信号线进行传输，命令通过 MOSI 进行单线传输。
- 配置寄存器 0x038 SPI_TMC.DUAL_EN 位为 1，可选择 SPI 双路 I/O 模式。

双路 IO 模式的时序图如下：



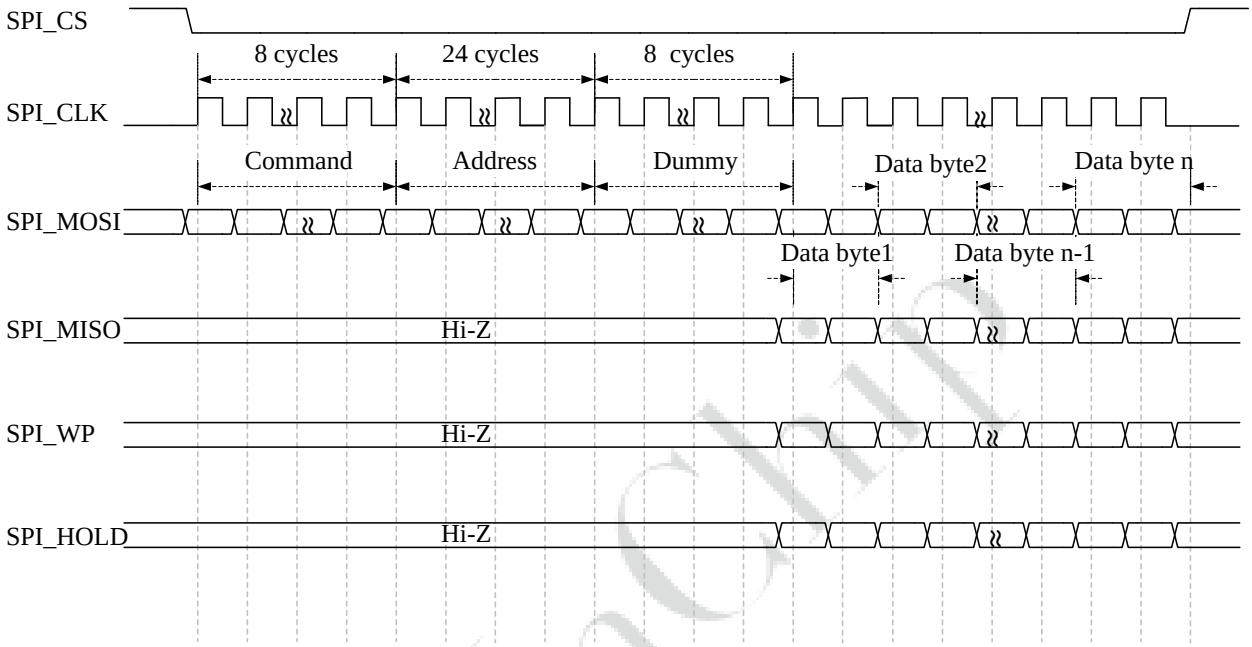
12.1.3.3.5. SPI 四路输入 /输出模式

SPI 四路输入输出模式是指通过四根信号线 （ MOSI/ MISO/ WP/ HOLD ）进行发送或接收数据，此时传输速率是标准 SPI 模式的四倍。

- 命令/ 地址/ 填充数据通过 MOSI 单线传输，有效数据通过四根信号线同时传输。
- 通过配置寄存器 0x038 SPI_TMC.QUAD_EN 位为 1 进行选择，

四路输入/输出模式的时序图如下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

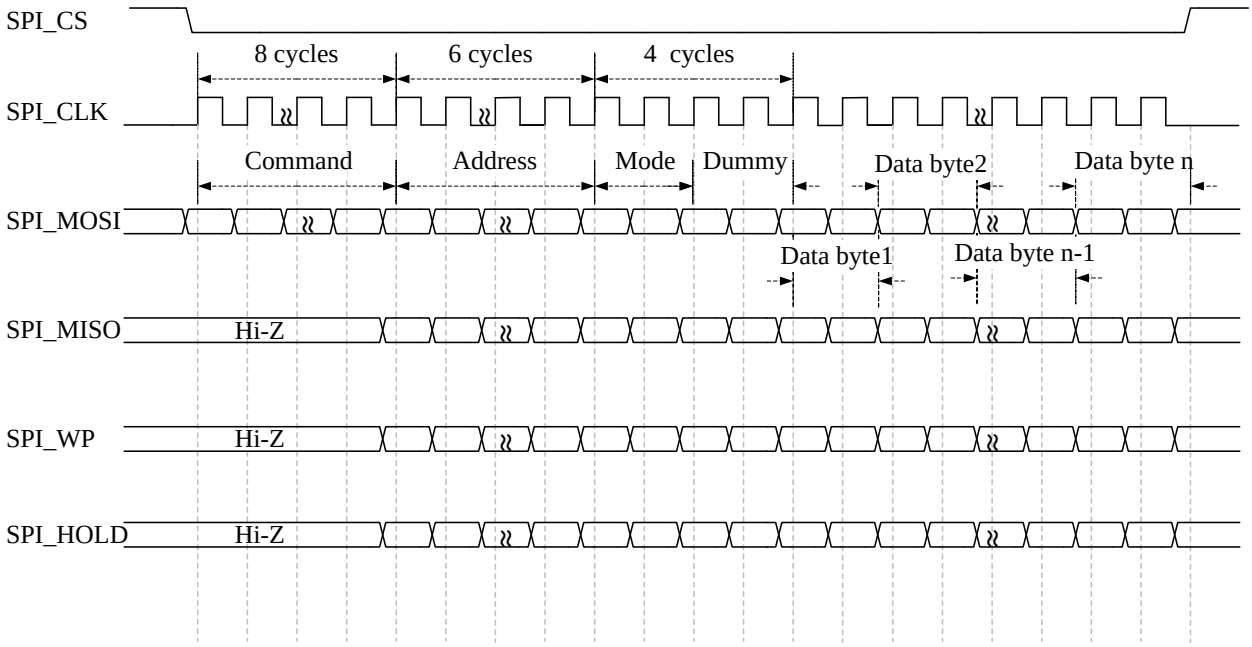


12.1.3.3.6. SPI 四路 I/O 模式

SPI 四路 I/O 模式的地址和数据信号全都使用 4 线传输。SPI 四路 I/O 模式下：

- 串行 Flash 设备支持快速读模式，即只需要发送一次读命令，后续操作可以跳过读命令，在 CS 信号拉低期间，仅发送地址即可读取数据。
- 支持带回环功能的突发读配置，支持的突发长度为 8/ 16/ 32/ 64 字节， 常用于 XIP 模式。
- 当 Mode 字节的 bit[5:4] = 2b'10 时，工作在读命令 bypass 模式， 即下一笔读操作不需要发送读命令。

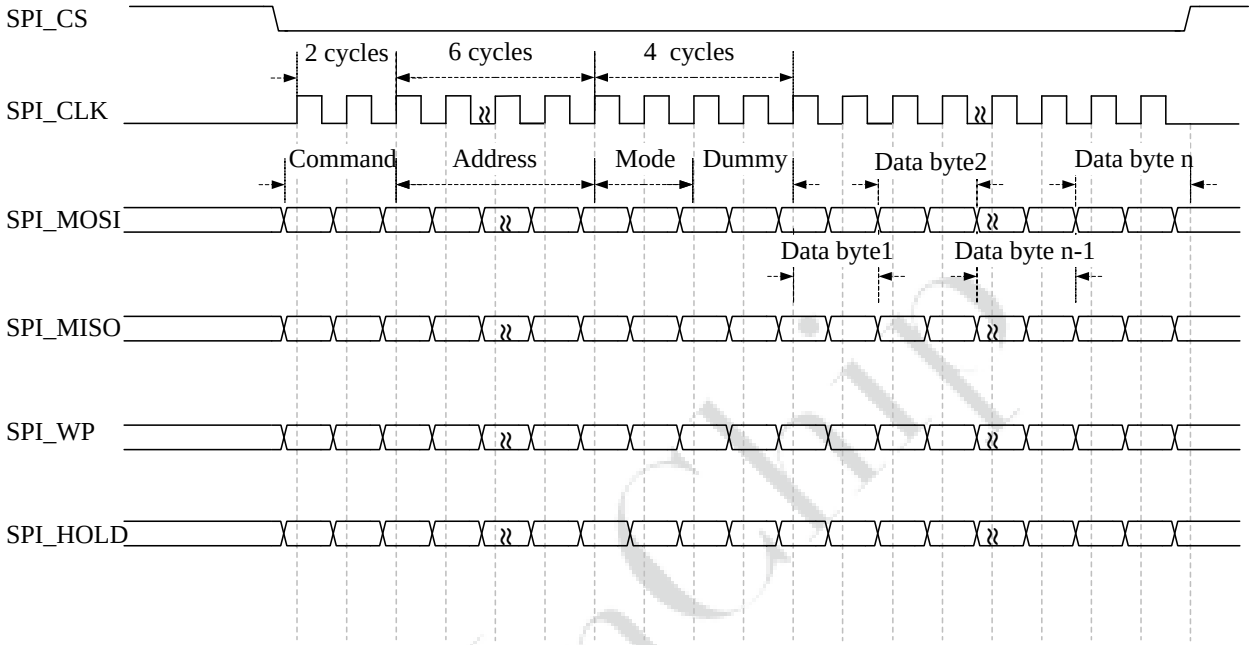
四路 I/O 模式时序图如下：



12.1.3.3.7. SPI QPI 模式

QPI 模式命令、地址和数据全部使用 4 线传输。QPI 模式不是标准的 SPI 模式，与标准 SPI 模式之间需要使用命令进行切换。QPI 模式也支持突发传输配置。时序图如下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



12.1.3.3.8. SPI 突发传输配置

当 SPI 工作在 Quad I/O 模式下，串行 Flash 设备支持带回环的突发传输 (Burst With Wrap)：

- 发送突发传输配置命令 0x77 或 0xC0，配置三个 dummy 字节。
- 配置 Mode 字节的 bit[6:4] 为相应的突发类型和突发长度。

常用于 XIP 访问模式下，CPU 进行 Cache 操作，可以提高总线的传输效率，提高代码的执行性能。当 CPU 使用 Cache，并开启预取功能，突发传输类型配置为 WRAP，突发传输长度配置为 32 个字节。

在 XIP 模式下，配置 SPI_BTR.WRAP_EN 位为 1，使能回环模式，控制器在发送第一笔接收命令之前，会自动根据 SPI_BTR 寄存器的配置，自动发送突发传输配置命令。

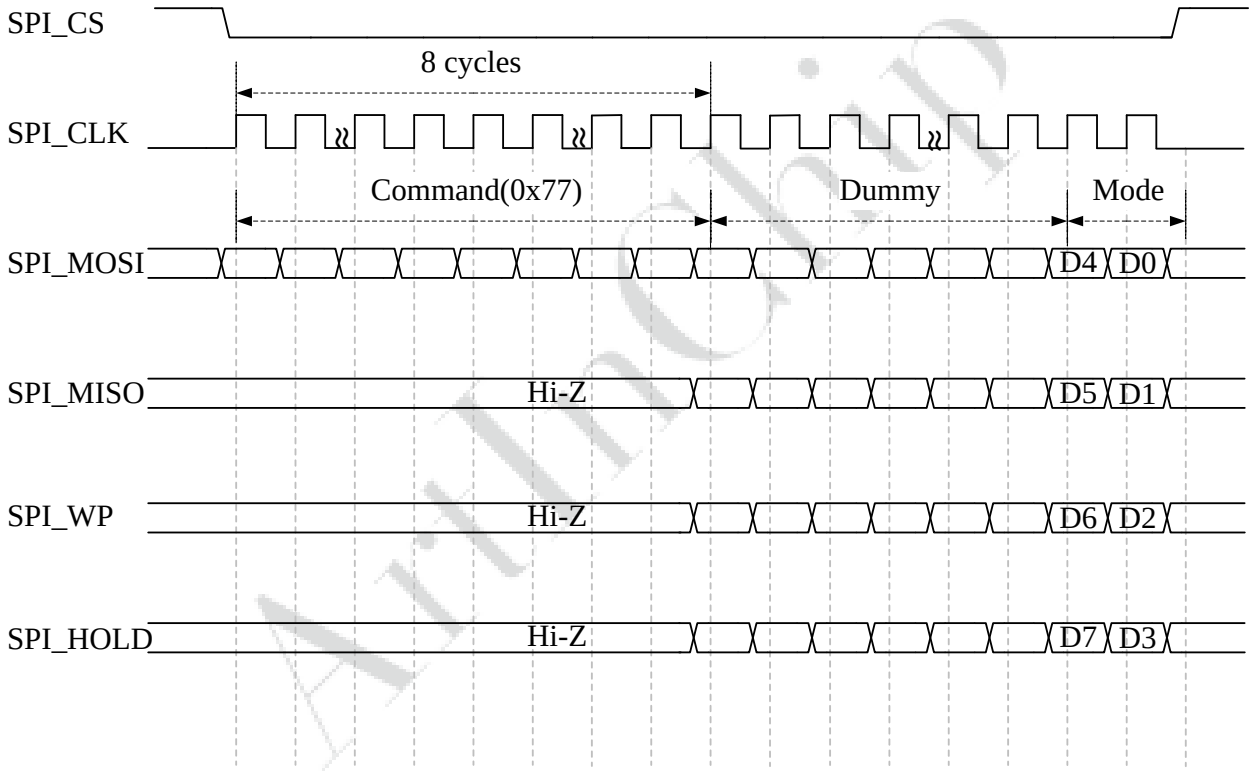
Mode Bit[6:4]	突发类型	突发长度
3b’001	Linear	取决与 Flash 容量
3b’000	WRAP	8 个字节
3b’010	WRAP	16 个字节
3b’100	WRAP	32 个字节
3b’110	WRAP	64 个字节

当 SPI 工作在 QPI 模式，突发传输配置命令与标准模式下的命令不同，增加了频率相关的 dummy 字节数量的配置。对于 QPI 模式，不同厂家的突发传输配置方式或许不同，需要参考具体厂家的芯片规格书。举例 Winbond 的 W25Q128JW 芯片定义如下：

Mode Bit[5:4]	Dummy 时钟周期数	Mode Bit[1:0]	WRAP 长度
2b’01	4（80MHz）	2b’01	16 个字节

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

Mode Bit[5:4]	Dummy 时钟周期数	Mode Bit[1:0]	WRAP 长度
2b’10	6（104MHz）	2b’10	32 个字节
2b’11	8（133MHz）	2b’11	64 个字节



12.1.3.4. XIP 模式

在主机模式下，配置 QSPI0 控制器工作于 XIP 模式，可以通过 AXI 直接读取 SPI 的物理地址空间，就像访问内存空间一样，常用于 XIP 启动。在 XIP 模式下，SPI 的物理地址需要映射为 AXI 可访问的逻辑地址。

XIP 控制逻辑主要包括如下模块：

- AXI slave 模块：用于和 AXI master 进行通讯；
- XIP 控制器：将 AXI 的地址和数据信息转换为 SPI 协议规定的传输格式；
- AXI 数据缓存：TX buffer 用于缓存 AXI 发送到 SPI 的命令和地址信息，而 RX buffer 用于缓存从 SPI 接口接收到的数据。

XIP 模式传输：允许 CPU 或 其它 AXI Master 直接从 SPI NOR 获取数据，无需先将数据读取到内存中再获取。这是一种高效的数据访问方式，可以减小内存的使用空间。常用于获取较大的图像数据。

XIP 模式启动：CPU 通过 AXI 直接访问 QSPI 的映射地址读取指令。在系统引导启动阶段，通过 AHB 接口初始化 QSPI 控制器。完成相关配置后，CPU 跳转到 QSPI 映射的起始地址进行取指令，完成系统启动。

XIP 的详细传输流程如下：

1. 系统上电后，在引导程序中通过 AHB 访问的方式对 QSPI 设备进行初始化。

   在系统启动阶段，需要配置基本的 SPI NOR 初始化信息，配置 SPI NOR 配置寄存器的 QE 位为 1 ，确保其工作在 Quad 模式。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

2. 通过 XIP 模式相关的寄存器，并使能 XIP 模式传输。配置流程如下：
   - 配置 SPI_RCM 寄存器，包括读命令指令值，dummy 数量，读命令 bypass 模式使能，Mode 字节等信息。
   - 配置 SPI_BTR 寄存器，包括突发传输配置命令指令值，传输位宽，dummy 数量，回环和线性模式指令等信息。
   - 配置 SPI_TMC 寄存器，配置 QADDR_EN 位置 1，选择 QIO 模式；配置 QPI_EN 位置 1，选择 QPI 模式。
   - 配置 SPI_CFG.AMOD 位为 1，使能 XIP 模式。

   注：

   如果 SPI NOR 使能读命令 bypass 模式，则除第一笔读操作外，后续的读操作将无需重复发送读命令。此配置可以简化读取流程，提升数据访问效率。
3. CPU 通过 AXI master 地址通道发起读请求，AXI slave 接收到读请求，获取地址信息、突发传输类型和突发长度。
4. XIP 控制器从寄存器 SPI_BTR 和 SPI_RCM 获取固定的数据传输格式，该方式仅支持 Quad I/O 和 QPI 模式。
5. 按照解析得到的传输格式，QSPI 控制器自动发送读命令，读取指令或数据信息并将其缓存到内部 RX buffer。
6. CPU 通过 AXI master 数据通道从 RX Buffer 中获取指令或数据。

   完成以上操作后，完整的读操作即告完成。

注：

如果开启 Cache 功能，CPU 每次读取 32 个字节数据，相当于一条 cache line 的大小，此时只需要配置一次突发长度即可。后续的 cache 读取都会使用这一固定的突发长度，且突发类型为 WRAP 模式，以此提高数据处理效率。

12.1.3.5. 连续传输

在主机模式下，SPI 可以进行连续数据传输，即发送或接收数据以字节为单位进行多个字节连续传输。详细功能配置如下：

- 发送数据的连续传输字节长度：通过寄存器 0x038 SPI_TMC.TXD_CNT 字段配置。
- 单路模式下，连续传输的长度：通过寄存器 0x038 SPI_TMC.STXD_CNT 字段配置。
- 控制器会自动发送填充数据，填充数据连续传输长度：通过寄存器 0x038 SPI_TMC.DMY_CNT 字段配置。
  - 如果不希望控制器自动传输填充数据，可以将填充数据作为发送数据一起写入寄存器0x038 SPI_TMC.TXD_CNT 字段。

在主机模式下，所有数据的突发传输长度配置在寄存器 0x030 SPI_TBC.TB_CNT 字段，当所有的发送数据和接收数据连续传输完成后，上述的配置字段都会被清零。

12.1.3.6. 采样模式

QSPI 控制器的接口最大时钟频率为 100 MHz。在主机模式下，内部的 SPI 时钟频率与外部接口的时钟频率一致。对于不同的时钟源，SPI 时钟可以工作在以下三种时钟模式下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 正常采样模式：采样时钟与输出时钟同相位，适用于标准的数据传输，没有额外的延时。
- 延时半周期采样模式：采样时钟相对于输出时钟延迟半个周期后进行数据采样，适用于当数据在半个时钟周期后达到稳定状态的场景，可以提供更准确的数据传输。

  QSPI 控制器默认工作在该采样模式。
- 延时一个周期采样模式：采样时钟相对于输出时钟延迟一个周期后进行数据采样，适用于数据线延时较大的场景，确保数据传输的完整性和准确性。

三种时钟的采样模式配置见下表：

表 12–2 采样模式配置

采样模式	RXDLY_DIS 位	RXINDLY_EN 位	时钟频率范围
正常模式	1	0	≤ 24 MHz
延时半个周期采样	0	0	24 ~ 60 MHz
延时一个周期采样	0	1	60 ~ 100 MHz

注：

采样模式与时钟频率的关系不是唯一确定的，取决于传输路径的延时时间，用户可根据实际情况选择合适的配置。

12.1.3.7. 位模式传输

位模式传输是以位为单位进行数据传输，用于对 SPI 接口的非 Flash 类型的外设进行访问，兼容更多的 SPI 设备。

位模式分为标准模式和三线模式。可通过下列步骤完成位模式传输配置：

1. 通过0x040 SPI_BMTC 寄存器的 BMOD_SEL 位域，选择两种不同的工作模式中的一种。

   注：

   关于标准模式和三线模式的详细说明，可查看总线模式 。
2. 通过 0x040 SPI_BMTC 寄存器选择位模式下的工作时钟。
3. 确定位模式的传输位数。位模式一次最大传输的位数为 32 bits：
   - 对于写操作，将待发送数据写入到 0x048 SPI_BMTXD 寄存器，并通过 0x040 SPI_BMTC.BM_TXCNT 配置写有效数据的长度。
   - 对于读操作，通过 0x040 SPI_BMTC.BM_RXCNT 配置读有效数据的长度， 读到的数据会自动存入 0x048 SPI_BMTXD 寄存器。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

4. 使能 BM_START 寄存器启动数据传输，传输完成后 BM_TD 寄存器会置 1，也可以使能 BM_TD_INTE 完成中断，在中断服务程序中检查数据传输完成标志。

> **注：**
>
> 位模式连续传输时，每笔传输结束后需要硬件复位一次，或者至少等待一个时钟周期，再传输下一笔数据。

12.1.3.8. 错误状态

QSPI 控制器在操作过程中可能会遇到多种错误状态，从而影响系统的正常运行和性能。对于 QSPI 控制器，如发生任何一种错误，硬件都会将 SPI 中断状态寄存器中对应的位置 1 并停止传输。以下是主要的错误状态描述及其对应的处理方法：

表 12–3 QSPI 控制器错误状态及处理方法

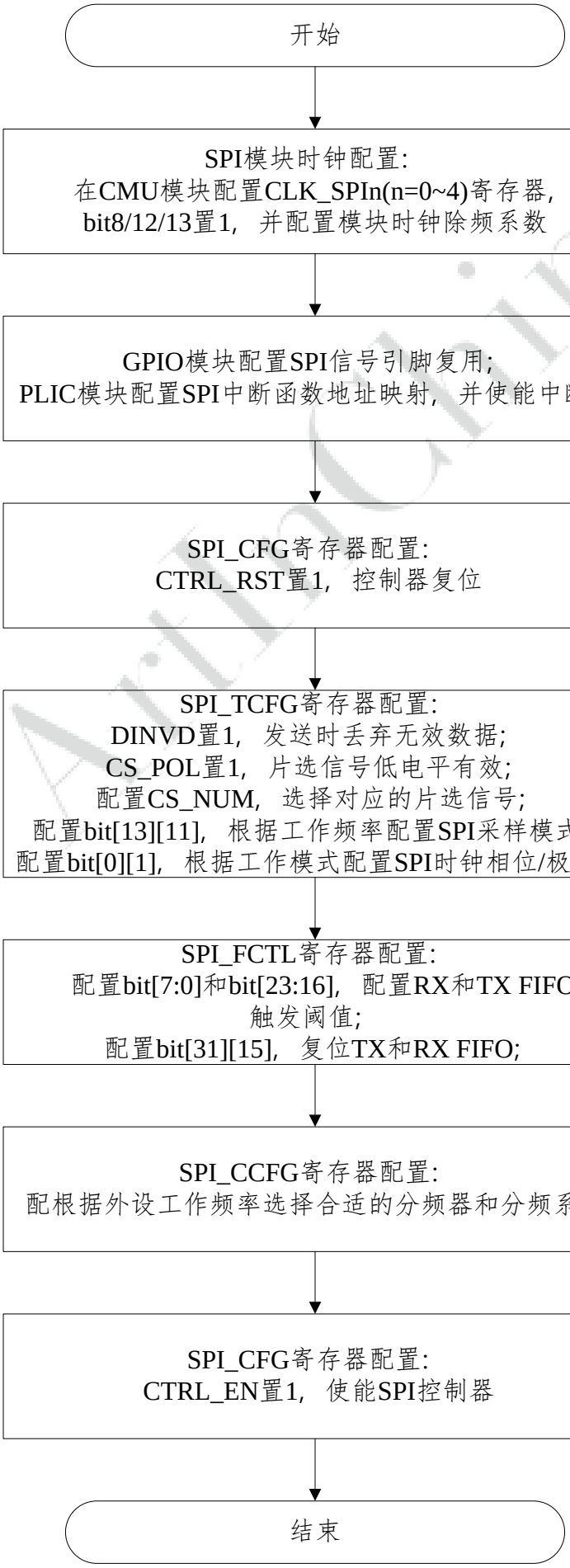
错误类型	原因	状态	处理方法及结果
TX_FIFO 下溢出	从空的 FIFO 中读取数据	控制器停止传输并将对应的中断状态位 TF_UDF 置 1。 此时控制器会产生一个硬件中断信号。	如果开始新的数据传输，需要在 0x004 SPI_CFG 寄存器中将 CTRL_RST 位置 1，将 FIFO 进行复位。处理完该异常后需要软件将该状态位清零。
TX_FIFO 上溢出	向满的 FIFO 中发送数据	控制器停止传输并将对应的中断状态位 TF_OVF 置 1。 此时控制器会产生一个硬件中断信号。	如果开始新的数据传输，需要在 0x004 SPI_CFG 寄存器中将 CTRL_RST 位置 1，将 FIFO 进行复位。处理完该异常后需要软件将该状态位清零。
RX_FIFO 下溢出	从空的 FIFO 中读取数据。	控制器停止传输并将对应的中断状态位 RF_UDF 置 1。 此时控制器会产生一个硬件中断信号。	如果开始新的数据传输，需要在 0x004 SPI_CFG 寄存器中将 CTRL_RST 位置 1，将 FIFO 进行复位。处理完该异常后需要软件将该状态位清零。
RX_FIFO 上溢出	向满的 FIFO 中发送数据	控制器停止传输并将对应的中断状态位 RF_OVF 置 1。 此时控制器会产生一个硬件中断信号	如果开始新的数据传输，需要在 0x004 SPI_CFG 寄存器中将 CTRL_RST 位置 1，将 FIFO 进行复位。处理完该异常后需要软件将该状态位清零。

12.1.4. 编程指南

12.1.4.1. 初始化流程

在进行数据传输之前，需要对 SPI 接口和控制器进行初始化。QSPI 控制器的初始化主要包括时钟配置，引脚配置，以及工作模式配置等，详情如下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.1.4.2. 字节模式传输流程

QSPI 在主机和从机之间传输串行数据，主要采用两种传输方式：

- CPU 模式
- DMA 模式

在数据传输过程中，QSPI 有两个数据通道：

- 发送通道：数据传输路径从 TX FIFO 到外部设备。

  发送数据时，CPU 或 DMA 将数据写入 TXD 寄存器，控制器自动将寄存器的数据搬到 TX FIFO 中，再由移位寄存器将 FIFO 中的数据发出。

- 接收通道：数据传输路径从外部设备到 RX FIFO。

  接收数据时，CPU 或 DMA 可以访问 SPI_RXD 寄存器获取接收数据，控制器自动将接收到的数据从 RX FIFO 搬到寄存器中。

每一笔数据传输完成后，QSPI 控制器都会发送一个传输完成的中断信号给处理器，此时 SPI 中断状态寄存器的 TD 位被置 1。

以下为不同模式下，QSPI 的数据传输流程：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

• 通过 CPU 在中断服务程序中读写 FIFO，完成数据传输，流程图如下：

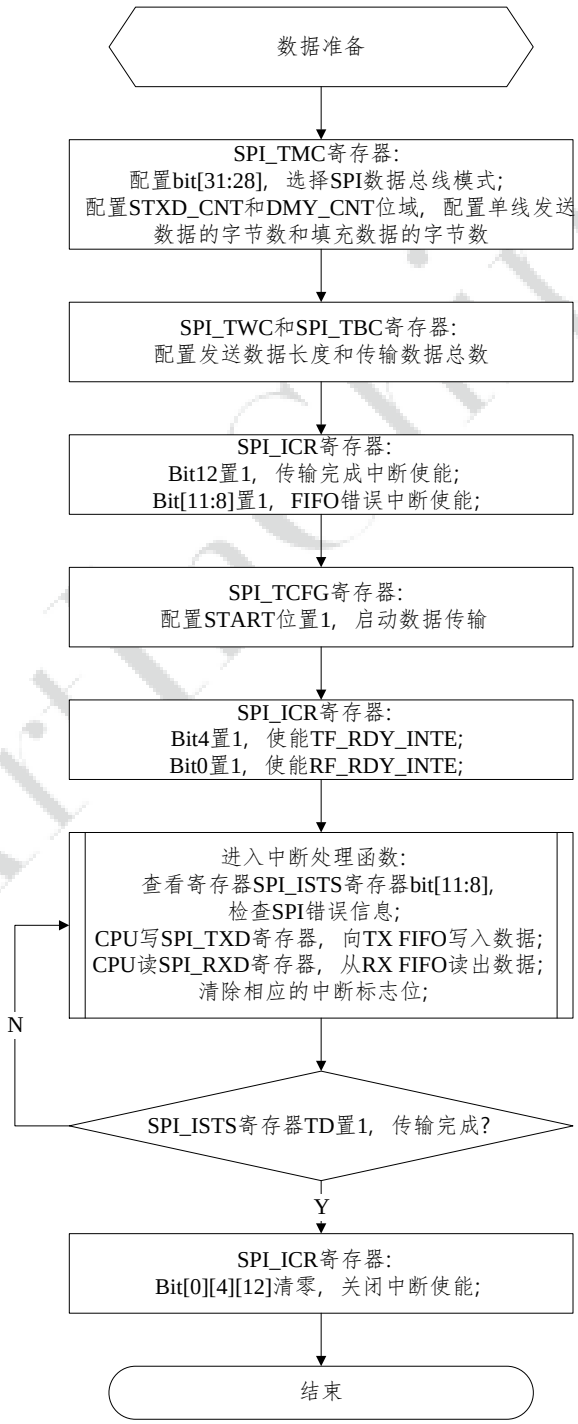


图 12–2 CPU 模式

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

• 通过 GPDMA 在内存和 FIFO 之间搬运数据，完成数据传输，流程图如下：

图 12–3 DMA 模式

QSPI 控制器同时支持使用内部 DMA 通过 AHB 主机接口搬运数据，只需要配置源地址和目标地址，以及数据长度即可。 相关的寄存器配置包括：

• SPI_IDMA_BTCFG

• SPI_IDMA_RXADDR

• SPI_IDMA_RXLEN

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- SPI_IDMA_TXADDR
- SPI_IDMA_TXLEN

图 12–4 IDMA 模式的配置流程

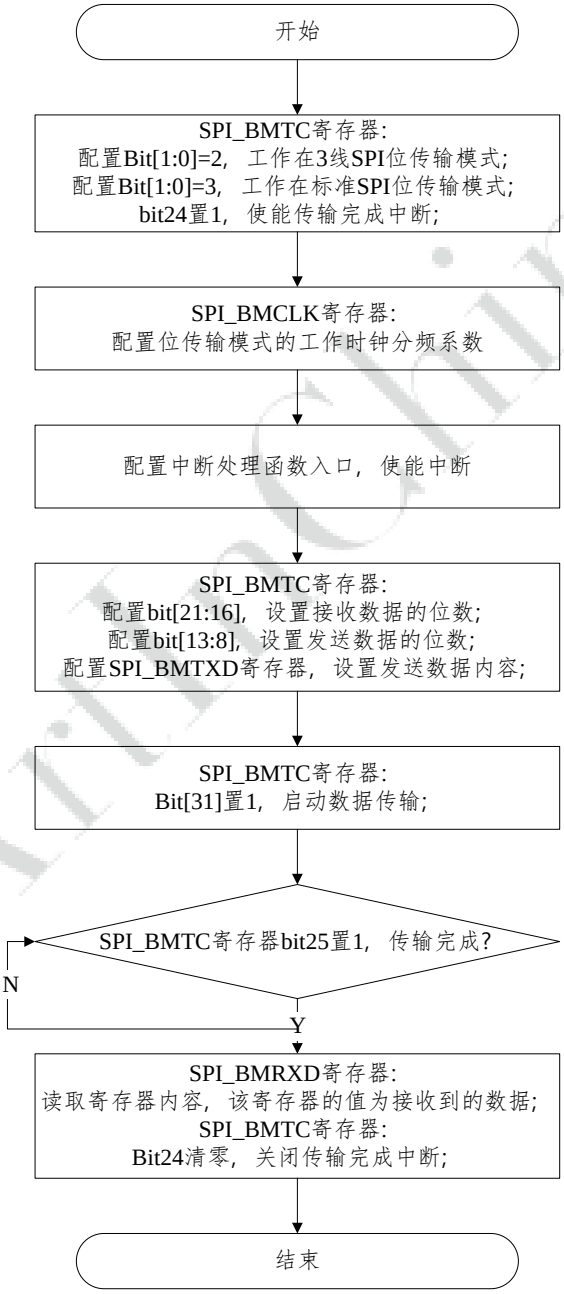
12.1.4.3. 位模式传输流程

SPI 在主机和从机之间传输串行数据，支持按位模式访问，相关的寄存器配置包括：

- SPI_BMTC
- SPI_BMCLK
- SPI_BMTXD
- SPI_BMRXD

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

流程图如下：



12.1.5. 寄存器列表

表 12–4 寄存器列表

偏移地址	缩写	寄存器描述	章节
0x000	SPI_VER	SPI 控制器版本	0x000 SPI_VER
0x004	SPI_CFG	SPI 配置	0x004 SPI_CFG
0x008	SPI_TCFG	SPI 传输配置	0x008 SPI_TCFG

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 12–4 寄存器列表 (续)

偏移地址	缩写	寄存器描述	章节
0x010	SPI_ICR	SPI 中断控制	0x010 SPI_ICR
0x014	SPI_ISTS	SPI 中断状态	0x014 SPI_ISTS
0x018	SPI_FCTL	SPI FIFO 控制	0x018 SPI_FCTL
0x01C	SPI_FSTS	SPI FIFO 状态	0x01C SPI_FSTS
0x024	SPI_CCFG	SPI 时钟配置	0x024 SPI_CCFG
0x028	SPI_MISC	SPI 其它功能配置	0x028 SPI_MISC
0x030	SPI_TBC	SPI 总传输字节计数器	0x030 SPI_TBC
0x034	SPI_TWC	SPI 写发送计数器	0x034 SPI_TWC
0x038	SPI_TMC	SPI 传输控制	0x038 SPI_TMC
0x040	SPI_BMTC	SPI 位模式传输配置	0x040 SPI_BMTC
0x044	SPI_BMCLK	SPI 位模式时钟配置	0x044 SPI_BMCLK
0x048	SPI_BMTXD	SPI 位模式发送数据值	0x048 SPI_BMTXD
0x04C	SPI_BMRXD	SPI 位模式接收数据值	0x04C SPI_BMRXD
0x050	SPI_BTR	SPI 突发传输配置	0x050 SPI_BTR
0x054	SPI_RCM	SPI 读命令模式	0x054 SPI_RCM
0x060	SPI_WRAP_LEN	SPI 读命令模式	0x060 SPI_WRAP_LEN
0x08C	SPI_IDMA_TXLEN	SPI IDMA 写传输长度	0x08C SPI_IDMA_TXLEN
0x090	SPI_IDMA_RXLEN	SPI IDMA 读传输长度	0x090 SPI_IDMA_RXLEN
0x094	SPI_IDMA_TXADDR	SPI IDMA 发送源地址	0x094 SPI_IDMA_TXADDR
0x098	SPI_IDMA_RXADDR	SPI IDMA 接收目的地址	0x098 SPI_IDMA_RXADDR
0x09C	SPI_IDMA_BTCFG	SPI IDMA 突发传输配置	0x09C SPI_IDMA_BTCFG
0x200	SPI_TXD	SPI 发送数据	0x200 SPI_TXD

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 12–4 寄存器列表 (续)

偏移地址	缩写	寄存器描述	章节
0x300	SPI_RXD	SPI 接收数据	0x300 SPI_RXD

12.1.6. 寄存器描述

12.1.6.1. 0x000 SPI_VER

默认值：0x00000300 or 0x00000201			SPI 控制器版本 (SPI Controller Version)
位域	类型	默认值	描述
31:16	–	–	–
15:0	R	0x300 or 0x201	VER 版本号 **注：** 不同控制器的版本号可能不同。

12.1.6.2. 0x004 SPI_CFG

默认值：0x00000080			SPI 配置 (SPI Configuration)
位域	类型	默认值	描述
31	R/WAC	0x0	CTRL_RST 控制器复位（Controller Reset） 置 1，SPI 控制器复位，完成后自动清零。
30:8	–	–	–
7	R/W	0x1	RXFULL_STOP 传输停止使能（RX FIFO Full Transfer Stop Enable） • 0：忽略 RXFIFO 状态 • 1：当 RXFIFO 满时停止传输数据
6:5	–	–	–
4	R/W	0x0	TXIDMA_EN TX 通道内部 DMA 使能（TX Inner DMA Enable）

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000080			SPI 配置 (SPI Configuration)
位域	类型	默认值	描述
3	R/W	0x0	RXIDMA_EN RX 通道内部 DMA 使能 （ RX Inner DMA Enable ）
2	R/W	0x0	AMOD 访问模式选择 （ Access Mode ） • 0：AHB 访问模式 • 1：AXI 访问模式 **注：** 使能 XIP 功能，该位必须置 1。退出 XIP 功能，该位必须清零。
1	R/W	0x0	CTRL_MODE_SEL 工作模式选择 （ Controller Work Mode Select ） • 0：保留 • 1：主机模式 **注：** 当 START = 1 时，不能写入。
0	R/W	0x0	CTRL_EN 控制器使能 （ Controller Enable ） • 0：禁用 • 1：使能 **注：** 当传输形式从位模式切换到字节模式时，必须使能一次。

12.1.6.3. 0x008 SPI_TCFG

默认值：0x00000087			SPI 传输配置 (SPI Transfer Configuration)
位域	类型	默认值	描述
31	R/WAC	0x0	START 启动数据传输 （ Start Data Transfer ）

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000087			SPI 传输配置 (SPI Transfer Configuration)
位域	类型	默认值	描述
			• 0：空闲状态 • 1：启动数据传输 置 1，开始执行 SPI 数据传输，完成后自动清零。 **注：** 当 START=1 时不能写入。
28:26	–	–	–
25	R/W	0x0	3WIRE_EN 三线 byte 模式传输使能（3-Wire Byte Mode Transfer Enable） • 0：标准模式，MOSI 与 MISO 引脚独立 • 1：使能三线 byte 模式传输，此时 MOSI 与 MISO 引脚复用
24:15	–	–	–
14	R/W	0x0	TXDLY_EN 发送数据延时模式使能（TX Data Delay Mode Enable） • 0：正常发送 • 1：使能延时发送 **注：** 在 Dual / Quad 模式下，该位必须置 1，发送数据延时半个时钟周期。
13	R/W	0x0	RXDLY_DIS 接收数据延时采样模式配置（RX Data Delay Sample Mode Disable） • 0：延时采样模式 • 1：正常采样模式 在正常采样模式下，SPI 在边沿采样数据。在延时采样模式下，SPI 延迟半个时钟后在下一个边沿采样。
12	R/W	0x0	LSB_EN 最低有效位优先传输使能（Low Significant Bit Transmit Enable） • 0：最高有效位先传输 • 1：最低有效位先传输

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000087			SPI 传输配置 (SPI Transfer Configuration)
位域	类型	默认值	描述
			注： 当 START=1 时不能写入。
11	R/W	0x0	RXINDLY_EN 接收数据内部采样点模式配置（RX Data Inner Sample Delay Mode Enable） • 0：内部读数据采样点不用延迟 • 1：内部读数据的采样点延迟半个时钟周期
10	R/W	0x0	HSWM 高速写模式配置（High Speed Write Mode Configuration） • 0：正常写模式 • 1：高速写模式，提前半个时钟周期切换到下一个数据 **注：** 当 START=1 时不能写入。
9	R/W	0x0	DMY_VAL 填充数据值配置（Dummy Byte Value Configuration） • 0：填充数据的值为 0x00 • 1：填充数据的值为 0xFF **注：** 当 START=1 时不能写入。
8	R/W	0x0	DINVD 丢弃无效数据（Discard Invalid Data） 发送数据时，设备端会主动发无效数据回来，此时如果不能及时读取 FIFO 的数据，该位需要使能，自动丢弃读到的无效数据。 • 0：接收所有的无效数据 • 1：丢弃所有的无效数据 **注：** 当 START=1 时不能写入。
7	R/W	0x1	CS_LEVEL 片选信号电平配置（CS Level Configuration）

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000087			SPI 传输配置 (SPI Transfer Configuration)
位域	类型	默认值	描述
			• 0：低电平 • 1：高电平 **注：** 当 START=1 时不能写入。
6	R/W	0x0	CS_CTL_SEL 片选信号控制模式选择 （CS Control Mode Select） • 0：SPI 控制器 • 1：软件控制 通常由控制器自动发送 CS 信号，当该位置 1，必须手动配置 LEVEL 的电平状态。 **注：** 当 START=1 时不能写入。
5:4	R/W	0x0	CS_NUM 选中外部 SPI 设备的 CS 引脚序号 （CS Pin Number） • 0：SPI_CS0 有效 • 1–3：保留 **注：** 当 START=1 时不能写入。
3	R/W	0x0	CS_VALID_CTL 片选信号有效控制 （CS Valid Mode Control） • 0：在 SPI 数据传输期间，片选信号一直有效 • 1：在 SPI 数据传输期间，传输暂停时片选信号无效 该位用于选择片选信号的输出波形形式。 **注：** 当 START=1 时不能写入，当 CS_CTL_SEL=0 时该位才有效。
2	R/W	0x1	CS_POL 片选信号极性控制 (CS Polarity Control) • 0：高电平有效 • 1：低电平有效

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000087			SPI 传输配置 (SPI Transfer Configuration)
位域	类型	默认值	描述
			注： 当 START=1 时不能写入。
1	R/W	0x1	CPOL 时钟极性控制 （ Clock Polarity Control ） • 0：空闲时为低电平 • 1：空闲时为高电平 注： 当 START=1 时不能写入。
0	R/W	0x1	CPHA 时钟 / 数据相位控制 （ Clock Phase Control ） • 0：时钟的第一个边沿采样数据 • 1：时钟的第二个边沿采样数据 注： 当 START=1 时不能写入。

12.1.6.4. 0x010 SPI_ICR

注：

bit [23:16] SPI0 配置无效，SPI1 有效。

默认值：0x00000000			SPI 中断控制 (SPI Interrupt Control)
位域	类型	默认值	描述
31:24	–	–	–
23	R/W	0x0	TXIDMA_DONE_INTE 内部 DMA TX 传输完成中断使能
22	R/W	0x0	RXIDMA_DONE_INTE 内部 DMA RX 传输完成中断使能
21	R/W	0x0	RXFIFO_UNALIGN_INTE

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			SPI 中断控制 (SPI Interrupt Control)
位域	类型	默认值	描述
			内部 DMA 模式下，RX FIFO 数据非 4byte 对齐中断使能
20	R/W	0x0	IDMA_ERROR_INTE 内部 DMA 总线错误中断使能
19	R/W	0x0	AXI_TRAN_DONE_INTE AXI 传输完成中断使能
18	R/W	0x0	AXI_CFG_ERROR_INTE AXI 配置错误中断使能
17	R/W	0x0	AXI_TRAN_ERROR_INTE AXI 传输错误中断使能
16	R/W	0x0	AHB_TRAN_ERROR_INTE AHB 传输错误中断使能
15:14	–	–	–
13	R/W	0x0	CS_INTE CS 无效中断使能（CS Invalid Interrupt Enable） 片选信号从有效状态变到无效状态。 • 0：禁用 • 1：使能
12	R/W	0x0	TD_INTE 传输完成中断使能（Transfer Done Interrupt Enable） • 0：禁用 • 1：使能
11	R/W	0x0	TF_UDR_INTE TX FIFO 下溢出中断使能（TX FIFO Underrun Interrupt Enable） • 0：禁用 • 1：使能 当 CPU / DMA 读取空的 TX FIFO 会产生此中断。
10	R/W	0x0	TF_OVF_INTE TX FIFO 上溢出中断使能（TX FIFO Overflow Interrupt Enable）

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			SPI 中断控制 (SPI Interrupt Control)
位域	类型	默认值	描述
			• 0：禁用 • 1：使能 当 CPU / DMA 写入满的 TX FIFO 会产生此中断。
9	R/W	0x0	RF_UDR_INTE RX FIFO 下溢出中断使能（RX FIFO Underrun Interrupt Enable） • 0：禁用 • 1：使能 当 CPU / DMA 读取空的 RX FIFO 会产生此中断。
8	R/W	0x0	RF_OVF_INTE RX FIFO 上溢出中断使能（RX FIFO Overflow Interrupt Enable） • 0：禁用 • 1：使能 当 CPU / DMA 写入满的 RX FIFO 会产生此中断。
7	–	–	–
6	R/W	0x0	TF_FUL_INTE TX FIFO 满中断使能（TX FIFO Full Interrupt Enable） • 0：禁用 • 1：使能
5	R/W	0x0	TF_EMP_INTE TX FIFO 为空中断使能（TX FIFO Empty Interrupt Enable） • 0：禁用 • 1：使能
4	R/W	0x0	TF_RDY_INTE TX FIFO 准备好中断使能（TX FIFO Ready Interrupt Enable） • 0：禁用 • 1：使能
3	–	–	–
2	R/W	0x0	RF_FUL_INTE RX FIFO 满中断使能（RX FIFO Full Interrupt Enable）

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			SPI 中断控制 (SPI Interrupt Control)
位域	类型	默认值	描述
			• 0：禁用 • 1：使能
1	R/W	0x0	RF_EMP_INTE RX FIFO 为空中断使能（RX FIFO Empty Interrupt Enable） • 0：禁用 • 1：使能
0	R/W	0x0	RF_RDY_INTE RX FIFO 准备好中断使能（RX FIFO Ready Interrupt Enable） • 0：禁用 • 1：使能

12.1.6.5. 0x014 SPI_ISTS

注：

bit [23:16] SPI0 配置无效，SPI1 有效。

默认值：0x00000032			SPI 中断状态 (SPI Interrupt Status)
位域	类型	默认值	描述
31:24	–	–	–
23	R/W1C	0x0	TXIDMA_DONE 内部 DMA TX 完成信号中断 置 1，表示内部 DMA 发送数据传输完成。
22	R/W1C	0x0	RXIDMA_DONE 内部 DMA RX 完成信号中断 置 1，表示内部 DMA 接收数据传输完成。
21	R/W1C	0x0	RXFIFO_UNALIGN RX FIFO 非对齐中断 内部 DMA 模式下，该位置 1，表示 RX FIFO 数据非 4byte 对齐。
20	R/W1C	0x0	IDMA_ERROR IDMA 错误中断

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000032			SPI 中断状态 (SPI Interrupt Status)
位域	类型	默认值	描述
			置 1，表示内部 DMA 发生总线错误状态。
19	R/W1C	0x0	AXI_TRAN_DONE AXI 传输完成中断 置 1，表示 AXI 传输完成。此时可结束 AXI 传输，开启 AHB 传输。
18	R/W1C	0x0	AXI_CFG_ERROR AXI 配置错误中断 置 1，表示 AXI 模式下寄存器配置错误。此时需立刻结束 AXI 传输，重新配置寄存器。
17	R/W1C	0x0	AXI_TRAN_ERROR AXI 传输错误中断 置 1，表示 AHB 正在传输时，发起 AXI 传输请求。此时需立刻结束 AHB 传输，启动 AXI 传输。
16	R/W1C	0x0	AHB_TRAN_ERROR AHB 传输错误中断 置 1，表示 AXI 正在传输时，发起 AHB 传输请求。此时需立刻结束 AXI 传输，启动 AHB 传输。
15:14	–	–	–
13	R/W1C	0x0	CS_INV 片选信号无效中断（CS Invalid Interrupt） 置 1，表示片选状态从有效变为无效。
12	R/W1C	0x0	TD 传输完成中断（Transfer Done Interrupt） • 0：正在传输 • 1：传输完成
11	R/W1C	0x0	TF_UDF TX FIFO 下溢出中断（TX FIFO Underrun Interrupt） • 0：TXFIFO 没有下溢出 • 1：TXFIFO 下溢出
10	R/W1C	0x0	TF_OVF TX FIFO 上溢出中断（TX FIFO Overflow Interrupt）

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000032			SPI 中断状态 (SPI Interrupt Status)
位域	类型	默认值	描述
			• 0：TXFIFO 没有上溢出 • 1：TXFIFO 上溢出
9	R/W1C	0x0	RF_UDF RX FIFO 下溢出中断（RX FIFO Underrun Interrupt） • 0：RXFIFO 没有下溢出 • 1：RXFIFO 下溢出
8	R/W1C	0x0	RF_OVF RX FIFO 上溢出中断（RX FIFO Overflow Interrupt） • 0：RXFIFO 没有上溢出 • 1：RXFIFO 上溢出
7	–	–	–
6	R/W1C	0x0	TF_FULL TX FIFO 满中断（TX FIFO Full Interrupt） • 0：TXFIFO 未满 • 1：TXFIFO 已满
5	R/W1C	0x1	TF_EMP TX FIFO 为空中断（TX FIFO Empty Interrupt） • 0：TXFIFO 非空 • 1：TXFIFO 为空
4	R/W1C	0x1	TF_READY TX FIFO 准备好中断（TX FIFO Ready Interrupt） • 0：TX_WL > TX_WATER_MARK • 1：TX_WL <= TX_WATER_MARK 该位置 1，CPU 可以向 TX FIFO 写入数据。
3	–	–	–
2	R/W1C	0x0	RF_FULL RX FIFO 满中断（RX FIFO Full Interrupt） • 0：RXFIFO 未满 1：RXFIFO 已满

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000032			SPI 中断状态 (SPI Interrupt Status)
位域	类型	默认值	描述
1	R/W1C	0x1	RF_EMP RX FIFO 为空中断（RX FIFO Empty Interrupt） • 0：RXFIFO 非空 • 1：RXFIFO 为空
0	R/W1C	0x0	RF_READY RX FIFO 准备好中断（RX FIFO Ready Interrupt） • 0：RX_WL < RX_WATER_MARK • 1：RX_WL >= RX_WATER_MARK 该位置 1，CPU 可以从 RX FIFO 读出数据。

12.1.6.6. 0x018 SPI_FCTL

默认值：0x00400001			SPI FIFO 控制 (SPI FIFO Control)
位域	类型	默认值	描述
31	R/WAC	0x0	TF_RST TX FIFO 复位，完成后自动清零。
	–	–	–
24	R/W	0x0	TF_DREQ_EN TX FIFO DMA 请求使能 • 0：禁用 • 1：使能
23:16	R/W	0x40	TF_WATER_MARK TX FIFO 发出写请求信号的 FIFO 水位值，当 FIFO 水位小于该值，并且 TF_DREQ_EN=1, 向 DMA 发起写请求。
15	R/WAC	0x0	RF_RST RX FIFO 复位，完成后自动清零。
14:9	–	–	–
8	R/W	0x0	RF_DREQ_EN RX FIFO DMA 请求使能。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00400001			SPI FIFO 控制 (SPI FIFO Control)
位域	类型	默认值	描述
			• 0：禁用 • 1：使能
7:0	R/W	0x1	RF_WATER_MARK RX FIFO 发出读请求信号的 FIFO 水位值，当 FIFO 水位大于该值，并且 RF_DREQ_EN=1, 向 DMA 发起读请求。

12.1.6.7. 0x01C SPI_FSTS

默认值：0x00000000			SPI FIFO 状态 (SPI FIFO Status)
位域	类型	默认值	描述
31	R	0x0	TF_WBUF_STS TX FIFO 写缓存状态（TX FIFO Write Buffer Status） 置 1，表示写缓存正在执行写操作。
30:25	R	0x0	TF_WBUF_CNT 写缓存计数器（TX FIFO Write Buffer Counter） 表示 TX FIFO 写缓存中字节的数量。 **注：** 注意：QSPI0 最大 4 个字节，QSPI1 最大 32 个字节
24	–	–	–
23:16	R	0x0	TF_CNT TX FIFO 计数器（TX FIFO Counter） • 0：在 TX FIFO 中有 0 字节数据 • 1：在 TX FIFO 中有 1 字节数据 • … • 64：在 TX FIFO 中有 64 字节数据 **注：** QSPI0 最大 64 + 4 = 68 个字节，QSPI1 最大 64 + 32 = 96 个字节
15	R	0x0	RF_RBUF_STS RX FIFO 读缓存状态（RX FIFO Read Buffer Status）

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			SPI FIFO 状态 (SPI FIFO Status)
位域	类型	默认值	描述
			置 1，表示读缓存正在执行读操作。
14:8	R	0x0	RF_RBUF_CNT 读缓存计数器（RX FIFO Read Buffer Counter） 表示 RX FIFO 读缓存中字节的数量。 **注：** QSPI0 最大 4 个字节，QSPI1 最大 64 个字节
7:0	R	0x0	RF_CNT RX FIFO 计数器（RX FIFO Counter） • 0：在 RX FIFO 中有 0 字节 • 1：在 RX FIFO 中有 1 字节 • … • 64：在 RX FIFO 中有 64 字节 **注：** QSPI0 最大 68 个字节，QSPI1 最大 128 个字节

12.1.6.8. 0x024 SPI_CCFG

默认值：0x00000002			SPI 时钟配置 (SPI Clock Configuration)
位域	类型	默认值	描述
31:13	–	–	–
12	R/W	0x0	CKDIV_SEL 时钟分频器选择（Clock Divider Select） • 0：选择时钟分频器 1 • 1：选择时钟分频器 2
11:8	R/W	0x0	CKDIV1 时钟分频器 1（Clock Divider 1） SPI 接口时钟频率根据如下公式计算 SPI_CLK = 源时钟频率/(2^CKDIV1)
7:0	R/W	0x2	CKDIV2 时钟分频器 2（Clock Divider 2）

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000002			SPI 时钟配置 (SPI Clock Configuration)
位域	类型	默认值	描述
			SPI 接口时钟频率根据如下公式计算： SPI_CLK = 源时钟频率/(2*(CKDIV2 + 1))

12.1.6.9. 0x028 SPI_MISC

12.1.6.10. 0x030 SPI_TBC

默认值：0x00000000			SPI 总传输字节计数器 (SPI Total Bytes Counter)
位域	类型	默认值	描述
31:24	–	–	–
23:0	R/W	0x0	TB_CNT 总传输字节计数器（Total Bytes Counter） 该值定义了传输的字节总数，包括发送，接收和填充数据。 • 0：共传输 0 个字节 • 1：共传输 1 个字节 • … • N：共传输 N 个字节 **注：** 传输数据总数=发送数据+接收数据+填充数据

12.1.6.11. 0x034 SPI_TWC

默认值：0x00000000			SPI 写发送计数器 (SPI Transmit Write Counter)
位域	类型	默认值	描述
31:24	–	–	–
23:0	R/W	0x0	TXD_CNT 写数据计数器（TX Data Counter） 该值定义了在填充数据之前发送到 TX FIFO 的写数据字节总数，为了节约带宽，填充数据由控制器自动发送。 • 0：共发送 0 个字节 • 1：共发送 1 个字节

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			SPI 写发送计数器 (SPI Transmit Write Counter)
位域	类型	默认值	描述
			• … • N：共发送 N 个字节

12.1.6.12. 0x038 SPI_TMC

默认值：0x00000000			SPI 传输控制 (SPI Transmit Miscellaneous Control)
位域	类型	默认值	描述
31	R/W	0x0	QPI_EN QPI 模式使能（QPI Transfer mode Enable） • 0：命令使用 1 线模式传输 • 1：命令使用 4 线模式传输 QPI 模式下，命令/地址/数据都使用 4 线模式传输。 **注：** 仅当 QUAD_EN=1 时有效。
30	R/W	0x0	QADDR_EN QIO 模式使能（QIO Transfer mode Enable） • 0：地址使用 1 线模式传输 • 1：地址使用 4 线模式传输 QIO 模式下，该位置 1，命令使用 1 线传输，地址/数据使用 4 线传输。 **注：** 仅当 QUAD_EN=1 时有效。
29	R/W	0x0	QUAD_EN 四线数据传输模式使能（Quad Transfer Mode Enable） • 0：4 线数据模式禁用 • 1：4 线数据模式使能 **注：** 4 线数据模式包括 4 线输入和 4 线输出。
28	R/W	0x0	DUAL_EN 双线数据传输模式使能（Dual Transfer Mode Enable）

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			SPI 传输控制 (SPI Transmit Miscellaneous Control)
位域	类型	默认值	描述
			• 0：发送或接收数据使用单线数据传输模式 • 1：发送或接收数据使用双线数据传输模式 **注：** 当 START=1 时不能写入，仅当 QUAD_EN=0 时有效。
27:24	R/W	0x0	DMY_CNT 填充数据计数器（Dummy Bytes Counter） 定义了待发送的填充数据的字节数，用于双路模式和四路模式。 • 0：共发送 0 个字节填充数据 • 1：共发送 1 个字节填充数据 • … • N：共发送 N 个字节填充数据 **注：** 当 START=1 时不能写入。
23:0	R/W	0x0	STXD_CNT 单线发送数据计数器（Single TX Data Counter） 该值定义了需要使用单线模式发送的数据长度。 • 对于单线模式：包括命令、地址、数据。 • 对于双线和四线模式：仅包括命令、地址。 • 对于双线 IO 模式：仅包括命令。 • 0：单线共发送 0 个字节 • 1：单线共发送 1 个字节 • … • N：单线共发送 N 个字节 **注：** 当 START=1 时不能写入。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.1.6.13. 0x040 SPI_BMTC

默认值：0x000000A0			SPI 位模式传输配置 (SPI Bit-Mode Transfer Configuration)
位域	类型	默认值	描述
31	R/WAC	0x0	BM_START 启动传输数据（Bit-Mode Start Data Transfer） • 0：空闲状态 • 1：启动传输数据 该字段置 1，启动发送串行数据流，数据存放在 (SPI_BMTXD 和 SPI_BMRXD 寄存器），传输完成后自动清零。
30	R/W	0x0	BM_SMP_SEL 采样模式选择（Bit-Mode Sample Mode Select） • 0：延时采样模式 • 1：标准采样模式 在标准采样模式下，SPI 主机在 SCLK 的上升边沿采样数据。 在延时采样模式下，SPI 主机延时半个时钟周期，在 SCLK 的上升边沿采样数据。
29:26	–	–	–
25	R/W1C	0x0	BM_TD 传输完成（Bit-Mode Transfer Done） • 0：忙等待 • 1：传输完成 置 1，表示最后一位数据传输完成，该字段写 1 清零。 **注：** 传输完成后，该位需要重复写 1 清零，检查该位是否为 0，直到清零后再启动下一笔传输。
24	R/W	0x0	BM_TD_INTE 传输完成中断使能（Bit-Mode Transfer Done Interrupt Enable）。 • 0：禁用 • 1：使能 **注：** 当总线模式选择 2 /3 时有效。
23:22	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x000000A0			SPI 位模式传输配置 (SPI Bit-Mode Transfer Configuration)
位域	**类型**	**默认值**	**描述**
21:16	R/W	0x0	BM_RXCNT 接收串行数据长度（Bit-Mode RX Data Length） • 000000b：接收 0 比特 • 000001b：接收 1 比特 • … • 100000b：接收 32 比特 **注：** 当总线模式选择 2 /3 时有效，当 BM_START=1 时不能写入。
15:14	–	–	–
13:8	R/W	0x00	BM_TXCNT 发送串行数据长度（Bit-Mode TX Data Length） • 000000b：发送 0 比特 • 000001b：发送 1 比特 • … • 100000b：发送 32 比特 **注：** 当总线模式选择 2 /3 时有效，当 BM_START=1 时不能写入。
7	R/W	0x1	BMCS_LEVEL 片选信号电平配置 • 0：配置片选信号为低电平 • 1：配置片选信号为高电平 **注：** 当总线模式选择 2 /3 时有效，当 BM_START=1 时不能写入。
6	R/W	0x0	BMCS_CTRL_SEL 片选信号输出控制方式选择 • 0：SPI 控制器 • 1：软件控制 通常控制器会自动控制片选信号高低电平。当该位置 1，则需要通过软件编程写入 SPI_TCFG.CS_LEVEL 的值，来控制片选信号。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x000000A0			SPI 位模式传输配置 (SPI Bit-Mode Transfer Configuration)
位域	类型	默认值	描述
			注： 当总线模式选择 2 /3 时有效，当 BM_START=1 时不能写入。
5	R/W	0x1	BMCS_POL 片选信号极性控制 • 0：高电平有效 • 1：低电平有效 注： 当总线模式选择 2 /3 时有效，当 BM_START=1 时不能写入。
4	–	–	–
3:2	R/W	0x0	BMCS_NUM 片选信号引脚序号选择 • 0：SPI_CS0 选中 • 1–3：保留 注： 当总线模式选择 2 /3 时有效，当 BM_START=1 时不能写入。
1:0	R/W	0x0	BMOD_SEL 总线模式选择 （ Bus Mode Select ） • 0：数据流按字节对齐，工作在标准 SPI，双路输入 /双路输出 SPI，双路 IO SPI，或四路输入/四路输出 SPI • 1：预留 • 2：数据流按位对齐，工作在三线模式下 • 3：数据流按位对齐，工作在标准 SPI 模式下

12.1.6.14. 0x044 SPI_BMCLK

默认值：0x02000000			SPI 位模式时钟配置 (SPI Bit-Mode Clock Configuration)
位域	类型	默认值	描述
31:8	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x02000000			SPI 位模式时钟配置 (SPI Bit-Mode Clock Configuration)
位域	类型	默认值	描述
7:0	R/W	0x0	BM_CKDIV 位模式时钟分频比例（Bit-Mode Clock Divide） 接口时钟频率计算公式如下： SPI_CLK = 源时钟/(2*BM_CKDIV) **注：** 当 BM_CKDIV 为 0，SPI_CLK = 源时钟/2

12.1.6.15. 0x048 SPI_BMTXD

默认值：0x00000000			SPI 位模式发送数据值 (SPI Bit-Mode TX Data)
位域	类型	默认值	描述
31:0	R/W	0x0	BMTXD 位模式发送数据值 该寄存器用来存储发送的数据 **注：** 在传输过程中，最高有效位先发送。

12.1.6.16. 0x04C SPI_BMRXD

默认值：0x00000000			SPI 位模式接收数据值 (SPI Bit-Mode RX Data)
位域	类型	默认值	描述
31:0	R/W	0x0	BMRXD 位模式接收数据值 该寄存器用来存储接收的数据 **注：** 在传输过程中，最高有效位先接收。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.1.6.17. 0x050 SPI_BTR

注：

该寄存器仅在 XIP 模式下配置有效。

默认值：0x77304010			SPI 突发传输配置 (SPI Burst Set)
位域	**类型**	**默认值**	**描述**
31:24	R/W	0x77	CMD_INDEX SPI 命令指令值 SPI 在 XIP 模式下，发送的 Wrap Burst 命令，支持 0x77 和 0xC0 两种典型值。当 QPI_EN=1，CMD 使用 4 线传输。否则，CMD 使用 1 线传输。 **注：** 0x77 命令仅支持 QIO 模式。0xC0 命令支持标准单线模式或 QPI 模式。
23:20	R/W	0x3	DUMMY_BYTE 传输配置填充数据字节数 定义 dummy 的字节数，为固定值或取决于工作时钟频率，可根据 WRAP 的值配置。
19	R/W	0x0	WIDTH 填充数据和有效数据位宽 Dummy 和数据位宽配置寄存器，在 QPI 模式下无效。 • 该位为 0，默认为 4–bit 位宽。 • 该位为 1，配置为 1–bit 位宽。 **注：** 当 burst 命令为 0xC0，工作在 QIO 模式下，该位必须配置为 1。 **工作在 QPI 模式下，该位不用配置。**
18	R/W	0x0	AUTO_WRAP_LEN 自动 Wrap 长度 • 0：wrap 长度为固定值，通过 WRAPPED 位域获取设定值。 • 1：wrap 长度可变，由硬件根据 SPI_WLEN 寄存器获取设定值，自动配置 wrap 长度。
17	–	–	–
16	R/W	0x0	WRAP_EN Wrap 模式使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x77304010			SPI 突发传输配置 (SPI Burst Set)
位域	类型	默认值	描述
			当系统开启 Cache 功能，该位必须使能。
15:8	R/W	0x40	SPI_BURST_WRAPPED Wrap 模式指令值 SPI 模式 XIP 使能时，若 AXI burst type 是 linear 切换 wrapped，触发此命令发送。Wrap 的值可以确定突发传输的长度和 dummy 的字节数，根据厂家的规格书进行设置。
7:0	R/W	0x10	SPI_BURST_LINEAR 线性模式指令值 SPI 模式 XIP 使能时，若 AXI burst type 是 wrapped 切换 linear，触发此命令发送

12.1.6.18. 0x054 SPI_RCM

注：

该寄存器仅在 XIP 模式下配置有效。

默认值：0xEB20A000			SPI 读命令模式 (SPI Read Command Mode)
位域	类型	默认值	描述
31:24	R/W	0xEB	CMD_INDEX 读命令指令值 SPI 在 XIP 模式下，定义读命令序号。
23:20	R/W	0x2	DUMMY_BYTE 读操作填充数据字节数 SPI 在 XIP 模式下，定义 dummy 时钟周期的个数。对于不同颗粒该值可能为固定值，或根据工作频率进行选择。
19	–	–	–
18	R/W	0x0	ADDR_4BYTE_EN4 字节地址模式使能 SPI 在 XIP 模式下，使能地址位采用 4 bytes 模式传输
17	R/W	0x0	READ_MODE_BYTE_ENSPI 读命令模式位使能 该位置 1，地址后面增加 1 byte 模式位，模式位的线宽与地址位一致，模式位的设定值由 RDCMD_BYPASS_EN 决定。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0xEB20A000			SPI 读命令模式 (SPI Read Command Mode)
位域	类型	默认值	描述
			注： SPI 工作在 Dual I/O，Quad I/O，QPI 模式下时，用户可以配置该位使能，否则当作 dummy 处理。
16	R/W	0x0	RDCMD_BYPASS_ENSPI 读命令 bypass 模式使能 该位置 1，模式位的设定值为 RDCMD_BYPPASS_CODE，否则，模式位的设定值为 RDCMD_NORMAL_CODE。
15:8	R/W	0xA0	RDCMD_BYPASS_CODE 读命令 bypass 模式指令值 在 bypass 模式下，第一次需要发送读命令，之后的连续读操作不需要再次发送读命令。
7:0	R/W	0x00	RDCMD_NORMAL_CODE 读命令正常模式指令值 在正常模式下，每次读操作都需要发送读命令。

12.1.6.19. 0x060 SPI_WRAP_LEN

注：

该寄存器仅在 XIP 模式下配置有效。该寄存器的值与 Wrap 命令的值有关，0x77 和 0xC0 命令对应的值不同，有些器件还包含 dummy 长度的配置信息，该寄存器需要在 XIP 传输之前配置好。

默认值：0x03020100			SPI 读命令模式 (SPI Wrap Length Config)
位域	类型	默认值	描述
31:24	R/W	0x03	WRAP64 Wrap 长度为 64 字节对应的命令参数
23:16	R/W	0x02	WRAP32 Wrap 长度为 32 字节对应的命令参数
15:8	R/W	0x01	WRAP16 Wrap 长度为 16 字节对应的命令参数
7:0	R/W	0x00	WRAP08 Wrap 长度为 8 字节对应的命令参数

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.1.6.20. 0x08C SPI_IDMA_TXLEN

注：

该寄存器仅在 IDMA 模式下配置有效。

默认值：0x00000000			SPI IDMA 写传输长度 (SPI Inner DMA Write Transaction Length)
位域	类型	默认值	描述
31:24	–	–	–
23:0	R/W	0x00	IDMA_TXLEN 内部 DMA 发送的数据长度 内部 DMA 通过 AHB 总线发送数据到 SPI 控制器的写缓存。

12.1.6.21. 0x090 SPI_IDMA_RXLEN

注：

该寄存器仅在 IDMA 模式下配置有效。

默认值：0x00000000			SPI IDMA 读传输长度 (SPI Inner DMA Read Transaction Length)
位域	类型	默认值	描述
31:24	–	–	–
23:0	R/W	0x00	IDMA_RXLEN 内部 DMA 接收的数据长度 内部 DMA 通过 AHB 总线从 SPI 控制器的读缓存接收数据。

12.1.6.22. 0x094 SPI_IDMA_TXADDR

注：

该寄存器仅在 IDMA 模式下配置有效。

默认值：0x00000000			SPI IDMA 发送源地址 (SPI Inner DMA TX Source Address)
位域	类型	默认值	描述
31:0	R/W	0x0	IDMA_TXADDR 内部 DMA 发送数据的源地址

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.1.6.23. 0x098 SPI_IDMA_RXADDR

注：

该寄存器仅在 IDMA 模式下配置有效。

默认值：0x00000000			SPI IDMA 接收目的地址 (SPI Inner DMA RX Destination Address)
位域	**类型**	**默认值**	**描述**
31：0	R/W	0x0	IDMA_RXADDR 内部 DMA 接收数据的目标地址

12.1.6.24. 0x09C SPI_IDMA_BTCFG

注：

该寄存器仅在 IDMA 模式下配置有效。

默认值：0x00004A04			SPI IDMA 突发传输配置 (SPI Inner DMA Burst Config)
位域	**类型**	**默认值**	**描述**
31:14	–	–	–
13	R/W	0x0	RX_PRI_EN RX 优先传输使能
12	R/W	0x0	AUTO_LEN 自动突发传输长度配置使能 开启后控制器会根据 Buffer 空满状态自动调整突发传输长度。
11:10	R/W	0x2	RX_BURST_LEN RX 突发传输长度配置 • 0： 突发传输 4 字节 • 1： 突发传输 16 字节 • 2：突发传输 32 字节 • 3：突发传输 64 字节
9:8	R/W	0x2	TX_BURST_LEN TX 突发传输长度配置 • 0： 突发传输 4 字节 • 1： 突发传输 16 字节

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00004A04			SPI IDMA 突发传输配置 (SPI Inner DMA Burst Config)
位域	类型	默认值	描述
			• 2：突发传输 32 字节 • 3：预留
7:0	–	–	–

12.1.6.25. 0x200 SPI_TXD

注：

该寄存器为只写。

默认值：0x00000000			SPI 发送数据 (SPI TX Data)
位域	类型	默认值	描述
31:0	WO	0x0	TXD 发送数据值 该寄存器可以通过 AHB 总线以字节，半字或字为单位进行访问： • 以字节访问时，每次写一笔数据到 TX FIFO，FIFO 深度加 1。 • 以半字访问时，每次写两笔数据到 TX FIFO，FIFO 深度加 2。 • 以字为单位访问时，每次写四笔数据到 TX FIFO，FIFO 深度加 4。

12.1.6.26. 0x300 SPI_RXD

注：

该寄存器为只读。

默认值：0x00000000			SPI 接收数据 (SPI RX Data)
位域	类型	默认值	描述
31:0	RO	0x0	RXD 接收数据值 该寄存器可以通过 AHB 总线以字节，半字或字为单位进行访问。 • 以字节访问时，每次从 RX FIFO 读一笔数据到寄存器，FIFO 深度减 1。 • 以半字访问时，每次从 RX FIFO 读两笔数据，FIFO 深度减 2。 • 以字为单位访问时，每次从 RX FIFO 读四笔数据，FIFO 深度减 4。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.2. Universal Asynchronous Receiver/Transmitter (UART)

UART 是一种与外围设备、数据载体设备或者数据集等进行串行通信的接口。主处理器 (CPU) 通过 APB 总线将数据写到 UART，数据转换成串行形式后传输到目标设备。UART 也可以接收串行数据和存储，以供主处理器 (CPU) 读取。

- UART 包含了控制以下功能的寄存器：
  - 字符长度
  - 波特率
  - 奇偶校验
  - 中断控制

    尽管 UART 只有一个中断输出信号，但有几种优先级的中断类型可响应，每种中断类型都可以通过控制寄存器单独使能或未使能。
  - 中断可在发送缓冲/ FIFO、接收缓冲/ FIFO、调制解调器状态以及线路状态范围内生成。
- UART 具有以下两种工作模式，可与一系列标准软件驱动程序兼容。
  - 16550 模式：数据的发送与接收都由 FIFO 缓存。
  - 16450 模式：禁用 FIFO。
- UART 支持 5–8 位的字长，一个可选的奇偶校验位，1/ 1.5/ 2 个停止位，可通过 CPU APB 接口编程，包含 16 位可编程波特率发生器和 8 位暂存器，以及独立的发送和接收 FIFO。提供八条调制解调器控制线和调试回环模式。

12.2.1. 特性说明

- 兼容工业标准 16550 UART
- 256 x 8bit 发送与接收 FIFO
- 传输速度可达 3 Mbps
- 支持 5–8 数据位以及 1/ 1.5/ 2 停止位
- 支持奇校验，偶校验或者无奇偶校验
- 支持 DMA 控制器接口
- 支持软件/硬件流控
- 支持 9–bit RS–485 模式
- 支持 RS–485 硬件自动控制收发方向

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.2.2. 原理框图

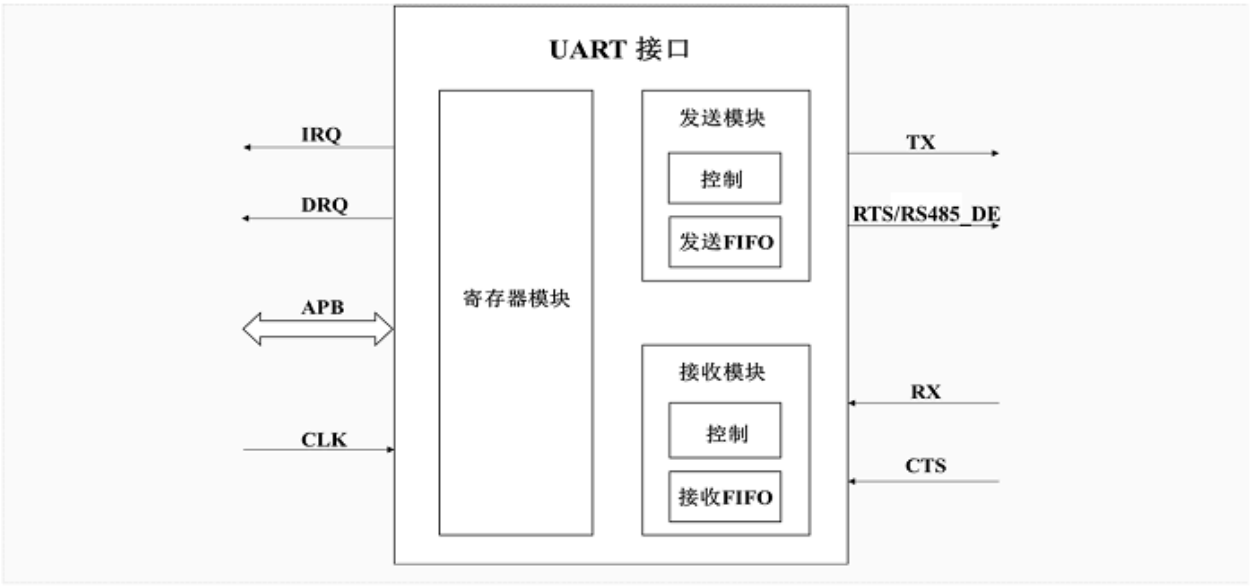


图 12–5 UART 原理框图

12.2.3. 功能描述

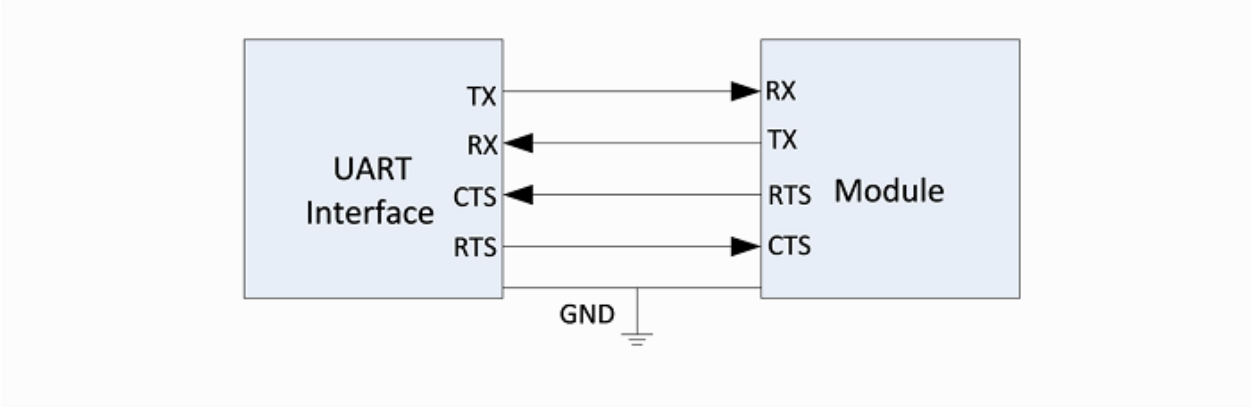
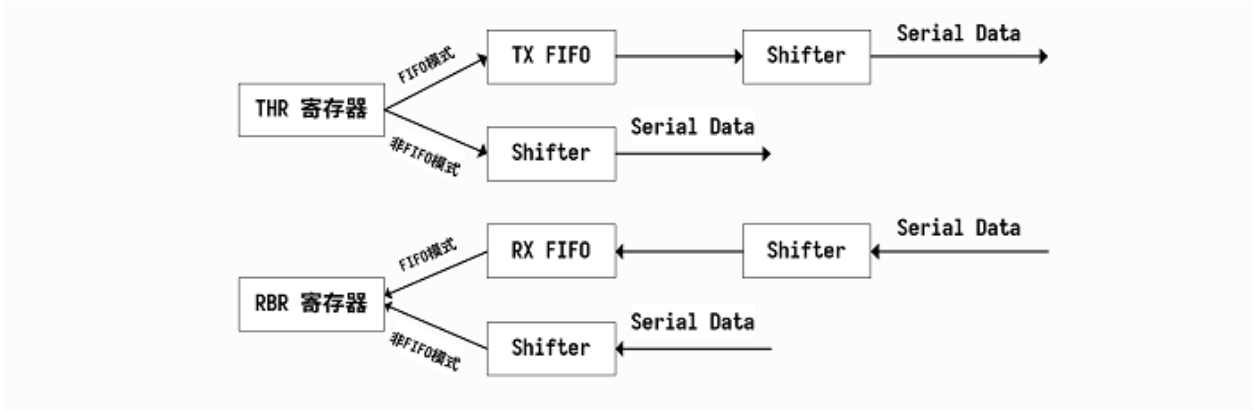


图 12–6 UART 典型应用

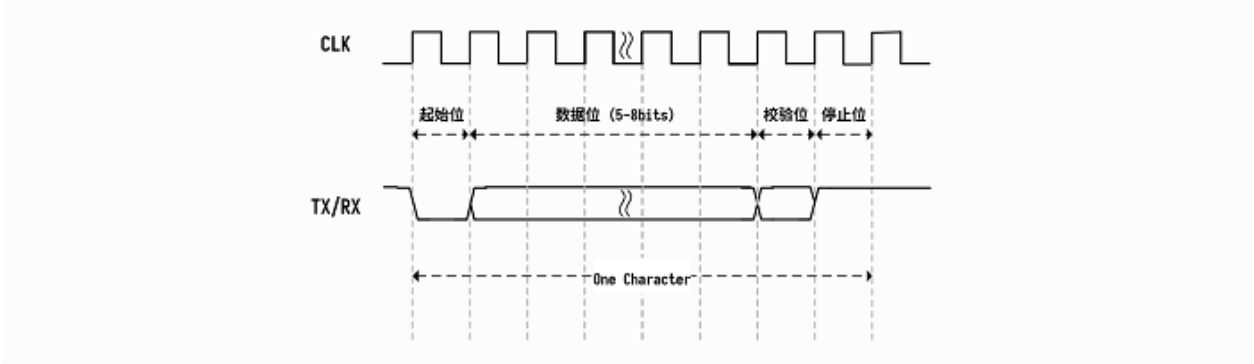
12.2.3.1. 功能实现

UART 数据流向：

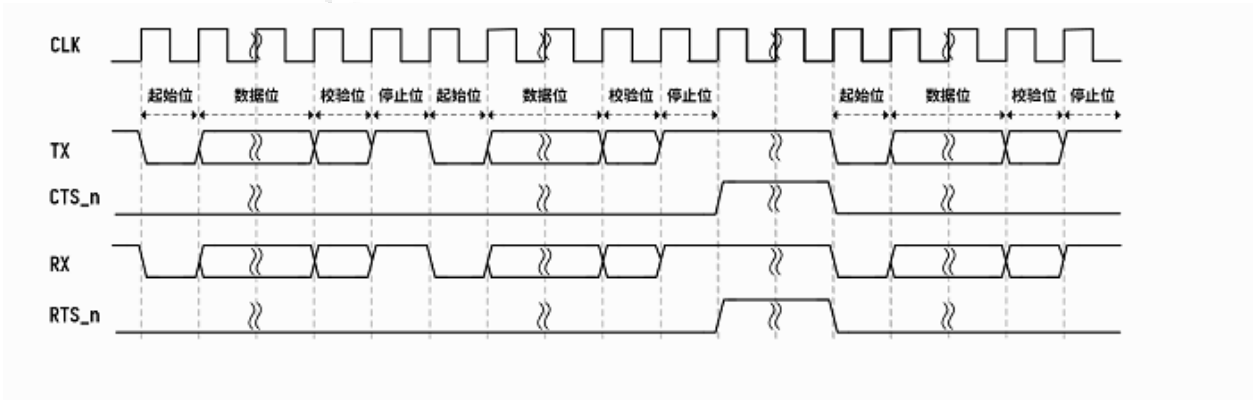
Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



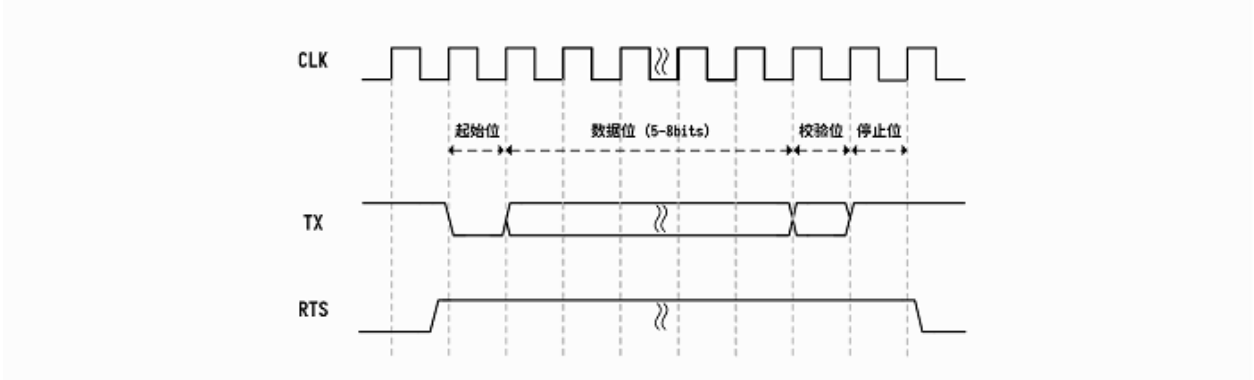
UART 串行数据格式：



UART 自动流控制时序：



UART RS-485 时序



Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

UART 电气特性

UART 接收时序：

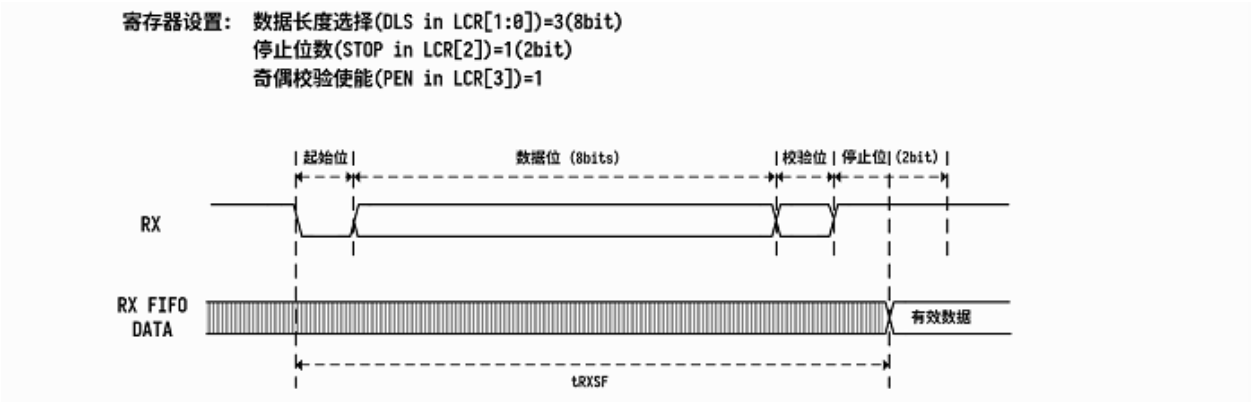


表 12–5 UART 接收时序参数

Parameter	Symbol	Min	Type	Max	Unit
RX start to RX FIFO	tRXSF	10.5 × BRP (1)	–	11 × BRP	ns

(1) BRP: Baud–Rate Period，波特率周期

UART nCTS 时序：

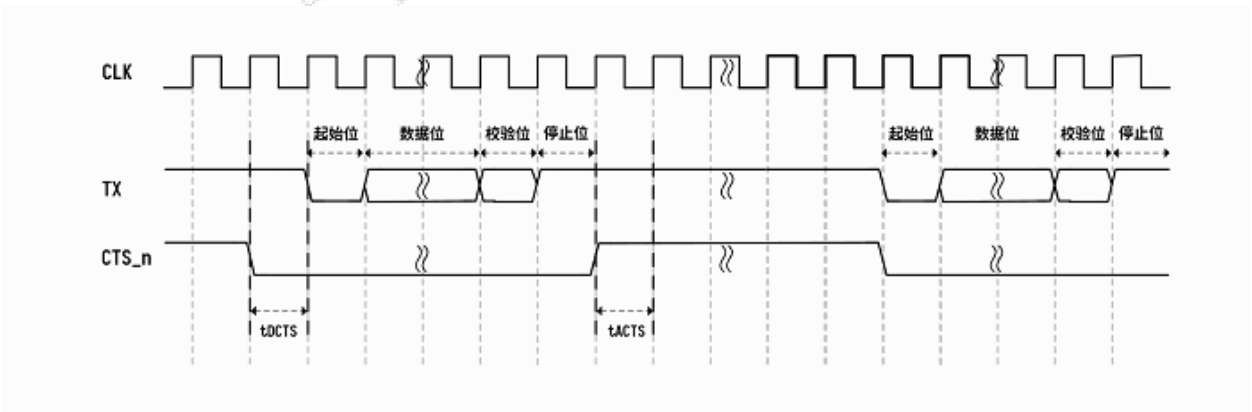


表 12–6 UART nCTS 时序参数

Parameter	Symbol	Min	Type	Max	Unit
Delay time of de–asserted nCTS to TX strat	tDCTS	–	–	BRP	ns
Step time of asserted nCTS to stop next transmission	tACTS	BRP /4	–	–	ns

UART nRTS 时序：当接收 FIFO 快满时 (FIFO 中的数据量 ≥ FIFO Depth – 2)，RTS_n 拉高。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

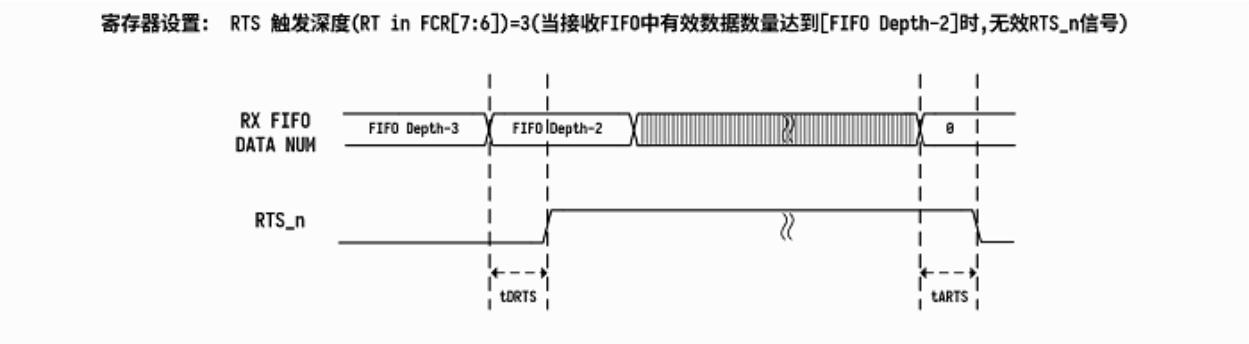


表 12–7 UART nRTS 时序参数

Parameter	Symbol	Min	Type	Max	Unit
Delay time of de–asserted nRTS	tDRTS	–	–	BRP	ns
Delay time of asserted nRTS	tARTS	–	–	BRP	ns

12.2.3.2. 操作模式

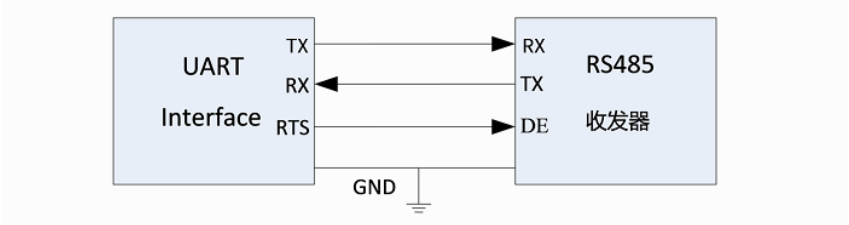
用户可以在 UART_MCR 寄存器中的 UART_FUNCTION (UART_MCR[7:6]) 位选择对应的操作模式。操作模式选择位可配置值及对应模式的描述如下：

- 0：基础 UART 模式，默认值。

  关于基础模式的详细配置，可查看 设置基础模式。

- 2：RS485 模式。

  即使用 UART–TX (RS485–TX)、UART–RX (RS485–RX) 和 UART–RTS (RS485–DE) 三线制通信。RS485 模式，三线制典型应用：



- 3：RS485–2 模式。

  即使用 UART–TX (RS485–TX/RX) 和 UART–RX (RS485–DE) 两线制通信。RS485 为半双工通信，同一时段只能发送或接收，即使用两线制实现 RS485 通信，详细描述可见 RS485 半双工控制。

12.2.3.2.1. 设置基础模式

如在UART_MCR 寄存器中的 UART_FUNCTION (UART_MCR[7:6]) 位选择了基础模式，可以在 UART_LCR 寄存器中配置 UART 控制器的基本功能。可配置的基本参数有：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 数据位宽 5–8 位
- 停止位个数 1/ 1.5/ 2
- 奇偶校验类型

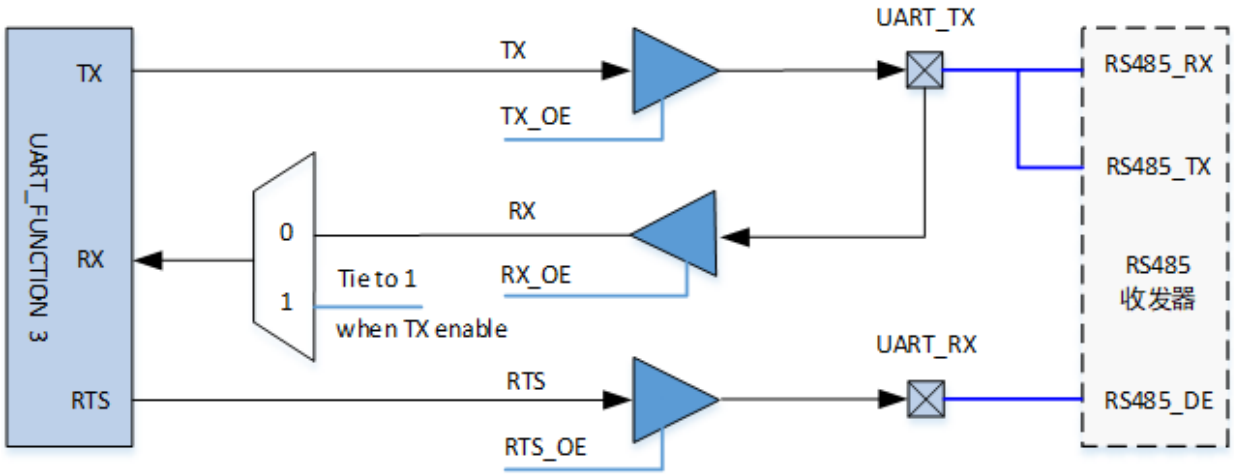
在基础模式下，UART 的一次帧传输主要包括起始信号、数据、校验位和结束信号，低位先传输 LSB。

- 起始信号 Start Bit：一个数据帧开始的标志，UART 协议规定 TX 信号出现一个低电平表示一个数据帧的开始。当 UART 没有传输数据时，保持高电平。
- 数据信号 Data Bit：数据宽度根据不同应用可进行调整，可以配置为 5/ 6/ 7/ 8 bits 数据位宽。
- 校验位 Parity Bit：校验位是 1 bit 纠错信号。校验位包括奇校验、偶校验，同时支持校验位的使能和禁止，可以通过 UART_LCR 寄存器设置。
- 结束信号 Stop Bit：数据位的停止位，支持 1/ 1.5/ 2 bit 停止位，可以通过 UART_LCR 寄存器设置。数据帧的结束信号就是把 TX 拉成高电平。

12.2.3.2.2. RS485 半双工控制

RS485 半双工控制即 RS485–2 线模式，可在 UART_MCR 寄存器中的 UART_FUNCTION (UART_MCR[7:6]) 功能选择位进行配置。

RS485–2 线模式原理框图如下：

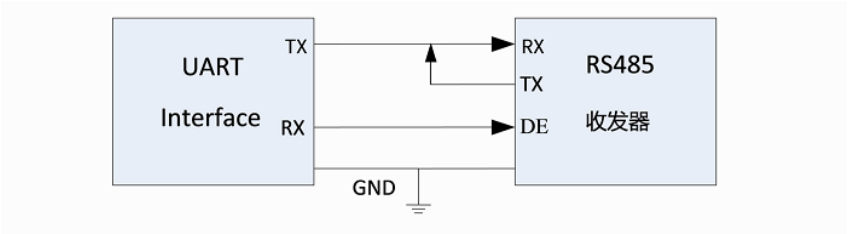


在 RS485–2 线模式中：

- UART–TX 既可以发送数据也可以接收数据。
- UART–RX 在模块内部与 UART–RTS 复用，用于控制 DE 使能发送。
- 当进入 RS485–2 模式，RTS 复用至 UART_RX PIN 脚，TX 和 RX 共用 UART_TX PIN 脚。TX 发送使能时，RX 拉高进入空闲状态且 RX 接收通路关闭。TX 发送使能关闭后，开启 RX 接收通路。数据收发处理逻辑与常规 RS485 模式类似。

RS485 模式的两线制典型应用如下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



RS485–2 线模式相关信号的控制如下所示：

- RX 在数据发送时输出高电平：在软件控制模式下，通过配置 MCR[1]=0，DE (rts_n) 输出高电平。在硬件控制模式下，当数据发送正在进行时，硬件自动控制 RX 输出高电平。

> **注：**
>
> 若正在接收数据，必须等待当前数据接收完成。

- RX 在数据接收时输出低电平：在软件控制模式下，通过配置 MCR[1]=1，DE (rts_n) 输出低电平。在硬件控制模式下，当发送 FIFO 为空时，硬件自动控制 RX 输出低电平。

> **注：**
>
> 若正在发送数据，必须等待当前数据发送完成。

- UART_485_CTL[7]（RS485_CTL_MODE，RS485 控制模式）
  - RS485_CTL_MODE = 0：硬件控制模式。DE 信号 (RTS) 在数据发送时通过内部硬件逻辑进行自动控制，发送数据前自动使能 DE 为有效高电平，当发送移位寄存器为空且发送 FIFO 为空，则自动切换 DE 为无效低电平。
  - RS485_CTL_MODE = 1：软件控制模式。DE 信号 (RTS) 在数据发送时通过外部软件配置进行手动控制，配置 MCR[1]=0 则 DE 为有效高电平可发送数据，配置 MCR[1]=1 则 DE 为无效低电平可接收数据。
- UART_485_DE[3:0]（DE_AT，DE 有效时间 t1）
  - DE 信号上升沿到 TX 起始位之间的时间间隔，以串行时钟 (sclk) 周期为单位。
- UART_485_DE[7:4]（DE_DAT，DE 无效时间 t2）
  - TX 结束位到 DE 信号下降沿之间的时间间隔，以串行时钟 (sclk) 周期为单位。

12.2.3.3. 设置 CHCFG_AT_BUSY

CHCFG_AT_BUSY：忙时配置使能控制位。CHCFG_AT_BUSY 使能后，当 UART 控制器为 BUSY 时，软件也能设置 UART 控制器，如寄存器 LCR（包括 DLAB 控制位）、DLH、DLL 等。

CHANGE_UPDATE：更新配置控制位。若忙时配置被使能，当对此位写 1，即可更新对 UART 控制器的配置。更新完成后，此位自动清零。

12.2.3.4. 设置 DLAB

DLAB 控制位，即 UART_LCR[7]，是分频系数寄存器的访问控制位。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 若 DLAB 为 0，偏移地址 0x00 为 TX 和 RX FIFO 寄存器，偏移地址 0x04 为 IER 寄存器。
- 若 DLAB 为 1，偏移地址 0x00 为 DLL 寄存器，偏移地址 0x04 为 DLH 寄存器。

UART 模块初始化时，如需设置分频系数，先要对 DLAB 写 1，才能访问寄存器 DLL 和 DLH。设置完成后，需对 DLAB 写 0，才能访问寄存器 TX、RX FIFO 以及 IER。

12.2.3.5. 设置波特率

波特率计算公式：Baudrate = SCLK / (CMU_divisor * 16 * UART_divisor)，其中

- SCLK：时钟源，每个 UART 通道可单独在 CMU 配置各自时钟。
- CMU_divisor：CMU 模块的分频系数。
- UART_divisor ：UART 模块的分频系数，共有 16 位。

  寄存器 UART_DLL 为低 8 位，寄存器 UART_DLH 为高 8 位。

关于时钟源在不同频率下波特率的误差说明，详见Clock Manage Unit (CMU) 功能描述。

12.2.3.6. 设置 UART 繁忙

BUSY 标志位，即 UART_USR[0]，用于表示 UART 控制器是否繁忙。BUSY 标志位配置值如下：

- 0：空闲
- 1：繁忙

以下情况，BUSY 位将会自动置 1，表示 UART 控制器繁忙：

- TX 正在发送数据
- RX 正在接收数据
- TX FIFO 非空
- RX FIFO 非空

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.2.4. 编程指南

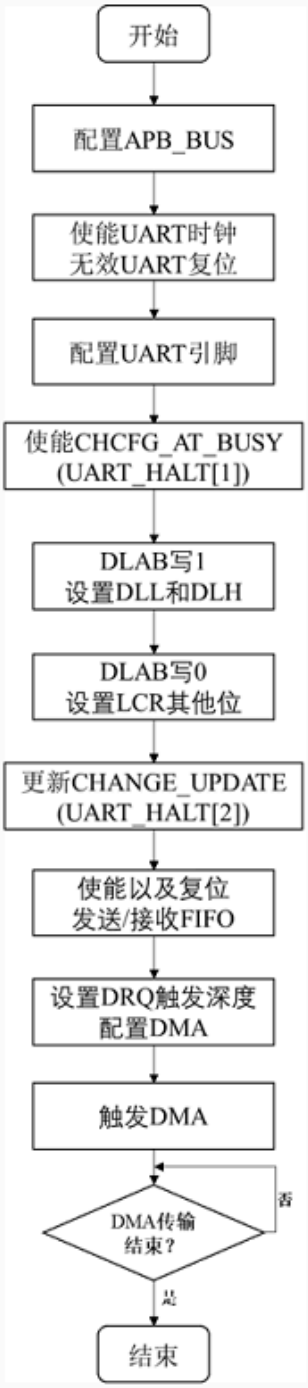


图 12–7 UART DRQ 流程

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

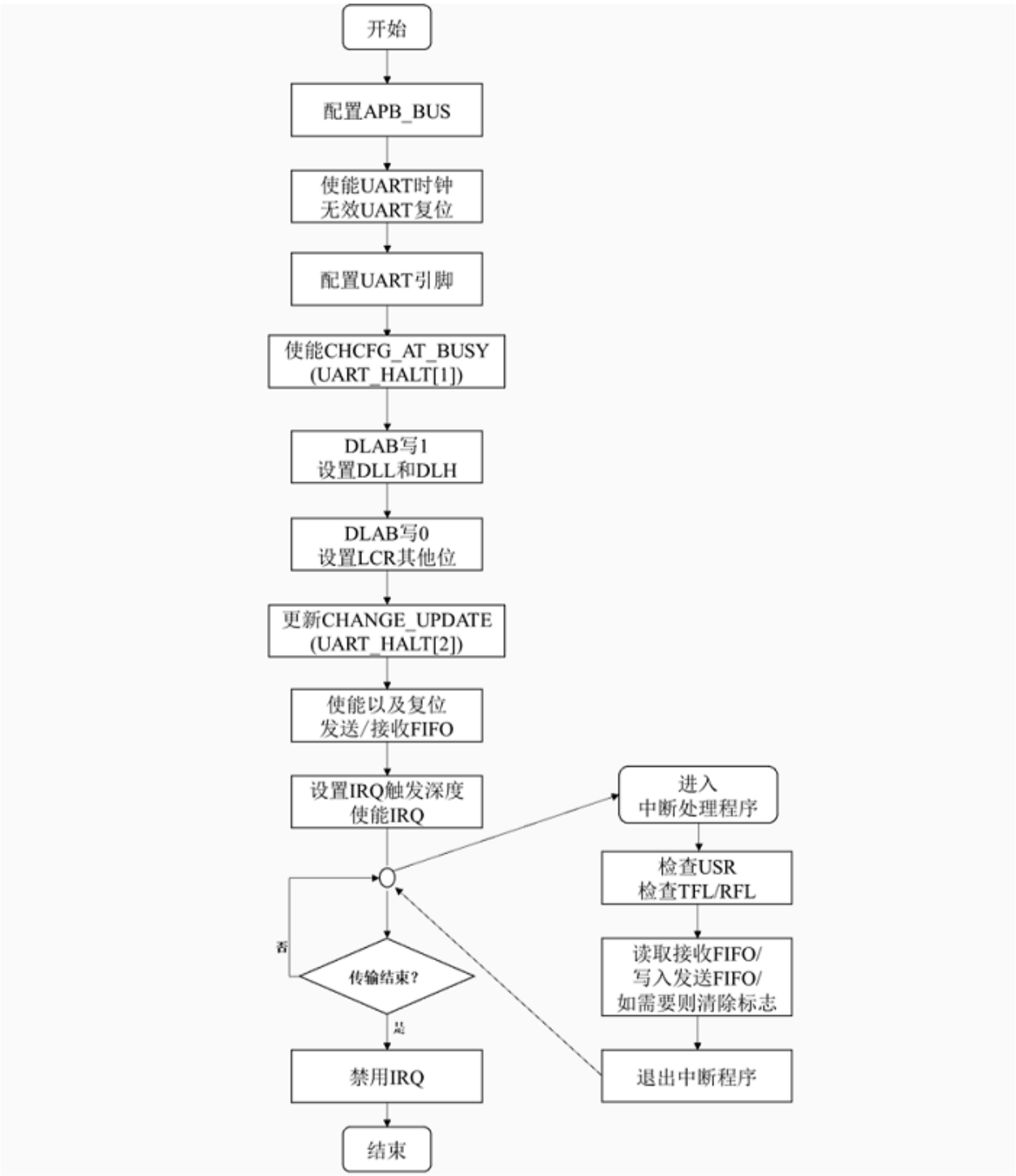


图 12–8 UART IRQ 流程

12.2.5. 寄存器列表

表 12–8 UART 寄存器列表

偏移地址	缩写	寄存器名称	章节
0x000	UART_RBR	UART 数据接收寄存器	0x000 UART_RBR

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 12–8 UART 寄存器列表 (续)

偏移地址	缩写	寄存器名称	章节
0x000	UART_THR	UART 数据发送寄存器	0x000 UART_THR
0x000	UART_DLL	UART 低位除数寄存器	0x000 UART_DLL
0x004	UART_DLH	UART 高位除数寄存器	0x004 UART_DLH
0x004	UART_IER	UART 中断使能寄存器	0x004 UART_IER
0x008	UART_IIR	UART 中断标志寄存器	0x008 UART_IIR
0x008	UART_FCR	UART FIFO 控制寄存器	0x008 UART_FCR
0x00C	UART_LCR	UART 线路控制寄存器	0x00C UART_LCR
0x010	UART_MCR	UART 调制解调器控制寄存器	0x010 UART_MCR
0x014	UART_LSR	UART 线路状态寄存器	0x014 UART_LSR
0x018	UART_MSR	UART 调制解调器状态寄存器	0x018 UART_MSR
0x01C	UART_SCH	UART 暂存器	0x01C UART_SCH
0x07C	UART_USR	UART 状态寄存器	0x07C UART_USR
0x080	UART_TFL	UART 发送 FIFO 深度寄存器	0x080 UART_TFL
0x084	UART_RFL	UART 接收 FIFO 深度寄存器	0x084 UART_RFL
0x088	UART_HSK	UART DMA 握手配置寄存器	0x088 UART_HSK
0x0A4	UART_HALT	UART 停止发送寄存器	0x0A4 UART_HALT
0x0B0	UART_DBG_DLL	UART 低位除数调试寄存器	0x0B0 UART_DBG_DLL
0x0B4	UART_DBG_DLH	UART 高位除数调试寄存器	0x0B4 UART_DBG_DLH
0x0B8	UART_485_DE	UART RS485 DE 时间寄存器	0x0B8 UART_485_DE

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 12–8 UART 寄存器列表 (续)

偏移地址	缩写	寄存器名称	章节
0x0C0	UART_485_CTL	UART RS485 控制与状态寄存器	0x0C0 UART_485_CTL
0x0C4	RS485_ADDR_MATCH	UART RS485 地址匹配寄存器	0x0C4 RS485_ADDR_MATCH
0x0C8	BUS_IDLE_CHK	UART RS485 总线空闲查询寄存器	0x0C8 BUS_IDLE_CHK
0x0CC	TX_DLY	UART 发送延时寄存器	0x0CC TX_DLY
0x0FC	UART_VERSION	UART 版本号	0x0FC UART_VERSION

注：

UART 模块初始化时，如需设置分频系数，先要对 DLAB 写 1，才能访问寄存器 DLL 和 DLH。设置完成后，需对 DLAB 写 0，才能访问寄存器 TX、RX FIFO 以及 IER。

12.2.6. 寄存器描述

12.2.6.1. 0x000 UART_RBR

注：

UART 模块初始化时，如需设置分频系数，先要对 DLAB 写 1，才能访问寄存器 DLL 和 DLH。设置完成后，需对 DLAB 写 0，才能访问寄存器 TX、RX FIFO 以及 IER。

默认值：0x00000000			UART 数据接收寄存器 (Receive Buffer)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R	0x0	RBR Receive Buffer Register，数据接收寄存器 串行输入端口接收到的数据，仅当线路状态寄存器 (LSR) 中的数据就绪位 (DR bit) 被设为 1 时，该寄存器中的数据才有效。 在 FIFO 模式中（FCR[0] 设为 1）读取该寄存器则访问接收 FIFO 的顶部（先进先出）。如果接收 FIFO 已满，并且在下一个数据到达之前没有读取该寄存器，FIFO 中已经存在的数据将被保留，但任何传入的数据都会丢失并发生溢出错误。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.2.6.2. 0x000 UART_THR

注：

UART 模块初始化时，如需设置分频系数，先要对 DLAB 写 1，才能访问寄存器 DLL 和 DLH。设置完成后，需对 DLAB 写 0，才能访问寄存器 TX、RX FIFO 以及 IER。

默认值：0x00000000			UART 数据发送寄存器 (Transmit Holding)
位域	类型	默认值	描述
31:8	–	–	–
7:0	W	0x0	THR Transmit Holding Register，数据发送寄存器 串行输出端口发送的数据，仅当线路状态寄存器 (LSR) 中的 THRE 位 (LSR[5]) 被设为 1 时，数据才会写到该寄存器。 在 FIFO 模式中（FCR[0] 设为 1）并且 THRE 为 1 时，数据可以在 FIFO 满之前写入该寄存器。当 FIFO 已满时，任何写入数据的尝试都会导致写入数据丢失。

12.2.6.3. 0x000 UART_DLL

注：

UART 模块初始化时，如需设置分频系数，先要对 DLAB 写 1，才能访问寄存器 DLL 和 DLH。设置完成后，需对 DLAB 写 0，才能访问寄存器 TX、RX FIFO 以及 IER。

默认值：0x00000000			UART 低位除数寄存器 (Divisor Latch Low)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	DLL Divisor Latch Low，低位除数寄存器 16 位 UART 波特率除数的低 8 位。仅当 DLAB 位 (LCR[7]) 为 1 并且 UART 空闲 (USR[0]) 时，该寄存器才能被访问。 输出波特率等于串行时钟频率除以波特率除数的 16 倍，如下： Baudrate = SCLK / (CMU_divisor * 16 * UART_divisor) **注：** 当除数寄存器（DLL 与 DLH）设置为 0 时，波特时钟被禁用，并且没有串行通信发生。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.2.6.4. 0x004 UART_DLH

注：

UART 模块初始化时，如需设置分频系数，先要对 DLAB 写 1，才能访问寄存器 DLL 和 DLH。设置完成后，需对 DLAB 写 0，才能访问寄存器 TX、RX FIFO 以及 IER。

默认值：0x00000000			UART 高位除数寄存器 (Divisor Latch High)
位域	**类型**	**默认值**	**描述**
31:8	–	–	–
7:0	R/W	0x0	DLH Divisor Latch High，高位除数寄存器 16 位 UART 波特率除数的高 8 位。仅当 DLAB 位 (LCR[7]) 为 1 并且 UART 空闲 (USR[0]) 时，该寄存器才能被访问。 输出波特率等于串行时钟频率除以波特率除数的 16 倍，如下： Baudrate = SCLK / (CMU_divisor * 16 * UART_divisor) **注：** 当除数寄存器 (DLL 与 DLH) 设置为零时，波特时钟被禁用，并且没有串行通信发生。

12.2.6.5. 0x004 UART_IER

注：

UART 模块初始化时，如需设置分频系数，先要对 DLAB 写 1，才能访问寄存器 DLL 和 DLH。设置完成后，需对 DLAB 写 0，才能访问寄存器 TX、RX FIFO 以及 IER。

默认值：0x00000000			UART 中断使能寄存器 (Interrupt Enable)
位域	**类型**	**默认值**	**描述**
31:8	–	–	–
7	R/W	0x0	PTIME Programmable THRE Interrupt Mode Enable，THRE 中断模式使能 • 0：未使能。THRE 中断会在非空的条件清除。 • 1：使能。THRE 中断会在低于触发深度 FCR[5:4] 的条件清除。
6	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			UART 中断使能寄存器 (Interrupt Enable)
位域	**类型**	**默认值**	**描述**
5	R/W	0x0	Shifter_Reg_Empty_EN 发送移位寄存器为空中断使能，用于发送 FIFO 为空并且移位寄存器也为空的中断使能控制： • 0：未使能 • 1：使能
4	R/W	0x0	RS485_INT_EN RS485 Interrupt Enable，RS485 中断使能 • 0：未使能 • 1：使能
3	R/W	0x0	EDSSI Enable Modem Status Interrupt，调制解调器状态中断使能 用于调制解调器状态 (MSR) 中断使能控制，为第四优先级中断。 • 0：未使能 • 1：使能
2	R/W	0x0	ELSI Enable Receive Line Status Interrupt，接收线路状态中断使能 用于接收线路状态中断使能控制，为最高级中断。 • 0：未使能 • 1：使能
1	R/W	0x0	ETBEI Enable Transmit Holding Register Empty Interrupt，数据发送寄存器为空中断使能 用于数据发送寄存器为空中断使能控制，为第三优先级中断。 • 0：未使能 • 1：使能
0	R/W	0x0	ERBFI Enable Receive Data Available Interrupt，接收数据可用中断使能 用于接收数据可用中断使能控制，以及字符超时中断控制（如果在 FIFO 模式并且 FIFO 使能），为第二优先级中断。 • 0：未使能 • 1：使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.2.6.6. 0x008 UART_IIR

默认值：0x00000001			UART 中断标志寄存器 (Interrupt Identity)
位域	类型	默认值	描述
31:8	–	–	–
7:6	R	0x0	FEFLAG FIFOs Enable Flag，FIFOs 使能标志 表明 FIFO 是否使能。 • 0：未使能 • 3：使能
5	–	–	–
4	RC	0x0	Shifter_Reg_Empty_INT 发送移位寄存器为空中断状态 • 1：中断触发 • 0：中断未触发或被清除
3:0	R	0x1	IID Interrupt ID，中断 ID ，表示最高优先级的挂起中断，可以是以下类型之一： • 0x0：调制解调器状态 • 0x1：无中断 • 0x2：THR 为空 • 0x3：RS485 中断 • 0x4：接收数据可用 • 0x6：接收线路状态 • 0x7：繁忙检测 • 0xC：字符超时 **注：** bit3 为 FIFO 相关触发条件的中断标志位，仅在 FIFO 使能时有效。

中断 ID	优先级	中断类型	中断源	中断复位
0x1	–	–	–	–
0x6	最高级	接收线路状态	溢位/ 奇偶检验/ 帧错误或者 break 中断	读取线路状态寄存器。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

中断 ID	优先级	中断类型	中断源	中断复位
0x3	第二级	RS485 中断	在 RS485 AAD 模式中，接收地址值与 ADDR_MATCH 值匹配。	地址标志写 1 复位。
0x4	第三级	接收数据可用	接收数据可用（非 FIFO 模式或者 FIFO 未使能）或达到接收 FIFO 触发水平（FIFO 模式和 FIFO 使能)。	读取接收缓冲寄存器（非 FIFO 模式或者 FIFO 未使能）或 FIFO 低于触发水平（FIFO 模式和 FIFO 使能）。
0xC	第四级	字符超时	在四个字符时间内，接收 FIFO 没有字符进出，并且在此期间 FIFO 中至少有一个字符。	读取数据接收寄存器。
0x2	第五级	数据发送寄存器为空	数据发送寄存器为空（THRE 模式未使能）或发送 FIFO 处于或者低于阈值（THRE 模式使能）。	根据情况选择对应方式： • 如果为中断源，则读取中断标志寄存器。 • 未选择或者未使能 FIFO 或者 THRE 模式时，写数据进数据发送寄存器。 • 选择并使能 FIFO 和 THRE 模式时，发送 FIFO 高于阈值。
0x0	第六级	调制解调器状态	CTS 如果处于自动流控，CTS 不会触发。	读取调制解调器状态寄存器。
0x7	第七级	繁忙检测标志	主处理器在 UART 繁忙时（USR[0]设置为 1）尝试写入线路控制寄存器。	读取 UART 状态寄存器。

12.2.6.7. 0x008 UART_FCR

默认值：0x00000000			UART FIFO 控制寄存器 (FIFO Control)
位域	类型	默认值	描述
31:8	–	–	–
7:6	W	0x0	RT RX Trigger，接收触发，用于在接收 FIFO 中选择接收数据可用中断的触发深度。 在自动流控制模式中，它用于确定何时使 rts_n (Request to Send) 信号无效,也用于确定 DRQ 信号在某些操作模式下何时有效。 • 0：FIFO 中一个字符 • 1：四分之一 FIFO • 2：二分之一 FIFO • 3：接近 FIFO 满状态 (FIFO – 2)
5:4	W	0x0	TFT TX Empty Trigger，发送为空触发，用于选择产生发送保持寄存器为空中断的空阈值电平，也用于确定 DRQ 信号在某些操作模式中何时有效。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			UART FIFO 控制寄存器 (FIFO Control)
位域	类型	默认值	描述
			• 0：FIFO 为空 • 1：FIFO 中两个字符 • 2：四分之一 FIFO • 3：二分之一 FIFO
3	W	0x0	DMAM DMA Mode，DMA 模式 • 0：模式 0 ◦ 如果 PTE (HALT[7]) 为 1 并且发送 FIFO 使能，当 TFL（发送 FIFO 深度）小于或者等于 FIFO 触发深度时，将发送 TX DMA 请求。 ◦ 如果 PTE 为 1 并且发送 FIFO 未使能，当 THRE 为 0 时，将发送 TX DMA 请求。 ◦ 如果 PTE 为 0，当发送 FIFO 为空时，将发送 TX DMA 请求。 ◦ 如果 DMA_PTE_RX 为 1 并且接收 FIFO 使能，当 RFL（接收 FIFO 深度）等于或者大于 FIFO 触发深度时将发送 RX_DRQ。 • 1：模式 1 ◦ 如果 PTE 为 1 并且发送 FIFO 使能，当 TFL 小于或者等于 FIFO 触发深度时，将发送 TX DMA 请求。 ◦ 如果 PTE 为低，TX DMA 请求将在 TX FIFO 为空时发送，并且仅当 TX FIFO 已满时停止请求。 ◦ 如果 RFL 等于或者大于 FIFO 触发深度，RX_DRQ 将会设为 1，在其它情况则为 0。
2	W1C	0x0	TXFIFOR TX FIFO Reset，发送 FIFO 复位 复位发送 FIFO 的控制并且将 FIFO 清空。也可以清除 DMA 发送请求。此位不需要清除，为自动清除。
1	W1C	0x0	RXFIFOR RX FIFO Reset，接收 FIFO 复位 复位接收 FIFO 的控制并且将 FIFO 清空。也可以清除 DMA 接收请求。此位不需要清除，为自动清除。
0	W	0x0	FIFOE FIFO Enable，FIFO 使能 使能/未使能发送以及接收 FIFO。每当该位的值改变时，FIFO 的发送以及接收控制部分都会复位。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.2.6.8. 0x00C UART_LCR

默认值：0x00000000			UART 线路控制寄存器 (Line Control)
位域	类型	默认值	描述
31:8	–	–	–
7	R/W	0x0	DLAB Divisor Latch Access Bit，除数存取位，用于使能除数寄存器（DLL 和 DLH）的读写，以设置 UART 的波特率。 仅当 UART 空闲时（USR[0] 为 0）此位才可写。在初始化波特率设置后，必须清除此位，以便访问其它寄存器。 • 0：选择接收缓冲寄存器（RBR）、发送保持寄存器 (THR) 以及中断使能寄存器 (IER) • 1：选择低位除数寄存器 (DLL) 以及高位除数寄存器 (DLH)
6	R/W	0x0	BC Break Control Bit，中断信号控制位 产生一个中断条件传输到接收端。 若此位设 1，串行输出将强制为逻辑 0 状态。如果处于非回环模式中（取决于 MCR[4]），串行输出线强制为低电平，直到清除此位。
5:4	R/W	0x0	EPS Even Parity Select，偶校验选择 **注：** 仅当 PEN (LCR[3]) 设为 1 时才能使用偶校验选择。 仅当 UART 空闲时（USR[0] 为 0）此位可写。 当使能奇偶校验时（PEN 设为 1），此位用于选择奇性或者偶性校验。 • 0：奇性校验 • 1：偶性校验 • 2 或 3：反转 LCR[4] 并直接用于校验计算 对于 RS485 模式，数据包中的奇偶校验位（即第 9 位）用于指示该包是地址还是数据，而非奇偶校验： • 0 表示这是一个数据字节。 • 1 表示这是一个地址字节。
3	R/W	0x0	PEN Parity Enable，奇偶校验使能，用于选择是否使能发送接收串行字符中的奇偶校验生成与检测。 仅当 UART 空闲时（USR[0] 为 0）此位才可写。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			UART 线路控制寄存器 (Line Control)
位域	类型	默认值	描述
			• 0：奇偶检验未使能 • 1：奇偶检验使能
2	R/W	0x0	STOP Number of Stop Bits，停止位数，用于选择外围设备发送和接收的每个字符的停止位数。 仅当 UART 空闲时（USR[0] 为 0）此位才可写。 **注：** 不管选择多少个停止位，接收器只检查第一个停止位。 • 0：1 个停止位 • 1：当 DLS (LCR[1：0]) 为 0 时，1.5 个停止位，否则 2 个停止位。当设为 1 且数据位设为 5 位时 (LCR[1：0])，传输 1.5 个停止位，否则传输 2 位停止位。
1:0	R/W	0x0	DLS Data Length Select，数据长度选择，用于选择外围设备发送和接收的每个字符的数据位数。 仅当 UART 空闲时（USR[0] 为 0）此位才可写。可选择的位数如下： • 0：5 位 • 1：6 位 • 2：7 位 • 3：8 位

12.2.6.9. 0x010 UART_MCR

默认值：0x00000000			UART 调制解调器控制寄存器 (Modem Control)
位域	类型	默认值	描述
31:8	–	–	–
7:6	R/W	0x0	UART_FUNCTION 选择 UART 或者 RS485 • 0：UART 模式 • 2：RS485 模式（常规 485，3 线模式，TX、RX 和 RTS） • 3：RS485–2 线模式（特殊 485，半双工，2 线模式，TX、RX，其中 TX 做数据输入输出，RX 做控制。）
5	R/W	0x0	AFCE

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			UART 调制解调器控制寄存器 (Modem Control)
位域	**类型**	**默认值**	**描述**
			Auto Flow Control Enable，自动流控制使能 当使能 FIFO 以及此位设 1 时，自动流控制功能使能。 • 0：自动流控制模式未使能 • 1：自动流控制模式使能
4	R/W	0x0	LOOP Loop Back Mode，回环模式，用于将 UART 置于调试模式以进行测试。 • 0：正常模式 • 1：回环模式，调制解调器控制输入 (cts_n) 断开，输出 (rts_n) 在内部回环输入。 在 UART 模式下运行（MCR[7:6] 设为 0），tx 线上的数据保持为高，而串行数据输出在内部回环到 rx 线路，所有中断均正常。
3:2	–	–	–
1	R/W	0x0	RTS Request To Send，发送请求，用于直接控制发送请求 (rts_n) 输出。 • 0：rts_n 信号无效状态（高电平） • 1：rts_n 信号有效状态（低电平） rts_n 输出用于通知调制解调器 UART 已准备好交换数据。 • 当自动流控制未使能（MCR[5]设为 0），则需通过将此位设为 1，使 rts_n 处于低电平。 • 在自动流模式中（MCR[5] 设为 1）并且 FIFO 使能（FCR[0] 设为 1），也需通过将此位设为 1，使 rts_n 处于低电平，但同时也受到接收 FIFO 触发深度门控（当超过阈值时 n 处于高电平无效状态）。 • 当 MCR[1] 设为 0 时 rts_n 无效。 **注：** 在回环模式中（MCR[4] 设为 1），当该位置的值在内部回环输入时，rts_n 输出保持高电平无效状态。在 RS485 或 RS485–2 模式中，RTS 外接 485 收发器方向控制脚 DE，需通过设置此位控制发送接收方向，设 1 则为接收方向，设 0 则为发送方向。
0	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.2.6.10. 0x014 UART_LSR

默认值：0x00000060			UART 线路状态寄存器 (Line Status)
位域	类型	默认值	描述
31:8	–	–	–
7	R	0x0	FIFOERR RX Data Error In FIFO，FIFO 接收数据错误 • 如果 FIFO 未使能，此位总为 0。 • 如果 FIFO 使能且接收 FIFO 中至少有 PE(LSR[2])、FE(LSR[3]) 和 BI(LSR[4]) 中一个，此位被置 1。 • 如果 FIFO 没有后续错误，则通过读取 LSR 寄存器清除此位。
6	R	0x1	TEMT Transmitter Empty，发送器为空 • 如果 FIFO 未使能，每当数据发送寄存器以及发送移位寄存器为空时，该位设 1。 • 如果 FIFO 使能，每当发送 FIFO 以及发送移位寄存器为空时，该位设 1。 • 以上任意情况下，当一个字符被写入发送数据信道时，该位被清除。
5	R	0x1	THRE TX Holding Register Empty，数据发送寄存器为空 • 如果 FIFO 未使能，每当数据发送寄存器为空并且准备接收新数据时，该位被设为 1。CPU 写入数据发送寄存器时，该位被清除。 • 如果 FIFO 使能，每当发送 FIFO 为空时，该位被设为 1。当至少有一个字符写入发送 FIFO 时，该位被清除。
4	RC	0x0	BI Break Interrupt，Break 中断，用于指示在串行输入数据上检测到中断序列。 break 条件会导致 UART 接收到仅且一个由 0 组成的字符。 • 在 FIFO 模式中，与 break 条件相关联的字符通过 FIFO 传输，当字符位于 FIFO 顶部时此位置 1，读取清除此位。 • 在非 FIFO 模式中，此位立即置 1 并持续到读取此位清除。
3	RC	0x0	FE Framing Error，帧错误，用于指示接收器中出现帧错误。 当接收器在接收的数据中没有检测到有效的停止位时，就会发生帧错误。在 FIFO 模式中，由于帧错误与接收到的字符相关，因此当具有帧错误的字符位于 FIFO 顶部时，才会显示该错误。帧错误出现时，UART 会尝试重新同步。接收器重新同步是通过假设错误是由下一个字符的起始位引起的，然后继续接收另一个数据位，以及奇偶检验位和停止位。 需要注意的是，如果发生 BI，则此位也置 1。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000060			UART 线路状态寄存器 (Line Status)
位域	类型	默认值	描述
			• 0：无帧错误 • 1：帧错误，读取 LSR 清除此位
2	RC	0x0	PE Parity Error，奇偶校验错误 此位用于指示在接收中发生奇偶校验错误。 • 0：无奇偶校验错误 • 1：奇偶校验错误，读取 LSR 清除此位 **注：** 仅在使能奇偶校验 (PEN) 位 (LCR[3]) 功能时此位有效。
1	RC	0x0	OE Overrun Error，溢出错误 • 0：无溢出错误 • 1：溢出错误，读取 LSR 清除此位 如果在读取先前数据之前接收到新的数据字符，则出现此错误。 • 在非 FIFO 模式，当从 RBR 读取先前字符之前接收一个新字符，则此位被置 1，发生这种情况时，RBR 中的数据则会被覆盖。 • 在 FIFO 模式，当 FIFO 为满时接收到一个新的字符，则会发生此错误，此时 FIFO 中的数据被保留，而接收移位寄存器中的数据丢失。
0	R	0x0	DR Data Ready，数据就绪 用于指示在 RBR 或者接收 FIFO 中至少包含一个字符。 • 0：无数据就绪 • 1：数据就绪 当非 FIFO 模式中读取 RBR 或者 FIFO 模式中 FIFO 为空时清除此位。

12.2.6.11. 0x018 UART_MSR

默认值：0x00000010			UART 调制解调器状态寄存器 (Modem Status)
位域	类型	默认值	描述
31:5	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000010			UART 调制解调器状态寄存器 (Modem Status)
位域	类型	默认值	描述
4	R	0x1	CTS Line State Of Clear To Send，可发送状态 用于指示调制解调器控制线 cts_n 信号的当前状态。当 cts_n 有效时表示调制解调器已准备好 UART 交换数据。 • 0：cts_n 无效（高电平状态） • 1：cts_n 有效（低电平状态） **注：** 在回环模式中（MCR[4] 设为 1），CTS 内部连接 RTS (MCR[1])。
3:1	–	–	–
0	RC	0x0	DCTS Delta Clear To Send，表示自上次读取 MSR 后 cts_n 信号发生了改变。 • 0：上一次读取 MSR 后 cts_n 无变化 • 1：上一次读取 MSR 后 cts_n 有变化 读取 MSR 清除此位。在回环模式中 (MSR[4]=1)，DCTS 反映 MCR[1] (RTS) 的变化。 **注：** 如果未设置此位，并且 cts_n 信号有效（低电平）以及发生复位（软件复位或其它情况），如 cts_n 信号仍然保持有效，在移除复位时此位置 1。

12.2.6.12. 0x01C UART_SCH

默认值：0x00000000			UART 暂存器 (Scratch)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	SCRATCH_REG 暂存器，为用户提供临时存储空间，UART 中未定义用途。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.2.6.13. 0x07C UART_USR

默认值：0x00000006			UART 状态寄存器 (Status)
位域	类型	默认值	描述
31:5	–	–	–
4	R	0x0	RFF Receive FIFO Full，接收 FIFO 为满 用于指示接收 FIFO 已完全满。 • 0：接收 FIFO 未满 • 1：接收 FIFO 已满当接收 FIFO 不再满时自动清零
3	R	0x0	RFNE Receive FIFO Not Empty，接收 FIFO 非空 用于指示接收 FIFO 包含一个或多个数据。 • 0：接收 FIFO 为空 • 1：接收 FIFO 非空当接收 FIFO 为空时自动清零
2	R	0x1	TFE Transmit FIFO Empty，发送 FIFO 为空 用于指示发送 FIFO 完全空。 • 0：发送 FIFO 非空。 • 1：发送 FIFO 为空。当发送 FIFO 不再空时自动清零。
1	R	0x1	TFNF Transmit FIFO Not Full，发送 FIFO 未满 用于指示发送 FIFO 未满。 • 0：发送 FIFO 已满 • 1：发送 FIFO 未满。当发送 FIFO 已满时自动清零。
0	R	0x0	BUSY UART Busy Bit，UART 繁忙位 • 0：空闲 • 1：繁忙

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.2.6.14. 0x080 UART_TFL

默认值：0x00000000			UART 发送 FIFO 深度寄存器 (Transmit FIFO Level)
位域	类型	默认值	描述
31:9	–	–	–
8:0	R	0x0	TFL Transmit FIFO Level，发送 FIFO 深度 用于指示发送 FIFO 中的数据数量

12.2.6.15. 0x084 UART_RFL

默认值：0x00000000			UART 接收 FIFO 深度寄存器 (Receive FIFO Level)
位域	类型	默认值	描述
31:9	–	–	–
8:0	R	0x0	RFL Receive FIFO Level，接收 FIFO 深度 用于指示接收 FIFO 中的数据数量

12.2.6.16. 0x088 UART_HSK

默认值：0x000000A5			UART DMA 握手配置寄存器 (DMA Handshake Configuration)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0xA5	HSK Handshake Configuration，握手配置 • 0xA5：DMA wait cycle 模式 • 0xE5：DMA handshake 模式

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.2.6.17. 0x0A4 UART_HALT

默认值：0x00000000			UART 停止发送寄存器 (Halt TX)
位域	类型	默认值	描述
31:8	–	–	–
7	R/W	0x0	DMA_PTE_RX The Sending Of RX_DRQ，发送 RX_DRQ • 在 DMA1 模式中，当 RFL 大于或者等于触发深度/接收超时，发送 DRQ。 • 在 DMA0 模式中，如果 DMA_PTE_RX=1 并且 FIFO 使能，当 RFL 大于触发深度时，发送 DRQ。 • 在其它情况下，一旦接收数据有效，发送 DRQ。
6	R/W	0x0	PTE The Sending Of TX_REQ，发送 TX_REQ • 在 DMA1 模式（FIFO 使能）中： ◦ 如果 PTE 设为 1，当 TFL 小于触发深度时，发送 DMA 请求。 ◦ 如果 PTE 设为 0，当 FIFO 为空，发送 DMA 请求。当 FIFO 为满时，DMA 请求将停止。 • 在 DMA0 模式中： ◦ 如果 PTE 设为 1 并且 FIFO 使能，当 TFL 小于触发深度时，发送 DMA 请求。 ◦ 如果 PTE 设为 1 并且 FIFO 未使能，当 THRE 为空时，发送 DMA 请求。 ◦ 如果 PTE 设为 0，当 FIFO 为空时发送 DMA 请求。
5:1	–	–	–
0	R/W	0x0	HALT_TX 停止发送 此位用于停止传输以进行测试，以便在使能 FIFO 时，主控可以填充发送 FIFO。 • 0：停止发送未使能 • 1：停止发送使能 **注：** 如果 FIFO 未使能，设置此位对操作没有影响。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.2.6.18. 0x0B0 UART_DBG_DLL

默认值：0x00000000			UART 低位除数调试寄存器 (DBG DLL)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	DEBUG DLL

12.2.6.19. 0x0B4 UART_DBG_DLH

默认值：0x00000000			UART 高位除数调试寄存器 (DBG DLH)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	DEBUG DLH

12.2.6.20. 0x0B8 UART_485_DE

默认值：0x000000ff			UART RS485 DE 时间寄存器 (RS485 DE Time)
位域	类型	默认值	描述
31:8	–	–	–
7:4	R/W	0xf	DE_DAT Driver Enable De-assertion Time，DE 无效时间 T2 串行数据结束位到 DE 信号下降沿之间的时间间隔，以串行时钟周期为单位。
3:0	R/W	0xf	DE_AT Driver Enable Assertion Time，DE 有效时间 T1 DE 信号上升沿到串行数据起始位之间的时间间隔，以串行时钟周期为单位

12.2.6.21. 0x0C0 UART_485_CTL

默认值：0x00000080			UART RS485 控制与状态寄存器 (RS485 Control and Status)
位域	类型	默认值	描述
31:8	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000080			UART RS485 控制与状态寄存器 (RS485 Control and Status)
位域	类型	默认值	描述
7	R/W	0x1	RS485_CTL_MODE RS485 Control Mode，RS485 控制模式 • 0：硬件控制模式。DE 信号 (RTS) 在数据发送时通过内部硬件逻辑进行自动控制，发送数据前自动使能 DE 为有效高电平，发送移位寄存器为空且发送 FIFO 为空，则自动切换 DE 为无效低电平。 • 1：软件控制模式。DE 信号 (RTS) 在数据发送时通过外部软件配置进行手动控制，配置 MCR[1]=0 则 DE 为有效高电平可发送数据，MCR[1]=1 则 DE 为无效低电平可接收数据。
6	R/W1C	0x0	AAD_ADDR_F AAD 地址匹配标志 在 AAD 模式中，当 UART 接收到地址字节且与 RS485_ADDR_MATCH 相同时，该位将置为 1。如果 RS485 中断使能，则将产生 RS485 中断。 写 1 清除该位并且复位 RS485 中断。
5	R/W1C	0x0	RS485_ADDR_DET_F RS485 地址检测标志 此位为地址字节检测标志。当 UART 接收到地址字节时，此位将设为 1。如果 RS485 中断使能，则将产生 RS485 中断。 写 1 清除该位并且复位 RS485 中断。
4	–	–	–
3	R/W	0x0	RX_BF_ADDR 地址之前接收 在 NMM 模式中，如果此位设为 1，UART 将在接收到一个地址字节之前将所有字节接收到 FIFO 中。如果设为 0，则不进行此操作。 • 1：接收 • 0：不接收
2	R/W	0x0	RX_AF_ADDR 地址之后接收 • 1：接收 • 0：不接收 在 NMM 模式中： • 如果此位设为 1，UART 将在接收到一个地址字节之后将所有字节接收到 FIFO 中。 • 如果设为 0，则不进行此操作。
1:0	R/W	0x0	RS485_SLAVE_MODE_SEL

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000080			UART RS485 控制与状态寄存器 (RS485 Control and Status)
位域	类型	默认值	描述
			RS485 从机模式选择 RS485 从机模式： • 0：标准多点操作 (NMM) • 1：自动地址检测操作 (AAD) • 2：保留 • 3：保留

12.2.6.22. 0x0C4 RS485_ADDR_MATCH

默认值：0x00000000			UART RS485 地址匹配寄存器 (RS485 Address Match)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	ADDR_MATCH 地址匹配 在 AAD 模式下使用的匹配地址。 **注：** 仅在 AAD 模式下可用。

12.2.6.23. 0x0C8 BUS_IDLE_CHK

默认值：0x00000000			UART RS485 总线空闲查询寄存器 (RS485 Bus Idle Check)
位域	类型	默认值	描述
31:8	–	–	–
7	R/W	0x0	BUS_IDLE_CHK_EN 总线空闲查询使能 • 1：使能总线空闲查询功能。 • 0：未使能总线空闲查询功能
6	R	0x0	BUS_STATUS 总线状态标志

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			UART RS485 总线空闲查询寄存器 (RS485 Bus Idle Check)
位域	类型	默认值	描述
			• 1：繁忙 • 0：空闲
5:0	R	0x0	ADJ_TIME 总线空闲时间 单位为 8 × 16 × T_{clk}

12.2.6.24. 0x0CC TX_DLY

默认值：0x00000000			UART 发送延时寄存器 (TX Delay)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	DLY 发送延时 最后一个停止位与下一个开始位之间的延迟时间。单位为 16 × T_{clk}。它用于控制数据发送中两个字节之间的间隔。

12.2.6.25. 0x0D0 UART_DBR

默认值：0x00000000			UART 调试寄存器 (Debug Register)
位域	类型	默认值	描述
31:5	–	–	–
4	W	0x0	NOR_ACT_485_2WIRE • 0：普通模式 • 1：UART 2wire 模式（特殊 UART，2 线模式，TX/RX、RTS）
3	W	0x0	Bypass 预取调试
2:0	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.2.6.26. 0x0FC UART_VERSION

默认值：0x00000101			UART 版本号寄存器 (Version)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R	0x00000101	UART 版本号寄存器，V1.1

12.3. Inter Integrated Circuit (I2C)

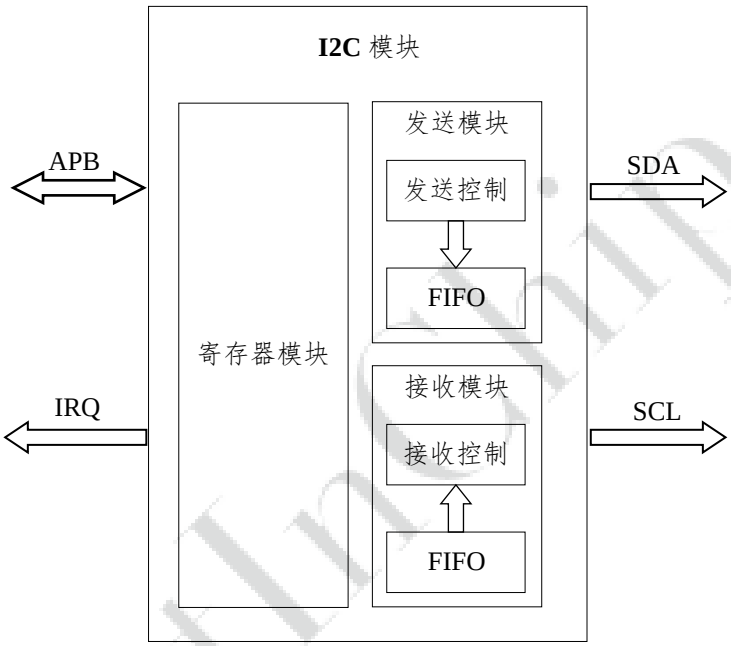
I2C 模块是一种用于连接 CPU 主机与 I^2C 总线之间的接口，包括一根串行数据线 (SDA) 和一根串行时钟线 (SCL)，分别用于传输数据和时钟信号，是实现 I2C 通信的基础。在 I^2C 总线上，每个设备都有一个唯一的地址。当主设备 (master) 希望与从设备 (slave) 通信时，会通过发送从设备 (slave) 的地址来建立连接。这种寻址方式允许在同一总线上连接多个设备，并进行精确的数据传输。主设备负责发起数据传输并生成时钟信号，而被主设备寻址的设备则作为从设备进行响应。这种主从关系确保了通信的有序性和有效性。此外，I2C 支持多种传输模式，以满足不同的应用需求。

12.3.1. 特性说明

- 支持主和从模式。
- 支持最高 400 Kb/s 传输。
- 支持 7 bits 和 10 bits 寻址。
- 支持时钟同步。
- 支持中断和轮询模式。
- 可编程 SDA 保持时间。
- 可编程防止 I^2C 通信异常总线挂死。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.3.2. 原理框图



12.3.3. 功能描述

12.3.3.1. 协议约定

I^2C 协议是一种应用广泛的串行通信协议，主要用于低速外设与处理器之间的数据传输。以下是对 I^2C 协议中一些关键特性和机制的详细解析：

表 12–9 协议约定

类型	描述
数据帧	大小固定为 8 bits（一字节）
传输方向	高位先发
总线释放状态	SCL 和 SDA 均为高电平，其中 SCL 一直由主设备控制。
启动信号（START）	SCL 高，SDA 由高转低
停止信号（STOP）	SCL 高，SDA 由低转高
应答信号	• ACK：主设备控制 SCL，发送完 8 bits 后释放 SDA，在第九个 SCL 波形时，从设备拉低 SDA。 • NACK：主设备控制 SCL，发送完 8 bits 后释放 SDA，在第九个 SCL 波形时，从设备不拉低 SDA。

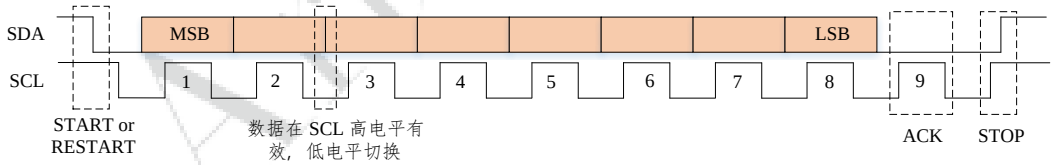
Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 12–9 协议约定 (续)

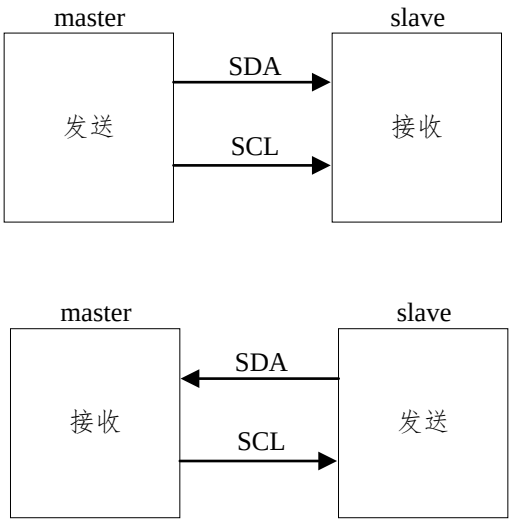
类型	描述
从设备地址	• 含 CPU 的智能器件，地址由软件初始化时定义，但不能与其它的器件有冲突。 • 不含 CPU 的非智能器件： ◦ 高 4 bits 由厂家在器件内部固化，不可改变。 ◦ 低 3 bits 由引脚连接电平决定，例如 eeprom，三个引脚常用名字 A1、A2、A3。
读写位	1 bit，以 7 位地址模式为例，与 从设备地址的 7 bits，组成读写数据帧的数据内容（从设备 7 bits + 1 bit），0 表示写，1 表示读。
空闲检测	主设备检测到 SDA、 SCL 均为高电平。

12.3.3.2. 主和从 模式

I^{2}C 协议：



主设备和从设备关系图：



主设备和从设备详细行为描述如下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 主设备发送操作：
  - 发起寻址：主设备通过 I2C_DATA_CMD[8] 写 0 进行发送请求。
  - 数据写入：主设备将数据写进 I2C_DATA_CMD[7:0]，然后发送到 I2C 总线。
  - 应答信号：响应的从设备从总线接收数据后，会向主设备发送 ACK 或 NACK 信号，以确认数据是否成功接收。
- 主设备接收操作：
  - 发起寻址：主设备通过 I2C_DATA_CMD[8] 写 1 进行接收请求。
  - 数据读取：响应的从设备将数据发送到 I2C 总线，由主设备从 I2C_DATA_CMD[7:0] 获取数据。
  - 应答信号：主设备在接收到数据后，会向从设备发送 ACK 或 NACK 信号。
- 从设备发送操作：
  - 接收请求：接收到主设备的接收请求后，从设备将数据写进 I2C_DATA_CMD[7:0]，然后发送到 I2C 总线。
  - 等待应答：从设备发送完数据后，等待主设备设备的应答信号。
- 从设备接收操作：
  - 接收数据：接收到主设备的发送请求后，从设备从 I2C_DATA_CMD[7:0] 获取数据。
  - 应答信号：从设备在接收数据后，向主设备发送 ACK 或 NACK 信号。
- 主设备发起寻址，发起 START 信号，生成 SCL，生成 STOP 信号，可发送和接收。从设备作为被寻址的一方，发起 ACK 或 NACK 信号，可发送和接收。

注：

I2C_DATA_CMD[8] 只能在主模式下配置，I2C 任何的通信行为都由主设备主动发起。

12.3.3.3. 7–bit & 10–bit 寻址

I²C 协议的寻址方式主要分为 7 位寻址和 10 位寻址，两种方式各有其特点和应用场景，以适应不同规模和复杂度的系统需求。了解寻址方式有助于正确配置和使用 I²C 设备，实现高效的数据传输和通信。

`7-bit` 寻址格式：允许主设备通过发送一个字节来寻址从设备，进行数据的读写操作。

在 7 位寻址模式下，I2C 设备的从机地址由 7 位组成，第 8 位是读写位（R/W），用于指示数据的方向：0 表示写操作，1 表示读操作。具体来说，I2C_TAR[6:0] 对应此 7 位地址，而 I2C_DATA_CMD[8] 则作为 R/W 位。

	MSB						LSB		
START	A6	A5	A4	A3	A2	A1	A0	R/W	ACK

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

注：

在某些情况下，从机设备可能支持 START 字节，即主设备先发起 0000 0001 字节（不需要 ACK），再发起 START 信号进行寻址。

10-bit 寻址格式：

在 10 位寻址模式下，寻址过程分为两个字节发送，确保了即使地址空间扩大到 10 位，也能保持与 7 位寻址的兼容性：

- 第一个字节：该字节的 bit[7:3] 向从设备指示这是一个 10 bit 的地址。紧接着的 bit[2:1]为 I2C_TAR[9:8]，而 R/W 为 I2C_DATA_CMD[8]。

  这个字节用于告诉从机接下来的是一个 10 位地址。

- 第二个字节：这个字节直接包含 I2C_TAR[7:0]，即从机的后 8 位地址。

START	1	1	1	1	0	A9	A8	R/W	A7	A6	A5	A4	A3	A2	A1	A0	ACK

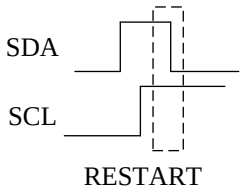
12.3.3.4. Restart 说明

在 I^2C 通信协议中，Restart（重复起始条件）是一种重要的信号，允许主设备在不产生停止条件的情况下继续与从设备进行通信。这种机制使得数据传输更加高效和连贯，尤其是在需要连续执行多次读写操作的场景中。

与起始条件类似，重复起始条件发生在停止条件之前。主设备在一个字节传输完成后，不产生停止条件而是发送重复起始条件来启动下一次传输，以便继续给从设备发送消息。

Restart 的时序要求与起始条件类似，都是在 SCL（串行时钟线）为高电平时，产生一个 SDA（串行数据线）的下降沿。具体来说，需要先令 SDA 和 SCL 都处于高电平状态，再令 SDA 拉低，以形成重复起始条件的时序。在 Restart 过程中，SDA 的下降沿必须在 SCL 为高电平时发生，这是判断是否为有效 Restart 的关键。

Restart 时序：

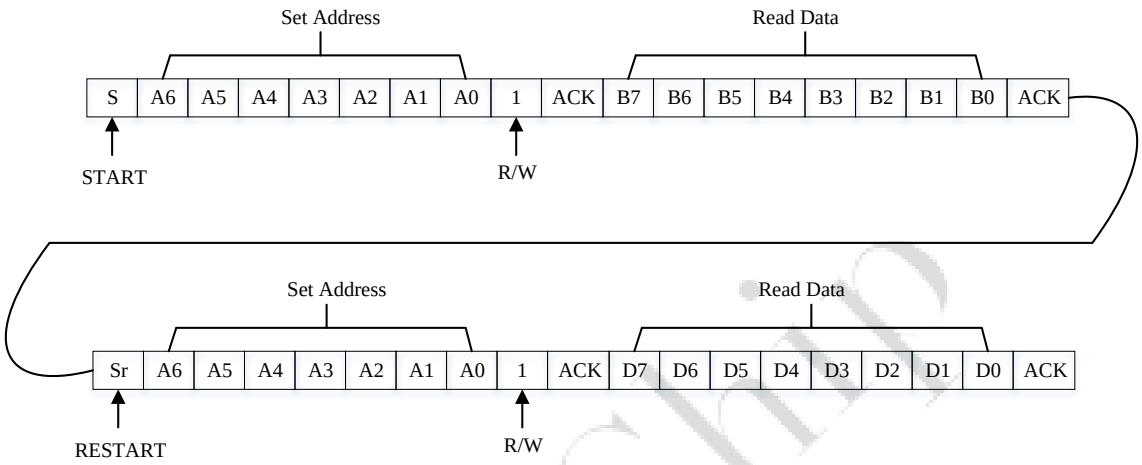


Restart 传输时序具有多种应用场景，示例如下：

多次读数据：

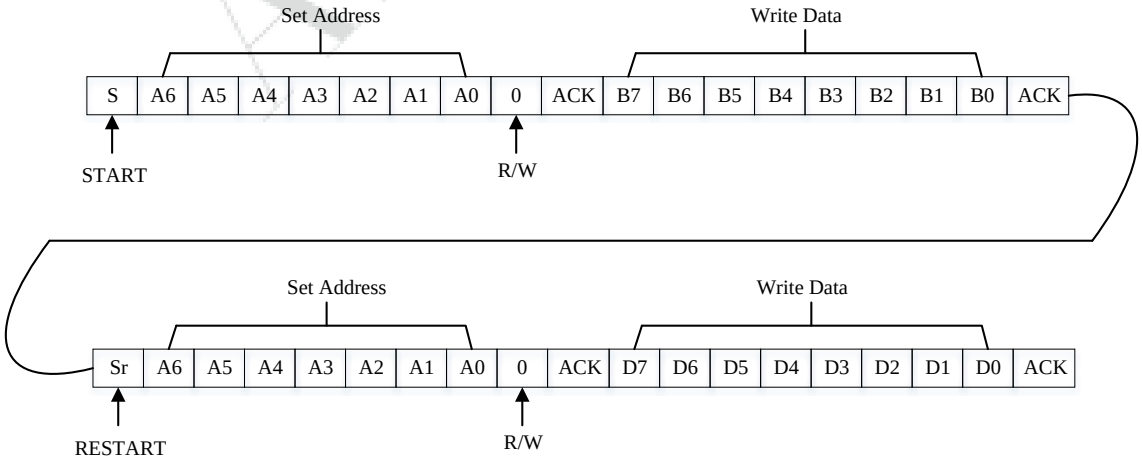
- 当主设备需要从从设备的多个寄存器中读取数据时，可以使用 Restart 来避免在每次读取之间产生停止条件。
- 主设备首先发送起始条件和从设备地址，然后发送要读取的第一个寄存器的地址。在接收到从设备返回的数据后，主设备不产生停止条件，而是直接发起 Restart，再次发送起始条件和从设备地址，以及下一个要读取的寄存器的地址。如此重复，直到所有需要的数据都被读取完毕。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



多次写数据：

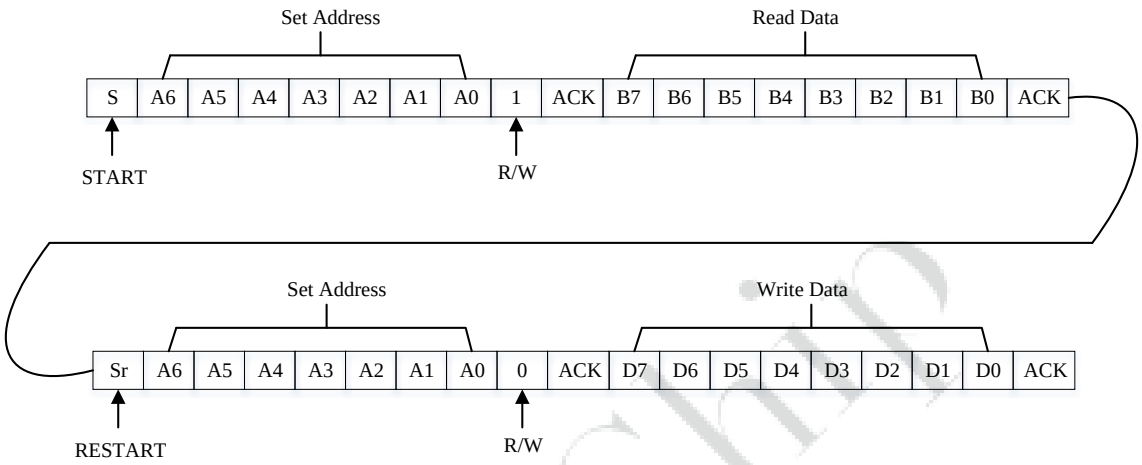
- 当主设备需要从从设备的多个寄存器中写入数据时，可以使用 Restart 来避免在每次写入之间产生停止条件。
- 主设备首先发送起始条件和从设备地址，然后发送要写入的第一个寄存器的地址和数据。在完成这次写入后，主设备不产生停止条件，而是直接发起 Restart，再次发送起始条件和从设备地址，以及下一个要写入的寄存器的地址和数据。如此重复，直到所有需要的数据都被写入完毕。



读数据转写数据：

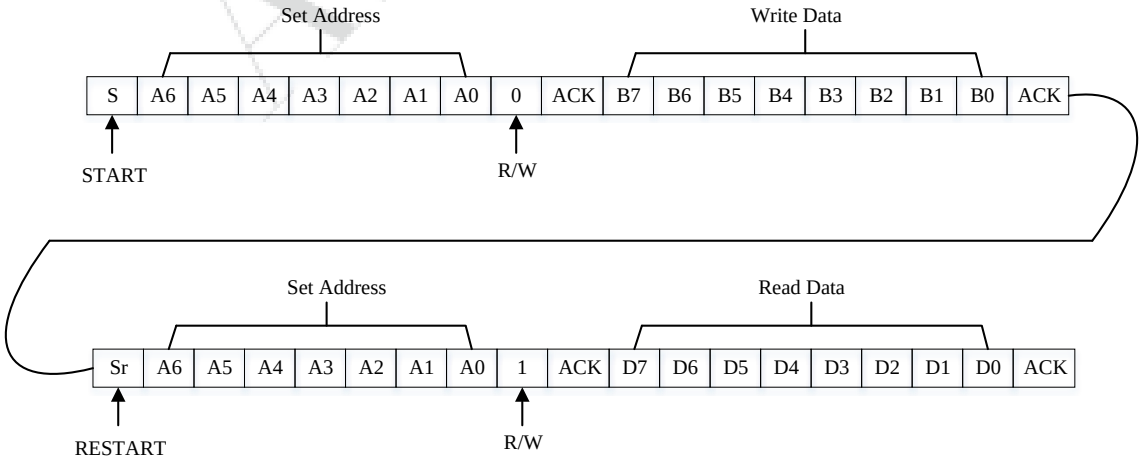
- 在某些情况下，主设备可能需要先从从设备读取数据，然后根据读取到的数据来决定是否需要写入新的数据。
- 主设备首先发送起始条件和从设备地址，然后发送要读取的寄存器的地址。在接收到从设备返回的数据后，主设备可以根据需要决定是否发起 Restart 并切换到写模式。如果需要写入新的数据，则主设备再次发送起始条件和从设备地址，以及要写入的寄存器的地址和数据。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



写数据转读数据：

- 某些情况下，主设备可能需要先向从设备写入数据，然后再读取从设备的状态或响应。
- 主设备首先发送起始条件和从设备地址，然后发送要写入的寄存器的地址和数据。在完成这次写入后，如果主设备需要读取从设备的状态或响应，则可以发起 Restart 并切换到读模式。具体来说，主设备再次发送起始条件和从设备地址，然后发送要读取的寄存器的地址。从设备会根据请求返回相应的数据或状态信息给主设备。



12.3.3.5. SDA_SETUP_TIME 和 SDA_HOLD_TIME 说明

在 I^2C 通信协议中，SDA_SETUP_TIME 和 SDA_HOLD_TIME 是两个至关重要的时序参数，确保了数据在传输过程中的准确性和稳定性。这两个参数主要涉及到 SDA（串行数据线）在 SCL（串行时钟线）上升沿或下降沿前后的状态保持时间。

- SDA_SETUP_TIME：SCL 延迟时间，表示 SDA 在 SCL 上升沿到来之前需要保持稳定的时间，仅当 I2C 作为 从设备发送数据时使用。

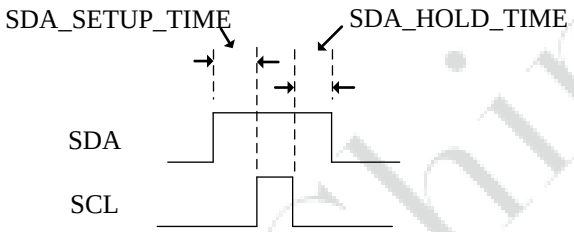
  SDA_SETUP_TIME 确保了接收设备能够在 SCL 上升沿到来之前正确地读取 SDA 上的数据。如果 SDA 在 SCL 上升沿到来之前未能稳定，接收设备可能会读取到错误的数据。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- SDA_HOLD_TIME：SDA 保持时间，表示 SDA 在 SCL 上升沿到来之后需要保持稳定的时间。

  SDA_HOLD_TIME 确保了接收设备在 SCL 上升沿到来之后有足够的时间来读取并处理 SDA 上的数据。如果 SDA 在 SCL 上升沿到来之后未能保持稳定，接收设备可能无法正确读取数据。

SETUP 和 HOLD 说明：



12.3.3.6. START BYTE 机制

START BYTE 机制是一种用于 I²C 通信中的技术，特别适用于片上系统没有专用的 I2C 模块的情况。这种机制通过延长开始条件的过程，使得处理器可以以较低的频率采样 SDA 线，从而降低功耗并提高系统效率。通过在生成 START 条件后，继续发送一个特定的字节信号 (0000_0001)，主设备持续的将 SDA 线拉低七个周期，使得处理器可以在低速下侦听总线，一旦检测到起始字节中的任一 0 位，便切换到高速模式进行后续的数据传输。

当 I2C 设为从设备时，无需使用 START BYTE 机制。

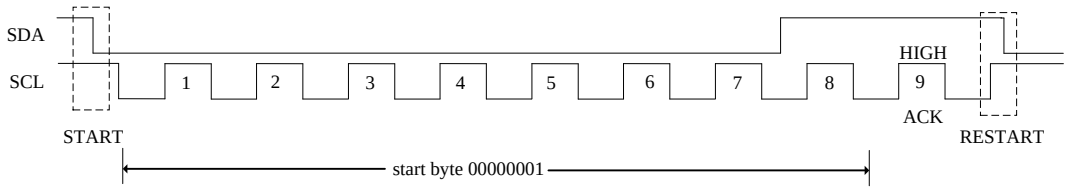
START BYTE 的过程：

1. 主设备先产生一个 START 条件。
2. 主设备 发送 byte（ 0000_0001 ），并发送 ACK 信号。

   主设备在整个字节发送期间保持 SDA 线为低电平。
3. 在此过程中，不能有任何从设备响应 ACK。
4. 主设备生成 restart 信号，标志着 START BYTE 机制的结束，随后进入正常的数据传输阶段。

START BYTE 时序要求如下：

- START 条件要求在 SCL 为高电平时，SDA 发生高到低的跳变；随后在 SCL 的第一个时钟周期内，SDA 必须保持低电平。
- 在发送特定字节信号时，主设备需确保在每个时钟周期内 SDA 线的正确电平状态，特别是要在整个字节期间保持 SDA 为低电平。
- RESTART 信号的产生时机是在特定字节信号发送完毕后，此时 SCL 应处于低电平状态，而 SDA 需由低电平跳变至高电平。



Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

在整个 START BYTE 过程中，主设备需要严格控制时钟信号（SCL）和数据信号（SDA）的配合，确保时序的准确性和稳定性。

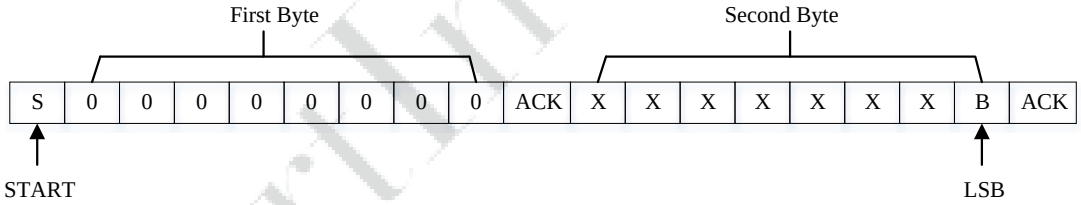
12.3.3.7. 通用广播地址

通用广播地址 (General call address) 是 I^2C 协议中的一种特殊机制，用于同时寻址总线上的所有设备。这种机制允许主设备通过发送特定的地址来广播消息或命令到所有连接在 I2C 总线上的从设备。

- 不需要使用广播功能的设备，可以不响应广播。
- 需要使用广播功能的设备，则在检测到广播地址后发送响应，并作为从设备接收读取总线上发送的数据。

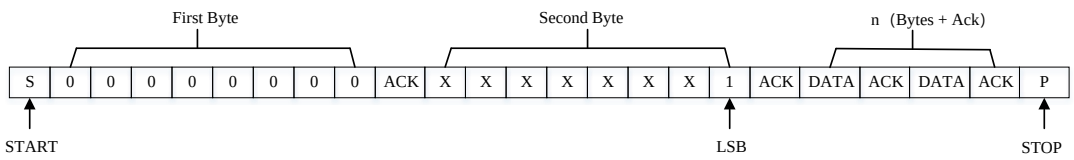
  主设备不知道总线上发送响应的从设备数量，总线上所有可以响应广播的从设备读取广播地址后的第二个以及后面的字节，不能处理广播数据的 从设备不发送响应来忽略广播。因此主设备检测不到未响应的从设备

广播信息的含义在第二个字节决定，如下所示：



GEN_CALL 格式有两种情况，详细说明如下：

1. 第二个 Byte 最低位 B 为 0：
   - 0000_0110(06h)：设备复位并通过硬件写数据到从设备的可编程部分。所有可以响应此类广播的从设备，收到此两个字节后，进行复位并进入其地址的可编程部分。
   - 0000_0100(04h)：通过硬件写内容到从设备的可编程部分，作用类似 (06h)，但设备不复位。
   - 0000_0000(00h)：不允许作为第二个字节使用。
2. 第二个 Byte 最低位 B 为 1：
   - 该两字节广播定义为硬件广播，由一个硬件 主设备器件发送，例如键盘扫描器。硬件广播不能编程，无法发送一个期望的从设备地址。由于硬件主设备预先不知道报文要传输给哪个器件，只能产生包含自身地址的硬件广播呼叫，便于系统识别。
   - 第二个字节中前 7 位是硬件主设备的地址，可由连接到总线的智能器件识别并指引硬件主设备的信息，例如，一个连接到总线的微控制器。如果硬件 主设备也可以作为从设备，其从设备地址和 主设备地址相同。
   - 硬件广播格式：



Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.3.3.8. 总线挂死恢复机制

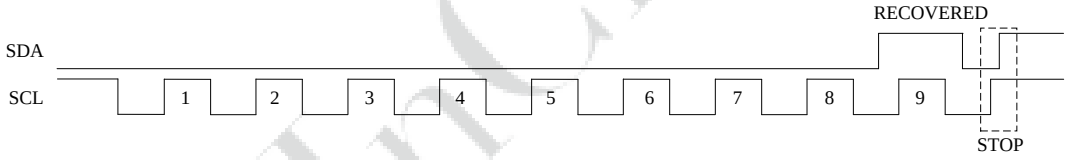
总线挂死恢复分为以下情况：

- **SDA 挂死恢复**：当检测到 SDA 挂死条件时，会触发恢复机制。

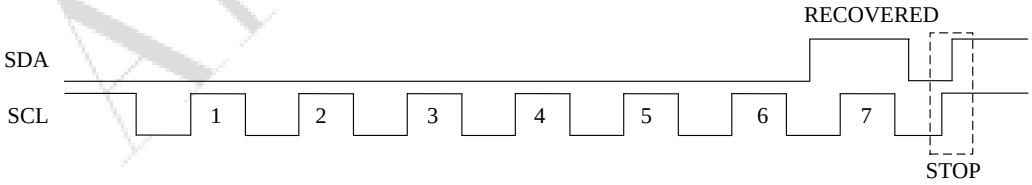
  I2C 通信异常可能会造成总线挂死情况，例如从设备应答 ACK 拉低 SDA 时出现异常。此时设备无法释放总线，造成主设备无法继续掌握总线进行通信。

  1. 主设备尝试在 SDA 发送逻辑 1 并且检查 SDA 是否恢复，最多连续发送 9 个时钟周期。
  2. 如果 SDA 在九个时钟周期内恢复，主设备则发送一个 STOP 信号释放总线。

     下图表示第九个时钟恢复：



     下图表示第七个时钟恢复



  3. 如果 SDA 在九个时钟周期内未恢复，则系统需要硬件复位。

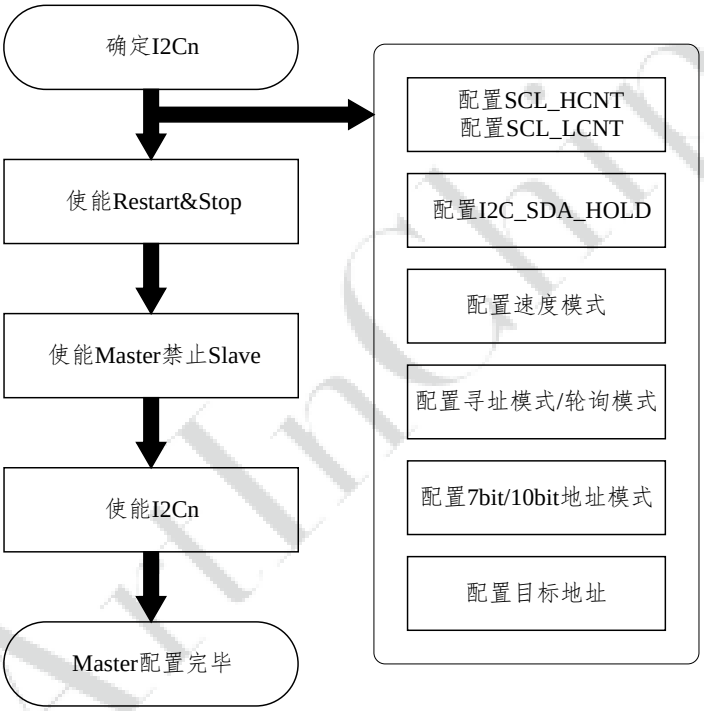
- **SCL 挂死恢复**

  通常只有在电路电气发生故障才存在 SCL 一直处于低电平，除了硬件复位系统外，没有有效的方法来解决此问题，若使能 SCL 挂死中断，可查看 I2C_INTR_STAT[14] 检查 SCL 状态。

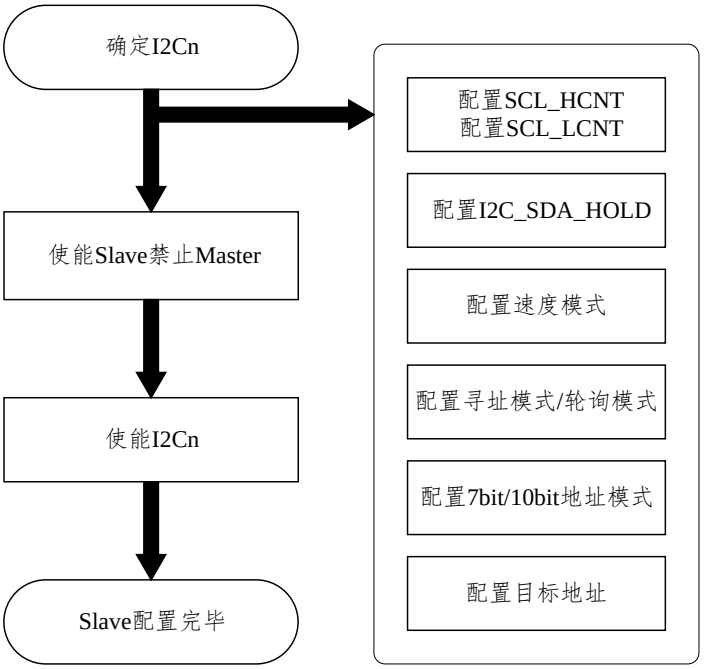
Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.3.4. 编程指南

12.3.4.1. 主模式初始化流程



12.3.4.2. 从模式初始化流程



Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.3.5. 寄存器列表

表 12–10 I2C 寄存器列表

偏移地址	缩写	寄存器名称	章节
0x000	I2C_CTL	I2C 控制	0x000 I2C_CTL
0x004	I2C_TAR	I2C 目标设备	0x004 I2C_TAR
0x008	I2C_SAR	I2C 从机地址	0x008 I2C_SAR
0x00C	I2C_ACK_GEN_CALL	I2C 响应广播	0x00C I2C_ACK_GEN_CALL
0x010	I2C_DATA_CMD	I2C 数据命令	0x010 I2C_DATA_CMD
0x020	I2C_SS_SCL_HCNT	I2C 标准速度时钟高位计数	0x020 I2C_SS_SCL_HCNT
0x024	I2C_SS_SCL_LCNT	I2C 标准速度时钟低位计数	0x024 I2C_SS_SCL_LCNT
0x028	I2C_FS_SCL_HCNT	I2C 快速时钟高位计数	0x028 I2C_FS_SCL_HCNT
0x02C	I2C_FS_SCL_LCNT	I2C 快速时钟低位计数	0x02C I2C_FS_SCL_LCNT
0x030	I2C_SDA_HOLD	I2C SDA 保持时间	0x030 I2C_SDA_HOLD
0x034	I2C_SDA_SETUP	I2C SDA 建立时间	0x034 I2C_SDA_SETUP
0x038	I2C_INTR_MASK	I2C 中断屏蔽	0x038 I2C_INTR_MASK
0x03C	I2C_INTR_CLR	I2C 中断清除	0x03C I2C_INTR_CLR
0x040	I2C_RAW_INTR_STAT	I2C 中断状态	0x040 I2C_RAW_INTR_STAT
0x048	I2C_ENABLE	I2C 使能	0x048 I2C_ENABLE
0x04C	I2C_ENABLE_STATUS	I2C 使能状态	0x04C I2C_ENABLE_STATUS
0x050	I2C_STATUS	I2C 状态	0x050 I2C_STATUS
0x054	I2C_TX_ABRT_SOURCE	I2C 发送停止源	0x054 I2C_TX_ABRT_SOURCE
0x090	I2C_RX_TL	I2C 接收阈值	0x090 I2C_RX_TL
0x094	I2C_TX_TL	I2C 发送阈值	0x094 I2C_TX_TL
0x098	I2C_TXFLR	I2C 发送 FIFO 水位	0x098 I2C_TXFLR

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 12–10 I2C 寄存器列表 (续)

偏移地址	缩写	寄存器名称	章节
0x09C	I2C_RXFLR	I2C 接收 FIFO 水位	0x09C I2C_RXFLR
0x0A0	I2C_SCL_STUCK_TIMEOUT	I2C SCL 挂死超时	0x0A0 I2C_SCL_STUCK_TIMEOUT
0x0A4	I2C_SDA_STUCK_TIMEOUT	I2C SDA 挂死超时	0x0A4 I2C_SDA_STUCK_TIMEOUT
0x0B0	I2C_FS_SPIKELEN	I2C 快速模式毛刺抑制时长	0x0B0 I2C_FS_SPIKELEN
0x0FC	VERSION	I2C 版本号	0x0FC VERSION

12.3.6. 寄存器描述

12.3.6.1. 0x000 I2C_CTL

默认值：0x0000006F			I2C 控制 （I2C Control）
位域	类型	默认值	描述
31:11	–	–	–
10	R/W	0x0	BUS_CLEAR_FEATURE_CTL 总线清除控制 • 0：BUS_CLEAR_FEATURE 禁止 • 1：BUS_CLEAR_FEATURE 使能 此位置 1，SDA 挂死机制使能以及 SCL_STUCK_AT_LOW 中断可触发 仅主模式有效
9	R/W	0x0	RX_FIFO_FULL_HLD_CTL RX_FIFO 满状态保持总线控制，仅在 主模式下： • 0：RX_FIFO 满状态时数据溢出 • 1：RX_FIFO 满状态时保持总线
8	R/W	0x0	TX_EMPTY_CTL TX_EMPTY 中断控制 • 0：TX_EMPTY 中断行为 0 • 1：TX_EMPTY 中断行为 1 （详细描述见 I2C_INTR_STAT[4]）

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x0000006F			I2C 控制 （I2C Control）
位域	类型	默认值	描述
7	R/W	0x0	STOP_DET_IFADDRESSED STOP_DET 中断控制，在从模式下： • 0：不管被寻址与否，都发出 STOP_DET 中断 • 1：仅在被寻址时发出 STOP_DET 中断 **注：** 此位设为 1 时，且在 GEN_CALL 下，即使响应也不会发出 STOP_DET 中断
6	R/W	0x1	RESTART_ENABLE 重开始使能控制 • 0：主模式 RESTART 禁止 • 1：主模式 RESTART 使能 主设备 发送 RESTART 信号控制，一些旧从设备不支持处理 RESTART 信号。当 RESTART 禁止时，主设备 被禁止执行以下功能： • 发送 START 字节 • 使用 10bit 寻址模式进行读操作
5:4	R/W	0x2	SPEED_MODE_SELECT 速率模式选择 • 1：标准模式 （ 100 Kb/s ） • 2：快速模式（ <=400 Kb/s ） 此位只能写 1 或者 2。
3	R/W	0x1	10BITADDR_SELECT_SLAVE 从模式被寻址 bit 选择 • 0：7 bits 寻址模式 • 1：10 bits 寻址模式
2	R/W	0x1	10BITADDR_SELECT_MASTER 主模式寻址 bit 选择 • 0：7 bits 寻址模式 • 1：10 bits 寻址模式
1	R/W	0x1	SLAVE_MODE_DISABLE 从模式使能控制

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x0000006F			I2C 控制 （I2C Control）
位域	类型	默认值	描述
			• 0：从模式使能 • 1：从模式禁止 当 I2C_CTL[0]设置为 1 时，此位也需设置为 1。I2C_CTL [0]设置为 0 时，此位也设置为 0。
0	R/W	0x1	MASTER_MODE_ENABLE 主模式使能控制 • 0：主模式禁止 • 1：主模式使能 当 I2C_CTL [1]设置为 1 时，此位也需设置为 1。I2C_CTL [1]设置为 0 时，此位也设置为 0。

12.3.6.2. 0x004 I2C_TAR

默认值：0x0000005A			I2C 目标设备 （I2C Target）
位域	类型	默认值	描述
31:12	–	–	–
11	R/W	0	GEN_CALL_CTL • 0：GEN_CALL 模式禁止 • 1：GEN_CALL 模式使能 此位配 1，当 bit[10]为 0 时，主设备进行广播模式寻址，无视 bit[9:0]中的目标地址，进行 GEN_CALL 模式，GEN_CALL 模式只允许进行写操作，尝试读操作会导致 TX_ABRT（I2C_INTR_STAT[6]）中断产生。当 bit[10]为 1 时，start 字节使能时，主设备先发起 start 字节，再进行目标寻址。
10	R/W	0	START_BYTE_CTL Start 字节控制 • 0：start 字节禁止 • 1：start 字节使能
9:0	R/W	0x5A	I2C_TAR 目标地址 主设备依此目标地址寻址建立通信。如果 I2C_TAR 和 I2C_SAR 相同，则存在回环，但是 FIFO 在主设备与从设备之间共享，因此双向传输是不可行的，只能单向传输。主设备不能传输数据给自己，只能传输给从设备。 当进行 GEN_CALL 时，无视此地址。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.3.6.3. 0x008 I2C_SAR

默认值：0x0000005A			I2C 从机地址（I2C Slave Address）
位域	类型	默认值	描述
31:10	–	–	–
9:0	R/W	0x5A	I2C_SAR 从模式被寻址的地址 当为 7 bit 寻址时，只有 bit[6:0]有效。此位仅在 I2C 禁止时可写（I2C_ENABLE[0]写 0），其它时间写入无效。

12.3.6.4. 0x00C I2C_ACK_GEN_CALL

默认值：0x00000001			I2C 响应广播（I2C Acknowledge General Call）
位域	类型	默认值	描述
31:1	–	–	–
0	R/W	0x1	ACK_GEN_CALL GEN_CALL 应答状态。 • 0：接收到 GEN_CALL 地址后不应答 • 1：接收到 GEN_CALL 地址后应答

12.3.6.5. 0x010 I2C_DATA_CMD

默认值：0x00000000			I2C 数据命令（I2C Data Command）
位域	类型	默认值	描述
31:11	–	–	–
10	W	0x0	RESTART 重开始行为控制 • 0：当传输方向与上一个传输命令的方向不同时。如果 RESTART_ENABLE(I2C_CTL[6]) 为 1，发出 RESTART 信号；如果 RESTART_ENABLE (I2C_CTL[6]) 为 0，则在发送一个 STOP 信号后接着发送一个 START 信号。 • 1：无论当前数据传输方向与上一个传输命令的方向是否相同。如果 RESTART_ENABLE(I2C_CTL[6]) 为 1，发出 RESTART 信号；如果 RESTART_ENABLE (I2C_CTL[6]) 为 0，则在发送一个 STOP 信号后接着发送一个 START 信号。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			I2C 数据命令（I2C Data Command）
位域	**类型**	**默认值**	**描述**
9	W	0x0	STOP STOP 命令控制 • 0：一个 byte 传输完后不接着 STOP 信号 • 1：一个 byte 传输完后接着 STOP 信号
8	W	0x0	CMD 操作命令，仅在主模式下有效 • 0：主模式写操作命令 • 1：主模式读操作命令 **注：** 如果在收到 RD_REQ 请求后此位写 1，则产生 TX_ABRT 中断。
7:0	R/W	0x0	DAT 数据缓冲区 发送数据时，数据需写入此位段。接收数据时，数据亦接收到此位段。

12.3.6.6. 0x020 I2C_SS_SCL_HCNT

默认值：0x00000060			I2C 标准速度时钟高位计数（I2C Standard SCL High Count）
位域	**类型**	**默认值**	**描述**
31:16	–	–	–
15:0	R/W	0x60	I2C_SS_SCL_HCNT 标准速率模式下 SCL 高电平持续时间计数。 此位仅在 I2C 禁止时可写（I2C_ENABLE[0] 写 0），其它时间写入无效。最小值为 0x6，写入小于 0x6 的数值无效。 计算方法： • 标准速率 SCL 高电平最小时间：4000 ns • 标准速率 SCL 低电平最小时间：4700 ns 以标准速度模式为例： • SCL 高电平时间 = (I2C_SS_SCL_HCNT + I2C_FS_SPIKELEN + 8) * I2C_clk_period • I2C_clk_period 为 I2C 模块输入时钟周期

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.3.6.7. 0x024 I2C_SS_SCL_LCNT

默认值：0x00000070			I2C 标准速度时钟低位计数（I2C Standard SCL Low Count）
位域	类型	默认值	描述
31:16	–	–	–
15:0	R/W	0x70	I2C_SS_SCL_LCNT 标准速率模式下 SCL 低电平持续时间计数。 此位仅在 I2C 禁止时可写（I2C_ENABLE[0] 写 0），其它时间写入无效。最小值为 0x8，写入小于 0x8 的数值无效。 计算方法： • 标准速率 SCL 高电平最小时间：4000 ns • 标准速率 SCL 低电平最小时间：4700 ns 以标准速度模式为例 • SCL 低电平时间 = (I2C_SS_SCL_LCNT + 1) * I2C_clk_period • I2C_clk_period 为 I2C 模块输入时钟周期。

12.3.6.8. 0x028 I2C_FS_SCL_HCNT

默认值：0x0000000F			I2C 快速时钟高位计数（I2C Fast SCL High Count）
位域	类型	默认值	描述
31:16	–	–	–
15:0	R/W	0xF	I2C_FS_SCL_HCNT 快速模式下 SCL 高电平持续时间计数。 此位仅在 I2C 禁止时可写（I2C_ENABLE[0]写 0），其它时间写入无效。最小值为 0x6，写入小于 0x6 的数值无效。 计算方法： • 快速 SCL 高电平最小时间：600ns • 快速 SCL 低电平最小时间：1300ns 以快速模式为例 • SCL 高电平时间 = (I2C_FS_SCL_HCNT + I2C_FS_SPIKELEN + 8) * I2C_clk_period • I2C_clk_period 为 I2C 模块输入时钟周期

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.3.6.9. 0x02C I2C_FS_SCL_LCNT

默认值：0x0000001F			I2C 快速时钟低位计数（I2C Fast SCL Low Count）
位域	类型	默认值	描述
31:16	–	–	–
15:0	R/W	0x1F	I2C_FS_SCL_LCNT 快速模式下 SCL 低电平持续时间计数。此位仅在 I2C 禁止时可写（I2C_ENABLE[0]写 0），其它时间写入无效。最小值为 0x8，写入小于 0x8 的数值无效。 计算方法： • 快速 SCL 高电平最小时间：600ns • 快速 SCL 低电平最小时间：1300ns 以快速模式为例 • SCL 低电平时间 = (I2C_FS_SCL_LCNT + 1) * I2C_clk_period • I2C_clk_period 为 I2C 模块输入时钟周期

12.3.6.10. 0x030 I2C_SDA_HOLD

默认值：0x00000001			I2C SDA 保持时间（I2C SDA Hold Time）
位域	类型	默认值	描述
31:24	–	–	–
23:16	R/W	0x0	I2C_SDA_RX_HOLD I2C 接收时设置所需的 SDA 保持时间，以 I2C_clk 周期为单位
15:0	R/W	0x1	I2C_SDA_TX_HOLD I2C 发送时设置所需的 SDA 保持时间，以 I2C_clk 周期为单位

12.3.6.11. 0x034 I2C_SDA_SETUP

默认值：0x00000002			I2C SDA 建立时间（I2C SDA Setup Time）
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x2	I2C_SDA_SETUP SDA 设置时间，数值为（该值–1），以 I2C_clk 周期为单位。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000002			I2C SDA 建立时间（I2C SDA Setup Time）
位域	类型	默认值	描述
			如默认值为 (2–1)个 cycle 建议如果需要延迟 1000 ns，则对于 10 Mhz 的 I2C，时钟频率此字段设为 11，此字段最小值设 2

12.3.6.12. 0x038 I2C_INTR_MASK

默认值：0x000048FF			I2C 中断屏蔽（I2C Interrupt Mask）
位域	类型	默认值	描述
31:15	–	–	–
14	R/W	0x1	M_SCL_STUCK_AT_LOW • 0：中断屏蔽 • 1：中断未屏蔽
13	R/W	0x0	M_MASTER_ON_HOLD • 0：中断屏蔽 • 1：中断未屏蔽
12	–	–	–
11	R/W	0x1	M_GEN_CALL • 0：中断屏蔽 • 1：中断未屏蔽
10	R/W	0x0	M_START_DET • 0：中断屏蔽 • 1：中断未屏蔽
9	R/W	0x0	M_STOP_DET • 0：中断屏蔽 • 1：中断未屏蔽
8	R/W	0x0	M_ACTIVITY • 0：中断屏蔽 • 1：中断未屏蔽
7	R/W	0x1	M_RX_DONE

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x000048FF			I2C 中断屏蔽（I2C Interrupt Mask）
位域	类型	默认值	描述
			• 0：中断屏蔽 • 1：中断未屏蔽
6	R/W	0x1	M_TX_ABRT • 0：中断屏蔽 • 1：中断未屏蔽
5	R/W	0x1	M_RD_REQ • 0：中断屏蔽 • 1：中断未屏蔽
4	R/W	0x1	M_TX_EMPTY • 0：中断屏蔽 • 1：中断未屏蔽
3	–	–	–
2	R/W	0x1	M_RX_FULL • 0：中断屏蔽 • 1：中断未屏蔽
1	–	–	–
0	R/W	0x1	M_RX_UNDER • 0：中断屏蔽 • 1：中断未屏蔽

12.3.6.13. 0x03C I2C_INTR_CLR

默认值：0x00000000			I2C 中断清除（I2C Interrupt Clear）
位域	类型	默认值	描述
31:15	–	–	–
14	R/W1C	0x0	CLR _SCL_STUCK_AT_LOW • 0：中断未激活 • 1：写 1 清零

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			I2C 中断清除（I2C Interrupt Clear）
位域	类型	默认值	描述
			详细描述见 I2C_INTR_STAT
13	R/W1C	0x0	CLR _M_MASTER_ON_HOLD • 0：中断未激活 • 1：写 1 清零
12	–	–	–
11	R/W1C	0x0	CLR _GEN_CALL • 0：中断未激活 • 1：写 1 清零 ACK 需持续一个 SCL 周期，在此期间无法对 GEN_CALL 中断清零详细描述见 I2C_INTR_STAT
10	R/W1C	0x0	CLR _START_DET • 0：中断未激活 • 1：写 1 清零 详细描述见 I2C_INTR_STAT
9	R/W1C	0x0	CLR _STOP_DET • 0：中断未激活 • 1：写 1 清零 详细描述见 I2C_INTR_STAT
8	R/W1C	0x0	CLR _ACTIVITY • 0：中断未激活 • 1：写 1 清零 详细描述见 I2C_INTR_STAT
7	R/W1C	0x0	CLR _RX_DONE • 0：中断未激活 • 1：写 1 清零 详细描述见 I2C_INTR_STAT
6	R/W1C	0x0	CLR _TX_ABRT • 0：中断未激活 • 1：写 1 清零 详细描述见 I2C_INTR_STAT

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			I2C 中断清除 （I2C Interrupt Clear）
位域	类型	默认值	描述
5	R/W1C	0x0	CLR _RD_REQ • 0：中断未激活 • 1：写 1 清零 详细描述见 I2C_INTR_STAT
4:1	–	–	–
0	R/W1C	0x0	CLR _RX_UNDER • 0：中断未激活 • 1：写 1 清零 详细描述见 I2C_INTR_STAT

12.3.6.14. 0x040 I2C_RAW_INTR_STAT

默认值：0x00000000			I2C 中断状态 （I2C Interrupt Status）
位域	类型	默认值	描述
31:15	–	–	–
14	R	0x0	SCL_STUCK_AT_LOW • 0：中断未触发 • 1：中断触发 指示 SCL 处于低电平的持续时间是否为 I2C_SCL_STUCK_TIMEOUT 设置的时间，需配置 I2C_CTL[10]为 1
13	R	0x0	MASTER_ON_HOLD • 0：中断未触发 • 1：中断触发 指示 主设备是否保持总线并且 TX FIFO 为空，硬件自动清零
12	–	–	–
11	R	0x0	GEN_CALL • 0：中断未触发 • 1：中断触发 仅在接收到 GEN_CALL 地址并且回复 ACK 时触发。
10	R	0x0	START_DET

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			I2C 中断状态（I2C Interrupt Status）
位域	类型	默认值	描述
			• 0：中断未触发 • 1：中断触发 指示主或者 从模式下，是否检测到一个 START 信号
9	R	0x0	STOP_DET • 0：中断未触发 • 1：中断触发 指示 主或者从 模式下，是否检测到一个 STOP 信号。 在 从模式下：当 I2C_CON[7]（STOP_DET_IFADDRESSED）设为 1，中断 STOP_DET 仅在寻址模式下触发，GEN_CALL 模式下不触发。当 I2C_CON[7]（STOP_DET_IFADDRESSED）设为 0，中断 STOP_DET 不管是否被寻址都会触发。
8	R	0x0	ACTIVITY • 0：中断未触发 • 1：中断触发 指示 I2C 捕获活动状态并且一直保持为 1 直到被清零，清除此位有 3 种方式： • 关闭 I2C • I2C_INTR_CLR[8]写 1 清零 • 系统复位 一旦此位被设为 1，即使 I2C 处于空闲状态，也会保持直到以上其中一种方式清零
7	R	0x0	RX_DONE • 0：中断未触发 • 1：中断触发 当 I2C 作为从设备发送数据时，若主设备不再回复一个 ACK，则此位设为 1。此中断在传输的最后一个字节发生，指示传输结束
6	R	0x0	TX_ABRT • 0：中断未触发 • 1：中断触发 指示 I2C 作为发送者无法完成对发送 FIFO 内容的预期操作。主和从模式都可发生此中断。当此中断发生时，I2C_TX_ABRT_SOURCE 寄存器指示发生停止发送的原因
5	R	0x0	RD_REQ • 0：中断未触发 • 1：中断触发

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			I2C 中断状态（I2C Interrupt Status）
位域	类型	默认值	描述
			当此位设为 1 时，表明在 从模式下，主设备在尝试读取 I2C 的数据。I2C 将总线保持为等待状态，直到此中断被服务，这意味这从设备已经被主设备寻址。当中断发生时，处理器必须响应此中断，将请求的数据写入 I2C_DATA_CMD 寄存器。
4	R	0x0	TX_EMPTY • 0：中断未触发 • 1：中断触发 TX_EMPTY_CTL 设为 0：当发送缓冲器小于等于 I2C_TX_TL 寄存器设定的阈值时，此位设为 1。 TX_EMPTY_CTL 设为 1：当发送缓冲器小于等于 I2C_TX_TL 寄存器设定的阈值时，并且最新命令的地址/数据从内部的移位寄存器传输完成，此位设为 1。 当缓冲器数据量超过 I2C_TX_TL 寄存器设定值硬件会自动清零。
3	–	–	–
2	R	0x0	RX_FULL • 0：中断未触发 • 1：中断触发 当接收缓冲器达到或超过 I2C_RX_TL 寄存器设定的阈值时，此位设 1。当缓冲器深度低于阈值，硬件自动清零。
1	–	–	–
0	R	0x0	RX_UNDER（接收下溢出） • 0：中断未触发 • 1：中断触发 如果接收缓冲器为空时，通过 I2C_DATA_CMD 读取数据则设为 1。

12.3.6.15. 0x048 I2C_ENABLE

默认值：0x00000000			I2C 使能（I2C Enable）
位域	类型	默认值	描述
31:4	–	–	–
3	R/W	0x0	SDA_STUCK_RECOVERY_ENABLE SDA 挂死恢复使能控制 • 0: 主模式禁止 SDA 挂死恢复机制 • 1: 主模式使能 SDA 挂死恢复机制

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			I2C 使能（I2C Enable）
位域	**类型**	**默认值**	**描述**
			如果 SDA 挂死状态由 I2C_TX_ABRT_SOURCE[17] 指示的 TX_ABORT 中断，然后此位发起一个恢复机制，即发送 9 个 SCL 时钟和一个 STOP 信号释放 SDA 线，之后此位清零
2	–	–	–
1	R/W	0x0	ABORT， 传输停止控制 • 0：不发起传输停止或者传输停止已完成 • 1：停止正在进行的操作 软件设置此位可以停止主模式下的传输。只有在 ENABLE 设为 1 的情况下才能对此位操作，否则操作无效。一旦 ABORT 设为 1，软件则无法清除。响应一个 ABORT，控制器在当前传输完成后发出 STOP 信号并且冲刷 TX FIFO，然后在完成停止操作后设置 TX_ABORT 中断（I2C_INTR_STAT[6]）
0	R/W	0x0	ENABLE I2C 使能控制 • 0：I2C 禁止 • 1：I2C 使能

12.3.6.16. 0x04C I2C_ENABLE_STATUS

默认值：0x00000000			I2C 使能状态（I2C Enable Status）
位域	**类型**	**默认值**	**描述**
31:3	–	–	–
2	R	0x0	SLV_RX_DATA_LOST 从模式接收数据丢失。 • 0：从模式接收数据不丢失 • 1：从模式接收数据丢失 指示从模式接收下是否由于 I2C_ENABLE[0]由 1 设为 0 导致至少一个数据丢失。
1	R	0x0	SLV_DISABLED_WHILE_BUSY 从模式禁用时状态 • 0：从模式活跃时禁用 • 1：从模式空闲时禁用 指示由于 I2C_ENABLE[0]由 1 设为 0 时，导致活跃或者潜在的 从设备操作被停止。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			I2C 使能状态（I2C Enable Status）
位域	类型	默认值	描述
0	R	0x0	I2C_EN_STATUS I2C 使能状态 • 0：I2C 禁止状态 • 1：I2C 使能状态

12.3.6.17. 0x050 I2C_STATUS

默认值：0x00000006			I2C 状态（I2C Status）
位域	类型	默认值	描述
31:12	–	–	–
11	R	0x0	SDA_STUCK_NOT_RECOVERED SDA 挂死未恢复 • 1：SDA 挂死触发恢复机制后未恢复 • 0：SDA 挂死触发恢复机制后已恢复 此位仅在主模式有效，从模式无效
10	R	0x0	SLV_HOLD_RX_FIFO_FULL 从设备因 RX FIFO 为满保持总线 • 0：从设备未保持总线或者总线保持非因 RX FIFO 为满 • 1：RX FIFO 为满，从设备保持总线 指示当 RX FIFO 为满并且接收额外的字节数据，从设备则保持总线停止接收。
9	R	0x0	SLV_HOLD_TX_FIFO_EMPTY 从设备因 TX FIFO 为空保持总线 • 0：从设备未保持总线或者总线保持非因 TX FIFO 为空 • 1：TX FIFO 为空，从设备保持总线 指示当从设备收到读请求而 TX FIFO 为空，从设备则保持总线直到 TX FIFO 有数据响应读请求
8	R	0x0	MST_HOLD_RX_FIFO_FULL 主设备因 RX FIFO 为满保持总线 • 0：主设备未保持总线或者总线保持非因 RX FIFO 为满 • 1：RX FIFO 为满，主设备保持总线 指示当 RX FIFO 为满并且接收额外的字节数据，主设备则保持总线停止接收。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000006			I2C 状态（I2C Status）
位域	**类型**	**默认值**	**描述**
7	R	0x0	MST_HOLD_TX_FIFO_EMPTY 主设备因 TX FIFO 为空保持总线 • 0：主设备未保持总线或者总线保持非因 TX FIFO 为空 • 1：TX FIFO 为空，主设备保持总线 指示当 TX FIFO 为空并且最后一个字节数据没有 STOP 信号，主设备则保持总线停止发送。
6	R	0x0	SLV_ACTIVITY 从模式状态 • 0：从模式空闲状态 • 1：从模式活跃状态 当从模式处于活跃状态，此位置 1
5	R	0x0	MST_ACTIVITY 主模式状态 • 0：主模式空闲状态 • 1：主模式活跃状态 当主模式有限状态机处于活跃状态，此位置 1
4	R	0x0	RFF（RX FIFO FULL） 接收 FIFO 为满状态标志 • 0：接收 FIFO 非满 • 1：接收 FIFO 为满 当接收 FIFO 包含一个或多个数据时，此位置为 0。FIFO 为满时为 1
3	R	0x0	RFNE（RX FIFO NOT EMPTY） 接收 FIFO 非空状态标志 • 0：接收 FIFO 为空 • 1：接收 FIFO 非空 当接收 FIFO 包含一个或多个数据时，此位置 1。FIFO 为空时清零
2	R	0x1	TFE（TX FIFO EMPTY） 发送 FIFO 为空状态标志 • 0：发送 FIFO 非空 • 1：发送 FIFO 为空

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000006			I2C 状态（I2C Status）
位域	类型	默认值	描述
			当发送 FIFO 为空时，此位置 1。当 FIFO 包含一个或多个数据时清零。此位不请求中断
1	R	0x1	TFNF（TX FIFO NOT FULL） 发送 FIFO 非满状态标志 • 0：发送 FIFO 为满 • 1：发送 FIFO 非满 当发送 FIFO 包含一个或多个数据时，此位置 1。FIFO 为满时清零
0	R	0x0	ACTIVITY I2C 状态 • 0：I2C 处于空闲状态 • 1：I2C 处于活跃状态

12.3.6.18. 0x054 I2C_TX_ABRT_SOURCE

默认值：0x00000000			I2C 发送停止源（I2C Transport Abort Source）
位域	类型	默认值	描述
31:23	R	0x0	TX_FLUSH_CNT。 此位域指示由于 TX_ABRT 中断而冲刷的 TX FIFO 数据命令数。
22:18	–	–	–
17	R	0x0	ABRT_SDA_STUCK_AT_LOW • 0：不生成停止操作。 • 1：生成停止操作。 仅在主模式有效，主设备检测 SDA 持续低电平的时间超过 I2C_SDA_STUCK_TIMEOUT 设置的阈值。
16	R	0x0	ABRT_USER_ABRT • 0：不生成停止操作。 • 1：生成停止操作。 仅在 主模式有效，主设备检测到传输中止 (I2C_ENABLE[1])，生成停止操作。
15	R	0x0	ABRT_SLVRD_INTX • 0：不生成停止操作 。 • 1：生成停止操作。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			I2C 发送停止源 （I2C Transport Abort Source）
位域	**类型**	**默认值**	**描述**
			从模式下正在进行发送数据给远程主设备，此时用户在 I2C_CMD_DATA[8] 写 1，则产生停止操作。
14	R	0x0	ABRT_SLV_ARBLOST • 0：不生成停止操作。 • 1：生成停止操作。 指示在向远程主设备传输时，从模式丢失总线，同时 I2C_TX_ABRT_SOURCE[12] 设 1，生成停止操作。
13	R	0x0	ABRT_SLVFLUSH_TXFIFO • 0：不生成停止操作 。 • 1：生成停止操作。 指示从模式已接收到读取命令，并且某些数据存在于 TX FIFO 中，因此 从模式发出 TX_ABRT 中断以刷新 TX FIFO 中的旧数据，则生成停止操作。
12	R	0x0	ABRT_LOST • 0：不生成停止操作。 • 1：生成停止操作。 指示主模式失去仲裁，或者设置了 I2C_TX_ABRT_SOURCE[14],则从模式失去仲裁，生成停止操作。
11	R	0x0	ABRT_MASTER_DIS • 0：不生成停止操作。 • 1：生成停止操作。 指示用户在主模式禁用的情况下发起一个 主模式操作，则生成停止操作。
10	R	0x0	ABRT_10B_RD_NORSTRT • 0：不生成停止操作。 • 1：生成停止操作。 指示当 RESTART 禁止时 （I2C_CTL[6]设 0），主设备在 10bit 寻址模式发送一个读操作，则生成停止操作。
9	R	0x0	ABRT_SBYTE_NORSTRT • 0：不生成停止操作 。 • 1：生成停止操作。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			I2C 发送停止源 （I2C Transport Abort Source）
位域	类型	默认值	描述
			指示在 RESTART 禁止时 （I2C_CTL[6]设 0）用户发送一个 START 字节清除此位，需修复造成此位生成停止操作的源，RESTART 必须先使能 （I2C_CTL[6]设 1），I2C_TAR[11]或者 I2C_TAR[10]必须清除。一旦此位的源被修复，此位就可以读取清零。若在修复源之前尝试读取此位清零，则此位在一个周期内清除，随后重新赋值。
8	–	–	–
7	R	0x0	ABRT_SBYTE_ACKDET • 0：不生成停止操作。 • 1：生成停止操作。 指示 主设备发送一个 START 字节并且被应答 （错误行为），则生成停止操作。
6	–	–	–
5	R	0x0	ABRT_GCALL_READ • 0：不生成停止操作 。 • 1：生成停止操作。 指示 I2C 发送一个 GEN_CALL 但是紧接着用户进行一个读操作 （I2C_DATA_CMD[8]写 1），则生成停止操作。
4	R	0x0	ABRT_GCALL_NOACK • 0：不生成停止操作。 • 1：生成停止操作。 指示主模式发出一个 GEN_CALL,而总线上没有从设备 应答，则生成停止操作。
3	R	0x0	ABRT_TXDATA_NOACK • 0：不生成停止操作 。 • 1：生成停止操作。 仅在 主模式，当发送一个地址并且收到应答，接着发送数据而没有收到应答，则生成停止操作。
2	R	0x0	ABRT_10ADDR2_NOACK • 0：不生成停止操作。 • 1：生成停止操作。 仅在主模式，当 10 bit 寻址模式发送第一个地址并且收到应答，接着发送第二个地址而没有收到应答，则生成停止操作。
1	R	0x0	ABRT_10ADDR1_NOACK

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			I2C 发送停止源（I2C Transport Abort Source）
位域	类型	默认值	描述
			• 0：不生成停止操作。 • 1：生成停止操作。 仅在主模式，当 10 bit 寻址模式发送第一个地址没有收到应答，则生成停止操作。
0	R	0x0	ABRT_7B_ADDR_NOACK • 0：不生成停止操作。 • 1：生成停止操作。 仅在 主模式，当 7 bit 寻址模式发送第一个地址没有收到应答，则生成停止操作。

12.3.6.19. 0x090 I2C_RX_TL

默认值：0x00000000			I2C 接收阈值（I2C Receive Threshold）
位域	类型	默认值	描述
31:3	–	–	–
2:0	R/W	0x0	RX_TL 接收 FIFO 阈值深度。 控制触发 RX_FULL 中断的数据数量，接收数据数量大于等于此值时触发。有效范围为 0–7，硬件不允许该值设置为大于缓冲寄存器的最大深度。值 0 设置 1 项的阈值，值 7 设置 8 项的阈值。

12.3.6.20. 0x094 I2C_TX_TL

默认值：0x00000000			I2C 发送阈值（I2C Transport Threshold）
位域	类型	默认值	描述
31:3	–	–	–
2:0	R/W	0x0	TX_TL 发送 FIFO 阈值深度。 控制触发 TX_EMPTY 中断的数据数量，发送数据数量小于等于此值时触发。有效范围为 0–7，硬件不允许该值设置为大于缓冲寄存器的最大深度。值 0 设置 0 项的阈值，值 7 设置 7 项的阈值。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.3.6.21. 0x098 I2C_TXFLR

默认值：0x00000000			I2C 发送 FIFO 水位（I2C Transport FIFO Level）
位域	类型	默认值	描述
31:4	–	–	–
3:0	R	0x0	TXFLR，显示发送 FIFO 数据量。

12.3.6.22. 0x09C I2C_RXFLR

默认值：0x00000000			I2C 接收 FIFO 水位（I2C Receive FIFO Level）
位域	类型	默认值	描述
31:4	–	–	–
3:0	R	0x0	RXFLR，接收 FIFO 数据量。

12.3.6.23. 0x0A0 I2C_SCL_STUCK_TIMEOUT

默认值：0xFFFFFFFF			I2C SCL 挂死超时（I2C_SCL_STUCK_TIMEOUT）
位域	类型	默认值	描述
31:0	R/W	0xffff_ffff	I2C_SCL_STUCK_TIMEOUT SCL 挂死时间设置。 如果 I2C 检测 SCL 处于低的时间超过此位域设定的时间，则产生相应的中断（SCL_STUCK_AT_LOW）。仅在 I2C 禁止时（I2C_ENABLE[0] 设为 0）可配置，其它时间写入无效。

12.3.6.24. 0x0A4 I2C_SDA_STUCK_TIMEOUT

默认值：0xFFFFFFFF			I2C SDA 挂死超时（I2C_SDA_STUCK_TIMEOUT）
位域	类型	默认值	描述
31:0	R/W	0xffff_ffff	I2C_SDA_STUCK_TIMEOUT SDA 挂死时间设置。 如果 I2C 检测 SDA 处于低的时间超过此位域设定的时间，并且 I2C_ENABLE[3]设为 1，则触发 SDA 挂死恢复机制。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.3.6.25. 0x0B0 I2C_FS_SPIKELEN

默认值：0x00000002			I2C 快速模式毛刺抑制时长（I2C_FS_SPIKELEN）
位域	类型	默认值	描述
31:0	R/W	0x2	I2C_FS_SPIKELEN I2C FS 模式毛刺抑制寄存器。 此位设置 SCL 与 SDA 线持续抑制毛刺的最长时间，以 I2C_clk 周期为单位。 此位只有在 I2C_ENABLE[0]设为 0 时可配置，其它时间配置无效。此位最小值为 1，硬件会阻止写入小于此值的配置，即写 0 则此位会被设为 1。

12.3.6.26. 0x0FC VERSION

默认值：0x00000100			I2C 版本号（VERSION）
位域	类型	默认值	描述
31:0	RO	0x00000100	Version (模块版本号) 采用 BCD 码显示，V1.0。

12.4. Controller Area Network (CAN)

CAN 是一种广泛应用于汽车控制系统和一般工业环境中的区域网络总线。作为一款多主机、多广播的通信协议，CAN 以其高可靠性和卓越的错误检测能力而著称。

12.4.1. 特性说明

- 支持 CAN2.0A 和 CAN2.0B 协议。
- 支持 11 位标准格式标识符和 29 位扩展格式标识符。
- 可编程通信速率最高可达 1 Mbps。
- 支持多种操作模式：正常模式、只听模式、自测模式、休眠模式、复位模式。
- 支持接收过滤器，支持两种过滤模式。
- 64 bytes 缓冲器。
- 支持错误检测与处理：错误计数、错误报警阈值可配置、错误捕获、仲裁丢失捕获。

12.4.2. 原理框图

CAN 控制器原理如下：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 位时序逻辑 BTL：负责同步位时序，确保数据传输的稳定性，以预先配置的波特率发送和接受报文。
- 位流处理器 BSP：负责对发送数据进行位填充、附加 CRC 等处理，并为位时序逻辑模块生成位流。负责对接收的位流进行去填充、验证 CRC 等处理。负责检测总线上的错误并将错误报告给错误管理逻辑。
- 接收过滤器：是一个可编程的报文过滤单元，允许控制器根据报文的标识符接收或拒绝该报文，可以过滤掉总线发送的与节点无关的消息，使节点效率提高。
- 缓冲 FIFO：大小为 64 bytes，负责存储通过接收过滤器的报文。FIFO 中的第一条报文将被映射到 13 bytes 的接收缓冲寄存器中可被 CPU 访问，直到该报文被清除，空间被释放。随后，接收缓冲寄存器将继续映射 FIFO 中的下一条报文。
- 错误管理逻辑 EML：负责更新发送错误计数 TEC 和接收错误计数 REC 的数值，记录错误类型、错误位置，更新控制器错误状态，确保 BSP 模块发送正确的错误标志。还负责记录报文仲裁丢失时的 bit 位置。

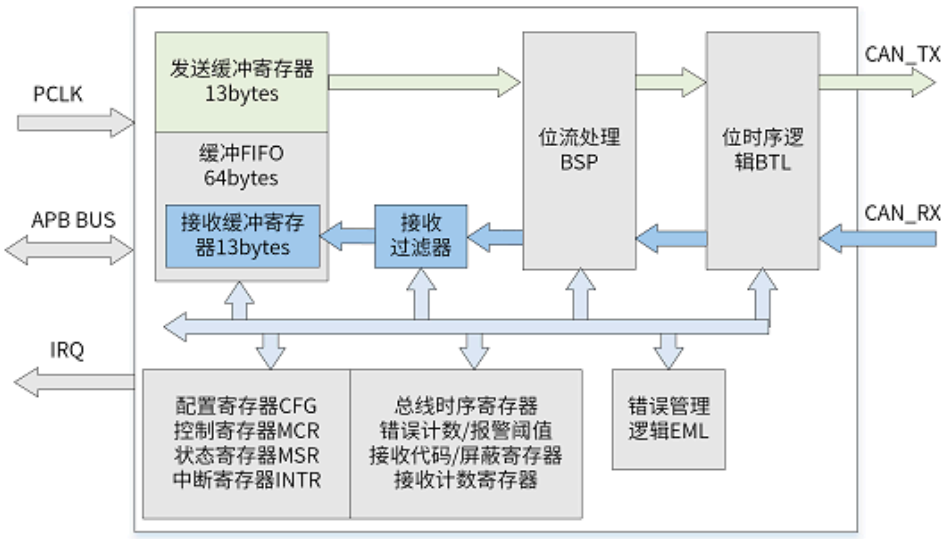
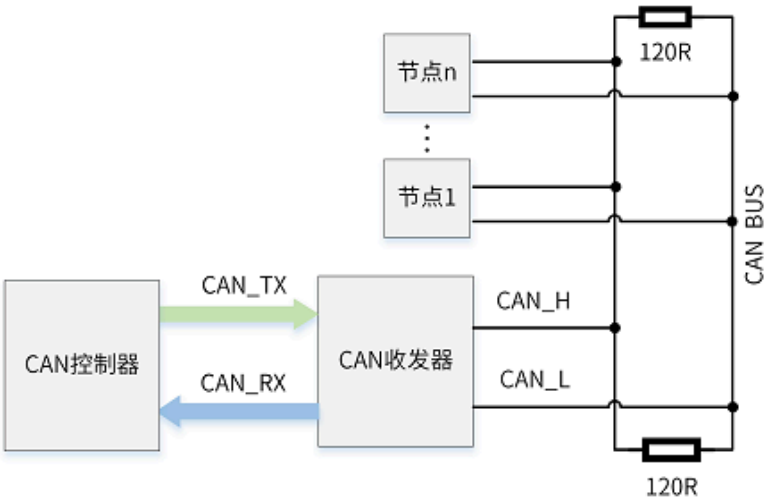


图 12–9 CAN 控制器原理框图

12.4.3. 典型应用

CAN 典型应用如图所示：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



12.4.3.1. 运行模式

控制器包含五种运行模式，由模式寄存器 CAN_MODE 定义设置。

- 复位模式：Reset Mode。
  - 进入复位模式，控制器离线不参与总线任何活动，可修改 CAN 控制器的各种参数配置。
  - 退出复位模式，控制器需等待 11 个连续隐性位出现（等效于总线空闲），才能正常接收和发送报文。
- 正常模式： Normal Mode，CAN 控制器可以发送和接收包含错误信号在内的报文。
- 自测模式：Self-Test Mode，与正常模式相同，但在该模式下，CAN 控制器发送报文时，即使没有接收到应答，也不会产生应答错误。

  通常在控制器自测时使用。
- 只听模式： Listen Only Mode，CAN 控制器可以接收报文，但在 CAN 总线上保持完全被动。因此， CAN 控制器将无法发送任何报文、应答或错误信号，错误计数将保持冻结状态。

  通常用于 CAN 总线监控。
- 休眠模式：Sleep Mode，无总线活动及无中断发生，可进入低功耗休眠模式，该模式下时钟停止。

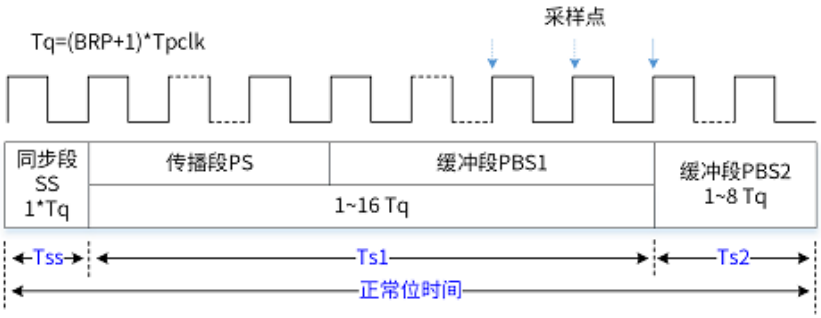
  SLEEP_MOD 为 0 或总线有信号输入时唤醒，产生唤醒中断。

12.4.3.2. 位时序

为了实现位同步，每个正常位时间划分为几个不重叠时间的片段，包括同步段 SS，传播段 PS，相位缓冲段 PBS1，相位缓冲段 PBS2，一个完整的位由 8~25 个时间定额 T_q 组成。

- 如图所示，将传播段 PS 和相位缓冲段 PBS1 合并形成新的时间段 T_{S1}。
- 在复位模式下，配置总线时序 0 寄存器 CAN_BTR0 和总线时序 1 寄存器 CAN_BTR1 中的 BRP、SJW、T_{S1}、T_{S2}、SAM 等参数，决定 CAN 的波特率。
- SJW 表示 CAN 波特率的容许偏差量，其大小与波特率值没有直接关系，SJW 值越大则总线获得更宽波特率容忍度。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



位时序各参数计算关系如下：

- 时间定额 $T_q = 2 \times (BRP[5:0] + 1) \times T_{pclk}$，其中 T_{pclk} 为 APB 时钟（通常为 24 MHz）周期。
- 位时间 $T_{bit} = [1 + (T_{S1} + 1) + (T_{S2} + 1)] \times Tq$。
- 波特率 $Baud = 1 / T_{bit}$。
- 采样点 $SP = [1 + (T_{S1} + 1)]/ [1 + (T_{S1} + 1) + (T_{S2} + 1)]$。

位时序各参数计算

假设单次采样的参数值如下：

SAM = 0，且 SJW = 2，TS1 = 8，TS2 = 1，BRP = 0，则波特率为 24 MHz / [2 × (0 + 1) × (1 + 9 + 2)] = 1 Mbps，采样点为 (1 + 9) / (1 + 9 + 2) = 83%。

建议选取规则：

- $T_{S2} \geq SJW$
- BRP 尽量小、SJW 尽量大。
- 采样点选取范围 75%~85% 之间。
- 尽量采用单次采样。

12.4.3.3. 通信报文

CAN 通信报文类型包含数据帧、远程帧、错误帧、过载帧和间隔帧。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

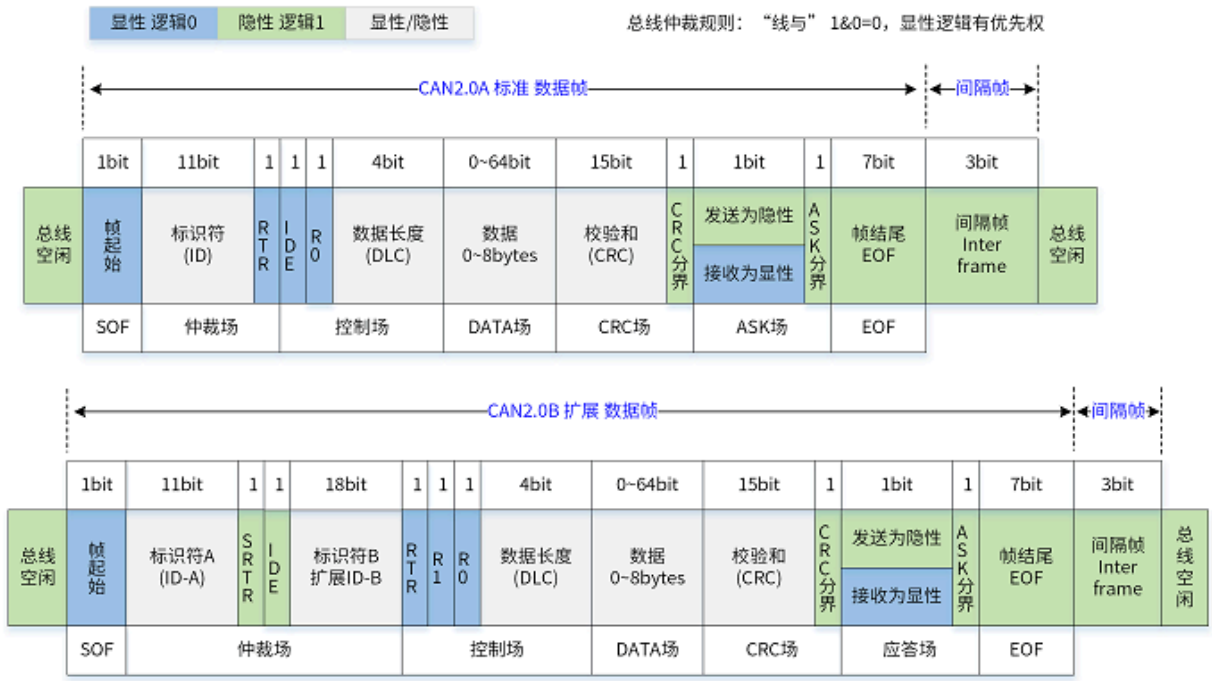


图 12–10 CAN 标准格式和扩展格式数据帧

图 12–11 CAN 标准格式和扩展格式远程帧

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

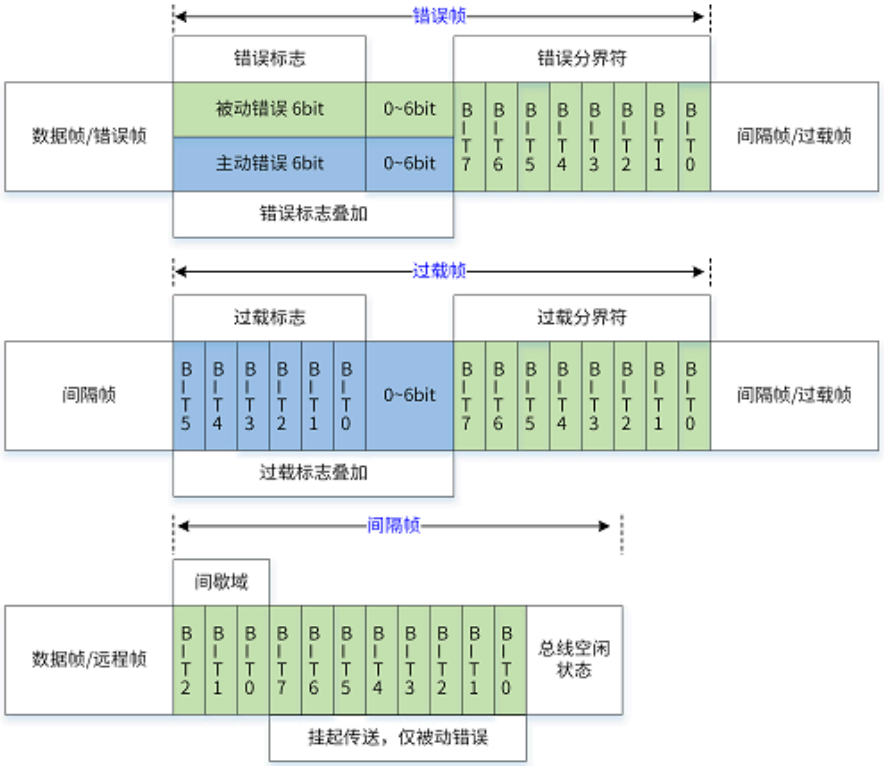


图 12–12 CAN 错误帧、过载帧、间隔帧

表 12–11 数据帧/远程帧描述

数据/远程帧	描述
SOF	帧起始，1 bit，用于同步总线上节点的单个显性位。
ID–A	标识符 A，11 bits，对应标准格式中的 11 位标识符（ID10~ ID0），或扩展格式中 29 位标识符的前 11–bit（ID28~ ID18）。
RTR	远程发送请求位，1 bit，显示当前报文是数据帧（显性 0）还是远程帧（隐性 1）。 当某个数据帧和远程帧有相同标识符时，数据帧始终优先于远程帧仲裁。
SRTR	代替远程发送请求位，1 bit，在扩展格式中以替代标准格式相同位置的 RTR 位。
IDE	标识符扩展位，1 bit，显示当前报文是标准格式（显性 0）还是扩展格式（隐性 1）。 当某标准帧和扩展帧有相同基标识符时，标准帧将始终优先于扩展帧仲裁。
ID–B	标识符 B，18 bits，扩展格式中 29 位标识符的剩余 18 bit（ID17 ~ ID0）。
R1	保留位，始终是显性位。
R0	保留位，始终是显性位。
DLC	数据长度代码， 4 bits，包含 0 ~ 8 中任一数值。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 12–11 数据帧/远程帧描述 (续)

数据/远程帧	描述
	• 数据帧用于表示包含的数据字节数量。 • 远程帧用于表示从其它节点请求的数据字节数量。
数据字节	表示数据帧的数据负载量，该字节数量应与 DLC 的值匹配。首先发送数据字节 0，各数据字节优先发送最高有效位。
CRC	CRC 序列是一个 15–bit 的循环冗余校验码。具体实现方式为： 1. 被除多项式的系数由 SOF、仲裁场、控制场、数据场（若存在）及 15 位（最低系数）0 组成的未经填充的位流给定，而生成多项式为 $x^{15}+x^{14}+x^{10}+x^{8}+x^{7}+x^{4}+x^{3}+x^{0}$。 2. 被除多项式被生成多项式除（按模 2 计算），余数即为将要发送到总线上的 CRC 序列。 3. 发送节点和接收节点均采用相同方法生成 CRC 校验码，并与发送节点送出的 CRC 校验码进行比较： • 若出错，控制器依据仲裁原则及受损报文优先原则对已损坏报文自动重发。
CRC 分界符	分界符，1 bit，隐性位。
ASK 槽	应答槽，1 bit，用于接收节点，表示是否已成功接收数据帧或远程帧。发送节点将在应答槽中发送一个隐性位，若接收到的帧无错误，则接收节点用一个显性位以示应答。
ASK 分界符	分界符，1 bit，隐性位。
EOF	帧结束，7 bits，隐性位，标志着数据帧或远程帧的结束。

12.4.3.4. 中断管理

CAN 控制器提供八种中断，具体如下：

- 总线错误中断 ERRB_INT

  控制器在总线上检测到错误时触发此中断。错误类型和错误位置都将被捕获记录在状态寄存器 CAN_STAT 的 ERR_TYPE 和 ERR_CODE 中。状态被清除之前（通过 CPU 的读取），将不会再记录新的总线错误信息。

- 仲裁丢失中断 ARBLOST_INT

  控制器发送报文并且丢失仲裁时触发此中断。丢失仲裁位置将被捕获记录在状态寄存器 CAN_STAT 的 ARBLOST_CAP 中，状态被清除之前（通过 CPU 读取），将不会再记录新的仲裁丢失位置。

- 被动错误中断
- 唤醒中断
- 数据溢出中断
- 错误报警中断 ERRW_INT

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

状态寄存器 CAN_STAT 中 ERR_STAT 或 BUS_STAT 的位值发生改变（由 0 变 1 或由 1 变 0）都会触发此中断。触发时状态值可分以下几种：

- ERR_STAT = 0 与 BUS_STAT = 0：
  - 如果控制器处于主动错误状态，则表示 TEC 和 REC 的值都返回 ERRWT 所设阈值之下。
  - 如果控制器此前处于总线恢复状态，则表示此时总线恢复已成功完成。
- ERR_STAT = 1 与 BUS_STAT = 0：表示 TEC 或 REC 数值已超过 ERRWT 所设阈值。
- ERR_STAT = 1 与 BUS_STAT = 1：表示控制器已进入离线状态（TEC ≥ 255）。
- ERR_STAT = 0 与 BUS_STAT = 1：表示总线恢复期间，控制器 TEC 数值已低于 ERRWT 所设阈值。

- 发送中断 TX_INT

  当发送缓冲器空闲，将其它报文加载到缓冲器中等待发送时，都会触发此中断。

- 接收中断 RX_INT

  当接收 FIFO 中有待读取报文（接收数据计数寄存器 CAN_RXC 中 RXC > 0）时触发此中断。计数的报文数量包括有效报文和溢出报文。直到通过 RXB_REL 指令位清除所有挂起接收报文后，接收中断才会失效。

中断可由中断寄存器 CAN_INTR 定义，设置中断使能寄存器 CAN_INTEN 中相应的使能位，可以单独允许和禁用每个中断源。中断的工作原理和规则如下：

- 当有一个或多个中断触发，控制器 IRQ 有效，当所有中断位都被清除，控制器 IRQ 失效。
- 中断寄存器被读取后，除接收中断外，其中的中断位将自动清除。
- 接收中断直到通过 RXB_REL 指令位清除所有接收报文后，才能被清除，否则会影响接收缓冲状态 RXB_STAT。

12.4.3.5. 数据缓冲器

数据缓冲寄存器组是缓存发送和接收帧的存储空间，其访问地址范围为 0x40~0x70，读写共用相同的地址空间。

注：

复位状态下不可访问。

表 12-12 数据缓冲寄存器组

寄存器	偏移地址	标准格式内容	扩展格式内容
CAN_BUF0	0x40	TX/RX 帧信息	TX/RX 帧信息
CAN_BUF1	0x44	TX/RX 标识符 1	TX/RX 标识符 1
CAN_BUF2	0x48	TX/RX 标识符 2	TX/RX 标识符 2
CAN_BUF3	0x4C	TX/RX 数据 1	TX/RX 标识符 3
CAN_BUF4	0x50	TX/RX 数据 2	TX/RX 标识符 4

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 12–12 数据缓冲寄存器组 (续)

寄存器	偏移地址	标准格式内容	扩展格式内容
CAN_BUF5	0x54	TX/RX 数据 3	TX/RX 数据 1
CAN_BUF6	0x58	TX/RX 数据 4	TX/RX 数据 2
CAN_BUF7	0x5C	TX/RX 数据 5	TX/RX 数据 3
CAN_BUF8	0x60	TX/RX 数据 6	TX/RX 数据 4
CAN_BUF9	0x64	TX/RX 数据 7	TX/RX 数据 5
CAN_BUF10	0x68	TX/RX 数据 8	TX/RX 数据 6
CAN_BUF11	0x6C	保留	TX/RX 数据 7
CAN_BUF12	0x70	保留	TX/RX 数据 8

- 发送缓冲寄存器：CPU 对数据缓冲寄存器组的写操作访问，配置待发送报文，指定报文的帧类型、帧格式、帧标识符和数据。CPU 通过控制寄存器 CAN_MCR 配置发送报文模式：
  - 正常报文发送：TX_REQ 置 1。
  - 自发自收：SELF_REQ 置 1。
  - 单次发送：TX_REQ 和 ABORT_REQ 同时置 1。
- 接收缓冲寄存器：CPU 对数据缓冲寄存器组的读取操作访问，接收缓冲寄存器映射到缓冲 FIFO 中的第一条报文，获取第一条报文的帧类型、帧格式、帧标识符和数据。

  读取完接收缓冲寄存器中的报文后，CPU 通过控制寄存器 CAN_MCR 中的 RXB_REL 置 1 来释放接收缓冲寄存器。

  若接收 FIFO 中仍有待处理的报文，按照接收报文的先后次序，将最早接收到的报文映射到接收缓冲寄存器中。
- 缓冲 FIFO： 是一个 64 字节大小的内部缓冲器，以先进先出原则存储接收到的报文。

  1. 接收缓冲寄存器为 FIFO 中可访问窗口，偏移地址 0x40~0x70，将被映射到 FIFO 中的第一条报文。

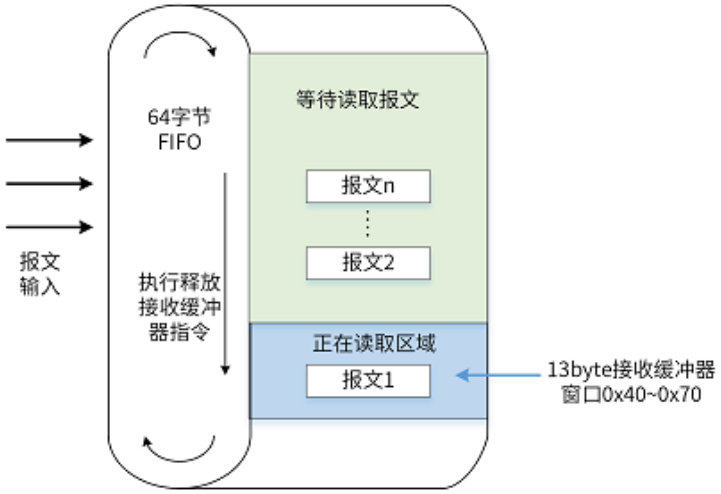
     一条报文可在 FIFO 中占 3 ~ 13 字节空间，其中字节序与接收缓冲寄存器相同。
  2. 当控制器接收到一条报文时，接收数据计数器 RXC 增加 1，最大值为 64。

     如果 FIFO 中有足够空间，报文内容将被写入到 FIFO 中。
  3. CPU 读取接收缓冲寄存器中的报文后，通过将 RXB_REL 置 1，释放 FIFO 中第一条报文所占的空间，RXC 将减小 1。
  4. 接收缓冲寄存器接着映射 FIFO 中的下一条报文。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- FIFO 数据溢出：任何情况下，FIFO 中可以存储的报文数量取决于各条报文的长度。当 FIFO 中没有足够空间完整地存储新报文时，控制器会产生数据溢出，通过状态位或数据溢出中断反馈给 CPU。

  FIFO 在内部将溢出报文标记为无效，后续接收到的溢出报文仍然将增加 RXC 到最大值 64。为了读取接收 FIFO 中的所有有效报文并清除所有溢出报文，应重复调用 RXB_REL，直到 RXC 为 0。



12.4.3.6. 接收过滤器

接收过滤器，是一个可编程的报文过滤单元，允许控制器根据报文的标识符、帧类型、第一个数据字节接收或拒绝该报文。接收过滤器由四个接收代码寄存器 CAN_RXCODE 和四个接收屏蔽寄存器 CAN_RXMASK 定义组成。

图 12–13 : 接收过滤器原理中，每条报文中的位必须匹配 RXCODE 值所指定的模式或者被 RXMASK 值屏蔽，才能使该报文通过过滤并存储到接收 FIFO 中。

接收代码寄存器、接收屏蔽寄存器，分别与缓冲 0~3 寄存器（偏移地址 0x40~0x4C）、缓冲 4~7 寄存器（偏移地址 0x50~5C）地址空间相同，因此只有控制器在复位模式时，才允许访问这两个寄存器。

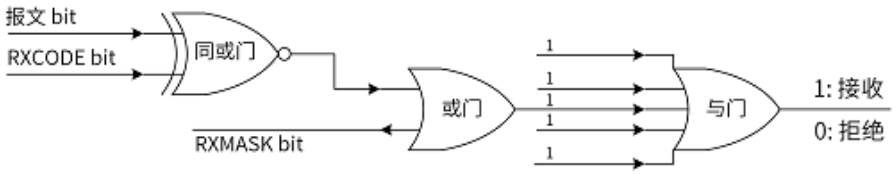


图 12–13 接收过滤器原理

- 单过滤模式：在模式寄存器 CAN_MODE 中将 FILTER_SEL 设置为 1 ，启动单过滤模式。

  根据 RXCODE/ RXMASK 的值定义单个过滤器，其定义及过滤报文格式如图所示。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

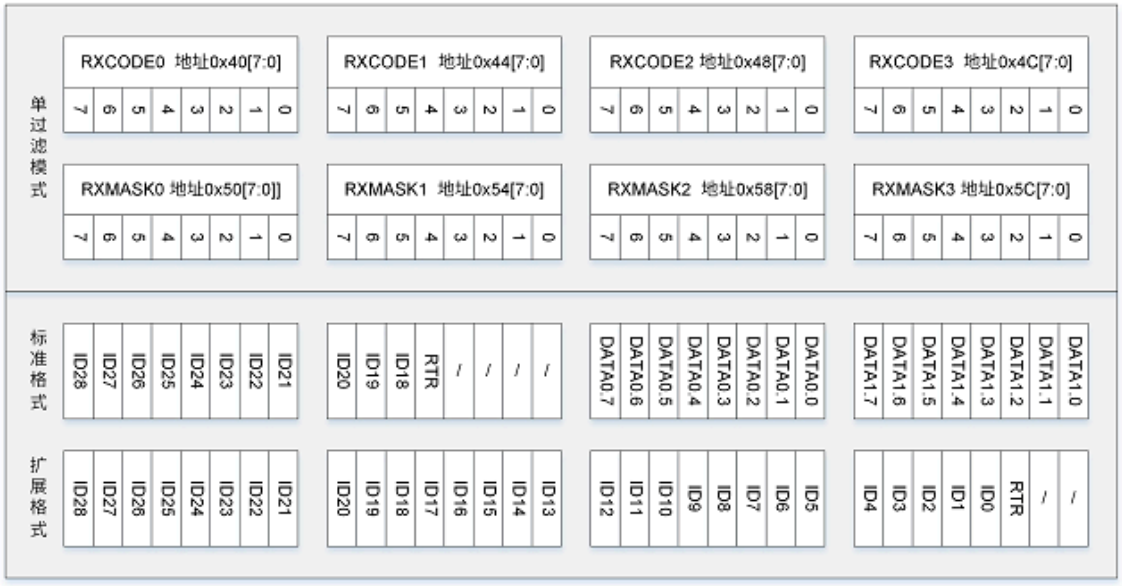


图 12–14 单过滤模式下单个过滤器定义及其可过滤报文格式

- 双过滤模式：在模式寄存器 CAN_MODE 中将 FILTER_SEL 设置为 0， 启动双过滤模式。

  根据 RXCODE/ RXMASK 的值将定义两个过滤器，分别为过滤器 0 和过滤器 1。两个过滤器的标准格式 (SFF) 和扩展格式 (EFF) 下的过滤报文定义分别如下图所示。双过滤模式下，如果报文通过至少一个滤波器，则表示报文成功通过过滤。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

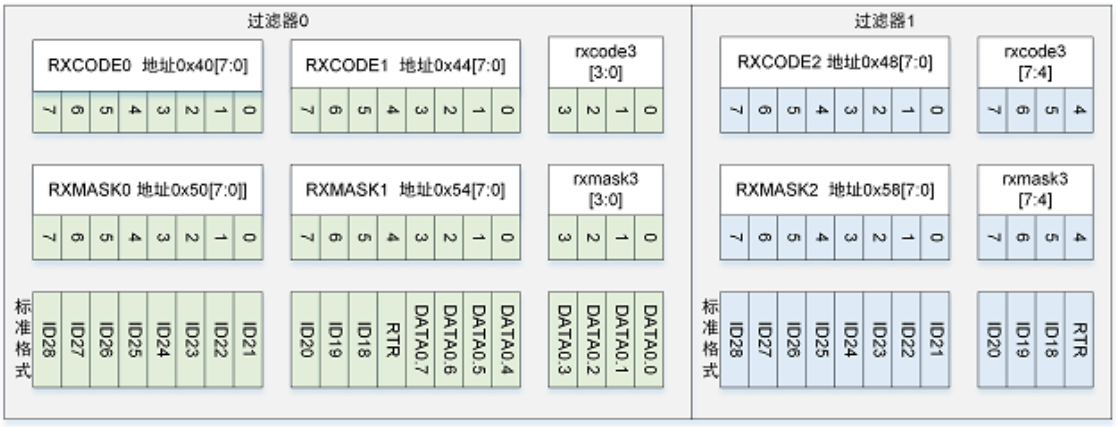


图 12–15 双过滤模式下 SFF 标准格式过滤报文定义

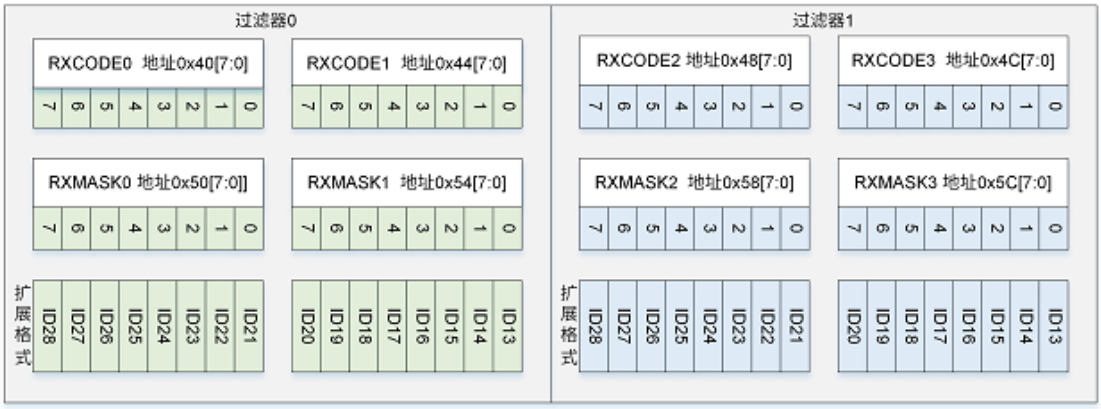


图 12–16 双过滤模式下 EFF 扩展格式过滤报文定义

12.4.3.7. 错误管理

错误计数包含发送错误计数 TEC 和接收错误计数 REC。错误计数决定 CAN 控制器当前的错误状态，例包括主动错误、被动错误或离线。

除了错误状态外，控制器还提供错误报警阈值 ERRWT 功能。ERRWT 功能可在控制器进入被动错误状态前，提醒用户当前发生的严重总线错误。

控制器当前的错误状态通过 TEC、REC、ERR_STAT 、BUS _STAT 和 ERRWT 体现，数值和状态位的变化可以触发中断，从而提醒用户当前的错误状态变化，详情可参考 中断管理。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

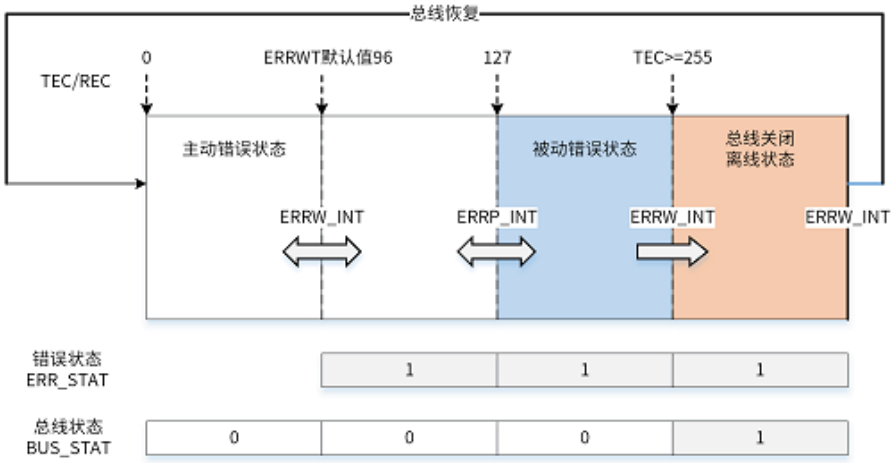


图 12–17 错误状态变化及中断关系

图 12–17 : 错误状态变化及中断关系表示了错误状态、计数数值、状态位以及相关中断之间的关系。

- 作为报警功能，错误报警阈值 ERRWT 可提示当前发生的总线错误，在控制器进入被动错误状态之前触发。在复位模式下，ERRWT 数值可在错误报警阈值寄存器 CAN_ERRWT 中进行配置。
  - 当 TEC 和 REC 中任意数值大于等于 ERRWT 时，ERR_STAT 被置 1。
  - 当 TEC 和 REC 数值都小于 ERRWT 时， ERR_STAT 被复位 0。
  - 只要 ERR_STAT 或 BUS_STAT 位值发生变化，便会触发错误报警中断 ERRW_INT。
- 当 TEC 或 REC 数值大于 127 时，节点进入被动错误状态。
- 当 TEC 和 REC 数值都小于等于 127 时，节点重新变为主动错误状态。

  节点在主动错误和被动错误状态之间切换时，都将触发被动错误中断 ERRP_INT。
- 当 TEC 数值大于 255 时，节点进入总线关闭离线状态。

  此时，控制器将 REC 数值置为 0，TEC 数值置为 127，BUS_STAT 位置 1，产生错误报警中断 ERRW_INT，控制器进入复位模式。
- 为了返回主动错误状态，必须进行离线恢复。
  1. 退出复位模式，进入正常操作模式。
  2. 要求节点在总线上检测到 128 次 11 个连续隐性位。

     每一次检测到 11 个连续隐性位时，TEC 数值都将减 1。
  3. 当离线恢复完成后（TEC 数值从 127 减小到 0），BUS_STAT 位复位为 0，从而触发错误报警中断 ERRW_INT。

12.4.3.8. 错误捕获

错误捕获功能允许控制器以错误代码的形式，记录总线错误类型和错误位置。

当检测到一个总线错误时，触发总线错误中断 ERRB_INT，状态寄存器 CAN_STAT 的 ERR_CODE 会记录相应的错误代码。在当前错误代码被读取前，后续的总线错误中断触发时，将不会再记录。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.4.3.9. 仲裁丢失捕获

仲裁丢失捕捉功能允许控制器记录仲裁丢失的 bit 位置。当控制器丢失仲裁时，所丢失 bit 位置将被记录在状态寄存器 CAN_STAT 的 ARBLOST_CAP 中，同时触发仲裁丢失中断 ARBLOST_INT。在当前仲裁丢失捕获 bit 位置被读取前，后续的仲裁丢失中断触发时，将不会再记录。

12.4.4. 寄存器列表

表 12–13 寄存器列表

偏移地址	缩写	寄存器描述	章节
0x000	CAN_MODE	CAN 模式寄存器	0x000 CAN_MODE
0x004	CAN_MCR	CAN 控制寄存器	0x004 CAN_MCR
0x008	CAN_STAT	CAN 状态寄存器	0x008 CAN_STAT
0x00C	CAN_INTR	CAN 中断寄存器	0x00C CAN_INTR
0x010	CAN_INTEN	CAN 中断使能寄存器	0x010 CAN_INTEN
0x018	CAN_BTR0	CAN 总线时序 0 寄存器	0x018 CAN_BTR0
0x01C	CAN_BTR1	CAN 总线时序 1 寄存器	0x01C CAN_BTR1
0x02C	CAN_ARBLOST	CAN 仲裁丢失捕获寄存器	0x02C CAN_ARBLOST
0x030	CAN_ERRCODE	CAN 错误代码捕获寄存器	0x030 CAN_ERRCODE
0x034	CAN_ERRWT	CAN 错误报警阈值寄存器	0x034 CAN_ERRWT
0x038	CAN_RXERR	CAN 接收错误计数寄存器	0x038 CAN_RXERR
0x03C	CAN_TXERR	CAN 发送错误计数寄存器	0x03C CAN_TXERR
0x040	CAN_BUF0	CAN 缓冲 0 寄存器	0x040 CAN_BUF0
0x044	CAN_BUF1	CAN 缓冲 1 寄存器	0x044 CAN_BUF1
0x048	CAN_BUF2	CAN 缓冲 2 寄存器	0x048 CAN_BUF2

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 12–13 寄存器列表 (续)

偏移地址	缩写	寄存器描述	章节
0x04C	CAN_BUF3	CAN 缓冲 3 寄存器	0x04C CAN_BUF3
0x050	CAN_BUF4	CAN 缓冲 4 寄存器	0x050 CAN_BUF4
0x054	CAN_BUF5	CAN 缓冲 5 寄存器	0x054 CAN_BUF5
0x058	CAN_BUF6	CAN 缓冲 6 寄存器	0x058 CAN_BUF6
0x05C	CAN_BUF7	CAN 缓冲 7 寄存器	0x05C CAN_BUF7
0x060	CAN_BUF8	CAN 缓冲 8 寄存器	0x060 CAN_BUF8
0x064	CAN_BUF9	CAN 缓冲 9 寄存器	0x064 CAN_BUF9
0x068	CAN_BUF10	CAN 缓冲 10 寄存器	0x068 CAN_BUF10
0x06C	CAN_BUF11	CAN 缓冲 11 寄存器	0x06C CAN_BUF11
0x070	CAN_BUF12	CAN 缓冲 12 寄存器	0x070 CAN_BUF12
0x040	CAN_RXCODE0	CAN 接收代码 0 寄存器	0x040 CAN_RXCODE0
0x044	CAN_RXCODE1	CAN 接收代码 1 寄存器	0x044 CAN_RXCODE1
0x048	CAN_RXCODE2	CAN 接收代码 2 寄存器	0x048 CAN_RXCODE2
0x04C	CAN_RXCODE3	CAN 接收代码 3 寄存器	0x04C CAN_RXCODE3
0x050	CAN_RXMASK0	CAN 接收屏蔽 0 寄存器	0x050 CAN_RXMASK0
0x054	CAN_RXMASK1	CAN 接收屏蔽 1 寄存器	0x054 CAN_RXMASK1
0x058	CAN_RXMASK2	CAN 接收屏蔽 2 寄存器	0x058 CAN_RXMASK2
0x05C	CAN_RXMASK3	CAN 接收屏蔽 3 寄存器	0x05C CAN_RXMASK3
0x074	CAN_RXC	CAN 接收数据计数寄存器	0x074 CAN_RXC
0x078	CAN_RSADDR	CAN 接收缓冲起始地址寄存器	0x078 CAN_RSADDR
0x080	CAN_RXFIFO	CAN 接收 FIFO 寄存器	0x080 CAN_RXFIFO
0x180	CAN_TXB_RO	CAN 发送缓冲只读寄存器	0x180 CAN_TXB_RO

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 12–13 寄存器列表 (续)

偏移地址	缩写	寄存器描述	章节
0xFFC	CAN_VERSION	CAN 版本寄存器	0xFFC CAN_VERSION

12.4.5. 寄存器描述

12.4.5.1. 0x000 CAN_MODE

默认值：0x00000001			CAN 模式寄存器 (CAN_MODE)
位域	类型	默认值	描述
31:5	–	–	–
4	R/W	0x0	SLEEP_MOD 休眠模式 (Sleep Mode) • 0x0：正常操作，控制器唤醒。 • 0x1：休眠模式，CAN 无总线活动和无中断发生。 **注：** 仅复位状态可写
3	R/W	0x0	FILTER_MOD 接收过滤器模式 (Acceptance Filter Mode) • 0x0：双过滤模式，使用两个过滤器。 • 0x1：单过滤模式，使用一个 4 字节过滤器。
2	R/W	0x0	SELFTEST_MOD，。 自测模式 (Self Test Mode) • 0x0：正常操作，发送成功需有应答。 • 0x1：自测模式，在 SELF_REQ 自接收请求指令下，无需其它节点参与也能完成整个节点测试，发送成功无需应答。
1	R/W	0x0	LISTEN_MOD。 只听模式 (Listen Only Mode) • 0x0：正常操作，错误计数停止于当前值。 • 0x1：只听模式，只接收总线上数据，不产生应答信号，也不更新接收错误计数。
0	R/W	0x1	RST_MOD 复位模式 (Reset Mode)

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000001			CAN 模式寄存器 (CAN_MODE)
位域	类型	默认值	描述
			• 0x0：正常操作,控制器返回正常操作模式。 • 0x1：复位模式,发送或接收中止，控制器处于可配置状态。

12.4.5.2. 0x004 CAN_MCR

默认值：0x00000000			CAN 控制寄存器 (CAN_MCR)
位域	类型	默认值	描述
31:6	–	–	–
4	WO	0x0	SELF_REQ 自接收请求 (Self Reception Request) • 0x0：无动作。 • 0x1：允许发送数据的同时接收总线上的数据。
3	WO	0x0	CLR_OVF_FLAG 清除溢出标志 (Clear Overflow Flag) • 0x0：无动作。 • 0x1：清除数据溢出标志位。
2	WO	0x0	RXB_REL 释放接收缓冲器 (Release Receive Buffer) • 0x0：无动作。 • 0x1：释放，缓冲器中存放信息的内存空间被释放。
1	WO	0x0	ABORT_REQ 中止发送 (Abort Transmission) • 0x0：无动作。 • 0x1：中止当前未开始的发送任务。
0	WO	0x0	TX_REQ 发送请求 (Transmission Request) • 0x0：无动作。 • 0x1：信息被发送。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.4.5.3. 0x008 CAN_STAT

默认值：0x0000003C			CAN 状态寄存器 (CAN_STAT)
位域	类型	默认值	描述
31:8	–	–	–
7	RO	0x0	BUS_STAT 总线状态 (Bus Status) • 0x0：正常状态,控制器参与总线活动。 • 0x1：Bus Off 离线状态,控制器不参与总线活动。
6	RO	0x0	ERR_STAT 错误状态 (Error Status) • 0x0：所有错误计数器低于警报限制值。 • 0x1：至少一个错误计数器高于警报限制值。
5	RO	0x1	TX_STAT 发送状态 (Transmit Status) • 0x0：无任何发送。 • 0x1：正在发送数据。
4	RO	0x1	RX_STAT 接收状态 (Receive Status) • 0x0：无任何接收。 • 0x1：正在接收数据。
3	RO	0x1	TXC_STAT 发送完成状态 (Transmission Complete Status) • 0x0：没有完成发送。 • 0x1：成功完成发送。
2	RO	0x1	TXB_STAT 发送缓冲状态 (Transmit Buffer Status) • 0x0：发送缓冲不可用，发送缓冲已锁定，不可访问。 • 0x1：发送缓冲可用，发送缓冲已释放，可写入数据。
1	RO	0x0	OVF_FLAG 溢出状态标志 (Overflow Flag) • 0x0：数据没有溢出。 • 0x1：数据溢出。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x0000003C			CAN 状态寄存器 (CAN_STAT)
位域	类型	默认值	描述
0	RO	0x0	RXB_STAT 接收缓冲状态 (Receive Buffer Status) • 0x0：接收缓冲为空，无数据可读。 • 0x1：接收缓冲不为空，有数据可读。

12.4.5.4. 0x00C CAN_INTR

默认值：0x00000000			CAN 中断寄存器 (CAN_INTR)
位域	类型	默认值	描述
31:8	–	–	–
7	R/RC	0x0	ERRB_INT 总线错误中断 (Bus Error Interrupt) 节点检测到总线上发生了错误 读中断寄存器 CAN_INTR 任意位，清除该中断
6	R/RC	0x0	ARBLOST_INT 仲裁丢失中断 (Arbitration Lost Interrupt) 发送节点仲裁丢失 读中断寄存器 CAN_INTR 任意位，清除该中断
5	R/RC	0x0	ERRP_INT 被动错误中断 (Error Passive Interrupt) 节点在主动错误与被动错误状态间发生了切换，触发 读中断寄存器 CAN_INTR 任意位，清除该中断
4	RC	0x0	WAKEUP 唤醒中断 (Wake-Up Interrupt) 读中断寄存器 CAN_INTR 任意位，清除该中断
3	R/RC	0x0	OVF_INT 溢出中断 (Overflow Interrupt) 当前接收缓冲中存储的报文为溢出失效报文，触发 读中断寄存器 CAN_INTR 任意位，清除该中断
2	R/RC	0x0	ERRW_INT 错误报警中断 (Error Warning Interrupt)

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			CAN 中断寄存器 (CAN_INTR)
位域	类型	默认值	描述
			错误状态和总线状态发生变化，触发 读中断寄存器 CAN_INTR 任意位，清除该中断
1	R/RC	0x0	TX_INT 发送中断 (Transmit Interrupt) 数据发送任务结束，可以执行新的数据发送任务，触发 读中断寄存器 CAN_INTR 任意位，清除该中断
0	R	0x0	RX_INT 接收中断 (Receive Interrupt) RXB_STAT=1 接收缓冲不为空时触发，有数据待处理 对控制寄存器 CAN_MCR[2] 写 1，清除该中断

12.4.5.5. 0x010 CAN_INTEN

默认值：0x00000000			CAN 中断使能寄存器 (CAN_INTEN)
位域	类型	默认值	描述
31:8	–	–	–
7	R/W	0x0	ERRB_EN 总线错误中断使能 (Bus Error Interrupt Enable)
6	R/W	0x0	ARBLOST_EN 仲裁丢失中断使能 (Arbitration Lost Interrupt Enable)
5	R/W	0x0	ERRP_EN 被动错误中断使能 (Error Passive Interrupt Enable)
4	R/W	0x0	WAKEUP_EN 唤醒中断使能 (Wake-Up Interrupt Enable)
3	R/W	0x0	OVF_EN 溢出中断使能 (Overflow Interrupt Enable)
2	R/W	0x0	ERRW_EN 错误报警中断使能 (Error Warning Interrupt Enable)
1	R/W	0x0	TXI_EN 发送中断使能 (Transmit Interrupt Enable)

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			CAN 中断使能寄存器 (CAN_INTEN)
位域	类型	默认值	描述
0	R/W	0x0	RXI_EN 接收中断使能 (Receive Interrupt Enable)

12.4.5.6. 0x018 CAN_BTR0

默认值：0x00000000			CAN 总线时序 0 寄存器 (CAN_BTR0)
位域	类型	默认值	描述
30:8	–	–	–
7:6	R/W	0x0	SJW 重新同步跳转宽度 (Synchronization Jump Width) • 0x0：1Tq • 0x1：2Tq • 0x2：3Tq • 0x3：4Tq **注：** 仅复位模式可写
5:0	R/W	0x0	BRP 波特率预分频 (Baud Rate Prescaler)，$T_q = 2 \times (BRP[5:0] + 1) \times T_{pclk}$ **注：** 仅复位模式可写

12.4.5.7. 0x01C CAN_BTR1

默认值：0x00000000			CAN 总线时序 1 寄存器 (CAN_BTR1)
位域	类型	默认值	描述
30:24	–	–	–
7	R/W	0x0	SAM 采样次数 (Sample Time)

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			CAN 总线时序 1 寄存器 (CAN_BTR1)
位域	类型	默认值	描述
			• 0x0：采样 1 次，用于高速总线 • 0x1：采样 3 次，用于低和中速总线。 **注：** 仅复位模式可写
6:4	R/W	0x0	TS2 缓冲时间段 2 的宽度 [0x0~0x7]：1~8T_q **注：** 仅复位模式可写
3:0	R/W	0x0	TS1 缓冲时间段 1 的宽度 [0x0~0xF]：1~16T_q **注：** 仅复位模式可写

12.4.5.8. 0x02C CAN_ARBLOST

默认值：0x00000000			CAN 仲裁丢失捕获寄存器 (CAN_ARBLOST)
位域	类型	默认值	描述
31:5	–	–	–
4:0	RO	0x00	ARBLOST_CAP 仲裁丢失捕获 (Arbitration Lost Capture)，记录丢失 bit 的位置 • 0x00~0x0A：SFF ID10~ID0 或者 EFF ID28~ID18 仲裁丢失 • 0x0B：SRTR 仲裁丢失 • 0x0C：IDE 仲裁丢失 • 0x0D~0x1E：EFF ID17~ID0 仲裁丢失 • 0x1F：RTR 仲裁丢失

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.4.5.9. 0x030 CAN_ERRCODE

默认值：0x00000000			CAN 错误代码捕获寄存器 (CAN_ERRCODE)
位域	类型	默认值	描述
31:8	–	–	–
7:6	RO	0x0	ERR_TYPE 错误类型 (Error Type) • 0x0：位错误 • 0x1：格式错误 • 0x2：填充错误 • 0x3：其它类型错误
5	RO	0x0	ERR_DIR (Error Direction) 错误发生时数据传输方向 • 0x0：错误发生于发送 • 0x1：错误发生于接收
4:0	RO	0x00	ERR_CODE 错误发生段代码 (Error Segment Code) • 0x03：SOF，帧起始 • 0x02：ID28~ID21，标识符 • 0x06：ID20~ID18，标识符 • 0x04：SRTR，代替远程发送请求位 • 0x05：IDE，标识符扩展位 • 0x07：ID17~ID13，标识符 • 0x0F：ID12~ID5，标识符 • 0x0E：ID4~ID0，标识符 • 0x0C：RTR，远程发送请求位 • 0x0D：R1，保留位 • 0x09：R0，保留位 • 0x0B：DLC，数据长度代码 • 0x0A：数据场 • 0x08：CRC，CRC 序列 • 0x18：CRC 分界符 • 0x19：ASK，应答槽 • 0x1B：ASK 分界符

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			CAN 错误代码捕获寄存器 (CAN_ERRCODE)
位域	类型	默认值	描述
			• 0x1A：EOF，帧结尾 • 0x12：间隔帧 • 0x11：主动错误标志 • 0x16：被动错误标志 • 0x13：兼容显性位 • 0x17：错误分界符 • 0x1C：过载标志

12.4.5.10. 0x034 CAN_ERRWT

默认值：0x00000060			CAN 错误报警阈值寄存器 (CAN_ERRWT)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x60	ERRWT 错误报警阈值 (Error Warning Threshold) **注：** 正常模式为只读状态，复位模式下可读写。任一错误计数值超过该阈值，触发错误报警中断（使能信号有效情况下）

12.4.5.11. 0x038 CAN_RXERR

默认值：0x00000000			CAN 接收错误计数寄存器 (CAN_RXERR)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	REC 接收错误计数 (Receive Error Counter) **注：** 正常模式为只读状态，复位模式下可读写。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.4.5.12. 0x03C CAN_TXERR

默认值：0x00000000			CAN 发送错误计数寄存器 (CAN_TXERR)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	TEC 发送错误计数 (Transmit Error Counter) **注：** 正常模式为只读状态，复位模式下可读写。

12.4.5.13. 0x040 CAN_BUF0

默认值：0x00000000			CAN 缓冲 0 寄存器 (CAN_BUF0)
位域	类型	默认值	描述
31:8	–	–	–
7	R/W	0x0	IDE 标识符扩展位 (Identifier Extended)，用于明确报文格式 • 0x0：标准格式，SFF • 0x1：扩展格式，EFF **注：** 对于接收缓冲只读，发送缓冲只写，复位模式时可读写。
6	R/W	0x0	RTR 远程发送请求位 (Remote Transmit Request) • 0x0：数据帧 • 0x1：远程帧 **注：** 对于接收缓冲只读，发送缓冲只写，复位模式时可读写。
5:4	–	–	–
3:0	R/W	0x0	DLC 数据长度代码 (Data Length Code)

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			CAN 缓冲 0 寄存器 (CAN_BUF0)
位域	类型	默认值	描述
			• 0x0：无数据内容 • 0x1~0x8：数据内容 **注：** 对于接收缓冲只读，发送缓冲只写，复位模式时可读写。

12.4.5.14. 0x044 CAN_BUF1

默认值：0x00000000			CAN 缓冲 1 寄存器 (CAN_BUF1)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	ID • 标准格式标识符 [10:3] • 扩展格式标识符[28:21] **注：** 对于接收缓冲只读，发送缓冲只写，复位模式时可读写。

12.4.5.15. 0x048 CAN_BUF2

默认值：0x00000000			CAN 缓冲 2 寄存器 (CAN_BUF2)
位域	类型	默认值	描述
31:8	–	–	–
7:5	R/W	0x0	STID_EXID • 标准格式标识 [2:0] • 扩展格式标识符[20:18] **注：** 对于接收缓冲只读，发送缓冲只写，复位模式时可读写。
4:0	R/W	0x0	EXID

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			CAN 缓冲 2 寄存器 (CAN_BUF2)
位域	类型	默认值	描述
			• 扩展格式标识符[17:13] **注：** 对于接收缓冲只读，发送缓冲只写，复位模式时可读写。

12.4.5.16. 0x04C CAN_BUF3

默认值：0x00000000			CAN 缓冲 3 寄存器 (CAN_BUF3)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	STDATA1_EXID • EXID[12:5]，扩展格式标识符[12:5] • STDATA1[7:0]，标准格式数据字节 1 **注：** 对于接收缓冲只读，发送缓冲只写，复位模式时可读写。

12.4.5.17. 0x050 CAN_BUF4

默认值：0x00000000			CAN 缓冲 4 寄存器 (CAN_BUF4)
位域	类型	默认值	描述
31:8	–	–	–
7:3	R/W	0x0	STDATA2_EXID • EXID[4:0]，扩展格式标识符[4:0] • STDATA2[7:3] ，标准格式数据字节 2[7:3] **注：** 对于接收缓冲只读，发送缓冲只写，复位模式时可读写。
2:0	R/W	0x0	STDATA2 • STDATA2[2:0]，标准格式数据字节 2[2:0]

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			CAN 缓冲 4 寄存器 (CAN_BUF4)
位域	类型	默认值	描述
			注： 对于接收缓冲只读，发送缓冲只写，复位模式时可读写。

12.4.5.18. 0x054 CAN_BUF5

默认值：0x00000000			CAN 缓冲 5 寄存器 (CAN_BUF5)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	STDATA3_EXDATA1 • STDATA3[7:0]，标准格式数据字节 3 • EXDATA1[7:0]，扩展格式数据字节 1 注： 对于接收缓冲只读，发送缓冲只写，复位模式时可读写。

12.4.5.19. 0x058 CAN_BUF6

默认值：0x00000000			CAN 缓冲 6 寄存器 (CAN_BUF6)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	STDATA4_EXDATA2 • STDATA4[7:0]，标准格式数据字节 4 • EXDATA2[7:0]，扩展格式数据字节 2 注： 对于接收缓冲只读，发送缓冲只写，复位模式时可读写。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.4.5.20. 0x05C CAN_BUF7

默认值：0x00000000			CAN 缓冲 7 寄存器 (CAN_BUF7)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	STDATA5_EXDATA3 • STDATA5[7:0]，标准格式数据字节 5 • EXDATA3[7:0]，扩展格式数据字节 3 **注：** 对于接收缓冲只读，发送缓冲只写，复位模式时可读写。

12.4.5.21. 0x060 CAN_BUF8

默认值：0x00000000			CAN 缓冲 8 寄存器 (CAN_BUF8)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	STDATA6_EXDATA4 • STDATA6[7:0]，标准格式数据字节 6 • EXDATA4[7:0]，扩展格式数据字节 4 **注：** 对于接收缓冲只读，发送缓冲只写，复位模式时可读写。

12.4.5.22. 0x064 CAN_BUF9

默认值：0x00000000			CAN 缓冲 9 寄存器 (CAN_BUF9)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	STDATA7_EXDATA5 • STDATA7[7:0]，标准格式数据字节 7 • EXDATA5[7:0]，扩展格式数据字节 5

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			CAN 缓冲 9 寄存器 (CAN_BUF9)
位域	类型	默认值	描述
			注： 对于接收缓冲只读，发送缓冲只写，复位模式时可读写。

12.4.5.23. 0x068 CAN_BUF10

默认值：0x00000000			CAN 缓冲 10 寄存器 (CAN_BUF10)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	STDATA8_EXDATA6 • STDATA8[7:0]，标准格式数据字节 8 • EXDATA6[7:0]，扩展格式数据字节 6 注： 对于接收缓冲只读，发送缓冲只写，复位模式时可读写。

12.4.5.24. 0x06C CAN_BUF11

默认值：0x00000000			CAN 缓冲 11 寄存器 (CAN_BUF11)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	EXDATA7 • EXDATA7[7:0]，扩展格式数据字节 7 注： 对于接收缓冲只读，发送缓冲只写，复位模式时可读写。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.4.5.25. 0x070 CAN_BUF12

默认值：0x00000000			CAN 缓冲 12 寄存器 (CAN_BUF12)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	EXDATA8 • EXDATA8[7:0]，扩展格式数据字节 8 **注：** 对于接收缓冲只读，发送缓冲只写，复位模式时可读写。

12.4.5.26. 0x040 CAN_RXCODE0

默认值：0x00000000			CAN 接收代码 0 寄存器 (CAN_RXCODE0)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	RXCODE0 接收过滤代码 0 (Acceptance Code0)，详细描述见 接收过滤器 **注：** 正常模式只读，复位模式时可读写。

12.4.5.27. 0x044 CAN_RXCODE1

默认值：0x00000000			CAN 接收代码 1 寄存器 (CAN_RXCODE1)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	RXCODE1 接收过滤代码 1 (Acceptance Code1)，详细描述见 接收过滤器 **注：** 正常模式只读，复位模式时可读写。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.4.5.28. 0x048 CAN_RXCODE2

默认值：0x00000000			CAN 接收代码 2 寄存器 (CAN_RXCODE2)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	RXCODE2 接收过滤代码 2 (Acceptance Code2)，详细描述见 接收过滤器 **注：** 正常模式只读，复位模式时可读写。

12.4.5.29. 0x04C CAN_RXCODE3

默认值：0x00000000			CAN 接收代码 3 寄存器 (CAN_RXCODE3)
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	RXCODE3 接收过滤代码 3 (Acceptance Code3)，详细描述见 接收过滤器 **注：** 正常模式只读，复位模式时可读写。

12.4.5.30. 0x050 CAN_RXMASK0

默认值：0x00000000			CAN 接收屏蔽 0 寄存器 (CAN_RXMASK0)
位域	类型	默认值	描述
7:0	R/W	0x0	RXMASK0 接收屏蔽 0 (Acceptance Mask0)，详细描述见 接收过滤器 **注：** 正常模式只读，复位模式时可读写。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.4.5.31. 0x054 CAN_RXMASK1

默认值：0x00000000			CAN 接收屏蔽 1 寄存器 (CAN_RXMASK1)
位域	类型	默认值	描述
7:0	R/W	0x0	RXMASK1 接收屏蔽 1 (Acceptance Mask1)，详细描述见 接收过滤器 **注：** 正常模式只读，复位模式时可读写。

12.4.5.32. 0x058 CAN_RXMASK2

默认值：0x00000000			CAN 接收屏蔽 2 寄存器 (CAN_RXMASK2)
位域	类型	默认值	描述
7:0	R/W	0x0	RXMASK2 接收屏蔽 2 (Acceptance Mask2)，详细描述见 接收过滤器 **注：** 正常模式只读，复位模式时可读写。

12.4.5.33. 0x05C CAN_RXMASK3

默认值：0x00000000			CAN 接收屏蔽 3 寄存器 (CAN_RXMASK3)
位域	类型	默认值	描述
7:0	R/W	0x0	RXMASK3 接收屏蔽 3 (Acceptance Mask3)，详细描述见接收过滤器 **注：** 正常模式只读，复位模式时可读写。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.4.5.34. 0x074 CAN_RXC

默认值：0x00000000			CAN 接收数据计数寄存器 (CAN_RXC)
位域	类型	默认值	描述
31:8	–	–	–
7:0	RO	0x0	RXC 接收数据计数 (Receive Message Counter)，存储接收缓冲中数据包个数。

12.4.5.35. 0x078 CAN_RSADDR

默认值：0x00000000			CAN 接收缓冲起始地址寄存器 (CAN_RSADDR)
位域	类型	默认值	描述
31:6	–	–	–
5:0	R/W	0x0	RSADDR 接收缓冲器起始地址 (Receive Buffer Start Address) **注：** 正常模式只读，复位模式时可读写。

12.4.5.36. 0x080 CAN_RXFIFO

默认值：0x00000000			CAN 接收 FIFO 寄存器 (RX FIFO)
位域	类型	默认值	描述
31:8	–	–	–
7:0	RO	0x0	RXFIFO 接收 FIFO (Receive FIFO)，0x80~0x17C 共 64 bytes **注：** 正常模式只读，复位模式时可读写。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.4.5.37. 0x180 CAN_TXB_RO

默认值：0x00000000			CAN 发送缓冲只读寄存器 (TX Buffer Read Only)
位域	类型	默认值	描述
31:8	–	–	–
7:0	RO	0x0	TXB_RO 发送缓冲只读寄存器 (Transmit Buffer Read Only) 每个寄存器只有低 8 位可有效访问，其余高位被读时返回 0。

12.4.5.38. 0xFFC CAN_VERSION

默认值：0x00000000			CAN 版本寄存器 (VERSION)
位域	类型	默认值	描述
31:0	RO	0x00000000	Version 版本号，采用 BCD 码显示，V1.0

12.5. Consumer Infrared Radiation (CIR)

Consumer IR (CIR) 模块包含发送器和接收器，其特殊设计支持生成或捕获各类数字脉冲信号。模块使用游程编码 (RLC，Run-Length Code) 方式编码数字脉冲信号，以 Byte 为单位记录编码数据，其中：

- MSB 位表示信号电平：
  - 1：高电平
  - 0：低电平
- 其余 7 位表示信号宽度，以采样时钟为单位。

  最大宽度为 128，假如某一电平宽度大于 128，则使用另一字节存储。

12.5.1. 特性说明

- 全物理层执行。
- 支持 CIR 远程控制或无线键盘。
- 64 levels × 8 bits FIFO 接收缓存，可编程 FIFO 阈值。
- 128 bytes FIFO 发送缓存，可编程载波频率及占空比，支持任意波形发生。
- 支持中断。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.5.2. 原理框图

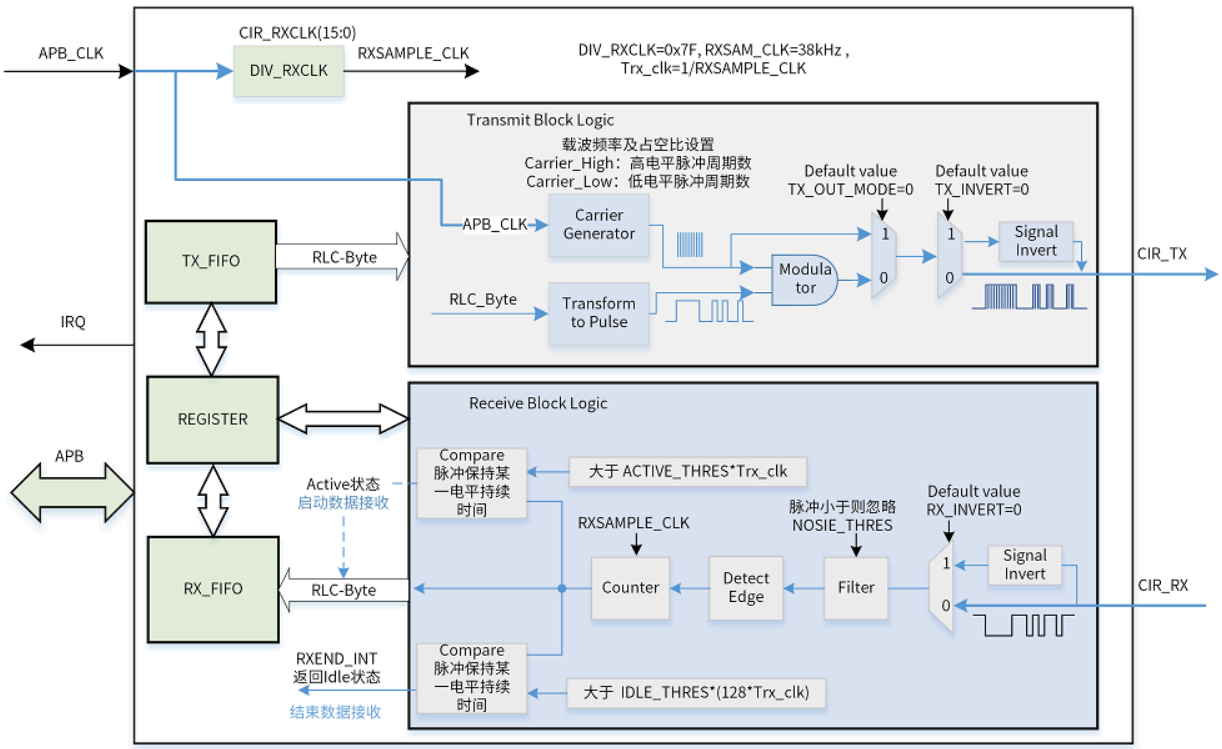


图 12–18 CIR 模块原理框图

12.5.3. 功能描述

12.5.3.1. 典型应用

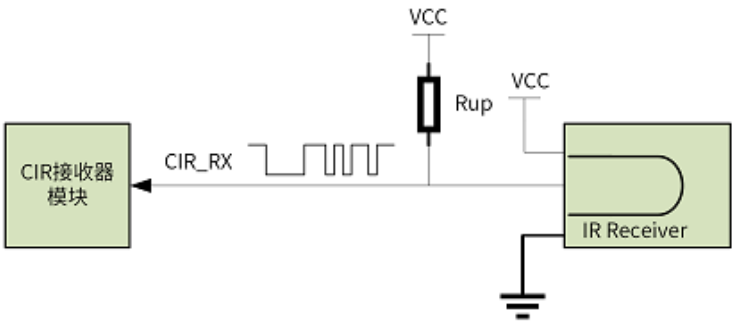


图 12–19 CIR 接收器典型应用

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

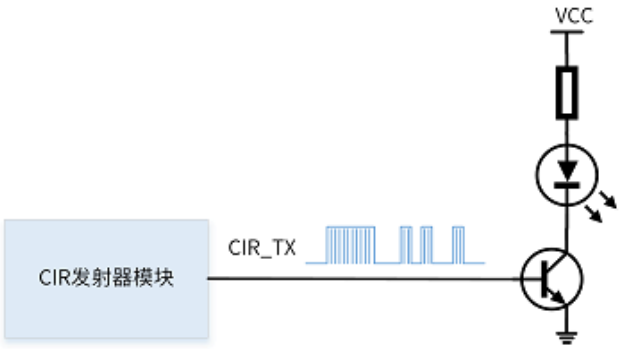


图 12-20 CIR 发送器典型应用

12.5.3.2. 功能实现

CIR 模块可以实现以下功能：

- CIR_RX 接收器，可视为具有捕获功能的定时器。

  以 NEC 编码协议为例说明如下，当 CIR_RX 信号输入：

  - 若脉冲（高电平脉冲或低电平脉冲）宽度小于噪声阈值（NOSIE_THRES），则忽略 CIR_RX 信号。
  - 当脉冲保持某一电平持续时间满足启动阈值时 (ACTIVE_THRES)，开始数据接收和存储。
  - 当脉冲保持某一电平持续时间满足空闲阈值时 (IDLE_THRES)，本次接收结束，接收器产生 RXEND_INT 中断，并返回 Idle 空闲状态。
  - 在捕获过程中，CIR_RX 信号的每个有效高低电平脉冲，都将按照游程编码方式 (Run-Length Code, RLC))，以 Byte 为单位缓存到 RX_FIFO。

    RLC-Byte（8 位）编码方式：

    - MSB 位表示信号电平，高电平表示 1，低电平表示 0。
    - 其余 7 位以采样时钟（Sample Clock）为基本单位，表示脉冲宽度。
    - 当电平发生变化或脉冲宽度计数溢出时，RLC-Byte 将被缓存到 RX_FIFO，脉冲宽度计数最大 128，超出 128 时使用另一字节缓存。

图 12-21 NEC 协议

- CIR_TX 发送器，可视为具有载波调制功能的任意宽度波形发生器。

Copyright © 2023-2025 ArtInChip Technology Co., Ltd.
All rights reserved

- CIR 发送器从 TX_FIFO 中读取连续的数据码，通过格式及逻辑转换，数据码可变成特定宽度的数字脉冲序列。数据码可编程，以游程编码 RLC-byte 方式存储于 FIFO 中。
- 载波发生器默认生成 38 KHz 载波，与转换得到的数字脉冲序列进行调制，输出调制信号 CIR_TX。

  载波发生器时钟源选择 APB_CLK，可通过高电平脉冲周期数（CARRIER_HIGH）和低电平脉冲周期数（CARRIER_LOW）配置载波频率和占空比，生成任意频率与宽度的数字波形，通过 TX_OUT_MODE（Transmit Out Mode），选择发送输出模式（默认为 0，调制输出）。

12.5.3.3. 操作模式

采样时钟 (Sample_CLK)，APB 时钟经过 CIR_CLK 时钟寄存器分频配置得出，用于 RLC-Byte，表示脉冲宽度的基本单位。

CIR_RX 接收器操作模式

- CIR_RX 接收器采样说明，以 NEC 协议、RC-5 协议为例：

  举例 1：假设采样时钟 38 KHz，即采样周期为 1/38 KHz = 26.3 us。

  - 对于 NEC 协议逻辑 0，高低电平都持续 21 cycles，采样得到 2 个 RLC-Byte，分别为高电平 1001 0101（0x95）、低电平 0001 0101（0x15）。
  - 对于 NEC 协议逻辑 1，高电平持续 21 cycles，低电平持续 64 cycles，采样得到 2 个 RLC-Byte，分别为高电平 1001 0101（0x95）、低电平 0100 0000（0x40），其中高位 bit7 表示电平极性，bit6: 0 表示脉冲宽度。

  举例 2：假设采样时钟 36 KHz，即采样周期为 1/36 KHz = 27.778 us。

  - 对于 RC-5 协议逻辑 0，高低电平都持续 32 cycles，采样得到 2 个 RLC-Byte，分别为高电平 1010 0000（0xA0）、低电平 0010 0000（0x20）。
  - 对于 RC-5 协议逻辑 1，采样得到 2 个 RLC-Byte，分别为低电平 0010 0000（0x20）、高电平 1010 0000（0xA0），其中高位 bit7 表示电平极性，bit6: 0 表示脉冲宽度。

- ACTIVE_THRES（Active Threshold，启动阈值）：当 CIR 接收器模块处于 Idle 空闲状态，若 CIR_RX 信号电平发生变化，且持续时间达到该阈值，CIR 接收器模块认为这是一个有效激活码，将转变为 Active 状态，开始捕获 CIR_RX 信号。
- NOISE_THRES（Noise Threshold，噪声阈值）：CIR_RX 信号输入时，宽度小于该阈值的脉冲信号，被当作是无效短脉冲，都将被忽略。
- IDLE_THRES（Idle Threshold，空闲阈值）：若 CIR_RX 信号保持某一电平无变化，且持续时间达到该阈值，CIR 接收器模块进入 Idle 空闲状态并结束本次捕获。
- RX_INVERT：RX 输入信号可配置是否翻转反相，默认 RX_INVERT=0，不反相

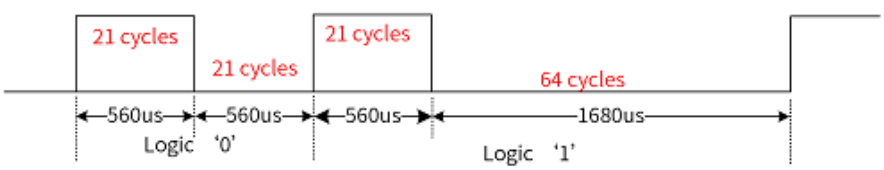


图 12-22 NEC 协议逻辑‘0’和逻辑‘1’

Copyright © 2023-2025 ArtInChip Technology Co., Ltd.
All rights reserved

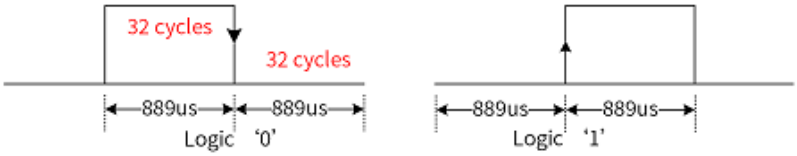


图 12–23 RC–5 协议逻辑‘0’和逻辑‘1’

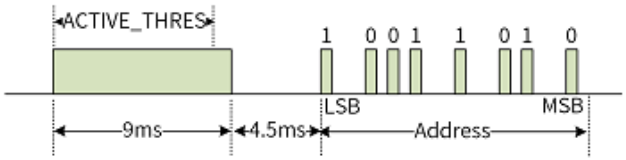


图 12–24 Active Threshold 启动阈值

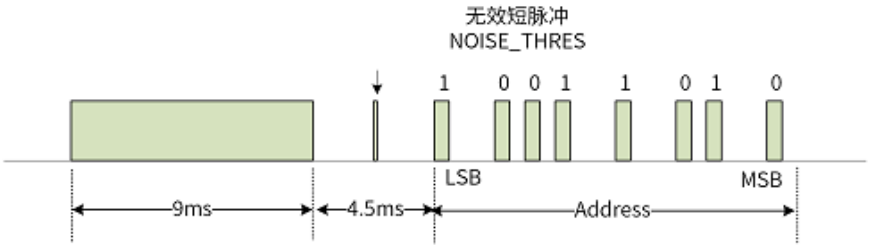


图 12–25 Noise Threshold 噪声阈值

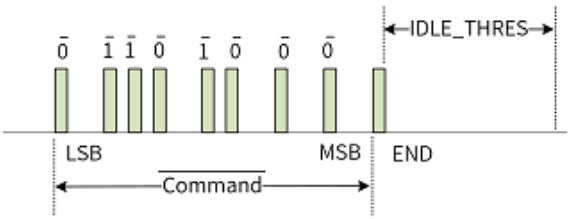


图 12–26 Idle Threshold 空闲阈值

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

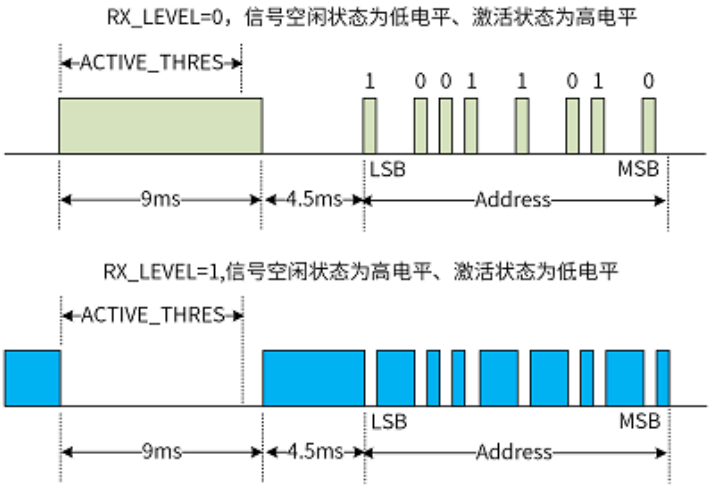


图 12–27 RX 输入信号电平选择

CIR_TX 发送器操作模式

- TX_OUT_MODE：Transmit Out Mode，发送输出模式。选择是否调制输出。
- CARRIER_HIGH 载波高电平脉冲周期数。CARRIER_LOW 载波低电平脉冲周期数。以 APB_CLK 为时钟，用于配置载波频率及占空比。
- TX_MODE：Transmit Mode，发送模式。CIR 发送器有两种发送模式，一种是非循环发送，另一种是循环发送。
  - 当 CIR_EN 置为 1 使能 CIR 模块时，发生器内部的时钟分频逻辑等开始动作，但只有 TX_START 为 1 时，发送器才会开始从 TX_FIFO 中读取数据。每读取一次 TX_FIFO，可以获得一字节数据。
  - 非循环发送，是将 TX_FIFO 中的数据全部发送出去，直至 FIFO 空。发送过程中，反馈 TX FIFO 空、满状态，软件可以在检测空满状态后是否继续往 FIFO 中填充数据。发送器结束工作产生发送结束中断，然后返回空闲状态，空闲状态默认发送的电平为最后数据脉冲电平。
  - 循环发送，是将 FIFO 中的数据全部发送出去，发送完之后等待一定时间之后恢复 FIFO 中的数据然后再次发送（即从第一个数据发送到最后一个数据，再从第一个发送到最后一个，依此类推）。
  - 在循环发送模式下，第 N 次和第 N+1 次之间会有空闲持续时间（即发送间隔时间，IDC_THRES）。
  - 直至检测到循环发送停止信号（CYC_TX_CTRL=0），循环发送停止，由开始到停止，FIFO 中所有数据必须被发送完，完成后产生发送结束中断。其中恢复 FIFO 中的数据是通过将 TX_FIFO 的读指针清零实现。
- TX_STOP：Transmit Stop Control，发送停止控制。TX_STOP =1，循环发送停止，由开始到停止，FIFO 中所有数据必须被发送完。硬件发送完 FIFO 的数据后自动将该 bit 位清 0。
- IDC_THRES：Idle Duration Counter Threshold，空闲持续时间计数阈值 (0~4095)。用于循环发送模式，当发送完 FIFO 中所有数据后，间隔多长时间才再次发送。循环发送间隔时间= IDC_THRES × 128 × Ttx_clk。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

图 12–28 数据定义

ArtInChip

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.5.4. 编程指南

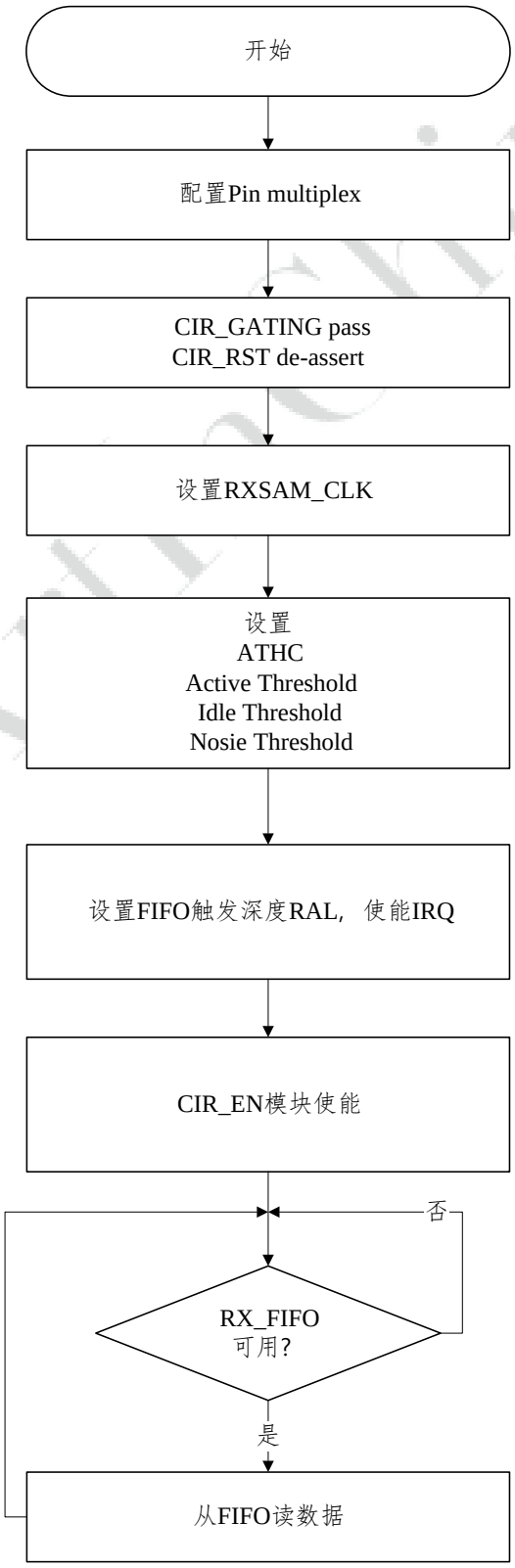


图 12–29 CIR 接收器编程指南

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

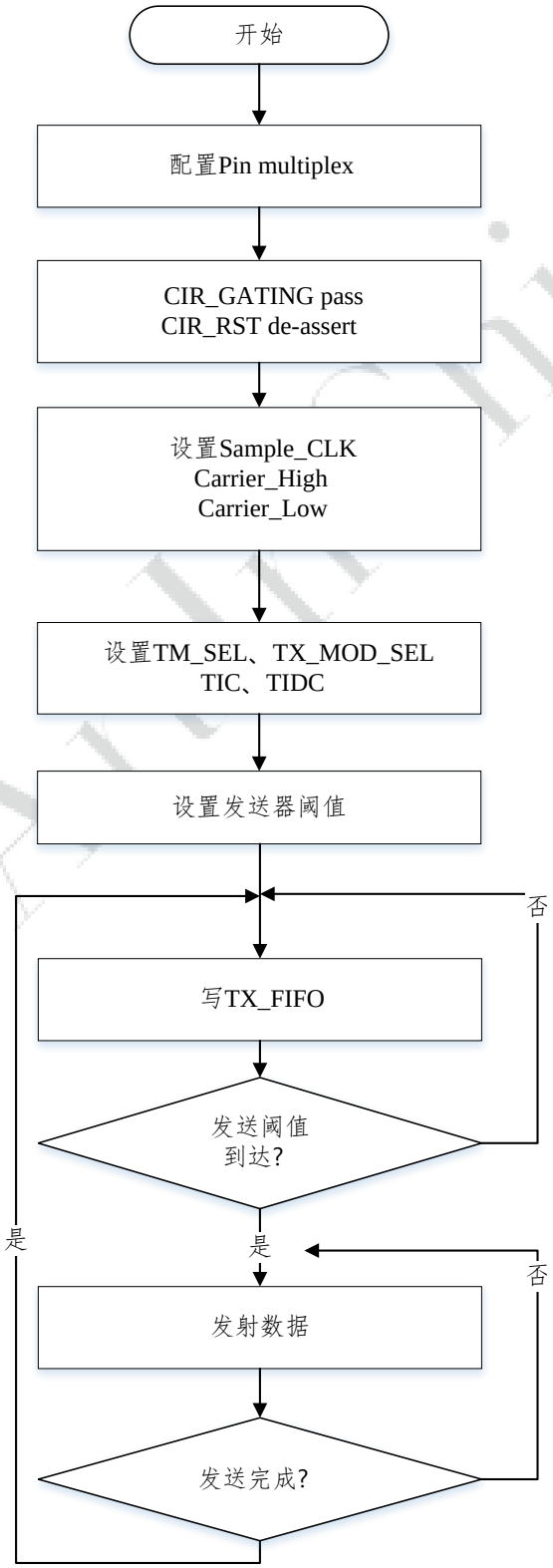


图 12–30 CIR 发送器编程指南

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.5.5. 寄存器列表

寄存器列表

偏移地址	缩写	寄存器名称	章节
0x000	CIR_MCR	CIR 模块控制寄存器	0x000 CIR_MCR
0x004	CIR_INTR	CIR 模块中断控制寄存器	0x004 CIR_INTR
0x008	CIR_INTEN	CIR 模块中断使能寄存器	0x008 CIR_INTEN
0x00C	CIR_TXSTAT	CIR 发送状态寄存器	0x00C CIR_TXSTAT
0x010	CIR_RXSTAT	CIR 接收状态寄存器	0x010 CIR_RXSTAT
0x014	CIR_RXCLK	CIR 接收采样时钟寄存器	0x014 CIR_RXCLK
0x018	CIR_RX_THRES	CIR 接收阈值配置寄存器	0x018 CIR_RX_THRES
0x01C	CIR_RX_CFG	CIR 接收配置寄存器	0x01C CIR_RX_CFG
0x020	CIR_TX_CFG	CIR 发送配置寄存器	0x020 CIR_TX_CFG
0x024	CIR_TIDC	CIR 发送间隔时间寄存器	0x024 CIR_TIDC
0x02C	CIR_CARR_CFG	CIR 载波配置寄存器	0x02c CIR_CARR_CFG
0x030	CIR_RXFIFO	CIR 接收缓存寄存器	0x030 CIR_RXFIFO
0x080	CIR_TXFIFO	CIR 发送缓存寄存器	0x080 CIR_TXFIFO
0xFFC	CIR_VERSION	CIR 模块版本号	0xFFC CIR_VERSION

12.5.6. 寄存器描述

12.5.6.1. 0x000 CIR_MCR

默认值：0x00000000			CIR 控制寄存器（CIR_MCR）
位域	类型	默认值	描述
31:18	–	–	–
17	WAC	0x0	TX_FIFO_CLR TX FIFO CLEAR，清 TX FIFO。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			CIR 控制寄存器（CIR_MCR）
位域	类型	默认值	描述
			• 0x0：未使能。 • 0x1：Clear Tx FIFO。
16	WAC	0x0	RX_FIFO_CLR RX FIFO CLEAR，清 RX FIFO。 • 0x0：未使能。 • 0x1：Clear Tx FIFO。
15:10	–	–	–
9	WAC	0x0	TX_STOP Transmit Stop Control，发送停止控制，仅用于循环模式。 • 0x0：硬件发送完 FIFO 的数据后自动将该 bit 清 0。 • 0x1：循环发送停止。由开始到停止，FIFO 中数据必须被发送完。
8	WAC	0x0	TX_START Transmit Start Control，发送启动控制。在循环模式和非循环模式都需要。 • 0x0：硬件发送完 FIFO 的数据后自动将该 bit 位清 0。 • 0x1：发送开始。由开始到停止，FIFO 中数据必须被发送完。 在循环模式下，FIFO 中的数据会在一定的间隔时间 IDC_THRES 后反复发送，直到 TX_STOP 为 1 时才会在发送完当次命令后停止。
7:2	–	–	–
1	R/W	0x0	CIR_EN CIR Module Enable，CIR 模块使能。 • 0x0：CIR 模块未使能。 • 0x1：CIR 模块使能。
0	–	–	–

12.5.6.2. 0x004 CIR_INTR

默认值：0x00000000			CIR 中断寄存器（CIR_INTR）
位域	类型	默认值	描述
31:7	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			CIR 中断寄存器（CIR_INTR）
位域	类型	默认值	描述
6	R/W1C	0x0	TXB_AVL _INT TX FIFO Available Interrupt Flag，发送缓存可用中断。 • 0x0：发送缓存不可用。 • 0x1：发送缓存可用。达到 TXB_EMPTY_LEVEL 时触发。
5	R/W1C	0x0	TXEND_INT Transmit End Interrupt，发送结束中断。 • 0x0：发送未完成。 • 0x1：发送完成。
4	R/W1C	0x0	TX_UNF_INT Transmit FIFO Underflow Interrupt for Non-cyclical Pulse，非循环发送下溢出中断。 • 0x0：发送器 FIFO 无下溢出。 • 0x1：发送器 FIFO 下溢出。
3	–	–	–
2	R/W1C	0x0	RXB_AVL _INT RX FIFO Available Interrupt Flag，接收缓存可用中断。 • 0x0：接收缓存不可用。 • 0x1：接收缓存可用。达到 LEVEL 时触发。
1	R/W1C	0x0	RX_END_INT Receive End Interrupt，接收结束中断 • 0x0：接收未结束 • 0x1：接收结束
0	R/W1C	0x0	RX_OVF_INT Receive Overflow Interrupt，接收缓存溢出中断 • 0x0：接收器 FIFO 未溢出 • 0x1：接收器 FIFO 溢出

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.5.6.3. 0x008 CIR_INTEN

默认值：0x00000063			CIR 中断使能控制寄存器（CIR_INTEN）
位域	类型	默认值	描述
31:24	–	–	–
23:16	R/W	0x0	TXB_EMPTY_LEVEL TX FIFO Empty Level，发送缓存空闲深度。达到该设置值触发 TXB_AVL_INT。
15:14	–	–	–
13:8	R/W	0x0	RXB_AVL _LEVEL RX FIFO Available Level，接收缓存可用字节深度。达到该设置值触发 RXB_AVL_INT。
7	–	–	–
6	R/W	0x1	TXB_AVL _EN TX FIFO Available Interrupt Enable，发送缓存可用中断使能。
5	R/W	0x1	TXEND_EN Transmit Ended Interrupt Enable，发送结束中断使能。
4	R/W	0x0	TX_UNF_EN Transmit FIFO Underflow Interrupt Enable for non-Cyclical Pulse，非循环发送下溢出中断使能。
3	–	–	–
2	R/W	0x0	RXB_AVL_EN RX FIFO Available Interrupt Enable，接收缓存可用中断使能。
1	R/W	0x1	RXEND_EN Receive Ended Interrupt Enable，接收结束中断使能。
0	R/W	0x1	RX_OVF_EN Receive Overflow Interrupt Enable，接收缓存溢出断使能。

12.5.6.4. 0x00C CIR_TXSTAT

默认值：0x0000017F			CIR 发送状态寄存器（CIR_TXSTAT）
位域	类型	默认值	描述
31:12	–	–	–

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x0000017F			CIR 发送状态寄存器（CIR_TXSTAT）
位域	类型	默认值	描述
11	RO	0x0	TX_STAT Transmitter Status，发送器状态。 • 0x0：Idle，空闲，当 FIFO 中所有数据被发送完时自动清 0。 • 0x1：Active，激活。
10	RO	0x0	TX_FIFO_ERR FIFO Error Flag，发送 FIFO 错误标志。 • 0x0：no error • 0x1：error, such as overflow or underflow
9	RO	0x0	TX_FIFO_FULL FIFO Full Flag，发送 FIFO 满标志。 • 0x0：not full • 0x1：full
8	RO	0x1	TX_FIFO_EMPTY TX FIFO Empty Flag，发送 FIFO 空标志。 • 0x0：not empty • 0x1：empty
7	–	–	–
6:0	RO	0x7F	TX_FIFO_DLEN TX FIFO Data Length，TX FIFO 当前有效数据量指示。 • 0x00：发送缓存中还有 1 byte 待发送 • 0x01：发送缓存中还有 2 bytes 待发送 • … • 0x7F：发送缓存中还有 128 bytes 待发送

12.5.6.5. 0x010 CIR_RXSTAT

默认值：0x00000100			CIR 接收状态寄存器（CIR_RXSTAT）
位域	类型	默认值	描述
31:12	–	–	–
11	RO	0x0	RX_STAT

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000100			CIR 接收状态寄存器（CIR_RXSTAT）
位域	类型	默认值	描述
			Receiver Status，接收器状态。 • 0x0：Idle，空闲 • 0x1：Active，激活
10	RO	0x0	RX_FIFO_ERR FIFO Error Flag，接收 FIFO 错误标志。 • 0x0：no error • 0x1：error, such as overflow or underflow
9	RO	0x0	RX_FIFO_FULL FIFO Full Flag，接收 FIFO 满标志。 • 0x0：not full • 0x1：full
8	RO	0x1	RX_FIFO_EMPTY RX FIFO Empty Flag，接收 FIFO 空标志。 • 0x0：not empty • 0x1：empty
7:6	–	–	–
5:0	RO	0x00	RX_FIFO_DLEN RX FIFO Data Length，RX FIFO 当前有效数据量指示。 • 0x00：接收缓存中还有 1 byte 待取走 • 0x01：接收缓存中还有 2 bytes 待取走 • … • 0x3F：接收缓存中还有 64 bytes 待取走

12.5.6.6. 0x014 CIR_RXCLK

默认值：0x00000000			CIR 接收采样时钟寄存器（CIR_RXCLK）
位域	类型	默认值	描述
31:16	–	–	–
15:0	R/W	0x0	DIV_RXCLK Receive Sample Clock Divider，接收采样时钟分频系数（1~65535）

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.5.6.7. 0x018 CIR_RX_THRES

默认值：0x000F00FF			CIR 接收阈值配置寄存器（CIR_RX_THRES）
位域	类型	默认值	描述
31:16	R/W	0x0F	ACTIVE_THRES Active Threshold，激活阈值 接收器由空闲状态转变到激活状态的阈值。 当接收器处于空闲状态，若输入信号电平发生变化（上升沿或下降沿），且持续时间大于 ACTIVE_THRES × Trx_clk 时，接收器认为这是一个激活码，将转变为激活状态，并开始捕获 CIR_RX 信号。
15:0	R/W	0xFF	IDLE_THRES Idle Threshold，空闲阈值 接收器根据该阈值决定当前指令是否完成被接收。若无 CIR_RX 输入，接收器处于空闲状态。当 CIR_RX 信号保持高电平或低电平，且持续时间大于 IDLE_THRES × Trx_clk 时，表示前一个指令完成接收。

12.5.6.8. 0x01C CIR_RX_CFG

默认值：0x01FF0002			CIR 接收配置寄存器（CIR_RX_CFG）
位域	类型	默认值	描述
31:16	R/W	0x1FF	NOSIE_THRES Nosie Threshold，噪声阈值 CIR_RX 信号脉冲（高电平或低电平）持续时间小于 NOSIE_THRES 个采样周期 Tpclk，该脉冲将被当作噪声干扰而被忽略。 • 0x0：所有采样都存储于 RX_FIFO • 0x1：若信号小于等于 1 个周期 Tpclk，忽略 • 0x2：若信号小于等于 2 个周期 Tpclk，忽略 • … • 0xFFFF：若信号小于等于 65535 个周期 Tpclk，忽略
15:8	–	–	–
7:3	–	–	–
2	W1C	0x0	DBC_LEVEL_SET Debounce 输出初始化电平设置 • 0x0：Debounce 输出初始化电平为 1 • 0x1：Debounce 输出初始化电平与 RX_LEVEL 一致

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x01FF0002			CIR 接收配置寄存器（CIR_RX_CFG）
位域	类型	默认值	描述
1	R/W	0x1	RX_LEVEL RX 输入信号电平 • 0x0：空闲状态下为低电平、激活状态下为高电平 • 0x1：空闲状态下为高电平、激活状态下为低电平
0	R/W	0x0	RX_INVERT RX 信号极性反相 • 0x0：信号不反相 • 0x1：信号反相

12.5.6.9. 0x020 CIR_TX_CFG

默认值：0x00000000			CIR 发送配置寄存器（CIR_TX_CFG）
位域	类型	默认值	描述
31:8	–	–	–
7:3	–	–	–
2	R/W	0x0	TX_MODE Transmit Mode，发送模式 • 0x0：非循环发送模式 • 0x1：循环发送模式
1	R/W	0x0	TX_OUT_MODE Transmit Out Mode，发送输出模式 • 0x0：发送信号调制 • 0x1：发送信号不调制
0	R/W	0x0	TX_INVERT Transmit Signal Invert，TX 信号极性反相 • 0x0：信号不反相 • 0x1：信号反相

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.5.6.10. 0x024 CIR_TIDC

默认值：0x00000000			CIR 发送间隔时间寄存器（CIR_TIDC）
位域	类型	默认值	描述
31:16	–	–	–
15:0	R/W	0x0	IDC_THRES Idle Duration Counter Threshold，空闲持续时间计数阈值，0~65536 循环发送间隔时间= IDC_THRES × Ttx_clk

12.5.6.11. 0x02c CIR_CARR_CFG

默认值：0x0003000b			CIR 发送载波配置寄存器（CIR_CARR_CFG）
位域	类型	默认值	描述
31:16	R/W	0x3	CARRIER_HIGH Carrier High Pulse Number，载波高电平脉冲个数，以 APB_CLK 为基本单位。用于配置载波频率及占空比。 TX_CLK=24MHz/（CARR_HIGH+1+CARR_LOW+1）。 DUTY=（CARR_HIGH+1）/（CARR_HIGH+1+CARR_LOW+1）
15:0	R/W	0xb	CARRIER_LOW Carrier Low Pulse Number，载波低电平脉冲个数，以 APB_CLK 为基本单位。用于配置载波频率及占空比。

12.5.6.12. 0x030 CIR_RXFIFO

默认值：0x00000000			CIR 接收缓存寄存器（CIR_RXFIFO）
位域	类型	默认值	描述
31:8	–	–	–
7:0	R/W	0x0	RXFIFO Receive Byte FIFO，接收字节缓存。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.5.6.13. 0x080 CIR_TXFIFO

默认值：0x00000000			CIR 发送缓存寄存器（CIR_TXFIFO）
位域	类型	默认值	描述
31:8	–	–	–
7:0	W	0x0	TXFIFO Transmit Byte FIFO，发送字节缓存。 当发送触发，FIFO 中的数据将被发送，直至所有数据字节数量都被发送完。

12.5.6.14. 0xFFC CIR_VERSION

默认值：0x00000101			CIR 版本寄存器（VERSION）
位域	类型	默认值	描述
31:0	RO	0x00000101	Version（版本号），采用 BCD 码，V1.1。

12.6. General Purpose Input Output (GPIO)

General Purpose Input Output (GPIO)，通用引脚控制器，用于管理具有通用电气特性的引脚，区别于 USB、DDR 等带专用 PHY 的引脚。

12.6.1. 特性说明

- 每个引脚可单独配置为关闭，通用功能和最多七个专用功能
- 每个引脚可单独使能输出，可单独配置中断触发模式
- 每个引脚驱动能力单独配置，可选有八个档位
- 每个引脚上下拉单独配置，可选无上下拉/ 上拉 33 KΩ/ 下拉 33 KΩ
- 五组引脚 (PA/ PB/ PC/ PD/ PE/ PF))，IO 电源只有一组 VCC33_IO0，使用 3.3 V 供电
- 每组引脚输入中断脉宽可配置，时间可配置为 10 ns~9.83 ms

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

12.6.2. 原理框图

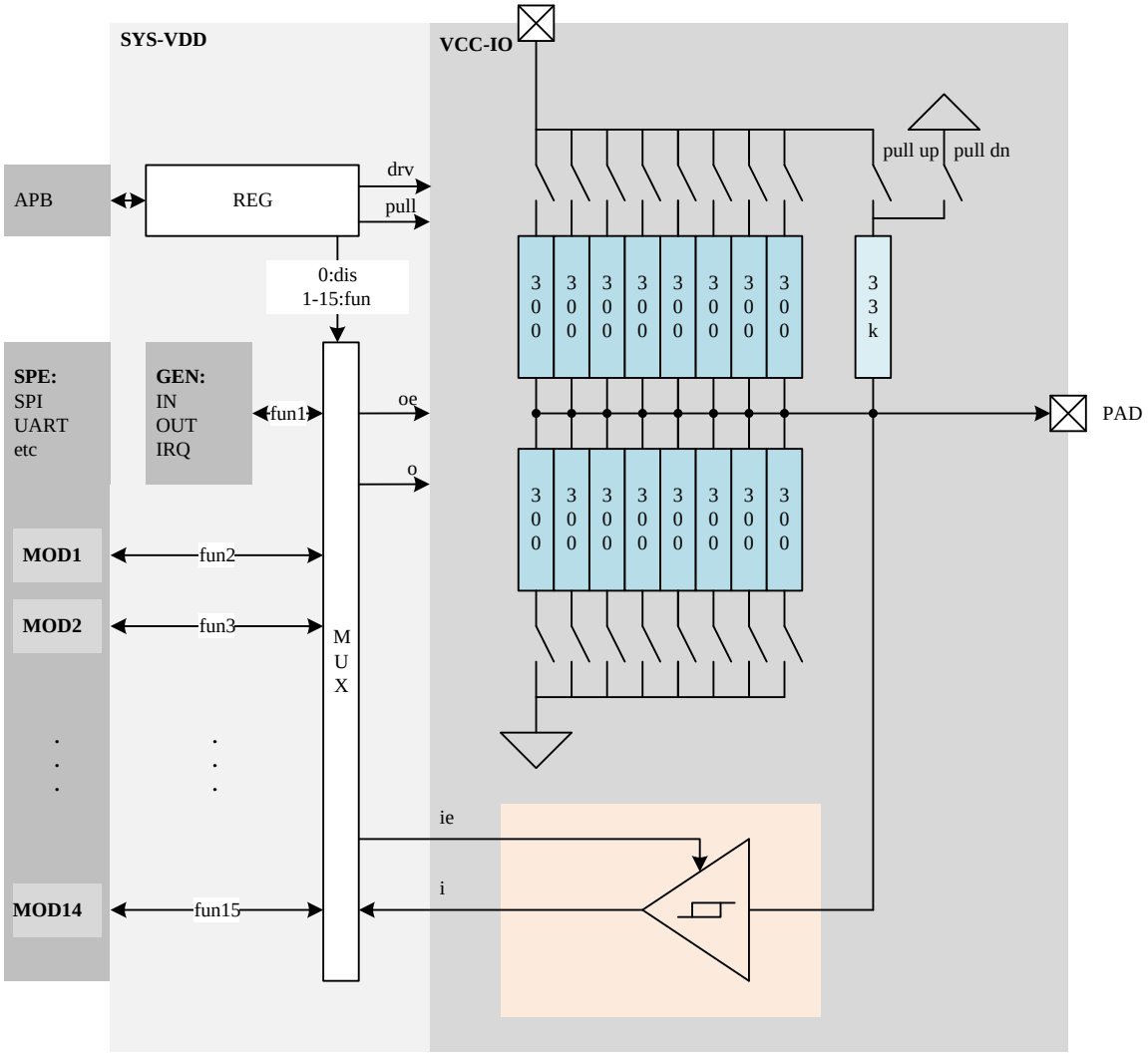


图 12–31 GPIO 原理框图

12.6.3. 功能描述

GPIO 有以下几种配置：

- 关闭
- 通用功能：在 GPIO 模块内部控制。通用功能又分为以下三个功能：
  - 读取输入电平：当引脚配置为通用功能后，使能 GEN_IE，即可在 GEN_IN_STA 对应的位读取到当前的输入电平。
  - 设置输出电平：当引脚配置为通用功能后，使能 GEN_OE，即可配置 GEN_OUT_CFG 使得对应的引脚输出电平。
  - 中断设置：当使用到中断检测功能，同样需要使能 GEN_IE，每个引脚可配置中断模式为：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 下降沿中断
- 上升沿中断
- 低电平中断
- 高电平中断
- 双沿中断（上升沿或下降沿）

通用功能的输入，支持两级去抖，详情可查看通用输入去抖功能。

- 专用功能：指复用到其它模块使用，比如 UART、I2C 等。

12.6.3.1. 通用输入去抖功能

通用功能的输入，支持两级去抖，去抖后作用于寄存器和产生中断，如下所示：

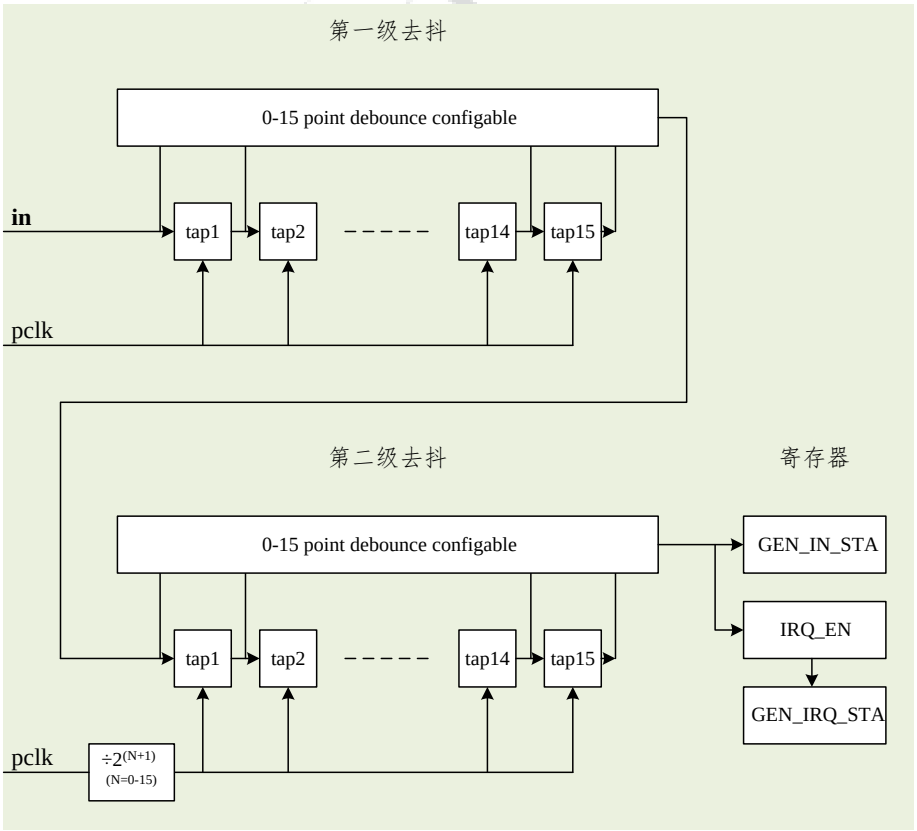


图 12–32 通用输入去抖功能示意图

- 第一级去抖用于滤除高频脉冲，去抖可配置使用 1–15 个参考点（配置为 0 表示不做去抖）。典型应用示例如下：

表 12–14 第一级去抖典型应用示例

参数	对应数值示例
APB0 时钟频率	100 MHz
采样周期	10 ns

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

参数	对应数值示例
过滤脉冲	最小为 10 ns，最大为 150 ns

- 第二级去抖用于滤除宽脉冲，如机械按键抖动等，去抖可配置使用 1–15 个参考点（配置为 0 表示不做去抖）。典型应用示例如下：

表 12–15 第二级去抖典型示例

参数	对应数值示例
APB0 时钟频率	100 MHz
采样周期	最大采样周期为0.65 ms 配置方式为 $2^{(N+1)}$ (N = 0 – 15)
过滤脉冲	最大过滤脉冲为9.83 ms

12.6.4. 寄存器列表

表 12–16 寄存器列表

偏移地址	缩写	寄存器描述	章节
0x00+G*0x100	GEN_IN_STA	GPIO 通用输入状态寄存器	0x00+G*0x100 GEN_IN_STA
0x04+G*0x100	GEN_OUT_CFG	GPIO 通用输出控制寄存器	0x04+G*0x100 GEN_OUT_CFG
0x08+G*0x100	GEN_IRQ_EN	GPIO 通用中断使能寄存器	0x08+G*0x100 GEN_IRQ_EN
0x0C+G*0x100	GEN_IRQ_STA	GPIO 通用中断状态寄存器	0x0C+G*0x100 GEN_IRQ_STA
0x10+G*0x100	GEN_OUT_CLR	GPIO 通用输出清除寄存器	0x10+G*0x100 GEN_OUT_CLR
0x14+G*0x100	GEN_OUT_SET	GPIO 通用输出置位寄存器	0x14+G*0x100 GEN_OUT_SET
0x18+G*0x100	GEN_OUT_TOG	GPIO 反转寄存器	0x18+G*0x100 GEN_OUT_TOG
0x80+P*0x4+G*0x100	PIN_CFG	GPIO 引脚功能配置寄存器	0x80+P*0x4+G*0x100 PIN_CFG
0xFFC	VERSION	GPIO 版本寄存器	0xFFC VERSION

其中，P 和 G 取值范围如下：

表 12–17 P 和 G 取值范围

分组	偏移地址	G 取值	P 取值
PA	0x000	0	0~5 8~11

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 12–17 P 和 G 取值范围 (续)

分组	偏移地址	G 取值	P 取值
PB	0x100	1	0~17
PC	0x200	2	0~7
PD	0x300	3	0~27
PE	0x400	4	12~13

12.6.5. 寄存器描述

12.6.5.1. 0x00+G*0x100 GEN_IN_STA

注：

该寄存器在满足以下任一情况下才有效：

- PIN_FUN=1 且 GEN_IN_EN=1 时有效。
- PIN_FUN = 2~15 且 SPE_IE_FORCE=1 时有效。

默认值：0x00000000			GPIO 通用输入状态寄存器
位域	**类型**	**默认值**	**描述**
31:0	RO	0x0	IN_STA 引脚输入状态 • 0: 表示输入为低电平 • 1: 表示输入为高电平 每位对应一个引脚。

12.6.5.2. 0x04+G*0x100 GEN_OUT_CFG

注：

该寄存器仅在 PIN_FUN=1 且 GEN_OUT_EN=1 时有效。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			GPIO 通用输出控制寄存器
位域	类型	默认值	描述
31:0	R/W	0x0	OUT_CFG 引脚输出配置 • 0: 输出设置为低电平 • 1: 输出设置为高电平 每位对应一个引脚。

12.6.5.3. 0x08+G*0x100 GEN_IRQ_EN

注：

该寄存器仅在 PIN_FUN=1 且 GEN_IN_EN=1 时有效。

默认值：0x00000000			GPIO 通用中断使能寄存器
位域	类型	默认值	描述
31:0	R/W	0x0	IRQ_EN 引脚中断使能 • 0: 关闭 • 1: 使能 每位对应一个引脚。

12.6.5.4. 0x0C+G*0x100 GEN_IRQ_STA

注：

该寄存器仅在同时满足以下条件时才有效：

- PIN_FUN=1
- GEN_IN_EN=1
- IRQ_EN=1

默认值：0x00000000			GPIO 通用中断状态寄存器
位域	类型	默认值	描述
31:0	R/W1C	0x0	IRQ_STA 引脚中断状态，该寄存器写 1 清零，即对对应位写 1 为清除对应的寄存器

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			GPIO 通用中断状态寄存器
位域	类型	默认值	描述
			• 0: 无中断 • 1: 有中断 每位对应一个引脚。

12.6.5.5. 0x10+G*0x100 GEN_OUT_CLR

默认值：0x00000000			GPIO 通用输出清除寄存器
位域	类型	默认值	描述
31:0	WO	0x0	OUT_CFG_CLR 输出控制寄存器清除 • 0: 对应引脚电平保持不变 • 1: 对应引脚设置为低电平 每位对应一个引脚。

12.6.5.6. 0x14+G*0x100 GEN_OUT_SET

默认值：0x00000000			GPIO 通用输出置位寄存器
位域	类型	默认值	描述
31:0	WO	0x0	OUT_CFG_SET 输出控制寄存器置位 • 0: 对应引脚电平保持不变 • 1: 对应引脚设置为高电平 每位对应一个引脚。

12.6.5.7. 0x18+G*0x100 GEN_OUT_TOG

默认值：0x00000000			GPIO 反转寄存器
位域	类型	默认值	描述
31:0	WO	0x0	OUT_CFG_TOG 输出控制寄存器置位

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			GPIO 反转寄存器
位域	类型	默认值	描述
			• 0: 对应引脚电平不变 • 1: 对应引脚电平反转 每位对应一个引脚。

12.6.5.8. 0x80+P*0x4+G*0x100 PIN_CFG

默认值：PA10~11: 0x00000338 默认值：PB12~17: 0x00000338 默认值：OTHERS: 0x00000000			GPIO 引脚功能配置寄存器
位域	类型	默认值	描述
31:28	R/W	0x0	GEN_IN_DB1_POINT 通用输入第二级去抖参考点数，仅在 PIN_FUN=1 且 GEN_IE=1 才有效。 该配置值为去抖采样点数，配置为 0 则关闭本级去抖。
27:24	R/W	0x0	GEN_IN_DB1_SAMP 通用输入第二级去抖采样除频，仅在 PIN_FUN=1 且 GEN_IE=1 才有效。 第二级去抖采样周期为 $2^{(SAMP+1)}$ 个 APB0 时钟 (100 MHz) 周期。
23:20	R/W	0x0	GEN_IN_DB0_POINT 通用输入第一级去抖采样点数，仅在 PIN_FUN=1 且 GEN_IE=1 才有效。 第一级去抖采样周期为一个 APB0 时钟 (100 MHz) 周期。 该配置值为去抖采样点数，配置为 0 则关闭本级去抖。
19	–	–	–
18	R/W	0x0	SPE_IE_FORCE 专用功能输入强制使能，仅在 PIN_FUN=2~6 才有效 • 0: 关闭 • 1: 打开 此字段用于调试使用，强制使能输入，并将输入状态存到通用输入状态寄存器。
17	R/W	0x0	GEN_OE 通用功能输出使能，仅在 PIN_FUN=1 才有效。 • 0: 关闭 • 1: 打开

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：PA10~11: 0x00000338 默认值：PB12~17: 0x00000338 默认值：OTHERS: 0x00000000			GPIO 引脚功能配置寄存器
位域	**类型**	**默认值**	**描述**
16	R/W	0x0	GEN_IE 通用功能输入使能，仅在 PIN_FUN=1 才有效。 • 0: 关闭 • 1: 打开
15	–	–	–
14:12	R/W	0x0	GEN_IRQ_MODE 通用功能中断检测模式，仅在 PIN_FUN=1 且 GEN_IE=1 才有效： • 0: 下降沿 • 1: 上升沿 • 2: 低电平 • 3: 高电平 • 4: 双沿（上升沿或下降沿） • 5~7: 预留
11:10	–	–	–
9:8	R/W	PA10~11: 3 PB12~17: 3 OTHERS: 0	PIN_PULL 引脚上下拉设置 • 0: 关闭，引脚处于高阻态 • 1: 预留 • 2: 下拉 33 KΩ 到 GND • 3: 上拉 33 KΩ 到 VCC-IO **注：** 引脚功能设置 PIN_FUN 配置为通用功能或专用功能，该寄存器都起作用。
7	–	–	–
6:4	R/W	PA10~11: 3 PB12~17: 3 OTHERS: 0	PIN_DRV 引脚输出驱动能力

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

<table>
<tr><td colspan="3">默认值：PA10~11: 0x00000338
默认值：PB12~17: 0x00000338
默认值：OTHERS: 0x00000000</td><td>GPIO 引脚功能配置寄存器</td></tr>
<tr><td>位域</td><td>类型</td><td>默认值</td><td>描述</td></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>–</td><td>3.3V</td><td>1.8V</td></tr>
<tr><td>0</td><td>180Ω</td><td>300Ω</td></tr>
<tr><td>1</td><td>90Ω</td><td>150Ω</td></tr>
<tr><td>2</td><td>60Ω</td><td>100Ω</td></tr>
<tr><td>3</td><td>45Ω</td><td>75Ω</td></tr>
<tr><td>4</td><td>36Ω</td><td>60Ω</td></tr>
<tr><td>5</td><td>30Ω</td><td>50Ω</td></tr>
<tr><td>6</td><td>26Ω</td><td>43Ω</td></tr>
<tr><td>7</td><td>23Ω</td><td>38Ω</td></tr>
</table>
注：
引脚功能设置 PIN_FUN 配置为通用功能或专用功能，该寄存器都起作用。
</td></tr>
<tr><td>3:0</td><td>R/W</td><td>PA10~PA11: 8
PB12~PB17: 8
OTHERS: 0</td><td>PIN_FUN
引脚功能设置
• 0: 关闭
• 1: 通用功能 (GEN)
• 2~15: 专用功能 (SPE)</td></tr>
</table>

12.6.5.9. 0xFFC VERSION

默认值：0x00000100			GPIO 版本寄存器
位域	类型	默认值	描述
31:0	RO	0x00000100	VERSION GPIO 版本，V1.0。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

13. 模拟

13.1. ADC Interface Module (ADCIM)

ADC 接口管理模块 (ADCIM) 接收来自不同功能应用模块的指令来控制 ADC 运行。通过控制 ADC 进行数据转换操作，将模拟信号转换为数字信号，并将转换后的数据返回给相应的功能应用模块，供其进一步处理和使用。

ADC 的功能应用模块包括 GPAI/ THS/ RTP 模块。在 ADC 功能应用模块启动之前，需要初始化 ADC 接口管理模块，否则应用模块无法访问到 ADC 接口。

13.1.1. 特性说明

- APB 总线访问寄存器
- 多通道可配，每个通道捕获时间 TACQ 可配置
- 支持两种采样优先级：低优先级（非实时采样），高优先级（实时采样）
- 支持深度为 16 的低优先级采样命令 FIFO
- 多通道间相互独立
- 不支持数据 FIFO

13.1.2. 原理框图

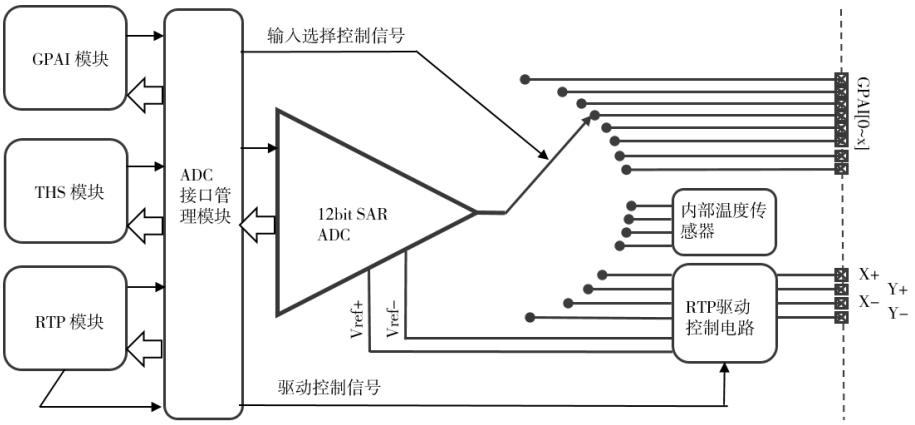


图 13–1 ADC 接口管理模块在系统中的应用

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

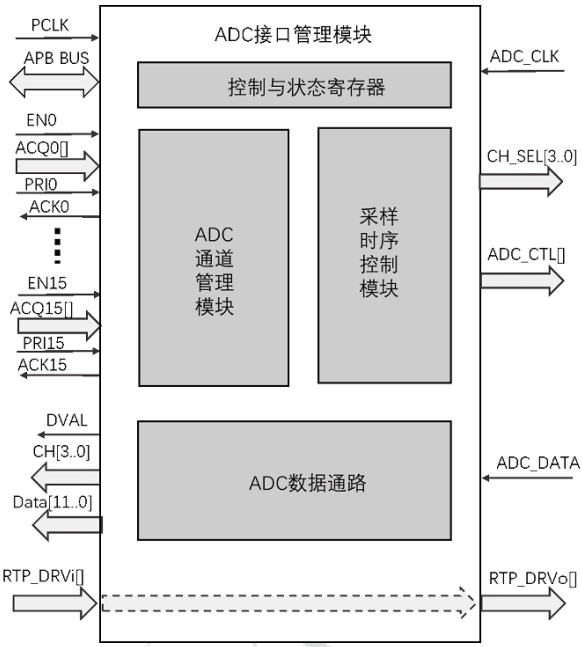


图 13–2 ADC 接口管理模块原理图框图

13.1.3. 功能描述

13.1.3.1. 非实时采样与实时采样

ADC 应用模块传递给 ADCIM 的采样命令分为以下两类，用来实现非实时采样和实时采样功能：

> **注：**
>
> 关于采样命令详细说明，可查看 采样命令接口说明。

- 低优先级命令：通常是面对非实时需求的场景。

  例如，在 GPAI 的多数应用场景，以及 THS 温度传感器采样的应用场景中，没有严格的完成时间要求（时序要求），只需要在大概的时间内对某个输入量进行采样量化并返回给 ADC 应用模块即可。

- 高优先级命令：通常是面对有实时性需求的场景。

  例如，RTP 模块的采样需要配合使用外部的驱动电路， 有较高的时序要求。RTP 模块需要明确知道采样的起始和结束时间，而且期望在较快的时间内完成采样，以获得更好的用户体验。

对于低优先级与高优先级命令，ADCIM 模块内部有不同的处理和响应机制：

- 高优先级命令 (PRI = 1) 的响应优先级高于低优先级命令 (PRI = 0)。
- 对于同时接收到的、优先级相同的命令，使用轮询的机制，从通道号小的先开始。
- 低优先级命令响应机制为先请求先排队先装入 FIFO 进行排队并等待响应。
  - 对于同时请求的命令，使用轮询的机制，从通道号小的先开始。
- 高优先级的实时命令不会经过 FIFO，直接参与仲裁响应。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

如图 13–3 : 低优先级与高优先级命令的响应机制举例所示，其中 PRI 表示优先级，CH 表示通道号，响应的优先顺序如下：

1. CH6 (PRI=1)、CH7 (PRI=1)、CH15 (PRI=0) 与 CH0 (PRI=0)，同时先发出采样请求。
2. CH1(PRI=0) 与 CH2 (PRI=0) 发出采样请求。
3. CH8 (PRI=1) 发出采样请求。

根据高低优先级的响应机制，最终采样执行的先后顺序为 **CH6（最先） > CH7 > CH0 > CH8 > CH15 > CH1 > CH2（最后）**。

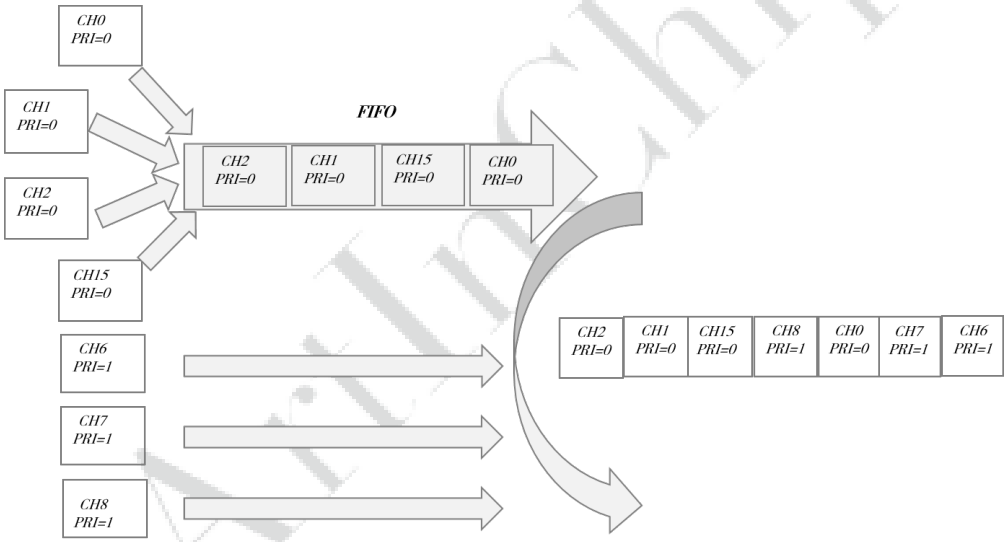


图 13–3 低优先级与高优先级命令的响应机制举例

13.1.3.1.1. 采样命令接口说明

采样命令由三部分构成：

- 优先级：分为高优先级和低优先级， 由对应通道的 PRI 信号控制，可实现实时采样和非实时采样功能。
- 通道号：在模块内部产生，直接对应不同通道的端口。
- 捕获时长：以 ADC_CLK 的周期为单位的计数值，来自对应通道的 ACQx 信号：

$$T_{ACQx} = (ACQx + 1) \times T_{ADC_CLK}$$

注：

每个通道的捕获时长控制信号 ACQx 直接反映 ADC 应用模块对每个通道的配置值。在已经发出采样命令但未收到返回的采样数据值之前，不能改变此配置值。

以下分别展示了低优先级和高优先级命令的时序图：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

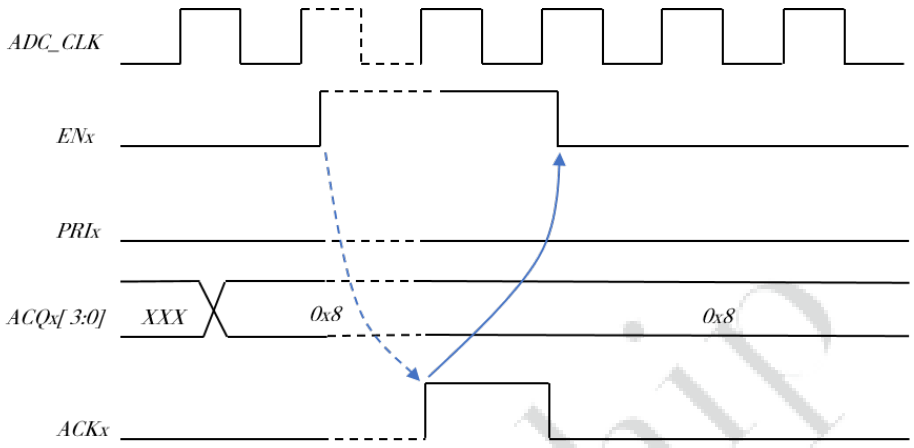


图 13–4 低优先级命令时序

低优先级采样命令时序中：

- 从 ENx 有效，到 ACKx 有效，中间会经过不确定的时钟周期数的时间，原因为：
  1. 多个通道同时发送采样命令，ADCIM 不能及时响应。
  2. 如果内部 FIFO 已经满，则会通过 ACKx 信号延迟命令接收。
- 从 ACKx 有效到 ENx 无效只需要经历一个周期。ACKx 有效意味着本采样命令已经进入命令 FIFO。

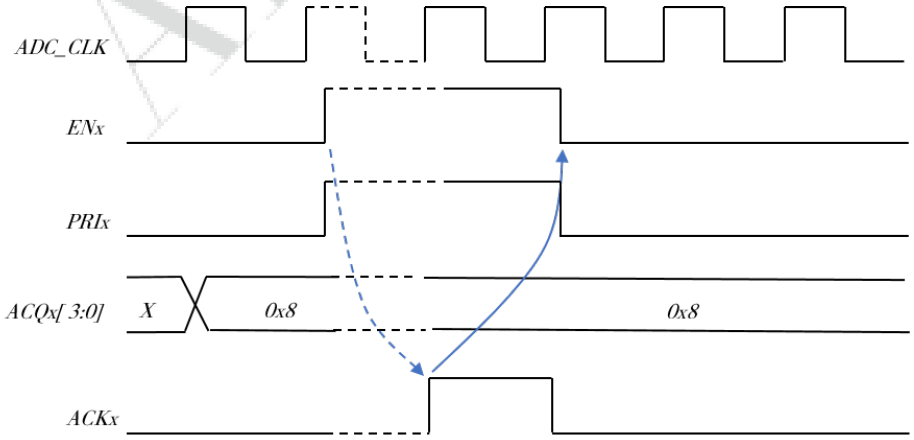


图 13–5 高优先级命令时序

实时采样命令时序中：

- 从 ENx 有效，到 ACKx 有效，与图 13–4 : 低优先级命令时序一样，中间会经过不确定的时钟周期数， 原因是：
  1. 正在处理的低优先级非实时命令，需要先处理完这一个命令。
  2. 有其它的同样是高优先级命令正在处理。
- 从 ACKx 有效到 ENx 无效只需要一个时钟周期。ACKx 有效意味着本采样命令已经处理完成。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

13.1.3.1.2. 采样数据返回接口说明

采样数据返回通道包括 {DVAL, CH，DATA}，多个通道的数据通过共同的 DATA 返回通道传递给不同的 ADC 应用模块，例如 RTP/ GPAI/ THS。ADC 应用模块在 DVAL 有效时接收根据 DVAL 与 CH[3:0] 值锁存与自身通道号匹配的采样值 (DATA)。

由于 ADC 采样数据是从 ADC 模拟电路串行输入，并转化为并行数据输出，所以 DVAL 信号不存在连续多个时钟周期有效的情况， 都是一个个单周期的有效信号。图 13–6 : 数据返回通道时序如下所示。

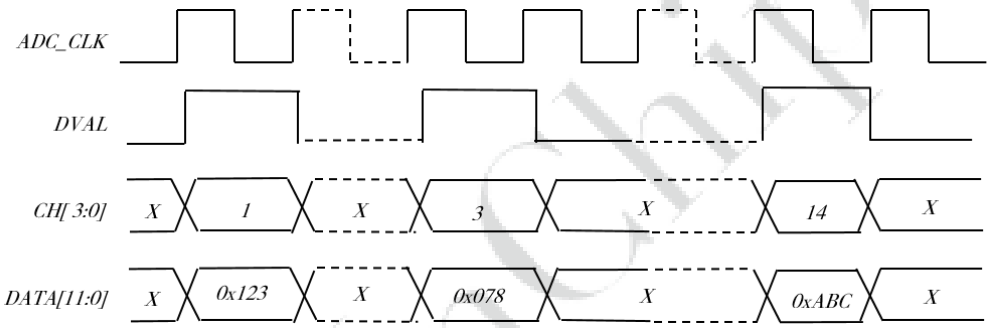


图 13–6 数据返回通道时序

如图 13–6 : 数据返回通道时序所示：

- 当 DVAL=1，CH=1，DATA=0x123，表示 0x123 为通道 1 的有效数据。
- 当 DVAL=1，CH=3，DATA=0x078，表示 0x078 为通道 3 的有效数据。
- 当 DVAL=1，CH=14，DATA=0xABC ，表示 0xABC 为通道 14 的有效数据。

13.1.3.2. ADC 校准功能说明

ADC 接口管理模块 （ ADCIM ）在实际应用中可以提供 ADC 硬件自动校准功能，以确保采样数据的准确性，提高数据转换的精度。

ADC 硬件自动校准是一种特殊的采样功能，通过将一个参考电平输入到 ADC 进行采样，并计算出偏差值来校准后续的采样数据。理论上，采样返回的结果应该是 （ 2^{12})/2=2048，但实际上总会存在偏差，此时硬件会计算出此偏差，并把此偏差应用于对后续采样数据的校准。

ADCIM 模块默认情况下会对采样数据进行校准，并将校准后的数据输出到 ADC 应用模块。然而，软件也可以通过寄存器修改校准值或禁用校准功能：

- 硬件自动校准：把 **CALVAL** 回写到 **CALCSR.CALVAL**，并且开启校准功能。

  硬件会自动进行校准过程，将计算出的偏差值应用到后续的采样数据上。

  硬件自动校准可以提供更精确的结果，但需要额外的硬件支持。

- 软件校准：根据 `DATA （计算之后结果） = ADC_DATA - (CALVAL - 2048)` 公式进行校准，并且屏蔽校准功能。

  软件校准则更加灵活，可以在不需要修改硬件的情况下进行校准。

ADC 校准的算法： `DATA = ADC_DATA - (CALVAL - 2048)`。其中

- **DATA** ： ADCIM 模块经过校准后输出到 ADC 应用模块的数据。
- **ADC_DATA**：ADC 数据串并转换后未经过校准的数据。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- **CALVAL：CALCSR.CALVAL** 寄存器值
- **CALVAL：CALCSR.CALVAL** – 8 。

13.1.4. 编程指南

13.1.4.1. 初始化流程

在 RTP/ GPAI/ THS 工作之前，需要对 ADCIM 进行时钟使能，即 CMU 开启 ADCIM 时钟。

13.1.4.2. 自动校准

ADCIM 使能 ADC 对参考电平的中间值进行采样 (CALEN=1)。理论上返回结果应该是 (2^{12}) / 2 = 2048，但实际上总会存在偏差， 硬件会计算出此偏差，并把此偏差应用于对后续采样数据的校准。

自动校准流程：

1. CMU 使能 ADCIM 模块。
2. 使能自动校准 （ CALEN = 1 ）。
3. 轮询自动校准使能 (bit) 是否为 0 (CALEN = 0)。
4. 自动校准使能 (bit) 为 0，自动校准完成。
5. 获取自动校准完成之后的 ADCIM_CALCSR.CALVAL。

13.1.5. 寄存器列表

表 13–1 ADCIM 寄存器列表

偏移	缩写	寄存器名称	章节
0x000	ADCIM_MCSR	ADCIM 模块控制状态	0x000 ADCIM_MCSR
0x004	ADCIM_CALCSR	ADCIM 校准控制	0x004 ADCIM_CALCSR
0x008	ADCIM_FIFOSTS	ADCIM FIFO 状态	0x008 ADCIM_FIFOSTS
0xFFC	ADCIM_VERSION	ADCIM 模块版本	0xFFC ADCIM_VERSION

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

13.1.6. 寄存器描述

13.1.6.1. 0x000 ADCIM_MCSR

默认值：0x00000000			ADCIM 模块控制状态 (ADCIM Module Control State Register)
位域	类型	默认值	描述
31:17	_	_	_
16	RO	0x0	SAMPLE STATE 采样状态 • 0: ADC 模块处于空闲状态，没有采样命令需要处理。 • 1: ADC 模块正处于 工作 状态，正在处理采样命令（包括捕获和转换）。
15:8	R/W	0x00	SEMFLAG 信号量标志 • 8-bit 寄存器，可以为软件任意读写，不会对硬件行为产生任何影响。 • 作为 ADCIM 被多个应用模块共享时的信号量标志。 • 每个寄存器位由软件自由定义。
7:4	_	_	_
3:0	RO	0x00	CHN 正在处理的通道号 (Number of Channel Processing) 作为 debug 使用，表示当前正在采样的通道号。

13.1.6.2. 0x004 ADCIM_CALCSR

默认值：0x08002F00			ADCIM 校准控制 (ADCIM Calibration Control Set Register)
位域	类型	默认值	描述
31	R/WAC	0x0	CALVAL_UPD ADC 校准值更新 (Calibration Value Update) 软件设置 CALVAL 的同时设置 CALVAL_UPD 为 1，表示软件需要更新 CALVAL 值，CALVAL_UPD 被置为 1 后会触发内部更新，硬件马上置 CALVAL _UPD 为 0。
30:28	_	_	_
27:16	R/W	0x800	CALVAL ADC 校准输出值 (Calibration Value) ADC 的校准过程如下：

Copyright © 2023-2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x08002F00			ADCIM 校准控制 (ADCIM Calibration Control Set Register)
位域	类型	默认值	描述
			1. ADC 采样一个参考电压，并返回一个 12 bits 的值。 2. 硬件保存并使用 12 bits 的值对所有的采样数据进行校准。 3. 如希望使用一个非硬件自动校准取得的值对 ADC 数据进行校准，可通过软件设置一个合适的值，则硬件会参考该值对 ADC 采样数据进行自动校准。
15:8	R/W	0x2F	ADC_ACQ ADC 校准捕获时间 (ADC Acquire) 校准时 ADC 通道的捕获时间长度。 计数值单位为 ADC_CLK 的周期数，$T_{ADC_ACQ} = (ADC_ACQ+1) \times T_{ADC_CLK}$。假设 ADC_CLK = 24 MHz，ADC_ACQ = 0x2F，则 $T_{ADC_ACQ} = (47 + 1) \times 41.67\ ns = 2\ us$。
7:2	–	–	–
1	R/W	0x0	DCALMASK 采样数据校准屏蔽 (Data Calibration Mask ） • 0：硬件默认对采样数据进行校准后送给 ADC 应用功能模块。 • 1：屏蔽采样数据校准功能，模块直接把来自 ADC 模拟电路的采样数据不经校准地送给 ADC 应用功能模块。
0	R/WAC	0x0	CALEN ADC 校准使能 (Calibration Enable) 使能信号由软件置位，使能 ADC 校准。 • ADC 校准完后，硬件会对 CALEN 自动清零。软件查询到 CALEN 被清零后，意味着校准已完成。 • 自动校准存在一个固定偏差 (OFFSET)，需要进行校准。 • OFFSET= 8 。

13.1.6.3. 0x008 ADCIM_FIFOSTS

默认值：0x00000040			ADCIM FIFO 状态 (ADCIM FIFO States)
位域	类型	默认值	描述
31:7	–	–	–
6	RO	0x1	ADC Arbiter Status ADC 仲裁状态： 仅 debug 使用

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000040			ADCIM FIFO 状态 (ADCIM FIFO States)
位域	类型	默认值	描述
			• 0：ADC 接口模块正在接收其它模块的命令。 • 1：ADC 接口模块正处于 Idle 状态。
5	RO	0x0	FIFO Error Status FIFO 错误状态： 仅 debug 使用 • 0：表示命令 FIFO 正常。 • 1：表示命令 FIFO Error (Overflow 或者 Underflow)。
4:0	RO	0x00	FIFOCTR FIFO 计数器 (FIFO Counter)，用于显示采样命令 FIFO 的占用情况。 仅 debug 使用。 FIFO 深度为 16，有效值为 0~16（0x00 ~ 0x10），表示 FIFO 内未处理的采样命令数。

13.1.6.4. 0xFFC ADCIM_VERSION

默认值：0x00000100			模块版本 (ADCIM Version)
位域	类型	默认值	描述
31:0	RO	0x00000100	Version 模块版本号 采用 BCD 码显示，V1.0。

13.2. General Purpose Analog Input (GPAI)

通用模拟输入接口 (GPAI) 模块，通过 ADC 接口管理模块 (ADCIM) 来获取模拟输入信号的输入电平，产生高低阈值报警。

ADCIM 模块是用于配置和控制模数转换器（ADC）的硬件模块，负责将模拟信号转换为数字信号，以便 GPAI 可以读取和处理。因此在启动 GPAI 之前，需要先启动 ADC Interface Module (ADCIM) 模块。

13.2.1. 特性说明

- 支持 APB 总线访问寄存器
- 最多支持六个模拟信号输入
- 支持单点采样与周期采样两种采样模式
- 每次采样突发采样数量可配 2^0 / 2^1/ 2^2/ 2^3

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 支持高电平报警阈值与低电平报警阈值独立可配
- 每个通道 FIFO 为 12*8， 单位 bits
- 基准电压：2.5 V

13.2.2. 原理框图

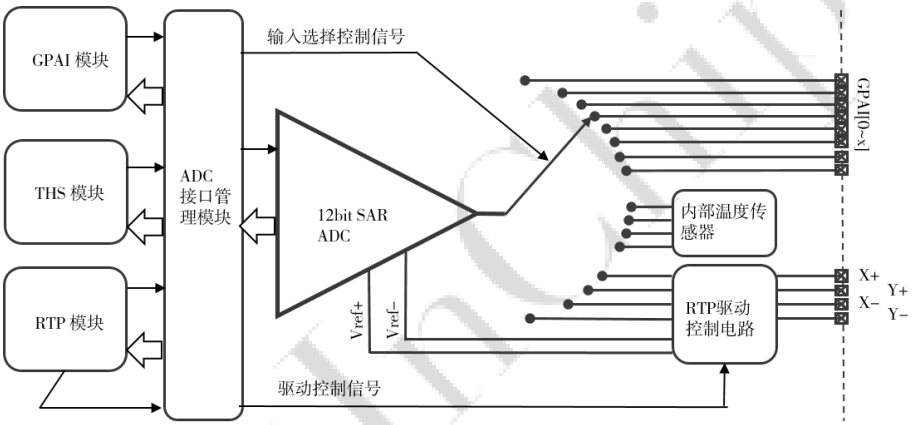


图 13–7 GPAI 接口模块与 ADC 连接关系

13.2.3. 功能描述

13.2.3.1. 关于采样的定义

在 GPAI 模块中，采样被细分为以下三种类型，以便更好地理解和应用不同的采样类型：

- 单次采样：是指对任意输入信号进行一次 ADC 采样的过程 ，一次采样对应一条 ADC 采样命令。

  单次采样简单直接，适用于需要快速获取单个数据点的场景。

- 单点采样：是指 ADC 在任意一个时间点开始的连续采样过程。单点采样过程可能是单次采样，也可能包含多次采样，具体由 SBC 配置决定。

  例如 ADC 在一个时间点开始对某个特定信号进行多次采样，即执行多条 ADC 采样命令，以获得多个采样值。然后通过对采样值的适当的滤波，如算数平均，以获得一个更加准确和稳定的平均结果。

  单点采样适用于需要在嘈杂环境中准确测量模拟信号的应用。通过连续采样并应用算术平均滤波，可以有效地抑制随机噪声，从而得到更平滑的测量结果。

- 周期采样：是指 ADC 在对特定信号进行固定间隔的周期性单点采样，每个周期就进行一次单点采样。

  周期采样模式允许用户以一定的时间间隔重复采样，从而捕捉到信号的周期性变化。通过调整采样间隔，可以根据具体应用的需求来优化数据的采集频率和精度。

  周期采样适用于需要连续监测信号变化的场景，通过周期性地获取数据点，可以有效地跟踪和分析信号的变化趋势。

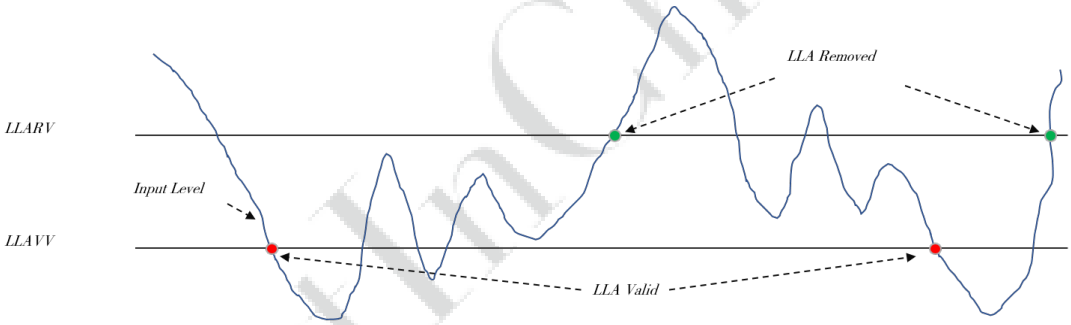
Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

13.2.3.2. 高/ 低电平报警阈值

GPAI 模块能够根据设定的高低阈值来检测信号的变化，并在信号超过或低于设定的阈值时触发相应的报警。用户可以设定一个高阈值和一个低阈值，用于确定何时触发报警：

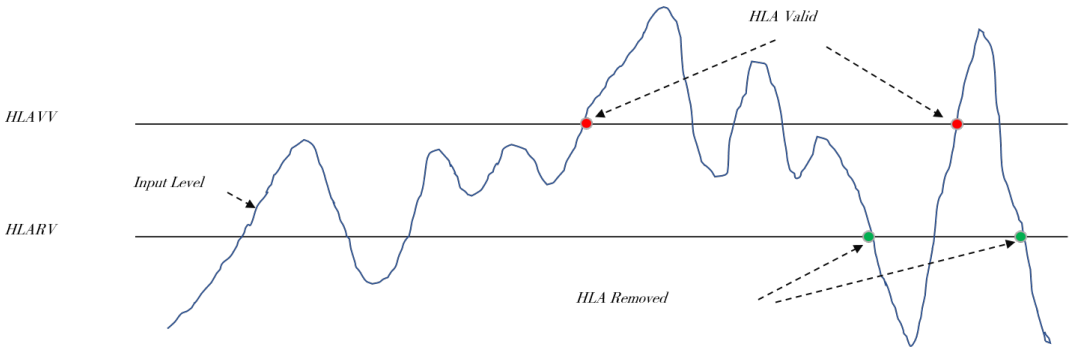
- 低电平报警阈值：
  - 低电平报警有效阈值 (LLAVV)
  - 低电平报警解除阈值 (LLARV)

  LLAVV ≤ LLARV，并且 LLARV – LLAVV 为低电平报警的迟滞值，必须覆盖信号的正常抖动范围，才能保证不会反复出现虚警。



- 高电平报警阈值：
  - 高电平报警有效阈值 (HLAVV)
  - 高电平报警解除阈值 (HLARV)

  HLAVV ≥ HLARV，并且 HLAVV – HLARV 为高电平报警的迟滞值，必须覆盖信号的正常抖动范围，才能保证不会反复出现虚警。



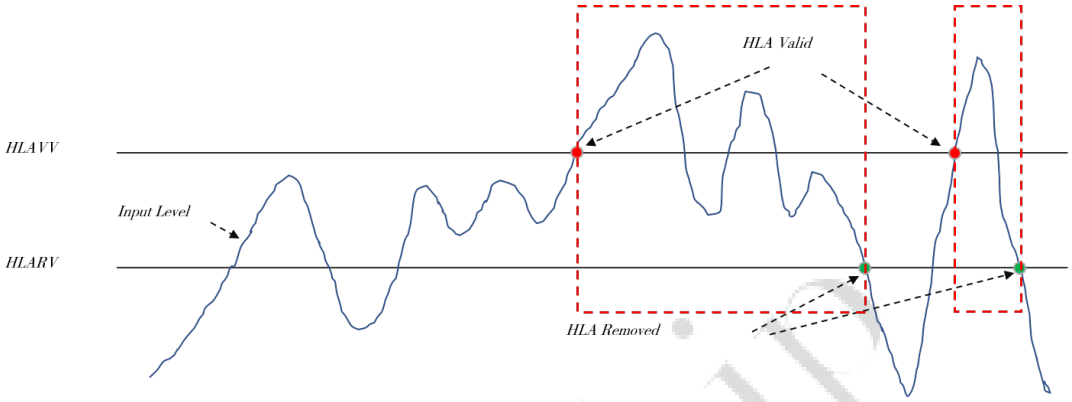
13.2.3.3. 高/ 低/ 正常电平数据界定

为了区分不同状态下的信号数据，寄存器 GPAI_CHnACR 中定义了电平数据的相关概念，确保在分析信号时能够准确识别出异常情况：

- 高电平数据：是指高电平报警使能之后，在高电平报警有效 (HLAV) 和高电平报警解除 (HLAR) 期间所取得的采样数据。高电平采样数据的数值一定不小于高电平报警解除阈值 (HLARV)。

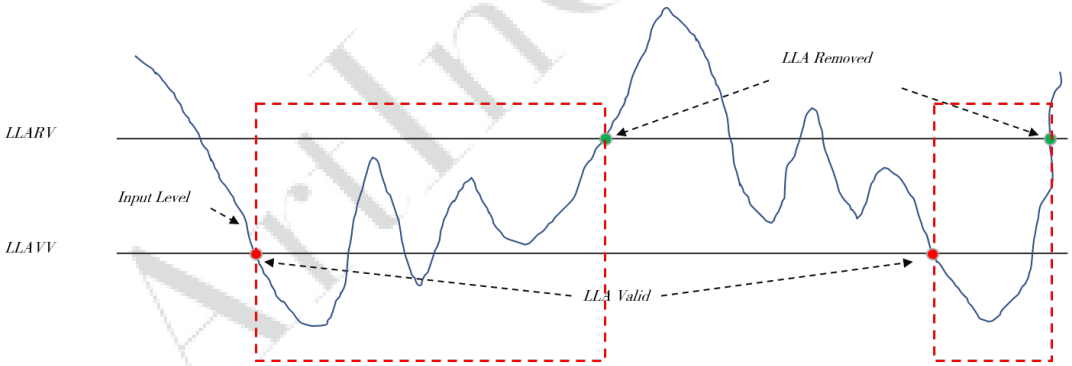
  如下图所示，虚框范围内的信号就是高电平采样数据对应的信号：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



- 低电平数据：是指低电平报警使能之后，在低电平报警有效 (LLAV) 和 低电平报警解除 （ LLAR ）期间所取得的采样数据。低电平采样数据的数值一定不大于低电平报警解除阈值 (LLARV)。

  如下图所示，虚框范围内的信号就是低电平采样数据对应的信号：



- 正常电平数据：是指排除了高电平数据与低电平数据之后，剩余的采样值。

  正常电平数据有助于了解系统在正常运行状态下的表现。

注：

上述数据都是算数平均之后的值。

通过设定高低阈值并监测信号的变化，可以有效地捕捉到高电平和低电平事件，并在必要时触发相应的报警或处理措施。同时，正常电平数据的获取则有助于了解系统在正常运行状态下的表现，为后续的数据分析和决策提供参考。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

13.2.4. 编程指南

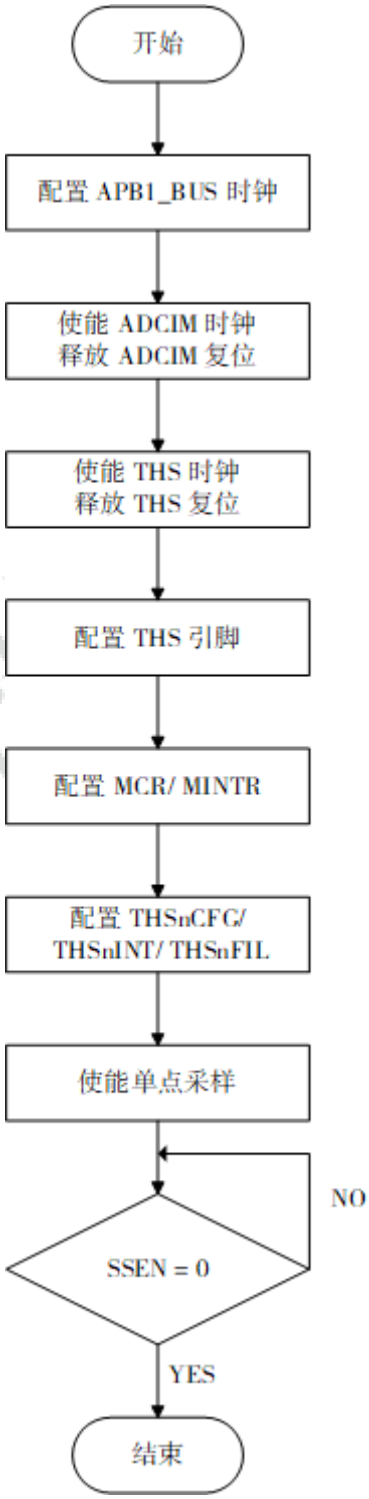


图 13–8 单点采样

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

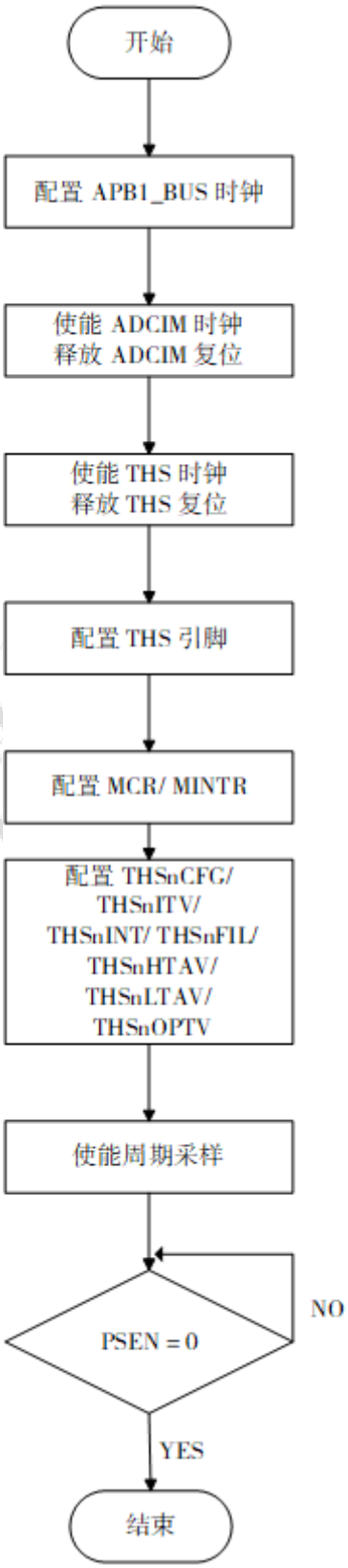


图 13–9 周期采样

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

13.2.5. 寄存器列表

表 13–2 GPAI 寄存器列表

偏移	缩写	寄存器名称	章节
0x000	GPAI_MCR	GPAI 模块控制	0x000 GPAI_MCR
0x004	GPAI_INTR	GPAI 中断控制与状态	0x004 GPAI_INTR
0x100	GPAI_CHnCR	GPAI 通道控制	0x100+0x40*n GPAI_CHnCR
0x104	GPAI_CHnINT	GPAI 通道中断控制与状态	0x104+0x40*n GPAI_CHnINT
0x108	GPAI_CHnPSI	GPAI 通道采样间隔配置	0x108+0x40*n GPAI_CHnPSI
0x110	GPAI_CHnHLAT	GPAI 高电平报警阈值	0x110+0x40*n GPAI_CHnHLAT
0x114	GPAI_CHnLLAT	GPAI 低电平报警阈值	0x114+0x40*n GPAI_CHnLLAT
0x118	GPAI_CHnACR	GPAI 报警控制	0x118+0x40*n GPAI_CHnACR
0x120	GPAI_CHnFCR	GPAI FIFO 控制	0x120+0x40*n GPAI_CHnFCR
0x124	GPAI_CHnDATA	GPAI 通道数据	0x124+0x40*n GPAI_CHnDATA
0xFFC	VERSION	GPAI 版本寄存器	0xFFC VERSION

13.2.6. 寄存器描述

13.2.6.1. 0x000 GPAI_MCR

默认值：0x00000000			GPAI 模块控制 (GPAI Module Control Register)
位域	类型	默认值	描述
31:16	–	–	–
13:8	R/W	0x00	CHEN 通道使能控制 (Channel Enable) 包含 8 bits，分别对应八个通道的独立使能控制。每个 bit 分别对应一个通道： • Bit 8：通道 0 • Bit 9：通道 1 • Bit 10：通道 2

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			GPAI 模块控制 (GPAI Module Control Register)
位域	类型	默认值	描述
			• Bit 11：通道 3 • Bit 12：通道 4 • Bit 13：通道 5 每个 bit 位都可通过设置以下值控制对应的通道使能： • 0：禁止通道 • 1：使能通道
7:1	–	–	–
0	R/W	0	EN 模块使能 (Module Enable) • 0：本模块被禁止，本模块停止工作。 • 1：本模块被使能。

13.2.6.2. 0x004 GPAI_INTR

默认值：0x00000000			GPAI 中断控制与状态 (GPAI Interrupt Register)
位域	类型	默认值	描述
31:24	–	–	–
23	RO	0	CH7_FLG 通道 7 事件更新标志 (Channel 7 Flag) 通道 7 是否有新的事件发生，例如高/低电平报警事件，数据有效事件，FIFO 错误事件。 • 0：无新的待处理事件。 • 1：有新的待处理事件。
22	RO	0	CH6_FLG 通道 6 事件更新标志 (Channel 6 Flag) 通道 6 是否有新的事件发生，例如高/低电平报警事件，数据有效事件，FIFO 错误事件。 • 0：无新的待处理事件。 • 1：有新的待处理事件。
21	RO	0	CH5_FLG 通道 5 事件更新标志 (Channel 5 Flag)

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			GPAI 中断控制与状态 (GPAI Interrupt Register)
位域	**类型**	**默认值**	**描述**
			通道 5 是否有新的事件发生，例如高/低电平报警事件，数据有效事件，FIFO 错误事件。 • 0：无新的待处理事件。 • 1：有新的待处理事件。
20	RO	0	CH4_FLG 通道 4 事件更新标志 (Channel 4 Flag) 通道 4 是否有新的事件发生，例如高/低电平报警事件，数据有效事件，FIFO 错误事件。 • 0：无新的待处理事件。 • 1：有新的待处理事件。
19	RO	0	CH3_FLG 通道 3 事件更新标志 (Channel 3 Flag) 通道 3 是否有新的事件发生，例如高/低电平报警事件，数据有效事件，FIFO 错误事件。 • 0：无新的待处理事件。 • 1：有新的待处理事件。
18	RO	0	CH2_FLG 通道 2 事件更新标志 (Channel 2 Flag) 通道 2 是否有新的事件发生，例如高/低电平报警事件，数据有效事件，FIFO 错误事件。 • 0：无新的待处理事件。 • 1：有新的待处理事件。
17	RO	0	CH1_FLG 通道 1 事件更新标志 (Channel 1 Flag) 通道 1 是否有新的事件发生，例如高/低电平报警事件，数据有效事件，FIFO 错误事件。 • 0：无新的待处理事件。 • 1：有新的待处理事件。
16	RO	0	CH0_FLG 通道 0 事件更新标志 (Channel 0 Flag) 通道 0 是否有新的事件发生，例如高/低电平报警事件，数据有效事件，FIFO 错误事件。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			GPAI 中断控制与状态 (GPAI Interrupt Register)
位域	类型	默认值	描述
			• 0：无新的待处理事件。 • 1：有新的待处理事件。
15:8	–	–	–
7	R/W	0	CH7_INTEN 通道 7 中断使能 (Channel 7 Interrupt Enable) • 0：禁止中断 • 1：使能中断
6	R/W	0	CH6_INTEN 通道 6 中断使能 (Channel 6 Interrupt Enable) • 0：禁止中断 • 1：使能中断
5	R/W	0	CH5_INTEN 通道 5 中断使能 (Channel 5 Interrupt Enable) • 0：禁止中断 • 1：使能中断
4	R/W	0	CH4_INTEN 通道 4 中断使能 (Channel 4 Interrupt Enable) • 0：禁止中断 • 1：使能中断
3	R/W	0	CH3_INTEN 通道 3 中断使能 (Channel 3 Interrupt Enable) • 0：禁止中断 • 1：使能中断
2	R/W	0	CH2_INTEN 通道 2 中断使能 (Channel 2 Interrupt Enable) • 0：禁止中断 • 1：使能中断
1	R/W	0	CH1_INTEN 通道 1 中断使能 (Channel 1 Interrupt Enable)

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			GPAI 中断控制与状态 (GPAI Interrupt Register)
位域	类型	默认值	描述
			• 0：禁止中断 • 1：使能中断
0	R/W	0	CH0_INTEN 通道 0 中断使能 (Channel 0 Interrupt Enable) • 0：禁止中断 • 1：使能中断

13.2.6.3. 0x100+0x40*n GPAI_CHnCR

(1)

默认值：0x000n2F00			GPAI 通道控制 (GPAI Channel N Control Register)
位域	类型	默认值	描述
31:26	–	–	–
25:24	R/W	0x0	SBC 采样突发次数 (Sample Burst Count) 单点采样（或者周期采样模式中的每点采样）中包含的采样次数。 • 每点采样的突发次数 = 2^{SBC}。 • SBC = 0~3 分别对应突发次数 1/ 2/ 4/ 8。 每次单点采样使能 (SSEN=1) 或者周期采样使能 (PSEN=1)，GPAI 模块就会连续向 ADC 模块发出 2^{SBC} 个采样命令，对应地 ADC 就会返回 2^{SBC} 个采样数据。2^{SBC} 个采样数据经过简单的算数平均计算（保留整数部分）之后，合并为一个采样数据后，存入 FIFO，并等待软件进行后续处理。
23:20	–	–	–
19:16	RO	0xn	ADC_CHN ADC 通道号 (ADC Channel Number) 显示系统分配的 ADC 通道号，硬件会根据此通道号匹配 ADC 并获取返回的数据，软件驱动不需要使用，仅 debug 使用。n=0~7。
15:8	R/W	0x2F	ADC_ACQ ADC 捕获时间 (ADC Acquisition Time)

(1) n=0~7 为 GPAI 通道号。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x000n2F00			GPAI 通道控制 (GPAI Channel N Control Register)
位域	**类型**	**默认值**	**描述**
			对应 ADC 通道的捕获时间长度。计数值单位为 ADC_CLK 的周期数，T_{ADC_ACQ} = (ADC_ACQ+1) *T_{ADC_CLK}。假设 ADC_CLK=24MHz， ADC_ACQ=0x2F，T_{ADC_ACQ} = (47+1)*41.67ns = 2us。
7:5	–	–	–
4	R/W	0x0	ADC_PRI ADC 采样优先级 (ADC Priority) 本次 ADC 采样的优先级控制。 • 0： 低优先级，采样命令会与其它低优先级命令进入采样队列。 • 1：高优先级，采样命令会被直接处理，不进入采样队列。
3	–	–	–
2	R	0x0	CHn_State 通道工作状态 (Channel N State) • 0： 空闲。 • 1：忙碌。
1	R/W	0	PSEN 周期采样使能 (Periodic Sample Enable) 使能 GPAI 通道周期采样。由软件置位，表示启动相应 GPAI 通道的周期性采样： • 采样间隔由 GPAI_CHnPSI 寄存器设定。 • 软件清零表示停止周期性采样。 • 如果在软件清零之前硬件已经启动了一次新的采样，则继续完成新采样命令，并最终返回数据。 **注：** 不允许同时使能周期采样与单次采样，即 SSEN 与 PSEN 不可以同时有效。
0	R/WAC	0	SSEN 单点采样使能 (Single Sample Enable) 使能 GPAI 通道单点采样。由软件置位，表示启动相应通道信号的采样： • 单点采样完成之后，本字段就会由硬件自动清零。 **注：** 尽管本字段描述为单点采样，但此处单点采样实际上会包含多次采样，避免单次采样产生误差。通过多次采样后进行算术平均（保留整数部分），得到一个相对稳定的信号电平值。由硬件向 ADC 接口控制器连续发出多个采样命

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x000n2F00			GPAI 通道控制 (GPAI Channel N Control Register)
位域	类型	默认值	描述
			令，连续发出采样命令的个数由 SBC 设定。例如 SBC 配置为 0x2，则每次启动 SSEN，硬件会连续发出 $2^2 = 4$ 个采样命令，四个采样命令发出后，硬件自动清零 SSEN。

13.2.6.4. 0x104+0x40*n GPAI_CHnINT

默认值：0x00000000			GPAI 通道中断控制与状态 (DMA Channel N Interrupt)
位域	类型	默认值	描述
31:24	–	–	–
23	R/W1C	0	LLAR_FLG 低电平警报解除标志 (Low-Level Alarm Removed Flag) 本字段高电平有效，表示低电平警报解除，即采样后的数据从低于 LLAVV（低电平警报有效阈值）到高于 LLARV（低电平警报解除阈值）。 • 由软件写 1 清零。 • 仅周期采样模式 (PSEN=1) 下有效。
22	R/W1C	0	LLAV_FLG 低电平警报有效标志 (Low-Level Alarm Valid Flag) 本字段高电平有效，表示低电平警报有效，即采样后的数据从高于 LLAVR（低电平警报解除阈值）到低于 LLAVV（低电平警报有效阈值）。 • 由软件写 1 清零。 • 仅周期采样模式 (PSEN=1) 下有效。
21	R/W1C	0	HLAR_FLG 高电平警报解除标志 (High-Level Alarm Removed Flag) 本字段高电平有效，表示高电平警报解除，即采样后的数据从高于 HLAVV（高电平警报有效阈值）到低于 HLARV（高电平警报解除阈值）。 • 由软件写 1 清零。 • 仅周期采样模式 (PSEN=1) 下有效。
20	R/W1C	0	HLAV_FLG 高电平警报有效标志 (High-Level Alarm Valid Flag) 本字段高电平有效，表示高电平警报有效，即采样后的数据从低于 HLARV（高电平警报解除阈值）到高于 HLAVV（高电平警报有效阈值）。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			GPAI 通道中断控制与状态 (DMA Channel N Interrupt)
位域	类型	默认值	描述
			• 由软件写 1 清零。 • 仅周期采样模式（ PSEN=1) 下有效。
19:18	–	–	–
17	RO	0	FIFO_FLG FIFO 异常标志 (FIFO Error Flag) 本字段高电平有效，表示 FIFO 异常，异常包括上溢出 (Overflow) 与下溢出 (Underflow)。 本字段只读，可以通过清除 FIFO 控制寄存器来清除本标志位。
16	R/W1C	0	DRDY_FLG 数据有效标志 (Data Ready Flag) 本字段高电平有效，表示 FIFO 数据有效，即 FIFO 内数据量大于等于 FIFO 数据有效阈值。
15:8	–	–	–
7	R/W	0	LLAR_INTEN 低电平警报解除中断使能 (Low-Level Alarm Removed Interrupt Enable) • 0：禁止 LLAR 中断 • 1：使能 LLAR 中断。
6	R/W	0	LLAV_INTEN 低电平警报有效中断使能 (Low-Level Alarm Valid Interrupt Enable) • 0：禁止 LLAV 中断 • 1：使能 LLAV 中断
5	R/W	0	HLAR_INTEN (High-Level Alarm Removed Interrupt Enable，高电平警报解除中断使能) • 0：禁止 HLAR 中断 • 1：使能 HLAR 中断
4	R/W	0	HLAV_INTEN (High-Level Alarm Valid Interrupt Enable，高电平警报有效中断使能) • 0： 禁止 HLAV 中断 • 1： 使能 HLAV 中断
3:2	–	–	–
1	R/W	0	FIFO_INTEN FIFO 异常中断使能 (FIFO Error Interrupt Enable)

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			GPAI 通道中断控制与状态 (DMA Channel N Interrupt)
位域	类型	默认值	描述
			• 0：禁止 FIFO 异常中断 • 1：使能 FIFO 异常中断
0	R/W	0	DRDY_INTEN 数据有效中断使能 (Data Ready Interrupt Enable) • 0：禁止数据有效中断 • 1：使能数据有效中断

13.2.6.5. 0x108+0x40*n GPAI_CHnPSI

默认值：0x00005DC0			GPAI 通道采样间隔配置 (GPAI Channel N Periodic Sample Interval)
位域	类型	默认值	描述
31:28	–	–	–
27:0	R/W	0x0005DC0	PSI 周期性采样间隔 (Periodic Sample Interval) 定义周期性采样的采样间隔，仅周期采样模式 (PSEN=1) 下有效。 采样间隔为 PSI*TPCLK，示例如下： 假设 `PCLK = 24 MHz`， `PSI = 0x5DC0 = 24000`，采样周期 `TPSI = 24000/24000000 = 0.001 s = 1 ms`。 本字段为 0 表示连续采样，两次采样循环之间没有 延时。

13.2.6.6. 0x110+0x40*n GPAI_CHnHLAT

默认值：0x00000000			GPAI 高电平报警阈值 (Channel N High-Level Alarm Threshold)
位域	类型	默认值	描述
31:28	–	–	–
27:16	R/W	0x000	HLARV 高电平报警解除阈值 (High-Level Alarm Removed Value) 本字段定义了高电平报警解除的比较阈值，仅周期采样模式 (PSEN=1) 下有效： • 当采样值从高于 HLAVV 到低于 HLARV，表示高电平报警解除。 • HLAVV 必须保证不小于 HLARV，(HLAVV-HLARV) 就是高电平报警的迟滞值。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			GPAI 高电平报警阈值 (Channel N High-Level Alarm Threshold)
位域	类型	默认值	描述
15:12	–	–	–
11:0	R/W	0x000	HLAVV 高电平报警有效阈值 (High-Level Alarm Valid Value) 本字段定义了高电平报警有效的比较阈值，仅周期采样模式 (PSEN=1) 下有效： • 当采样值从低于 HLARV 到高于 HLAVV，表示高电平报警有效。 • HLAVV 必须保证不小于 HLARV，(HLAVV-HLARV) 就是高电平报警的迟滞值。

13.2.6.7. 0x114+0x40*n GPAI_CHnLLAT

默认值：0x00000000			GPAI 低电平报警阈值 (Channel N Low-Level Alarm Threshold)
位域	类型	默认值	描述
31:28	–	–	–
27:16	R/W	0x000	LLARV 低电平报警解除阈值 (Low-Level Alarm Removed Value) 本字段定义了低电平报警解除的比较阈值，仅周期采样模式 (PSEN=1) 下有效： • 当采样值从低于 LLAVV 到高于 LLARV，表示低电平报警解除。 • LLAVV 必须保证不大于 LLARV，(LLARV-LLAVV) 就是低电平报警的迟滞值。
15:12	–	–	–
11:0	R/W	0x000	LLAVV 低电平报警有效阈值 (Low-Level Alarm Valid Value) 本字段定义了低电平报警有效的比较阈值，仅周期采样模式 (PSEN=1) 下有效： • 当采样值从高于 LLARV 到低于 LLAVV，表示低电平报警有效。 • LLAVV 必须保证不大于 LLARV，(LLARV-LLAVV) 就是低电平报警的回滞值。

13.2.6.8. 0x118+0x40*n GPAI_CHnACR

默认值：0x00000000			GPAI 报警控制 (GPAI Alarm Control Register)
位域	类型	默认值	描述
31:7	–	–	–

Copyright © 2023-2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			GPAI 报警控制 (GPAI Alarm Control Register)
位域	类型	默认值	描述
6	R/W	0	NLD_DIS 丢弃正常电平的采样数据 (Discard Normal–Level Data) 在高/低电平报警使能，即 HLA_EN=1 或 LLA_EN=1 的情况下，本字段控制是否丢弃保留正常电平的采样数据： • 0：保留正常电平采样数据，存入 FIFO。 • 1：丢弃正常电平采样数据，不会存入 FIFO。
5	R/W	0	LLD_DIS 丢弃低电平的采样数据 (Discard Low–Level Data) 在低电平报警使能 (LLA_EN=1) 情况下，本字段控制是否丢弃保留低电平报警有效之后至低电平报警解除之前的采样数据： • 0：保留低电平采样数据，存入 FIFO。 • 1：丢弃低电平采样数据，不会存入 FIFO。
4	R/W	0	HLD_DIS 丢弃高电平的采样数据 (Discard High–Level Data) 在高电平报警使能 (HLA_EN=1) 情况下，本字段控制是否丢弃保留高电平报警有效之后至高电平报警解除之前的采样数据: • 0：保留高电平采样数据，存入 FIFO。 • 1：丢弃高电平采样数据，不会存入 FIFO。
3:2	–	–	–
1	R/W	0	LLA_EN 低电平报警使能 (Low–Level Alarm Enable) • 0：禁止低电平报警功能，低温警报阈值无效。 • 1：使能低电平报警功能，仅在周期采样模式有效。
0	R/W	0	HLA_EN 高电平报警使能 (High–Level Alarm Enable) • 0：禁止高电平报警功能，高温警报阈值无效。 • 1：使能高电平报警功能，仅在周期采样模式有效。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

13.2.6.9. 0x120+0x40*n GPAI_CHnFCR

默认值：0x00000000			GPAI FIFO 控制 (GPAI Channel N FIFO Control Register)
位域	类型	默认值	描述
31	–	–	–
30:24	RO	0x0	FIFO_DCNT FIFO 有效数据量 (FIFO Data Count) 当前 FIFO 有效数据量： • 0x0 ~ 0x8 为合法值。
23:19	–	–	–
18	R/W1C	0	FIFO_UFS FIFO 下溢出状态 (FIFO Underflow Status) 当前 FIFO 下溢出状态，高电平有效。 本字段有效，会导致 GPAI_CHnINT.FIFO_FLG 置位。 本字段由硬件置位，并由软件写 1 清零。
17	R/W1C	0	FIFO_OFS FIFO 上溢出状态 (FIFO Overflow Status) 本字段定义了当前 FIFO 上溢出状态，高电平有效。 本字段有效，会导致 GPAI_CHnINT.FIFO_FLG 置位。 本字段由硬件置位，并由软件写 1 清零。
16	–	–	–
15:8	R/W	0x0	FIFO_DRTH FIFO 数据有效阈值 (FIFO Data Ready Threshold) 当 FIFO 的数据量达到了本阈值后，就会置位 GPAI_CHnINT.DRDY_FLG 标志位，如果同时 GPAI_CHnINT.DRDY_INTEN 也有效，就会产生中断。 • ◦ 0x00：不设置有效阈值，不会产生 DRDY_FLG 标志。 ◦ 0x01~0x08：有效阈值范围，当 FIFO 数据量达到了对应数值后，就会置位 DRDY_FLG 标志。 ◦ 0x09~0x7F：非法配置，效果等同于 0x00。
7:1	–	–	–
0	R/WAC	0	FIFO_FLUSH FIFO 清空操作 (FIFO Flush)

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			GPAI FIFO 控制 (GPAI Channel N FIFO Control Register)
位域	类型	默认值	描述
			由软件置位硬件清零。软件置位启动 FIFO 清空操作，FIFO 清空操作完成后，由硬件清零。

13.2.6.10. 0x124+0x40*n GPAI_CHnDATA

默认值：0x00000000			GPAI 通道数据 (GPAI Channel N Data)
位域	类型	默认值	描述
31:12	–	–	–
11:0	RO	0x000	DATA 采样数据 (Sample Data) FIFO 数据读取访问入口。 如果 SBC 不为 0，则本寄存器返回的数据，均为经过算术平均（保留整数部分）之后的计算值。

13.2.6.11. 0xFFC VERSION

默认值：0x00000100			GPAI 版本寄存器 (GPAI Version)
位域	类型	默认值	描述
31:0	RO	0x00000100	Version 模块版本号 (Module Version) 采用 BCD 码显示，V1.0

13.3. Resistance Touch Panel (RTP)

电阻触摸屏接口 (RTP) 是一种检测和处理电阻触摸屏输入信号的硬件模块，适用于各种需要触控功能的设备和应用场景。RTP 模块具有下列功能：

- 检测电阻触摸屏的触摸动作：通过检测电阻变化来识别触摸屏上的触摸事件。
- 通过 ADC 获取电阻触摸屏的 X 轴/Y 轴/触摸点的分压值。

  软件驱动通过 X 轴/ Y 轴/触摸点的分压值计算，可以获得电阻触摸屏触摸点的坐标，以及触摸压感值。

RTP 接口模块通过 ADC 来获取电阻触摸屏的电压值，因此在启动 RTP 之前，需要先启动 ADC Interface Module (ADCIM) 模块。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

13.3.1. 特性说明

- 支持 APB 总线访问寄存器
- 仅支持四线 RTP，即 X+/X–/Y+/Y–，用于获取触摸点的电压分压值
- 支持压力感应，能够测量触摸的压力强度
- 支持采样滤波，对输入信号进行滤波以减少噪声干扰，滤波参数可配置
- 支持触摸检测中断，及时响应触摸事件
- 支持 16*16 FIFO，用于存储采样数据

13.3.2. 原理框图

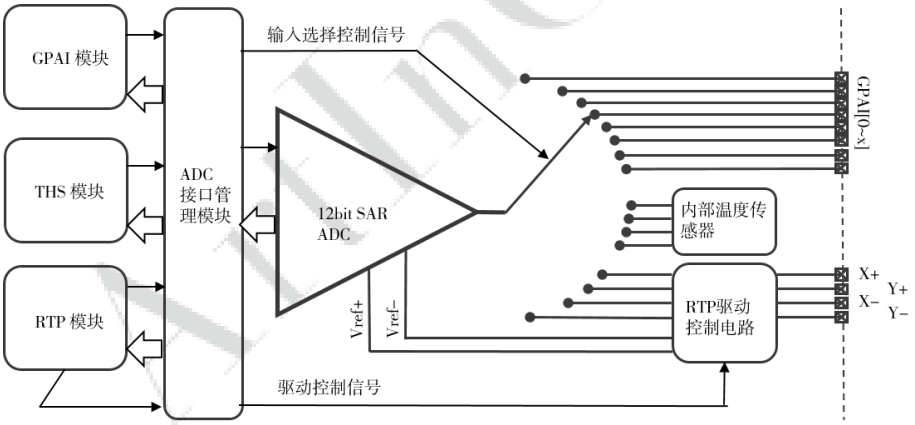


图 13–10 RTP 接口模块与 ADC 连接关系

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

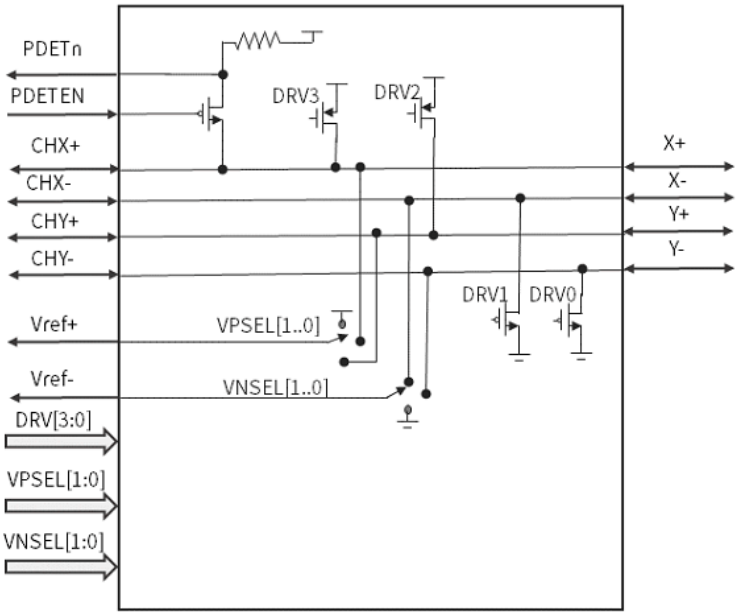


图 13–11 RTP 驱动电路原理框图

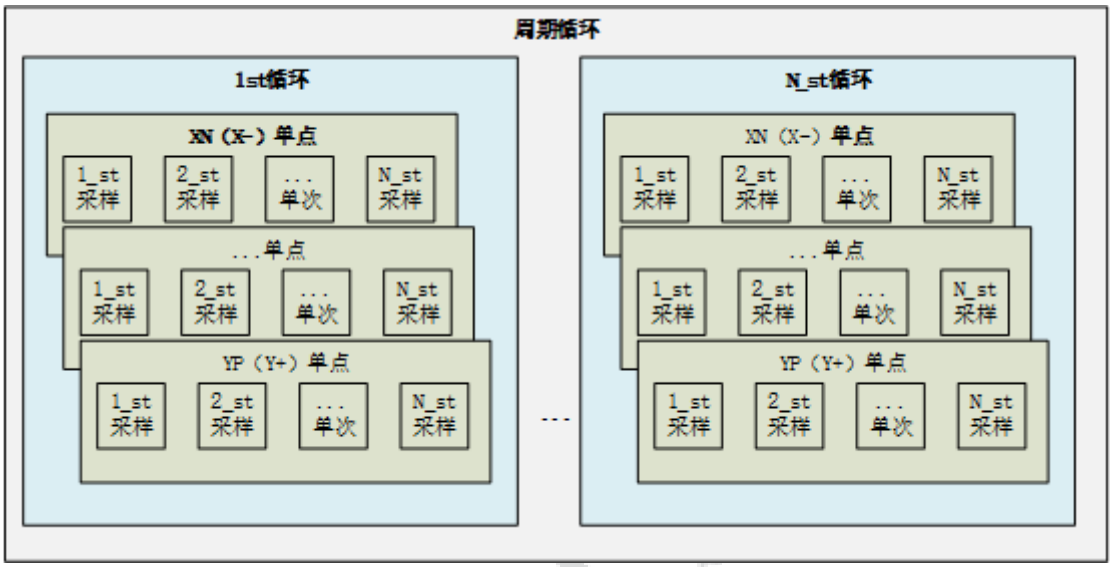
13.3.3. 功能描述

13.3.3.1. 关于采样的定义

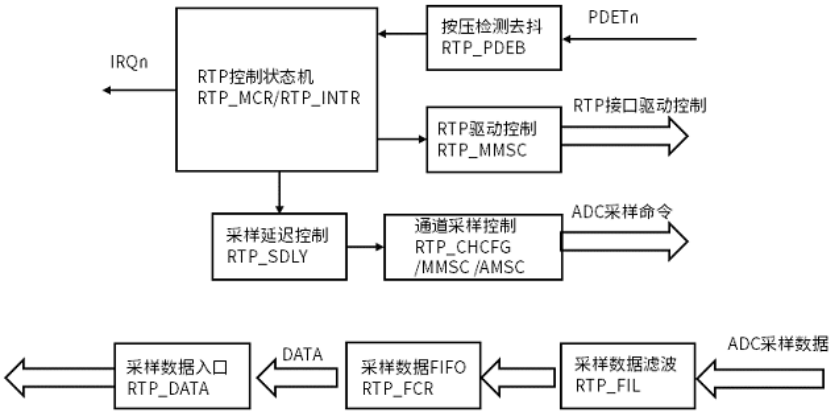
为了方便表述，RTP 模块引入了四个有关采样的定义。

- **单次采样**：任意输入的一次 ADC 采样，对应一条 ADC 采样命令。
- **单点采样**：ADC 在任意一个时间点或者信号点开始的连续采样，有可能是单次采样，也有可能包含多次采样。 例如 ADC 在一个时间点开始对某个特定信号进行多次采样（多条 ADC 采样命令），以获得多个采样值，然后进行适当的滤波（例如算数平均），以获得一个更加准确和稳定的结果。
- **单循环采样**：自动采样模式下，为了能够计算出一个完整的触摸坐标，需要对多个点进行采样，一个单循环采样包括多个采样点， 包含的采样点数不确定，根据不同的自动模式会不同，例如自动模式 1（XN–YN），就对应着两个采样点 X– 和 Y–。
- **周期循环采样**：自动采样模式下，为了获取连续的触摸坐标变化，需要间隔一定时间重复完成单循环采样。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



13.3.3.2. RTP 工作流程与控制机制



13.3.3.3. 采样模式描述

RTP 模块的采样模式涉及有四个采样点，分别是 X+，X−，Y+，Y−。通过不同的 RTP 驱动电压组合和采样点选择，可以定义多种采样模式，以满足不同的触摸检测需求。

以下定义了八种常见采样模式，每种采样模式都对应不同的驱动电压配置和采样点选择，以适应不同的触摸场景和精度要求。通过合理选择和配置采样模式，可以优化触摸屏的性能和响应速度：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 图 13–12 : XP 采样所示分别是单点触摸和两点触摸的情况，DRV[3:0] = 0x5，VPSEL[1:0] = VNSEL[1:0] =0x2，采样点选择 X+。

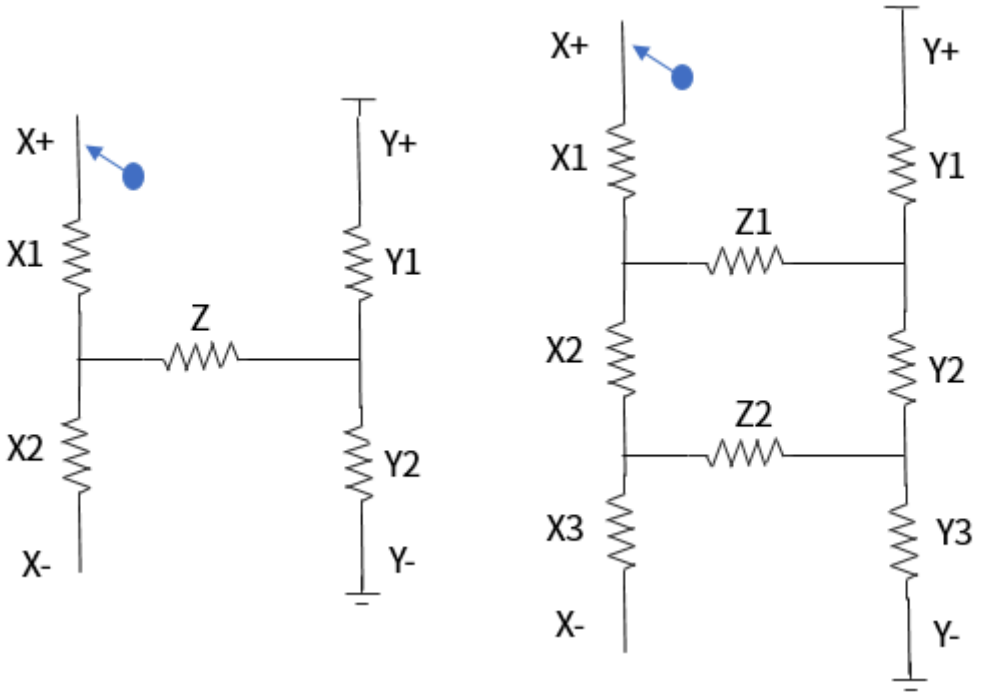


图 13–12 XP 采样

- 图 13–13 : XN 采样所示分别是单点触摸和两点触摸的情况，DRV[3:0] = 0x5，VPSEL[1:0] = VNSEL[1:0] = 0x2，采样点选择 X–。

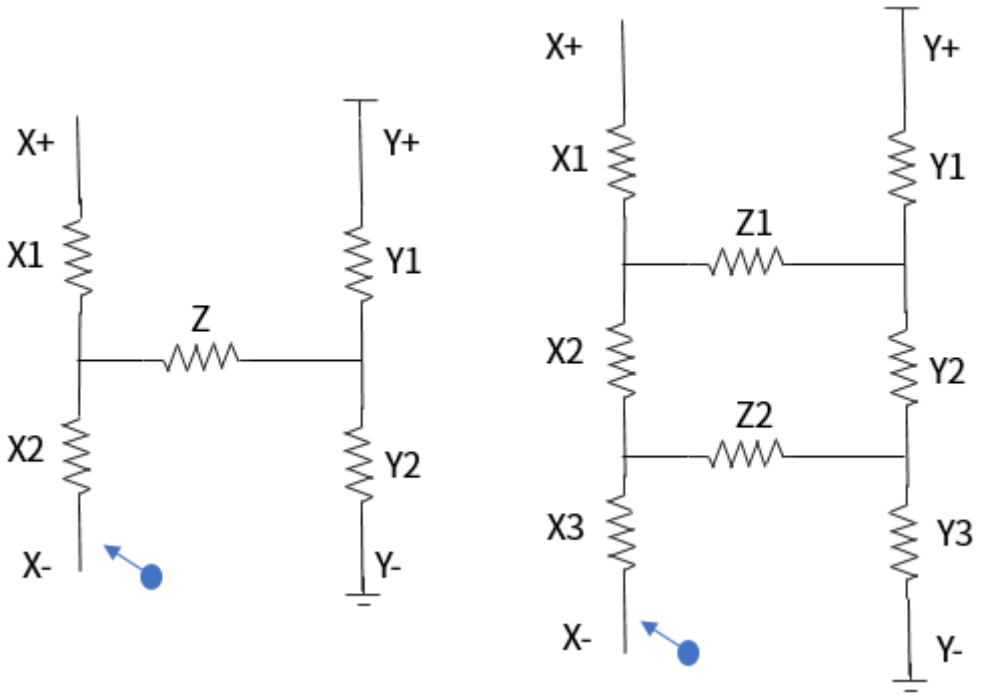


图 13–13 XN 采样

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 图 13–14 : YP 采样所示分别是单点触摸和两点触摸的情况，DRV[3:0] = 0xA，VPSEL[1:0] = VNSEL[1:0] = 0x1，采样点选择 Y+。

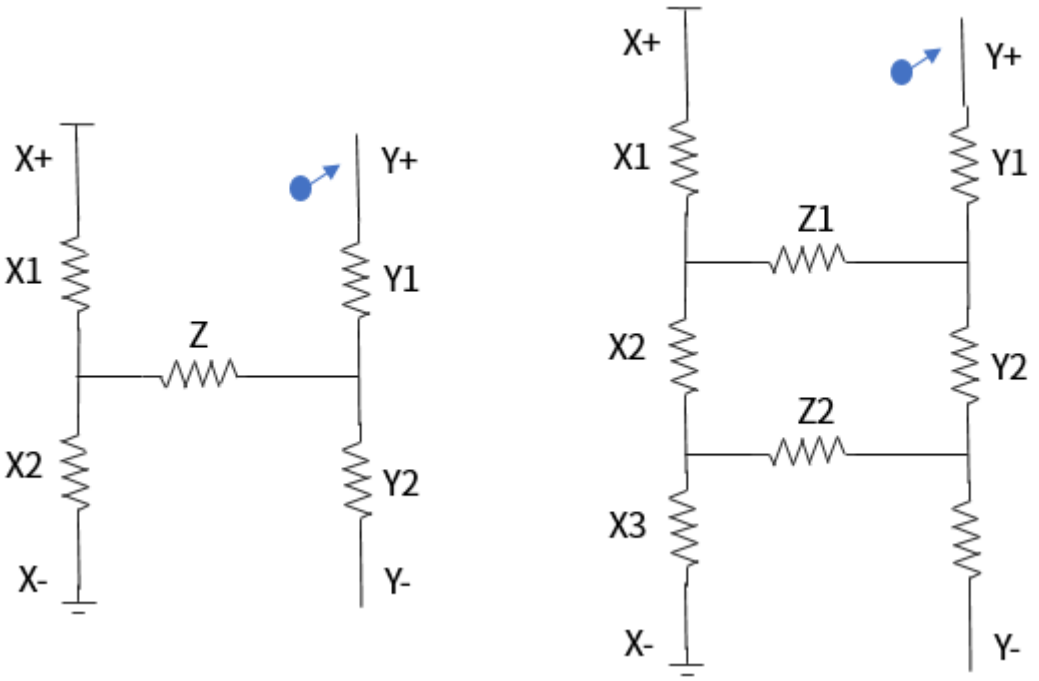


图 13–14 YP 采样

- 图 13–15 : YN 采样所示分别是单点触摸和两点触摸的情况，DRV[3:0] = 0xA，VPSEL[1:0] = VNSEL[1:0] = 0x1，采样点选择 Y–。

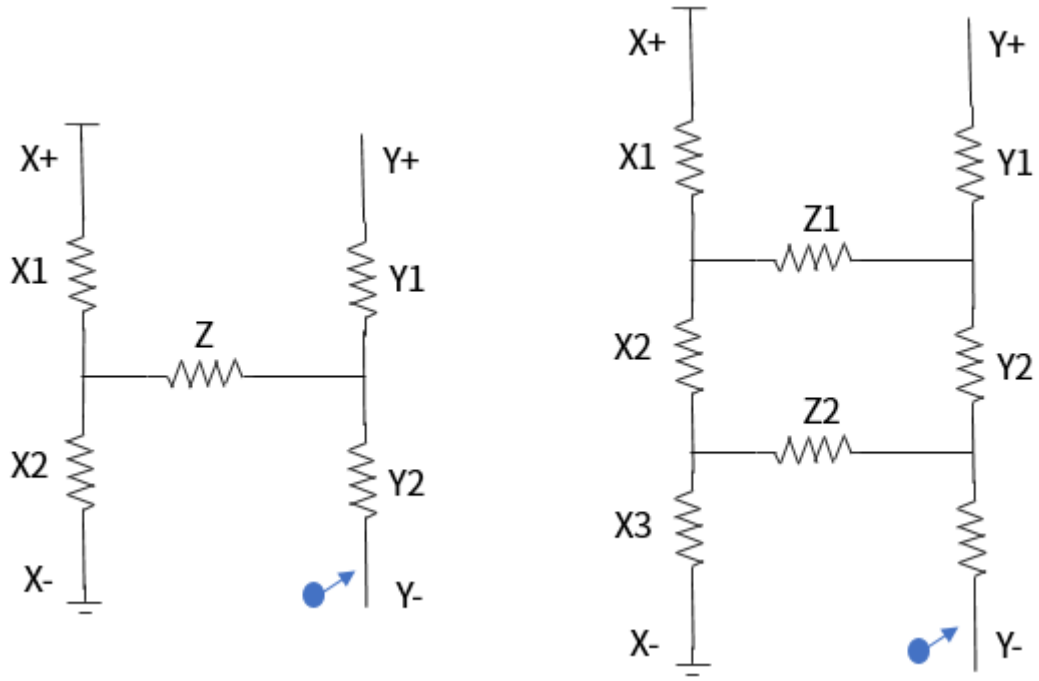


图 13–15 YN 采样

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

• 图 13–16 : ZA 采样所示分别是单点触摸和两点触摸的情况，DRV[3:0] = 0x9，VPSEL[1:0] = 0x1，VNSEL[1:0] = 0x2，采样点选择 Y+。

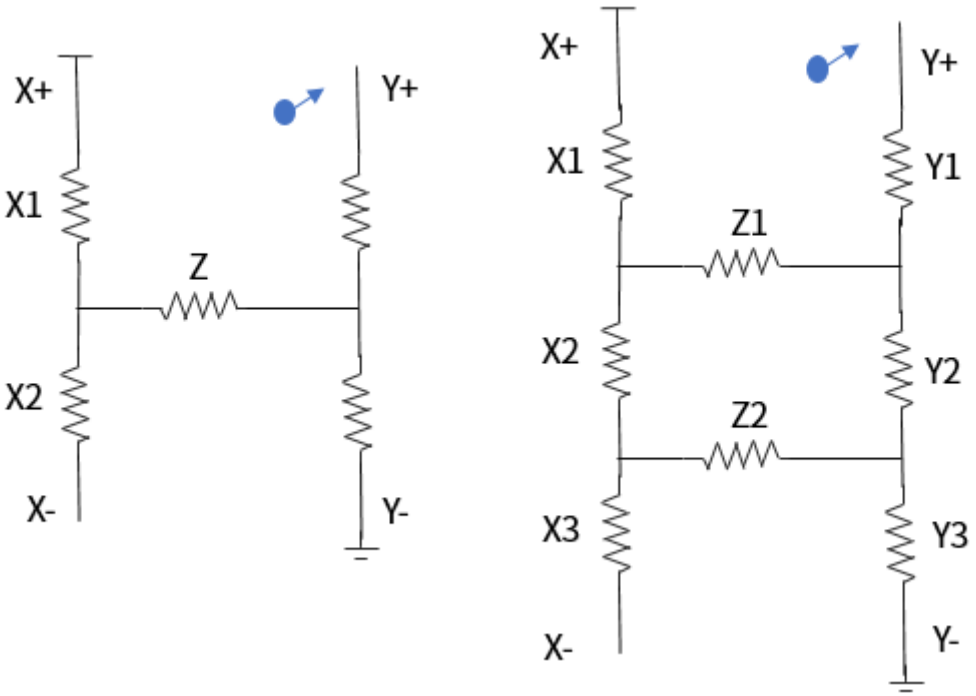


图 13–16 ZA 采样

• 图 13–17 : ZB 采样所示分别是单点触摸和两点触摸的情况，DRV[3:0] = 0x9，VPSEL[1:0] = 0x1，VNSEL[1:0] = 0x2，采样点选择 X–。

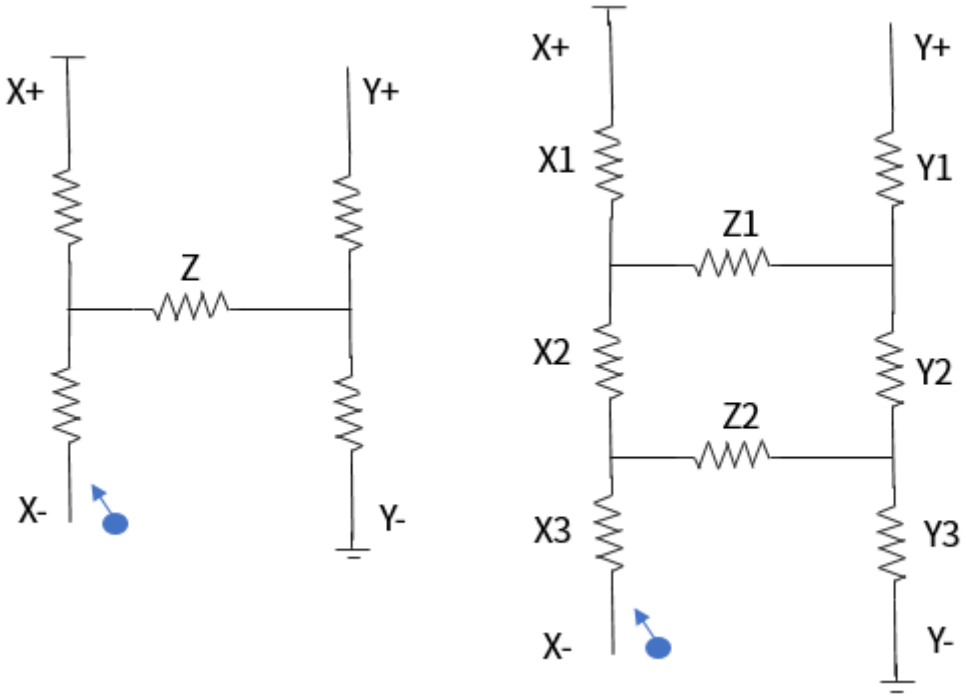


图 13–17 ZB 采样

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

- 图 13–18 : ZC 采样所示分别是单点触摸和两点触摸的情况，DRV[3:0] = 0x6，VPSEL[1:0] = 0x2，VNSEL[1:0] = 0x1，采样点选择 X+。

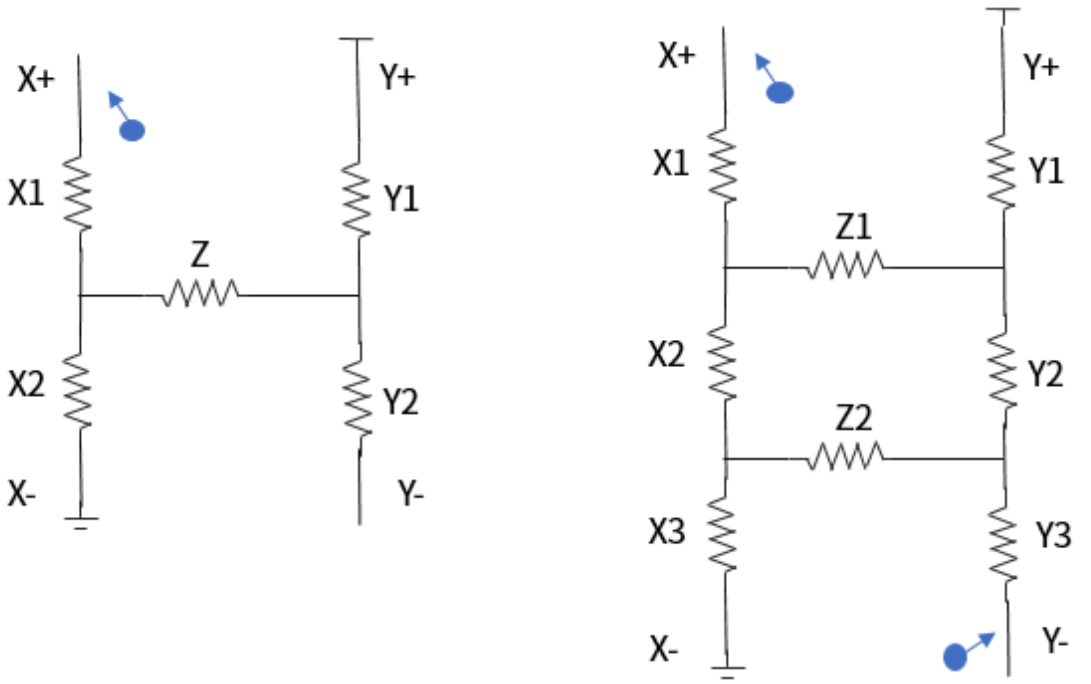


图 13–18 ZC 采样

- 图 13–19 : ZD 采样 所示分别是单点触摸和两点触摸的情况，DRV[3:0] = 0x6，VPSEL[1:0] = 0x2，VNSEL[1:0] = 0x1，采样点选择 Y–。

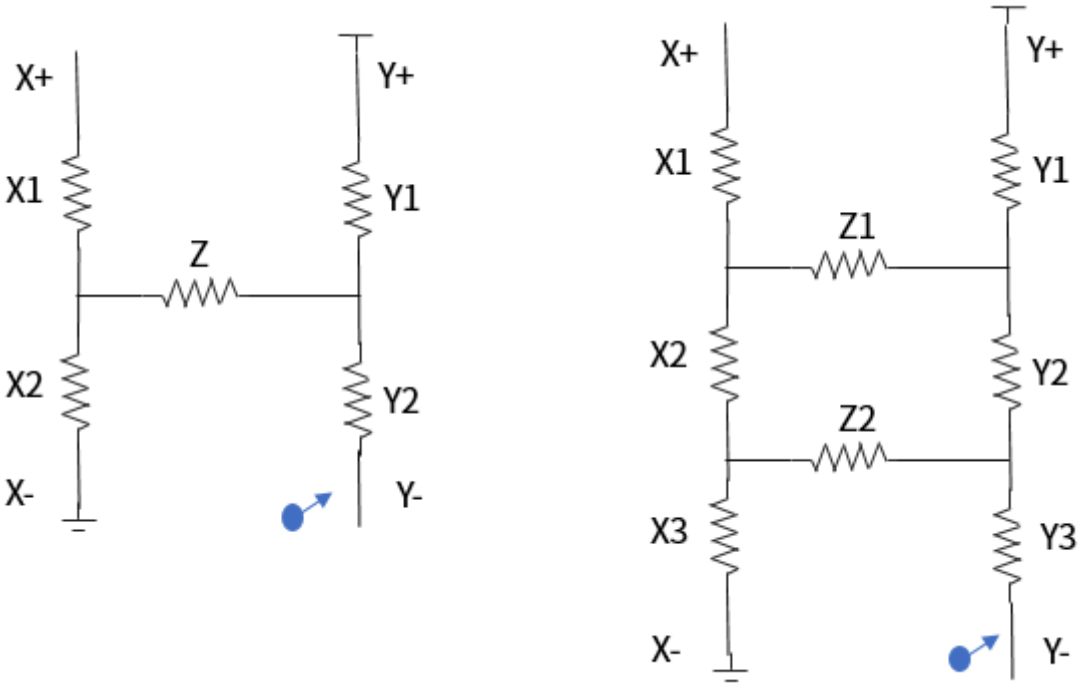


图 13–19 ZD 采样

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

13.3.3.4. 工作模式描述

RTP 模块规定了两类工作模式：

- 手动工作模式：采样模式由软件通过控制寄存器配置来构造而来，不限于采样模式描述中定义的八种采样模式。

  不同采样模式的先后顺序由软件进行控制，可以获得最大的灵活度，以面对未来应用的不确定性。

- 自动工作模式：按照预先定义的八种采样模式之一以及预定的顺序进行采样。

  最大程度简化软件驱动控制，一键完成所需的所有采样工作。软件只需使用获得的采样值进行坐标运算即可。根据不同应用需要，自动模式分了四种情况，详细参考寄存器描述。

13.3.3.4.1. 自动模式下采样数据滤波

自动工作模式下，采样数据滤波包括以下步骤：

1. 排序：把单点多次采样结果进行排序，找出最大和最小的值。
2. 超限判断，用最大值–最小值，然后进行相对值和绝对值分别比较，如果超过其中任一值，则整组采样数据判为无效，否则就认为数据有效。
3. 算数平均：把最大值/最小值排除之后，对中间值进行算数平均，得到的结果作为单点采样的最终结果存入 FIFO。

滤波配置包含：

- 无滤波。
- 2/4：单点采样连续采样 4 次，获得同一个点的连续的 4 个 ADC 样本数据，然后去掉最大和最小，再对剩下的数据进行算数平均。
- 4/6：单点采样连续采样 6 次，获得同一个点的连续的 6 个 ADC 样本数据，然后去掉最大和最小，再对剩下的数据进行算数平均。
- 4/8：单点采样连续采样 8 次，获得同一个点的连续的 8 个 ADC 样本数据，然后去掉最大 2 个和最小 2 个，再对剩下的数据进行算数平均。

示例

假设采样模式为 XP，滤波配置为 4/6，XYRUR=1/64，XYAUR=16，则对 X+进行连续采样，得到 6 个数值，分别为 A，B，C，D，E 和 F，并且 A < B < C < D < E < F，则采样数据滤波步骤如下：

1. 排序：最小值 A，最大值 F。
2. 超限判断：如果 ((F−A) > XYAUR) || ((F−A) > ((F+A)/2* XYRUR)) == 1（绝对值或相对值超出限制范围），则保留采样所得的数据，并立即停止自动采样。

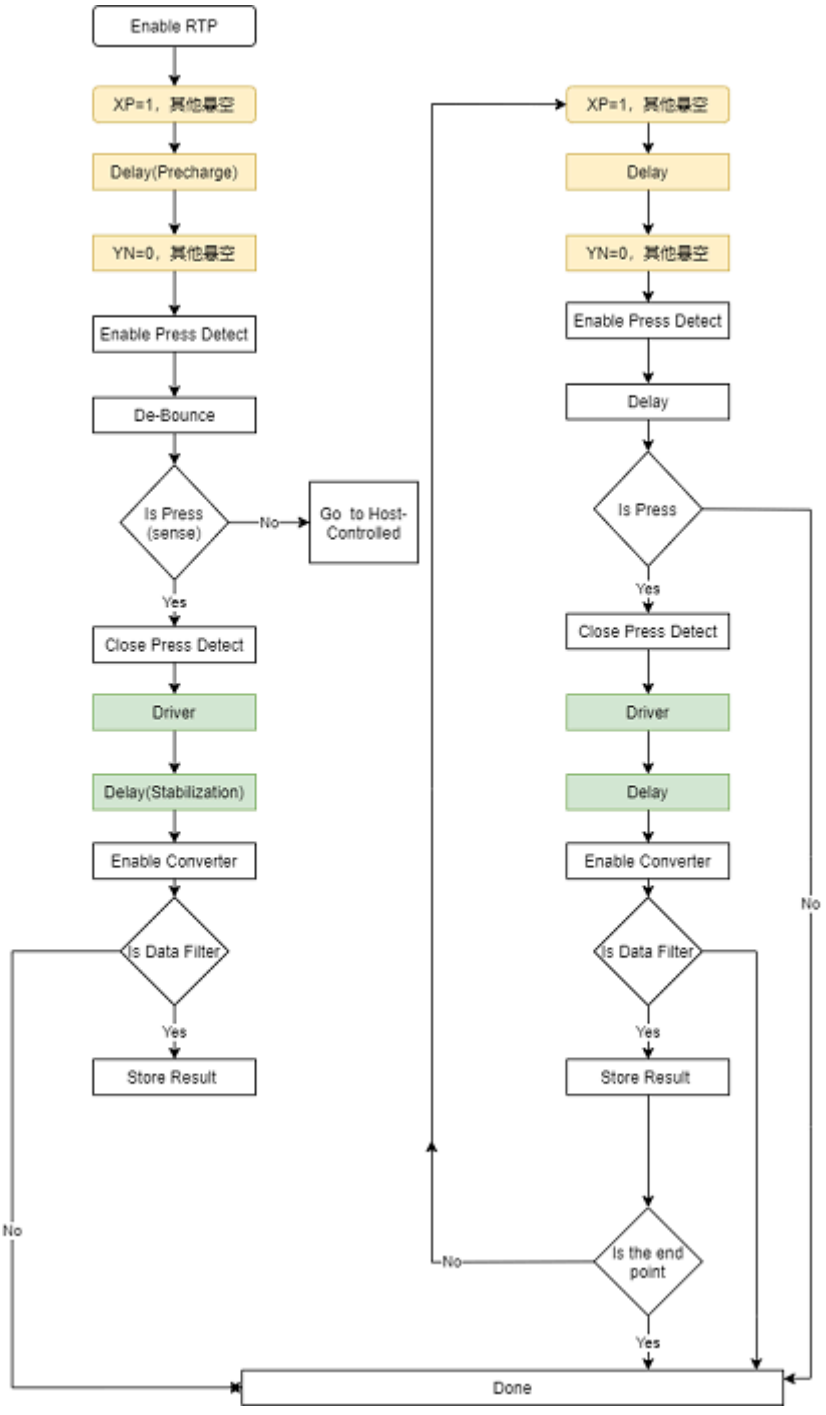
   数据超限与抬起都会导致 RTP 进入 Error 状态并停止自动采样（仅#unique_215/unique_215_Connect_42_table_ezb_2dc_bzbMODE>0 有效），无需等待自动采样完成，但需要等待正在执行的单点采样完成，需软件清除采样不完整与数据超限标志。
3. 算术平均：DATA = (B + C + D + E) ÷ 4。

> **注：**
>
> 超限判断与算术平均同时进行。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

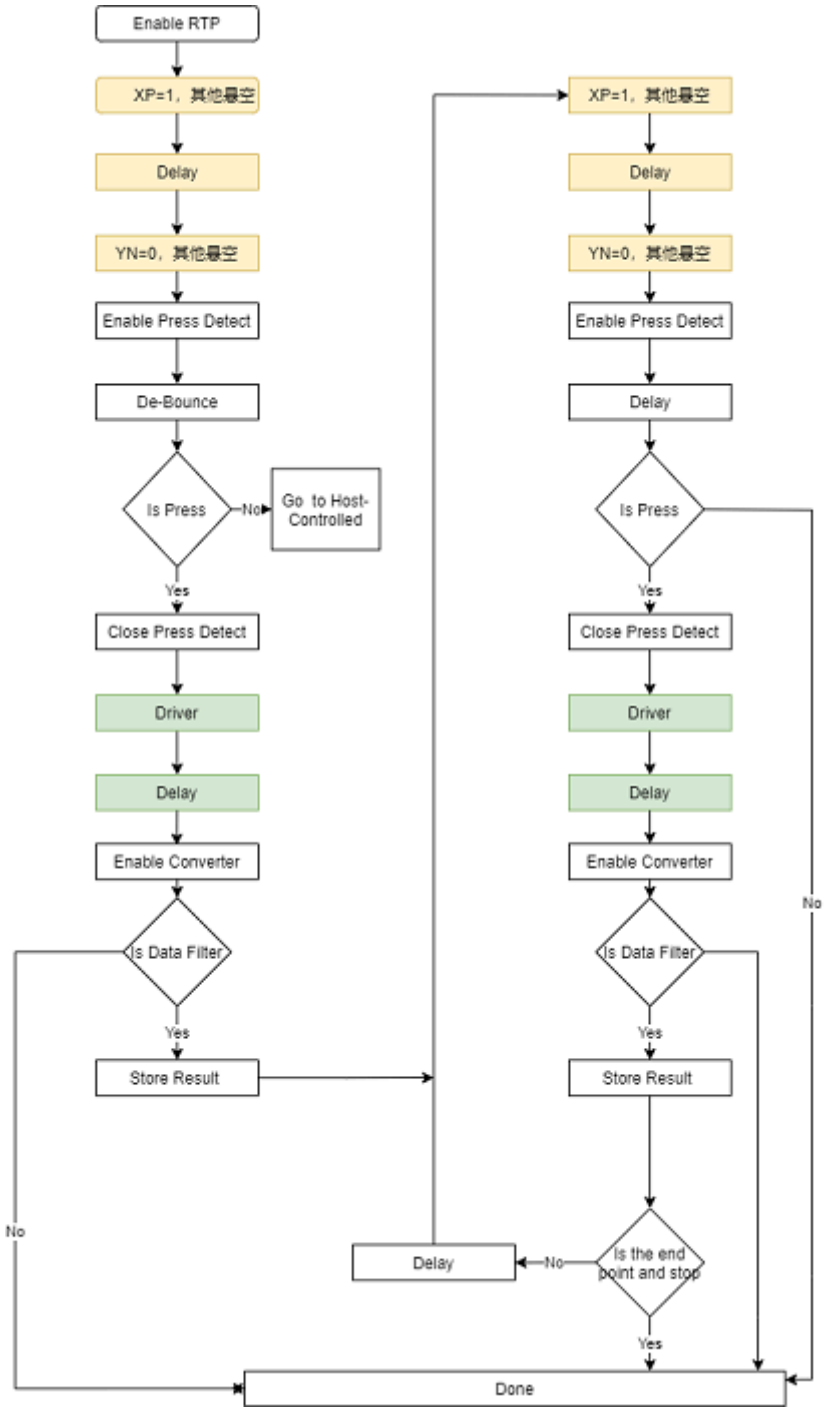
13.3.3.4.2. 自动模式下的采样循环描述

自动模式单循环采样说明：



自动模式周期循环采样说明：

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved



13.3.3.4.3. 自动模式下的单点采样描述

自动模式下单点采样步骤如下：

1. 检查按压检测信号。

   确保 DRV[3:0]=4’h1，并且 PDETEN 有效，检查经过去抖前的按压检测信号（PDETi）。

   如果按压有效，则进入步骤 2，否则置位 FLG（采样循环不完整标志），并进入步骤 6。

2. 配置驱动 RTP 电路。

   根据采样模式，分别配置 DRV/ VPSEL/ VNSEL 驱动电路，同时 PDETEN 无效，进入步骤 3。

3. 发送 ADC 采样命令。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

根据滤波配置，连续向 ADC 发出多个采样命令，采样优先级 PRI = 1，收到对应数量的采样数据后，进入步骤 4。

4. 采样数据滤波。

   按照前面描述的方式，对采样数据进行滤波，如果滤波完成则进入步骤 5，否则置位 FLG（数据超限标志），并进入步骤 6。

5. 完成采样。

   数据写入 FIFO。

6. 等待软件干预。

   进入 Error 状态，直到（SCI_FLG & DOUR_FLG）= 0，重新进行下一次采样循环。

13.3.3.5. 按压检测

图 13–20 : 按压检测中，按压检测信号从外部到内部经过屏蔽控制以及去抖两个过程。

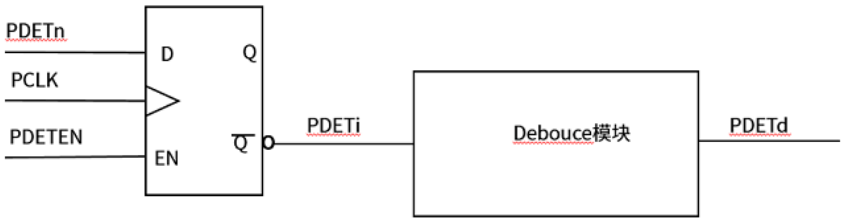


图 13–20 按压检测

1. 按压检测信号 PDETn 输出采用 PDETEN 信号进行屏蔽，当 PDETEN 为高电平，PDETn 输入有效，否则输入无效。
2. 按压检测信号经过屏蔽控制之后，还需要经过去抖才能为模块内部所使用。图 13–21 : 去抖原理如下。

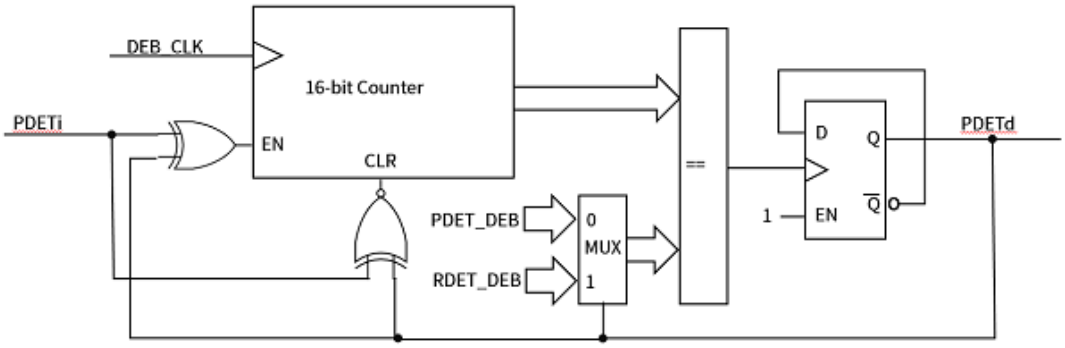


图 13–21 去抖原理

• 按压去抖过程：

a. 模块复位后，内部 PDETd=0，表示已抬起。

b. 当按压信号有效（PDETi=1）时，去抖计数开始 。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

c. 期间如果检测信号有抖动（PDETi=0），计数器就会被清零，如果在足够的时 间内无抖动 (PDETi=1)，计数器就会计数到按压检测的去抖值（PDET_DEB）。

d. PDETd 取反（PDETd=1），表示按压事件已被确定，随后计数器就会被清零。

• 抬起去抖过程：

a. 当触摸抬起（PDETi=0），去抖计数开始。

b. 期间如果检测信号有抖动（PDETi=1），计数器就会被清零，如果在足够 的时间内无抖动 (PDETi=0)，计数器就会计数到抬起检测的去抖值（RDET_DEB）。

c. PDETd 取反（PDETd=0 ），表示抬起事件已被确定，随后计数器就会被清零。

13.3.3.6. 中断产生机制

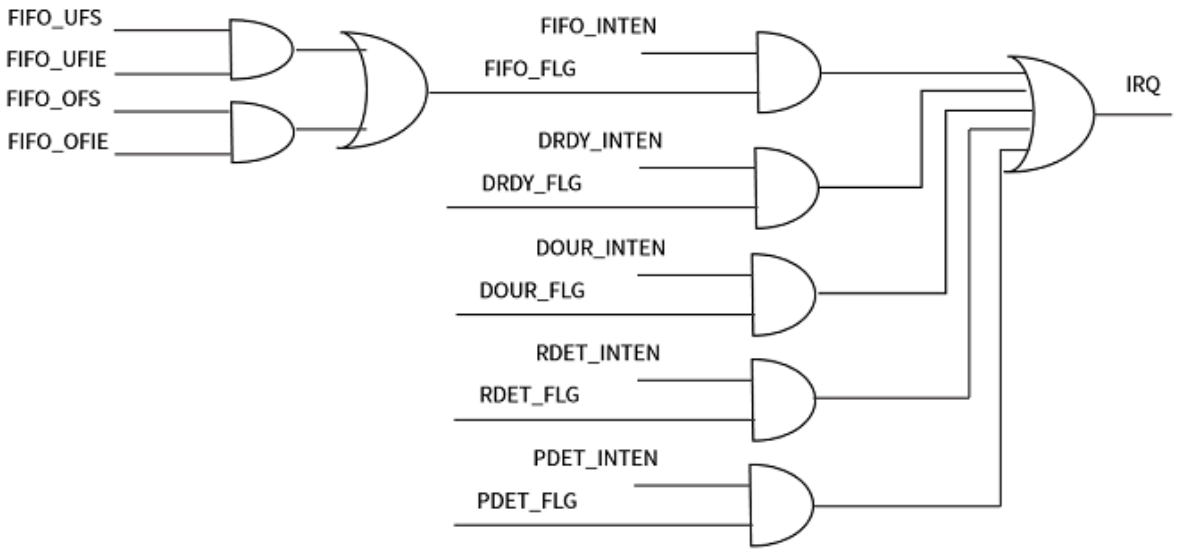


图 13–22 中断连接框图

如图所示，RTP 模块有 FIFO_FLG、DOUR_FLG、SCI_FLG、DRDY_FLG、RDET_FLG、PDET_FLG 六种中断源，每种中断源有独立使能开关。

13.3.4. 寄存器列表

表 13–3 RTP 寄存器列表

偏移	缩写	寄存器名称	章节
0x000	RTP_MCR	RTP 模块控制	0x000 RTP_MCR
0x004	RTP_INTR	RTP 中断控制与状态	0x004 RTP_INTR
0x008	RTP_PDEB	RTP 按压检测去抖配置	0x008 RTP_PDEB

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 13–3 RTP 寄存器列表 (续)

偏移	缩写	寄存器名称	章节
0x00C	RTP_PCTL	RTP 按压检测延时控制	0x00C RTP_PCTL
0x010	RTP_CHCFG	RTP 通道配置	0x010 RTP_CHCFG
0x014	RTP_MMSC	RTP 手动模式控制	0x014 RTP_MMSC
0x018	RTP_FIL	RTP 自动采样滤波配置	0x018 RTP_FIL
0x01C	RTP_AMSC	RTP 自动模式采样控制	0x01C RTP_AMSC
0x020	RTP_FCR	RTP FIFO 控制	0x020 RTP_FCR
0x024	RTP_DATA	RTP 数据	0x024 RTP_DATA
0x028	RTP_DELAY	RTP 延时控制	0x028 RTP_DELAY
0xFFC	VERSION	RTP 版本	0xFFC VERSION

13.3.5. 寄存器描述

13.3.5.1. 0x000 RTP_MCR

默认值：0x00001000			RTP 模块控制 (RTP Module Control Register)
位域	类型	默认值	描述
31:17	–	–	–
16	R/W	0	PDET_BYP Press Detect Bypass，按压检测屏蔽功能 仅在自动模式（RTP_MODE > 0）下有效。 RTP 自动采样过程中，每个单点采样完成之后，是否判断按压状态： • 0: 判断按压检测信号，即如果自动采样过程中抬起，会有可能产生 SCI（采样循环不完整）中断。 如果自动采样过程中抬起，会有可能产生 SCI（采样循环不完整）中断。 • 1: 屏蔽按压检测信号，即自动采样过程中不会判断按压检测信号。
15:13	–	–	–
12	RO	1	PDET_STS RTP Press Detect Status，RTP 按压检测状态

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00001000			RTP 模块控制 (RTP Module Control Register)
位域	类型	默认值	描述
			反映 RTP 模块的按压检测 （PDETn）输入信号状态： • 0: 按压状态 • 1: 抬起状态。
11:10	–	–	–
9	R/W	0	PU_SEL RTP Press Detect Pull–up Resistor Select，RTP 按压检测上拉电阻选择 控制按压检测上拉电阻阻值： • 0: 上拉电阻 = 50 KΩ • 1: 上拉电阻 = 90 KΩ。
8	R/W	0	PDET_EN RTP Press Detection Enable，RTP 按压检测使能 控制按键检测使能信号： • 0: 按压检测禁止，关闭 X+ 信号上的按压检测上拉电阻，检测状态 PDET_STS 不能真实反映按压真实状态。 • 1: 按压检测使能，打开 X+ 信号上按压检测上拉电阻，检测状态 PDET_STS 反映按压真实状态。
7:4	R/W	0x0	MODE RTP Mode Select, RTP 工作模式选择，包括手动模式以及多种自动模式： • 0x0: 手动模式：包括 RTP 驱动控制，采样命令，采样滤波，采样选择均由软件手动完成。 • 0x1: 自动模式 1，对应单点无压感功能，自动完成 XN –YN 采样，并返回滤波后的采样值。 • 0x2: 自动模式 2，对应单点有压感，自动完成 XN –YN –ZA –ZB 采样，并返回滤波后的采样值。 • 0x3: 自动模式 3，对应两点有压感，自动完成 XN – XP –YN –YP –ZA –ZB 采样，并返回滤波后的采样值。 • 0x4: 自动模式 4，对应全采样，自动完成 XN – XP –YN –YP –ZA –ZB –ZC –ZD 采样，并返回滤波后的采样值。
3:1	–	–	–
0	R/W	0	EN RTP Module Enable，RTP 模块使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00001000			RTP 模块控制 (RTP Module Control Register)
位域	类型	默认值	描述
			• 0: RTP 被禁止，停止工作 • 1: RTP 被使能。

13.3.5.2. 0x004 RTP_INTR

默认值：0x00000000			RTP 中断状态与控制 (RTP Interrupt Register)
位域	类型	默认值	描述
31:22	–	–	–
21	R/W1C	0	SCI_FLG RTP Sample Cycle Incomplete Flag，采样循环不完整标志 仅在自动模式 (RTP_MODE>0) 下有效。 数据超限与抬起都会导致不完整标志起来。 RTP 单点采样完成之后出现抬起 (PDETn = 0) 或数据超限，都会导致采样循环被打断，会置位 SCI_FLG 标志位。 • 0 : RTP 无新的采样循环不完整事件。 • 1 : RTP 有新的采样循环不完整事件。 清除该标志之后，硬件自动从第一点重新开始，完成单循环采样。
20	R/W1C	0	DOUR_FLG RTP Data Over Uncertain Range Flag，数据超限（超出范围）标志 仅在自动模式且滤波功能使能 ((RTP_MODE > 0) && (FIL > 0)) 下有效。 • 0: RTP 无新的数据超限事件。 • 1: RTP 有新的数据超限事件。
19	RO	0	FIFO_FLG RTP FIFO Error Flag，RTP FIFO 异常标志 总的 FIFO 异常状态： • 0: RTP 无 FIFO 异常状况。 • 1: RTP 发生了 FIFO 异常状况。 具体异常状态见 FIFO 管理寄存器（0x020 RTP_FCR）定义。FIFO_FLG 标志位不能被直接清除，可以通过清除 FIFO 管理寄存器0x020 RTP_FCR中的异常状态间接清除本标志。
18	R/W1C	0	DRDY_FLG RTP FIFO Data Ready Flag，RTP FIFO 数据有效标志

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			RTP 中断状态与控制 (RTP Interrupt Register)
位域	类型	默认值	描述
			表示 FIFO 的数据数量满足了 FIFO 数据有效阈值，具体阈值参考 FIFO 控制寄存器 0x020 RTP_FCR。 • 0: RTP 无新的数据有效事件。 • 1: RTP 有新的数据有效事件。
17	R/W1C	0	RDET_FLG RTP Rise Detect Flag，RTP 抬起检测标志 表示 RTP 控制器检测到了从按压到抬起的一个变化过程。 RTP 抬起检测标志反映去抖之后的检测结果。 • 0: RTP 无新的抬起事件。 • 1: RTP 发生新的抬起事件。
16	R/W1C	0	PDET_FLG RTP Press Detect Flag，RTP 按压检测标志 表示 RTP 控制器检测到了从抬起到按压的一个变化过程，本标志是反映去抖之后的检测结果。 • 0: RTP 无新的按压事件； • 1: RTP 发生新的按压事件。
15:6	–	–	–
5	R/W	0	SCI_INTEN RTP Sample Cycle Incomplete Interrupt Enable，采样循环不完整中断使能 RTP 采样循环不完整（采样循环被打断）中断使能： • 0: 禁止 RTP 采样循环不完整中断。 • 1: 使能 RTP 采样循环不完整中断。
4	R/W	0	DOUR_INTEN（） RTP Data Over Uncertain Range Interrupt Enable，RTP 数据超限（超出确定范围）中断使能： • 0: 禁止 RTP 数据超限中断。 • 1: 使能 RTP 数据超限中断。
3	R/W	0	FIFO_INTEN RTP FIFO Interrupt Enable，RTP FIFO 异常中断使能 • 0: 禁止 RTP FIFO 异常中断。 • 1: 使能 FIFO 异常中断。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			RTP 中断状态与控制 (RTP Interrupt Register)
位域	类型	默认值	描述
			FIFO 异常状态参考 FIFO 控制寄存器（0x020 RTP_FCR）定义。
2	R/W	0	DRDY_INTEN（ ） RTP Data Ready Interrupt Enable，RTP 数据有效中断使能 • 0: 禁止 RTP FIFO 数据有效中断； • 1: 使能 RTP FIFO 数据有效中断。 数据有效的阈值参考 FIFO 控制寄存器（0x020 RTP_FCR）中的配置。
1	R/W	0	RDET_INTEN RTP Rise Detect Interrupt Enable，RTP 抬起检测中断使能信号： • 0: 禁止 RTP 抬起检测中断； • 1: 使能 RTP 抬起检测中断。
0	R/W	0	PDET_INTEN RTP Press Detect Interrupt Enable，RTP 按压检测中断使能信号： • 0: 禁止 RTP 按压检测中断； • 1: 使能 RTP 按压检测中断。

13.3.5.3. 0x008 RTP_PDEB

默认值：0x23012301			RTP 按压检测去抖配置 (RTP Press Detect Enable Debounce)
位域	类型	默认值	描述
31:28	R/W	0x2	FLRDET_DEB First Level Rise Detect Debounce，一级抬起去抖配置 抬起事件去抖时间计数，计数单位为 TPCLK，去抖时间范围为 0~（2^4–1）*TPCLK。 当本字段为 0（即 0x0000）时，抬起去抖功能无效。 推荐值 0xF，按压与抬起去抖参数需设置一样
27:24	R/W	0x3	SLRDET_DEB Second Level Rise Detect Debounce，二级抬起去抖配置 抬起事件去抖时间计数，计数单位为 TRiseDivClk，去抖时间范围为 0~（2^4–1）*TRiseDivClk。 当本字段为 0（即 0x0000）时，抬起去抖功能无效。 推荐值 0xF，按压与抬起去抖参数需设置一样
23:16	R/W	0x01	SLRDET_DEBDC Second Level Rise Detect Debounce Div Clk，二级抬起去抖分频计数

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x23012301			RTP 按压检测去抖配置 (RTP Press Detect Enable Debounce)
位域	类型	默认值	描述
			抬起事件去抖时间计数 RiseDivClk，计数单位为 TPCLK，分频时间范围为 0~（2^8–1）*256*TPCLK。 当本字段为 0（即 0x0000）时，抬起去抖功能无效。 推荐值 0x0F,按压与抬起去抖参数需设置一样。
15:12	R/W	0x2	FLRDET_DEB First Level Press Detect Debounce，一级按压去抖配置 按压事件去抖时间计数，计数单位为 TPCLK，去抖时间范围为 0~（2^4–1）*TPCLK。 当本字段为 0（即 0x0000）时，抬起去抖功能无效。 推荐值 0xF，按压与抬起去抖参数需设置一样
11:7	R/W	0x3	SLRDET_DEB Second Level Press Detect Debounce，二级按压去抖配置 按压事件去抖时间计数，计数单位为 TPressDivClk，去抖时间范围为 0~（2^4–1）*TPressDivClk。 当本字段为 0（即 0x0000）时，抬起去抖功能无效。 推荐值 0xF，按压与抬起去抖参数需设置一样
7:0	R/W	0x01	SLRDET_DEBDC Second Level Press Detect Debounce Div Clk，二级按压去抖分频计数 按压事件去抖时间计数 PressDivClk，计数单位为 TPCLK，分频时间范围为 0~（2^8–1）*256*TPCLK。 当本字段为 0（即 0x0000）时，抬起去抖功能无效。 推荐值 0x0F,按压与抬起去抖参数需设置一样。

13.3.5.4. 0x00C RTP_PCTL

默认值：			RTP 按压检测延时控制 (RTP Press Dectect Control Delay)
位域	类型	默认值	描述
31:16	R/W		PDET_ISD Press Dectect Idle State Delay，按压检测空闲状态延时，定义了 RTP 在 IDLE 状态与 PDET_EN==1 条件下，延长一定时间进行按压检测，仅在自动模式（RTP_MODE > 0）下有效。 本字段表示按压检测空闲状态延时为 `(PDET_ISD+1) *TPCLK`，推荐值 0xFFFF。 使用限制：如果已使用二级去抖功能，则需要满足 PDET_ISD ≥(DET_DEB +1)*DET_DEBDC*256
15:0	R/W	0x00FF	PDET_DRV2DETD

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：			RTP 按压检测延时控制 (RTP Press Dectect Control Delay)
位域	类型	默认值	描述
			Press Dectect Drive-to-Detect Delay，按压检测驱动到检测延时 本字段定义了 RTP 自动采样过程中，RTP 由驱动状态变为检测状态时，延时一段时间之后才开始按压检测，仅在自动模式（RTP_MODE > 0）下有效。 在自动采样模式下，两点采样之间会对按压检测信号进行判断。为了更加高效，这个判断点未经过去抖（不受 PDEB 寄存器控制），需要在驱动到检测之间增加一个 Delay 控制，以适应 XN 的不同的外部负载。 按压检测驱动到检测的延时为 (PDET_DRV2DET+1) *TPCLK，有效值为 1~65536 个 PCLK 周期。例如，默认值为 0x0F 表示这个延时为 16 个 PCLK 周期。

13.3.5.5. 0x010 RTP_CHCFG

默认值：0xCEDF0801			RTP 通道配置 (RTP Channel Config)
位域	类型	默认值	描述
31:28	RO	0xC	XP_CHN X+ ADC Channel Number，X+对应的 ADC 通道号，显示系统分配给 X+信号的 ADC 通道号。 硬件会根据此 ADC 通道号匹配 ADC 并获取返回的数据，软件驱动不需要使用，本字段仅作 Debug 用。
27:24	RO	0xE	YP_CHN Y+ ADC Channel Number， Y+对应的 ADC 通道号，显示系统分配给 Y+信号的 ADC 通道号 硬件会根据此 ADC 通道号匹配 ADC 并获取返回的数据，软件驱动不需要使用，本字段仅作 Debug 用。
23:20	RO	0xD	XN_CHN X- ADC Channel Number，X-对应的 ADC 通道号，显示系统分配给 X- 信号的 ADC 通道号 硬件会根据此 ADC 通道号匹配 ADC 并获取返回的数据，软件驱动不需要使用，本字段仅作 Debug 用。
19:16	RO	0xF	YN_CHN Y- ADC Channel Number，Y-对应的 ADC 通道号，显示系统分配给 Y- 信号的 ADC 通道号 硬件会根据此 ADC 通道号匹配 ADC 并获取返回的数据，软件驱动不需要使用，本字段仅作 Debug 用。
15:8	R/W	0x08	ADC_ACQ 本次采样的 ADC 通道的捕获时间长度。

Copyright © 2023-2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0xCEDF0801			RTP 通道配置 (RTP Channel Config)
位域	类型	默认值	描述
			计数值单位为 CLK 的周期数，`T (ADCACQ) = (ADC_ACQ+1) *16*T (ADC_CLK)`。假设 `ADC_CLK=24MHz`， `ADC_ACQ=0x2F`， `T(ADCACQ) = (47+1)*16*41.67ns = 2*16us=32us`。
7:1	–	–	–
0	R/W	1	ADC_PRI 本次 ADC 采样的优先级控制。 • 0: 低优先级非实时采样，采样命令会与其它非实时采样命令进入采样队列 • 1: 示高优先级实时采样，采样命令会被直接处理，不进入采样队列。

13.3.5.6. 0x014 RTP_MMSC

默认值：0x00000000			RTP 手动模式控制 (RTP Manual Mode Sample Control)
位域	类型	默认值	描述
31:24	–	–	–
23:22	R/W	0x0	VNSEL Vref– Select，Vref–选通控制，定义了 ADC 参考电压负端（Vref–）的选通控制。仅在手动模式（RTP_MODE == 0）下有效。 • 0x0: Vref–连通 GND • 0x1: Vref–连通 X– 信号 • 0x2: Vref–连通 Y– 信号 • 0x3: Vref–连通 GND
21:20	R/W	0x0	VPSEL Vref+ Select，Vref+选通控制，定义了 ADC 参考电压正端（Vref+）的选通控制，仅在手动模式（RTP_MODE = 0）下有效。 • 0x0: Vref+连通 VCC • 0x1: Vref+连通 X+信号 • 0x2: Vref+连通 Y+信号 • 0x3: Vref+连通 VCC
19:16	R/W	0x0	DRV（） RTP X/Y Drive，RTP X/Y 信号驱动控制 定义了 RTP X+/X–/Y+/Y–信号的驱动控制，高电平有效。仅在手动模式（RTP_MODE == 0）下有效。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			RTP 手动模式控制 (RTP Manual Mode Sample Control)
位域	类型	默认值	描述
			• bit3: X+，高电平 X+连通 VCC，低电平 X+断开 VCC • bit2: Y+，高电平 Y+连通 VCC，低电平 Y+断开 VCC • bit1: X–，高电平 X– 连通 GND，低电平 X– 断开 GND • bit0: Y–，高电平 Y– 连通 GND，低电平 Y– 断开 GND。
15:11	–	–	–
10:8	R/W	0x0	SCPT（ ） Sample Count per Trigger，每次触发的采样次数 仅在手动模式（RTP_MODE = 0）下有效。 本字段定义了每次采样触发之后，连续采样的次数，次数为 SCPT+1，SCPT 的有效值为 0x0~0x7，对应有效的连续采样次数是 1~8。每次触发之后采样得到的数据，都存储在 FIFO 中。
7:6	–	–	–
5:4	R/W	0	SCSEL Sample Channel Select，采样通道选通，仅在手动模式（RTP_MODE = 0）下有效。本字段定义了当前发起采样的对应通道。 • 0x0: Y–通道 • 0x1: X–通道 • 0x2: Y+通道 • 0x3: X+通道。
3:1	–	–	–
0	R/WAC	0	STRG（ ） Sample Trigger，手动采样触发，仅在手动模式（RTP_MODE = 0）下有效。 本字段为本通道采样的触发信号。由软件置位，表示启动本通道 ADC 采样，采样完成之后，本字段就会由硬件自动清零。 **注：** 一次触发有可能会包含多次采样，对应地由硬件向 ADC 接口控制器连续发出多个采样命令，连续发出采样命令的个数由 SCPT 字段设定。例如 如 SCPT = 0x7，则每次启动 STRG，硬件会连续发出 SCPT + 1 = 8 个采样命令。8 个采样命令发出后，硬件自动清零 STRG。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

13.3.5.7. 0x018 RTP_FIL

默认值：0x00000002			RTP 自动采样滤波配置 (RTP Auto Mode Filter)
位域	类型	默认值	描述
31	–	–	–
30:28	R/W	0x0	ZRUR Z Data Relative Uncertainty Range，Z 采样数据相对不确定范围，仅在自动模式 ((RTP_MODE > 0) 下有效。 本字段定义了 ZA/ZB/ZC/ZD 采样样本的相对的有效性判定范围。 • 0x0: 数据相对范围判定功能禁止 • 0x1: 不超过 1/8 • 0x2: 不超过 1/16 • 0x3: 不超过 1/32 • 0x4: 不超过 1/64 • 0x5: 不超过 1/128 • 0x6: 不超过 1/256 • 0x7: 不超过 1/512
27	–	–	–
26:24	R/W	0x0	ZAUR Z Data Absolute Uncertainty Range，Z 采样数据绝对不确定范围，仅在自动模式 (RTP_MODE > 0) 下有效。 本字段定义了 ZA/ZB/ZC/ZD 采样样本的绝对的有效性判断范围。 • 0x0: 数据绝对范围判定功能禁止； • 0x1: 不超过 29 • 0x2: 不超过 28 • 0x3: 不超过 27 • 0x4: 不超过 26 • 0x5: 不超过 25 • 0x6: 不超过 24 • 0x7: 不超过 23
23	–	–	–
22:20	R/W	0x0	XYRUR（ ） X/Y Data Relative Uncertainty Range，X/Y 采样数据相对不确定范围，仅在自动模式 (RTP_MODE > 0) 下有效

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000002			RTP 自动采样滤波配置 (RTP Auto Mode Filter)
位域	**类型**	**默认值**	**描述**
			本字段定义了 XP/XN/YP/YN 采样样本的相对的有效性判定范围。 • 0x0: 数据相对范围判定功能禁止 • 0x1: 不超过 1/8 • 0x2: 不超过 1/16 • 0x3: 不超过 1/32 • 0x4: 不超过 1/64 • 0x5: 不超过 1/128 • 0x6: 不超过 1/256 • 0x7: 不超过 1/512
19	–	–	–
18:16	R/W	0x0	XYAUR（ ） X/Y Data Absolute Uncertainty Range，X/Y 采样数据绝对不确定范围，仅在自动模式 (RTP_MODE > 0) 下有效。 本字段定义了 XP/XN/YP/YN 采样样本的绝对的有效性判断范围。 • 0x0: 数据绝对范围判定功能禁止 • 0x1: 不超过 29 • 0x2: 不超过 28 • 0x3: 不超过 27 • 0x4: 不超过 26 • 0x5: 不超过 25 • 0x6: 不超过 24 • 0x7: 不超过 23
15:2	–	–	–
1:0	R/W	0x2	FIL Filter，采样数据滤波器配置，仅在自动模式（RTP_MODE > 0）下有效，定义了 RTP 采样数据滤波配置。 • 0x0：1/1，无滤波 • 0x1：2/4，连续采样 4 次，去掉最大最小各 1 个，剩下 2 个进行算数平均 • 0x2：4/6，连续采样 6 次，去掉最大最小各 1 个，剩下 4 个进行算数平均 • 0x3：4/8，连续采样 8 次，去掉最大最小各 2 个，剩下 4 个进行算数平均 本字段也规定了自动模式下每次的连续采样数。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

13.3.5.8. 0x01C RTP_AMSC

默认值：0x0024A000			RTP 自动模式采样控制 (RTP Auto Mode Sample Control)
位域	类型	默认值	描述
31:28	–	–	–
27:12	R/W	0x024A	PSI Periodic Sample Interval，周期性采样间隔，仅在自动模式（RTP_MODE > 0）下有效。 自动模式周期性采样的采样间隔，采样间隔为 `PSI*4096*TPCLK`。举例，假设 `PCLK=24MHz`，`PSI = 0x24A = 586`。采样周期 `T (PSI) = 586*4096/24000000 = 0.1s = 100ms`。 本字段为 0 表示连续采样，两次采样循环之间没有 Delay。 **使用限制：如果已使用二级去抖功能，则需要满足 `PSI*16≥(DET_DEB +1)*DET_DEBDC`**
11:3	–	–	–
2	R	0x0	RTP BUSY 状态 仅在自动模式（RTP_MODE > 0）下有效。 • 0: 空闲 • 1: 忙碌
1	R/W	0	PSEN Periodic Sample Enable，周期性循环采样使能信号，仅在自动模式（RTP_MODE > 0）下有效。 • 0: 停止周期性循环采样 • 1:启动周期性循环采样。 周期性循环采样使能信号，由软件置位清零。周期性采样流程参考前文描述。软件清 0：完成本周期循环采样之后才停止采样。
0	R/WAC	0	SSEN Single Sample Enable，单循环采样使能，仅在自动模式（RTP_MODE > 0）下有效。 • 0：单循环采样完成。 • 1：启动单循环采样。 单循环完成之后，本字段就会由硬件自动清零。单循环采样的定义是指一个完整自动采样流程，流程涵盖多个采样模式，具体参考 MODE 定义#unique_215/unique_215_Connect_42_table_ezb_2dc_bzb。 **单循环模式下，必须开启屏蔽按压检测信号（PDET_BYP=1）**

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

13.3.5.9. 0x020 RTP_FCR

默认值：0x00000000			RTP FIFO 控制 (RTP FIFO Control Register)
位域	类型	默认值	描述
31:29	–	–	–
28:24	RO	0x0	FIFO_DCNT FIFO Data Count，FIFO 有效数据量 当前 FIFO 有效数据量，0x0~0x10 为合法值。
23:19	–	–	–
18	R/W1C	0	FIFO_UFS FIFO Underflow Status，FIFO 下溢出状态 当前 FIFO 下溢出状态，高电平有效。由硬件置位，并由软件写 1 清零。 当本字段有效，且 FIFO_UFIE 有效，则会导致 RTP_INTR. FIFO_FLG 置位。
17	R/W1C	0	FIFO_OFS FIFO Overflow Status，FIFO 上溢出状态 当前 FIFO 上溢出状态，高电平有效。由硬件置位，并由软件写 1 清零。当本字段有效，且 FIFO_OFIE 有效，则会导致 RTP_INTR. FIFO_FLG 置位。
16:13	–	–	–
12:8	R/W	0x0	FIFO_DRTH FIFO Data Ready Threshold，FIFO 数据有效阈值 FIFO 数据有效的阈值，当 FIFO 的数据量达到了有效阈值后， RTP_INTR. DRDY_FLG 标志位置位，如果此时 INTEN 也有效，就会产生中断。 • 0x0: 不设置有效阈值，不会产生 DRDY_FLG 标志。 • 0x1~0x10: 有效阈值范围，当 FIFO 数据量达到了对应数值后，就会置位 DRDY_FLG 标志。 • 0x11~0x1F: 非法配置，效果等同于 0x0。
7:3	–	–	–
2	R/W	0	FIFO_UFIE FIFO Underflow Interrupt Enable，FIFO 下溢出中断使能 FIFO 下溢出中断使能信号，高电平有效。 当本字段有效，且 FIFO_UFS 有效，则会导致 RTP_INTR. FIFO_FLG 置位。
1	R/W	0	FIFO_OFIE FIFO Overflow Interrupt Enable，FIFO 上溢出中断使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			RTP FIFO 控制 (RTP FIFO Control Register)
位域	类型	默认值	描述
			FIFO 上溢出中断使能信号，高电平有效。 当本字段有效，且 FIFO_OFS 有效，则会导致 RTP_INTR. FIFO_FLG 置位。
0	R/WAC	0	FIFO_FLUSH FIFO Flush，FIFO 清空操作 由软件置位硬件清零。软件置位启动 FIFO 清空操作，FIFO 清空操作完成后，由硬件清零。

13.3.5.10. 0x024 RTP_DATA

默认值：0x00000000			RTP 数据 (RTP Data)
位域	类型	默认值	描述
21	RO	0x0	超限标志
20	RO	0x0	抬起标志
17:16	RO	0x0	CHN（ ） Channel Number，采样数据的对应通道号 当前返回数据的对应通道号，表示当前数据（DATA）来自该通道，多用于 Debug 或者容错设计。具体对应如下： • 0x0: Y–通道 • 0x1: X–通道 • 0x2: Y+通道 • 0x3: X+通道
11:0	RO	0x0	DATA Sample Data，采样数据 采样数据返回值。 • 对应手动模式（RTP_MODE == 0），本 FIFO 数据是一个通道的多次采样值，未经过滤波。 • 对应自动模式 1（RTP_MODE == 1），本 FIFO 数据的顺序是 YN，均为滤波后的数值。 • 对应自动模式 2（RTP_MODE == 2），本 FIFO 数据的顺序是 ZB，均为滤波后的数值。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			RTP 数据 (RTP Data)
位域	类型	默认值	描述
			• 对应自动模式 3（RTP_MODE == 3），本 FIFO 数据的顺序是 ZB，均为滤波后的数值。 • 对应自动模式 4（RTP_MODE == 4），本 FIFO 数据的顺序是 ZD，均为滤波后的数值。 如果 SCI_FLG=1，所有采样数据滤波有效，但是不足对应采样模式所需的数据量； 如果 DOUR_FLG=1，采样数据前面 DCNT-3 个数据有效，倒数第三个数据为超限数据滤波后的值，倒数第二个数据为超限数据组的最大值，最后一个数据为超限数据组的最小值。

13.3.5.11. 0x028 RTP_DELAY

默认值：0x000F0001			RTP 延时控制 (RTP Delay)
位域	类型	默认值	描述
31:16	R/W	0x0F	Precharge Delay 按压检测预充电延时，仅自动模式有效 按压检测预充电延时，计数单位为 TPCLK。当本字段为 0（即 0x0000）时，无延时。本字段定义的延时时间范围为 `0~ (2^16-1) *TPCLK`。
15:0	R/W	0x01	Stabilization_Delay 驱动与 ADC 采样之间延时（仅自动模式有效） 驱动延时，计数单位为 TPCLK。当本字段为 0（即 0x0000）时，无延时。本字段定义的延时时间范围为 `0~ (2^16-1) *256*TPCLK`。举例，当 `PCLK =24MHz, TPCLK =41.67ns, 256*TPCLK =10.67us`，假设本字段的配置 `RDET_DEB =0x100 =256`，抬起去抖时间为 `256*10.67us =2.73ms`。

13.3.5.12. 0xFFC VERSION

默认值：0x00000100			RTP 版本 (RTP Version)
位域	类型	默认值	描述
31:0	RO	0x00000100	Version（模块版本号） 采用 BCD 码显示，V1.0

13.4. Thermal Sensor (THS)

THS 温度传感器模块，可以管理和读取芯片内置温度传感器，并基于读取的温度值进行高低温警告功能。

THS 温度传感器模块通过 ADC 来获取内部温度传感器温度值，因此在启动之前，需要先启动 ADC Interface Module (ADCIM) 模块。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

13.4.1. 特性说明

- 最多支持一路温度传感器
- ADC 与温度传感器捕获时间可配
- 支持单次读取和周期读取两种模式
- 支持一次多点读取，并进行算术平均滤波
- 基于周期读取模式时，支持高/ 低温报警（产生中断）和过温保护功能（产生中断/复位）

13.4.2. 原理框图

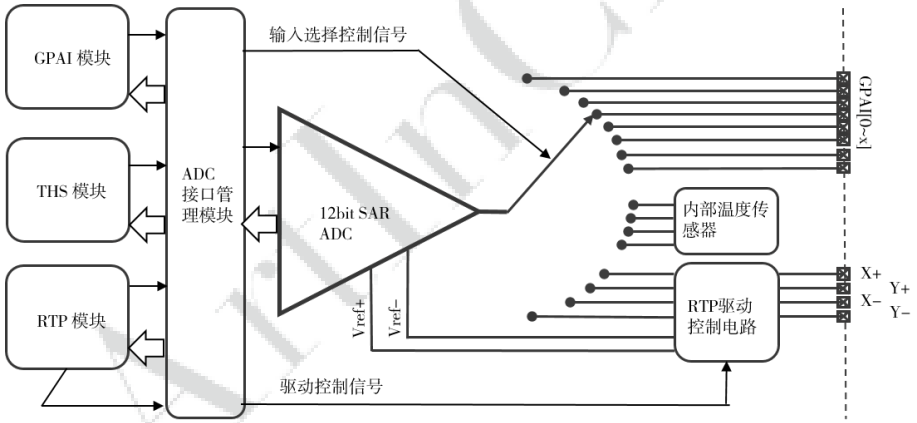


图 13–23 THS 接口模块与 ADC 连接关系

13.4.3. 功能描述

13.4.3.1. 中断与过温保护复位产生机制

THS 模块的中断输出，以及过温保护系统复位输出如下图所示。

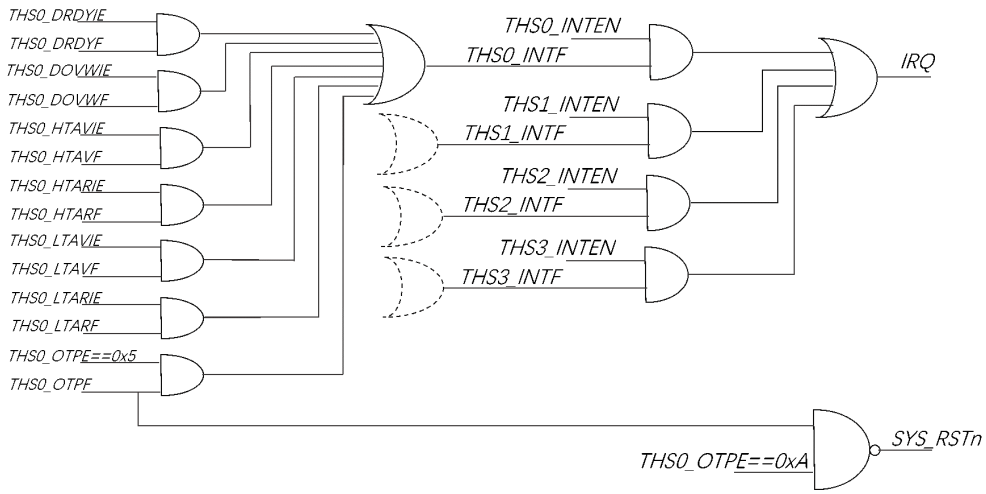


图 13–24 中断，以及过温保护系统复位输出

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

13.4.3.2. 高/ 低温报警的双阈值描述

由于温度在边界处反复低幅度变动时，单阈值容易产生频繁报警的问题，因此高/ 低温报警采用了双阈值，详情如下所示：

- 高温报警：
  - 高温报警有效阈值 (HTAVV)
  - 高温报警解除阈值 (HTARV)
- 低温报警：
  - 低温报警有效阈值 (LTAVV)
  - 低温报警解除阈值 (LTARV)

双阈值的使用示意图如下：

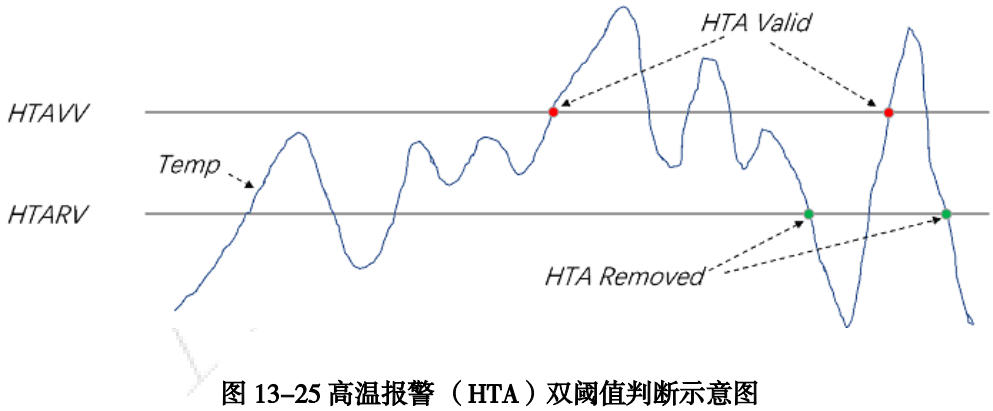


图 13–25 高温报警（HTA）双阈值判断示意图

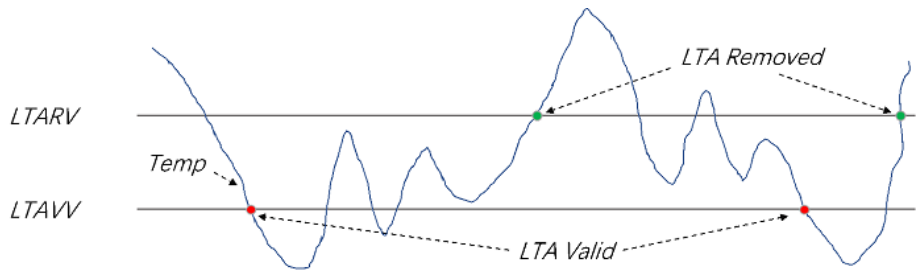


图 13–26 低温报警（LTA）双阈值判断示意图

注：

- 通常情况下，LTAVV < LTARV < HTARV < HTAVV。
- 在周期模式下，仅对采样后取平均值进行阀值判断，而不对原始数据进行阀值判断，避免某一次误触发。
- 温度越高，输入电压越高，ths 值越大

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

13.4.4. 编程指南

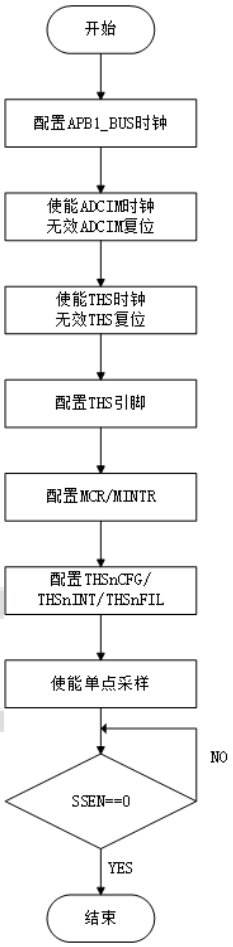


图 13–27 单点采样

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

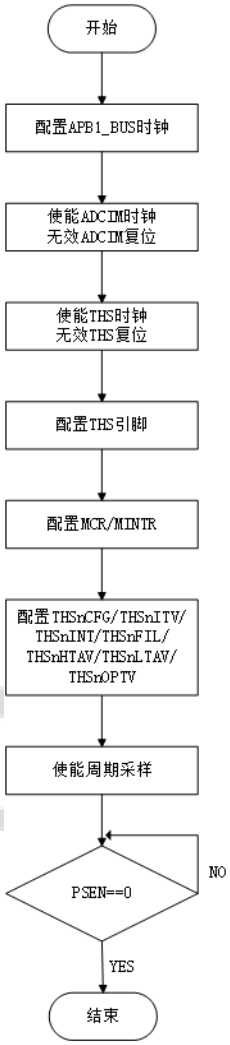


图 13–28 周期采样

13.4.5. 寄存器列表

表 13–4 寄存器列表

偏移地址	缩写	寄存器描述	章节
0x000	MCR	THS 模块控制	0x000 MCR
0x004	MINTR	THS 中断控制与状态	0x004 MINTR
0x100+n*0x20	THSn_CFG	THS 通道控制配置	0x100+n*0x20 THSn_CFG
0x104+n*0x20	THSn_ITV	THS 通道采样间隔配置	0x104+n*0x20 THSn_ITV
0x108+n*0x20	THSn_FIL	THS 采样滤波配置	0x108+n*0x20 THSn_FIL
0x10C+n*0x20	THSn_DATA	THS 数据	0x10C+n*0x20 THSn_DATA

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

表 13–4 寄存器列表 (续)

偏移地址	缩写	寄存器描述	章节
0x110+n*0x20	THSn_INT	THS 中断状态	0x110+n*0x20 THSn_INT
0x114+n*0x20	THSn_HTAV	THS 高温报警阈值设置	0x114+n*0x20 THSn_HTAV
0x118+n*0x20	THSn_LTAV	THS 低温报警阈值设置	0x118+n*0x20 THSn_LTAV
0x11C+n*0x20	THSn_OTPV	THS 过温保护阈值设置	0x11C+n*0x20 THSn_OTPV
0xFFC	VERSION	THS 版本号	0xFFC VERSION

13.4.6. 寄存器描述

13.4.6.1. 0x000 MCR

默认值：0x00002F30			THS 模块控制 (THS Module Control Register)
位域	类型	默认值	描述
31:22	–	–	–
21	R/W	0	THS1_EN THS1 Enable, THS1 控制器使能 使用时需与 SEN1 同时使能或禁止 • 0: THS1 禁止 • 1: THS1 使能
20	–	–	THS0_EN THS0 Enable, THS0 控制器使能 使用时需与 SEN1 同时使能或禁止 • 0: THS0 禁止 • 1: THS0 使能
19:18	–	–	–
17	R/W	0	SEN1_EN SEN1 Enable, SEN1 控制器使能 使用时需与 SEN1 同时使能或禁止 • 0: SEN1 禁止 • 1: SEN1 使能

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00002F30			THS 模块控制 (THS Module Control Register)
位域	类型	默认值	描述
16	R/W	0	SEN0_EN SEN0 Enable, SEN0 控制器使能 使用时需与 SEN1 同时使能或禁止 • 0: SEN0 禁止 • 1: SEN0 使能
15:8	R/W	0x2F	THS_ACQ THS Acquisition Time, THS 捕获时间 温度传感器的捕获时间长度，单位为 ADC_CLK 的周期数。`T (THS_ACQ) = (THS_ACQ +1) *T(ADC_CLK)`。假设 `ADC_CLK=24 MHz,THS_ACQ=0x2F`， `T(THS_ACQ) = (47+1)*41.67 ns = 2 us`。
7:6	–	–	–
5:4	RO	0x3	THS_MAX THS Maximum Number , THS 最大编号 本字段规定了 THSn 中 n 的最大编号，n+1 即为本模块实际使用的温度传感器数量。本字段小于等于 3，即传感器数量最少为 1 个，最多为 4 个
3:1	–	–	–
0	R/W	0	EN（模块使能） • 0: 本模块被禁止，本模块停止工作 • 1: 本模块被使能

13.4.6.2. 0x004 MINTR

默认值：0x00000000			THS 中断控制与状态 (THS Module Interrupt Register)
位域	类型	默认值	描述
31:18	–	–	–
17	RO	0	THS1_INTF THS1 Interrupt Flag，THS1 中断标志，只可读。 只要清除了相应 THS 的中断标志，就会自动清零 • 0: THS1 无中断请求 • 1: THS1 有中断请求

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			THS 中断控制与状态 (THS Module Interrupt Register)
位域	类型	默认值	描述
16	RO	0	THS0_INTF（ ） THS0 Interrupt Flag，THS0 中断标志，只可读。 只要清除了相应 THS 的中断标志，就会自动清零 • 0: THS0 无中断请求 • 1: THS0 有中断请求
15:2	–	–	–
1	R/W	0	THS1_INTEN THS1 Interrupt Enable，THS1 中断使能信号 • 0：禁止 THS1 中断 • 1：使能 THS1 中断
0	R/W	0	THS0_INTEN THS0 Interrupt Enable，THS0 中断使能信号 • 0：禁止 THS0 中断 • 1：使能 THS0 中断

13.4.6.3. 0x100+n*0x20 THSn_CFG

(2)

默认值：0x000(n+12)2F00			THS 通道控制配置 (THS Channel N Config)
位域	类型	默认值	描述
31:28	–	0x0	–
27	R/W	0x0	Inverted Select • 0：取反 • 1：不变
26:20	–	–	–
19:16	RO	n+12	ADC_CHN ADC Channel Number，ADC 通道号

(2) n = 0~1 为 ths 通道号

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x000(n+12)2F00			THS 通道控制配置 (THS Channel N Config)
位域	类型	默认值	描述
			显示系统分配的 ADC 通道号，硬件会根据此通道号匹配 ADC 并获取返回的数据，软件驱动不需要使用，仅作 Debug 用。
15:8	R/W	0x2F	ADC_ACQ ADC Acquisition Time，ADC 捕获时间 ADC 通道的捕获时间长度。 • 计数值单位为 ADC_CLK 的周期数，T(ADC_ACQ) = (ADC_ACQ+1) * T(ADC_CLK) • 假设 ADC_CLK = 24 MHz，ADC_ACQ = 0x2F，T (ADC_ACQ) = (47 + 1) * 41.67 ns = 2 us。
7:5	–	–	–
4	R/W	0x0	ADC_PRI ADC Priority，ADC 采样的优先级控制 • 0：低优先级非实时采样，采样命令与其他非实时采样命令进入采样队列。 • 1：高优先级实时采样，采样命令会被直接处理，不进入采样队列。
3	–	–	–
2	R	0x0	THS Channel State，工作状态： • 0: 当前 THS 通道处于空闲状态 • 1: 当前 THS 通道处于忙碌状态
1	R/W	0x0	PSEN Periodic Sample Enable，温度传感器周期采样使能。 • 0：停止相应温度传感器周期采样。 • 1：启动相应温度传感器周期采样。 采样间隔由 THSn_ITV 寄存器设定。如果在软件清零之前硬件已经启动了一次新的采样，新采样会继续完成，并最终返回数据。 **注：** 不允许同时使能周期采样与单次采样，即 SSEN 与 PSEN 不能同时有效。
0	R/WAC	0x0	SSEN Single Sample Enable，温度传感器单点采样使能。 • 0：相应传感器的温度单点采样已完成。 • 1：启动相应传感器的温度单点采样。 由硬件自动清零，表示相应传感器的温度单点采样已完成。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x000(n+12)2F00			THS 通道控制配置 (THS Channel N Config)
位域	类型	默认值	描述
			注： 尽管描述为单点采样，但为了避免单次采样误差，可能需要通过单点多次采样后的数据进行算术平均，得到一个相对稳定的温度值，因此单点采样实际上会包含多次采样。对应的由硬件向 ADC 接口控制器连续发出多个采样命令。连续发出采样命令的个数由滤波器配置 FIL 设定。例如滤波器配置为 0x1（2 点平均），则每次启动 SSEN，硬件会连续发出 2 个采样命令，接收到 2 个温度数据之后，硬件自动清零 SSEN。

13.4.6.4. 0x104+n*0x20 THSn_ITV

(3)

默认值：0x016EFFFF			THS 通道采样间隔配置 (THS Channel N Interval)
位域	类型	默认值	描述
31:16	R/W	0x016E	ITV Interval，采样间隔 周期采样的采样间隔，单位为 Multiple * TPCLK，即采样间隔 = ITV* Multiple * TPCLK。 假设 PCLK = 24 MHz，ITV = 0x16E = 366，DIV=65535，采样间隔 = 366 * (65535 + 1) * 41.67 ns = 1 s 注： 不可为 0。硬件自动保护：当软件配置为 0 时，效果等同于 1。
15:0	R/W	0xFFFF	DIV Divide CLK，分频时钟 时钟 (div_clk) 分频系数，单位为 PCLKT (div_clk) = (DIV+1) *T (PCLK)

13.4.6.5. 0x108+n*0x20 THSn_FIL

(4)

(3) n = 0~1 为 ths 通道号

(4) n = 0~1 为 ths 通道号

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			THS 采样滤波配置 (THS Channel Filter)
位域	类型	默认值	描述
31:2	–	–	–
1:0	R/W	0x0	FIL Filter，滤波器配置 温度传感器采样数据算数平均滤波器的输入样本数。 • 0x0：无滤波 • 0x1：2 • 0x2：4 • 0x3：8 此字段同时定义了单次采样和周期采样每次的连续采样命令数量（采样点数），控制器接收对应的采样点数据后，进行算数平均，并把平均值通过本传感器的 THSn_DATA 寄存器返回。

13.4.6.6. 0x10C+n*0x20 THSn_DATA

(5)

默认值：0x00000000			THS 数据 (THS Channel N Data)
位域	类型	默认值	描述
31:14	–	–	–
13:0	RO	0x000	DATA ADC DATA,，ADC 采样数据 温度传感器的 ADC 采样数据结果。 • 如果 THSn_FIL.FIL 为 0x0，直接返回 ADC 的采样值。 • 否则返回的是多个采样点的算数平均值。 **注：** 模块无数据 FIFO，本寄存器只能暂存一次的采样结果，如果未被及时读取，则会被后来新的采样数据直接覆盖。

13.4.6.7. 0x110+n*0x20 THSn_INT

(6)

(5) n = 0~1 为 ths 通道号

(6) n = 0~1 为 ths 通道号

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			THS 中断状态 (THS Channel N Interrupt)
位域	**类型**	**默认值**	**描述**
31:29	–	–	–
28	R/W1C	0x0	OTPF Over–Temperature–Protected Flag，过温保护标志，仅周期采样模式（THSn_CFG. PSEN=1）时有效。 操作方式为硬件置位，软件写 1 清零。当温度传感器采样值超过 THSn_OTPV.OTPV，就会置为 1。 • 如果 OTPE=中断使能，则产生中断信号。 • 如果 OTPE=复位使能，则产生系统复位信号
27	R/W1C	0x0	LTARF Low–Temperature–Alarm Removed Flag，低温警告解除标志 • 仅周期采样模式 (THSn_CFG. PSEN = 1) 时有效。 • 硬件置位，软件写 1 清零。 • 当温度传感器采样值从低于 THSn_LTAV.LTAVV 值到高于 THSn_LTAV.LTARV 值，会置为 1。
26	R/W1C	0x0	LTAVF Low–Temperature–Alarm Valid Flag，低温警告有效标志 • 仅周期采样模式 (THSn_CFG. PSEN = 1) 时有效。 • 硬件置位，软件写 1 清零。 • 当温度传感器采样值从高于 THSn_LTAV.LTARV 值到低于 THSn_LTAV.LTAVV 值，会置为 1。
25	R/W1C	0x0	HTARF High–Temperature–Alarm Removed Flag，高温警告解除标志 • 仅周期采样模式（THSn_CFG. PSEN = 1）时有效。 • 硬件置位，软件写 1 清零。 • 当温度传感器采样值从高于 THSn_HTAV.HTAVV 值到低于 THSn_HTAV.HTARV 值，会置为 1。
24	R/W1C	0x0	HTAVF High–Temperature–Alarm Valid Flag，高温警告有效标志 • 仅周期采样模式（THSn_CFG. PSEN = 1）时有效。 • 硬件置位，软件写 1 清零。 • 当温度传感器采样值从低于 THSn_HTAV.HTARV 值到高于 THSn_HTAV.HTAVV 值，会置为 1。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			THS 中断状态 (THS Channel N Interrupt)
位域	类型	默认值	描述
23:18	–	–	–
17	R/W1C	0x0	DOVWF Data Overwrite Flag，数据覆盖标志 硬件置位，软件写 1 清零。当 DRDYF = 1（软件未清零数据有效标志），而且又有新的有效数据时，硬件置为 1，表示数据寄存器由于软件未及时读取，产生了上次有效数据被覆盖的情况。
16	R/W1C	0x0	DRDYF Data Ready Flag，数据有效标志 硬件置位，软件写 1 清零。当 THSn_DATA 寄存器有新的有效数据时，硬件置为 1，软件读取数据后，由软件操作清除。
15:12	–	–	OTPIRE Over–Temperature–Protection Interrupt or Reset Enable，过温保护中断或复位使能 使能之后，当温度传感器采样值超过 THSn_OTPV.OTPV，THS 模块会输出中断或者系统复位（SYS_RSTn），从而引起 CPU 中断或者系统复位。 • 0x5: 过温保护中断使能 • 0xA: 过温保护复位使能 • Others: 过温保护禁止
11	R/W	0x0	LTARIE Low–Temperature–Alarm Removed Interrupt Enable，低温警告解除中断使能 当使能时，低温警告解除标志 (LTARF) 的状态就会传递到温度传感器的总中断。 • 0：禁止低温警告解除中断。 • 1：使能低温警告解除中断。
10	R/W	0x0	LTAVIE Low–Temperature–Alarm Valid Interrupt Enable，低温警告有效中断使能 当使能时，低温警告有效标志 (LTAVF) 的状态就会传递到温度传感器的总中断。 • 0：禁止低温警告有效中断。 • 1：使能低温警告有效中断。
9	R/W	0x0	HTARIE High–Temperature–Alarm Removed Interrupt Enable，高温警告解除中断使能 当使能时，高温警告解除标志 (HTARF) 的状态就会传递到温度传感器的总中断。

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			THS 中断状态 (THS Channel N Interrupt)
位域	类型	默认值	描述
			• 0：禁止高温警告解除中断。 • 1：使能高温警告解除中断。
8	R/W	0x0	HTAVIE High-Temperature-Alarm Valid Interrupt Enable，高温警告有效中断使能 当使能时，高温警告有效标志 (HTAVF) 的状态就会传递到温度传感器的总中断。 • 0：禁止高温警告有效中断。 • 1：使能高温警告有效中断。
7:2	–	–	–
1	R/W	0x0	DOVWIE Data Overwrite Interrupt Enable，数据覆盖中断使能 当使能时，数据覆盖标志 (DOVWF) 的状态就会传递到温度传感器的总中断。 • 0：禁止数据覆盖中断。 • 1：使能数据覆盖中断。
0	R/W	0x0	DRDYIE Data Ready Interrupt Enable，数据有效中断使能 当使能时，数据有效标志 (DRDYF) 的状态就会传递到温度传感器的总中断。 • 0：禁止数据有效中断。 • 1：使能数据有效中断。

13.4.6.8. 0x114+n*0x20 THSn_HTAV

默认值：0x00000000			THS 高温报警阈值设置 (Channel N High-Temperature Alarm Value)
位域	类型	默认值	描述
31	R/W	0	HTAE High-Temperature-Alarm Enable，高温警报使能 • 0：禁止高温警报功能，高温警报阈值无效。 • 1：使能高温警报功能，高温警报阈值有效。
30:28	–	–	–
27:16	R/W	0x000	HTARV High-Temperature-Alarm Remove Value，高温警报解除阈值

Copyright © 2023–2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			THS 高温报警阈值设置 (Channel N High-Temperature Alarm Value)
位域	类型	默认值	描述
			当温度从高于高温警报有效阈值 (HTAVV) 的数值，降到高温警报解除阈值（HTARV）以下，高温警报解除标志（HTARF）就会被硬件置位。 注： HTARV < HTAVV。
15:12	–	–	–
11:0	R/W	0x000	HTAVV High-Temperature-Alarm Valid Value，高温警报有效阈值 当温度从低于高温警报解除阈值（HTARV）的数值，升到高温警报有效阈值 (HTAVV) 以上，高温警报有效标志 (HTAVF) 就会被硬件置位。 注： HTARV < HTAVV。

13.4.6.9. 0x118+n*0x20 THSn_LTAV

默认值：0x00000000			THS 低温报警阈值设置 (Channel N Low-Temperature Alarm Value)
位域	类型	默认值	描述
31	R/W	0	LTAE Low-Temperature-Alarm Enable，低温警报使能 • 0：禁止低温警报功能，低温警报阈值无效。 • 1：使能低温警报功能，低温警报阈值有效。
30:28	–	–	–
27:16	R/W	0x000	LTARV Low-Temperature-Alarm Remove Value，低温警报解除阈值 当温度从低于低温警报有效阈值 (LTAVV) 的数值，升到低温警报解除阈值 (LTARV) 以上，低温警报解除标志 (LTARF) 就会被硬件置位。 注： LTAVV < LTARV。
15:12	–	–	–

Copyright © 2023-2025 ArtInChip Technology Co., Ltd.
All rights reserved

默认值：0x00000000			THS 低温报警阈值设置 (Channel N Low-Temperature Alarm Value)
位域	类型	默认值	描述
11:0	R/W	0x000	LTAVV Low-Temperature-Alarm Valid Value，低温警报有效阈值 当温度从高于低温警报解除阈值 (LTARV) 的数值，降到低温警报有效阈值 (LTAVV) 以下，低温警报有效标志（LTAVF）就会被硬件置位。 **注：** LTAVV < LTARV。

13.4.6.10. 0x11C+n*0x20 THSn_OTPV

(7)

默认值：0x00000000			THS 过温保护阈值设置 (Channel N Over Temperature Protect Value)
位域	类型	默认值	描述
31	R/W	0	OTPE 过温保护使能 • 0：禁止过温保护功能，过温保护阈值无效。 • 1：使能过温保护功能，过温保护阈值有效。
30:12	–	–	–
11:0	R/W	0x000	OTPV 过温保护阈值，大于高温阀值 当温度上升到过温保护阈值 (OTPV) 以上，过温保护标志 (OTPF) 就会被硬件置位。

13.4.6.11. 0xFFC VERSION

默认值：0x00000200			THS 版本号 (THS Version)
位域	类型	默认值	描述
31:0	RO	0x00000200	Version（模块版本号） 采用 BCD 码显示，V2.0

(7) n = 0~1 为 ths 通道号

Copyright © 2023-2025 ArtInChip Technology Co., Ltd.
All rights reserved